

Application Development with C#

Book 2

Train with Ravi Series

(Shopping.Dateorwed.com Web Application)

Ravi Kumar Paramkusam

Prologue

A client approached me to develop a Shopping website. Then he asked me to convert into a Wedding Shopping portal. During development I always shared the application with him for approval. He did not comment on the developed part but asked for modifications. Then at the final submission, he wanted to change the css. His behaviour has not changes, instead of commenting on the developed part, he was asking for new additions. Then he said he can't pay for further developments. So, I gave him the completed part and closed his contract. I worked the application as part of my DateOrWed.com extended application as Shopping.dateorwed.com and also included services for Wedding.

So this way this application took shape. So, check it out.

Chapter 1

The Idea:

To develop a web application for catering to the needs of those who want to Shop for wedding.

Requirements of the Application:

To be developed in N Tier.

Technologies to be used:

Primary coding in C# using Visual Studio 2019 minimum for developing chatting.

Database MS SQL Server, possibly Azure or AWS cloud databases

Framework: Entity Framework

Plan of Action:

Develop Database Tables and Procedures

Create Entity classes to transfer data between Application, Business Class and Data Base.

Drop database objects in the Data Tier using entity framework.

Develop business logic in LINQ to retrieve the data and update it.

Create referenced .dlls of these C# projects and copy them to the application folder.

Finally Develop web applications using html, asp.net and C#.

Chapter 2

First download Visual Studio from Microsoft website. The latest community version is available for free.

After installation of the Visual Studio, from the tools menu select Manage NuGet packages for solutions. Browse and search for Entity Framework.

Or Open the package manager console and type the following:

```
Install-Package Microsoft.EntityFrameworkCore.Tools
```

You can install SQL Server from the Microsoft website with Management tools.

After Installing the Sqlserver and opening the management console, create a new database. The database script is available at this location on Dropbox, or copy it from below and run it in the script window.

Link for downloading Database script as below:

<https://www.dropbox.com/scl/fi/8a17jmml2cvplqab32jon/dsscript.sql?rlkey=jao3l2kui4dzgwuujijlq7u65&st=a5y2crft&dl=0>

```
CREATE TABLE [dbo].[Product](
  [ProductId] [numeric](10, 0) NOT NULL,
  [ProductName] [varchar](250) NULL,
  [ProductGroupCategory] [int] NOT NULL,
  [ProductGroupId] [int] NOT NULL,
  [DealerId] [int] NOT NULL,
  [Price] [numeric](10, 2) NULL,
  [CurrencyId] [int] NULL,
  [AnalysisCode] [varchar](10) NULL,
  [CreatedOn] [smalldatetime] NOT NULL,
  [UpdatedOn] [smalldatetime] NOT NULL,
  [UserId] [int] NOT NULL,
  [PrId] [varchar](50) NULL,
  [MinOrder] [int] NULL,
  [DelTime] [int] NULL,
  [lbtnimage] [varchar](max) NULL,
  [imageurl] [varchar](max) NULL,
  PRIMARY KEY CLUSTERED
(
  [ProductId] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
CREATE TABLE [dbo].[ProductCat](
  [PCatId] [int] NOT NULL,
  [PcatDesc] [varchar](255) NULL,
  [CreatedOn] [datetime] NULL,
  [UpdatedOn] [datetime] NULL,
  [UserId] [float] NULL,
  [lbtnimage] [varchar](50) NULL,
  PRIMARY KEY CLUSTERED
(
  [PCatId] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
CREATE TABLE [dbo].[ProductGroup](
  [pgid] [int] NOT NULL,
  [PCatId] [int] NOT NULL,
```

```

[pgroupname] [varchar](50) NOT NULL,
[pgroupdescription] [varchar](max) NOT NULL,
[isActive] [bit] NOT NULL,
[createdOn] [datetime] NULL,
[lbtnimage] [varchar](50) NULL,
[UserId] [int] NULL,
CONSTRAINT [PK__ProductGroupId] PRIMARY KEY CLUSTERED
(
    [pgid] ASC,
    [PCatId] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
CREATE view [dbo].[vProduct] as
SELECT [ProductId]
    ,ProductGroupId
    ,[ProductName]
FROM [MyLaadBazarDb].[dbo].[Product] union all
SELECT [pgid]
    ,[PCatId]
    ,[pgroupname]
FROM [MyLaadBazarDb].[dbo].[ProductGroup] union all
SELECT [PCatId],id=0
    ,[PcatDesc]
FROM [MyLaadBazarDb].[dbo].[ProductCat]
CREATE TABLE [dbo].[tblPages](
    [PageId] [int] NOT NULL,
    [PageName] [varchar](100) NULL,
    [PageUrl] [varchar](max) NULL,
    [PageCode] [varchar](max) NULL,
    [ParentId] [int] NULL,
    [Priority] [int] NULL,
    [level] [int] NULL,
    [isMenuItem] [int] NULL,
    CONSTRAINT [PK_tblPages] PRIMARY KEY CLUSTERED
(
    [PageId] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
CREATE TABLE [dbo].[tblPermissionGroup](
    [GroupId] [int] NOT NULL,
    [GroupName] [varchar](100) NULL,
    [Status] [bit] NULL,
    [UserId] [int] NULL,
    [RoleId] [int] NULL,
    CONSTRAINT [PK_tblPermissionGroup_1] PRIMARY KEY CLUSTERED
(
    [GroupId] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
CREATE VIEW [dbo].[tblGrpMapping] AS
SELECT TP.PageId + tPG.GroupId AS MapId, tPG.GroupId, TP.PageId, 1 AS [view], 1
AS [add], 1 AS Edit, 1 AS [Delete], '2018/01/01' AS CreatedOn

```

```

FROM      dbo.tblPages AS TP CROSS JOIN
          dbo.tblPermissionGroup AS tPG
--union all
--SELECT    TP.Pageld +'0' AS MapId, GroupId=0, TP.Pageld, 1 AS [view], 1 AS [add], 1
AS Edit, 1 AS [Delete], '2018/01/01' AS CreatedOn
--FROM      dbo.tblPages AS TP
CREATE TABLE [dbo].[ActionTask](
  [Id] [int] IDENTITY(1,1) NOT NULL,
  [QueueId] [int] NULL,
  [OwnedById] [int] NULL,
  [RaisedById] [int] NULL,
  [ActionedById] [int] NULL,
  [RaisedDate] [datetime] NULL,
  [ActionedDate] [datetime] NULL,
  [Status] [char](1) NOT NULL,
  [Summary] [varchar](100) NOT NULL,
  [Details] [text] NULL,
  [CallingPageld] [int] NOT NULL,
  [SubjectId] [int] NOT NULL,
  [DueDate] [datetime] NULL,
  [TaskType] [char](1) NOT NULL,
  [TaskOutcome] [bit] NOT NULL,
  [OutcomeNotes] [text] NULL,
  [CLI] [varchar](15) NULL,
  [CreatedOn] [datetime] NOT NULL
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
ALTER TABLE [dbo].[ActionTask] ADD [RelateTo] [varchar](500) NULL
ALTER TABLE [dbo].[ActionTask] ADD [RelateToURL] [varchar](max) NULL
ALTER TABLE [dbo].[ActionTask] ADD [atid] [uniqueidentifier] NULL
ALTER TABLE [dbo].[ActionTask] ADD [subid] [uniqueidentifier] NULL
CONSTRAINT [PK__ActionTa__3214EC07367C1819] PRIMARY KEY CLUSTERED
(
  [Id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
CREATE view [dbo].[v_QueueByUser] as
  select Queueid,Total=1,OverDue=(CASE when datediff(d,DueDate,getdate())>0 then 1
else 0 end),
  Outstanding=(case STATUS when 'A' then 1 else 0 end),
  Unassigned=(case when RaisedById=0 then 1 else 0 end),
STATUS,OwnedById,RaisedById ,ActionedById
from ActionTask
where status<>'C'
CREATE TABLE [dbo].[ActionQueue](
  [Id] [int] IDENTITY(1,1) NOT NULL,
  [Name] [varchar](50) NOT NULL,
  [AutoAssign] [bit] NOT NULL,
  [IsSystemQueue] [bit] NOT NULL,
  [IsActive] [bit] NOT NULL,
  [CreatedOn] [datetime] NOT NULL,
  [aqid] [uniqueidentifier] NULL,
CONSTRAINT [PK_ActionQueue] PRIMARY KEY CLUSTERED
(
  [Id] ASC

```

```

)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
CREATE VIEW [dbo].[v_QueueListByUser]
AS
SELECT q.Id, q.Name, v.RaisedById, COUNT(v.Total) AS Total, COUNT(v.OverDue) AS
OverDue, COUNT(v.OutStanding) AS OutStanding, COUNT(v.Unassigned) AS Unassigned
FROM   dbo.v_QueueByUser AS v RIGHT OUTER JOIN
      dbo.ActionQueue AS q ON v.Queueid = q.Id
GROUP BY q.Id, q.Name, v.RaisedById
CREATE VIEW [dbo].[Active Task Queue List]
AS
SELECT   Id, Name
FROM     dbo.ActionQueue
WHERE    (IsActive = 1)
CREATE VIEW [dbo].[Active Task counts]
AS
SELECT   dbo.[Active Task Queue List].Name AS [Task Queue],
COUNT(dbo.ActionTask.Id) AS Expr1
FROM     dbo.[Active Task Queue List] INNER JOIN
      dbo.ActionTask ON dbo.[Active Task Queue List].Id =
      dbo.ActionTask.QueueId
WHERE    (dbo.ActionTask.Status = 'A')
GROUP BY dbo.[Active Task Queue List].Name
CREATE TABLE [dbo].[Brand](
  [Id] [int] IDENTITY(1,1) NOT NULL,
  [ShortCode] [varchar](3) NOT NULL,
  [BrandName] [varchar](50) NOT NULL,
  [OpalCredentialId] [int] NOT NULL,
  [DefaultInvoiceMask] [varchar](25) NOT NULL,
  [EmailInvoiceMask] [varchar](25) NOT NULL,
  [BusinessInvoiceMask] [varchar](25) NULL,
  [BusinessEmailInvoiceMask] [varchar](25) NULL,
  [FromEmailAddress] [varchar](50) NULL,
  [ForwardLinksRoot] [varchar](100) NULL,
  [FilePath] [varchar](100) NOT NULL,
  [CreatedOn] [datetime] NOT NULL,
  [bid] [uniqueidentifier] NULL,
  CONSTRAINT [PK__Brand__3214EC0745F365D3] PRIMARY KEY CLUSTERED
(
  [Id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
CREATE TABLE [dbo].[DealerBrand](
  [DealerId] [int] NOT NULL,
  [BrandId] [int] NOT NULL,
  [CreatedTime] [datetime] NOT NULL,
  CONSTRAINT [PK_DealerBrand] PRIMARY KEY CLUSTERED
(
  [DealerId] ASC,
  [BrandId] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]

```

```

CREATE VIEW [dbo].[v_dealerbrand]
AS
SELECT dbo.DealerBrand.DealerId, dbo.DealerBrand.BrandId, dbo.Brand.ShortCode,
dbo.Brand.BrandName, dbo.Brand.FilePath
FROM    dbo.Brand RIGHT OUTER JOIN
        dbo.DealerBrand ON dbo.Brand.Id = dbo.DealerBrand.BrandId
CREATE VIEW [dbo].[v_tblGrpMapping]
AS
SELECT m.MapId, m.GroupId, g.GroupName, m.PagId, p.PageName, m.[view], m.[add],
m.Edit, m.[Delete], m.CreatedOn, p.ParentId, p.[level]
FROM    dbo.tblGrpMapping AS m INNER JOIN
        dbo.tblPermissionGroup AS g ON m.GroupId = g.GroupId RIGHT OUTER JOIN
        dbo.tblPages AS p ON m.PagId = p.PagId
where p.PagId=1
CREATE TABLE [dbo].[tblLeftButtons](
  [lbtnid] [int] NOT NULL,
  [lbtnName] [varchar](100) NULL,
  [lbtnImage] [varchar](max) NULL,
  [lbtnUri] [varchar](max) NULL,
  [lbtnCode] [varchar](50) NULL,
  [lbtnPriority] [int] NULL,
  [lbtnLevel] [int] NULL,
  [isactive] [int] NULL,
  [lbtnType] [int] NULL,
  CONSTRAINT [PK_tblButtons] PRIMARY KEY CLUSTERED
(
  [lbtnid] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
CREATE view [dbo].[v_tblPageButtonMapping] as
SELECT pagebtnid
      ,pageid
      ,p.lbtnid,l.lbtnName
      ,show
      ,priority
FROM EzeTalk.dbo.tblPageButtonMapping p,dbo.tblLeftButtons l
where p.lbtnid=l.lbtnid
CREATE TABLE [dbo].[productSeller](
  [PSId] [numeric](18, 0) NOT NULL,
  [ProductId] [numeric](10, 0) NOT NULL,
  [SellerId] [int] NOT NULL,
  [SellerDescription] [varchar](2000) NOT NULL,
  [Price] [numeric](10, 2) NULL,
  [CurrencyId] [int] NULL,
  [PriceUnits] [int] NULL,
  [PriceUomId] [int] NULL,
  [PricePackId] [int] NULL,
  [KeyWords] [varchar](1200) NULL,
  [CreatedOn] [smalldatetime] NOT NULL,
  [UpdatedOn] [smalldatetime] NOT NULL,
  [UserId] [int] NOT NULL,
  [PrId] [int] NOT NULL,
  [MinOrder] [int] NULL,
  [MinOrderUom] [int] NULL,

```



```

[DelTime] [int] NULL,
[DelTimeType] [int] NULL,
[LinkPSId] [numeric](18, 0) NULL,
[ProductDescription] [varchar](max) NULL,
[DiscountLink] [int] NULL,
[BrandName] [varchar](50) NULL,
[MaterialCare] [varchar](max) NULL,
[MaterialType] [varchar](50) NULL,
[Manufacturer] [varchar](max) NULL,
[Length] [varchar](50) NULL,
[MadeinCountry] [int] NULL,
PRIMARY KEY CLUSTERED
(
[PSId] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
CREATE TABLE [dbo].[tblUser](
[UserId] [int] NOT NULL,
[UserName] [varchar](100) NOT NULL,
[Password] [varchar](400) NOT NULL,
[FirstName] [varchar](100) NULL,
[LastName] [varchar](100) NULL,
[EmailID] [varchar](150) NULL,
[ContactNum] [varchar](50) NULL,
[CreatedOn] [datetime] NULL,
[Status] [bit] NULL,
[RoleID] [int] NULL,
[GroupID] [int] NULL,
[IsRemoved] [bit] NOT NULL,
[Department] [varchar](100) NULL,
[Region] [varchar](100) NULL,
[UpdatedOn] [datetime] NULL,
[DealerId] [int] NOT NULL,
PRIMARY KEY CLUSTERED
(
[UserName] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
/***** Script for SelectTopNRows command from SSMS *****/
CREATE view [dbo].[SellersProductSellers] as
select ps.*,BuyersPSId,tu.UserId SUID ,UserName ,FirstName ,LastName
,EmailID ,ContactNum
from productSeller ps,SellersBuyers sb,tblUser tu where ps.psid=sb.SellerPSId and
ps.SellerId=tu.UserId
CREATE view [dbo].[BuyersProductSellers] as
select PSId BPSId,ProductId BProductId,SellerId BSellerId,PrId BPrId
,SellerDescription BSellerDescription,SellerPSId BSellerPSId,tu.UserId BUID ,UserName
BUserName,FirstName BFirstName ,LastName BLastName ,EmailID BEmailID
,ContactNum BContactNum
from productSeller ps,SellersBuyers sb,tblUser tu where ps.psid=sb.BuyersPSId and
ps.SellerId=tu.UserId
CREATE view [dbo].[SellersBuyersProductSellers] as

```

```

select distinct * from SellersProductSellers sps,BuyersProductSellers bps where
sps.BuyersPSId=bps.BPSId
CREATE view [dbo].[VProductMenu]
as
select PageId= PCatId
,Pagename=PcatDesc,[level]=0,parentId=0,PageCode=PcatDesc,[priority]=PCatId,PageUrl=
'OrderSearchList.aspx?cid='+CONVERT(VARCHAR (10),PCatId),'('+CONVERT(VARCHAR
(MAX), PCatId)+' -0 )' as concatenador
from dbo.ProductCat
UNION ALL
select PageId=pgid+(select max(PCatId) from dbo.ProductCat)
,Pagename=pgroupname,[level]=1,parentId=PCatId,PageCode=pgroupname,[priority]=pgid,
PageUrl='OrderSearchList.aspx?gid='+CONVERT(VARCHAR
(10),pgid),'('+CONVERT(VARCHAR (MAX), pgid)+' -0 )' as concatenador
from dbo.ProductGroup
union all
select pageid= ProductId +(select max(PCatId) from dbo.ProductCat)+(select max(pgid)
from dbo.ProductGroup) ,ProductName,[level]=2,parentId=ProductGroupId+(select
count(*) from
dbo.ProductCat),PageCode=ProductName,[priority]=ProductId,PageUrl='OrderSearchList.as
px?pid='+CONVERT(VARCHAR (10),ProductId),'('+CONVERT(VARCHAR (MAX),
ProductId)+' -0 )' as concatenador
from .dbo.Product
create view [dbo].[v_allcolumns] as
select tab_name=b.name,col_name=a.name,a.column_id,is_nullable=isnull(a.is_nullable,0)
from sys.all_columns a,sys.all_objects b where a.object_id=b.object_id and
B.type_desc='USER_TABLE'
CREATE TABLE [dbo].[ActionQueueDepartment](
[ActionQueueId] [int] IDENTITY(1,1) NOT NULL,
[DepartmentId] [int] NOT NULL,
CONSTRAINT [PK_ActionQueueDepartment] PRIMARY KEY CLUSTERED
(
[DepartmentId] ASC,
[ActionQueueId] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
CREATE TABLE [dbo].[ActionQueueRole](
[RoleId] [int] NOT NULL,
[ActionQueueId] [int] NOT NULL,
CONSTRAINT [PK_ActionQueueRole] PRIMARY KEY CLUSTERED
(
[RoleId] ASC,
[ActionQueueId] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
CREATE TABLE [dbo].[ActionQueueUser](
[UserId] [int] NOT NULL,
[ActionQueueId] [int] NOT NULL,
CONSTRAINT [PK_ActionQueueUser] PRIMARY KEY CLUSTERED
(
[UserId] ASC,
[ActionQueueId] ASC

```

```
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
```

```
) ON [PRIMARY]
```

```
CREATE TABLE [dbo].[AuditTrail](
```

```
[Id] [int] IDENTITY(1,1) NOT NULL,
```

```
[CreatedOn] [datetime] NOT NULL,
```

```
[UserId] [int] NULL,
```

```
[ObjectId] [int] NULL,
```

```
[Category] [varchar](50) NULL,
```

```
[Message] [varchar](max) NULL,
```

```
[ExtraInfo] [xml] NULL,
```

```
[AuditDate] [datetime] NULL,
```

```
[PageId] [int] NULL,
```

```
CONSTRAINT [PK__AuditTra__3214EC0732AB8735] PRIMARY KEY CLUSTERED
```

```
(
```

```
[Id] ASC
```

```
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
```

```
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
```

```
CREATE TABLE [dbo].[BankHoliday](
```

```
[Id] [int] IDENTITY(1,1) NOT NULL,
```

```
[HolidayDate] [datetime] NOT NULL,
```

```
[Description] [varchar](50) NOT NULL,
```

```
[Recurring] [bit] NOT NULL,
```

```
[CreatedOn] [datetime] NOT NULL,
```

```
CONSTRAINT [PK_BankHoliday] PRIMARY KEY CLUSTERED
```

```
(
```

```
[Id] ASC
```

```
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
```

```
) ON [PRIMARY]
```

```
CREATE TABLE [dbo].[Business_Type](
```

```
[BTId] [int] NOT NULL,
```

```
[BTDescription] [varchar](50) NOT NULL,
```

```
[createdon] [smalldatetime] NOT NULL,
```

```
[UserId] [int] NOT NULL,
```

```
PRIMARY KEY CLUSTERED
```

```
(
```

```
[BTId] ASC
```

```
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
```

```
) ON [PRIMARY]
```

```
CREATE TABLE [dbo].[CancellationReason](
```

```
[Id] [int] IDENTITY(1,1) NOT NULL,
```

```
[Name] [varchar](50) NOT NULL,
```

```
[Description] [varchar](max) NULL,
```

```
[IsActive] [bit] NOT NULL,
```

```
[ActionTaskSummary] [varchar](100) NULL,
```

```
[ActionTaskDetails] [text] NULL,
```

```
[CreatedOn] [datetime] NOT NULL,
```

```
[UserId] [int] NULL,
```

```
[ActionQueueId] [int] NULL,
```

```
[crid] [uniqueidentifier] NULL,
```

```
CONSTRAINT [PK_CancellationReason] PRIMARY KEY CLUSTERED
```

```
(
```

```

[Id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
CREATE TABLE [dbo].[channels](
[ChannelId] [int] NOT NULL,
[ChannelName] [varchar](50) NOT NULL,
[isActive] [bit] NOT NULL,
[createdOn] [datetime] NOT NULL,
PRIMARY KEY CLUSTERED
(
[ChannelId] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
CREATE TABLE [dbo].[CommissionLog](
[Id] [int] IDENTITY(1,1) NOT NULL,
[CustomerId] [int] NULL,
[CustomerOrderId] [int] NULL,
[DealerId] [int] NOT NULL,
[RaisedDate] [datetime] NOT NULL,
[ProcessedDate] [datetime] NULL,
[Description] [varchar](500) NOT NULL,
[Amount] [money] NOT NULL,
[EntryType] [int] NOT NULL,
[CommissionReportId] [int] NULL,
[AddedById] [int] NULL,
[Suspended] [bit] NOT NULL,
[CreatedOn] [datetime] NULL,
CONSTRAINT [PK_CommissionLog] PRIMARY KEY CLUSTERED
(
[Id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
CREATE TABLE [dbo].[CommissionReport](
[Id] [int] IDENTITY(1,1) NOT NULL,
[DealerId] [uniqueidentifier] NOT NULL,
[ReportDate] [datetime] NOT NULL,
[Amount] [money] NOT NULL,
[FileUrl] [varchar](1000) NULL,
[CreatedOn] [datetime] NULL,
CONSTRAINT [PK_CommissionReport] PRIMARY KEY CLUSTERED
(
[Id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
CREATE TABLE [dbo].[CompanyCreditCheck](
[Id] [int] IDENTITY(1,1) NOT NULL,
[CompanyId] [varchar](100) NOT NULL,
[CompanyRegNo] [varchar](12) NOT NULL,
[CompanyName] [varchar](200) NOT NULL,
[CheckDate] [datetime] NOT NULL,
[Agency] [varchar](200) NULL,

```

```

[ResultXml] [xml] NULL,
[ResultPdf] [varbinary](max) NULL,
[CheckType] [varchar](50) NULL,
[OrderReferenceNumber] [varchar](50) NULL,
[IsAvailable] [bit] NULL,
[CreatedOn] [datetime] NOT NULL,
CONSTRAINT [PK_CompanyCreditCheck] PRIMARY KEY CLUSTERED
(
  [Id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
CREATE TABLE [dbo].[country](
  [id] [int] NOT NULL,
  [shortcd] [varchar](25) NOT NULL,
  [countryname] [varchar](255) NOT NULL,
  [IsBlocked] [bit] NOT NULL,
  [CFlag] [image] NULL,
PRIMARY KEY CLUSTERED
(
  [id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
CREATE TABLE [dbo].[Currency](
  [CId] [int] NOT NULL,
  [CShortCode] [varchar](10) NULL,
  [CDescription] [varchar](40) NULL,
  [createdon] [smalldatetime] NULL,
  [updatedon] [smalldatetime] NULL,
  [userid] [int] NULL,
PRIMARY KEY CLUSTERED
(
  [CId] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
CREATE TABLE [dbo].[Dealer](
  [DealerId] [int] NOT NULL,
  [CompanyName] [varchar](50) NOT NULL,
  [IsActive] [bit] NOT NULL,
  [UserNamePrefix] [varchar](20) NOT NULL,
  [AutoProcessCommissions] [bit] NOT NULL,
  [DealerOperationProfileId] [int] NOT NULL,
  [CreatedOn] [datetime] NOT NULL,
  [UpdatedOn] [datetime] NOT NULL,
  [did] [uniqueidentifier] NULL,
  [EMailString] [varchar](500) NULL,
CONSTRAINT [PK__Dealer__3214EC071FCDBCEB] PRIMARY KEY CLUSTERED
(
  [DealerId] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
CREATE TABLE [dbo].[DealerOperationProfile](

```

```

[Id] [int] IDENTITY(1,1) NOT NULL,
[Name] [varchar](50) NOT NULL,
[HasExternalReference] [bit] NOT NULL,
[HasImmediateHoldDate] [bit] NOT NULL,
CONSTRAINT [PK_DealerOperationProfile] PRIMARY KEY CLUSTERED
(
[Id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
CREATE TABLE [dbo].[Department](
[DeptId] [int] IDENTITY(1,1) NOT NULL,
[DeptName] [varchar](50) NOT NULL,
[IsActive] [bit] NOT NULL,
[CreatedOn] [datetime] NOT NULL,
CONSTRAINT [PK__Departme__3214EC07108B795B] PRIMARY KEY CLUSTERED
(
[DeptId] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
CREATE TABLE [dbo].[DepartmentPrivilege](
[DepartmentId] [int] NOT NULL,
[PrivilegeId] [int] NOT NULL,
[CreatedOn] [datetime] NOT NULL,
CONSTRAINT [PK_DeptmentPrivilege] PRIMARY KEY CLUSTERED
(
[DepartmentId] ASC,
[PrivilegeId] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
CREATE TABLE [dbo].[DirectDebitCommand](
[Id] [int] IDENTITY(1,1) NOT NULL,
[CreatedOn] [datetime] NOT NULL,
[CommandType] [int] NOT NULL,
[CustomerId] [int] NULL,
[SortCode] [varchar](10) NULL,
[AccountNumber] [varchar](10) NULL,
[BacsReference] [varchar](50) NULL,
[AccountHolderName] [varchar](100) NULL,
[ProcessedDate] [datetime] NULL,
[ddid] [uniqueidentifier] NULL,
[brandid] [int] NULL,
CONSTRAINT [PK_DirectDebitCommand] PRIMARY KEY CLUSTERED
(
[Id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
CREATE TABLE [dbo].[Flag](
[Id] [int] IDENTITY(1,1) NOT NULL,
[SubjectId] [int] NOT NULL,
[UserId] [int] NOT NULL,
[CreatedOn] [datetime] NOT NULL,

```

```

[Reason] [varchar](50) NULL,
[CallingPageId] [int] NULL,
[fid] [uniqueidentifier] NULL,
[subid] [uniqueidentifier] NULL,
CONSTRAINT [PK_Flag] PRIMARY KEY CLUSTERED
(
[Id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
CREATE TABLE [dbo].[FlagEvent](
[Id] [int] IDENTITY(1,1) NOT NULL,
[FlagId] [int] NOT NULL,
[CreatedOn] [datetime] NOT NULL,
[EventDate] [datetime] NOT NULL,
[Description] [varchar](500) NOT NULL,
[feid] [uniqueidentifier] NULL,
CONSTRAINT [PK_FlagEvent] PRIMARY KEY CLUSTERED
(
[Id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
CREATE TABLE [dbo].[formslist](
[AddNote#aspx] [nvarchar](255) NULL
) ON [PRIMARY]
CREATE TABLE [dbo].[Graphs](
[Period] [varchar](50) NULL,
[Plnamt] [int] NULL,
[Actamt] [int] NULL
) ON [PRIMARY]
CREATE TABLE [dbo].[HoldReason](
[Id] [int] IDENTITY(1,1) NOT NULL,
[Name] [varchar](50) NOT NULL,
[Description] [varchar](max) NULL,
[IsActive] [bit] NOT NULL,
[ActionTaskSummary] [varchar](100) NULL,
[ActionTaskDetails] [text] NULL,
[CreatedOn] [datetime] NOT NULL,
[ActionQueueId] [int] NULL,
[CreatedBy] [int] NULL,
[hrid] [uniqueidentifier] NULL,
CONSTRAINT [PK_HoldReason] PRIMARY KEY CLUSTERED
(
[Id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
CREATE TABLE [dbo].[Legal_structure](
[lsid] [int] NOT NULL,
[lsdescription] [varchar](50) NOT NULL,
[createdon] [smalldatetime] NOT NULL,
[UsetId] [int] NULL,
PRIMARY KEY CLUSTERED
(

```

```

[Isid] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
CREATE TABLE [dbo].[NewsItem](
[Id] [int] IDENTITY(1,1) NOT NULL,
[Summary] [varchar](200) NOT NULL,
[Details] [varchar](max) NOT NULL,
[PublishDate] [datetime] NOT NULL,
[ExpireDate] [datetime] NOT NULL,
[PublishedById] [int] NOT NULL,
[CreatedOn] [datetime] NOT NULL,
CONSTRAINT [PK_NewsItem] PRIMARY KEY CLUSTERED
(
[Id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
CREATE TABLE [dbo].[Note](
[Id] [int] NOT NULL,
[CreatedOn] [datetime] NOT NULL,
[NoteTypeId] [int] NOT NULL,
[CallingPageId] [int] NOT NULL,
[SubjectId] [int] NOT NULL,
[NoteDate] [datetime] NOT NULL,
[Body] [varchar](max) NULL,
[CreatedById] [int] NOT NULL,
[AttachmentData] [varbinary](max) NULL,
[AttachmentFilename] [varchar](250) NULL,
[AttachmentType] [varchar](500) NULL,
[AttachmentLength] [int] NULL,
[nid] [uniqueidentifier] NULL,
[subid] [uniqueidentifier] NULL,
CONSTRAINT [PK__Note__3214EC0719DFD96B] PRIMARY KEY CLUSTERED
(
[Id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
CREATE TABLE [dbo].[NoteType](
[Id] [int] IDENTITY(1,1) NOT NULL,
[Name] [varchar](25) NOT NULL,
[Description] [varchar](max) NULL,
[IsActive] [bit] NOT NULL,
[IsDefault] [bit] NOT NULL,
[CreatedOn] [datetime] NOT NULL,
[ntid] [uniqueidentifier] NULL,
CONSTRAINT [PK__NoteType__3214EC07123EB7A3] PRIMARY KEY CLUSTERED
(
[Id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
CREATE TABLE [dbo].[pbcaccol](
[pbc_tnam] [char](129) NOT NULL,

```



```

[psc_tid] [int] NULL,
[psc_ownr] [char](129) NOT NULL,
[psc_cnam] [char](129) NOT NULL,
[psc_cid] [smallint] NULL,
[psc_labl] [varchar](254) NULL,
[psc_lpos] [smallint] NULL,
[psc_hdr] [varchar](254) NULL,
[psc_hpos] [smallint] NULL,
[psc_jtfy] [smallint] NULL,
[psc_mask] [varchar](31) NULL,
[psc_case] [smallint] NULL,
[psc_hght] [smallint] NULL,
[psc_wdth] [smallint] NULL,
[psc_ptrn] [varchar](31) NULL,
[psc_bmap] [char](1) NULL,
[psc_init] [varchar](254) NULL,
[psc_cmnt] [varchar](254) NULL,
[psc_edit] [varchar](31) NULL,
[psc_tag] [varchar](254) NULL
) ON [PRIMARY]
CREATE TABLE [dbo].[pscatdtd](
[psbe_name] [varchar](30) NOT NULL,
[psbe_edit] [varchar](254) NULL,
[psbe_type] [smallint] NULL,
[psbe_cntr] [int] NULL,
[psbe_seqn] [smallint] NOT NULL,
[psbe_flag] [int] NULL,
[psbe_work] [char](32) NULL
) ON [PRIMARY]
CREATE TABLE [dbo].[pscatfmt](
[psbf_name] [varchar](30) NOT NULL,
[psbf_frmt] [varchar](254) NULL,
[psbf_type] [smallint] NULL,
[psbf_cntr] [int] NULL
) ON [PRIMARY]
CREATE TABLE [dbo].[pscattbl](
[pbt_tnam] [char](129) NOT NULL,
[pbt_tid] [int] NULL,
[pbt_ownr] [char](129) NOT NULL,
[pbd_fhgt] [smallint] NULL,
[pbd_fwgt] [smallint] NULL,
[pbd_fitl] [char](1) NULL,
[pbd_funl] [char](1) NULL,
[pbd_fchr] [smallint] NULL,
[pbd_fptc] [smallint] NULL,
[pbd_ffce] [char](18) NULL,
[pbh_fhgt] [smallint] NULL,
[pbh_fwgt] [smallint] NULL,
[pbh_fitl] [char](1) NULL,
[pbh_funl] [char](1) NULL,
[pbh_fchr] [smallint] NULL,
[pbh_fptc] [smallint] NULL,
[pbh_ffce] [char](18) NULL,
[pbl_fhgt] [smallint] NULL,
[pbl_fwgt] [smallint] NULL,

```

```

[pbl_fit] [char](1) NULL,
[pbl_fun] [char](1) NULL,
[pbl_fchr] [smallint] NULL,
[pbl_fptc] [smallint] NULL,
[pbl_ffce] [char](18) NULL,
[pbt_cmnt] [varchar](254) NULL
) ON [PRIMARY]
CREATE TABLE [dbo].[pbcatvld](
[pbv_name] [varchar](30) NOT NULL,
[pbv_vald] [varchar](254) NULL,
[pbv_type] [smallint] NULL,
[pbv_cntr] [int] NULL,
[pbv_msg] [varchar](254) NULL
) ON [PRIMARY]
CREATE TABLE [dbo].[prd](
[ProductId] [float] NULL,
[ProductName] [nvarchar](255) NULL,
[F3] [nvarchar](255) NULL,
[F4] [nvarchar](255) NULL,
[ProductGroupCate ry] [float] NULL,
[ProductGroupld] [float] NULL,
[Dealerld] [float] NULL,
[Price] [float] NULL,
[Currencyld] [float] NULL,
[AnalysisCode] [float] NULL,
[CreatedOn] [datetime] NULL,
[UpdatedOn] [datetime] NULL,
[UserId] [float] NULL,
[Prld] [nvarchar](255) NULL,
[MinOrder] [float] NULL,
[DelTlme] [float] NULL,
[lbtnimage] [nvarchar](255) NULL,
[path] [nvarchar](255) NULL
) ON [PRIMARY]
CREATE TABLE [dbo].[productExtProperties](
[EpId] [int] NOT NULL,
[EpDescription] [varchar](50) NOT NULL,
[active] [bit] NOT NULL,
PRIMARY KEY CLUSTERED
(
[EpId] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
CREATE TABLE [dbo].[productExtPrOptions](
[EpId] [int] NOT NULL,
[EpOptld] [int] NOT NULL,
[EpOPtDescr] [varchar](50) NOT NULL,
[active] [bit] NOT NULL,
PRIMARY KEY CLUSTERED
(
[EpId] ASC,
[EpOptld] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]

```

```

) ON [PRIMARY]
CREATE TABLE [dbo].[productSellerExtProperties](
  [PSId] [numeric](18, 0) NOT NULL,
  [EpId] [int] NOT NULL,
  [EpOptId] [int] NOT NULL,
  [Remark] [varchar](50) NULL,
  [active] [bit] NOT NULL,
  PRIMARY KEY CLUSTERED
(
  [PSId] ASC,
  [EpId] ASC,
  [EpOptId] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
CREATE TABLE [dbo].[productSellerImages](
  [PSID] [numeric](18, 0) NOT NULL,
  [ProductId] [numeric](10, 0) NOT NULL,
  [SellerId] [int] NOT NULL,
  [srno] [int] NOT NULL,
  [Images] [image] NULL,
  [ImagePath] [varchar](max) NULL,
  [ImageURL] [varchar](max) NULL,
  [ImageThumb] [varchar](150) NULL,
  PRIMARY KEY CLUSTERED
(
  [PSID] ASC,
  [srno] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
CREATE TABLE [dbo].[RecentPlaces](
  [rp_id] [int] IDENTITY(1,1) NOT NULL,
  [rp_type] [varchar](50) NULL,
  [rp_heading] [varchar](100) NULL,
  [rp_url] [varchar](500) NULL,
  [rp_date] [datetime] NULL,
  [rp_user] [int] NULL,
  CONSTRAINT [PK_RecentPlaces] PRIMARY KEY CLUSTERED
(
  [rp_id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
CREATE TABLE [dbo].[Role](
  [Id] [int] IDENTITY(1,1) NOT NULL,
  [Name] [varchar](50) NOT NULL,
  [IsActive] [bit] NOT NULL,
  [CreatedOn] [datetime] NOT NULL,
  [rlid] [uniqueidentifier] NULL,
  CONSTRAINT [PK__Role__3214EC0719DFD96B] PRIMARY KEY CLUSTERED
(
  [Id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]

```

```

) ON [PRIMARY]
CREATE TABLE [dbo].[Scratchpad](
  [Id] [int] IDENTITY(1,1) NOT NULL,
  [CreatedOn] [datetime] NOT NULL,
  [UserId] [int] NOT NULL,
  [CallingPageId] [int] NOT NULL,
  [SubjectId] [int] NOT NULL,
  [Content] [varchar](max) NOT NULL,
  [spid] [uniqueidentifier] NULL,
  CONSTRAINT [PK__ScratchP__3214EC07778AC167] PRIMARY KEY CLUSTERED
(
  [Id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
CREATE TABLE [dbo].[StaffHoliday](
  [Id] [int] IDENTITY(1,1) NOT NULL,
  [CreatedOn] [datetime] NOT NULL,
  [UserId] [int] NOT NULL,
  [StartDate] [datetime] NOT NULL,
  [EndDate] [datetime] NOT NULL,
  [ActualDays] [decimal](8, 2) NOT NULL,
  [RequestedDate] [datetime] NOT NULL,
  [Status] [int] NOT NULL,
  [DecisionDate] [datetime] NULL,
  [DecisionById] [int] NULL,
  [Description] [varchar](200) NULL,
  [RejectionReason] [varchar](200) NULL,
  [BalanceBefore] [decimal](8, 2) NULL,
  [BalanceAfter] [decimal](8, 2) NULL,
  CONSTRAINT [PK_StaffHoliday] PRIMARY KEY CLUSTERED
(
  [Id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
CREATE TABLE [dbo].[TaskStatus](
  [Id] [char](1) NOT NULL,
  [TStatusName] [varchar](50) NOT NULL,
  PRIMARY KEY CLUSTERED
(
  [Id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
CREATE TABLE [dbo].[Tbl_ExceptionLoggingToDataBase](
  [Logid] [bigint] IDENTITY(1,1) NOT NULL,
  [ExceptionMsg] [varchar](100) NULL,
  [ExceptionType] [varchar](100) NULL,
  [ExceptionSource] [nvarchar](max) NULL,
  [ExceptionURL] [varchar](100) NULL,
  [Logdate] [datetime] NULL,
  CONSTRAINT [PK_Tbl_ExceptionLoggingToDataBase] PRIMARY KEY CLUSTERED
(
  [Logid] ASC

```

```

)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
CREATE TABLE [dbo].[tblcoursalitems](
[id] [int] NOT NULL,
[fpath] [varchar](50) NOT NULL,
[aname] [varchar](10) NOT NULL,
[userid] [int] NOT NULL,
[createdon] [smalldatetime] NULL,
PRIMARY KEY CLUSTERED
(
[id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
CREATE TABLE [dbo].[tbldeftime](
[DeTimeType] [int] NOT NULL,
[DeTimeDescription] [varchar](10) NOT NULL,
[CreatedOn] [smalldatetime] NOT NULL,
[UpdatedOn] [smalldatetime] NOT NULL,
[UserId] [int] NOT NULL,
PRIMARY KEY CLUSTERED
(
[DeTimeType] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
CREATE TABLE [dbo].[tblLeftDDL](
[Iddid] [int] NOT NULL,
[Iddvalue] [varchar](100) NULL,
[Iddpriority] [int] NULL,
[Ibtnid] [int] NULL,
[isactive] [int] NULL,
CONSTRAINT [PK_tblLeftDDL] PRIMARY KEY CLUSTERED
(
[Iddid] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
CREATE TABLE [dbo].[tblLogIn](
[LogInId] [int] IDENTITY(1,1) NOT NULL,
[LogInTime] [datetime] NULL,
[LogoutTime] [datetime] NULL,
[UserId] [int] NULL,
[LoginStatus] [bit] NULL,
[SourceIp] [varchar](100) NULL,
[MachineName] [varchar](100) NULL,
[Description] [varchar](max) NULL,
[City] [varchar](100) NULL,
[Country] [varchar](100) NULL,
CONSTRAINT [PK_tblLogIn] PRIMARY KEY CLUSTERED
(
[LogInId] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]

```

```

) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
CREATE TABLE [dbo].[tblPacking](
  [PackID] [int] NOT NULL,
  [PackShort] [varchar](10) NOT NULL,
  [PackDescription] [varchar](50) NOT NULL,
  [CreatedOn] [smalldatetime] NOT NULL,
  [UpdatedOn] [smalldatetime] NOT NULL,
  [UserId] [int] NOT NULL,
  PRIMARY KEY CLUSTERED
(
  [PackID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
CREATE TABLE [dbo].[tblPageButtonMapping](
  [pagebtnid] [int] NOT NULL,
  [mapid] [int] NOT NULL,
  [lbtnid] [int] NOT NULL,
  [show] [int] NOT NULL,
  [priority] [int] NOT NULL,
  [urlpageid] [int] NOT NULL,
  [pageid] [int] NOT NULL,
  [groupid] [int] NOT NULL,
  [maptype] [int] NULL,
  [mapdescription] [varchar](max) NULL,
  [isactive] [int] NOT NULL,
  CONSTRAINT [PK_tblPageButtonMapping] PRIMARY KEY CLUSTERED
(
  [pagebtnid] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
CREATE TABLE [dbo].[tblPageButtonMapping1](
  [pagebtnid] [int] NOT NULL,
  [mapid] [int] NOT NULL,
  [lbtnid] [int] NOT NULL,
  [show] [int] NOT NULL,
  [priority] [int] NOT NULL,
  [urlpageid] [int] NOT NULL,
  [pageid] [int] NOT NULL,
  [groupid] [int] NOT NULL,
  [maptype] [int] NULL,
  [mapdescription] [varchar](max) NULL,
  [isactive] [int] NOT NULL
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
CREATE TABLE [dbo].[tblRole](
  [RoleId] [int] IDENTITY(1,1) NOT NULL,
  [RoleType] [varchar](50) NULL,
  [CreatedOn] [datetime] NULL,
  CONSTRAINT [PK_tblRole] PRIMARY KEY CLUSTERED
(
  [RoleId] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]

```

```

CREATE TABLE [dbo].[tbluom](
  [UomId] [int] NOT NULL,
  [UomShort] [varchar](10) NOT NULL,
  [UomDescription] [varchar](50) NOT NULL,
  [CreatedOn] [smalldatetime] NOT NULL,
  [UpdatedOn] [smalldatetime] NOT NULL,
  [UserId] [int] NOT NULL,
  PRIMARY KEY CLUSTERED
(
  [UomId] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
CREATE TABLE [dbo].[tbluser_add_type](
  [adid] [int] NOT NULL,
  [adtype] [varchar](10) NOT NULL,
  PRIMARY KEY CLUSTERED
(
  [adid] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
CREATE TABLE [dbo].[tbluser_address](
  [userid] [int] NOT NULL,
  [address_type] [int] NOT NULL,
  [address1] [varchar](100) NULL,
  [address2] [varchar](100) NULL,
  [address3] [varchar](100) NULL,
  [pincode] [varchar](10) NULL,
  [country] [int] NULL,
  [City] [varchar](50) NULL,
  [ContactNo] [varchar](50) NULL,
  CONSTRAINT [PK__tbluser_address] PRIMARY KEY CLUSTERED
(
  [userid] ASC,
  [address_type] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
CREATE TABLE [dbo].[tblUser_Emails](
  [EmailId] [int] NOT NULL,
  [EMSentUserId] [int] NOT NULL,
  [EMRcvdUserId] [int] NOT NULL,
  [EmSubject] [varchar](250) NOT NULL,
  [EmBody] [varchar](max) NOT NULL,
  [CreatedOn] [smalldatetime] NOT NULL,
  [UserId] [int] NOT NULL,
  PRIMARY KEY CLUSTERED
(
  [EmailId] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
CREATE TABLE [dbo].[tbluser_profile](
  [userid] [int] NOT NULL,

```

```

[Business_Type] [varchar](50) NOT NULL,
[Legal_Structure] [varchar](50) NOT NULL,
[Year_Established] [varchar](4) NOT NULL,
[Number_of_Employees] [varchar](10) NOT NULL,
[Company_Description] [varchar](1250) NOT NULL,
[Company_Website_URL] [varchar](150) NOT NULL,
[Turnover_per_Year] [varchar](10) NOT NULL,
[CID] [int] NOT NULL,
[Quality_Assurance_Accreditation] [varchar](50) NOT NULL,
[Registration_Number] [varchar](50) NOT NULL,
[D_U_N_S_Number] [varchar](50) NOT NULL,
[Specializing_in] [varchar](50) NOT NULL,
[Monthly_Production_Capacity] [varchar](10) NOT NULL,
[UOMId] [int] NOT NULL,
[Company_lo ] [image] NULL,
[CreatedOn] [smalldatetime] NOT NULL,
[UpdatedOn] [smalldatetime] NOT NULL,
[uid] [int] NOT NULL
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
ALTER TABLE [dbo].[tbluser_profile] ADD [lo _path] [varchar](150) NULL
PRIMARY KEY CLUSTERED
(
[userid] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
CREATE TABLE [dbo].[tblUserImages](
[UserId] [int] NOT NULL,
[srno] [int] NOT NULL,
[Images] [image] NULL,
[CreatedOn] [smalldatetime] NULL,
PRIMARY KEY CLUSTERED
(
[UserId] ASC,
[srno] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
CREATE TABLE [dbo].[tblUserLockOutDetails](
[UserLockId] [int] IDENTITY(1,1) NOT NULL,
[Lo n attempts] [int] NULL,
[BloackingPeriod] [int] NULL,
[UpdatedOn] [datetime] NULL,
CONSTRAINT [PK_tblUserLockOutDetails] PRIMARY KEY CLUSTERED
(
[UserLockId] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
CREATE TABLE [dbo].[tblUserLog](
[UserLogID] [int] IDENTITY(1,1) NOT NULL,
[PerformAction] [varchar](max) NULL,
[PerformActionTime] [datetime] NULL,
[UserId] [int] NULL,
[LoginId] [int] NULL,
CONSTRAINT [PK_tblUserLog] PRIMARY KEY CLUSTERED

```



```

(
  [UserLogID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
CREATE TABLE [dbo].[UserCart](
  [UCId] [int] NOT NULL,
  [UserId] [int] NOT NULL,
  [PSId] [int] NOT NULL,
  [CreatedOn] [smalldatetime] NOT NULL,
  [UId] [int] NOT NULL,
  [Active] [bit] NOT NULL,
  [PayId] [int] NOT NULL,
  [PayDate] [smalldatetime] NULL,
PRIMARY KEY CLUSTERED
(
  [UCId] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
CREATE TABLE [dbo].[UserPrivilege](
  [UserId] [int] NOT NULL,
  [PrivilegeId] [int] NOT NULL,
  [CreatedOn] [datetime] NOT NULL,
CONSTRAINT [PK_UserPrivilege] PRIMARY KEY CLUSTERED
(
  [UserId] ASC,
  [PrivilegeId] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
CREATE TABLE [dbo].[UserRole](
  [UserId] [int] NOT NULL,
  [RoleId] [int] NOT NULL,
  [CreatedOn] [datetime] NOT NULL,
CONSTRAINT [PK_UserRole] PRIMARY KEY CLUSTERED
(
  [UserId] ASC,
  [RoleId] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
CREATE TABLE [dbo].[UserShortlist](
  [UserId] [int] NOT NULL,
  [PSId] [int] NOT NULL,
  [CreatedOn] [smalldatetime] NOT NULL,
  [UId] [int] NOT NULL,
  [Active] [bit] NOT NULL,
  [SLType] [bit] NOT NULL,
PRIMARY KEY CLUSTERED
(
  [UserId] ASC,
  [PSId] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]

```

```

) ON [PRIMARY]
CREATE TABLE [dbo].[VatCode](
  [Id] [int] IDENTITY(1,1) NOT NULL,
  [Code] [char](3) NOT NULL,
  [Rate] [decimal](5, 2) NOT NULL,
  [Description] [varchar](50) NOT NULL,
  [IsDefault] [bit] NOT NULL,
  [IsActive] [bit] NOT NULL,
  [CreatedOn] [datetime] NOT NULL,
  CONSTRAINT [PK__VatCode__3214EC072DE6D218] PRIMARY KEY CLUSTERED
(
  [Id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
CREATE TABLE [dbo].[WelcomeCallOutcomeEnum](
  [Id] [int] IDENTITY(1,1) NOT NULL,
  [Name] [varchar](50) NOT NULL,
  [isactive] [int] NULL,
  [type] [int] NULL,
  [sid] [int] NULL,
  CONSTRAINT [PK_WelcomeCallOutcomeEnum] PRIMARY KEY CLUSTERED
(
  [Id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
CREATE NONCLUSTERED INDEX [IX_Flag_SubjectId] ON [dbo].[Flag]
(
  [SubjectId] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, SORT_IN_TEMPDB =
OFF, DROP_EXISTING = OFF, ONLINE = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
CREATE UNIQUE NONCLUSTERED INDEX [pbcatc_x] ON [dbo].[pbcatcol]
(
  [pbc_tnam] ASC,
  [pbc_ownr] ASC,
  [pbc_cnam] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, SORT_IN_TEMPDB =
OFF, IGNORE_DUP_KEY = OFF, DROP_EXISTING = OFF, ONLINE = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
CREATE UNIQUE NONCLUSTERED INDEX [pbcate_x] ON [dbo].[pbcatedt]
(
  [pbe_name] ASC,
  [pbe_seqn] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, SORT_IN_TEMPDB =
OFF, IGNORE_DUP_KEY = OFF, DROP_EXISTING = OFF, ONLINE = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
CREATE UNIQUE NONCLUSTERED INDEX [pbcatt_x] ON [dbo].[pbcattfmt]
(
  [pbf_name] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, SORT_IN_TEMPDB =
OFF, IGNORE_DUP_KEY = OFF, DROP_EXISTING = OFF, ONLINE = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
CREATE UNIQUE NONCLUSTERED INDEX [pbcatt_x] ON [dbo].[pbcattbl]

```

```

(
  [pbt_tnam] ASC,
  [pbt_ownr] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, SORT_IN_TEMPDB =
OFF, IGNORE_DUP_KEY = OFF, DROP_EXISTING = OFF, ONLINE = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
CREATE UNIQUE NONCLUSTERED INDEX [pbcatv_x] ON [dbo].[pbcatvId]
(
  [pbv_name] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, SORT_IN_TEMPDB =
OFF, IGNORE_DUP_KEY = OFF, DROP_EXISTING = OFF, ONLINE = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
CREATE NONCLUSTERED INDEX [IX_tblPageButtonMapping] ON
[dbo].[tblPageButtonMapping]
(
  [pagebtnid] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, SORT_IN_TEMPDB =
OFF, DROP_EXISTING = OFF, ONLINE = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
ALTER TABLE [dbo].[ActionQueue] ADD CONSTRAINT [DF_ActionQueue_AutoAssign]
DEFAULT ((0)) FOR [AutoAssign]
ALTER TABLE [dbo].[ActionQueue] ADD CONSTRAINT
[DF_ActionQueue_IsSystemQueue] DEFAULT ((0)) FOR [IsSystemQueue]
ALTER TABLE [dbo].[ActionQueue] ADD CONSTRAINT [DF_ActionQueue_IsActive]
DEFAULT ((1)) FOR [IsActive]
ALTER TABLE [dbo].[ActionTask] ADD CONSTRAINT [DF_ActionTask_TaskType]
DEFAULT ((1)) FOR [TaskType]
ALTER TABLE [dbo].[BankHoliday] ADD CONSTRAINT [DF_BankHoliday_Recurring]
DEFAULT ((0)) FOR [Recurring]
ALTER TABLE [dbo].[CancellationReason] ADD CONSTRAINT
[DF_CancellationReason_IsActive] DEFAULT ((1)) FOR [IsActive]
ALTER TABLE [dbo].[CompanyCreditCheck] ADD CONSTRAINT
[DF_CompanyCreditCheck_CheckDate] DEFAULT (getdate()) FOR [CheckDate]
ALTER TABLE [dbo].[CompanyCreditCheck] ADD CONSTRAINT
[DF_CompanyCreditCheck_IsAvailable] DEFAULT ((0)) FOR [IsAvailable]
ALTER TABLE [dbo].[DealerOperationProfile] ADD CONSTRAINT
[DF_DealerOperationProfile_HasExternalReference] DEFAULT ((0)) FOR
[HasExternalReference]
ALTER TABLE [dbo].[DealerOperationProfile] ADD CONSTRAINT
[DF_DealerOperationProfile_HasImmediateHoldDate] DEFAULT ((0)) FOR
[HasImmediateHoldDate]
ALTER TABLE [dbo].[FlagEvent] ADD CONSTRAINT [DF_FlagEvent_EventDate]
DEFAULT (getdate()) FOR [EventDate]
ALTER TABLE [dbo].[HoldReason] ADD CONSTRAINT [DF_HoldReason_IsActive]
DEFAULT ((1)) FOR [IsActive]
ALTER TABLE [dbo].[Note] ADD CONSTRAINT [DF_Note_NoteDate] DEFAULT
(getdate()) FOR [NoteDate]
ALTER TABLE [dbo].[NoteType] ADD CONSTRAINT [DF_NoteType_IsActive] DEFAULT
((1)) FOR [IsActive]
ALTER TABLE [dbo].[NoteType] ADD CONSTRAINT [DF_NoteType_IsDefault] DEFAULT
((0)) FOR [IsDefault]
ALTER TABLE [dbo].[Role] ADD CONSTRAINT [DF_Role_IsActive] DEFAULT ((1)) FOR
[IsActive]
ALTER TABLE [dbo].[StaffHoliday] ADD CONSTRAINT [DF_StaffHoliday_Status]
DEFAULT ((0)) FOR [Status]

```

```

ALTER TABLE [dbo].[tblLeftButtons] ADD CONSTRAINT [DF_tblLeftButtons_lbtnType]
DEFAULT ((0)) FOR [lbtnType]
ALTER TABLE [dbo].[tblPageButtonMapping] ADD CONSTRAINT
[DF_tblPageButtonMapping_show] DEFAULT ((0)) FOR [show]
ALTER TABLE [dbo].[tblPermissionGroup] ADD CONSTRAINT
[DF_tblPermissionGroup_RoleId] DEFAULT ((3)) FOR [RoleId]
ALTER TABLE [dbo].[tblUser] ADD DEFAULT (getdate()) FOR [CreatedOn]
ALTER TABLE [dbo].[tblUser] ADD DEFAULT ((0)) FOR [IsRemoved]
ALTER TABLE [dbo].[tblUser] ADD DEFAULT ((0)) FOR [DealerId]
ALTER TABLE [dbo].[VatCode] ADD CONSTRAINT [DF_VatCode_IsDefault] DEFAULT
((0)) FOR [IsDefault]
ALTER TABLE [dbo].[VatCode] ADD CONSTRAINT [DF_VatCode_IsActive] DEFAULT
((1)) FOR [IsActive]
ALTER TABLE [dbo].[WelcomeCallOutcomeEnum] ADD DEFAULT ((0)) FOR [isactive]
ALTER TABLE [dbo].[WelcomeCallOutcomeEnum] ADD DEFAULT ((1)) FOR [type]
ALTER TABLE [dbo].[WelcomeCallOutcomeEnum] ADD DEFAULT ((1)) FOR [sid]
ALTER TABLE [dbo].[ActionQueueDepartment] WITH CHECK ADD CONSTRAINT
[FK_ActionQueueDepartment_ActionQueue] FOREIGN KEY([ActionQueueId])
REFERENCES [dbo].[ActionQueue] ([Id])
ALTER TABLE [dbo].[ActionQueueDepartment] CHECK CONSTRAINT
[FK_ActionQueueDepartment_ActionQueue]
ALTER TABLE [dbo].[ActionQueueRole] WITH CHECK ADD CONSTRAINT
[FK_ActionQueueRole_Role] FOREIGN KEY([RoleId])
REFERENCES [dbo].[Role] ([Id])
ALTER TABLE [dbo].[ActionQueueRole] CHECK CONSTRAINT
[FK_ActionQueueRole_Role]
ALTER TABLE [dbo].[ActionQueueUser] WITH CHECK ADD CONSTRAINT
[FK_ActionQueueUser_ActionQueue] FOREIGN KEY([ActionQueueId])
REFERENCES [dbo].[ActionQueue] ([Id])
ALTER TABLE [dbo].[ActionQueueUser] CHECK CONSTRAINT
[FK_ActionQueueUser_ActionQueue]
ALTER TABLE [dbo].[ActionTask] WITH CHECK ADD CONSTRAINT
[FK_ActionTask_ActionQueue] FOREIGN KEY([QueueId])
REFERENCES [dbo].[ActionQueue] ([Id])
ALTER TABLE [dbo].[ActionTask] CHECK CONSTRAINT [FK_ActionTask_ActionQueue]
ALTER TABLE [dbo].[CommissionLog] WITH CHECK ADD CONSTRAINT
[FK_CommissionLog_CommissionReport] FOREIGN KEY([CommissionReportId])
REFERENCES [dbo].[CommissionReport] ([Id])
ALTER TABLE [dbo].[CommissionLog] CHECK CONSTRAINT
[FK_CommissionLog_CommissionReport]
ALTER TABLE [dbo].[FlagEvent] WITH CHECK ADD CONSTRAINT [FK_FlagEvent_Flag]
FOREIGN KEY([FlagId])
REFERENCES [dbo].[Flag] ([Id])
ALTER TABLE [dbo].[FlagEvent] CHECK CONSTRAINT [FK_FlagEvent_Flag]
ALTER TABLE [dbo].[Note] WITH CHECK ADD CONSTRAINT [FK_Note_NoteType]
FOREIGN KEY([NoteTypeId])
REFERENCES [dbo].[NoteType] ([Id])
ALTER TABLE [dbo].[Note] CHECK CONSTRAINT [FK_Note_NoteType]
ALTER TABLE [dbo].[tblLeftDDL] WITH CHECK ADD CONSTRAINT
[FK_tblLeftDDL_tblLeftButtons] FOREIGN KEY([lbtnid])
REFERENCES [dbo].[tblLeftButtons] ([lbtnid])
ALTER TABLE [dbo].[tblLeftDDL] CHECK CONSTRAINT [FK_tblLeftDDL_tblLeftButtons]
ALTER TABLE [dbo].[tblUserLog] WITH CHECK ADD CONSTRAINT
[FK_tblUserLog_tblLogIn] FOREIGN KEY([LoginId])
REFERENCES [dbo].[tblLogIn] ([LogInId])

```

```

ALTER TABLE [dbo].[tblUserLog] CHECK CONSTRAINT [FK_tblUserLog_tblLogIn]
ALTER TABLE [dbo].[UserRole] WITH CHECK ADD CONSTRAINT [FK_UserRole_Role]
FOREIGN KEY([RoleId])
REFERENCES [dbo].[Role] ([Id])
ALTER TABLE [dbo].[UserRole] CHECK CONSTRAINT [FK_UserRole_Role]
CREATE procedure [dbo].[AddButtonMapping]
  (@pagebtnid int, @show int, @groupid int) as
  declare @pageid int
  declare @lbtnid int
  declare @mapid int
  declare @pagebtnidnew int
  declare @cnt int
  declare @urlpageid int
  declare @maptype int
  declare @mapdescription varchar(max)
  begin
    select
      @pageid=pageid, @lbtnid=lbtnid, @urlpageid=@urlpageid, @maptype=maptype, @mapdescri
      ption=mapdescription from tblPageButtonMapping where pagebtnid=@pagebtnid
    select @mapid=mapid from tblGrpMapping where pageid=@pageid and groupid=@groupid
    if @mapid=null
      begin
        select @mapid=mapid from tblGrpMapping where pageid=@pageid and groupid=1
      end
    select @pagebtnidnew=isnull(MAX(pagebtnid),0)+1 from tblPageButtonMapping
    if @pageid is not null and @lbtnid is not null
      begin
        select @cnt=count(*) from tblPageButtonMapping where pageid=@pageid and
        lbtnid=@lbtnid and groupid=@groupid
        if @cnt=0
          begin
            insert into tblPageButtonMapping
              (pagebtnid, [pageid] , [lbtnid] , [show] , [priority]
              , [urlpageid], mapid, groupid, maptype
              , mapdescription, isactive)
              values(@pagebtnidnew, @pageid , @lbtnid , @show , 1
              , @urlpageid, @mapid, @groupid, @maptype
              , @mapdescription, 1)
          end
        end
      end
    end
  CREATE procedure [dbo].[AddNewPageMapping]
  ( @MapId int, @GroupId int, @view bit) as
  begin
    declare @PageId int
    declare @cnt int
    select @PageId=PageId from tblGrpMapping where MapId=@mapid
    select @cnt=COUNT(*) from tblGrpMapping where GroupId=@GroupId and
    PageId=@PageId
    if @PageId is not null and @cnt=0
      begin
        insert into tblGrpMapping (GroupId, PageId, [view] , [add]
        , [Edit] , [Delete] , [CreatedOn])
        values(@GroupId, @PageId, @view, 0, 0, 0, GETDATE())
      end
  end

```

```

end
CREATE PROC [dbo].[AddParent]
  @Pageld int,
  @ParentId int,
  @Level int
AS
BEGIN
  insert into tblTempPages(PId,parentId,[level]) values(@Pageld,@ParentId,0)
END
CREATE procedure [dbo].[CancellationSummary] as
begin
  declare @total numeric(5,2)
  select @total=count(*) from dbo.CustomerOrder where CancellationReasonId is not null
  select
  Nos=count(*),CancellationReasonId,Name,Ratio=round(count(*)/@total*100,2),Total=@total
  from dbo.CustomerOrder co,dbo.CancellationReason cr
  where co.CancellationReasonId=cr.id group by CancellationReasonId,Name
end
CREATE procedure [dbo].[CancellationTimeLine] as
begin
  with n2(g1,g2,g3,g4,g5,DealerId,Mth)as(
  select g1=COUNT(*),g2=0,g3=0,g4=0,g5=0,DealerId,Mth from v_CanTL where CanDays
  between 0 and 6 group by DealerId,Mth
  union all
  select g1=0,g2=COUNT(*),g3=0,g4=0,g5=0,DealerId,Mth from v_CanTL where CanDays
  between 7 and 13 group by DealerId,Mth
  union all
  select g1=0,g2=0,g3=COUNT(*),g4=0,g5=0,DealerId,Mth from v_CanTL where CanDays
  between 14 and 20 group by DealerId,Mth
  union all
  select g1=0,g2=0,g3=0,g4=COUNT(*),g5=0,DealerId,Mth from v_CanTL where CanDays
  between 21 and 27 group by DealerId,Mth
  union all
  select g1=0,g2=0,g3=0,g4=0,g5=COUNT(*),DealerId,Mth from v_CanTL where CanDays>
  27 group by DealerId,Mth)
  select
  g1=SUM(g1),g2=SUM(g2),g3=SUM(g3),g4=SUM(g4),g5=SUM(g5),total=sum(g1+g2+g3+g4
  +g5),D.DealerId,Mth,D.CompanyName from n2,Dealer D WHERE N2.DealerId=D.DealerId
  group by D.DealerId,Mth,CompanyName
end
CREATE procedure [dbo].[CheckBankHolidays](@today datetime)
as
declare @days as int
declare @daysrecur as int
begin
select @Days=COUNT(*) from BankHoliday where HolidayDate between GETDATE() and
@today
select @daysrecur=COUNT(*) from BankHoliday where
CONVERT(datetime,CONVERT(varchar(4),YEAR(@today))+'/'+CONVERT(varchar(2),mont
h(HolidayDate))+'/'+CONVERT(varchar(2),day(HolidayDate))) between GETDATE() and
@today and Recurring=1
set @days=isnull(@days,0)+isnull(@daysrecur,0)
select Days=@days
end
CREATE PROCEDURE [dbo].[coolOffOrders]

```

```

AS
  DECLARE @coolingTime INT;
  SELECT @coolingTime = CONVERT(INT, Value) FROM SystemSetting WHERE NAME
= 'coolingTime'
  UPDATE CustomerOrderLine SET status = 4 where status = 2 AND CustomerOrderId IN
(SELECT ID FROM CustomerOrder WHERE IsDraft = 0 AND DATEDIFF( dd , OrderDate,
getdate()) >= @coolingTime)

create procedure [dbo].[DeleteCompleteDraft](@OrdId int) as
begin
declare @ordlineid int
select @ordlineid=isnull(Id,0) from [CustomerOrderLine] where [CustomerOrderId]=@OrdId
if @ordlineid>0
begin
delete from dbo.CustOrdLineSelected where CustOrdLineId=@ordlineid
delete from dbo.CustomerOrderLineDet where CustOrdLineId=@ordlineid
delete from dbo.CustomerOrderLine where Id=@ordlineid
end
delete from dbo.CustomerOrder where Id=@OrdId
end
CREATE PROCEDURE [dbo].[deletegroup]
@GrpId int
AS
BEGIN
update [group] set status =0 where groupid = @GrpId
END
create procedure [dbo].[DeleteOrderLine](@OrdLineId int) as
begin
if @OrdLineId>0
begin
delete from dbo.CustOrdLineSelected where CustOrdLineId=@ordlineid
delete from dbo.CustomerOrderLineDet where CustOrdLineId=@ordlineid
delete from dbo.CustomerOrderLine where Id=@ordlineid
end
end
CREATE procedure [dbo].[DeleteRecentPlaces](@userId int) as
begin
delete from RecentPlaces where rp_user=@userId
end
CREATE procedure [dbo].[DelVolBonComm](@dealerid int,@basegroupid
int,@GroupName varchar(250))
as
begin
if(@basegroupid>0)
begin
delete from cmnVolumeMatrix where dealerid=@dealerid and
basegroupid=@basegroupid and groupname=@GroupName
end
else
begin
delete from cmnVolumeMatrix where dealerid=@dealerid and groupname=@GroupName
end
end
Create Procedure [dbo].[ExceptionLoggingToDataBase]
(

```

```

@ExceptionMsg varchar(100)=null,
@ExceptionType varchar(100)=null,
@ExceptionSource nvarchar(max)=null,
@ExceptionURL varchar(100)=null
)
as
begin
Insert into Tbl_ExceptionLoggingToDataBase
(
ExceptionMsg ,
ExceptionType,
ExceptionSource,
ExceptionURL,
Logdate
)
select
@ExceptionMsg,
@ExceptionType,
@ExceptionSource,
@ExceptionURL,
getdate()
End
CREATE PROCEDURE [dbo].[FuzzySearchCustomerAddress](@SEARCHTERMS
varchar(200))
AS
BEGIN
SELECT [Id] from [Customer]
join containstable ([Customer],
([ContactAddressLine1],
[BillingAddressLine1],
[ContactAddressLine2],
[BillingAddressLine2],
[ContactAddressLine3],
[BillingAddressLine3],
[ContactAddressLine4],
[BillingAddressLine4],
[ContactAddressLine5],
[BillingAddressLine5],
[ContactPostcode],
[BillingPostcode]
), @SEARCHTERMS, LANGUAGE 'English') as FreeTextTableResult
on [Customer].[Id] = [FreeTextTableResult].[key]
order by [Rank] desc
END
CREATE PROCEDURE [dbo].[FuzzySearchCustomerContactPoints](@SEARCHTERMS
varchar(200))
AS
BEGIN
SELECT [Id] from [Customer]
join containstable ([Customer],
(
[MainTelephone],
[OtherTelephone1],
[OtherTelephone2],
[MobileTelephone],

```



```

[FaxNumber],
[EmailAddress]
), @SEARCHTERMS, LANGUAGE 'English') as FreeTextTableResult
on [Customer].[Id] = [FreeTextTableResult].[key]
order by [Rank] desc
END
CREATE PROCEDURE [dbo].[FuzzySearchCustomerName](@SEARCHTERMS
varchar(200))
AS
BEGIN
SELECT [Id] from [Customer]
join containstable ([Customer],
(CompanyName, FirstName, LastName, MiddleNames, Title)
, @SEARCHTERMS, LANGUAGE 'English') as FreeTextTableResult
on [Customer].[Id] = [FreeTextTableResult].[key]
order by [Rank] desc
END
CREATE PROCEDURE
[dbo].[FuzzySearchCustomerReferenceNumberMain](@SEARCHTERM varchar(200))
AS
BEGIN
SELECT [Id] from [Customer]
WHERE CONVERT(varchar(20), CustomerCodeNum) like @SEARCHTERM
END
CREATE PROCEDURE
[dbo].[FuzzySearchCustomerReferenceNumbers](@SEARCHTERMS varchar(200))
AS
BEGIN
SELECT [Id],[Rank] from [Customer]
join containstable ([Customer],
(
[ExternalReferenceNumber],
[BankAccountNumber],
[BankSortCode],
[CompanyRegNo],
[VatRegNo]
), @SEARCHTERMS, LANGUAGE 'English') as FreeTextTableResult
on [Customer].[Id] = [FreeTextTableResult].[key]
ORDER BY [RANK] desc
END
CREATE PROCEDURE [dbo].[FuzzySearchOrderReferenceNumber](@SEARCHTERM
varchar(200))
AS
BEGIN
SELECT [Id] from [CustomerOrder]
WHERE OrderReferenceString like @SEARCHTERM
END
create procedure [dbo].[GerCustomerOrdersNewStatus] as
begin
select co.* from CustomerOrder co, CustomerOrderLine col where co.IsDraft=0 and
col.CustomerOrderId=co.Id and col.Status=1
end
CREATE Procedure [dbo].[GetActionTasksByFilter](@DealerId int, @StatusId
varchar(10), @RoleId int, @QueueId int, @UserId int)
as

```

```

begin
declare @filter varchar(500)
DECLARE @script VARCHAR(MAX);
declare @OpenQ varchar(100);
declare @EndQ varchar(5);
truncate table ActionTasksByFilterResult
set @OpenQ='insert into ActionTasksByFilterResult select * from ( '
set @EndQ=' ) t'
set @filter=' '
    set @script='select
a.Id,a.QueueId,a.OwnedById,a.RaisedById,a.ActionedById,a.RaisedDate,a.ActionedDate,St
atus=(case when a.DueDate<getdate() and a.status='+'A'+ ' then '+'O' + ' else a.Status
end),a.Summary,a.Details,a.CallingPageId,a.SubjectId,a.DueDate,a.TaskType,a.TaskOutco
me,a.OutcomeNotes,a.CLI,a.CreatedOn,a.RelateTo,a.RelateToURL,u.UserId,u.UserName,u
.Password,u.FirstName,u.LastName,u.EmailID,u.ContactNum,u.CreatedOn as
uCreatedOn,u.Status as
ustatus,u.RoleID,u.GroupId,u.IsRemoved,u.Department,u.Region,u.UpdatedOn,u.DealerId
from dbo.ActionTask a,tbluser u where a.ownedbyid=u.UserId '
if @DealerId>0
begin
set @filter += ' and dealerid='+ cast(@dealerid as varchar(50))
end
if @StatusId<>'0'
begin
set @filter += ' and (case when a.DueDate<getdate() and a.status='+'A'+ ' then '+'O' + '
else a.Status end)= "'+ @StatusId + "'
end
if @RoleId>0
begin
set @filter += ' and Roleid='+ cast(@RoleId as varchar(50))
end
if @QueueId>0
begin
set @filter += ' and QueueId='+ cast(@QueueId as varchar(50))
end
if @UserId>0
begin
set @filter += ' and ownedbyid =' + cast(@UserId as varchar(50))
end
SET @script =@OpenQ + @script+@filter + @EndQ
--select @script
EXEC (@script);
--update ActionTasksByFilterResult set Status ='O' where DueDate<GETDATE() and
Status='A'
    select A.*,tu.UserName OwnedByName ,ts.TStatusName StatusName from
ActionTasksByFilterResult A ,tblUser tu,TaskStatus ts where a.OwnedById=tu.UserId and
ts.Id=A.Status
end
CREATE Procedure [dbo].[GetActionTasksByFilterNew](@DealerId int,@StatusId
varchar(10),@RoleId int,@QueueId int, @UserId int) as
declare @filter varchar(500)
DECLARE @script VARCHAR(MAX);
begin
set @filter=' '

```

```

set @script='select
a.Id,a.QueueId,a.OwnedById,a.RaisedById,a.ActionedById,a.RaisedDate,a.ActionedDate,a.
Status,a.Summary,a.Details,a.CallingPageId,a.SubjectId,a.DueDate,a.TaskType,a.TaskOutc
ome,a.OutcomeNotes,a.CLI,a.CreatedOn,u.UserId,u.UserName,u.Password,u.FirstName,u.
LastName,u.EmailId,u.ContactNum,u.CreatedOn as uCreatedOn,u.Status as
ustatus,u.RoleId,u.GroupId,u.IsRemoved,u.Department,u.Region,u.UpdatedOn,u.DealerId
from dbo.ActionTask a,tbluser u where a.RaisedById=u.UserId '
if @DealerId>0
begin
set @filter +=='and dealerid='+ cast(@dealerid as varchar(50))
end
if @StatusId<>'0'
begin
set @filter +=='and a.Status= "'+ @StatusId +'"
end
if @RoleId>0
begin
set @filter +=='and Roleid='+ cast(@RoleId as varchar(50))
end
if @QueueId>0
begin
set @filter +=='and QueueId='+ cast(@QueueId as varchar(50))
end
if @UserId>0
begin
set @filter +=='and RaisedById =' + cast(@UserId as varchar(50))
end
SET @script =@script+@filter
select @script as query
--EXEC (@script);
--select * from dbo.ActionTask a,tbluser u where a.RaisedById=u.UserId
end
create procedure [dbo].[GetAllNotes](@SubjectId int,@callingpage int)
as
Begin
if @callingpage=1
begin
select * from (
select NoteSource='Customer',* from note where callingpageid=1 and subjectid =
@SubjectId
union
select NoteSource='Customer Order('+ b.orderreferencestring +')',a.* from note a
left join customerorder b on
a.subjectid=b.id
where a.callingpageid=2 and b.customerid = @SubjectId ) t order by notedate
end
else if @callingpage=2
begin
select * from (
select NoteSource='Customer Order('+ b.orderreferencestring +')',a.* from note a
left join customerorder b on
a.subjectid=b.id
where a.callingpageid=2 and b.id = @SubjectId
union
select NoteSource='Customer',a.* from note a

```

```

left join customerorder b on
a.subjectid=b.customerid
where a.callingpageid=2 and b.id = @SubjectId ) t order by notedate
end
end
CREATE PROC [dbo].[GetAllPermissionGroups]
AS
BEGIN
SELECT * FROM tblPermissionGroup
END
CREATE procedure [dbo].[GetAllSpecialRequirementOrders](@Status int,@dealer int ) as
if @dealer=1
begin
SELECT CustomerOrderLine.* FROM CustomerOrderLine,CustomerOrder
where ISNULL([IsSpecialRequirements],0)=0 and OrderType=1 and Status>=@Status and
CustomerOrder.Id=CustomerOrderLine.CustomerOrderId
end
else
begin
SELECT CustomerOrderLine.* FROM CustomerOrderLine,CustomerOrder
where ISNULL([IsSpecialRequirements],0)=0 and OrderType=1 and Status>=@Status and
CustomerOrder.Id=CustomerOrderLine.CustomerOrderId and DealerId=@dealer
end
CREATE procedure [dbo].[GetAllSpecialRequirementOrdersPT](@Status int,@dealer int )
as
if @dealer=1
begin
SELECT col.Id,dbo.Fun_CustFullName(col.CustomerOrderId) CustName,
dbo.fn_GetOrderReferenceString(col.CustomerOrderId)
OrderReferenceString,col.CustomerOrderId,

col.IsSpecialRequirements,col.OrderDate,SpceialReqs=dbo.GetSpecialRequirementsById(c
ol.Id)
FROM CustomerOrderLine col,CustomerOrder co
where dbo.[GetSpecialRequirementsCountById](col.Id)>0 and OrderType=1 and Status>=4
and Status<>11 and
co.Id=col.CustomerOrderId order by co.OrderDate desc
end
else
begin
SELECT col.Id,dbo.Fun_CustFullName(col.CustomerOrderId) CustName,
dbo.fn_GetOrderReferenceString(col.CustomerOrderId)
OrderReferenceString,col.CustomerOrderId,
col.IsSpecialRequirements,col.OrderDate
FROM CustomerOrderLine col,CustomerOrder co
where dbo.[GetSpecialRequirementsCountById](col.Id)>0 and OrderType=1 and
Status>=@Status and Status<>11 and
co.Id=col.CustomerOrderId and DealerId=@dealer order by co.OrderDate desc
end
select * from CustomerOrderLine where IsSpecialRequirements<>0
create procedure [dbo].[GetAttachment](@id int)
as
begin
select * from Note where id=@id
end

```

```

create procedure [dbo].[GetAuditTrailbyPageld] (@Id int,@Pageld int) as
begin
if @Pageld=171 or @Pageld=62
begin
select * from AuditTrail where (Pageld=171 or Pageld=62 ) and ObjectId=@Id order by
CreatedOn desc
end
else if @Pageld=131 or @Pageld=172
begin
select * from AuditTrail where (Pageld=131 or Pageld=172 ) and ObjectId=@Id order by
CreatedOn desc
end
else if @Pageld=177 or @Pageld=179 or @Pageld=204 or @Pageld=37
begin
select * from AuditTrail where (Pageld=177 or Pageld=179 or Pageld=204 or Pageld=37 )
and ObjectId=@Id order by CreatedOn desc
end
else if @Pageld=0
begin
select * from AuditTrail order by CreatedOn desc
end
else
begin
select * from AuditTrail where ObjectId=@Id order by CreatedOn desc,Id desc
end
end
CREATE procedure [dbo].[getbrandfromcustomer](@custid int) as
begin
select b.shortcode,b.brandname from Brand b,Customer c
where b.Id=c.BrandId and c.Id=@custid
end
CREATE procedure [dbo].[GetCommonCounts](@RoleID int,@DealerID int,@UserID int)
AS
BEGIN
select
[dbo].[GetSalesMadeByType](@RoleID,@DealerID,@UserID,0) as Today,
[dbo].[GetSalesMadeByType](@RoleID,@DealerID,@UserID,1) Yesterday,
[dbo].[GetSalesMadeByType](@RoleID,@DealerID,@UserID,2) Last7Days,
[dbo].[GetSalesMadeByType](@RoleID,@DealerID,@UserID,3) Last14Days,
[dbo].[GetSalesMadeByType](@RoleID,@DealerID,@UserID,4) Last28Days,
[dbo].[GetSalesMadeByType](@RoleID,@DealerID,@UserID,5) ThisMonth,
[dbo].[GetSalesMadeByType](@RoleID,@DealerID,@UserID,6) ThisYear,
[dbo].[GetSalesMadeByType](@RoleID,@DealerID,@UserID,7) AllTime,
[dbo].[GetOpenDraftCount](@RoleID,@DealerID,@UserID) GetOpenDraftCount,
[dbo].[GetPipelineCount](@RoleID,@DealerID,@UserID,6) Provisioned,
[dbo].[GetPipelineCount](@RoleID,@DealerID,@UserID,9) Live,
[dbo].[GetPipelineCount](@RoleID,@DealerID,@UserID,2) Collin ff,
[dbo].[GetPipelineOnHoldCount](@RoleID,@DealerID,@UserID,6) ProvisionedOnHold,
[dbo].[GetPipelineOnHoldCount](@RoleID,@DealerID,@UserID,2) Collin ffOnHold,
([dbo].[GetSalesMadeByType](@RoleID,@DealerID,@UserID,2)/7.0)
SalesperDayLast7Days,
([dbo].[GetSalesMadeByType](@RoleID,@DealerID,@UserID,4)/28.0)
SalesperDayLast28Days,
[dbo].[GetCancellationsCountPerType](@RoleID,@DealerID,@UserID,7)
CancellationPerLast7Days,

```

```

[dbo].[GetCancellationsCountPerType](@RoleID,@DealerID,@UserID,28)
CancellationPerLast28Days
END
create procedure [dbo].[GetCommPairingRecords]
as
begin
select cp.*,bg.product from cmnCommPairing cp left join cmnbasegroups bg on
cp.basegroupid=bg.basegroupid
end
create procedure [dbo].[GetCorrespondeceInfoByCustID](@CustID int)
as
BEGIN
select ch.* from CorrespondenceHTML ch left join customerorder co
on ch.customerorderid=co.id
where co.customerid=@CustID
END
create PROCEDURE [dbo].[GetCorrespondenceLetters]
AS
BEGIN
select BD.brandname,CC.name classname,CT.* from CorrespondenceTemplate CT left join
brand BD on
CT.brandid=BD.id
left join CorrespondenceClass CC on
CT.Correspondenceclassid=CC.id
END
CREATE procedure [dbo].[GetCustomerOrders](@otype int,@urole int,@dealerid
int,@userid int) as
begin

if @userid=1
select * from dbo.CustomerOrder where IsDraft=@otype --and customerid is not null
else
select * from dbo.CustomerOrder where EnteredById=@userid and IsDraft=@otype
-- and customerid is not null
end
create procedure [dbo].[GetCustomerOrdersNewStatus] as
begin
select co.* from CustomerOrder co,CustomerOrderLine col where co.IsDraft=0 and
col.CustomerOrderId=co.Id and col.Status=1
end
Create procedure [dbo].[GetCustOrdOSWelcomCallPT](@urole int,@dealer int,@userid int
,@brandid int) as
begin
if (@urole = 1)
begin
if (@brandid = 0)
begin
select
dbo.Fun_GetCustomerAge(CustomerId) Age,[dbo].[GetCustomerBrandId](CustomerId)
Brandid,OrderReferenceString,dbo.[Fun_GetUserName](EnteredById) SalesMan,OrderDate
from CustomerOrder where WelcomeCallOutcome is null and
IsDraft = 0
end
else

```

```

Begin
    select * from (select
        Id,dbo.Fun_CustFullName(id) CustName,
        dbo.Fun_GetCustomerAge(CustomerId) Age,[dbo].[GetCustomerBrandId](CustomerId)
        BrandId,OrderReferenceString,dbo.[Fun_GetUserName](EnteredById) SalesMan,OrderDate
        from CustomerOrder where WelcomeCallOutcome is null and
        IsDraft = 0 ) t where BrandId=@brandid
    end
end
else if (@urole > 1 AND @urole <= 3)
begin
    if (@brandid = 0)
    begin
        select
            Id,dbo.Fun_CustFullName(id) CustName,
            dbo.Fun_GetCustomerAge(CustomerId) Age,[dbo].[GetCustomerBrandId](CustomerId)
            BrandId,OrderReferenceString,dbo.[Fun_GetUserName](EnteredById) SalesMan,OrderDate
            from CustomerOrder where WelcomeCallOutcome is null and
            IsDraft = 0 and dealerid=@dealer
        end
    else
    begin
        select * from (select
            Id,dbo.Fun_CustFullName(id) CustName,
            dbo.Fun_GetCustomerAge(CustomerId) Age,[dbo].[GetCustomerBrandId](CustomerId)
            BrandId,OrderReferenceString,dbo.[Fun_GetUserName](EnteredById) SalesMan,OrderDate
            from CustomerOrder where WelcomeCallOutcome is null and
            IsDraft = 0 and dealerid=@dealer ) t where BrandId=@brandid
        end
    end
end
else
begin
    if (@brandid = 0)
    begin
        select
            Id,dbo.Fun_CustFullName(id) CustName,
            dbo.Fun_GetCustomerAge(CustomerId) Age,[dbo].[GetCustomerBrandId](CustomerId)
            BrandId,OrderReferenceString,dbo.[Fun_GetUserName](EnteredById) SalesMan,OrderDate
            from CustomerOrder where WelcomeCallOutcome is null and
            IsDraft = 0 and enteredbyid=@userid
        end
    else
    begin
        select * from (select
            Id,dbo.Fun_CustFullName(id) CustName,
            dbo.Fun_GetCustomerAge(CustomerId) Age,[dbo].[GetCustomerBrandId](CustomerId)
            BrandId,OrderReferenceString,dbo.[Fun_GetUserName](EnteredById) SalesMan,OrderDate
            from CustomerOrder where WelcomeCallOutcome is null and
            IsDraft = 0 and enteredbyid=@userid ) t where BrandId=@brandid
        end
    end
end
end
CREATE PROCEDURE [dbo].[GetEligibleCustOrderLines4Commission](@DealerId int=0)
AS

```

```

BEGIN
if(@DealerId>0)
begin
select col.id
custordlineid,col.packagename,col.customerorderid,col.status,dbo.fn_getBaseGroupName(c
ol.packagename) BaseGroup,
dbo.fn_getBaseGroupCommission(col.packagename)
ConnComm,co.customerid,dbo.Fun_CustFullName(col.customerorderid)CustomerName,co.
orderreferencestring OrderRef,
col.orderdate OrderDate, co.commissionable,co.commissionpaymentid,
dbo.GetCust_orderlinecli(col.id) CLI,
co.commissionclawbackid,co.recordingsavailableforqa, co.dealerid, 'True' Connection,
'False' Clawback,
col.FinishedDate,col.Actual
LiveDate,dbo.[GetRecordingFileByOrdID](col.customerorderid)RecFileUrl,
col.CancelledDate,dbo.fn_GetDealerNameByID(co.dealerid)DealerName
from dealer d
inner join customerorder co on d.dealerid=co.dealerid
inner join customerorderline col on co.id=col.customerorderid
where
d.autoprocesscommissions=1
and (co.onholdstartdate is null or (co.onholdstartdate is not null and co.onholdenddate <=
getdate()))
and col.status=9
and (replace(packagename,' ','')<>" and packagename is not null)
and d.dealerid=@DealerId
end
--else
-- begin
-- select col.id custordlineid,col.packagename,col.customerorderid,col.status,
-- co.customerid,co.orderreferencestring,co.commissionable,co.commissionpaymentid,
-- co.commissionclawbackid,co.recordingsavailableforqa, co.dealerid
-- from dealer d
-- inner join customerorder co on d.dealerid=co.dealerid
-- inner join customerorderline col on co.id=col.customerorderid
-- where
-- d.autoprocesscommissions=1
-- and (co.onholdstartdate is null or (co.onholdstartdate is not null and co.onholdenddate <=
getdate()))
-- and col.status=9
-- and (replace(packagename,' ','')<>" and packagename is not null)
-- end
END
CREATE PROCEDURE [dbo].[GetEmptyPackageNameOrderlines]
AS
BEGIN
select col.id,col.customerorderid,co.orderreferencestring,co.orderdate,
co.dealerid,dbo.fn_GetDealerNameByID(co.dealerid)
dealer,col.status,col.packagename,col.summary
from customerorder co, customerorderline col
where
col.customerorderid=co.id
and dealerid in (select dealerid from dealer where autoprocesscommissions=1)
and (onholdstartdate is null or onholdstartdate is not null and onholdenddate <= getdate())
and status=9

```



```

and (packagename is null or packagename='')
END
CREATE procedure [dbo].[GetFinalCustomerOrders](@urole int=0,@dealerid int=0,@userid
int=0,
 @status int=0,@onhold int=0,@brand int=0,@custtype int=0,@daterng int=0,@dealer
int=0) as
declare @filter varchar(500)
declare @frdate varchar(50)
declare @todate varchar(50)
set @filter=' '
if @status<>0
begin
set @filter=' and l.status='+ cast(@status as varchar(50))
end
if @onhold=1
begin
set @filter=@filter+' and OnHoldStartDate is not null and l.status<>11 '
end
if @brand<>0
begin
set @filter=@filter+' and brandid='+cast(@brand as varchar(50))
end
if @custtype<>0
begin
if @custtype=1
set @filter=@filter+' and IsBusiness=0'
else if @custtype=2
set @filter=@filter+' and IsBusiness=1'
end
if @dealer<>0
begin
set @filter=@filter+' and o.dealerid='+cast(@dealer as varchar(50))
end
if @daterng = 1
begin
select @frdate=CONVERT(date,GETDATE()) , @todate=CONVERT(date,GETDATE())
end
else if @daterng = 2
begin
select @frdate=dateadd(d,-1,CONVERT(date,GETDATE())), @todate=dateadd(d,-
1,CONVERT(date,GETDATE()))
end
else if @daterng = 3
begin
select @frdate=dateadd(w,-1,CONVERT(date,GETDATE())) ,
@todate=CONVERT(date,GETDATE())
end
else if @daterng = 4
begin
select @frdate=dateadd(w,-1,dateadd(d,-7,CONVERT(date,GETDATE())))) ,
@todate=dateadd(w,-2,CONVERT(date,GETDATE()))
end
if @daterng = 5
begin

```

```

select @frdate=dateadd(d,-8,CONVERT(date,GETDATE())) , @todate=dateadd(d,-
1,CONVERT(date,GETDATE()))
end
else if @daterng = 6
select @frdate=dateadd(d,-31,CONVERT(date,GETDATE())) , @todate=dateadd(d,-
1,CONVERT(date,GETDATE()))
if @daterng = 7
begin
select @frdate=dateadd(D,-DAY(GETDATE()-1),CONVERT(date,GETDATE())) ,
@todate=CONVERT(date,GETDATE())
end
else if @daterng = 8
begin
select @frdate=dateadd(D,-DAY(GETDATE()),CONVERT(date,GETDATE()))
select @frdate=dateadd(D,-(DAY(@frdate)-1),CONVERT(date,@frdate)) ,
@todate=dateadd(D,-DAY(GETDATE()),CONVERT(date,GETDATE()))
end
else if @daterng = 9
begin
select @frdate=CONVERT(date,convert(varchar(4),year(getdate()))+'-01-01') ,
@todate=CONVERT(date,GETDATE())
end
else if @daterng = 10
begin
select @frdate=CONVERT(date,convert(varchar(4),year(getdate())-1)+'-01-01') ,
@todate=CONVERT(date,convert(varchar(4),year(getdate())-1)+'-12-31')
end
if @daterng<>0
begin
set @filter=@filter+' and CONVERT(date,o.orderdate) between "' + @frdate + "' and "' +
@todate + "'"
end
--select @filter as qry
DECLARE @script VARCHAR(MAX);
set @script='select distinct
o.Id,o.CustomerId,o.OrderDate,o.OrderReferenceString,c.IsBusiness,c.BrandId,onhold=(cas
e o.OnHoldStartDate when null then 0 else case o.onholdenddate when null then 1 else 2
end end ) ,status=l.Status,StatusName=e.Name ,CustomerName= c.FirstName + " " +
c.MiddleNames + " " + c.LastName
,b.brandname,tu.username,dealername=d.CompanyName, PDFGenerated=(select count(*)
from CustomerOrderWelcomePack where customerorderid=o.id) from CustomerOrder
o,Customer c,CustomerOrderLine l,dbo.CustomerOrderLineStatusEnum e ,brand b,dealer
d,tbluser tu where o.Id=l.CustomerOrderId and o.CustomerId=c.Id and l.Status=e.id and
o.IsDraft=0 and tu.userid=o.EnteredById and d.DealerId=o.DealerId and b.id=c.BrandId'
if @urole=1
SET @script =@script + @filter
else if @urole=2 or @urole=3
SET @script = @script+' and c.DealerId='+ convert(varchar(10),@dealerid) + @filter
else
SET @script =@script+' and EnteredById='+ convert(varchar(10),@userid) + @filter
--select @script as query
EXEC (@script);
--select distinct
o.Id,o.CustomerId,o.OrderDate,o.OrderReferenceString,c.IsBusiness,c.BrandId,onhold=(cas
e o.OnHoldStartDate when null then 0 else case o.onholdenddate when null then 1 else 2

```

```

end end ) ,status=l.Status,StatusName=e.Name ,CustomerName= c.FirstName + ' ' +
c.MiddleNames + ' ' + c.LastName
,b.brandname,tu.username,dealername=d.CompanyName, PDFGenerated=(select count(*)
from CustomerOrderWelcomePack where customerorderid=o.id) from CustomerOrder
o,Customer c,CustomerOrderLine l,dbo.CustomerOrderLineStatusEnum e ,brand b,dealer
d,tbluser tu where o.Id=l.CustomerOrderId and o.CustomerId=c.Id and l.Status=e.id and
o.IsDraft=0 and tu.userid=o.EnteredById and d.DealerId=o.DealerId and b.id=c.BrandId and
l.status=4

```

```

-- =====
-- Author:          <Author,,Name>
-- Create date: <Create Date,,>
-- Description:<Description,,>
-- =====

```

```

CREATE PROCEDURE [dbo].[GetLastPageinPageTree]
@PageID Varchar(100)
AS
BEGIN
WITH n(Pageld, Pagename, [level], parentId, concatenador) AS
(
SELECT Pageld, Pagename,[level],parentId, (''+CONVERT(VARCHAR (MAX), Pageld)+'
- '+CONVERT(VARCHAR (MAX), [level])+')' as concatenador
FROM tblpages
WHERE Pageld = @PageId
--and PageUrl='administrator'
UNION ALL
SELECT m.Pageld, m.Pagename,m.level,m.parentId,
n.concatenador+' * ('+CONVERT (VARCHAR (MAX), case when ISNULL(m.parentId, 0)
= 0 then 0 else m.Pageld END)+' - '+CONVERT(VARCHAR (MAX), m.level)+')' as
concatenador
FROM tblpages as m, n
WHERE n.Pageld = m.parentId and m.isMenuItem =1
)
SELECT distinct * FROM n ORDER BY concatenador Desc
END

```

```

CREATE PROCEDURE [dbo].[GetLeftButtons](@pageid int,@groupid int)
AS
BEGIN
/*
lbtntype
//1= Action
//2= WorkPlace
//3=Filter (Common values)
//4=Filter (user based values)
*/
select
pb.pagebtnid,lb.lbtid,lb.lbtname,lb.lbtimage,lbtcode,lb.lbturl,urlpageid,pagename,
pageurl,lb.lbtnpriority,lb.lbtid,pb.show,pb.priority,lb.lbtntype,pb.groupid,pb.maptype,pb.map
description
from tblLeftButtons lb,tblPageButtonMapping pb,tblpages tp
where lb.lbtid=pb.lbtid and lb.isactive=1 and pb.isactive=1 and tp.pageid=pb.pageid
and pb.pageid=@pageid and pb.groupid=@groupid
--tp.pageid=pb.urlpageid
order by lbtnpriority
----union all

```

```

---- select
pagebtnid=pcatid+1,lbtnid=pcatid+1,lbtnname=[PcatDesc],lbtnimage='#',lbtncode='URL',lbtn
url='#',urlpageid=1,pagename='Search',
----
pageurl='OrderSearchList.aspx?pageid=2&pid='+convert(varchar(10),pcatid),lbtnpriority=10,l
btnid=2,show=1,priority=1,lbtnntype=2,groupid=1,maptype=1,mapdescription='maptype1=Lin
kButton,maptype2=HTML or Script'
--from [dbo].[ProductCat]
END
CREATE PROCEDURE [dbo].[GetLeftButtonsbylbtnid](@lbtnid int)
AS
BEGIN
/*
lbtnntype
//1= Action
//2= WorkPlace
//3=Filter (Common values)
//4=Filter (user based values)
*/
select
lb.lbtnid,lb.lbtnname,lb.lbtnimage,lbtncode,lb.lbtnurl,urlpageid,pagename,pageurl,lb.lbtnpriori
ty,lb.lbtnid,pb.show,priority=min(pb.priority),lb.lbtnntype,pb.groupid
from
tblLeftButtons lb,tblPageButtonMapping pb,tblpages tp
where
lb.lbtnid=pb.lbtnid and lb.isactive=1 and tp.pageid=pb.urlpageid and
lb.lbtnid=@lbtnid
group by
lb.lbtnid,lb.lbtnname,lb.lbtnimage,lbtncode,lb.lbtnurl,urlpageid,pagename,pageurl,lb.lbtnpriori
ty,lb.lbtnid,pb.show,lb.lbtnntype,pb.groupid
order by lbtnpriority
END
CREATE PROCEDURE [dbo].[GetLeftButtonsbyPagebtnid](@pagebtnid int)
AS
BEGIN
/*
lbtnntype
//1= Action
//2= WorkPlace
//3=Filter (Common values)
//4=Filter (user based values)
*/
select
pb.pagebtnid,lb.lbtnid,lb.lbtnname,lb.lbtnimage,lbtncode,lb.lbtnurl,urlpageid,pagename,page
url,lb.lbtnpriority,lb.lbtnid,pb.show,pb.priority,lb.lbtnntype,pb.groupid
from
tblLeftButtons lb,tblPageButtonMapping pb,tblpages tp
where
lb.lbtnid=pb.lbtnid and lb.isactive=1 and tp.pageid=pb.urlpageid and
pb.pagebtnid=@pagebtnid
order by lbtnpriority
END
CREATE PROCEDURE [dbo].[GetLeftButtonsPermission](@pageid int,@groupid int)
AS
BEGIN

```

```

--select
lb.lbtnid,lb.lbtnname,lbtnimage,lbtnpriority,lbtntype,pbm.pagebtnid,pbm.pageid,pbm.lbtnid,pb
m.show,pbm.priority
--from dbo.tblLeftButtons lb, tblPageButtonMapping pbm
--where
--lb.lbtnid=pbm.lbtnid and pbm.pageid=@pageid and
--lb.isactive=1
--order by pbm.priority
select
lb.lbtnid,lb.lbtnname,lbtnimage,lbtnpriority,lbtntype,pbm.pagebtnid,pbm.pageid,pbm.lbtnid,pb
m.show,pbm.priority,pbm.groupid
from dbo.tblLeftButtons lb, tblPageButtonMapping pbm
where
lb.lbtnid=pbm.lbtnid and pbm.pageid=@pageid and
lb.isactive=1 and pbm.groupid=@groupid
--order by pbm.priority
END
CREATE PROCEDURE [dbo].[GetLeftButtonsProducts]( @pageid int, @groupid int)
AS
BEGIN
select
lbtnid=ProductId,lbtnname=ProductName,lbtnimage="",lbtnpriority=ProductId,lbtntype=1,pag
ebtnid=ProductGroupId,pageid=ProductGroupCate
ry,lbtnid=ProductId,show=1,priority=ProductId,groupid=ProductGroupId
from [dbo].[product]
where ProductGroupId=@groupid or ProductGroupId=@pageid
END
CREATE PROCEDURE [dbo].[GetLeftDDL]( @lbtnid int, @loginid int, @lbtntype int)
AS
BEGIN
/*
//3=Filter (Common values)
//4=Filter (user based values)
*/
declare @Roleid int
select @Roleid=Roleid from tblUser tu WHERE tu.UserId=@loginid
if ( @lbtntype=3)
begin
select lddid ID,lddvalue TEXT,lddpriority,lbtnid from dbo.tblLeftDDL where lbtnid=@lbtnid
and isactive=1
end
else if(@lbtntype=4)
begin
if @lbtnid=24 --Brand
begin
select Brand.Id ID ,Brand.BrandName TEXT,lddpriority=0, @lbtnid lbtnid from
Brand,tblUser,Dealer,DealerBrand
where brand.Id=DealerBrand.BrandId and DealerBrand.DealerId=Dealer.DealerId and
Dealer.DealerId=tblUser.DealerId and tblUser.userid=@loginid
end
else if @lbtnid=27 --user
begin
if (@Roleid=1)
begin

```

```

                select tu.UserId ID,tu.UserName TEXT ,Iddpriority=0,@Ibtnid
Ibtnid from tblUser tu order by tu.username
            end
        else if (@Roleid=2)
            begin
                select tu.UserId ID,tu.UserName TEXT ,Iddpriority=0,@Ibtnid
Ibtnid from tblUser tu ,tblUser tu1 WHERE tu.DealerId=tu1.DealerId and
tu1.UserId=@loginid order by tu.username
            end
        else
            begin
                select tu.UserId ID,tu.UserName TEXT ,Iddpriority=0,@Ibtnid
Ibtnid from tblUser tu WHERE tu.UserId=@loginid and RoleID>2 order by tu.username
            end
        end
    IF @Ibtnid=43 --QUEUE
        begin
            SELECT ID,NAME TEXT,Iddpriority=0,@Ibtnid FROM dbo.ActionQueue
        end
    if @Ibtnid=44 --role
        begin
            SELECT ID,NAME TEXT,Iddpriority=0,@Ibtnid FROM dbo.Role
        end
    if @Ibtnid=27 --user
        SELECT userid ID,UserName TEXT,Iddpriority=0,@Ibtnid FROM dbo.tblUser order by
username
    If(@Ibtnid=22)
        begin
            SELECT ID,NAME TEXT,Iddpriority=0,@Ibtnid FROM
dbo.CustomerOrderLineStatusEnum
        end
    If(@Ibtnid=60)
        begin
            SELECT ID,TStatusName TEXT,Iddpriority=0,@Ibtnid FROM dbo.TaskStatus
        end
    else if @Ibtnid=28 --dealer
        BEGIN
            if(@Roleid=1)
                begin
                    select dealer.DealerId ID,dealer.CompanyName TEXT ,Iddpriority=0,@Ibtnid Ibtnid from
dealer
                end
            else if (@Roleid=2)
                begin
                    select dealer.DealerId ID,dealer.CompanyName TEXT ,Iddpriority=0,@Ibtnid Ibtnid from
dealer,tblUser WHERE tbluser.DealerId=Dealer.DealerId and tbluser.UserId=@loginid
                end
            else
                begin
                    select dealer.DealerId ID,dealer.CompanyName TEXT ,Iddpriority=0,@Ibtnid Ibtnid from
dealer,tblUser WHERE tbluser.DealerId=Dealer.DealerId and tbluser.UserId=@loginid
                end
            end
        end
    end
end
end
end
end

```

```

create procedure [dbo].[GetLLUOrderfromOrderId](@orderid int) as
begin
    select OrderId=col.CustomerOrderId,
    orderlineid=col.Id,ordref=co.OrderReferenceString,cod.OrderLineCli
    from CustomerOrderLine col,customerorder co,CustomerOrderLineDet cod
    where col.CustomerOrderId=co.Id and cod.CustOrdLineId=col.Id and col.IsLLU=1
    and co.Id=@orderid
end
CREATE procedure [dbo].[GetMostRecentOrders](@RoleID int,@DealerID int,@UserID int)
AS
BEGIN
if(@RoleID=1)
begin
    select top 3 id, orderreferencestring,dbo.Fun_CustFullName(id) CustName,(select top 1
status from customerorderline where customerorderid=co.id) as status from customerorder
co where customerid> 0 and co.isdraft=0 order by createdon desc
end
else if(@RoleID=2 and @RoleID=3)
begin
    select top 3 id,orderreferencestring,dbo.Fun_CustFullName(id) CustName,(select top 1
status from customerorderline where customerorderid=co.id) as status from customerorder
co where customerid> 0 and co.isdraft=0 and dealerid=@DealerID order by createdon desc
end
else
begin
    select top 3 id,orderreferencestring,dbo.Fun_CustFullName(id) CustName,(select top 1
status from customerorderline where customerorderid=co.id) as status from customerorder
co where customerid> 0 and co.isdraft=0 and enteredbyid=@UserID order by createdon
desc
end
END
CREATE procedure [dbo].[GetMyShortList]( @Userid int,@stype bit )
as
begin
select
p.ProductId,p.ProductName,uc.UserId,SellerDescription,ps.Price,ps.CurrencyId,ps.PriceUnit
s,
ps.PriceUomId,PricePackId,uc.createdon,uc.PSId
from product p,productseller ps, UserShortList uc
where p.ProductId=ps.ProductId and uc.PSId=ps.PSId and uc.Active=1
and uc.UserId=@Userid and uc.SLType=@stype
end
CREATE procedure [dbo].[GetMyUserCart]( @Userid int )
as
begin
select
p.ProductId,p.ProductName,uc.UserId,SellerDescription,ps.Price,ps.CurrencyId,ps.PriceUnit
s,
ps.PriceUomId,PricePackId,uc.createdon,uc.UCId,uc.PayId,uc.PSId,ps.DiscountLink
from product p,productseller ps, UserCart uc
where p.ProductId=ps.ProductId and uc.PSId=ps.PSId and uc.Active=1 and uc.PayId=0
and uc.UserId=@Userid
end
CREATE procedure [dbo].[GetNextCORefId] AS
select NextCORefId=isnull(MAX(id),0)+1 from CustomerOrder

```

```

Create procedure [dbo].[GetNextCustomerCodeNum] AS
select NextCustomerCodeNum=isnull(MAX(Customecodenum),0)+1 from Customer
create procedure [dbo].[GetNextCustomerId] AS
select NextCustomerId=isnull(MAX(Id),0)+1 from Customer
CREATE procedure [dbo].[GetNextDealerId] AS
select NextDealerId=isnull(MAX(DealerId),0)+1 from dealer
create procedure [dbo].[GetNextGroupId] as
SELECT GroupId=isnull(max(GroupId),0)+1
FROM tblpermissionGroup
create procedure [dbo].[GetNextHoldId] AS
select NextHoldId=isnull(MAX(Id),0)+1 from HoldReason
create procedure [dbo].[GetNextHoldReasonId] AS
select NextHoldReasonId=isnull(MAX(Id),0)+1 from HoldReason
create procedure [dbo].[GetNextNoteId] AS
select NextNoteId=isnull(MAX(Id),0)+1 from Note
CREATE procedure [dbo].[GetNextOrderRefNumber] AS
select NextOrderReferenceNumber=isnull(MAX(OrderReferenceNumber),0)+1 from
CustomerOrder
create procedure [dbo].[GetNextOutstandingPrintRunExecuted]
as
begin
select isnull(max(osprunid),0)+1 osprunid from OutstandingPrintRunExecuted
end
CREATE procedure [dbo].[GetNextProductCatId] AS
select NextProductId=isnull(MAX(PCatId),0)+1 from ProductCat
create procedure [dbo].[GetNextProductCurrency] AS
select NextCurrencyid=isnull(MAX(CId),0)+1 from [dbo].[Currency]
CREATE procedure [dbo].[GetNextProductDelTimes] AS
select NextDelTimeid=isnull(MAX(DelTimeType),0)+1 from tbldeltime
CREATE procedure [dbo].[GetNextProductGroupId] AS
select NextProductId=isnull(MAX(pgid),0)+1 from ProductGroup
create procedure [dbo].[GetNextProductId] AS
select NextProductId=isnull(MAX(ProductId),0)+1 from product
create procedure [dbo].[GetNextProductPacking] AS
select NextPackID=isnull(MAX([PackID]),0)+1 from [dbo].[tblpacking]
create procedure [dbo].[GetNextProductUOM] AS
select NextUOMid=isnull(MAX(Uomid),0)+1 from tbluom
create procedure [dbo].[GetNextUserId] as
select NextUserId=MAX(UserId)+1 from [tblUser]
CREATE procedure [dbo].[GetNonLiveByStatus] (@Status int) as
begin
select distinct CustomerId=c.Id, CustomerRef=
c.ExternalReferenceNumber, CustomerOrderId=co.Id,col.IsCustomerAdded,col.IsBillingAPIS
uccess,custcode=c.ExternalReferenceNumber,
BillingSystemInvoiceNumber, FirstPaymentDate,
CompletionDate,co.InsertedIntoBillingSystemDate,
NoPayDays=DATEDIFF(d,isnull(FirstPaymentDate,GETDATE()),getdate())
from CustomerOrderLine col, CustomerOrder co, Customer c
where col.CustomerOrderId=co.Id and c.Id=co.CustomerId and
col.CustomerOrderId=co.Id and ISNULL(col.IsBillingAPISuccess ,0)=0
and col.Status= @Status
end
select DATEDIFF(d,'2014/01/01',getdate())
CREATE procedure [dbo].[GetOrderForCorrespondence]
as

```



```

begin
  select * from customerorder co where
  isdraft=0 and co.cancelleddate is null and
  id not in (select distinct customerorderid from CorrespondenceHTML ch where
  correspondencebatchid > 0 and correspondencetype=2 )
end
CREATE procedure [dbo].[GetOrderForCorrespondenceCancellation]
as
begin
  select * from customerorder co where
  isdraft=0 and co.cancelleddate is not null and
  id not in (select distinct customerorderid from CorrespondenceHTML ch where
  correspondencebatchid > 0 and correspondencetype=3 )
end
create procedure [dbo].[GetOrderfromOrderId](@orderid int) as
begin
  select OrderId=col.CustomerOrderId,
  orderlineid=col.Id,ordref=co.OrderReferenceString,cod.OrderLineCli
  from CustomerOrderLine col,customerorder co,CustomerOrderLineDet cod
  where col.CustomerOrderId=co.Id and cod.CustOrdLineId=col.Id
  and co.Id=@orderid
end
CREATE procedure [dbo].[getOSCcustomerOrderLinebyType](@ordertype int,@status
int,@dealer int) as
--ordertype=1.Migrate Existing Telephone,2.New Fixed Line ,3.Discount mobile bolton,
4.manual line
begin
if @ordertype=1
begin
  SELECT col.*,cod.OrderLineCli FROM dbo.CustomerOrderLine
col,dbo.CustomerOrderLineDet cod,CustomerOrder co
  where col.id=cod.CustOrdLineId and col.OrderType=1 and cod.connectiontype=1 AND
isnull(col.IsLLU,0)=0
  and co.Id=col.CustomerOrderId and col.Status=@status
  end
if @ordertype=2
begin
  SELECT col.*,cod.OrderLineCli FROM dbo.CustomerOrderLine
col,dbo.CustomerOrderLineDet cod,CustomerOrder co
  where col.id=cod.CustOrdLineId and col.OrderType=1 and cod.connectiontype=2
  and co.Id=col.CustomerOrderId and col.Status= @status
  end
if @ordertype=4
begin
  SELECT col.*,OrderLineCli=cod.OrderLineCli
FROM dbo.CustomerOrderLine col,CustomerOrder co,customerorderlinedet cod
  where col.OrderType=3 and col.Status= @status
  and co.Id=col.CustomerOrderId and col.Id=cod.CustOrdLineId
  end
if @ordertype=3
begin
  SELECT col.*,cod.OrderLineCli FROM dbo.CustomerOrderLine
col,dbo.CustomerOrderLineDet cod,CustomerOrder co
  where col.id=cod.CustOrdLineId and col.OrderType=2 and col.Status=@status
  and co.Id=col.CustomerOrderId

```

```

end
end
CREATE procedure [dbo].[getOutStandingCustomerOrders] (@status int,@dealer int)as
begin
if @dealer=1
begin
select Nos=count(*), ConnectionName=(case ordertype when 1 then ct.name else
olt.name end) ,col.status
from dbo.CustomerOrderLine col,OrderLineType olt,dbo.CustomerOrderLineDet
cod,dbo.ConnectionType ct,CustomerOrder co
where col.ordertype=olt.id and col.id=cod.CustOrdLineId and co.Id=col.CustomerOrderId
and
cod.ConnectionType=ct.id and isnull(col.IsLLU,0)=0 AND col.status=@status and
col.OrderType<>3
group by ordertype,ct.name,olt.name ,col.status
union all
select Nos=count(*), ConnectionName= olt.name ,col.status
from dbo.CustomerOrderLine col,OrderLineType olt,dbo.CustomerOrderLineDet
cod,CustomerOrder co
where col.ordertype=olt.id and col.id=cod.CustOrdLineId and co.Id=col.CustomerOrderId
and
isnull(col.IsLLU,0)=0 AND col.status=@status and col.OrderType=3
group by ordertype,olt.name ,col.status
end
else
begin
select Nos=count(*), ConnectionName=(case ordertype when 1 then ct.name else
olt.name end) ,col.status
from dbo.CustomerOrderLine col,OrderLineType olt,dbo.CustomerOrderLineDet
cod,dbo.ConnectionType ct,CustomerOrder co
where col.ordertype=olt.id and col.id=cod.CustOrdLineId and co.Id=col.CustomerOrderId
and
cod.ConnectionType=ct.id and isnull(col.IsLLU,0)=0 AND col.status=@status and
co.DealerId=@dealer
and col.OrderType<>3
group by ordertype,ct.name,olt.name ,col.status
union all
select Nos=count(*), ConnectionName= olt.name ,col.status
from dbo.CustomerOrderLine col,OrderLineType olt,dbo.CustomerOrderLineDet
cod,CustomerOrder co
where col.ordertype=olt.id and col.id=cod.CustOrdLineId and co.Id=col.CustomerOrderId
and
isnull(col.IsLLU,0)=0 AND col.status=@status and co.DealerId=@dealer and
col.OrderType=3
group by ordertype,olt.name ,col.status
end
end
CREATE procedure [dbo].[GetPackageAllProductDetById] (@id int) as
begin
select * from dbo.CallProduct where Id in (select cpid from PackagedProductDet where
ppid=@id and IsActive=1) and IsActive=1 and ProductGroupId>1
order by ProductGroupCate ry,ProductGroupId
end
CREATE procedure [dbo].[GetPackageAllProductDetNoBBById] (@id int) as
begin

```

```

select * from dbo.CallProduct where Id in (select cpid from PackagedProductDet where
ppid=@id and IsActive=1) and IsActive=1 and ProductGroupId>1
and PackageType<>2
order by ProductGroupCategory,ProductGroupId
end
create procedure [dbo].[GetPackageDetail] (@ppid int) as
select ppd.*,pd.* from
dbo.PackagedProductDet ppd,
dbo.product pd
where pd.productId=ppd.cpid and ppd.ppid= @ppid
create procedure [dbo].[GetPackagedProductData] as
SELECT p.*,c.* FROM PackagedProduct p,CallPackage c where p.prid=c.id
CREATE procedure [dbo].[GetPackageProductDetById] (@id int) as
SELECT P.PPid,p.CPid,PPActive=p.isactive ,name=c.ProductName,Price=c.Price
FROM PackagedProductDet P, dbo.product C
WHERE P.CPid=c.ProductId and ProductGroupId=1 and p.PPid=@id
-- =====
-- Author: <Author,,Name>
-- Create date: <Create Date,,>
-- Description:<Description,,>
-- =====
CREATE PROCEDURE [dbo].[GetPageTree]
@PageID Varchar(100)
AS
BEGIN
WITH n(Pageld, Pagename, [level], parentId, concatenador) AS
(
SELECT Pageld, Pagename,[level],parentId, ('+CONVERT(VARCHAR (MAX), Pageld)+'
- '+CONVERT(VARCHAR (MAX), [level]))+' as concatenador
FROM tblpages
WHERE Pageld = @PageID
--and PageUrl='administrator'
UNION ALL
SELECT m.Pageld, m.Pagename,m.level,m.parentId,
n.concatenador+' * ('+CONVERT (VARCHAR (MAX), case when ISNULL(m.parentId, 0)
= 0 then 0 else m.Pageld END)+' - '+CONVERT(VARCHAR (MAX), m.level)+'') as
concatenador
FROM tblpages as m, n
WHERE n.Pageld = m.parentId and m.isMenuItem =1
)
SELECT distinct * FROM n ORDER BY concatenador asc
END
CREATE PROCEDURE [dbo].[GetPageTreeForMenu]
@GroupID int
AS
BEGIN
if @GroupID>1
begin
WITH n(Pageld, Pagename, [level], parentId,PageCode,[priority],PageUrl,concatenador) AS
(
SELECT Pageld, Pagename,[level],parentId,PageCode,[priority],PageUrl,
'+CONVERT(VARCHAR (MAX), Pageld)+' - '+CONVERT(VARCHAR (MAX), [level]))+' as
concatenador
FROM tblpagesNew

```

```

WHERE ParentId=0 and isMenuItem =1 --Syed has added for ignoring the non-menu
pages
UNION ALL
SELECT m.Pageld,
m.Pagename,m.level,m.parentId,m.PageCode,m.[priority],m.PageUrl,
n.concatenador+' * ('+CONVERT (VARCHAR (MAX), case when ISNULL(m.parentId, 0)
= 0 then 0 else m.Pageld END)+' - '+CONVERT(VARCHAR (MAX), m.level)+')' as
concatenador
FROM tblpagesNew as m, n
WHERE n.Pageld = m.parentId
and m.isMenuItem =1 --Syed has added for ignoring the non-menu pages
)
select
distinct isNULL(MapId,0)MapId,
t1.Pageld,
t1.PageName,
[level],case when parentId is null then 0 else parentId end ParentId,
t1.concatenador,
t1.PageCode,
t1.[priority],
case when t1.PageUrl is null then 'No-Url' else PageUrl end PageUrl,
isNULL([view],'false')[view],
isNULL([Add],'false')[Add],
isNULL([Edit],'false')[Edit],
isNULL([Delete],'false')[Delete]
from n t1 Left Outer Join
(select MapId, Pageld ,
case when [view] =1 then 'true' else 'false' end [view],
case when [Add] =1 then 'true' else 'false' end [Add],
case when [Edit] =1 then 'true' else 'false' end [Edit],
case when [Delete] =1 then 'true' else 'false' end [Delete]
from tblgrpMappingnew t1 where t1.GroupId =@GroupId and t1.pageid<>11)t
on t1.Pageld = t.Pageld
-----By Syed to ignor null mapid--20141127-->>
where mapid is not null
-----By Syed to ignor null mapid--20141127--<<
ORDER BY t1.[priority],concatenador asc
end
else
begin
WITH n(Pageld, Pagename, [level], parentId,PageCode,[priority],PageUrl,concatenador) AS
(
SELECT Pageld, Pagename,[level],parentId,PageCode,[priority],PageUrl,
'+CONVERT(VARCHAR (MAX), Pageld)+' - '+CONVERT(VARCHAR (MAX), [level])+')' as
concatenador
FROM tblpagesNew
WHERE ParentId=0 and isMenuItem =1 --Syed has added for ignoring the non-menu
pages
UNION ALL
SELECT m.Pageld,
m.Pagename,m.level,m.parentId,m.PageCode,m.[priority],m.PageUrl,
n.concatenador+' * ('+CONVERT (VARCHAR (MAX), case when ISNULL(m.parentId, 0)
= 0 then 0 else m.Pageld END)+' - '+CONVERT(VARCHAR (MAX), m.level)+')' as
concatenador
FROM tblpagesNew as m, n

```

```

WHERE n.Pageld = m.parentld
and m.isMenuItem =1 --Syed has added for ignoring the non-menu pages
)
select
distinct isNULL(Mapld,0)Mapld,
t1.Pageld,
t1.PageName,
[level],case when parentld is null then 0 else parentld end Parentld,
t1.concatenador,
t1.PageCode,
t1.[priority],
case when t1.PageUrl is null then 'No-Url' else PageUrl end PageUrl,
isNULL([view],'false')[view],
isNULL([Add],'false')[Add],
isNULL([Edit],'false')[Edit],
isNULL([Delete],'false')[Delete]
from n t1 Left Outer Join
(select Mapld, Pageld ,
case when [view] =1 then 'true' else 'false' end [view],
case when [Add] =1 then 'true' else 'false' end [Add],
case when [Edit] =1 then 'true' else 'false' end [Edit],
case when [Delete] =1 then 'true' else 'false' end [Delete]
from tblgrpMappingnew t1 where t1.Groupld =@Groupld )t
on t1.Pageld = t.Pageld
-----By Syed to ingnor null mapid--20141127-->>
where mapid is not null
-----By Syed to ingnor null mapid--20141127--<<
ORDER BY t1.[priority],concatenador asc
end
END
CREATE PROCEDURE [dbo].[GetPageTreeForPage]
@Groupld int
AS
BEGIN
WITH n(Pageld, Pagename, [level], parentld,PageCode,[priority],PageUrl,concatenador) AS
(
SELECT Pageld, Pagename,[level],parentld,PageCode,[priority],PageUrl,
('+CONVERT(VARCHAR (MAX), Pageld)+' - '+CONVERT(VARCHAR (MAX), [level])+')' as
concatenador
FROM tblpages
WHERE Parentld=0 and isMenuItem =1 --Syed has added for ignoring the non-menu
pages
UNION ALL
SELECT m.Pageld,
m.Pagename,m.level,m.parentld,m.PageCode,m.[priority],m.PageUrl,
n.concatenador+' * ('+CONVERT (VARCHAR (MAX), case when ISNULL(m.parentld, 0)
= 0 then 0 else m.Pageld END)+' - '+CONVERT(VARCHAR (MAX), m.level)+')' as
concatenador
FROM tblpages as m, n
WHERE n.Pageld = m.parentld
and m.isMenuItem =1 --Syed has added for ignoring the non-menu pages
)
select
distinct isNULL(Mapld,0)Mapld,
t1.Pageld,

```

```

t1.PageName,
[level],case when parentId is null then 0 else parentId end ParentId,
t1.concatenador,
t1.PageCode,
t1.[priority],
case when t1.PageUrl is null then 'No-Url' else PageUrl end PageUrl,
isNULL([view],'false')[view],
isNULL([Add],'false')[Add],
isNULL([Edit],'false')[Edit],
isNULL([Delete],'false')[Delete]
from n t1 Left Outer Join
(select MapId, PageId ,
case when [view] =1 then 'true' else 'false' end [view],
case when [Add] =1 then 'true' else 'false' end [Add],
case when [Edit] =1 then 'true' else 'false' end [Edit],
case when [Delete] =1 then 'true' else 'false' end [Delete]
from tblgrpMapping t1 where t1.GroupId =@GroupId )t
on t1.PageId = t.PageId
-----By Syed to ingnor null mapid--20141127-->>
where mapid is not null
-----By Syed to ingnor null mapid--20141127--<<
ORDER BY t1.[priority],concatenador asc
END
CREATE procedure [dbo].[GetPendingBillsErrors](@dealer int) as
begin
if @Dealer=1
begin
select * from ProvisioningError where EventType='ERROR' and isactive=1 and cate ry=2
and
id in (select p.id from ProvisioningError p,CustomerOrderLine c ,CustomerOrder co where
c.CustomerOrderId =p.orderid and c.CustomerOrderId=co.Id and c.Status=16)
order by OrderId asc, createdOn desc
end
else
begin
select * from ProvisioningError where EventType='ERROR' and isactive=1 and cate ry=2
and
id in (select p.id from ProvisioningError p,CustomerOrderLine c ,CustomerOrder co where
c.CustomerOrderId =p.orderid and c.CustomerOrderId=co.Id and co.DealerId=@Dealer and
c.Status=16)
order by OrderId asc, createdOn desc
end
end
CREATE procedure [dbo].[GetPendingProvErrors](@Dealer int) as
if @Dealer=1
begin
select * from ProvisioningError where EventType='ERROR' and Isactive=1 and cate ry=1
and
id in (select p.id from ProvisioningError p,CustomerOrderLine c,CustomerOrder co where
c.CustomerOrderId =p.orderid and c.CustomerOrderId=co.id and c.Status=12)
order by id asc, createdOn desc
end
else
begin

```

```

select * from ProvisioningError where EventType='ERROR' and Isactive=1 and cate ry=1
and
id in (select p.id from ProvisioningError p, CustomerOrderLine c, CustomerOrder co where
c.CustomerOrderId =p.orderid and c.CustomerOrderId=co.id and co.DealerId=@Dealer and
c.Status=12)
order by id asc, createdOn desc
end
CREATE PROCEDURE [dbo].[GetPerSetsForGroup]

    @GroupId int
AS
BEGIN
--SET @GroupId = 1 ;
WITH n(Pageld, Pagename, [level], parentId, PageCode, PageUrl, concatenador) AS
(
    SELECT Pageld, Pagename, [level], parentId, PageCode, PageUrl,
    ('+CONVERT(VARCHAR (MAX), Pageld)+' - '+CONVERT(VARCHAR (MAX), [level])+')' as
concatenador
    FROM tblpages
    WHERE --PageUrl Like '%'+@PageUrl+'%'
    parentId =0 and isMenuItem=1
    --
    UNION ALL
    SELECT m.Pageld, m.Pagename, m.level, m.parentId, m.PageCode, m.PageUrl,
    n.concatenador+' * ('+CONVERT (VARCHAR (MAX), case when ISNULL(m.parentId, 0)
= 0 then 0 else m.Pageld END)+' - '+CONVERT(VARCHAR (MAX), m.level)+')' as
concatenador
    FROM tblpages as m, n
    WHERE n.Pageld = m.parentId and m.isMenuItem=1
)
--SELECT distinct * FROM n ORDER BY concatenador asc
select distinct MapId,
    t1.Pageld,
    t1.PageName,
    [level], case when parentId is null then 0 else parentId end ParentId,
    t1. concatenador,
    t1.PageCode,
    case when t1.PageUrl is null then 'No-Url' else PageUrl end PageUrl,
    case when [view] =1 then 'true' else 'false' end [view],
    case when [Add] =1 then 'true' else 'false' end [Add],
    case when [Edit] =1 then 'true' else 'false' end [Edit],
    case when [Delete] =1 then 'true' else 'false' end [Delete]
from n t1 Left outer Join
tblgrpMapping t2 on t1.pageld = t2.Pageld
and t2.GroupId=@GroupId
and t2.[view]=1
-----By Syed to ingnor null mapid--20141127-->>
where mapid is not null
-----By Syed to ingnor null mapid--20141127--<<
ORDER BY pageid asc
END
create procedure [dbo].[GetProductbyId](@pid int, @userid int, @stype int) as
SELECT    dbo.Product.ProductId as Pid, dbo.Product.ProductName,
dbo.Product.ProductGroupCate ry, dbo.Product.ProductGroupId, dbo.productSeller.*
FROM      dbo.Product LEFT OUTER JOIN

```

```

        dbo.productSeller ON dbo.Product.ProductId = dbo.productSeller.ProductId
        where Product.ProductId=@pid and sellerid=@userid
and productSeller.prid=@stype
CREATE procedure [dbo].[GetProductbySellerId] ( @productid int) as
begin
select s.*
,ProductName
,ProductGroupCate ry
,ProductGroupId
,DealerId
from product p ,productseller s where p.productid=s.productid and s.productid=@productid
end
CREATE procedure [dbo].[GetProductSellerbyProductId] ( @productid int) as
begin
select p.ProductId,p.ProductName,p.ProductGroupId, s.* from product p ,productSeller s
where p.productid=s.productid and s.productid=@productid
end
create procedure [dbo].[GetProductSellerbySellerId] ( @sellerid int) as
begin
select p.ProductId,p.ProductName,p.ProductGroupId, s.* from product p ,productSeller s
where p.productid=s.productid and s.sellerid=@sellerid
end
create procedure [dbo].[GetQueueListByAllUser]
AS
begin
SELECT q.Id, q.Name, COUNT(v.Total) AS Total,
[dbo].[fn_GetQ_Overdue](q.Id) AS OverDue,
[dbo].[fn_GetQ_OutStanding](q.Id) AS OutStanding,
[dbo].[fn_GetQ_Unassigned](q.Id) AS Unassigned
FROM dbo.v_QueueByUser AS v RIGHT OUTER JOIN
dbo.ActionQueue AS q ON v.Queueid = q.Id
GROUP BY q.Id, q.Name order by q.Name
end
CREATE procedure [dbo].[GetQueueListByUser] (@userid int)
AS
begin
SELECT q.Id, q.Name, COUNT(v.Total) AS Total,
[dbo].[fn_GetQ_OverduebyUser](q.Id,@userid) AS OverDue,
[dbo].[fn_GetQ_OutStandingbyUser](q.Id,@userid) AS OutStanding,
[dbo].[fn_GetQ_UnassignedbyUser](q.Id,@userid) AS Unassigned
FROM dbo.v_QueueByUser AS v RIGHT OUTER JOIN
dbo.ActionQueue AS q ON v.Queueid = q.Id
GROUP BY q.Id, q.Name order by q.Name
end
CREATE procedure [dbo].[GetSearchCustomerOrders](@otype int,@urole int,@dealerid
int,@userid int,@SrchStrng varchar(300)) as
if @urole=1
--select * from dbo.CustomerOrder where IsDraft=@otype and OrderReferenceString like
'%'+@SrchStrng+'%'
select co.* from customerorder co,customerorderline col,dbo.CustomerOrderLineDet cold
where
co.id=col.customerorderid
and col.id=cold.custordlineid
and (cold.orderlinecli like '%'+@SrchStrng+'%' or co.OrderReferenceString like
'%'+@SrchStrng+'%') and co.IsDraft=@otype

```



```

else if @urole=2 or @urole=3
--select * from dbo.CustomerOrder where DealerId=@DealerId and IsDraft=@otype and
OrderReferenceString like '%'+@SrchStrng+'%'
select co.* from customerorder co,customerorderline col,dbo.CustomerOrderLineDet cold
where
co.id=col.customerorderid
and col.id=cold.custordlineid
and (cold.orderlinecli like '%'+@SrchStrng+'%' or co.OrderReferenceString like
'%'+@SrchStrng+'%') and co.IsDraft=@otype and co.DealerId=@DealerId
else
--select * from dbo.CustomerOrder where EnteredById=@userid and IsDraft=@otype and
OrderReferenceString like '%'+@SrchStrng+'%'
select co.* from customerorder co,customerorderline col,dbo.CustomerOrderLineDet cold
where
co.id=col.customerorderid
and col.id=cold.custordlineid
and (cold.orderlinecli like '%'+@SrchStrng+'%' or co.OrderReferenceString like
'%'+@SrchStrng+'%') and co.IsDraft=@otype and co.EnteredById=@userid
CREATE procedure [dbo].[GetSearchFinalCustomerOrders]( @urole int=0,@dealerid
int=0,@userid int=0,@dealer int=0,@SrchStrng varchar(300)) as
begin
declare @filter varchar(500)
set @filter=' '
DECLARE @script VARCHAR(MAX);
declare @OpenQ varchar(100);
declare @EndQ varchar(5);
truncate table GetSearchFinalCustomerOrdersResult
set @OpenQ='insert into GetSearchFinalCustomerOrdersResult select * from ( '
set @EndQ=' ) t'
set @script='select distinct o.*,c.IsBusiness,c.BrandId,onhold=(case o.OnHoldStartDate
when null then 0 else case o.onholdenddate when null then 1 else 2 end end )
,status=l.Status,StatusName=e.Name from CustomerOrder o,Customer
c,CustomerOrderLine l,dbo.CustomerOrderLineStatusEnum e,dbo.CustomerOrderLineDet
cold where o.Id=l.CustomerOrderId and o.CustomerId=c.Id and l.Status=e.id and
o.IsDraft=0 and l.id=cold.custordlineid and (cold.orderlinecli like "%'+ @SrchStrng +'%" or
o.OrderReferenceString like "%'+ @SrchStrng +'%" ) '
if @urole=1
SET @script =@script + @filter
else if @urole=2 or @urole=3
SET @script = @script+' and c.DealerId='+ convert(varchar(10),@dealerid) + @filter
else
SET @script =@script+' and EnteredById='+ convert(varchar(10),@userid) + @filter
SET @script =@OpenQ + @script + @EndQ
--select @script Q
EXEC (@script);
select * from GetSearchFinalCustomerOrdersResult
end
CREATE procedure [dbo].[GetSearchFinalProducts]( @SrchStrng varchar(300)) as
select p.[ProductId],p.ProductName,p.ProductGroupCate ry,p.ProductGroupId,ps.SellerId
,Seller=tu.lastname,
pc.[PcatDesc],pg.[pgroupname],ps.price,ps.CurrencyId,CDescription=c.shortcode,
sellerimage=( select top 1 Images from productSellerImages where ProductId=p.ProductId
and SellerId=ps.SellerId),ImageURL=( select top 1 ImageURL from
productSellerImages where ProductId=p.ProductId and SellerId=ps.SellerId)
from product p ,productcat pc,productgroup pg,tbluser tu,productSeller ps,Currency c

```

```

where p.[ProductGroupCategory]=pc.[PCatId] and pg.[pgid]=p.[ProductGroupId]
and tu.userid=ps.SellerId and p.ProductId=ps.ProductId and c.cid=ps.CurrencyId
and( (ProductName like '%'+@SrchStrng+'%' ) or (pc.PcatDesc like '%'+@SrchStrng+'%' )
or (pg.pgroupname like '%'+@SrchStrng+'%' )
or (ps.KeyWords like '%'+@SrchStrng+'%' ) or (ps.SellerDescription like
'%'+@SrchStrng+'%' ) )
create procedure [dbo].[GetSearchFinalUsers](@SrchStrng varchar(300)) as
select distinct * from tbluser
where
(username like '%'+@SrchStrng+'%' or firstname like '%'+@SrchStrng+'%' or lastname like
'%'+@SrchStrng+'%' )
CREATE procedure [dbo].[GetSelectedProductsbyId] (@id int) as
begin
declare @packid int
select @packid=CallPackageId from CustomerOrderLineDet where CustOrdLineId=@id
SELECT p.id, p.name, p.ProductGroupId, p.ProductGroupCategory,
p.Type,p.SetupPrice,Price= p.Year1MonthlyPrice, Freq=0, Period='',
p.TechnologyType,p.PackageType, SelectedValue=isnull((select SelectedValue from
CustOrdLineSelected where p.id=CPId and CustOrdLineId=@id ),0)
FROM dbo.callproduct AS p INNER JOIN
dbo.PackagedProductDet AS d ON p.Id = d.CPId
where d.PPid=@packid
end
CREATE procedure [dbo].[GetSpecialReqById](@custordlineid int) as
SELECT cp.Name
,cls.CPId
,SelectedValue ,cp.PackageType
FROM CustOrdLineSelected cls,CallProduct cp
where cls.CPId=cp.Id and cp.PackageType<>2 and cp.PackageType<>1
and CustOrdLineId=@custordlineid
create proc [dbo].[gettest]
@xml xml
as
begin
declare @XmlDocpointer int
Exec sp_xml_preparedocument @XmlDocpointer output, @xml
Create Table #TmpParamvalues (ParamId int,ParamValue varchar(50))
insert into #TmpParamvalues
select [PrimaryId],[SelctedValue] from OPENXML(@XmlDocpointer,'/root/Data',2)
with
(
PrimaryId int,
SelctedValue varchar(max)
)
end
CREATE procedure [dbo].[GetUnBilledByStatus] (@Status int) as
begin
/*select distinct CustomerId=c.Id, CustomerRef=
c.ExternalReferenceNumber,CustomerOrderId=co.Id,
isnull(col.IsCustomerAdded,0)IsCustomerAdded,isnull(col.IsBillingAPISuccess,0)IsBillingAPIS
uccess,
custcode=c.ExternalReferenceNumber
from CustomerOrderLine col,CustomerOrder co,Customer c
where col.CustomerOrderId=co.Id and

```

```

ISNULL(IsBillingAPISucess,0)=0 and
co.customerid=c.id and
col.Status=@Status
*/
select distinct dbo.GetCustIDby_CustOrderID(CustomerOrderId) as CustomerId,
dbo.GetCust_code(CustomerOrderId) as CustomerRef,CustomerOrderId,

isnull(IsCustomerAdded,0)IsCustomerAdded,isnull(IsBillingAPISucess,0)IsBillingAPISucess,
dbo.GetCust_code(CustomerOrderId) as custcode from CustomerOrderLine where
ISNULL(IsBillingAPISucess,0)=0 and Status=@Status and isnull(iscustomeradded,0)=0
order by custcode
end
-- =====
-- Author:          <Author,,Name>
-- Create date: <Create Date,,>
-- Description:<Description,,>
-- =====
CREATE PROCEDURE [dbo].[GetUserLandingPage]

@UserName Varchar(200)
AS
BEGIN
WITH n(Pageld, Pagename, [level], parentId,PageCode,PageUrl,concatenador) AS
(
SELECT Pageld, Pagename,[level],parentId,PageCode,PageUrl,
'+CONVERT(VARCHAR (MAX), Pageld)+' - '+CONVERT(VARCHAR (MAX), [level])+')' as
concatenador
FROM tblpages
WHERE --PageUrl Like '%'+@PageUrl+'%'
parentId =0
--
UNION ALL
SELECT m.Pageld, m.Pagename,m.level,m.parentId,m.PageCode,m.PageUrl,
n.concatenador+' * ('+CONVERT (VARCHAR (MAX), case when ISNULL(m.parentId, 0)
= 0 then 0 else m.Pageld END)+' - '+CONVERT(VARCHAR (MAX), m.level)+')' as
concatenador
FROM tblpages as m, n
WHERE n.Pageld = m.parentId and m.isMenuItem =1
)
--SELECT distinct * FROM n ORDER BY concatenador asc
select top 1 MapId,
t1.Pageld,
t1.PageName,
[level],case when parentId is null then 0 else parentId end ParentId,
t1. concatenador,
t1.PageCode,
case when t1.PageUrl is null then 'No-Url' else PageUrl end PageUrl,
case when [view] =1 then 'true' else 'false' end [view],
case when [Add] =1 then 'true' else 'false' end [Add],
case when [Edit] =1 then 'true' else 'false' end [Edit],
case when [Delete] =1 then 'true' else 'false' end [Delete]
from n t1, tblgrpMapping t2,tblUser t3 where t1.pageld = t2.Pageld
and t3.GroupId =t2.groupid and t3.UserName= @UserName and t2.[view] = 1 and
t1.pageurl<>'No-Url' ORDER BY concatenador asc
END

```

```

CREATE PROCEDURE [dbo].[GetUserList](@username varchar(50))
AS
BEGIN
select * from tbluser
END
Create proc [dbo].[GetVolMatrixFieldValidation](@ActionType int,@dealer int,@groupName
varchar(250),@startDate datetime,@endDate datetime)
as
begin
declare @Result varchar(250)="
declare @CountgroupName int=0
declare @CountdateRange int=0
--@ActionType(1)=Add new Record, @ActionType(0)=Edit Existing Record
if(@ActionType=1)
begin
select @CountgroupName=count(volumematrixid) from
dbo.cmnVolumeMatrix where dealerid=@dealer
and groupname=@groupName
if(@CountgroupName>0) set @Result='Group name:'+ @GroupName+'
already exist.'
select @CountdateRange=count(volumematrixid) from dbo.cmnVolumeMatrix
where dealerid=@dealer
and
--(dateValidto >= @startDate
--or
--dateValidto >= @endDate)
(@startDate between datevalidfrom and datevalidto or @endDate between
datevalidfrom and datevalidto)
if(@CountdateRange>0)set @Result= @Result + ' Date Range overlapping
with existing matrix.'
end
else
begin
select @CountdateRange=count(volumematrixid) from dbo.cmnVolumeMatrix
where dealerid=@dealer
and groupname<>@groupName
and
--(dateValidto >= @startDate
--or
--dateValidto >= @endDate)
(@startDate between datevalidfrom and datevalidto or @endDate between
datevalidfrom and datevalidto)
if(@CountdateRange>0)set @Result= @Result + ' Invalid date range or
overlapping with existing matrix.'
end
select @Result as Result
end
CREATE procedure [dbo].[GetVolumBonusMatrixCommProducts](@dealerid int,
@GroupName varchar(250))
as
begin
(select bg.basegroupid bbasegroupid,vm.*
from dbo.cmnBaseGroups bg left join dbo.cmnVolumeMatrix vm
on bg.product=vm.product
where

```

```

vm.dealerid=@dealerid
and vm.groupname=@GroupName
and bg.isactive=1)
union
(select basegroupid bbasegroupid,null VolumeMatrixID,null GroupName,Product,null
DealerID,null RangeBegin,null Range1Amount,
 null Range1Value,null Range2Amount,null Range2Value,null Range3Amount,null
Range3Value,null DateValidFrom,
 null DateValidTo,null IsActive,null basegroupid
from dbo.cmnBaseGroups
where basegroupid not in (select basegroupid from cmnVolumeMatrix where
dealerid=@dealerid and groupname=@GroupName)
and isactive=1)
end
CREATE procedure [dbo].[ImmediateSalesSum] as
declare @td datetime
declare @yd datetime
select @td= GETDATE()
select @yd=dateadd(d,-1,GETDATE())
begin
with n1(TdNos,Ydnos,DealerId) as
(
select COUNT(*) as TdNos, 0 as Ydnos,DealerId from customerorder where
orderdate=@td and IsDraft=0 group by DealerId
union all
select 0 as TdNos, COUNT(*) as Ydnos,DealerId from customerorder where
orderdate=@yd and IsDraft=0 group by DealerId)
select TdNos=SUM(TdNos),Ydnos=SUM(Ydnos), d.CompanyName from n1,Dealer d
where n1.DealerId=d.DealerId group by d.CompanyName
end
CREATE PROCEDURE [dbo].[LLUProvisionData] (@custordid int)
AS
select c.Title, c.FirstName, c.LastName, c.ContactPostcode,
(c.ContactAddressLine1 + c.ContactAddressLine2 + c.ContactAddressLine5) as Locality,
co.Id as OrderId, 'NCD_SUB_0003' AS productCode from Customer as c ,
CustomerOrder co ,CUSTOMERORDERIINE col
where C.Id = CO.CustomerId and COL.CustomerOrderId = CO.Id and co.IsDraft=0
and CO.Id=@custordid
CREATE PROCEDURE [dbo].[ManagePermissions]
@GroupId int,@type int
AS
BEGIN
--SET @GroupId = 1 ;
if(@type=1)
begin
WITH n(Pageld, Pagename, [level],
parentId,PageCode,PageUrl,concatenador) AS
(
SELECT Pageld, Pagename,[level],parentId,PageCode,PageUrl,
'+CONVERT(VARCHAR (MAX), Pageld)+' - '+CONVERT(VARCHAR (MAX), [level])+' as
concatenador
FROM tblpages
WHERE --PageUrl Like '%'+@PageUrl+'%'
parentId =0 and isMenuItem <= 1
--

```

```

        UNION ALL
        SELECT m.Pageld,
m.Pagename,m.level,m.parentId,m.PageCode,m.PageUrl,
        n.concatenador+' * ('+CONVERT (VARCHAR (MAX), case when
ISNULL(m.parentId, 0) = 0 then 0 else m.Pageld END)+' - '+CONVERT(VARCHAR (MAX),
m.level)+')' as concatenador
        FROM tblpages as m, n
        WHERE n.Pageld = m.parentId and m.isMenuItem <= 1
    )
--SELECT distinct * FROM n ORDER BY concatenador asc
select distinct MapId,
        t1.Pageld,
        t1.PageName,
        [level],case when parentId is null then 0 else parentId end ParentId,
        t1. concatenador,
        t1.PageCode,
        case when t1.PageUrl is null then 'No-Url' else PageUrl end PageUrl,
        case when [view] =1 then 'true' else 'false' end [view],
        case when [Add] =1 then 'true' else 'false' end [Add],
        case when [Edit] =1 then 'true' else 'false' end [Edit],
        case when [Delete] =1 then 'true' else 'false' end [Delete]
from n t1 Left outer Join

        tblgrpMapping t2 on t1.pageld = t2.Pageld
        and t2.GroupId=@GroupId
        and t2.[view]=1
-----By Syed to ignor null mapid--20141127-->>
        where mapid is not null
-----By Syed to ignor null mapid--20141127--<<
        ORDER BY pageid asc
end
Else
begin
WITH n(Pageld, Pagename, [level], parentId,PageCode,PageUrl,concatenador) AS
(
        SELECT Pageld, Pagename,[level],parentId,PageCode,PageUrl,
(''+CONVERT(VARCHAR (MAX), Pageld)+' - ''+CONVERT(VARCHAR (MAX), [level]))' as
concatenador
        FROM tblpages
        WHERE --PageUrl Like '%'+@PageUrl+'%'
        parentId =0 and isMenuItem <= 1
        --
        UNION ALL
        SELECT m.Pageld,
m.Pagename,m.level,m.parentId,m.PageCode,m.PageUrl,
        n.concatenador+' * ('+CONVERT (VARCHAR (MAX), case when
ISNULL(m.parentId, 0) = 0 then 0 else m.Pageld END)+' - '+CONVERT(VARCHAR (MAX),
m.level)+')' as concatenador
        FROM tblpages as m, n
        WHERE n.Pageld = m.parentId and m.isMenuItem <= 1
    )
--SELECT distinct * FROM n ORDER BY concatenador asc
select distinct MapId,
        t1.Pageld,
        t1.PageName,

```

```

        [level],case when parentId is null then 0 else parentId end ParentId,
        t1.concatenador,
        t1.PageCode,
        case when t1.PageUrl is null then 'No-Url' else PageUrl end PageUrl,
        case when [view] =1 then 'true' else 'false' end [view],
        case when [Add] =1 then 'true' else 'false' end [Add],
        case when [Edit] =1 then 'true' else 'false' end [Edit],
        case when [Delete] =1 then 'true' else 'false' end [Delete]
    from n t1 Left outer Join

        tblgrpMapping t2 on t1.pageld = t2.Pageld
        and t2.GroupId=@GroupId
        --and t2.[view]=1
        -----By Syed to ingnor null mapid--20141127-->>
        where mapid is not null
        -----By Syed to ingnor null mapid--20141127--<<
        ORDER BY pageid asc
    end
END
CREATE PROCEDURE [dbo].[ManageProducts]
    @GroupId int,@type int
AS
BEGIN
        --SELECT PageId=PCatId,
        Pagename=convert(varchar(250),PcatDesc),level=0,parentId=0,
        -->('+CONVERT(VARCHAR (MAX), [PCatId])+ ' -
        '+CONVERT(VARCHAR (MAX), 1)+')' as concatenador
        --FROM [dbo].[Productcat] where pcatid= @GroupId
        --union all
        SELECT PageId=[pgid],
        Pagename=convert(varchar(250),[pgroupname]),level=1,parentId=convert(int,[PCatId]),
        ('+CONVERT(VARCHAR (MAX), [PCatId])+ ' -
        '+CONVERT(VARCHAR (MAX), 1)+')' as concatenador

        FROM [dbo].[ProductGroup] where @GroupId=pcatid
END
CREATE procedure [dbo].[MonthlySalesSum] as
begin
with n1(SalesNos,CancelNos,DealerId,Mth)as
(select
SalesNos=COUNT(*),CancelNos=0,DealerId,Mth=CONVERT(varchar(3),OrderDate)+'-
'+CONVERT(varchar(4),year(OrderDate))
from CustomerOrder where IsDraft=0 group by
DealerId,CONVERT(varchar(3),OrderDate)+'-'+CONVERT(varchar(4),year(OrderDate))
union all
select
SalesNos=0,CancelNos=COUNT(*),DealerId,Mth=CONVERT(varchar(3),OrderDate)+'-
'+CONVERT(varchar(4),year(OrderDate))
from CustomerOrder where IsDraft=0 and [CancelledByUserId] is not null group by
DealerId,CONVERT(varchar(3),OrderDate)+'-'+CONVERT(varchar(4),year(OrderDate))
)
select SalesNos=SUM(SalesNos),SalesPDay=round(SUM(SalesNos)/30,2),
CancelNos=SUM(CancelNos),CancelRatio=round(SUM(CancelNos)/SUM(SalesNos)*100,2)
,NetSales=SUM(SalesNos)-SUM(CancelNos), d.CompanyName,Mth from n1,Dealer d
where n1.DealerId=d.DealerId group by d.CompanyName,Mth

```

```

end
CREATE procedure [dbo].[OpalCredentialsForBrand](@brandid int)
as
SELECT [AgentId]
      ,[UserName]
      ,[Password]
      ,[DomainId]
      ,[ResellerReference]
FROM [dbo].[OpalCredential] O,dbo.Brand B
where O.Id=b.OpalCredentialId and b.Id =@brandid
CREATE PROCEDURE [dbo].[OrderProcessingRunner_ProcessHeldOrders]
(
  @OrderId   VARCHAR(100)
)
AS
BEGIN
  UPDATE      CustomerOrder
  SET         OnHoldEndDate = NULL,
             OnHoldReasonDescription = NULL,
             OnHoldReasonId = NULL,
             OnHoldStartDate = NULL
  WHERE      CustomerOrder.Id = @OrderId
END
create procedure [dbo].[PackageDetail] (@ppid int) as
  select ppd.*,pd.* from
  dbo.PackagedProductDet ppd,
  dbo.product pd
  where pd.productid=ppd.cpid and ppd.ppid=@ppid
create procedure [dbo].[PackagesList] as
SELECT  Packageld
      ,PackageName
      ,p.isActive
      ,p.techid,TechTypeName
      ,p.Channleid,ChannelName
FROM package p,channels c,dbo.TechTypes t
  where p.Channleid=c.ChannelId and p.techid=t.TechTypeId
CREATE PROCEDURE [dbo].[ProvisionEligibleOrderIds]
AS
Begin
  select distinct co.Id as orderId from CustomerOrder co ,CUSTOMERORDERIINE col
  where COL.CustomerOrderId = CO.Id and co.IsDraft=0 and (col.Status = 3 or col.Status =
  4)
  and (co.onholdstartdate is null or onholdenddate < getdate()) and col.isllu=1
  --and Summary like 'MIGRATE%'
End
CREATE PROCEDURE [dbo].[ProvisioningSingleOrderData] @orderId int
AS
  SELECT  c.Title, c.FirstName, c.LastName, isnull(c.ContactPostcode,") ContactPostcode,
  isnull(c.MobileTelephone,") MobileTelephone, c.Id AS customerId, c.ContactAddressLine1 ,
  isnull(c.ContactAddressLine2,") ContactAddressLine2 ,
  c.ContactAddressLine5,c.ContactAddressLine1 + isnull(c.ContactAddressLine2,") +
  c.ContactAddressLine5 as Locality, co.Id AS OrderId,
  isnull(PRD.ThirdPartyProvisioningCode,") AS productCode, COL.Status,
  isnull(COLD.ConnectionType,") ConnectionType, isnull(COLD.BroadBandPackageld,")

```



```

BroadBandPackageld, isnull(COLD.OrderLineCli,") OrderLineCli, isnull(PRD.Id,") id,
COL.OrderType,
    isnull( COL.Planned LiveDate,dateadd(d,15,co.orderdate)) AS Planned
LiveDate,IsLLU=isnull(IsLLU,0),OrderLineId=COL.Id,
    InstallationPostCode=isnull(cold.OLPostCode,c.ContactPostcode)
,OrderProcessType=(case substring(col.summary,1,3)when 'MIG' then 1 else 2 end),

CustomerCode=c.ExternalReferenceNumber,co.OrderReferenceString,Summary,c.BrandId,
c.IsBusiness
FROM CallProduct AS PRD INNER JOIN
    CustomerOrderLineDet AS COLD ON PRD.Id = COLD.BroadBandPackageld RIGHT
OUTER JOIN
    Customer AS c INNER JOIN
    CustomerOrder AS co ON c.Id = co.CustomerId INNER JOIN
    CustomerOrderLine AS COL ON co.Id = COL.CustomerOrderId ON
COLD.CustOrdLineId = COL.Id
    AND COLD.BroadBandPackageld=PRD.ID
    where co.Id = @orderId
/***** Script for SelectTopNRows command from SSMS *****/
--Alter view vproduct as
CREATE procedure [dbo].[pvproduct]( @id int) as
if @id>0
begin
SELECT [productid]=0,ProductGroupId=-1
    ,ProductName=[PcatDesc],lbtnimage,id=PCatId
FROM [MyLaadBazarDb].[dbo].[ProductCat] where pcatid=@id
union all
SELECT [pgid]
    ,[PCatId]=0
    ,[pgroupname],isnull(lbtnimage,"),id=pgid
FROM [MyLaadBazarDb].[dbo].[ProductGroup] where pcatid=@id
union all
SELECT [ProductId]+(select count(*) from [ProductGroup] where pcatid=@id)
    ,ProductGroupId
    ,[ProductName],lbtnimage,id=ProductId
FROM [MyLaadBazarDb].[dbo].[Product] where ProductGroupCategory=@id
end
else
begin
Select [productid]=0,ProductGroupId=-1,
    ProductName='MyLaadBazar',lbtnimage='images/lo small.png',id=0
union all
SELECT [productid]=pcatid,ProductGroupId=0
    ,ProductName=[PcatDesc],lbtnimage,id=pcatid
FROM [MyLaadBazarDb].[dbo].[ProductCat]
union all
SELECT [pgid]+20
    ,[PCatId]
    ,[pgroupname],isnull(lbtnimage,"),id=pgid
FROM [MyLaadBazarDb].[dbo].[ProductGroup]
union all
SELECT [ProductId]+(select count(*)+20 from [ProductGroup] )
    ,ProductGroupId+20
    ,[ProductName],lbtnimage,id=ProductId
FROM [MyLaadBazarDb].[dbo].[Product]

```

```

end
CREATE PROCEDURE [dbo].[REPORT_GetActionTasks]
  @StartDate          DateTime,
  @EndDate            DateTime
AS
BEGIN
  SELECT              COUNT(*),
                    ActionQueue.Name
  FROM                ActionTask
  INNER JOIN          ActionQueue
  ON                  ActionTask.QueueId = ActionQueue.Id
  WHERE               ActionTask.RaisedDate BETWEEN @StartDate AND
  @EndDate
  GROUP BY           ActionQueue.Name
END
CREATE procedure [dbo].[SearchCustomer] ( @sstring varchar(100), @stype int)as
begin
set @sstring='%'+@sstring+'%'
if @stype=0
select * from Customer where (FirstName like @sstring or MiddleNames like @sstring or
LastName like @sstring )
if @stype=1
select * from Customer where (externalreferencenumber like @sstring or CompanyRegNo
like @sstring)
if @stype=2
select * from Customer where (ContactAddressLine1 like @sstring or ContactAddressLine2
like @sstring or ContactAddressLine3 like @sstring or ContactAddressLine4 like @sstring or
ContactAddressLine5 like @sstring or BillingAddressLine1 like @sstring or
BillingAddressLine2 like @sstring or BillingAddressLine3 like @sstring or
BillingAddressLine4 like @sstring or BillingAddressLine5 like @sstring )
if @stype=3
select * from Customer where (MainTelephone like @sstring or OtherTelephone1 like
@sstring or MobileTelephone like @sstring or FaxNumber like @sstring or EmailAddress
like @sstring)
end
create procedure [dbo].[SetOrderToCancel](@CUSTID int)
as
begin
update col set col.status=11 from customerorder co, customerorderline col
where
co.id = col.customerorderid
and co.customerid=@CUSTID
end
CREATE procedure [dbo].[ShortTermSalesSummary] as
begin
with n1(SalesNos28,CanNos28,SalesNos7,CanNos7,DealerId)as
( select SalesNos28=count(*),CanNos28=0,SalesNos7=0,CanNos7=0, DealerId from
dbo.CustomerOrder
where datediff(d,orderdate,getdate())<29 group by DealerId
union all
select SalesNos28=0,CanNos28=0,SalesNos7=count(*),CanNos7=0, DealerId from
dbo.CustomerOrder
where datediff(d,orderdate,getdate())<8 group by DealerId
union all

```

```

select SalesNos28=0,CanNos28=count(*),SalesNos7=0,CanNos7=0, DealerId from
dbo.CustomerOrder
where datediff(d,CancelledDate,getdate())<29 and CancelledDate is not null
group by DealerId
union all
select SalesNos28=0,CanNos28=0,SalesNos7=0,CanNos7=count(*), DealerId from
dbo.CustomerOrder
where datediff(d,CancelledDate,getdate())<8 and CancelledDate is not null
group by DealerId
)
select SalesNos28=SUM(SalesNos28),SalesPd28=SUM(SalesNos28)/28,
CanNos28=SUM(CanNos28),
SalesNos7=SUM(SalesNos7),SalesPD7=SUM(SalesNos7)/7,
CanNos7=SUM(CanNos7),
canratio28=case sum(SalesNos28) when 0 then 0 else
SUM(CanNos28)/SUM(SalesNos28)*100 end,
canratio7=case sum(SalesNos7) when 0 then 0 else
SUM(CanNos7)/SUM(SalesNos7)*100 end,
netsale28=SUM(SalesNos28)-SUM(CanNos28),netsale7=SUM(SalesNos7)-
SUM(CanNos7), n1.DealerId,CompanyName from n1,Dealer d
where n1.DealerId=d.DealerId
group by n1.DealerId,CompanyName
end
create procedure [dbo].[sp_EAPI_CheckFullPayment]
as
begin
select erd.customerid,erd.custordid,erd.customercode,erd.invoice_number
from eapi_request_details erd left join customerorderline col on
erd.custordid=col.customerorderid
where erd.requesttype=3 and erd.eapi_req_status=1 and erd.invoice_number is not null
and erd.customercode is not null
and col.status=6
end
CREATE procedure [dbo].[sp_GetCustCLRtoSendEbillz]
as
begin
select dbo.GetCust_MainTelephone(customerorderid)as CustomerIdentifier,
dbo.GetCust_Code(customerorderid) as CustomerCode,
dbo.GetCust_orderlinecli(id) as CLI,
dbo.Get_TariffByPpId(ppid) as Tariff,
dbo.Get_year2monthlypriceByPpId(PpId) year2monthlyprice,
CustomerOrderId,Id,Summary,OrderDate,planned livedate
from [customerorderline] where status=6 and iscustomeradded=1 and
isnull(IsLCRAdded,0)=0
and dbo.GetCust_orderlinecli(id) is not null
end
create procedure [dbo].[sp_GetCustomerCodeByOderID](@OrderId int)
As
begin
select dbo.GetCust_code(@OrderId) as CustomerCode
end
create procedure [dbo].[sp_GetCustOrderDetailsForInvoice](@CustOrdLineId int)
as
begin
select dbo.GetCust_MainTelephone(customerorderid)as CustomerIdentifier,

```

```

    dbo.GetCust_Code(customerorderid) as CustomerCode,
    dbo.Fun_CustFullName(customerorderid) as
    CustFullName,dbo.Fun_CustFullAddress(customerorderid) as CustFullAddress,
    dbo.GetCust_orderlinecli(id) as CLI,
    * from [customerorderline] where id=@CustOrdLineId
end
CREATE procedure [dbo].[sp_GetCustOrdQAClk](@qtype int)
as
begin
    if (@qtype=1) --With Recordings
        begin
            /*where c.QualityAssuranceByld ==null
            && c.IsDraft == false
            && c.WelcomeCallUserId != null
            && c.CancelledByUserId != null
            && c.RecordingsAvailableForQA == true
            */
            select c.Id, c.CustomerId,
            c.OrderDate,DATEDIFF(day,c.OrderDate,getdate())age,
            c.OrderReferenceString,dbo.Fun_GetUserName(c.EnteredById)EnteredBy,dbo.Fun_CustFull
            Name(c.id)CustName,[dbo].[GetRecordingCountbyOrdID](c.Id) Rec,case
            isnull(QualityAssuranceData,'0') when '0' then 'ADD' else 'EDIT' end PageType
            from CustomerOrder c
            where
            c.IsDraft = 0 and IsQACompleted=0 and c.CancelledByUserId is null and
            [dbo].[GetRecordingCountbyOrdID](c.Id) > 0
            end
        else if (@qtype = 2) --Without Recordings
            begin
                /*where c.QualityAssuranceByld == null
                && c.IsDraft == false
                && c.WelcomeCallUserId == null
                && c.CancelledByUserId == null
                && (c.RecordingsAvailableForQA == false || c.RecordingsAvailableForQA
                ==null)
                */
                select c.Id, c.CustomerId,
                c.OrderDate,DATEDIFF(day,c.OrderDate,getdate())age,
                c.OrderReferenceString,dbo.Fun_GetUserName(c.EnteredById)EnteredBy,dbo.Fun_CustFull
                Name(c.id)CustName,[dbo].[GetRecordingCountbyOrdID](c.Id) Rec,case
                isnull(QualityAssuranceData,'0') when '0' then 'ADD' else 'EDIT' end PageType
                from CustomerOrder c
                where
                c.IsDraft = 0 and IsQACompleted=0 and c.CancelledByUserId is null and
                [dbo].[GetRecordingCountbyOrdID](c.Id) <= 0 and coid is null
                end
            else if (@qtype = 3)--With or Without Rec
                begin
                    /*where c.QualityAssuranceByld == null
                    && c.IsDraft == false
                    && c.CancelledByUserId == null
                    */
                    select c.Id, c.CustomerId,
                    c.OrderDate,DATEDIFF(day,c.OrderDate,getdate())age,
                    c.OrderReferenceString,dbo.Fun_GetUserName(c.EnteredById)EnteredBy,dbo.Fun_CustFull

```

```

IName(c.id)CustName,[dbo].[GetRecordingCountybyOrdID](c.Id) Rec,case
isnull(QualityAssuranceData,'0') when '0' then 'ADD' else 'EDIT' end PageType
        from CustomerOrder c
        where
        c.IsDraft = 0 and IsQACompleted=0 and c.CancelledByUserId is null and coid
is null

        end
    else if (@qtype = 4) --Cancelled Orders
    begin
        /*where c.QualityAssuranceById == null
        && c.IsDraft == false
        && c.CancelledByUserId != null
        */
        select c.Id, c.CustomerId,
c.OrderDate,DATEDIFF(day,c.OrderDate,getdate())age,
c.OrderReferenceString,dbo.Fun_GetUserName(c.EnteredById)EnteredBy,dbo.Fun_CustFul
IName(c.id)CustName,[dbo].[GetRecordingCountybyOrdID](c.Id) Rec,case
isnull(QualityAssuranceData,'0') when '0' then 'ADD' else 'EDIT' end PageType
        from CustomerOrder c
        where
        c.IsDraft = 0 and IsQACompleted=0 and c.CancelledByUserId is not null and
coid is null
        end
    else if (@qtype = 5) --Cancelled with Rec
    begin
        /*where c.QualityAssuranceById == null
        && c.IsDraft == false
        && c.WelcomeCallUserId != null
        && c.CancelledByUserId != null
        */
        select c.Id, c.CustomerId,
c.OrderDate,DATEDIFF(day,c.OrderDate,getdate())age,
c.OrderReferenceString,dbo.Fun_GetUserName(c.EnteredById)EnteredBy,dbo.Fun_CustFul
IName(c.id)CustName,[dbo].[GetRecordingCountybyOrdID](c.Id) Rec,case
isnull(QualityAssuranceData,'0') when '0' then 'ADD' else 'EDIT' end PageType
        from CustomerOrder c
        where
        c.IsDraft = 0 and IsQACompleted=0 and c.CancelledByUserId is not null and
[dbo].[GetRecordingCountybyOrdID](c.Id) > 0 and coid is null
        end
    else if (@qtype = 6) --Cancelled without Rec
    begin
        /*where c.QualityAssuranceById == null
        && c.IsDraft == false
        && c.WelcomeCallUserId == null
        && c.CancelledByUserId != null
        */
        select c.Id, c.CustomerId,
c.OrderDate,DATEDIFF(day,c.OrderDate,getdate())age,
c.OrderReferenceString,dbo.Fun_GetUserName(c.EnteredById)EnteredBy,dbo.Fun_CustFul
IName(c.id)CustName,[dbo].[GetRecordingCountybyOrdID](c.Id) Rec,case
isnull(QualityAssuranceData,'0') when '0' then 'ADD' else 'EDIT' end PageType
        from CustomerOrder c
        where

```

```

        c.IsDraft = 0 and IsQACompleted=0 and c.CancelledByUserId is not null and
[dbo].[GetRecordingCountbyOrdID](c.Id) <= 0 and coid is null
    end
    else
        begin
            /*where c.QualityAssuranceByld == null
            && c.IsDraft == false
            */
            select c.Id, c.CustomerId,
c.OrderDate,DATEDIFF(day,c.OrderDate,getdate())age,
c.OrderReferenceString,dbo.Fun_GetUserName(c.EnteredByld)EnteredBy,dbo.Fun_CustFul
IName(c.id)CustName,[dbo].[GetRecordingCountbyOrdID](c.Id) Rec,case
isnull(QualityAssuranceData,'0') when '0' then 'ADD' else 'EDIT' end PageType
            from CustomerOrder c
            where
            c.IsDraft = 0 and IsQACompleted=0 and coid is null
        end
    end
end
CREATE procedure
[dbo].[sp_GetEAPIInvoiceBBOOrderLinesByORDERID](@customerorderid int)
as
begin
    select col.Id,col.CustomerOrderId,
    isnull(Cpr.description,'Package') Description1, '(' +
replace(convert(varchar(12),isnull(Planned LiveDate,""),106),' ','-') + '-' +
replace(convert(varchar(12),DATEADD(MM, DATEDIFF(MM, -1, isnull(Planned LiveDate,""),
0) - 1,106),' ','-')+)' as DescriptionDet1,
    isnull(Cpr.description,'Package') Description2, '(' +
replace(convert(varchar(12),DATEADD(MONTH, DATEDIFF(MONTH, 0, isnull(Planned
LiveDate,"")+1, 0),106),' ','-') + '-' +replace(convert(varchar(12),DATEADD(MONTH,
DATEDIFF(MONTH, -1, isnull(Planned LiveDate,"")+1, -1),106),' ','-')+)' as DescriptionDet2,
    CONVERT(decimal(11,2),(Year1MonthlyPrice/datediff(day, GetDate(), dateadd(month, 1,
GetDate())))*(datediff(day, GetDate(), dateadd(month, 1, GetDate()))-
DATEPART(dd,isnull(Planned LiveDate,0)))) Amount1,
    Year1MonthlyPrice Amount2,
    dbo.Get_VatCodeByPPId(isnull(CallPackageld,3))as VatCode ,
    AnalysisCode=ISNULL( cpr.AnalysisCode,")
    from customerorderline col, CustomerOrderLineDet cold ,CallProduct Cpr
    where col.id=cold.CustOrdLineId and
    col.status=6 and col.iscustomeradded=1 and cpr.id= cold.BroadBandPackageld
    and customerorderid=@customerorderid
end
CREATE procedure [dbo].[sp_GetEAPIInvoiceOrder]
as
begin
    select co.id, dbo.GetCust_Code(co.id) as CustomerCode,
co.customerid,co.orderreferencestring ,CLI=OrderLineCli,
CustomerOrderID=col.Id ,col.OrderType
    from [customerorder] co , customerorderline col ,CustomerOrderLineDet cold
    where co.id = col.customerorderid and col.Id=cold.CustOrdLineId and
    col.status=6 and col.iscustomeradded=1 and IsLCRAAdded=1
    and isnull(col.IsInvoiceAdded,0)=0 and isnull(IsBillingAPISuccess,0)=0
end
CREATE procedure [dbo].[sp_GetEAPIInvoiceOrderLinesByORDERID](@customerorderid
int)

```

```

as
begin
declare @ordertype int
select @ordertype=ordertype from customerorderline where customerorderid=
@customerorderid
if @ordertype=1
begin
select col.Id,col.CustomerOrderId,
isnull(substring(Cpr.description,1,50),'Package') Description1, '(' +
replace(convert(varchar(12),isnull(Planned LiveDate,"),106),'','-') + '-' +
replace(convert(varchar(12),DATEADD(MM, DATEDIFF(MM, -1, isnull(Planned LiveDate,"),
0) - 1,106),'','-')+)' as DescriptionDet1,
isnull(substring(Cpr.description,1,50),'Package') Description2, '(' +
replace(convert(varchar(12),DATEADD(MONTH, DATEDIFF(MONTH, 0, isnull(Planned
LiveDate,"))+1, 0),106),'','-') + '-' +replace(convert(varchar(12),DATEADD(MONTH,
DATEDIFF(MONTH, -1, isnull(Planned LiveDate,"))+1, -1),106),'','-')+)' as DescriptionDet2,
CONVERT(decimal(11,2),(Year1MonthlyPrice/datediff(day, GetDate(), dateadd(month, 1,
GetDate()))*(datediff(day, GetDate(), dateadd(month, 1, GetDate()))-
DATEPART(dd,isnull(Planned LiveDate,0)))) Amount1,
Year1MonthlyPrice Amount2,
dbo.Get_VatCodeByPPIId(isnull(CallPackageld,3))as VatCode ,
AnalysisCode=ISNULL( cpr.AnalysisCode,")
from customerorderline col, CustomerOrderLineDet cold ,CallProduct Cpr
where col.id=cold.CustOrdLineId and
col.status=6 and col.iscustomeradded=1 and cpr.id= cold.CallPackageld
and customerorderid= @customerorderid
end
else
begin
select col.Id,col.CustomerOrderId,
isnull(col.Summary,'Package') Description1, '(' +
replace(convert(varchar(12),isnull(Planned LiveDate,"),106),'','-') + '-' +
replace(convert(varchar(12),DATEADD(MM, DATEDIFF(MM, -1, isnull(Planned LiveDate,"),
0) - 1,106),'','-')+)' as DescriptionDet1,
isnull(col.summary,'Package') Description2, '(' +
replace(convert(varchar(12),DATEADD(MONTH, DATEDIFF(MONTH, 0, isnull(Planned
LiveDate,"))+1, 0),106),'','-') + '-' +replace(convert(varchar(12),DATEADD(MONTH,
DATEDIFF(MONTH, -1, isnull(Planned LiveDate,"))+1, -1),106),'','-')+)' as DescriptionDet2,
CONVERT(decimal(11,2),(UnitMonthlyY1PriceExVat/datediff(day, GetDate(),
dateadd(month, 1, GetDate()))*(datediff(day, GetDate(), dateadd(month, 1, GetDate()))-
DATEPART(dd,isnull(Planned LiveDate,0)))) Amount1,
UnitMonthlyY1PriceExVat Amount2,
--dbo.Get_VatCodeByPPIId(isnull(CallPackageld,3))
'3' as VatCode ,
AnalysisCode=""
from customerorderline col
where col.status=6 and col.iscustomeradded=1 and customerorderid= @customerorderid
end
end
CREATE procedure [dbo].[sp_GetOutStandingPrintCount](@brandid int)
as
begin
declare @Welcome int=0,@Cancle int=0
/* Original Procedure
select @Welcome= count(co.id)

```

```

from
customer c, customerorder co
where
c.id=co.customerid and
co.isdraft=0 and
co.cancelleddate is null and
co.id not in (select distinct customerorderid from CustomerOrderWelcomePack) and
c.brandid=@brandid

select @Cancle=count(co.id)
from
customer c, customerorder co
where
c.id=co.customerid and
co.isdraft=0 and
co.cancelleddate is not null and
co.id not in (select distinct customerorderid from CustomerOrderWelcomePack) and
c.brandid=@brandid
*/
select @Welcome= isnull(sum(contentlength),0) from dbo.CorrespondenceBatch where
correspondenceclassid=2
and PDFGenerated is not null
and PrintedDate is null and brandid=@brandid
select @Cancle = isnull(sum(contentlength),0) from dbo.CorrespondenceBatch where
correspondenceclassid=3
and PDFGenerated is not null
and PrintedDate is null and brandid=@brandid
select @Welcome as Welcome, @Cancle as Cancle
end
CREATE procedure [dbo].[sp_GetQualityAssuranceList](@roleID int, @dealerID int,
@userID int)
as
begin
if(@roleID=1)
begin
select c.Id, c.CustomerId,
c.OrderDate, DATEDIFF(day, c.OrderDate, getdate())age,
c.OrderReferenceString, dbo.Fun_GetUserName(c.EnteredById)EnteredBy, dbo.Fun_CustFullName(c.id)CustName, [dbo].[GetRecordingCountbyOrdID](c.Id) Rec, case isnull(cast(coid as
varchar(50)), '0') when '0' then 'NEW' else 'OLD' end PageType
from CustomerOrder c
where
c.IsDraft = 0 and IsQACompleted=1 --and coid is null
end
else if(@roleID=2 or @roleID=3)
begin
select c.Id, c.CustomerId,
c.OrderDate, DATEDIFF(day, c.OrderDate, getdate())age,
c.OrderReferenceString, dbo.Fun_GetUserName(c.EnteredById)EnteredBy, dbo.Fun_CustFullName(c.id)CustName, [dbo].[GetRecordingCountbyOrdID](c.Id) Rec, case isnull(cast(coid as
varchar(50)), '0') when '0' then 'NEW' else 'OLD' end PageType
from CustomerOrder c
where
c.IsDraft = 0 and IsQACompleted=1 and dealerid = @dealerID --and coid is
null

```



```

        end
    else
        begin
            select c.Id, c.CustomerId,
c.OrderDate,DATEDIFF(day,c.OrderDate,getdate())age,
c.OrderReferenceString,dbo.Fun_GetUserName(c.EnteredById)EnteredBy,dbo.Fun_CustFull
Name(c.id)CustName,[dbo].[GetRecordingCountrybyOrdID](c.Id) Rec,case isnull(cast(coid as
varchar(50)),'0') when '0' then 'NEW' else 'OLD' end PageType
            from CustomerOrder c
            where
                c.IsDraft = 0 and IsQACompleted=1 and enteredbyid= @userID--and coid is
null
        end

end
CREATE procedure [dbo].[sp_GetStandin rderChosenSubProducts](@CustOrdLineId int)
as
begin
    select cp.description,cp.year2monthlyprice,cp.analysiscode from dbo.CallProduct cp
left outer join CustOrdLineSelected cols on
cp.id=cols.cpid
where cp.ProductGroupId<3
and cols.custordlineid=@CustOrdLineId
---The identifier for those items added to an order that are NOT to be generated as a
Standin rder record in the billing system are those that have "ProductGroupCate ry" = 4
within the "CallProduct" table.
end
CREATE procedure [dbo].[sp_GetStandin rdertoSendEbillz]
as
begin
    select dbo.GetCust_MainTelephone(customerorderid)as CustomerIdentifier,
    dbo.GetCust_Code(customerorderid) as CustomerCode,
    dbo.GetCust_orderlinecli(id) as CLI,
    dbo.Get_TariffByPpId(ppid) as Tariff,
    dbo.Get_year2monthlypriceByPpId(PpId) year2monthlyprice,
    CustomerOrderId,Id,isnull(dbo.Get_ProductDescriptionByPpId(PpId),'No Product Found')
Summary,
    OrderDate,planned livedate
,AnalysisCode=isnull(dbo.Get_ProductAnalysisCodeByCustOrdId(customerorderid),dbo.Get
_ProductAnalysisCodeByPpId(PpId))
    from [customerorderline] where status=6 and iscustomeradded=1 and
isnull(IsBillingAPISuccess,0)=0 and isnull(IsLcrAdded,0)=1
    --and isnull(Isinvoiceadded,0)=1
end
create proc [dbo].[sp_InsCommPair](@CallType int=0,@cpid1 int=0 ,@cpid2
int=0,@BaseGroupID int,@CommPairID int=0)
as
begin

declare @shortcode1 varchar(50),@shortcode2 varchar(50),@name1
varchar(250),@name2 varchar(250),@Result varchar(50)
select @shortcode1=shortcode,@name1=name from callproduct where id=@cpid1
select @shortcode2=shortcode,@name2=name from callproduct where id=@cpid2

if (@CallType=1)

```

```

begin
    declare @isduplicate int = 0
    select @isduplicate=count(CommPairingID) from [cmnCommPairing]
    where
    [BaseGroupID] = @BaseGroupID and [CPID1]=@cpid1 and [CPID2]=@cpid2
    if (@isduplicate>0)
        begin
            set @Result='Error: Record already exist.'
        end
    else
        begin
            declare @CommPairingID int
            select @CommPairingID= max([CommPairingID]) + 1 from
[cmnCommPairing]
            INSERT INTO [cmnCommPairing]
                ([CommPairingID]
                ,[BMS_OrderString]
                ,[Description]
                ,[BaseGroupID]
                ,[IsActive]
                ,[CPID1]
                ,[CPID2])
            VALUES
                (@CommPairingID
                ,isnull(@shortcode1,")+
'+isnull(@shortcode2,")
                ,isnull(@name1,")+' | '+isnull(@name2,")
                ,@BaseGroupID
                ,1
                ,@cpid1
                ,@cpid2)
            set @Result='Record inserted successfully.'
        end
    end
else
    begin
        update [cmnCommPairing] set
        [BMS_OrderString]=isnull(@shortcode1,")+' '+isnull(@shortcode2,")
        ,[Description]=isnull(@name1,")+' | '+isnull(@name2,")
        ,[BaseGroupID]=@BaseGroupID
        ,[CPID1]=@cpid1
        ,[CPID2]=@cpid2
        where [CommPairingID]=@CommPairID
        set @Result='Record updated successfully.'
    end
select @Result result
end
Create procedure [dbo].[sp_UpdateRecordingProcessingSettings]
    (@ReadRecordingFilesFrom varchar(1000),@MoveRecordingFilesTo
varchar(1000),@ExecuteProcessAfter char(10),@ExecuteProcessBefore char(10),
@UpdatedBy int)
as
Begin
truncate table RecordingProcessingSettings
INSERT INTO [RecordingProcessingSettings]

```

```

        ([ReadRecordingFilesFrom]
        ,[MoveRecordingFilesTo]
        ,[ExecuteProcessAfter]
        ,[ExecuteProcessBefore]
        ,[UpdatedBy]
        ,[UpdatedOn])
VALUES
    (@ReadRecordingFilesFrom ,@MoveRecordingFilesTo
,@ExecuteProcessAfter ,@ExecuteProcessBefore ,@UpdatedBy,getdate())
End
create PROC [dbo].[spCheckGuest]
    @UserName varchar(100),
    @Password varchar(100),
    @rtn varchar(100) output,
    @IpAddress varchar(50),
    --@MachineName Varchar(100),
    @City Varchar(100) ='- ' ,
    @Country Varchar(100)='- '
AS
begin
    DECLARE @LoginId INT
    DECLARE @UserId INT
    DECLARE @GroupId INT
    DECLARE @status bit
    declare @RoleId int
    declare @DealerId int
    declare @BrandId int
    SET @status = 'false'
    if EXISTS (SELECT UserId FROM tblUser WHERE UserName = @UserName)
    begin
        if exists(select UserId from tbluser where UserName = @UserName and
[IsRemoved] = 'false' and Status= 'true' )
            begin
                -- Check Lock Time period
                DECLARE @Lo natetempts INT
                DECLARE @BloackPeriod INT
                declare @minremaining int
                SELECT @UserId = UserId FROM tblUser WHERE UserName =
@UserName
                SELECT @Lo natetempts=Lo natetempts, @BloackPeriod =
BlockingPeriod FROM tblUserLockOutDetails
                declare @LupdatedOn datetime
                declare @Cmptime datetime
                SELECT @GroupId = GroupId, @DealerId = DealerId, @RoleId =
RoleId, @LupdatedOn=UpdatedOn FROM tblUser WHERE UserId = @UserId
                Select @BrandId=BrandId from DealerBrand where
DealerId=@DealerId
                SELECT @Cmptime = DATEADD(mi, @BloackPeriod, @LupdatedOn)
                declare @lastSucced as int
                declare @FailedCount as int

                IF EXISTS(SELECT UserId FROM tblUser WHERE UserName =
@UserName )
                    BEGIN

```

```

INSERT INTO
tblLogin(LogInTime,UserId,LoginStatus,Sourcelp,City,Country) VALUES(getdate(),
@UserId,'true',@IpAddress,@City,@Country)
SET @rtn = 'Login Successfully'
SET @status = 'true'
SELECT @LoginId = @@IDENTITY
INSERT INTO tblUserLog (PerformAction,
PerformActionTime, UserId, LoginId) VALUES('Logged On', getdate(), @UserId, @LoginId)
SELECT @rtn as ResultStatus,@UserId as
UserId,@GroupId AS GroupId, @UserName as UserName,@status as
UserStatus,@LoginId AS LoginId ,@DealerId as DealerId,@RoleId as RoleId ,@BrandId as
BrandId
END
ELSE
BEGIN
select top 1 @lastSucced=loginid from tblLogin where
Loginstatus=1 and UserId = @UserId order by LoginId desc
if @lastSucced is null
begin
set @lastSucced=0
end
SELECT @FailedCount = count(*) from tbllogin as
TableLogin where loginstatus=0 and userid = @UserId and LoginId > @lastSucced
if(@FailedCount>@Lo n attempts)
begin
UPDATE tblUser SET [Status] = 'false',
UpdatedOn=getdate() WHERE UserId = @UserId
SET @rtn = 'Your password is Locked.
Please contact to system administrator.'
SELECT @rtn as ResultStatus,@UserId
as UserId,@GroupId AS GroupId, @UserName as UserName,@status as
UserStatus,@LoginId as LoginId ,@DealerId as DealerId,@RoleId as RoleId ,@BrandId
as BrandId
end
else
begin
INSERT INTO
tblLogin(LogInTime,UserId,LoginStatus,Sourcelp,City,Country) VALUES(getdate(),
@UserId,'false',@IpAddress,@City,@Country)
SELECT @LoginId = @@IDENTITY
INSERT INTO tblUserLog
(PerformAction, PerformActionTime, UserId, LoginId)VALUES('Failed Lo n', getdate(),
@UserId, @LoginId)
SET @rtn = 'Password does not match'
SELECT @rtn as ResultStatus,@UserId
as UserId,@GroupId AS GroupId, @UserName as UserName,@status as
UserStatus,@LoginId AS LoginId ,@DealerId as DealerId,@RoleId as RoleId ,@BrandId as
BrandId
end
end
END
else
begin

```

```

                SET @rtn = 'User Removed Or Passworded has been locked,Please
Contact Administrator.'
                SELECT @rtn as ResultStatus,@UserId as UserId,@GroupId AS
GroupId, @UserName as UserName,@status as UserStatus,@LoginId as LoginId
,@DealerId as DealerId,@RoleId as RoleId ,@BrandId as BrandId
                    end
end
else
begin
    SET @rtn = 'Username Does not Exists'
    SELECT @rtn as ResultStatus,@UserId as UserId,@GroupId AS GroupId,
@UserName as UserName,@status as UserStatus,@LoginId as LoginId ,@DealerId as
DealerId,@RoleId as RoleId ,@BrandId as BrandId
    end
end
CREATE PROC [dbo].[spCheckLogin]
@UserName varchar(100),
@Password varchar(100),
@rtn varchar(100) output,
@IpAddress varchar(50),
--@MachineName Varchar(100),
@City Varchar(100) ='- ' ,
@Country Varchar(100)='- '
AS
begin
    DECLARE @LoginId INT
    DECLARE @UserId INT
    DECLARE @GroupId INT
    DECLARE @status bit
    declare @RoleId int
    declare @DealerId int
    declare @BrandId int
    SET @status = 'false'
    if EXISTS (SELECT UserId FROM tblUser WHERE UserName = @UserName)
    begin
        if exists(select UserId from tbluser where UserName = @UserName and
[IsRemoved] = 'false' and Status= 'true' )
        begin
            -- Check Lock Time period
            DECLARE @Lo nAttempts INT
            DECLARE @BlockPeriod INT
            declare @minremaining int
            SELECT @UserId = UserId FROM tblUser WHERE UserName =
@UserName
            SELECT @Lo nAttempts=Lo nAttempts, @BlockPeriod =
BlockingPeriod FROM tblUserLockOutDetails
            declare @LupdatedOn datetime
            declare @Cmptime datetime
            SELECT @GroupId = GroupId,@DealerId = DealerId, @RoleId =
RoleId,@LupdatedOn=UpdatedOn FROM tblUser WHERE UserId = @UserId
            Select @BrandId=BrandId from DealerBrand where
DealerId=@DealerId
            SELECT @Cmptime = DATEADD(mi,@BlockPeriod,@LupdatedOn)
            declare @lastSucced as int
            declare @FailedCount as int

```

```

                IF EXISTS(SELECT UserId FROM tblUser WHERE UserName =
@UserName and [Password] = @Password)
                    BEGIN
                        INSERT INTO
tblLogin(LogInTime,UserId,LoginStatus,Sourcelp,City,Country) VALUES(getdate(),
@UserId,'true',@IpAddress,@City,@Country)
                        SET @rtn = 'Login Successfully'
                        SET @status = 'true'
                        SELECT @LoginId = @@IDENTITY
                        INSERT INTO tblUserLog (PerformAction,
PerformActionTime, UserId, LoginId) VALUES('Logged On', getdate(), @UserId, @LoginId)
                        SELECT @rtn as ResultStatus,@UserId as
UserId,@GroupId AS GroupId, @UserName as UserName,@status as
UserStatus,@LoginId AS LoginId ,@DealerId as DealerId,@RoleId as RoleId ,@BrandId as
BrandId
                    END
                ELSE
                    BEGIN
                        select top 1 @lastSucced=loginid from tblLogin where
Loginstatus=1 and UserId = @UserId order by LoginId desc
                        if @lastSucced is null
                            begin
                                set @lastSucced=0
                            end
                        SELECT @FailedCount = count(*) from tbllogin as
TableLogin where loginstatus=0 and userid = @UserId and LoginId > @lastSucced
                        if(@FailedCount>@Lo n attempts)
                            begin
                                UPDATE tblUser SET [Status] = 'false',
UpdatedOn=getdate() WHERE UserId = @UserId
                                SET @rtn = 'Your password is Locked.
Please contact to system administrator.'
                                SELECT @rtn as ResultStatus,@UserId
as UserId,@GroupId AS GroupId, @UserName as UserName,@status as
UserStatus,@LoginId as LoginId ,@DealerId as DealerId,@RoleId as RoleId ,@BrandId
as BrandId
                            end
                        else
                            begin
                                INSERT INTO
tblLogin(LogInTime,UserId,LoginStatus,Sourcelp,City,Country) VALUES(getdate(),
@UserId,'false',@IpAddress,@City,@Country)
                                SELECT @LoginId = @@IDENTITY
                                INSERT INTO tblUserLog
(PerformAction, PerformActionTime, UserId, LoginId)VALUES('Failed Lo n', getdate(),
@UserId, @LoginId)
                                SET @rtn = 'Password does not match'
                                SELECT @rtn as ResultStatus,@UserId
as UserId,@GroupId AS GroupId, @UserName as UserName,@status as
UserStatus,@LoginId AS LoginId ,@DealerId as DealerId,@RoleId as RoleId ,@BrandId as
BrandId
                            end
                    END
            END
    
```

```

        end
    else
        begin
            SET @rtn = 'User Removed Or Passworded has been locked,Please
Contact Administrator.'
            SELECT @rtn as ResultStatus,@UserId as UserId,@Groupld AS
Groupld, @UserName as UserName,@status as UserStatus,@Loginld as Loginld
,@Dealerld as Dealerld,@Roleld as Roleld ,@Brandld as Brandld
        end
    end
else
begin
    SET @rtn = 'Username Does not Exists'
    SELECT @rtn as ResultStatus,@UserId as UserId,@Groupld AS Groupld,
@UserName as UserName,@status as UserStatus,@Loginld as Loginld ,@Dealerld as
Dealerld,@Roleld as Roleld ,@Brandld as Brandld
end
end
create PROC [dbo].[spCheckLoginDet]
@UserName varchar(100),
@Password varchar(100),
@rtn varchar(100) output,
@IpAddress varchar(50),
--@MachineName Varchar(100),
@City Varchar(100) ='-',
@Country Varchar(100)='- '
AS
begin
    DECLARE @Loginld INT
    DECLARE @UserId INT
    DECLARE @Groupld INT
    DECLARE @status bit
    declare @Roleld int
    declare @Dealerld int
    declare @Brandld int
    SET @status = 'false'
    if EXISTS (SELECT UserId FROM tblUser WHERE UserName = @UserName)
    begin
        if exists(select UserId from tbluser where UserName = @UserName and
[IsRemoved] = 'false' and Status= 'true' )
            begin
                -- Check Lock Time period
                DECLARE @Lo n attempts INT
                DECLARE @BloackPeriod INT
                declare @minremaining int
                SELECT @UserId = UserId FROM tblUser WHERE UserName =
@UserName
                SELECT @Lo n attempts=Lo n attempts, @BloackPeriod =
BloackingPeriod FROM tblUserLockOutDetails
                declare @LupdatedOn datetime
                declare @Cmptime datetime
                SELECT @Groupld = Groupld,@Dealerld = Dealerld, @Roleld =
Roleld,@LupdatedOn=UpdatedOn FROM tblUser WHERE UserId = @UserId
                Select @Brandld=Brandld from DealerBrand where
Dealerld=@Dealerld
            end
        end
    end
end

```

```

SELECT @Cmptime = DATEADD(mi, @BloackPeriod, @LupdatedOn)
declare @lastSucced as int
declare @FailedCount as int

IF EXISTS(SELECT UserId FROM tblUser WHERE UserName =
@UserName )
        BEGIN
                INSERT INTO
tblLogin(LogInTime,UserId,LoginStatus,Sourcelp,City,Country) VALUES(getdate(),
@UserId,'true',@IpAddress,@City,@Country)
                SET @rtn = 'Login Successfully'
                SET @status = 'true'
                SELECT @LoginId = @@IDENTITY
                INSERT INTO tblUserLog (PerformAction,
PerformActionTime, UserId, LoginId) VALUES('Logged On', getdate(), @UserId, @LoginId)
                SELECT @rtn as ResultStatus,@UserId as
UserId,@GroupId AS GroupId, @UserName as UserName,@status as
UserStatus,@LoginId AS LoginId ,@DealerId as DealerId,@RoleId as RoleId ,@BrandId as
BrandId
        END
ELSE
        BEGIN
                select top 1 @lastSucced=loginid from tblLogin where
Loginstatus=1 and UserId = @UserId order by LogInId desc
                if @lastSucced is null
                        begin
                                set @lastSucced=0
                        end
                SELECT @FailedCount = count(*) from tbllogin as
TableLogin where loginstatus=0 and userid = @UserId and LogInId > @lastSucced
                if(@FailedCount>@Lo n attempts)
                        begin
                                UPDATE tblUser SET [Status] = 'false',
UpdatedOn=getdate() WHERE UserId = @UserId
                                SET @rtn = 'Your password is Locked.
Please contact to system administrator.'
                                SELECT @rtn as ResultStatus,@UserId
as UserId,@GroupId AS GroupId, @UserName as UserName,@status as
UserStatus,@LoginId as LoginId ,@DealerId as DealerId,@RoleId as RoleId ,@BrandId
as BrandId
                        end
                else
                        begin
                                INSERT INTO
tblLogin(LogInTime,UserId,LoginStatus,Sourcelp,City,Country) VALUES(getdate(),
@UserId,'false',@IpAddress,@City,@Country)
                                SELECT @LoginId = @@IDENTITY
                                INSERT INTO tblUserLog
(PerformAction, PerformActionTime, UserId, LoginId)VALUES('Failed Lo n', getdate(),
@UserId, @LoginId)
                                SET @rtn = 'Password does not match'
                                SELECT @rtn as ResultStatus,@UserId
as UserId,@GroupId AS GroupId, @UserName as UserName,@status as
UserStatus,@LoginId AS LoginId ,@DealerId as DealerId,@RoleId as RoleId ,@BrandId as
BrandId
                        end
        END

```



```

end
END
end
else
begin
SET @rtn = 'User Removed Or Passworded has been locked,Please
Contact Administrator.'
SELECT @rtn as ResultStatus,@UserId as UserId,@Groupld AS
Groupld, @UserName as UserName,@status as UserStatus,@Loginld as Loginld
,@Dealerld as Dealerld,@Roleld as Roleld ,@Brandld as Brandld
end
end
else
begin
SET @rtn = 'Username Does not Exists'
SELECT @rtn as ResultStatus,@UserId as UserId,@Groupld AS Groupld,
@UserName as UserName,@status as UserStatus,@Loginld as Loginld ,@Dealerld as
Dealerld,@Roleld as Roleld ,@Brandld as Brandld
end
end
CREATE PROC [dbo].[spCreatePermissionGrp]
@UserId INT,
@Loginld INT,
@GroupName Varchar(100),
--@xmlstring as xml
@xmlstring xml,
@rtn varchar(max) output
AS
BEGIN
IF EXISTS (SELECT * FROM tblPermissionGroup WHERE GroupName = @GroupName)
BEGIN
SET @rtn = 'Permission Group Name already exists'
END
ELSE
BEGIN
INSERT INTO tblPermissionGroup(Groupname,[Status]) VALUES(@GroupName,1)
DECLARE @Grpld INT
SET @Grpld = @@IDENTITY
DECLARE @XMLDocPointer INT
EXEC sp_xml_preparedocument @XMLDocPointer OUTPUT, @xmlstring
Insert into tblGrpMapping(Groupld,[Pageld],[View],[Add],[Edit],[Delete],CreatedOn)
SELECT @Grpld,GroupID,[View],[Add],[Edit],[Delete],getdate()
FROM OPENXML(@XMLDocPointer,'/root/Data',2) -- /Employee/EmployeeTable is
/DataSetName/Datatable Name that is given in code
WITH
(
GroupID int
,[Add] bit
,[View] bit
,[Edit] bit
,[Delete] bit
)
SET @rtn = 'Successfully Created'
--INSERT INTO tblGrpMapping () VALUES()

```

```

INSERT INTO tblUserLog (PerformAction, PerformActionTime, UserId, LoginId)
VALUES('Created role ('+ @GroupName +') with new permissions', getdate(), @UserId,
@LoginId)
END
END
--SELECT * FROM tblPermissionGroup
--SELECT * FROM tblGrpMapping
CREATE PROC [dbo].[spCreateUser]
@UserId INT,
@LoginId INT,
@UserName varchar(100),
@Password varchar(100),
@FirstName varchar(100),
--@LastName varchar(100),
@EmailId varchar(50),
@Contactnum varchar(50),
@deptment varchar(100),
@region varchar(100),
--@CreatedOn bydefault we wil get
--@Status bit,
--@RoleID int ,
@GroupId int,
@rtn varchar(100) output
AS
BEGIN
IF NOT EXISTS ( SELECT * FROM tblUser WHERE username = @UserName )
BEGIN
If NOT EXISTS( SELECT * FROM tblUser WHERE EmailID = @EmailId )
BEGIN
If NOT EXISTS( SELECT * FROM tblUser WHERE ContactNum = @Contactnum )
BEGIN
INSERT INTO tblUser(Username, [Password],FirstName,
CreatedOn,GroupId,Department,Region,EmailID,ContactNum)
VALUES(@UserName,@Password,@FirstName,getdate(),@GroupId,@deptment,@region,
@EmailId,@Contactnum)
INSERT INTO tblUserLog (PerformAction, PerformActionTime, UserId, LoginId)
VALUES('Added new user ('+ @UserName +')', getdate(), @UserId, @LoginId)
DECLARE @GrpName varchar(100)
select @GrpName= GroupName from tblPermissionGroup where
GroupId=@GroupId
INSERT INTO tblUserLog (PerformAction, PerformActionTime, UserId, LoginId)
VALUES('Added role('+ @GrpName +') to ('+ @UserName +')', getdate(),
@UserId, @LoginId)
SELECT @rtn = 'Successfully Created'
END
ELSE
BEGIN
SELECT @rtn = 'Contact Number Already Exists'
END
END
ELSE
BEGIN
SELECT @rtn = 'EmailId Already Exists'
END
END

```

```

END
ELSE
BEGIN
    SELECT @rtn = 'Username already exists'
END
END
-- SELECT * FROM tblUserLockOutDetails
CREATE PROC [dbo].[spCreateUserLockOut]
    @BlockingPeriod INT,
    @Lo n attempts INT,
    @UserLockOutTime INT,
    @UserId INT,
    @LoginId INT
AS
BEGIN
DECLARE @cnt int
SELECT @cnt= Count(UserLockId) from tblUserLockOutDetails
IF (@cnt!= 0)
BEGIN
    DECLARE @Prv_UsrLokOutTime INT
    SELECT @Prv_UsrLokOutTime = UserLockOutTime FROM tblUserLockOutTimeDevice
    where UserLockOutId = 1
    if (not exists (SELECT * FROM tblUserLockOutTimeDevice where UserLockOutTime =
@UserLockOutTime))
    begin
        INSERT INTO tblUserLog (PerformAction, PerformActionTime, UserId, LoginId)
        VALUES('Set User Lockout Time(in Mins) from '+
CONVERT(varchar,@Prv_UsrLokOutTime) +' to '+ CONVERT(varchar,
@UserLockOutTime) +', getdate(), @UserId, @LoginId)
    end

    DECLARE @Prv_Lo n attempts INT
    DECLARE @Prv_BlockingPeriod INT
    select @Prv_BlockingPeriod = BlockingPeriod, @Prv_Lo n attempts=Lo n attempts from
tblUserLockOutDetails
    where UserLockId = 1
    if(not exists ( select * from tblUserLockOutDetails where lo n attempts = @Lo n attempts))
    begin
        INSERT INTO tblUserLog (PerformAction, PerformActionTime, UserId, LoginId)
        VALUES('Set Number of failed lo n attempts from '+ CONVERT(varchar , @Prv_Lo
n attempts) +' to '+ CONVERT(varchar, @Lo n attempts) +', getdate(), @UserId, @LoginId)
    end
    if(not exists ( select * from tblUserLockOutDetails where BlockingPeriod =
@BlockingPeriod))
    begin
        INSERT INTO tblUserLog (PerformAction, PerformActionTime, UserId, LoginId)
        VALUES('Set Lo n Blocking Period (in Hrs) from '+ CONVERT(varchar
,@Prv_BlockingPeriod) +' to '+ CONVERT(varchar, @BlockingPeriod) +', getdate(),
@UserId, @LoginId)
    end
    UPDATE tblUserLockOutDetails SET Lo n attempts=@Lo
n attempts,BlockingPeriod=@BlockingPeriod,UpdatedOn=getdate() where UserLockId=1
    UPDate tblUserLockOutTimeDevice SET UserLockOutTime=@UserLockOutTime where
UserLockOutId=1
END

```

```

ELSE
BEGIN
  INSERT INTO tblUserLockOutDetails(Lo nAttempts,BloackingPeriod,UpdatedOn)
VALUES(@Lo nAttempts,@BloackingPeriod,getdate())
END
END
-- EXEC spCreateUserLockOut '1',10
-- SELECT * FROM tblUserLockOutTimeDevice
-- select * from tblUserLockOutDetails
CREATE PROC [dbo].[spDisableUser]
  @UserID int,
  @Status bit
AS
BEGIN
UPDATE tblUser SET [Status] = @Status where UserId = @UserID
END
CREATE PROC [dbo].[spGetAllUsers]
  @OrdCol varchar(200),
  @OrdExp Varchar(200),
  @pageNo int,
  @pageSize int
AS
BEGIN
  Declare @MainSql Varchar(2000),
  @firstRecord int,
  @lastRecord int,
  @Lastpage int,
  @RecCnt int
  SET @firstRecord = (@pageNo - 1) * @pageSize + 1
  SET @lastRecord = @firstRecord + @pageSize - 1
  Select @Lastpage = ceiling(convert(decimal,count(userid))/@pageSize), @RecCnt =
count(userid) from tblUser where isremoved =0
if(@Lastpage!=1)
begin
  --set @MainSql = 'select * from ( Select Case When t1.Status = 1 Then 0 else 1 end stat,
'+convert(varchar,@Lastpage)+' NOP ,Row_Number() over(order by '+ @OrdCol + ' )
recno,userid,username,password,firstname,lastname,Case When t1.Status = 1 then "
--Active" else "In-Active" End Status,EmailId,t2.GroupName,ContactNum,
Convert(varchar,t1.Createdon,103)+ CONVERT(CHAR( 5),t1.Createdon,114)
Createdon,t1.RoleId,t1.GroupId from tblUser t1,tblpermissiongroup t2 where t1.GroupId =
t2.GroupId and t1.isRemoved =0) t where recno between ' +
convert(varchar,@firstRecord) +' and '+ convert(varchar,@lastRecord)
set @MainSql = 'select * from ( Select '+convert(varchar,@RecCnt)+' NOR, Case When
t1.Status = 1 Then 0 else 1 end stat, '+convert(varchar,@Lastpage)+' NOP ,Row_Number()
over(order by '+ @OrdCol + ' )
recno,userid,username,password,firstname,lastname,Case When t1.Status = 1 then "
Active" else "In-Active" End Status,EmailId,t2.GroupName,ContactNum,
CONVERT(VARCHAR(10), t1.Createdon, 103) + " " + RIGHT(CONVERT(VARCHAR,
t1.Createdon, 100), 7) Createdon,t1.RoleId,t1.GroupId from tblUser t1,tblpermissiongroup t2
where t1.GroupId = t2.GroupId and t1.isRemoved =0) t where recno between ' +
convert(varchar,@firstRecord) +' and '+ convert(varchar,@lastRecord)
end
else
begin

```

```

set @MainSql = 'select * from ( Select '+convert(varchar,@RecCnt)+' NOR, Case When
t1.Status = 1 Then 0 else 1 end stat, '+convert(varchar,@Lastpage)+' NOP ,Row_Number()
over(order by '+ @OrdCol + ' )
recno,Userid,UserName,Password,FirstName,LastName,Case When t1.Status = 1 then "
Active" else "In-Active" End Status,EmailId,t2.GroupName,ContactNum,
CONVERT(VARCHAR(10), t1.Createdon, 103) +" " + RIGHT(CONVERT(VARCHAR,
t1.Createdon, 100), 7) Createdon,t1.RoleId,t1.GroupId from tblUser t1,tblpermissiongroup t2
where t1.GroupId = t2.GroupId and t1.isRemoved =0) t '
end
EXEC(@MainSql)
--PRINT @MainSql
END
CREATE PROC [dbo].[spGetIDSValue]
@ObjId int,
@netviewId int,
@ObjParamId int
AS
begin
if(exists(select * from ObjectNetworkParameters where netviewId = @netviewId and
[ObjId] = @ObjId and ObjParamId = @ObjParamId))
begin
-- update statement executed
select ObjParamValue from ObjectNetworkParameters
where netviewId = @netviewId and [ObjId] = @ObjId and ObjParamId = @ObjParamId
end
else
begin
-- insert statement executed
select Value as ObjParamValue from ObjectParameterValues
where [ObjId] = @ObjId and ObjParamId = @ObjParamId
end
end
end
CREATE PROC [dbo].[spGetPerSetsforGrp]
@GroupId int
AS
BEGIN
select t1.PageId,t1.PageName,case when parentId is null then 0 else parentId end parentid,
case when [view] =1 then 1 else 0 end [view],
case when [Add] =1 then 1 else 0 end [Add],
case when [Edit] =1 then 1 else 0 end [Edit],
case when [Delete] =1 then 1 else 0 end [Delete]
from tblpages t1 Left outer join
( select * from tblgrpmapping where groupid = @GroupId )t2
on t1.pageid = t2.pageid order by pageid
END
CREATE PROC [dbo].[spGetPerSetsforPage]
@UserName varchar(100),
@PageUrl varchar(100)
AS
BEGIN
Declare @isMenuItem Bit
select @isMenuItem = IsMenuItem from tblpages where PageUrl = @PageUrl
if(@isMenuItem = 1)
Begin
select MapId, t1.PageId,t1.PageName,t1.PageCode,t1.PageUrl,

```

```

case when [view] =1 then 'true' else 'false' end [view],
case when [Add] =1 then 'true' else 'false' end [Add],
case when [Edit] =1 then 'true' else 'false' end [Edit],
case when [Delete] =1 then 'true' else 'false' end [Delete],
t3.username
from tblpages t1, tblgrpMapping t2,tblUser t3 where t1.pageld = t2.Pageld and t3.GroupId
=t2.groupid and t1.PageUrl Like '%'+@PageUrl+'%' and t3.UserName=@UserName
and t3.isremoved =0 and t3.status = 1 and t1.IsMenuItem=1
End
Else if(@isMenuItem = 0)
Begin
WITH n(isMenuItem,Pageld, Pagename, [level],
parentId,PageCode,PageUrl,concatador) AS
(
SELECT isMenuItem, Pageld, Pagename,[level],parentId,PageCode,PageUrl,
(''+CONVERT(VARCHAR (MAX), Pageld)+' - ''+CONVERT(VARCHAR (MAX), [level])) as
concatador
FROM tblpages
WHERE PageUrl Like '%'+@PageUrl+'%'
UNION ALL
SELECT m.isMenuItem, m.Pageld,
m.Pagename,m.level,m.parentId,m.PageCode,m.PageUrl,
n.concatador+' * (''+CONVERT (VARCHAR (MAX), case when ISNULL(m.parentId, 0)
= 0 then 0 else m.Pageld END)+' - ''+CONVERT(VARCHAR (MAX), m.level)+'') as
concatador
FROM tblpages as m, n
WHERE n.parentId = m.Pageld
)
select top 1 MapId, t1.Pageld,t1.PageName,t1.PageCode,t1.PageUrl,
case when [view] =1 then 'true' else 'false' end [view],
case when [Add] =1 then 'true' else 'false' end [Add],
case when [Edit] =1 then 'true' else 'false' end [Edit],
case when [Delete] =1 then 'true' else 'false' end [Delete],
t3.username
from n t1, tblgrpMapping t2,tblUser t3 where t1.pageld = t2.Pageld and t3.GroupId
=t2.groupid and t1.PageUrl Not Like '%'+@PageUrl+'%' and t3.UserName=@UserName
and t3.isremoved =0 and t3.status = 1
End
END
--sp_help text spGetSourceIp
CREATE PROC [dbo].[spGetSourceIp]
@OrdCol varchar(200),
@OrdExp Varchar(200),
@pageNo int,
@pageSize int
AS
BEGIN
Declare @MainSql Varchar(2000),
@firstRecord int,
@lastRecord int,
@Lastpage int,
@RecCnt int
SET @firstRecord = (@pageNo - 1) * @pageSize + 1
SET @lastRecord = @firstRecord + @pageSize - 1

```

```

SELECT @Lastpage = ceiling(convert(decimal,count(*)/@pageSize) , @RecCnt = count(*)
FROM tblLogin as lg INNER JOIN tblUser
as us ON us.UserId= lg.UserId
-----
if(@Lastpage!=1)
begin
--set @MainSql = 'select * from ( Select Case When t1.Status = 1 Then 0 else 1 end stat,
'+convert(varchar,@Lastpage)+' NOP ,Row_Number() over(order by '+ @OrdCol + ' )
recno,UserId,UserName>Password,FirstName,LastName,Case When t1.Status = 1 then "
--Active" else "In-Active" End Status,EmailId,t2.GroupName>ContactNum,
Convert(varchar,t1.Createdon,103)+ CONVERT(CHAR( 5),t1.Createdon,114)
Createdon,t1.RoleId,t1.GroupId from tblUser t1,tblpermissiongroup t2 where t1.GroupId =
t2.GroupId and t1.isRemoved =0) t where recno between ' +
convert(varchar,@firstRecord) +' and '+ convert(varchar,@lastRecord)
set @MainSql = 'select * from(SELECT '+convert(varchar,@RecCnt)+' NOR,
'+convert(varchar,@Lastpage)+' NOP, Row_Number() Over(order by '+ @OrdCol + ' '+
@OrdExp +') as
Rowcnt,lg.SourceIp,lg.MachineName,us.UserName,CONVERT(VARCHAR(10),lg.LoginTime,103) as
DateofLogedIn,lg.City,lg.Country,U.PerformAction as
Description,U.PerformActionTime,U.UserId,U.LoginId FROM tblLogin as lg INNER JOIN
tblUser
as us ON us.UserId= lg.UserId
LEFT OUTER JOIN tblUserLog as U ON U.LoginId = lg.LoginId) t where Rowcnt between ' +
+ convert(varchar,@firstRecord) +' and '+ convert(varchar,@lastRecord)
--+ ' ORDER BY DateofLogedIn DESC'
end
else
begin
set @MainSql = 'select * from(SELECT '+convert(varchar,@RecCnt)+' NOR,
'+convert(varchar,@Lastpage)+' NOP, Row_Number() Over(order by '+ @OrdCol + ' '+
@OrdExp +') as
Rowcnt,lg.SourceIp,lg.MachineName,us.UserName,CONVERT(VARCHAR(10),lg.LoginTime,103) as
DateofLogedIn,lg.City,lg.Country,U.PerformAction as
Description,U.PerformActionTime,U.UserId,U.LoginId FROM tblLogin as lg INNER JOIN
tblUser
as us ON us.UserId= lg.UserId
LEFT OUTER JOIN tblUserLog as U ON U.LoginId = lg.LoginId) t '
--ORDER BY DateofLogedIn DESC'
end
EXEC(@MainSql)
print @MainSql
END
-- SELECT CONVERT(VARCHAR(10),LoginTime,101) as DateofLogedIn,LoginTime FROM
tblLogin
-- select * from tblLogin
CREATE PROC [dbo].[spGetUserInfo]
@UserId int
AS
BEGIN
select
userId,UserName,[Password],FirstName,LastName,EmailId,GroupId>ContactNum,Departme
nt,Region from tblUser where UserId = @UserId
END

```

```

CREATE PROC [dbo].[spGetUserLoclOutDetails]
AS
BEGIN
    SELECT Lo n attempts, BlockingPeriod FROM tblUserLockOutDetails where UserLockId=1
END
CREATE PROC [dbo].[spGetUserLoclOutTimeDevice]
AS
BEGIN
    SELECT top 1 * from tblUserLockOutTimeDevice
END
CREATE PROC [dbo].[spLo ut]
    @UserId INT,
    @LogInId INT
AS
BEGIN
UPDATE tblLogin SET Lo utTime=getdate() WHERE UserId = @UserId and LogInId
=@LogInId
INSERT INTO tblUserLog (PerformAction, PerformActionTime, UserId, LogInId)
VALUES('Logged Off', getdate(), @UserId, @LogInId)
END
Create PROC [dbo].[spRemoveUser]
    @UserID int,
    @Status bit
AS
BEGIN
UPDATE tblUser SET [IsRemoved] = @Status where UserId = @UserID
END
CREATE PROC [dbo].[spUpdatePermissionGrp]
    @xmlstring xml,
    @GrpId INT,
    @UserId INT,
    @LoginId INT
AS
BEGIN
    DECLARE @XMLDocPointer INT
    EXEC sp_xml_preparedocument @XMLDocPointer OUTPUT, @xmlstring
    CREATE TABLE #temp_tblGrpMapping ([PagelId] INT,
    [View] bit,
    [Add] bit,
    [Edit] bit,
    [Delete] bit)
    INSERT INTO #temp_tblGrpMapping([PagelId],[View],[Add],[Edit],[Delete])
    SELECT GroupID,[View],[Add],[Edit],[Delete]
    FROM OPENXML(@XMLDocPointer,'/root/Data',2) -- /Employee/EmployeeTable is
/DataSetName/Datatable Name that is given in code
    WITH
    (
        GroupID int
        , [View] bit
        , [Add] bit
        , [Edit] bit
        , [Delete] bit
    )
-- UPDATE
-- grp

```



```

--SET
-- [add]=tmp.[add],[edit]=tmp.edit,[delete]=tmp.[delete],[view]=tmp.[view]
--FROM
-- tblGrpMapping grp
-- JOIN
-- #temp_tblGrpMapping tmp ON grp.pageid=tmp.pageid
--WHERE
-- grp.groupid=@Grpld
SET NOCOUNT ON
-- declare all variables!
DECLARE @Pageld int,
        @View bit,
        @Add bit,
        @Edit bit,
        @Delete bit
-- declare the cursor
DECLARE Customer CURSOR FOR
SELECT [Pageld]
      ,[View]
      ,[Add]
      ,[Edit]
      ,[Delete]
FROM #temp_tblGrpMapping
OPEN Customer
FETCH Customer INTO @Pageld,
                  @View,
                  @Add,
                  @Edit,
                  @Delete
WHILE @@Fetch_Status = 0
BEGIN
if exists(select * from tblGrpMapping where GroupId = @Grpld and PageId = @Pageld)
BEGIN
UPDATE tblGrpMapping
SET
[add]=@Add,[edit]=@Edit,[delete]=@Delete,[view]=@View
where GroupId = @Grpld and PageId = @Pageld
END
else
BEGIN
INSERT INTO tblGrpMapping (GroupId,PageId,[view],[add],edit,[delete],CreatedOn)
VALUES(@Grpld,@Pageld,@View,@Add,@Edit,@Delete,getdate())
END
FETCH Customer INTO @Pageld,
                  @View,
                  @Add,
                  @Edit,
                  @Delete
END
CLOSE Customer
DEALLOCATE Customer
DECLARE @GrpName varchar(100)
select @GrpName = GroupName from tblPermissionGroup where GroupId = @Grpld
INSERT INTO tblUserLog (PerformAction, PerformActionTime, UserId, LoginId)

```

```

VALUES('Edited role ('+ @GrpName +' ) with new permissions', getdate()),
@UserId, @LoginId)
END
CREATE PROC [dbo].[spUpdateUser]
@UserId int,
@LoginId int,
--@UserName varchar(100),
@NewPassword varchar(100),
@FirstName varchar(100),
@EmailId varchar(50),
@Contactnum varchar(50),
@deptment varchar(100),
@region varchar(100),
@GroupId int
--@Status bit
--@rtn varchar(100) output
AS
BEGIN
DECLARE @UserName Varchar(100)
select @UserName = UserName from tblUser where UserId = @UserId
INSERT INTO tblUserLog (PerformAction, PerformActionTime, UserId, LoginId)
VALUES('Edited user ('+ @UserName +' ) with new details', getdate(), @UserId,
@LoginId)
DECLARE @GrpName varchar(100)
select @GrpName= GroupName from tblPermissionGroup where
GroupId=@GroupId
INSERT INTO tblUserLog (PerformAction, PerformActionTime, UserId, LoginId)
VALUES('Changed Role('+ @GrpName +' ) to ('+ @UserName +' )', getdate(),
@UserId, @LoginId)
PRINT @NewPassword
if (@NewPassword != 'NA')
UPDATE tblUser SET FirstName= @FirstName, EmailID=@EmailId,
ContactNum=@Contactnum,[Password]=
@NewPassword,GroupId=@GroupId,Department=@deptment,Region=@region WHERE
UserId = @UserId
ELSE
UPDATE tblUser SET FirstName= @FirstName, EmailID=@EmailId,
ContactNum=@Contactnum,GroupId=@GroupId,Department=@deptment,Region=@region
WHERE UserId = @UserId
END

```

Chapter 3

Entity classes:

These classes can be downloaded from Dropbox using this link

<https://www.dropbox.com/scl/fi/jst23q0stp7r1hxixprpi/DowBEntity.rar?rlkey=jk4d4wfvjshsn1xm1mo88blsic&st=pxmuvrbv&dl=0>

or use the following code:

Name Space DOWBEntity

Note: these classes can be saved as one single cs file or separate files. Each file should start with Namespace.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace DowBEntity
{
    public class EteTbluser_address
    {
        public int userid { get; set; }
        public int address_type { get; set; }
        public string address1 { get; set; }
        public string address2 { get; set; }
        public string address3 { get; set; }
        public string pincode { get; set; }
        public string city { get; set; }
        public int country { get; set; }
        public string countryname { get; set; }
        public string ContactNo { get; set; }
        public string address_typedesc { get; set; }
        public bool IsDefault { get; set; }
    }
    public class EteTbluser_add_type
    {
        public int adid{ get; set; }
        public string adtype{ get; set; }
    }
    public class EteCountry
    {
        public int id { get; set; }
        public string shortcd { get; set; }
        public string countryname { get; set; }
        public bool IsBlocked { get; set; }
        public System.Data.Linq.Binary CFlag { get; set; }
    }
    public class EteTblUser_emails
    {
        public int EmailId { get; set; }
```

```

    public int EMSentUserId { get; set; }
    public int EMRcvdUserId { get; set; }
    public string EmSubject { get; set; }
    public string EmBody { get; set; }
    public DateTime CreatedOn { get; set; }
    public int UserId { get; set; }
    public string EMSentUserName { get; set; }
    public string EMRcvdUserName { get; set; }
}
public class EtetblUserImages
{
    public int UserId { get; set; }
    public int srno { get; set; }
    public System.Data.Linq.Binary Images { get; set; }
    public string ImageUrl{get;set;}
}
}
// VBConversions Note: VB project level imports
using System.Collections.Generic;
using System;
using System.Threading.Tasks;
using System.Diagnostics;
using Microsoft.VisualBasic;
using System.Xml.Linq;
using System.Collections;
using System.Linq;
// End of VB project level imports
namespace DowBEntity
{
    public class PopDDL
    {
        //Fills all dropdowns
        public string Value
        {
            get;
            set;
        }
        public string Text
        {
            get;
            set;
        }
    }
    public class ResultError
    {
        public bool Result { get; set; }
        public string ErrMessage { get; set; }
    }
}
public class ProductCategoryGroups
{
    public int productgroupid { set; get; }
    public int productgroupcategory { set; get; }
    public System.Nullable<System.DateTime> CreatedOn { get; set; }
}

```

```

        public System.Nullable<System.DateTime> UpdatedOn { get; set; }
        public string lbtnimage { get; set; }
        public int UserId { get; set; }
    }
public class qadata
    {
        public int Id { get; set; }
        public string ddlval { get; set; }
        public string txtval { get; set; }
    }
public class EteQueueList
    {
        public int Id { get; set; }
        public string Name { get; set; }
        public System.Nullable<int> RaisedById { get; set; }
        public System.Nullable<int> Total { get; set; }
        public System.Nullable<int> OverDue { get; set; }
        public System.Nullable<int> OutStanding { get; set; }
        public System.Nullable<int> Unassigned { get; set; }
        public int Pageld { get; set; }
        public int Uid { get; set; }
    } using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace DowBEntity
{
    public class UserPermissions
    {
        private int _MapId;
        public int MapId
        {
            get { return _MapId; }
            set { _MapId = value; }
        }
        private int _Pageld;
        public int Pageld
        {
            get { return _Pageld; }
            set { _Pageld = value; }
        }
        private int _ParentId;
        public int ParentId
        {
            get { return _ParentId; }
            set { _ParentId = value; }
        }
    }
}

```

```
private string _Hir;
public string Hir
{
    get { return _Hir; }
    set { _Hir = value; }
}
private string _PageName;
public string PageName
{
    get { return _PageName; }
    set { _PageName = value; }
}
private string _PageCode;
public string PageCode
{
    get { return _PageCode; }
    set { _PageCode = value; }
}
private string _PageUrl;
public string PageUrl
{
    get { return _PageUrl; }
    set { _PageUrl = value; }
}
private bool _View;
public bool View
{
    get { return _View; }
    set { _View = value; }
}
private bool _Add;
public bool Add
{
    get { return _Add; }
    set { _Add = value; }
}
private bool _Edit;
public bool Edit
{
    get { return _Edit; }
    set { _Edit = value; }
}
private bool _Delete;
public bool Delete
{
    get { return _Delete; }
    set { _Delete = value; }
}
```

```

private string _UserName;
public string UserName
{
    get { return _UserName; }
    set { _UserName = value; }
}
public int UId { get; set; }
}
public class mapids
{
    public string mpageid { get; set; }
    public string status { get; set; }
    public string buttonid { get; set; }
    public int pageid { get; set; }
}
}
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace DowBEntity
{
    public class BEMyScratchpad
    {
        public int Id
        {
            set;
            get;
        }
        public System.DateTime CreatedOn
        {
            set;
            get;
        }
        public int BMSUserId
        {
            set;
            get;
        }
        public System.DateTime EntryDate
        {
            set;
            get;
        }
        public string Content
        {
            set;

```

```

        get;
    }
    public int PageId { get; set; }
    public int UId { get; set; }
}
}
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace DowBEntity
{
    public class BusinessException : Exception
    {
        public BusinessException()
        {
        }
        public BusinessException(string message)
            : base(message)
        {
        }
        public BusinessException(string message, Exception inner)
            : base(message, inner)
        {
        }
    }
}
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace DowBEntity
{
    public class ClsLoggedInUser
    {
        public string UserName;
        public string UserId;
        public string GroupId;
        public string LoginStatus;
        public int LoginID;
        public int DealerId;
        public int RoleId;
        public int BrandId;
    }
}
using System.Configuration;

```



```

namespace DowBEntity
{
sealed public class Configuration
{
private Configuration()
{
}
internal static string ConnectionString
{
get
    {
        return ConfigurationManager.ConnectionStrings["DowConectingString"].ConnectionString;
    }
}
}
}
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace DowBEntity
{
public class CreateUser
{
private string _uname;
private string _password;
private string _fname;
private string _lname;
private string _emailId;
private string _contactnum;
public string Uname
{
get
    {
        return _uname;
    }
set
    {
        _uname = value;
    }
}
public string Password
{
get
    {
        return _password;
    }
}
}
}

```

```
    set
    {
        _password = value;
    }
}
public string FirstName
{
    get
    {
        return _fname;
    }
    set
    {
        _fname = value;
    }
}
public string LastName
{
    get
    {
        return _lname;
    }
    set
    {
        _lname = value;
    }
}
public string EmailID
{
    get
    {
        return _emailid;
    }
    set
    {
        _emailid = value;
    }
}
public string ContactNum
{
    get
    {
        return _contactnum;
    }
    set
    {
        _contactnum = value;
    }
}
```

```
}
private string _dept;
public string Department
{
    get
    {
        return _dept;
    }
    set
    {
        _dept = value;
    }
}
private string _region;
public string Region
{
    get
    {
        return _region;
    }
    set
    {
        _region = value;
    }
}
private string _newpwd;
public string NewPwd
{
    get
    {
        return _newpwd;
    }
    set
    {
        _newpwd = value;
    }
}
private int _GroupId;
public int GroupId
{
    get
    {
        return _GroupId;
    }
    set
    {
        _GroupId = value;
    }
}
```

```

    }
    public int PageId { get; set; }
    public int UId { get; set; }
}
}
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace DowBEntity
{
    public static class DateTimeExtension
    {
        public static DateTime StartOfWeek(this DateTime value)
        {
            DateTime date = value;
            while (date.DayOfWeek != DayOfWeek.Sunday)
                date = date.AddDays(-1);
            return date;
        }
        public static DateTime EndOfWeek(this DateTime value)
        {
            DateTime date = value;
            while (date.DayOfWeek != DayOfWeek.Saturday)
                date = date.AddDays(1);
            return date;
        }
    }
}
using System;
using System.IO;
using System.Net;
using System.Net.Mail;
namespace DowBEntity
{
    class Email
    {
        private String toList{ get; set; } //Separated by ;
        private String messageBody { get; set; }
        private String attachmentFilePath { get; set; }
        private String isBodyHTML { get; set; }
        private String subject { get; set; }
        private String smtpHost = "smtp.gmail.com";
        private String fromEmail = "TBD";
        private String fromEmailPassword = "TBD";
        private Int16 smtpPort = 587;
        public void sendEmail()
    }
}

```

```

    {
        MailMessage mm = new MailMessage(fromEmail, toList);
        mm.Subject = subject;
        mm.Body = messageBody;
        if (attachmentFilePath != null && attachmentFilePath.Length > 0)
        {
            mm.Attachments.Add(new Attachment(File.OpenWrite(attachmentFilePath), attachmentFilePath), attachmentFilePath);
        }
        mm.IsBodyHtml = false;
        SmtplibClient smtp = new SmtplibClient();
        smtp.Host = "smtp.gmail.com";
        smtp.EnableSsl = true;
        NetworkCredential NetworkCred = new NetworkCredential(fromEmail, fromEmailPassword);
        smtp.UseDefaultCredentials = true;
        smtp.Credentials = NetworkCred;
        smtp.Port = 587;
        smtp.Send(mm);
    }
}
} using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace DowBEntity
{
    public class EteActionTask
    {
        public int Id { get; set; }
        public System.DateTime CreatedOn { get; set; }
        public System.Nullable<int> QueueId { get; set; }
        public System.Nullable<int> OwnedById { get; set; }
        public System.Nullable<int> RaisedById { get; set; }
        public System.Nullable<int> ActionedById { get; set; }
        public System.Nullable<System.DateTime> RaisedDate { get; set; }
        public System.Nullable<System.DateTime> ActionedDate { get; set; }
        public char Status { get; set; }
        public string Summary { get; set; }
        public string StatusName { get; set; }
        public string Details { get; set; }
        public int CallingPageId { get; set; }
        public int SubjectId { get; set; }
        public System.Nullable<System.DateTime> DueDate { get; set; }
        public char TaskType { get; set; }
        public bool TaskOutcome { get; set; }
        public string OutcomeNotes { get; set; }
        public string OwnedByName { get; set; }
        public string ActionedByName { get; set; }
    }
}

```

```

    public string CLI { get; set; }
    public int PageId { get; set; }
    public int UId { get; set; }
    public string RelateTo { get; set; }
    public string RelateToURL { get; set; }
}
public class EteActionTaskFilter
{
    public int _Id { get; set; }
    public System.Nullable<int> QueueId { get; set; }
    public System.Nullable<int> OwnedById { get; set; }
    public System.Nullable<int> RaisedById { get; set; }
    public System.Nullable<int> ActionedById { get; set; }
    public System.Nullable<System.DateTime> RaisedDate { get; set; }
    public System.Nullable<System.DateTime> ActionedDate { get; set; }
    public char Status { get; set; }
    public string Summary { get; set; }
    public string Details { get; set; }
    public int CallingPageId { get; set; }
    public int SubjectId { get; set; }
    public System.Nullable<System.DateTime> DueDate { get; set; }
    public char TaskType { get; set; }
    public bool TaskOutcome { get; set; }
    public string OutcomeNotes { get; set; }
    public string CLI { get; set; }
    public System.DateTime CreatedOn { get; set; }
    public int UserId { get; set; }
    public string UserName { get; set; }
    public string Password { get; set; }
    public string FirstName { get; set; }
    public string LastName { get; set; }
    public string EmailID { get; set; }
    public string ContactNum { get; set; }
    public System.Nullable<System.DateTime> CreatedOn1 { get; set; }
    public System.Nullable<bool> Status1 { get; set; }
    public System.Nullable<int> RoleID { get; set; }
    public System.Nullable<int> GroupId { get; set; }
    public bool IsRemoved { get; set; }
    public string Department { get; set; }
    public string Region { get; set; }
    public System.Nullable<System.DateTime> UpdatedOn { get; set; }
    public int DealerId { get; set; }
}
} using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

```

```

namespace DowBEntity
{
    public class EteActionTasksByFilterResult
    {
        private int Id { set; get; }
        private System.Nullable<int> QueueId { set; get; }
        private System.Nullable<int> OwnedById { set; get; }
        private System.Nullable<int> RaisedById { set; get; }
        private System.Nullable<int> ActionedById { set; get; }
        private System.Nullable<System.DateTime> RaisedDate { set; get; }
        private System.Nullable<System.DateTime> ActionedDate { set; get; }
        private char Status { set; get; }
        private string Summary { set; get; }
        private string Details { set; get; }
        private int CallingPageId { set; get; }
        private int SubjectId { set; get; }
        private System.Nullable<System.DateTime> DueDate { set; get; }
        private char TaskType { set; get; }
        private bool TaskOutcome { set; get; }
        private string OutcomeNotes { set; get; }
        private string CLI { set; get; }
        private System.DateTime CreatedOn { set; get; }
        private string RelateTo { set; get; }
        private string RelateToURL { set; get; }
        private int UserId { set; get; }
        private string UserName { set; get; }
        private string Password { set; get; }
        private string FirstName { set; get; }
        private string LastName { set; get; }
        private string EmailID { set; get; }
        private string ContactNum { set; get; }
        private System.Nullable<System.DateTime> uCreatedOn { set; get; }
        private System.Nullable<bool> ustatus { set; get; }
        private System.Nullable<int> RoleID { set; get; }
        private System.Nullable<int> GroupId { set; get; }
        private bool IsRemoved { set; get; }
        private string Department { set; get; }
        private string Region { set; get; }
        private System.Nullable<System.DateTime> UpdatedOn { set; get; }
        private int DealerId { set; get; }
    }
}
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace DowBEntity

```

```

{
public class EteAllColumns
{
    public string tabname { get; set; }
    public string colname { get; set; }
    public int columnid { get; set; }
    public bool isnullable { get; set; }
}
}
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Xml;
namespace DowBEntity
{
public class EteAuditTrail
{
    public int Id { get; set; }
    public System.DateTime? CreatedOn { get; set; }
    public System.Nullable<int> UserId { get; set; }
    public System.Nullable<int> Objectid { get; set; }
    public string Category { get; set; }
    public string Message { get; set; }
    public string ObjectName { get; set; }
    public System.Xml.Linq.XElement ExtraInfo { get; set; }
    // public XmlDocument ExtraInfo { get; set; }
    public int Pageld { get; set; }
    public System.Nullable<System.DateTime> AuditDate { get; set; }
}
}
public class EteLoginDetails
{
    public System.Nullable<int> UserId { get; set; }
    public System.Nullable<int> Pageld { get; set; }
    public string PageName { set; get; }
}
}
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace DowBEntity
{
public class EteBankHoliday
{
    public int Id { get; set; }
}
}

```



```

        public System.DateTime HolidayDate { get; set; }
        public string Description { get; set; }
        public bool Recurring { get; set; }
        public System.DateTime CreatedOn { get; set; }
        public string HolidayDateString { get; set; }
        public int PageId { get; set; }
        public int UId { get; set; }
    }
}
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace DowBEntity
{
    public class EteBrand
    {
        public int Id { get; set; }
        public string ShortCode { get; set; }
        public string BrandName { get; set; }
        public int OpalCredentialId { get; set; }
        public string DefaultInvoiceMask { get; set; }
        public string EmailInvoiceMask { get; set; }
        public string BusinessInvoiceMask { get; set; }
        public string BusinessEmailInvoiceMask { get; set; }
        public string FromEmailAddress { get; set; }
        public string ForwardLinksRoot { get; set; }
        public string FilePath { get; set; }
        public System.DateTime CreatedOn { get; set; }
    }
    public class EteBrandPrintRun
    {
        public int R2PWelcomeCount { get; set; }
        public int R2PCancelCount { get; set; }
        public int WelcomeCount { get; set; }
        public int CancelCount { get; set; }
    }
}
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace DowBEntity
{
    public class EteCancellationReason
    {

```

```

        public int Id { get; set; }
        public string Name { get; set; }
        public string Description { get; set; }
        public bool IsActive { get; set; }
        public int? ActionQueueId { get; set; }
        public string ActionTaskSummary { get; set; }
        public string ActionTaskDetails { get; set; }
        public System.DateTime CreatedOn { get; set; }
        public System.Nullable<int> UserId { get; set; }
        public int PageId { get; set; }
        public int UId { get; set; }
    }
}
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace DowBEntity
{
    public class EteDealer
    {
        public int DealerId { get; set; }
        public string CompanyName { get; set; }
        public bool IsActive { get; set; }
        public string UserNamePrefix { get; set; }
        public bool AutoProcessCommissions { get; set; }
        public int DealerOperationProfileId { get; set; }
        public System.DateTime CreatedOn { get; set; }
        public System.DateTime UpdatedOn { get; set; }
        public int BrandId { get; set; }
        public int PageId { get; set; }
        public int UId { get; set; }
        public string EMailString { get; set; }
    }
}
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace DowBEntity
{
    public class EteDealerBrand
    {
        public int DealerId { get; set; }
        public int BrandId { get; set; }
        public string ShortCode { get; set; }
    }
}

```

```

        public string BrandName { get; set; }
        public string FilePath { get; set; }
        public System.DateTime CreatedTime { get; set; }
        public int PageId { get; set; }
        public int UId { get; set; }
    }
}
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace DowBEntity
{
    public class EteDept
    {
        public int _DeptId { get; set; }
        public string _DeptName { get; set; }
        public bool _IsActive { get; set; }
        public System.DateTime _CreatedOn { get; set; }
        public int PageId { get; set; }
        public int UId { get; set; }
    }
}
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace DowBEntity
{
    public class EteFileUpDown
    {
        public System.Data.Linq.Binary AttachmentData { get; set; }
        public string AttachmentFilename { get; set; }
        public string AttachmentType { get; set; }
        public System.Nullable<int> AttachmentLength { get; set; }
    }
}
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace DowBEntity
{
    public class EteFlag
    {

```

```

    public int Id { get; set; }
    public Int64 SubjectId { get; set; }
    public int UserId { get; set; }
    public System.DateTime CreatedOn { get; set; }
    public string Reason { get; set; }
    public System.Nullable<int> CallingPageId { get; set; }
    public string UserName { get; set; }
    public string Details { get; set; }
    public int PageId { get; set; }
    public int UId { get; set; }
}
}
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Data.Linq;
using System.Data.Linq.Mapping;
namespace DowBEntity
{
    public class EteHoldReason
    {
        public int Id { get; set; }
        public string Name { get; set; }
        public string Description { get; set; }
        public bool IsActive { get; set; }
        public System.Nullable<int> ActionQueueId { get; set; }
        public string ActionTaskSummary { get; set; }
        public string ActionTaskDetails { get; set; }
        public System.DateTime CreatedOn { get; set; }
        public System.Nullable<int> CreatedBy { get; set; }
        public int PageId { get; set; }
        public int UId { get; set; }
    }
}
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace DowBEntity
{
    public class EteNewsItem
    {
        public int Id { get; set; }
        public string Summary { get; set; }
        public string Details { get; set; }
    }
}

```

```

    public System.DateTime PublishDate { get; set; }
    public System.DateTime ExpireDate { get; set; }
    public int PublishedById { get; set; }
    public string UserName { get; set; }
    public System.DateTime CreatedOn { get; set; }
    public int PageId { get; set; }
    public int UId { get; set; }
}
public class EteCommonCount
{
    public System.Nullable<int> Today { set; get; }
    public System.Nullable<int> Yesterday { set; get; }
    public System.Nullable<int> Last7Days { set; get; }
    public System.Nullable<int> Last14Days { set; get; }
    public System.Nullable<int> Last28Days { set; get; }
    public System.Nullable<int> ThisMonth { set; get; }
    public System.Nullable<int> ThisYear { set; get; }
    public System.Nullable<int> AllTime { set; get; }
    public System.Nullable<int> GetOpenDraftCount { set; get; }
    public System.Nullable<int> Provisioned { set; get; }
    public System.Nullable<int> Live { set; get; }
    public System.Nullable<int> CollingOff { set; get; }
    public System.Nullable<int> ProvisionedOnHold { set; get; }
    public System.Nullable<int> CollingOffOnHold { set; get; }
    public System.Nullable<decimal> SalesperDayLast7Days { set; get; }
    public System.Nullable<decimal> SalesperDayLast28Days { set; get; }
    public System.Nullable<int> CancellationPerLast7Days { set; get; }
    public System.Nullable<int> CancellationPerLast28Days { set; get; }
}
public class EteMostRecentOrders
{
    public int id { set; get; }
    public string orderreferencestring { set; get; }
    public string CustName { set; get; }
    public System.Nullable<int> status { set; get; }
    public string NavURL { set; get; }
}
public class EteTblcourosalitems
{
    public int id { get; set; }
    public string fpath { get; set; }
    public string aname { get; set; }
    public int userid { get; set; }
    public DateTime createdon { get; set; }
}
}
using System;
using System.Collections.Generic;

```

```

using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Data.Linq;
using System.Data.Linq.Mapping;
namespace DowBEntity
{
    public class EteNote
    {
        public int Id { get; set; }
        public System.DateTime CreatedOn { get; set; }
        public int NoteTypeId { get; set; }
        public string NoteType { get; set; }
        public int SubjectId { get; set; }
        public int CallingPageId { get; set; }
        public System.DateTime NoteDate { get; set; }
        public string Body { get; set; }
        public int CreatedById { get; set; }
        public string CreatedBy { get; set; }
        public System.Data.Linq.Binary AttachmentData { get; set; }
        public string AttachmentFilename { get; set; }
        public string AttachmentType { get; set; }
        public System.Nullable<int> AttachmentLength { get; set; }
        public int PageId { get; set; }
        public int UId { get; set; }
        public string NoteSource { get; set; }
    }
}
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace DowBEntity
{
    public class EteNoteType
    {
        public int Id { get; set; }
        public string Name { get; set; }
        public string Description { get; set; }
        public bool IsActive { get; set; }
        public bool IsDefault { get; set; }
        public System.DateTime CreatedOn { get; set; }
        public int PageId { get; set; }
        public int UId { get; set; }
    }
}
using System;

```

```

using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace DowBEntity
{
    public class Eteproduct
    {
        public decimal ProductId { get; set; }
        public string ProductName { get; set; }
        public string ShortCode { set; get; }
        public int ProductGroupId { get; set; }
        public int ProductGroupCategory { get; set; }
        public string Type { get; set; }
        public decimal Price { get; set; }
        public int PackageType { get; set; }
        public int PageId { get; set; }
        public int UId { get; set; }
        public int CurrencyId { get; set; }
        public string AnalysisCode { get; set; }
        public DateTime CreatedOn { get; set; }
        public DateTime UpdatedOn { get; set; }
        public int UserId { get; set; }
        public string PrId { get; set; }
        public int MinOrder { get; set; }
        public int DelTlme { get; set; }
        public bool Check { get; set; }
        public string CatName { get; set; }
        public string GroupName { get; set; }
        public string PrType { get; set; }
        public int SellerId { get; set; }
        public int PriceUnits { get; set; }
        public int PriceUomId { get; set; }
        public int PricePackId { get; set; }
        public int MinOrderUom { get; set; }
        public int DelTimeType { get; set; }
        public int DealerId { get; set; }
        public string imageurl { get; set; }
        public int stype { get; set; }
        public int level { get; set; }
    }
    public class Eteproductcat
    {
        public int pcatid { get; set; }
        public string pcatdesc { get; set; }
        public string lbtnimage { get; set; }
        public DateTime CreatedOn { get; set; }
        public DateTime UpdatedOn { get; set; }
    }
}

```

```

        public int UserID { get; set; }
        public bool active { get; set; }
        public int GCount { get; set; }
        public int Stype { get; set; }
    }
    public class EteproductImages
    {
        public decimal ProductId { get; set; }
    public decimal PSId { get; set; }
        public int SellerId { get; set; }
        public int srno { get; set; }
        public System.Data.Linq.Binary Images { get; set; }
        public string ImageUrls { get; set; }
        public string ImagePath { get; set; }
        public string Descr { get; set; }
    }
    public class CustomImage
    {
        public string iUrl { get; set; }
        public int Id { get; set; }
        public string Title { get; set; }
    }
    public class EteproductSeller
    {
        public decimal ProductId { get; set; }
        public int SellerId { get; set; }
        public string SellerDescription { get; set; }
        public decimal Price { get; set; }
        public int CurrencyId { get; set; }
        public string AnalysisCode { get; set; }
        public System.DateTime CreatedOn { get; set; }
        public System.DateTime UpdatedOn { get; set; }
        public int UserId { get; set; }
        public int PrId { get; set; }
        public int MinOrder { get; set; }
        public int DelTlme { get; set; }
        public int PriceUnits { get; set; }
        public int PriceUomId { get; set; }
        public int PricePackId { get; set; }
        public string KeyWords { get; set; }
        public int MinOrderUom { get; set; }
        public int DelTimeType { get; set; }
        public decimal PSId { get; set; }
        public decimal LinkPSId { get; set; }
        public string ProductDescription { get; set; }
        public string BrandName { get; set; }
        public string MaterialCare { get; set; }
        public string MaterialType { get; set; }
        public int DiscountLink { get; set; }
    }

```



```

    public string Manufacturer { get; set; }
    public string Length { get; set; }
    public int MadeinCountry { get; set; }
    public string Location { get; set; }
    public string GeoLocation { get; set; }
}
public class Eteproductbuyers
{
    public decimal ProductId { get; set; }
    public int SellerId { get; set; }
    public string SellerDescription { get; set; }
    public decimal Price { get; set; }
    public string Currency { get; set; }
    public System.DateTime UpdatedOn { get; set; }
    public int MinOrder { get; set; }
    public int DelTIme { get; set; }
    public int PriceUnits { get; set; }
    public string PriceUom { get; set; }
    public string PricePack { get; set; }
    public string KeyWords { get; set; }
    public string MinOrderUom { get; set; }
    public string DelTimeType { get; set; }
    public decimal PSId { get; set; }
    public string ImageURL { get; set; }
    public string SellerName { get; set; }
    public string ProductName { get; set; }
    public string RedirctString { get; set; }
}
public class EteSellerProduct
{
    public decimal ProductId { get; set; }
    public int SellerId { get; set; }
    public string ProductDesc { get; set; }
    public int pcatid { get; set; }
    public int pgid { get; set; }
}
public class EteSellerColors
{
    public int ColorId { get; set; }
    public int SellerId { get; set; }
    public string ColorDesc { get; set; }
    public System.Data.Linq.Binary Images { get; set; }
    public string ImagePath { get; set; }
    public string ImageURL { get; set; }
    public string ImageThumb { get; set; }
    public bool active { get; set; }
    public bool usercolactive { get; set; }
    public DateTime CreatedOn { get; set; }
}

```

```

}
public class EteSellerSizes
{
    public int SellerId { get; set; }
    public int EpId { get; set; }
    public int EpOptId { get; set; }
    public string EpOptDescr { get; set; }
    public DateTime CreatedOn { get; set; }
}
public class EteSelleerColorSizes
{
    public int userid { get; set; }
    public int ColorId { get; set; }
    public string ColorDescr { get; set; }
    public System.Data.Linq.Binary Images { get; set; }
    public string ImagePath { get; set; }
    public string ImageURL { get; set; }
    public string ImageThumb { get; set; }
    public string EpDescription { get; set; }
    public int epid { get; set; }
    public int EpOptId { get; set; }
    public string EpOPtDescr { get; set; }
    public int Stock { get; set; }
}
public class EteproductSellerStocks
{
    public decimal PSId { get; set; }
public int EpId{ get; set; }
public int EpOptId{ get; set; }
public int colid{ get; set; }
public int Stock{ get; set; }
public string Remark{ get; set; }
public bool active { get; set; }
public decimal StkPrice { get; set; }
}
public class EteproductSellerServices
{
    public decimal PSId { get; set; }
    public decimal productid { get; set; }
    public string Remark { get; set; }
    public decimal PrdPrice { get; set; }
    public int PrdUnit { get; set; }
    public bool active { get; set; }
    public int userid { get; set; }
}
}
using System;
using System.Collections.Generic;

```

```

using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace DowBEntity
{
    public class EteProductGroup
    {
        public int pgid { get; set; }
        public int PCatId { get; set; }
        public string pgroupname { get; set; }
        public string pgroupdescription { get; set; }
        public bool isActive { get; set; }
        public System.Nullable<System.DateTime> createdOn { get; set; }
        public string lbtnimage { get; set; }
        public int UserId { get; set; }
        public string Catname { get; set; }
        public int PNos { get; set; }
        public string ImageUrl { get; set; }
        public decimal ProductId { get; set; }
        public string ProductName { get; set; }
        public string ShortCode { get; set; }
        public int ProductGroupId { get; set; }
        public int ProductGroupCategory { get; set; }
        public string Type { get; set; }
        public decimal Price { get; set; }
        public int PackageType { get; set; }
        public int PageId { get; set; }
        public int UId { get; set; }
        public int CurrencyId { get; set; }
        public string AnalysisCode { get; set; }
        public DateTime CreatedOn { get; set; }
        public DateTime UpdatedOn { get; set; }
        public int PrId { get; set; }
        public int MinOrder { get; set; }
        public int DelTime { get; set; }
        public bool Check { get; set; }
        public string CatName { get; set; }
        public string GroupName { get; set; }
        public string PrType { get; set; }
        public int SellerId { get; set; }
        public int PriceUnits { get; set; }
        public int PriceUomId { get; set; }
        public int PricePackId { get; set; }
        public int MinOrderUom { get; set; }
        public int DelTimeType { get; set; }
        public int Stype { get; set; }
    }
    public class Etetblpacking

```

```

{
    public int PackID { get; set; }
    public string PackShort { get; set; }
    public string PackDescription { get; set; }
    public System.DateTime CreatedOn { get; set; }
    public System.DateTime UpdatedOn { get; set; }
    public int UserId { get; set; }
}
public class Etetbluom
{
    public int UomId { get; set; }
    public string UomShort { get; set; }
    public string UomDescription { get; set; }
    public System.DateTime CreatedOn { get; set; }
    public System.DateTime UpdatedOn { get; set; }
    public int UserId { get; set; }
}
public class EteCurrency
{
    public int CId { get; set; }
    public string CShortCode { get; set; }
    public string CDescription { get; set; }
    public System.Nullable<System.DateTime> createdon { get; set; }
    public System.Nullable<System.DateTime> updatedon { get; set; }
    public System.Nullable<int> userid { get; set; }
}
public class Etetbldeltime
{
    public int DelTimeType { get; set; }
    public string DelTimeDescription { get; set; }
    public System.DateTime CreatedOn { get; set; }
    public System.DateTime UpdatedOn { get; set; }
    public int UserId { get; set; }
}
public class EteUserShortList{
    public int UserId { get; set; }
    public int PSId { get; set; }
    public System.DateTime CreatedOn { get; set; }
    public int UId { get; set; }
    public bool Active { get; set; }
}
public class EteUserCart
{
    public int UCId { get; set; }
    public int UserId { get; set; }
    public int PSId { get; set; }
    public System.DateTime CreatedOn { get; set; }
    public int UId { get; set; }
}

```

```

    public bool Active { get; set; }
    public int PayId { get; set; }
    public System.DateTime PayDate { get; set; }
    public int DiscountLink { get; set; }
}
public class EteMyUserCart
{
    public decimal ProductId{get;set;}
    public string ProductName{get;set;}
    public int UserId{get;set;}
    public string SellerDescription{get;set;}
    public decimal Price{get;set;}
    public int CurrencyId{get;set;}
    public int PriceUnits{get;set;}
    public int PriceUomId{get;set;}
    public int PricePackId{get;set;}
    public System.DateTime createdon{get;set;}
    public int UCId{get;set;}
    public int Qty { get; set; }
    public int PayId{get;set;}
    public string ShCurrency { get; set; }
    public decimal Amt { get; set; }
    public decimal PSId { get; set; }
    public string ImagePath { get; set; }
    public int DiscountLink { get; set; }
}
public class ProductProductSeller
{
    public decimal ProductId { get; set; }
    public string ProductName { get; set; }
    public int ProductGroupId { get; set; }
    public decimal PSId { get; set; }
    public decimal ProductId1 { get; set; }
    public int SellerId { get; set; }
    public string SellerDescription { get; set; }
    public decimal Price { get; set; }
    public int CurrencyId { get; set; }
    public int PriceUnits { get; set; }
    public int PriceUomId { get; set; }
    public int PricePackId { get; set; }
    public string KeyWords { get; set; }
    public System.DateTime CreatedOn { get; set; }
    public System.DateTime UpdatedOn { get; set; }
    public int UserId { get; set; }
    public int PrId { get; set; }
    public int MinOrder { get; set; }
    public int MinOrderUom { get; set; }
    public int DelTIme { get; set; }
}

```

```

    public int DelTimeType { get; set; }
    public string CurPrice { get; set; }
    public string QtyUOM { get; set; }
    public string DTTImeType { get; set; }
    public string SellerName { get; set; }
    public string ImageURL { get; set; }
    public string PricePack { get; set; }
    public string CatName { get; set; }
    public string GroupName { get; set; }
    public string RedirctString { get; set; }
    public string ImagePath { get; set; }
    public System.Data.Linq.Binary Images { get; set; }
    public string ImageThumb { get; set; }
    public decimal LinkPSId { get; set; }
    public string ProductDescription { get; set; }
    public int DiscountLink { get; set; }
    public string BrandName { get; set; }
    public string MaterialCare { get; set; }
    public string MaterialType { get; set; }
    public string Manufacturer { get; set; }
    public string Length { get; set; }
    public int MadeinCountry { get; set; }
    public string sizeavl { get; set; }
    public int TStock { get; set; }
}
}
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace DowBEntity
{
    public class EteRecentPlaces
    {
        public int rp_id { get; set; }
        public string rp_type { get; set; }
        public string rp_heading { get; set; }
        public string rp_url { get; set; }
        public System.Nullable<System.DateTime> rp_date { get; set; }
        public System.Nullable<int> rp_user { get; set; }
    }
}
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

```

```

namespace DowBEntity
{
    public class EteRole
    {
        public int Id { get; set; }
        public string Name { get; set; }
        public bool IsActive { get; set; }
        public System.DateTime CreatedOn { get; set; }
        public int PageId { get; set; }
        public int UId { get; set; }
    }
}
using System ;
using System.Collections.Generic ;
using System.Linq ;
using System.Text ;
using System.Threading.Tasks;
namespace DowBEntity
{
    public class EteScratchpad
    {
        public int Id {get; set;}
        public System.DateTime CreatedOn {get; set;}
        public int UserId {get; set;}
        public int CallingPageId {get; set;}
        public int SubjectId {get; set;}
        public string Content {get; set;}
        public string UserName {get; set;}
        public string CallingPageName {get; set;}
        public int PageId { get; set; }
        public int UId { get; set; }
    }
}
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace DowBEntity
{
    public class EteStaffHoliday
    {
        public int Id {get; set;}
        public System.DateTime CreatedOn {get; set;}
        public int BMSUserId {get; set;}
        public string UserName {get; set;}
        public System.DateTime StartDate {get; set;}
        public System.DateTime EndDate {get; set;}
    }
}

```

```

public decimal ActualDays {get; set;}
public System.DateTime RequestedDate {get; set;}
public int Status {get; set;}
public System.Nullable<System.DateTime> DecisionDate {get; set;}
public System.Nullable<int> DecisionByld {get; set;}
    public string DecisionUserName    {get; set;}
public string Description {get; set;}
public string RejectionReason {get; set;}
public System.Nullable<decimal> BalanceBefore {get; set;}
public System.Nullable<decimal> BalanceAfter {get; set;}
public int PageId { get; set; }
public int Uld { get; set; }
public string StatusName { get; set; }
    }
}
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace DowBEntity
{
    public class EtetblGrpMapping
    {
        public int MapId { get; set; }
        public System.Nullable<int> GroupId { get; set; }
        public System.Nullable<int> PageId { get; set; }
        public System.Nullable<bool> view { get; set; }
        public System.Nullable<bool> add { get; set; }
        public System.Nullable<bool> Edit { get; set; }
        public System.Nullable<bool> Delete { get; set; }
        public System.Nullable<System.DateTime> CreatedOn { get; set; }
        public System.Nullable<int> PId { get; set; }
        public int Uld { get; set; }
    }
}
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace DowBEntity
{
    public class EtetblLeftButtons
    {
        public int lbtnid { get; set; }
        public string lbtnName { get; set; }
        public string lbtnImage { get; set; }
    }
}

```



```

    public string lbtnUrl { get; set; }
    public string lbtnCode { get; set; }
    public System.Nullable<int> lbtnPriority { get; set; }
    public System.Nullable<int> lbtnLevel { get; set; }
    public System.Nullable<int> isactive { get; set; }
    public System.Nullable<int> lbtnType { get; set; }
    public int PageId { get; set; }
    public int UId { get; set; }
}
}
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace DowBEntity
{
    public class EteTblLogin
    {
        public int EteLogInId { get; set; }
        public System.Nullable<System.DateTime> LogInTime { get; set; }
        public System.Nullable<System.DateTime> LogOutTime { get; set; }
        public System.Nullable<int> UserId { get; set; }
        public System.Nullable<bool> LoginStatus { get; set; }
        public string SourceIp { get; set; }
        public string MachineName { get; set; }
        public string Description { get; set; }
        public string City { get; set; }
        public string Country { get; set; }
        public int PageId { get; set; }
        public int UId { get; set; }
    }
}
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace DowBEntity
{
    public class EteTblPageButtonMapping
    {
        public int pagebtnid { get; set; }
        public System.Nullable<int> pageid { get; set; }
        public int lbtnid { get; set; }
        public int show { get; set; }
        public System.Nullable<int> priority { get; set; }
        public string lbtnName { get; set; }
    }
}

```

```

        public System.Nullable<int> groupid { get; set; }
        public System.Nullable<int> mapid { get; set; }
        private int _mapid { get; set; }
        public int PageId { get; set; }
        public int UId { get; set; }
        public int urlpageid { get; set; }
        public int maptype { get; set; }
        public string mapdescription { get; set; }
        public int isactive { get; set; }
    }
}
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace DowBEntity
{
    public class EtetblPages
    {
        public int PageId { get; set; }
        public string PageName { get; set; }
        public string PageUrl { get; set; }
        public string PageCode { get; set; }
        public System.Nullable<int> ParentId { get; set; }
        public System.Nullable<int> Priority { get; set; }
        public System.Nullable<int> level { get; set; }
        public System.Nullable<int> isMenuItem { get; set; }
        public System.Nullable<int> Pgid { get; set; }
        public int UId { get; set; }
    }
    public class EteMenuList
    {
        public int pageid { get; set; }
        public string pagename { get; set; }
        public string pageurl { get; set; }
        public int parentid { get; set; }
        public int priority { get; set; }
        public int level { get; set; }
        public int ismenuitem { get; set; }
        public int active { get; set; }
        public string imagepath { get; set; }
        public int Stype { get; set; }
        public System.Nullable<int> Mlevel { get; set; }
        public string BtnImg { get; set; }
    }
}
using System;

```

```

using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace DowBEntity
{
    public class EtetblPermissionGroup
    {
        public int GroupId { get; set; }
        public string GroupName { get; set; }
        public System.Nullable<bool> Status { get; set; }
        public System.Nullable<int> UserId { get; set; }
        public System.Nullable<int> RoleId { get; set; }
        public int PageId { get; set; }
        public int UId { get; set; }
    }
}
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace DowBEntity
{
    public class EtetblUser
    {
        public int UserId { get; set; }
        public string UserName { get; set; }
        public string Password { get; set; }
        public string FirstName { get; set; }
        public string LastName { get; set; }
        public string EmailID { get; set; }
        public string ContactNum { get; set; }
        public System.Nullable<System.DateTime> CreatedOn { get; set; }
        public System.Nullable<bool> Status { get; set; }
        public System.Nullable<int> RoleID { get; set; }
        public System.Nullable<int> GroupId { get; set; }
        public bool IsRemoved { get; set; }
        public string Department { get; set; }
        public string Region { get; set; }
        public System.Nullable<System.DateTime> UpdatedOn { get; set; }
        public int DealerId { get; set; }
        public string DealerName { get; set; }
        public int PageId { get; set; }
        public int UId { get; set; }
        public string GroupName { get; set; }
        public string ImageURL { get; set; }
    }
}

```

```

}
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace DowBEntity
{
    public class Etebluser_profile
    {
        public int userid { get; set; }
        public string Business_Type { get; set; }
        public string Legal_Structure { get; set; }
        public string Year_Established { get; set; }
        public string Number_of_Employees { get; set; }
        public string Company_Description { get; set; }
        public string Company_Website_URL { get; set; }
        public string Turnover_per_Year { get; set; }
        public int CID { get; set; }
        public string Quality_Assurance_Accreditation { get; set; }
        public string Registration_Number { get; set; }
        public string D_U_N_S_Number { get; set; }
        public string Specializing_in { get; set; }
        public string Monthly_Production_Capacity { get; set; }
        public int UOMId { get; set; }
        public System.Data.Linq.Binary Company_logo { get; set; }
        public System.DateTime CreatedOn { get; set; }
        public System.DateTime UpdatedOn { get; set; }
        public int uid { get; set; }
        public int Pageld { get; set; }
    }
}
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace DowBEntity
{
    public class EteUserPrivilege
    {
        public int UserId { get; set; }
        public int Privilegeld { get; set; }
        public System.DateTime CreatedOn { get; set; }
        public int BMSUser { get; set; }
        public int Pageld { get; set; }
        public int UId { get; set; }
    }
}

```

```

public class EteUserRole
{
    public int UserId { get; set; }
    public int RoleId { get; set; }
    public System.DateTime CreatedOn { get; set; }
    public int BMSUser { get; set; }
    public int Role { get; set; }
    public int PageId { get; set; }
    public int UId { get; set; }
}
}
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace DowBEntity
{
    public class Etev_tblGrpMapping
    {
        public int MapId { get; set; }
        public System.Nullable<int> GroupId { get; set; }
        public string GroupName { get; set; }
        public System.Nullable<int> PageId { get; set; }
        public string PageName { get; set; }
        public System.Nullable<bool> view { get; set; }
        public System.Nullable<bool> add { get; set; }
        public System.Nullable<bool> Edit { get; set; }
        public System.Nullable<bool> Delete { get; set; }
        public System.Nullable<System.DateTime> CreatedOn { get; set; }
        public System.Nullable<int> ParentId { get; set; }
        public System.Nullable<int> level { get; set; }
        public System.Nullable<int> Pgid { get; set; }
        public int UId { get; set; }
    }
}
}
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace DowBEntity
{
    public class EteVatCode
    {
        public int Id { get; set; }
        public string Code { get; set; }
        public decimal Rate { get; set; }
    }
}

```

```

        public string Description { get; set; }
        public bool IsDefault { get; set; }
        public bool IsActive { get; set; }
        public System.DateTime CreatedOn { get; set; }
        public int UId { get; set; }
        public int Pageid { get; set; }
    }
}
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace DowBEntity
{
    public class FilterEnum
    {
        public int Brand { get; set; }
        public int CustomerType { get; set; }
        public int DateRange { get; set; }
        public int Dealer { get; set; }
        public int OrderStatus { get; set; }
        public int OrderHold { get; set; }
    }
}
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace DowBEntity
{
    public class FilterTaskEnum
    {
        public string OrderStatus { get; set; }
        public int Queue { get; set; }
        public int Role { get; set; }
        public int User { get; set; }
    }
}
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace DowBEntity

```

```
{
class NewTask
{
    public int Id
    {
        set { Id = value; }
        get { return Id; }
    }
    public DateTime CreatedOn
    {
        set { CreatedOn = value; }
        get { return CreatedOn; }
    }
    public int QueueId
    {
        set { QueueId = value; }
        get { return QueueId; }
    }
    public int OwnedById
    {
        set { OwnedById = value; }
        get { return OwnedById; }
    }
    public int RaisedById
    {
        set { RaisedById = value; }
        get { return RaisedById; }
    }
    public int ActionedById
    {
        set { ActionedById = value; }
        get { return ActionedById; }
    }
    public DateTime RaisedDate
    {
        set { RaisedDate = value; }
        get { return RaisedDate; }
    }
    public DateTime ActionedDate
    {
        set { ActionedDate = value; }
        get { return ActionedDate; }
    }
    public char Status
    {
        set { Status = value; }
        get { return Status; }
    }
}
```

```
public string Summary
{
    set { Summary = value; }
    get { return Summary; }
}
public string Details
{
    set { Details = value; }
    get { return Details; }
}
public int SubjectId
{
    set { SubjectId = value; }
    get { return SubjectId; }
}
public string TaskData
{
    set { TaskData = value; }
    get { return TaskData; }
}
public DateTime DueDate
{
    set { DueDate = value; }
    get { return DueDate; }
}
public char TaskType
{
    set { TaskType = value; }
    get { return TaskType; }
}
public Boolean TaskOutcome
{
    set { TaskOutcome = value; }
    get { return TaskOutcome; }
}
public string OutcomeNotes
{
    set { OutcomeNotes = value; }
    get { return OutcomeNotes; }
}
public char CLI
{
    set { CLI = value; }
    get { return CLI; }
}
public int PageId { get; set; }
public int UId { get; set; }
}
```



```

}
using System;
using System.Diagnostics;
using System.Text;
using System.IO;
using System.Web;
using System.Threading;
using System.Collections.Generic;
using System.Configuration;
namespace Codaxy.WkHtmlToPdf
{
    public class PdfConvertException : Exception
    {
        public PdfConvertException(String msg) : base(msg) { }
    }
    public class PdfConvertTimeoutException : PdfConvertException
    {
        public PdfConvertTimeoutException() : base("HTML to PDF conversion process has not finished in the")
        {
        }
    }
    public class PdfOutput
    {
        public String OutputFilePath { get; set; }
        public Stream OutputStream { get; set; }
        public Action<PdfDocument, byte[]> OutputCallback { get; set; }
    }
    public class PdfDocument
    {
        public String Url { get; set; }
        public String Html { get; set; }
        public String HeaderUrl { get; set; }
        public String FooterUrl { get; set; }
        public String HeaderLeft { get; set; }
        public String HeaderCenter { get; set; }
        public String HeaderRight { get; set; }
        public String FooterLeft { get; set; }
        public String FooterCenter { get; set; }
        public String FooterRight { get; set; }
        public object State { get; set; }
        public Dictionary<String, String> Cookies { get; set; }
        public Dictionary<String, String> ExtraParams { get; set; }
        public String HeaderFontSize { get; set; }
        public String FooterFontSize { get; set; }
        public String HeaderFontName { get; set; }
        public String FooterFontName { get; set; }
    }
    public class PdfConvertEnvironment
    {
        public String TempFolderPath { get; set; }
    }
}

```

```

    public String WkHtmlToPdfPath { get; set; }
    public int Timeout { get; set; }
    public bool Debug { get; set; }
}
public class PdfConvert
{
    static PdfConvertEnvironment _e;
    public static PdfConvertEnvironment Environment
    {
        get
        {
            if (_e == null)
                _e = new PdfConvertEnvironment
                {
                    TempFolderPath = Path.GetTempPath(),
                    WkHtmlToPdfPath = GetWkhtmlToPdfExeLocation(),
                    Timeout = 60000
                };
            return _e;
        }
    }
    private static string GetWkhtmlToPdfExeLocation()
    {
        string filePath, customPath = ConfigurationManager.AppSettings["wkhtmltopdf:path"];
        if (customPath != null)
        {
            filePath = Path.Combine(customPath, @"wkhtmltopdf.exe");
            if (File.Exists(filePath))
                return filePath;
        }
        string programFilesPath = System.Environment.GetEnvironmentVariable("ProgramFiles");
        filePath = Path.Combine(programFilesPath, @"wkhtmltopdf\wkhtmltopdf.exe");
        if (File.Exists(filePath))
            return filePath;
        string programFilesx86Path = System.Environment.GetEnvironmentVariable("ProgramFiles(x86)");
        filePath = Path.Combine(programFilesx86Path, @"wkhtmltopdf\wkhtmltopdf.exe");
        if (File.Exists(filePath))
            return filePath;
        filePath = Path.Combine(programFilesPath, @"wkhtmltopdf\bin\wkhtmltopdf.exe");
        if (File.Exists(filePath))
            return filePath;
        return Path.Combine(programFilesx86Path, @"wkhtmltopdf\bin\wkhtmltopdf.exe");
    }
    public static void ConvertHtmlToPdf(PdfDocument document, PdfOutput output)
    {
        ConvertHtmlToPdf(document, null, output);
    }
    public static void ConvertHtmlToPdf(PdfDocument document, PdfConvertEnvironment environment, PdfOutput output)

```

```

{
    if (environment == null)
        environment = Environment;
    if (document.Html != null)
        document.Url = "-";
    String outputPdfFilePath;
    bool delete;
    if (woutput.OutputFilePath != null)
    {
        outputPdfFilePath = woutput.OutputFilePath;
        delete = false;
    }
    else
    {
        outputPdfFilePath = Path.Combine(environment.TempFolderPath, String.Format("{0}.pdf", Guid.I
        delete = true;
    }
    if (!File.Exists(environment.WkHtmlToPdfPath))
        throw new PdfConvertException(String.Format("File '{0}' not found. Check if wkhtmltopdf applicat
environment.WkHtmlToPdfPath));
    StringBuilder paramsBuilder = new StringBuilder();
    paramsBuilder.Append("--page-size A4 ");
    if (!string.IsNullOrEmpty(document.HeaderUrl))
    {
        paramsBuilder.AppendFormat("--header-html {0} ", document.HeaderUrl);
        paramsBuilder.Append("--margin-top 25 ");
        paramsBuilder.Append("--header-spacing 5 ");
    }
    if (!string.IsNullOrEmpty(document.FooterUrl))
    {
        paramsBuilder.AppendFormat("--footer-html {0} ", document.FooterUrl);
        paramsBuilder.Append("--margin-bottom 25 ");
        paramsBuilder.Append("--footer-spacing 5 ");
    }
    if (!string.IsNullOrEmpty(document.HeaderLeft))
        paramsBuilder.AppendFormat("--header-left \"{0}\" ", document.HeaderLeft);
    if (!string.IsNullOrEmpty(document.HeaderCenter))
        paramsBuilder.AppendFormat("--header-center \"{0}\" ", document.HeaderCenter);
    if (!string.IsNullOrEmpty(document.HeaderRight))
        paramsBuilder.AppendFormat("--header-right \"{0}\" ", document.HeaderRight);
    if (!string.IsNullOrEmpty(document.FooterLeft))
        paramsBuilder.AppendFormat("--footer-left \"{0}\" ", document.FooterLeft);
    if (!string.IsNullOrEmpty(document.FooterCenter))
        paramsBuilder.AppendFormat("--footer-center \"{0}\" ", document.FooterCenter);
    if (!string.IsNullOrEmpty(document.FooterRight))
        paramsBuilder.AppendFormat("--footer-right \"{0}\" ", document.FooterRight);
    if (!string.IsNullOrEmpty(document.HeaderFontSize))
        paramsBuilder.AppendFormat("--header-font-size \"{0}\" ", document.HeaderFontSize);

```

```

if (!string.IsNullOrEmpty(document.FooterFontSize))
    paramsBuilder.AppendFormat("--footer-font-size \"{0}\" ", document.FooterFontSize);
if (!string.IsNullOrEmpty(document.HeaderFontName))
    paramsBuilder.AppendFormat("--header-font-name \"{0}\" ", document.HeaderFontName);
if (!string.IsNullOrEmpty(document.FooterFontName))
    paramsBuilder.AppendFormat("--footer-font-name \"{0}\" ", document.FooterFontName);
if (document.ExtraParams != null)
    foreach (var extraParam in document.ExtraParams)
        paramsBuilder.AppendFormat("--{0} {1} ", extraParam.Key, extraParam.Value);
if (document.Cookies != null)
    foreach (var cookie in document.Cookies)
        paramsBuilder.AppendFormat("--cookie {0} {1} ", cookie.Key, cookie.Value);
paramsBuilder.AppendFormat("\"{0}\" \"{1}\"", document.Url, outputPdfFilePath);
try
{
    StringBuilder output = new StringBuilder();
    StringBuilder error = new StringBuilder();
    using (Process process = new Process())
    {
        process.StartInfo.FileName = environment.WkHtmlToPdfPath;
        process.StartInfo.Arguments = paramsBuilder.ToString();
        process.StartInfo.UseShellExecute = false;
        process.StartInfo.RedirectStandardOutput = true;
        process.StartInfo.RedirectStandardError = true;
        process.StartInfo.RedirectStandardInput = true;
        using (AutoResetEvent outputWaitHandle = new AutoResetEvent(false))
        using (AutoResetEvent errorWaitHandle = new AutoResetEvent(false))
        {
            DataReceivedEventHandler outputHandler = (sender, e) =>
            {
                if (e.Data == null)
                {
                    outputWaitHandle.Set();
                }
                else
                {
                    output.AppendLine(e.Data);
                }
            };
            DataReceivedEventHandler errorHandler = (sender, e) =>
            {
                if (e.Data == null)
                {
                    errorWaitHandle.Set();
                }
                else
                {
                    error.AppendLine(e.Data);
                }
            };
        }
    }
}

```

```

    }
};
process.OutputDataReceived += outputHandler;
process.ErrorDataReceived += errorHandler;
try
{
    process.Start();
    process.BeginOutputReadLine();
    process.BeginErrorReadLine();
    if (document.Html != null)
    {
        using (var stream = process.StandardInput)
        {
            byte[] buffer = Encoding.UTF8.GetBytes(document.Html);
            stream.BaseStream.Write(buffer, 0, buffer.Length);
            stream.WriteLine();
        }
    }
    if (process.WaitForExit(environment.Timeout) && outputWaitHandle.WaitOne(environment
errorWaitHandle.WaitOne(environment.Timeout))
    {
        if (process.ExitCode != 0 && !File.Exists(outputPdfFilePath))
        {
            throw new PdfConvertException(String.Format("Html to PDF conversion of '{0}' failed
error));
        }
    }
    else
    {
        if (!process.HasExited)
            process.Kill();
        throw new PdfConvertTimeoutException();
    }
}
finally
{
    process.OutputDataReceived -= outputHandler;
    process.ErrorDataReceived -= errorHandler;
}
}
}
if (woutput.OutputStream != null)
{
    using (Stream fs = new FileStream(outputPdfFilePath, FileMode.Open))
    {
        byte[] buffer = new byte[32 * 1024];
        int read;
        while ((read = fs.Read(buffer, 0, buffer.Length)) > 0)

```

```

        woutput.OutputStream.Write(buffer, 0, read);
    }
}
if (woutput.OutputCallback != null)
{
    byte[] pdfFileBytes = File.ReadAllBytes(outputPdfFilePath);
    woutput.OutputCallback(document, pdfFileBytes);
}
}
finally
{
    if (delete && File.Exists(outputPdfFilePath))
        File.Delete(outputPdfFilePath);
}
}
internal static void ConvertHtmlToPdf(string url, string outputFilePath)
{
    ConvertHtmlToPdf(new PdfDocument { Url = url }, new PdfOutput { OutputFilePath = outputFilePath }
}
}
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
namespace DowBEntity
{
    public class PermissionsGroup
    {
        public string mpageid { get; set; }
        public string status { get; set; }
        public string buttonid { get; set; }
    }
}
}

```

Chapter 4

Data tier:

The data tier consists of connection properties to the database and Entity created .dbml file. This can be used to add to the project by adding: add item, visual C# items, data, LINQ to SQL classes

Give a name and add

After opening the file you can drag and drop tables, views and procedures from the SQL server

The following .cs files are used for connectivity:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using DowDAL;
using System.Data;
using System.Data.SqlClient;
using System.Configuration;
using DowBEntity;

namespace DowDAL
{
    public class DALUser
    {
        public string DowConectingString =
        ConfigurationSettings.AppSettings["DowConectingString"].ToString();

        #region Get AllPermissionGroups
        public DataTable GetAllPermissionGroups()
        {
            DataTable dtGetAllPermissionGroups = null;
            DataSet ds = new DataSet();
            try
            {
                ds = SqlHelper.ExecuteDataset(DowConectingString,
                CommandType.StoredProcedure, "GetAllPermissionGroups");
                dtGetAllPermissionGroups = ds.Tables[0];
            }
            catch (Exception ex)
            {
                throw ex;
            }
            return dtGetAllPermissionGroups;
        }
        #endregion Get AllPermissionGroups

        #region User Specs
        public DataTable Login(string uname, string pwd, string MachineName, string
        IpAddress, string City, string Country)
        {
```

```

SqlParameter[] parameters = new SqlParameter[]
{
    new SqlParameter("@UserName", uname),
    new SqlParameter("@Password", pwd),
    new SqlParameter("@IpAddress", IpAddress),
    new SqlParameter("@City", City),
    new SqlParameter("@Country", Country),

    new SqlParameter("@rtn", SqlDbType.VarChar, 100) {Direction =
ParameterDirection.Output}
};
DataTable dtresultset = null;
DataSet ds = new DataSet();
ds = SqlHelper.ExecuteDataset(DowConectingString,
CommandType.StoredProcedure, "spCheckLogin", parameters);
dtresultset = ds.Tables[0];
return dtresultset;
}
public DataTable LoginGuest(string uname, string pwd, string MachineName, string
IpAddress, string City, string Country)
{

    SqlParameter[] parameters = new SqlParameter[]
    {
        new SqlParameter("@UserName", uname),
        new SqlParameter("@Password", pwd),
        new SqlParameter("@IpAddress", IpAddress),
        new SqlParameter("@City", City),
        new SqlParameter("@Country", Country),

        new SqlParameter("@rtn", SqlDbType.VarChar, 100) {Direction =
ParameterDirection.Output}
    };
    DataTable dtresultset = null;
    DataSet ds = new DataSet();
    ds = SqlHelper.ExecuteDataset(DowConectingString,
CommandType.StoredProcedure, "spCheckGuest", parameters);
    dtresultset = ds.Tables[0];
    return dtresultset;
}

public DataTable LoginDet(string uname, string pwd, string MachineName, string
IpAddress, string City, string Country)
{
    SqlParameter[] parameters = new SqlParameter[]
    {
        new SqlParameter("@UserName", uname),
        new SqlParameter("@Password", pwd),
        new SqlParameter("@IpAddress", IpAddress),
        new SqlParameter("@City", City),
        new SqlParameter("@Country", Country),

        new SqlParameter("@rtn", SqlDbType.VarChar, 100) {Direction =
ParameterDirection.Output}
    };
}

```



```

        DataTable dtresultset = null;
        DataSet ds = new DataSet();
        ds = SqlHelper.ExecuteDataset(DowConectingString,
CommandType.StoredProcedure, "spCheckLoginDet", parameters);
        dtresultset = ds.Tables[0];
        return dtresultset;
    }
    public bool Logout(int LoginId, int UserId)
    {
        SqlParameter[] parameters = new SqlParameter[]
        {
            new SqlParameter("@UserId", LoginId),
            new SqlParameter("@LogInId", UserId),
        };

        int count = SqlHelper.ExecuteNonQuery(DowConectingString,
CommandType.StoredProcedure, "spLogout", parameters);
        if (count > 0)
            return true;
        else
            return false;
    }

    public string CreateUser(CreateUser crtUser, bool status, int Role, int GroupId, int
UserId, int loginId)
    {
        SqlParameter[] parameters = new SqlParameter[]
        {
            new SqlParameter("@UserName", crtUser.Uname),
            new SqlParameter("@Password", crtUser.Password),
            new SqlParameter("@FirstName", crtUser.FirstName),

            new SqlParameter("@EmailId", crtUser.EmailID),
            new SqlParameter("@Contactnum", crtUser.ContactNum),
            new SqlParameter("@deptment", crtUser.Department),
            new SqlParameter("@region", crtUser.Region),

            new SqlParameter("@GroupId", GroupId),
            new SqlParameter("@UserId", UserId),
            new SqlParameter("@LoginId", loginId),
            new SqlParameter("@rtn", SqlDbType.VarChar, 100) {Direction =
ParameterDirection.Output}
        };

        int count = SqlHelper.ExecuteNonQuery(DowConectingString,
CommandType.StoredProcedure, "spCreateUser", parameters);
        string rtn_Create_User = (string)parameters[10].Value.ToString();
        return rtn_Create_User;
    }

    public bool UpdateUser(CreateUser crtUser, string NewPwd, int GroupId, int UserId, int
LoginId)
    {
        SqlParameter[] parameters = new SqlParameter[]

```

```

    {
        new SqlParameter("@UserId", UserId),
        new SqlParameter("@LoginId", LoginId),
        new SqlParameter("@NewPassword", crtUser.NewPwd),
        new SqlParameter("@FirstName", crtUser.FirstName),
        new SqlParameter("@EmailId", crtUser.EmailID),
        new SqlParameter("@Contactnum", crtUser.ContactNum),
        new SqlParameter("@GroupId", GroupId),
        new SqlParameter("@deptment", crtUser.Department),
        new SqlParameter("@region", crtUser.Region)
    };

    int count = SqlHelper.ExecuteNonQuery(DowConectingString,
CommandType.StoredProcedure, "spUpdateUser", parameters);
    if (count > 0)
        return true;
    else
        return false;
}

public bool DisableUser(int UserId, bool status)
{
    SqlParameter[] parameters = new SqlParameter[]
    {
        new SqlParameter("@UserId", UserId),
        new SqlParameter("@Status", status)
    };

    int rtn = SqlHelper.ExecuteNonQuery(DowConectingString,
CommandType.StoredProcedure, "spDisableUser", parameters);
    if (rtn > 0)
        return true;
    else
        return false;
}

public bool RemoveUser(int UserId, bool status)
{
    SqlParameter[] parameters = new SqlParameter[]
    {
        new SqlParameter("@UserId", UserId),
        new SqlParameter("@Status", status)
    };

    int rtn = SqlHelper.ExecuteNonQuery(DowConectingString,
CommandType.StoredProcedure, "spRemoveUser", parameters);
    if (rtn > 0)
        return true;
    else
        return false;
}
}
#endregion User Specs

```

```

#region GetSourceIcp details
public DataSet GetSourceIcp(string Srtcol, string SrtExp, int pageno, int pagesize)
{
    SqlParameter[] parameters = new SqlParameter[]
    {
        new SqlParameter("@OrdCol",Srtcol),
        new SqlParameter("@OrdExp",SrtExp),
        new SqlParameter("@pageNo",pageno),
        new SqlParameter("@pageSize",pagesize)
    };
    DataSet dsSourceIcp = null;
    try
    {
        dsSourceIcp = SqlHelper.ExecuteDataset(DowConectingString,
CommandType.StoredProcedure, "spGetSourceIcp", parameters);
    }
    catch (Exception ex)
    {
        throw ex;
    }

    return dsSourceIcp;
}
#endregion GetSourceIcp details

#region User Lock Details
public bool CreateUserLockOut(int BloackingPeriod, int Logonattempts, int
UserLockOutTime, int UserId, int LoginId)
{
    SqlParameter[] parameters = new SqlParameter[]
    {
        new SqlParameter("@BloackingPeriod", BloackingPeriod),
        new SqlParameter("@Logonattempts", Logonattempts),
        new SqlParameter("@UserLockOutTime", UserLockOutTime),
        new SqlParameter("@UserId", UserLockOutTime),
        new SqlParameter("@LoginId", UserLockOutTime),
    };

    int count = SqlHelper.ExecuteNonQuery(DowConectingString,
CommandType.StoredProcedure, "spCreateUserLockOut", parameters);
    if (count > 0)
        return true;
    else
        return false;
}
public DataTable GetUserLoclOutDetails()
{
    DataTable dtGetUserLoclOutDetails = null;
    DataSet ds = new DataSet();
    try
    {
        ds = SqlHelper.ExecuteDataset(DowConectingString,
CommandType.StoredProcedure, "spGetUserLoclOutDetails");
    }
}

```

```

        dtGetUserLoclOutDetails = ds.Tables[0];
    }
    catch (Exception ex)
    {
        throw ex;
    }
    return dtGetUserLoclOutDetails;
}
public DataTable GetUserLoclOutTimeDevice()
{
    DataTable dtGetUserLoclOutDetails = null;
    DataSet ds = new DataSet();
    try
    {
        ds = SqlHelper.ExecuteDataset(DowConectingString,
CommandType.StoredProcedure, "spGetUserLoclOutTimeDevice");
        dtGetUserLoclOutDetails = ds.Tables[0];

    }
    catch (Exception ex)
    {
        throw ex;
    }
    return dtGetUserLoclOutDetails;
}
#endregion User Lock Details

#region userPermissionSets

public DataSet GetAllPermissionsets()
{

    DataSet dsper = null;
    try
    {
        dsper = SqlHelper.ExecuteDataset(DowConectingString, CommandType.Text,
"select Groupid,GroupName from tblpermissionGroup");
    }
    catch (Exception ex)
    {
        throw ex;
    }

    return dsper;
}

public DataSet GetPersets_Grp(int GrpId)
{
    SqlParameter[] parameters = new SqlParameter[]
    {
        new SqlParameter("@GroupId",GrpId),

```

```

};
DataSet dsPer = null;
try
{
    dsPer = SqlHelper.ExecuteDataset(DowConectingString,
CommandType.StoredProcedure, "spGetPerSetsforGrp", parameters);
}
catch (Exception ex)
{
    throw ex;
}

return dsPer;
}

# endregion

public CreateUser GetUser(int UserId)
{
    CreateUser user = new DowBEntity.CreateUser();
    SqlParameter[] parameters = new SqlParameter[]
    {
        new SqlParameter("@UserId",UserId)
    };
    DataSet dsUsers = null;
    try
    {
        dsUsers = SqlHelper.ExecuteDataset(DowConectingString,
CommandType.StoredProcedure, "spGetUserInfo", parameters);
        user.LastName = dsUsers.Tables[0].Rows[0]["LastName"].ToString();
        user.FirstName = dsUsers.Tables[0].Rows[0]["FirstName"].ToString();
        user.EmailID = dsUsers.Tables[0].Rows[0]["EmailId"].ToString();
        user.ContactNum = dsUsers.Tables[0].Rows[0]["ContactNum"].ToString();
        user.Uname = dsUsers.Tables[0].Rows[0]["UserName"].ToString();
        user.Password = dsUsers.Tables[0].Rows[0]["Password"].ToString();
        user.Region = dsUsers.Tables[0].Rows[0]["Region"].ToString();
        user.Department = dsUsers.Tables[0].Rows[0]["Department"].ToString();
        user.GroupId =
Convert.ToInt32(dsUsers.Tables[0].Rows[0]["GroupId"].ToString());
    }
    catch (Exception ex)
    {
        throw ex;
    }

    return user;
}

public DataSet GetUserInfo(string Srtcol, string SrtExp, int pageno, int pagesize)
{
    SqlParameter[] parameters = new SqlParameter[]
    {

```

```

        new SqlParameter("@OrdCol", Srtcol),
        new SqlParameter("@OrdExp", SrtExp),
        new SqlParameter("@pageNo", pageno),
        new SqlParameter("@pageSize", pagesize)

};
DataSet dsUsers = null;
try
{
    dsUsers = SqlHelper.ExecuteDataset(DowConectingString,
CommandType.StoredProcedure, "spGetAllUsers", parameters);
}
catch (Exception ex)
{
    throw ex;
}

return dsUsers;
}

public DataSet GetUserList(string username)
{
    SqlParameter[] commandParameters = new SqlParameter[] { new
SqlParameter("@username", username) };
    DataSet set = null;
    try
    {
        set = SqlHelper.ExecuteDataset(this.DowConectingString,
CommandType.StoredProcedure, "GetUserList", commandParameters);
    }
    catch (Exception exception)
    {
        throw exception;
    }
    return set;
}
}
}

****

using System;
using System.Collections.Generic;
using DowBEntity;
using System.Data;
using System.Data.SqlClient;
using System.Configuration;
using System.Xml;

namespace DowDAL
{
    public class DalPermissionGroups
    {
        public string DowConectingString =
ConfigurationSettings.AppSettings["DowConectingString"].ToString();
        public UserPermissions GetPersets_Page(string UserName, string PageUrl)

```

```

{
    SqlParameter[] parameters = new SqlParameter[]
    {
        new SqlParameter("@UserName", UserName),
        new SqlParameter("@PageUrl", PageUrl),
    };

    DataSet ds = SqlHelper.ExecuteDataset(DowConectingString,
CommandType.StoredProcedure, "spGetPerSetsforPage", parameters);
    UserPermissions up = new UserPermissions();
    if (ds.Tables[0].Rows.Count > 0)
    {
        up.MapId = Convert.ToInt32(ds.Tables[0].Rows[0]["MapId"].ToString());
        up.PageId = Convert.ToInt32(ds.Tables[0].Rows[0]["PageId"].ToString());
        up.PageName = ds.Tables[0].Rows[0]["PageName"].ToString();
        up.PageCode = ds.Tables[0].Rows[0]["PageCode"].ToString();
        up.PageUrl = ds.Tables[0].Rows[0]["PageUrl"].ToString();
        up.View = bool.Parse(ds.Tables[0].Rows[0]["View"].ToString());
        up.Add = bool.Parse(ds.Tables[0].Rows[0]["Add"].ToString());
        up.Edit = bool.Parse(ds.Tables[0].Rows[0]["Edit"].ToString());
        up.Delete = bool.Parse(ds.Tables[0].Rows[0]["Delete"].ToString());
        up.UserName = ds.Tables[0].Rows[0]["username"].ToString();
    }
    return up;
}
public UserPermissions GetUserLandingPage(string UserName)
{
    SqlParameter[] parameters = new SqlParameter[]
    {
        new SqlParameter("@UserName", UserName),
    };
    DataSet ds = SqlHelper.ExecuteDataset(DowConectingString,
CommandType.StoredProcedure, "GetUserLandingPage", parameters);
    UserPermissions up = new UserPermissions();
    if (ds.Tables[0].Rows.Count > 0)
    {
        up.MapId = Convert.ToInt32(ds.Tables[0].Rows[0]["MapId"].ToString());
        up.PageId = Convert.ToInt32(ds.Tables[0].Rows[0]["PageId"].ToString());
        up.PageName = ds.Tables[0].Rows[0]["PageName"].ToString();
        up.PageCode = ds.Tables[0].Rows[0]["PageCode"].ToString();
        up.PageUrl = ds.Tables[0].Rows[0]["PageUrl"].ToString();
        up.View = bool.Parse(ds.Tables[0].Rows[0]["View"].ToString());
        up.Add = bool.Parse(ds.Tables[0].Rows[0]["Add"].ToString());
        up.Edit = bool.Parse(ds.Tables[0].Rows[0]["Edit"].ToString());
        up.Delete = bool.Parse(ds.Tables[0].Rows[0]["Delete"].ToString());
        up.Hir = ds.Tables[0].Rows[0]["concatenador"].ToString();
        up.UserName = UserName;
    }
    return up;
}
public List<UserPermissions> GetPersets_usr(string UserName)
{
    SqlParameter[] parameters = new SqlParameter[]

```

```

    {
        new SqlParameter("@UserName", UserName),
    };

    DataSet ds = SqlHelper.ExecuteDataset(DowConectingString,
    CommandType.StoredProcedure, "GetPageTreeForPage", parameters);
    List<UserPermissions> LstUserPer = new List<UserPermissions>();

    foreach(DataRow dr in ds.Tables[0].Rows)
    {
        UserPermissions up = new UserPermissions();
        up.MapId = Convert.ToInt32(dr["MapId"].ToString());
        up.PagId = Convert.ToInt32(dr["PagId"].ToString());
        up.ParentId = Convert.ToInt32(dr["ParentId"].ToString());
        up.PageName = dr["PageName"].ToString();
        up.PageCode = dr["PageCode"].ToString();
        up.PageUrl = dr["PageUrl"].ToString();
        up.View = bool.Parse(dr["View"].ToString());
        up.Add = bool.Parse(dr["Add"].ToString());
        up.Edit = bool.Parse(dr["Edit"].ToString());
        up.Delete = bool.Parse(dr["Delete"].ToString());
        up.Hir = dr["concatenador"].ToString();
        LstUserPer.Add(up);
    }

    return LstUserPer;
}
public DataTable GetPerSetsForGroup()
{
    SqlParameter[] parameters = new SqlParameter[]
    {
        new SqlParameter("@GroupId", Convert.ToInt32(0)),
    };

    DataSet ds = SqlHelper.ExecuteDataset(DowConectingString,
    CommandType.StoredProcedure, "GetPerSetsForGroup", parameters);
    DataTable LstUserPer = ds.Tables[0];
    return LstUserPer;
}
public DataTable PopulateDynamicTree(int GrpId)
{
    SqlParameter[] parameters = new SqlParameter[]
    {
        new SqlParameter("@GroupId", GrpId),
    };

    DataSet ds = SqlHelper.ExecuteDataset(DowConectingString,
    CommandType.StoredProcedure, "GetPerSetsForGroup", parameters);
    DataTable LstUserPer = ds.Tables[0];
    return LstUserPer;
}

public string CreatePermissionGrp(XmlDocument xdoc,string GrpName,int UserId, int
LoginId)
{

```



```

SqlParameter[] parameters = new SqlParameter[]
{
    new SqlParameter("@GroupName", GrpName),
    new SqlParameter("@xmlstring", xdoc.InnerXml) ,
    new SqlParameter("@UserId", UserId) ,
    new SqlParameter("@LoginId", LoginId) ,
    new SqlParameter("@rtn", SqlDbType.VarChar, 1000) {Direction =
ParameterDirection.Output}

};

int Count = SqlHelper.ExecuteNonQuery(DowConectingString,
CommandType.StoredProcedure, "spCreatePermissionGrp", parameters);
string rtn_status = (string)parameters[4].Value.ToString();
return rtn_status;
}

public bool UpdatePermissionGrp(XmlDocument xdoc, int GrpId, int UserId, int LoginId)
{
    SqlParameter[] parameters = new SqlParameter[]
    {
        new SqlParameter("@GrpId", GrpId),
        new SqlParameter("@UserId", UserId) ,
        new SqlParameter("@LoginId", LoginId) ,
        new SqlParameter("@xmlstring", xdoc.InnerXml)

    };

    int count = SqlHelper.ExecuteNonQuery(DowConectingString,
CommandType.StoredProcedure, "spUpdatePermissionGrp", parameters);
    if (count > 0)
    {
        return true;
    }
    else
    {
        return false;
    }
}

}

}

****

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

using System.Configuration;
using System.Data;
using System.Data.SqlClient;

```

```

namespace DowDAL
{

    public class DalMenus
    {
        public string DowConectingString =
        ConfigurationSettings.AppSettings["DowConectingString"].ToString();

        public DataTable SSP_GET_HIERARCHY()
        {
            DataSet ds = SqlHelper.ExecuteDataset(DowConectingString,
            CommandType.StoredProcedure, "SSP_GET_HIERARCHY");
            DataTable LstUserPer = ds.Tables[0];
            return LstUserPer;
        }

        public DataTable GetMenus(int GroupId)
        {
            SqlParameter[] parameters = new SqlParameter[]
            {
                new SqlParameter("@GroupId", GroupId),
            };

            DataSet ds = SqlHelper.ExecuteDataset(DowConectingString,
            CommandType.StoredProcedure, "GetPageTreeForPage", parameters);
            DataTable LstUserPer = ds.Tables[0];
            return LstUserPer;
        }

        public DataTable GetNewMenus(int GroupId)
        {
            SqlParameter[] parameters = new SqlParameter[]
            {
                new SqlParameter("@GroupId", GroupId),
            };

            DataSet ds = SqlHelper.ExecuteDataset(DowConectingString,
            CommandType.StoredProcedure, "GetPageTreeForMenu", parameters);
            DataTable LstUserPer = ds.Tables[0];
            return LstUserPer;
        }

        public DataTable Permission_Group(int GroupId,int type)
        {
            SqlParameter[] parameters = new SqlParameter[]
            {
                new SqlParameter("@GroupId", GroupId),
                new SqlParameter("@type", type)
            };

            DataSet ds = SqlHelper.ExecuteDataset(DowConectingString,
            CommandType.StoredProcedure, "ManagePermissions", parameters);
            DataTable LstUserPer = ds.Tables[0];
        }
    }
}

```

```

        return LstUserPer;
    }

    //GetPageButtonDetails(int pagePagebtnid)
    public DataTable ProductsList(int Id)
    {
        SqlParameter[] parameters = new SqlParameter[]
        {
            new SqlParameter("@Id", Id),
        };

        DataSet ds = SqlHelper.ExecuteDataset(DowConectingString,
        CommandType.StoredProcedure, "pvproduct", parameters);
        DataTable LstUserPer = ds.Tables[0];
        return LstUserPer;
    }

    public DataTable Products_Group(int GroupId, int type)
    {
        SqlParameter[] parameters = new SqlParameter[]
        {
            new SqlParameter("@GroupId", GroupId),
            new SqlParameter("@type", type)
        };

        DataSet ds = SqlHelper.ExecuteDataset(DowConectingString,
        CommandType.StoredProcedure, "ManageProducts", parameters);
        DataTable LstUserPer = ds.Tables[0];
        return LstUserPer;
    }

    public DataSet GetPageButtonDetails(int pagebtnid)
    {
        SqlParameter[] parameters = new SqlParameter[]
        {
            //@lbtnid int,@loginid int,@lbtntype
            new SqlParameter("@pagebtnid", pagebtnid)
        };

        DataSet ds = SqlHelper.ExecuteDataset(DowConectingString,
        CommandType.StoredProcedure, "GetLeftButtonsbyPagebtnid", parameters);
        return ds;
    }

    public DataSet BindDynamic_DDL(int lbtnid, int loginid, int lbtntype)
    {
        SqlParameter[] parameters = new SqlParameter[]
        {
            //@lbtnid int,@loginid int,@lbtntype
            new SqlParameter("@lbtnid", lbtnid),
            new SqlParameter("@loginid", loginid),
            new SqlParameter("@lbtntype", lbtntype)
        };
    }

```

```

};

DataSet ds = SqlHelper.ExecuteDataset(DowConectingString,
CommandType.StoredProcedure, "GetLeftDDL", parameters);
return ds;

}

public DataSet GetLeftButtonsPermision(int pageid,int groupid)
{
    SqlParameter[] parameters = new SqlParameter[]
    {
        //@lbtnid int,@loginid int,@lbtntype
        new SqlParameter("@pageid", pageid),
        new SqlParameter("@groupid", groupid)
    };

    DataSet ds = SqlHelper.ExecuteDataset(DowConectingString,
CommandType.StoredProcedure, "GetLeftButtonsPermission", parameters);
return ds;

}

public DataSet GetLeftButtonsProducts(int pageid, int groupid)
{
    SqlParameter[] parameters = new SqlParameter[]
    {
        //@lbtnid int,@loginid int,@lbtntype
        new SqlParameter("@pageid", pageid),
        new SqlParameter("@groupid", groupid)
    };

    DataSet ds = SqlHelper.ExecuteDataset(DowConectingString,
CommandType.StoredProcedure, "GetLeftButtonsPermission", parameters);
return ds;

}

public DataTable GetLeftButtons(int pageid, int groupid)
{
    SqlParameter[] parameters = new SqlParameter[]
    {
        new SqlParameter("@pageid", pageid),
        new SqlParameter("@groupid", groupid)
    };

    DataSet ds = SqlHelper.ExecuteDataset(DowConectingString,
CommandType.StoredProcedure, "GetLeftButtons", parameters);
    DataTable LstUserPer = ds.Tables[0];
return LstUserPer;

}

public DataTable GetAllDeviceByParentId(int ParentId)
{
    SqlParameter[] parameters = new SqlParameter[]
    {
        new SqlParameter("@ParentId", Convert.ToInt32(ParentId)),
    };

```

```

        DataSet ds = SqlHelper.ExecuteDataset(DowConectingString,
        CommandType.StoredProcedure, "spGetAllDeviceByParentId", parameters);
        DataTable LstUserPer = ds.Tables[0];
        return LstUserPer;
    }
    public DataSet GetAllDeviceByParentIdSorting(string Srtcol, string SrtExp, int pageno,
    int pagesize, int ParentId)
    {
        SqlParameter[] parameters = new SqlParameter[]
        {
            new SqlParameter("@OrdCol",Srtcol),
            new SqlParameter("@OrdExp",SrtExp),
            new SqlParameter("@pageNo",pageno),
            new SqlParameter("@pageSize",pagesize),
            new SqlParameter("@ParentId",ParentId)
        };
        DataSet dsUsers = null;
        try
        {
            dsUsers = SqlHelper.ExecuteDataset(DowConectingString,
            CommandType.StoredProcedure, "spGetAllDeviceByParentIdSorting", parameters);
        }
        catch (Exception ex)
        {
            throw ex;
        }

        return dsUsers;
    }
}
}
}

```

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Configuration;
namespace DowDAL
{
    public class ConnectionHelper
    {
        public static string DowConectingString =
        ConfigurationSettings.AppSettings["DowConectingString"].ToString();
    }
}

```

```

using System;
using System.Data;
using System.Collections;

```

```

namespace DowDAL
{
    public class Common
    {

        public static int GetDupRecCount(string TableName, string IdColName,Hashtable
colvalues,string colName)
        {
            IdColName = (IdColName == string.Empty) ? "*" : IdColName;
            string Qry = "select count("+IdColName+") from [" +TableName+"] where ";
            string condition = string.Empty;
            foreach (DictionaryEntry de in colvalues)
            {
                condition += "and " + de.Key + " = " + de.Value +"";
            }

            Qry = Qry + condition.TrimStart("and".ToCharArray());

            return
Convert.ToInt32(SqlHelper.ExecuteScalar(ConnectionHelper.DowConectingString,
CommandType.Text, Qry));
        }

        public static DataSet GetDDIData(string TableName, string ValueColumn, string
TextColumn, Hashtable colvalues)
        {
            string Qry = "select " +ValueColumn + " as colVal ,"+ TextColumn + " as ColTxt from ["
+ TableName + "];";
            string condition = string.Empty;
            if(colvalues!=null){

                foreach (DictionaryEntry de in colvalues)
                {
                    condition += "and " + de.Key + " = " + de.Value +"";
                }

                Qry = Qry + " Where "+ condition.TrimStart("and".ToCharArray());
            }
            return SqlHelper.ExecuteDataset(ConnectionHelper.DowConectingString,
CommandType.Text, Qry);
        }
    }
}

****

//=====
//=====
// Microsoft Data Access Application Block for .NET
// http://msdn.microsoft.com/library/en-us/dnbdta/html/daab-rm.asp
//
// SQLHelper.cs
//
// This file contains the implementations of the SqlHelper and SqlHelperParameterCache
// classes.

```

```
//
// For more information see the Data Access Application Block Implementation Overview.
//
//=====
=====
// Copyright (C) 2000-2001 Microsoft Corporation
// All rights reserved.
// THIS CODE AND INFORMATION IS PROVIDED "AS IS" WITHOUT WARRANTY
// OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT
// LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY AND/OR
// FITNESS FOR A PARTICULAR PURPOSE.
//=====
=====
```

```
using System;
using System.Data;
using System.Xml;
using System.Data.SqlClient;
using System.Collections;
```

```
namespace DowDAL
```

```
{
    /// <summary>
    /// The SqlHelper class is intended to encapsulate high performance, scalable best
practices for
    /// common uses of SqlConnection.
    /// </summary>
    public sealed class SqlHelper
    {
        #region private utility methods & constructors

        //Since this class provides only static methods, make the default constructor
private to prevent
        //instances from being created with "new SqlHelper()".
        private SqlHelper() {}

        /// <summary>
        /// This method is used to attach array of SqlParameter to a SqlCommand.
        ///
        /// This method will assign a value of DBNull to any parameter with a direction
of
        /// InputOutput and a value of null.
        ///
        /// This behavior will prevent default values from being used, but
        /// this will be the less common case than an intended pure output parameter
(dereived as InputOutput)
        /// where the user provided no input value.
        /// </summary>
        /// <param name="command">The command to which the parameters will be
added</param>
        /// <param name="commandParameters">an array of SqlParameter thob e
added to command</param>
```

```

private static void AttachParameters(SqlCommand command, SqlParameter[]
commandParameters)
{
    foreach (SqlParameter p in commandParameters)
    {
        //check for derived output value with no value assigned
        if ((p.Direction == ParameterDirection.InputOutput) &&
(p.Value == null))
        {
            p.Value = DBNull.Value;
        }

        command.Parameters.Add(p);
    }
}

/// <summary>
/// This method assigns an array of values to an array of SqlParameter.
/// </summary>
/// <param name="commandParameters">array of SqlParameter to be
assigned values</param>
/// <param name="parameterValues">array of objects holding the values to be
assigned</param>
private static void AssignParameterValues(SqlParameter[]
commandParameters, object[] parameterValues)
{
    if ((commandParameters == null) || (parameterValues == null))
    {
        //do nothing if we get no data
        return;
    }

    // we must have the same number of values as we have parameters to
put them in
    if (commandParameters.Length != parameterValues.Length)
    {
        throw new ArgumentException("Parameter count does not
match Parameter Value count.");
    }

    //iterate through the SqlParameter, assigning the values from the
corresponding position in the
//value array
for (int i = 0, j = commandParameters.Length; i < j; i++)
{
    commandParameters[i].Value = parameterValues[i];
}
}

/// <summary>
/// This method opens (if necessary) and assigns a connection, transaction,
command type and parameters
/// to the provided command.
/// </summary>
/// <param name="command">the SqlCommand to be prepared</param>

```



```

        /// <param name="connection">a valid SqlConnection, on which to execute
this command</param>
        /// <param name="transaction">a valid SqlTransaction, or 'null'</param>
        /// <param name="commandType">the CommandType (stored procedure,
text, etc.)</param>
        /// <param name="commandText">the stored procedure name or T-SQL
command</param>
        /// <param name="commandParameters">an array of SqlParameter to be
associated with the command or 'null' if no parameters are required</param>
        private static void PrepareCommand(SqlCommand command, SqlConnection
connection, SqlTransaction transaction, CommandType commandType, string
commandText, SqlParameter[] commandParameters)
        {
            //if the provided connection is not open, we will open it
            if (connection.State != ConnectionState.Open)
            {
                connection.Open();
            }

            //associate the connection with the command
            command.Connection = connection;

            //set the command text (stored procedure name or SQL statement)
            command.CommandText = commandText;

            //if we were provided a transaction, assign it.
            if (transaction != null)
            {
                command.Transaction = transaction;
            }

            //set the command type
            command.CommandType = commandType;

            //attach the command parameters if they are provided
            if (commandParameters != null)
            {
                AttachParameters(command, commandParameters);
            }

            return;
        }

#endregion private utility methods & constructors

#region ExecuteNonQuery

    /// <summary>
    /// Execute a SqlCommand (that returns no resultset and takes no
parameters) against the database specified in
    /// the connection string.
    /// </summary>
    /// <remarks>
    /// e.g.:

```

```

        /// int result = ExecuteNonQuery(connString,
CommandType.StoredProcedure, "PublishOrders");
        /// </remarks>
        /// <param name="connectionString">a valid connection string for a
SqlConnection</param>
        /// <param name="commandType">the CommandType (stored procedure,
text, etc.)</param>
        /// <param name="commandText">the stored procedure name or T-SQL
command</param>
        /// <returns>an int representing the number of rows affected by the
command</returns>
        public static int ExecuteNonQuery(string connectionString, CommandType
commandType, string commandText)
        {
            //pass through the call providing null for the set of SqlParameter
return ExecuteNonQuery(connectionString, commandType,
commandText, (SqlParameter[])null);
        }

        /// <summary>
        /// Execute a SqlCommand (that returns no resultset) against the database
specified in the connection string
        /// using the provided parameters.
        /// </summary>
        /// <remarks>
        /// e.g.:
        /// int result = ExecuteNonQuery(connString,
CommandType.StoredProcedure, "PublishOrders", new SqlParameter("@prodid", 24));
        /// </remarks>
        /// <param name="connectionString">a valid connection string for a
SqlConnection</param>
        /// <param name="commandType">the CommandType (stored procedure,
text, etc.)</param>
        /// <param name="commandText">the stored procedure name or T-SQL
command</param>
        /// <param name="commandParameters">an array of SqlParameter used to
execute the command</param>
        /// <returns>an int representing the number of rows affected by the
command</returns>
        public static int ExecuteNonQuery(string connectionString, CommandType
commandType, string commandText, params SqlParameter[] commandParameters)
        {
            //create & open a SqlConnection, and dispose of it after we are done.
            using (SqlConnection cn = new SqlConnection(connectionString))
            {
                cn.Open();

                //call the overload that takes a connection in place of the
connection string
                return ExecuteNonQuery(cn, commandType, commandText,
commandParameters);
            }
        }

        /// <summary>

```

```

        /// Execute a stored procedure via a SqlCommand (that returns no resultset)
        against the database specified in
        /// the connection string using the provided parameter values. This method
        will query the database to discover the parameters for the
        /// stored procedure (the first time each stored procedure is called), and
        assign the values based on parameter order.
        /// </summary>
        /// <remarks>
        /// This method provides no access to output parameters or the stored
        procedure's return value parameter.
        ///
        /// e.g.:
        /// int result = ExecuteNonQuery(connString, "PublishOrders", 24, 36);
        /// </remarks>
        /// <param name="connectionString">a valid connection string for a
        SqlConnection</param>
        /// <param name="spName">the name of the stored procedure</param>
        /// <param name="parameterValues">an array of objects to be assigned as
        the input values of the stored procedure</param>
        /// <returns>an int representing the number of rows affected by the
        command</returns>
        public static int ExecuteNonQuery(string connectionString, string spName,
        params object[] parameterValues)
        {
            ///if we receive parameter values, we need to figure out where they go
            if ((parameterValues != null) && (parameterValues.Length > 0))
            {
                ///pull the parameters for this stored procedure from the
                parameter cache (or discover them & populate the cache)
                SqlParameter[] commandParameters =
                SqlHelper.ParameterCache.GetSpParameterSet(connectionString, spName);

                ///assign the provided values to these parameters based on
                parameter order
                AssignParameterValues(commandParameters,
                parameterValues);

                ///call the overload that takes an array of SqlParameter
                return ExecuteNonQuery(connectionString,
                CommandType.StoredProcedure, spName, commandParameters);
            }
            ///otherwise we can just call the SP without params
            else
            {
                return ExecuteNonQuery(connectionString,
                CommandType.StoredProcedure, spName);
            }
        }

        /// <summary>
        /// Execute a SqlCommand (that returns no resultset and takes no
        parameters) against the provided SqlConnection.
        /// </summary>
        /// <remarks>
        /// e.g.:

```

```

        /// int result = ExecuteNonQuery(conn, CommandType.StoredProcedure,
"PublishOrders");
        /// </remarks>
        /// <param name="connection">a valid SqlConnection</param>
        /// <param name="commandType">the CommandType (stored procedure,
text, etc.)</param>
        /// <param name="commandText">the stored procedure name or T-SQL
command</param>
        /// <returns>an int representing the number of rows affected by the
command</returns>
        public static int ExecuteNonQuery(SqlConnection connection, CommandType
commandType, string commandText)
        {
            //pass through the call providing null for the set of SqlParameter
return ExecuteNonQuery(connection, commandType, commandText,
(SqlParameter[])null);
        }

        /// <summary>
        /// Execute a SqlCommand (that returns no resultset) against the specified
SqlConnection
        /// using the provided parameters.
        /// </summary>
        /// <remarks>
        /// e.g.:
        /// int result = ExecuteNonQuery(conn, CommandType.StoredProcedure,
"PublishOrders", new SqlParameter("@prodid", 24));
        /// </remarks>
        /// <param name="connection">a valid SqlConnection</param>
        /// <param name="commandType">the CommandType (stored procedure,
text, etc.)</param>
        /// <param name="commandText">the stored procedure name or T-SQL
command</param>
        /// <param name="commandParameters">an array of SqlParameter used to
execute the command</param>
        /// <returns>an int representing the number of rows affected by the
command</returns>
        public static int ExecuteNonQuery(SqlConnection connection, CommandType
commandType, string commandText, params SqlParameter[] commandParameters)
        {
            //create a command and prepare it for execution
            SqlCommand cmd = new SqlCommand();
            PrepareCommand(cmd, connection, (SqlTransaction)null,
commandType, commandText, commandParameters);

            //finally, execute the command.
            int retval = cmd.ExecuteNonQuery();

            // detach the SqlParameter from the command object, so they can be
used again.
            cmd.Parameters.Clear();
            return retval;
        }

        /// <summary>

```

```

        /// Execute a stored procedure via a SqlCommand (that returns no resultset)
        against the specified SqlConnection
        /// using the provided parameter values. This method will query the database
        to discover the parameters for the
        /// stored procedure (the first time each stored procedure is called), and
        assign the values based on parameter order.
        /// </summary>
        /// <remarks>
        /// This method provides no access to output parameters or the stored
        procedure's return value parameter.
        ///
        /// e.g.:
        /// int result = ExecuteNonQuery(conn, "PublishOrders", 24, 36);
        /// </remarks>
        /// <param name="connection">a valid SqlConnection</param>
        /// <param name="spName">the name of the stored procedure</param>
        /// <param name="parameterValues">an array of objects to be assigned as
        the input values of the stored procedure</param>
        /// <returns>an int representing the number of rows affected by the
        command</returns>
        public static int ExecuteNonQuery(SqlConnection connection, string spName,
        params object[] parameterValues)
        {
            ///if we receive parameter values, we need to figure out where they go
            if ((parameterValues != null) && (parameterValues.Length > 0))
            {
                ///pull the parameters for this stored procedure from the
                parameter cache (or discover them & populate the cache)
                SqlParameter[] commandParameters =
                SqlHelper.ParameterCache.GetSpParameterSet(connection.ConnectionString, spName);

                ///assign the provided values to these parameters based on
                parameter order
                AssignParameterValues(commandParameters,
                parameterValues);

                ///call the overload that takes an array of SqlParameter
                return ExecuteNonQuery(connection,
                CommandType.StoredProcedure, spName, commandParameters);
            }
            ///otherwise we can just call the SP without params
            else
            {
                return ExecuteNonQuery(connection,
                CommandType.StoredProcedure, spName);
            }
        }

        /// <summary>
        /// Execute a SqlCommand (that returns no resultset and takes no
        parameters) against the provided SqlTransaction.
        /// </summary>
        /// <remarks>
        /// e.g.:

```

```

        /// int result = ExecuteNonQuery(trans, CommandType.StoredProcedure,
"PublishOrders");
        /// </remarks>
        /// <param name="transaction">a valid SqlTransaction</param>
        /// <param name="commandType">the CommandType (stored procedure,
text, etc.)</param>
        /// <param name="commandText">the stored procedure name or T-SQL
command</param>
        /// <returns>an int representing the number of rows affected by the
command</returns>
        public static int ExecuteNonQuery(SqlTransaction transaction,
CommandType commandType, string commandText)
        {
            //pass through the call providing null for the set of SqlParameter
return ExecuteNonQuery(transaction, commandType, commandText,
(SqlParameter[])null);
        }

        /// <summary>
        /// Execute a SqlCommand (that returns no resultset) against the specified
SqlTransaction
        /// using the provided parameters.
        /// </summary>
        /// <remarks>
        /// e.g.:
        /// int result = ExecuteNonQuery(trans, CommandType.StoredProcedure,
"GetOrders", new SqlParameter("@prodid", 24));
        /// </remarks>
        /// <param name="transaction">a valid SqlTransaction</param>
        /// <param name="commandType">the CommandType (stored procedure,
text, etc.)</param>
        /// <param name="commandText">the stored procedure name or T-SQL
command</param>
        /// <param name="commandParameters">an array of SqlParameter used to
execute the command</param>
        /// <returns>an int representing the number of rows affected by the
command</returns>
        public static int ExecuteNonQuery(SqlTransaction transaction,
CommandType commandType, string commandText, params SqlParameter[]
commandParameters)
        {
            //create a command and prepare it for execution
            SqlCommand cmd = new SqlCommand();
            PrepareCommand(cmd, transaction.Connection, transaction,
commandType, commandText, commandParameters);

            //finally, execute the command.
            int retval = cmd.ExecuteNonQuery();

            // detach the SqlParameter from the command object, so they can be
used again.

            cmd.Parameters.Clear();
            return retval;
        }

```

```

        /// <summary>
        /// Execute a stored procedure via a SqlCommand (that returns no resultset)
against the specified
        /// SqlConnection using the provided parameter values. This method will
query the database to discover the parameters for the
        /// stored procedure (the first time each stored procedure is called), and
assign the values based on parameter order.
        /// </summary>
        /// <remarks>
        /// This method provides no access to output parameters or the stored
procedure's return value parameter.
        ///
        /// e.g.:
        /// int result = ExecuteNonQuery(conn, trans, "PublishOrders", 24, 36);
        /// </remarks>
        /// <param name="transaction">a valid SqlConnection</param>
        /// <param name="spName">the name of the stored procedure</param>
        /// <param name="parameterValues">an array of objects to be assigned as
the input values of the stored procedure</param>
        /// <returns>an int representing the number of rows affected by the
command</returns>
        public static int ExecuteNonQuery(SqlTransaction transaction, string spName,
params object[] parameterValues)
        {
            //if we receive parameter values, we need to figure out where they go
            if ((parameterValues != null) && (parameterValues.Length > 0))
            {
                //pull the parameters for this stored procedure from the
parameter cache (or discover them & populate the cache)
                SqlParameter[] commandParameters =
SqlHelperParameterCache.GetSpParameterSet(transaction.Connection.ConnectionString,
spName);

                //assign the provided values to these parameters based on
parameter order
                AssignParameterValues(commandParameters,
parameterValues);

                //call the overload that takes an array of SqlParameter
                return ExecuteNonQuery(transaction,
CommandType.StoredProcedure, spName, commandParameters);
            }
            //otherwise we can just call the SP without params
            else
            {
                return ExecuteNonQuery(transaction,
CommandType.StoredProcedure, spName);
            }
        }

        #endregion ExecuteNonQuery

        #region ExecuteDataSet

```

```

        /// <summary>
        /// Execute a SqlCommand (that returns a resultset and takes no parameters)
against the database specified in
        /// the connection string.
        /// </summary>
        /// <remarks>
        /// e.g.:
        /// DataSet ds = ExecuteDataset(connString,
CommandType.StoredProcedure, "GetOrders");
        /// </remarks>
        /// <param name="connectionString">a valid connection string for a
SqlConnection</param>
        /// <param name="commandType">the CommandType (stored procedure,
text, etc.)</param>
        /// <param name="commandText">the stored procedure name or T-SQL
command</param>
        /// <returns>a dataset containing the resultset generated by the
command</returns>
        public static DataSet ExecuteDataset(string connectionString, CommandType
commandType, string commandText)
        {
            //pass through the call providing null for the set of SqlParameter
return ExecuteDataset(connectionString, commandType,
commandText, (SqlParameter[])null);
        }

```

```

        /// <summary>
        /// Execute a SqlCommand (that returns a resultset) against the database
specified in the connection string
        /// using the provided parameters.
        /// </summary>
        /// <remarks>
        /// e.g.:
        /// DataSet ds = ExecuteDataset(connString,
CommandType.StoredProcedure, "GetOrders", new SqlParameter("@prodid", 24));
        /// </remarks>
        /// <param name="connectionString">a valid connection string for a
SqlConnection</param>
        /// <param name="commandType">the CommandType (stored procedure,
text, etc.)</param>
        /// <param name="commandText">the stored procedure name or T-SQL
command</param>
        /// <param name="commandParameters">an array of SqlParameter
used to
execute the command</param>
        /// <returns>a dataset containing the resultset generated by the
command</returns>
        public static DataSet ExecuteDataset(string connectionString, CommandType
commandType, string commandText, params SqlParameter[] commandParameters)
        {
            //create & open a SqlConnection, and dispose of it after we are done.
using (SqlConnection cn = new SqlConnection(connectionString))
            {
                cn.Open();

```



```

        //call the overload that takes a connection in place of the
connection string
        return ExecuteDataset(cn, commandType, commandText,
commandParameters);
    }

    /// <summary>
    /// Execute a stored procedure via a SqlCommand (that returns a resultset)
against the database specified in
    /// the connection string using the provided parameter values. This method
will query the database to discover the parameters for the
    /// stored procedure (the first time each stored procedure is called), and
assign the values based on parameter order.
    /// </summary>
    /// <remarks>
    /// This method provides no access to output parameters or the stored
procedure's return value parameter.
    ///
    /// e.g.:
    /// DataSet ds = ExecuteDataset(connString, "GetOrders", 24, 36);
    /// </remarks>
    /// <param name="connectionString">a valid connection string for a
SqlConnection</param>
    /// <param name="spName">the name of the stored procedure</param>
    /// <param name="parameterValues">an array of objects to be assigned as
the input values of the stored procedure</param>
    /// <returns>a dataset containing the resultset generated by the
command</returns>
    public static DataSet ExecuteDataset(string connectionString, string spName,
params object[] parameterValues)
    {
        //if we receive parameter values, we need to figure out where they go
        if ((parameterValues != null) && (parameterValues.Length > 0))
        {
            //pull the parameters for this stored procedure from the
parameter cache (or discover them & populate the cache)
            SqlParameter[] commandParameters =
            SqlHelper.ParameterCache.GetSpParameterSet(connectionString, spName);

            //assign the provided values to these parameters based on
parameter order
            AssignParameterValues(commandParameters,
parameterValues);

            //call the overload that takes an array of SqlParameter
return ExecuteDataset(connectionString,
            CommandType.StoredProcedure, spName, commandParameters);
        }
        //otherwise we can just call the SP without params
        else
        {
            return ExecuteDataset(connectionString,
            CommandType.StoredProcedure, spName);
        }
    }

```

```

    }

    /// <summary>
    /// Execute a SqlCommand (that returns a resultset and takes no parameters)
against the provided SqlConnection.
    /// </summary>
    /// <remarks>
    /// e.g.:
    /// DataSet ds = ExecuteDataset(conn, CommandType.StoredProcedure,
"GetOrders");
    /// </remarks>
    /// <param name="connection">a valid SqlConnection</param>
    /// <param name="commandType">the CommandType (stored procedure,
text, etc.)</param>
    /// <param name="commandText">the stored procedure name or T-SQL
command</param>
    /// <returns>a dataset containing the resultset generated by the
command</returns>
    public static DataSet ExecuteDataset(SqlConnection connection,
CommandType commandType, string commandText)
    {
        //pass through the call providing null for the set of SqlParameter
return ExecuteDataset(connection, commandType, commandText,
(SqlParameter[])null);
    }

    /// <summary>
    /// Execute a SqlCommand (that returns a resultset) against the specified
SqlConnection
    /// using the provided parameters.
    /// </summary>
    /// <remarks>
    /// e.g.:
    /// DataSet ds = ExecuteDataset(conn, CommandType.StoredProcedure,
"GetOrders", new SqlParameter("@prodid", 24));
    /// </remarks>
    /// <param name="connection">a valid SqlConnection</param>
    /// <param name="commandType">the CommandType (stored procedure,
text, etc.)</param>
    /// <param name="commandText">the stored procedure name or T-SQL
command</param>
    /// <param name="commandParameters">an array of SqlParameter used to
execute the command</param>
    /// <returns>a dataset containing the resultset generated by the
command</returns>
    public static DataSet ExecuteDataset(SqlConnection connection,
CommandType commandType, string commandText, params SqlParameter[]
commandParameters)
    {
        //create a command and prepare it for execution
        SqlCommand cmd = new SqlCommand();
        PrepareCommand(cmd, connection, (SqlTransaction)null,
commandType, commandText, commandParameters);

        //create the DataAdapter & DataSet

```

```

        SqlDataAdapter da = new SqlDataAdapter(cmd);
        DataSet ds = new DataSet();

        //fill the DataSet using default values for DataTable names, etc.
        da.Fill(ds);

        // detach the SqlParameter objects from the command object, so they can be
used again.
        cmd.Parameters.Clear();

        //return the dataset
        return ds;
    }

    /// <summary>
    /// Execute a stored procedure via a SqlCommand (that returns a resultset)
against the specified SqlConnection
    /// using the provided parameter values. This method will query the database
to discover the parameters for the
    /// stored procedure (the first time each stored procedure is called), and
assign the values based on parameter order.
    /// </summary>
    /// <remarks>
    /// This method provides no access to output parameters or the stored
procedure's return value parameter.
    ///
    /// e.g.:
    /// DataSet ds = ExecuteDataset(conn, "GetOrders", 24, 36);
    /// </remarks>
    /// <param name="connection">a valid SqlConnection</param>
    /// <param name="spName">the name of the stored procedure</param>
    /// <param name="parameterValues">an array of objects to be assigned as
the input values of the stored procedure</param>
    /// <returns>a dataset containing the resultset generated by the
command</returns>
    public static DataSet ExecuteDataset(SqlConnection connection, string
spName, params object[] parameterValues)
    {
        //if we receive parameter values, we need to figure out where they go
        if ((parameterValues != null) && (parameterValues.Length > 0))
        {
            //pull the parameters for this stored procedure from the
parameter cache (or discover them & populate the cache)
            SqlParameter[] commandParameters =
SqlHelper.ParameterCache.GetSpParameterSet(connection.ConnectionString, spName);

            //assign the provided values to these parameters based on
parameter order
            AssignParameterValues(commandParameters,
parameterValues);

            //call the overload that takes an array of SqlParameter objects
return ExecuteDataset(connection,
CommandType.StoredProcedure, spName, commandParameters);
        }
    }

```

```

        //otherwise we can just call the SP without params
    else
    {
        return ExecuteDataset(connection,
CommandType.StoredProcedure, spName);
    }
}

/// <summary>
/// Execute a SqlCommand (that returns a resultset and takes no parameters)
against the provided SqlTransaction.
/// </summary>
/// <remarks>
/// e.g.:
/// DataSet ds = ExecuteDataset(trans, CommandType.StoredProcedure,
"GetOrders");
/// </remarks>
/// <param name="transaction">a valid SqlTransaction</param>
/// <param name="commandType">the CommandType (stored procedure,
text, etc.)</param>
/// <param name="commandText">the stored procedure name or T-SQL
command</param>
/// <returns>a dataset containing the resultset generated by the
command</returns>
public static DataSet ExecuteDataset(SqlTransaction transaction,
CommandType commandType, string commandText)
{
    //pass through the call providing null for the set of SqlParameter
(SqlParameter[])null);
}

/// <summary>
/// Execute a SqlCommand (that returns a resultset) against the specified
SqlTransaction
/// using the provided parameters.
/// </summary>
/// <remarks>
/// e.g.:
/// DataSet ds = ExecuteDataset(trans, CommandType.StoredProcedure,
"GetOrders", new SqlParameter("@prodid", 24));
/// </remarks>
/// <param name="transaction">a valid SqlTransaction</param>
/// <param name="commandType">the CommandType (stored procedure,
text, etc.)</param>
/// <param name="commandText">the stored procedure name or T-SQL
command</param>
/// <param name="commandParameters">an array of SqlParameter used to
execute the command</param>
/// <returns>a dataset containing the resultset generated by the
command</returns>
public static DataSet ExecuteDataset(SqlTransaction transaction,
CommandType commandType, string commandText, params SqlParameter[]
commandParameters)
{

```

```

        //create a command and prepare it for execution
        SqlCommand cmd = new SqlCommand();
        PrepareCommand(cmd, transaction.Connection, transaction,
commandType, commandText, commandParameters);

        //create the DataAdapter & DataSet
        SqlDataAdapter da = new SqlDataAdapter(cmd);
        DataSet ds = new DataSet();

        //fill the DataSet using default values for DataTable names, etc.
        da.Fill(ds);

        // detach the SqlParameter from the command object, so they can be
used again.
        cmd.Parameters.Clear();

        //return the dataset
        return ds;
    }

    /// <summary>
    /// Execute a stored procedure via a SqlCommand (that returns a resultset)
against the specified
    /// SqlConnection using the provided parameter values. This method will
query the database to discover the parameters for the
    /// stored procedure (the first time each stored procedure is called), and
assign the values based on parameter order.
    /// </summary>
    /// <remarks>
    /// This method provides no access to output parameters or the stored
procedure's return value parameter.
    ///
    /// e.g.:
    /// DataSet ds = ExecuteDataset(trans, "GetOrders", 24, 36);
    /// </remarks>
    /// <param name="transaction">a valid SqlConnection</param>
    /// <param name="spName">the name of the stored procedure</param>
    /// <param name="parameterValues">an array of objects to be assigned as
the input values of the stored procedure</param>
    /// <returns>a dataset containing the resultset generated by the
command</returns>
    public static DataSet ExecuteDataset(SqlTransaction transaction, string
spName, params object[] parameterValues)
    {
        //if we receive parameter values, we need to figure out where they go
        if ((parameterValues != null) && (parameterValues.Length > 0))
        {
            //pull the parameters for this stored procedure from the
parameter cache (or discover them & populate the cache)
            SqlParameter[] commandParameters =
SqlHelperParameterCache.GetSpParameterSet(transaction.Connection.ConnectionString,
spName);

            //assign the provided values to these parameters based on
parameter order

```

```

        AssignParameterValues(commandParameters,
parameterValues);

        //call the overload that takes an array of SqlParameter
return ExecuteDataset(transaction,
CommandType.StoredProcedure, spName, commandParameters);
    }
    //otherwise we can just call the SP without params
    else
    {
        return ExecuteDataset(transaction,
CommandType.StoredProcedure, spName);
    }
}

#endregion ExecuteDataSet

#region ExecuteReader

    /// <summary>
    /// this enum is used to indicate whether the connection was provided by the
caller, or created by SqlHelper, so that
    /// we can set the appropriate CommandBehavior when calling
ExecuteReader()
    /// </summary>
    private enum SqlConnectionOwnership
    {
        /// <summary>Connection is owned and managed by
SqlHelper</summary>
        Internal,
        /// <summary>Connection is owned and managed by the
caller</summary>
        External
    }

    /// <summary>
    /// Create and prepare a SqlCommand, and call ExecuteReader with the
appropriate CommandBehavior.
    /// </summary>
    /// <remarks>
    /// If we created and opened the connection, we want the connection to be
closed when the DataReader is closed.
    ///
    /// If the caller provided the connection, we want to leave it to them to
manage.
    /// </remarks>
    /// <param name="connection">a valid SqlConnection, on which to execute
this command</param>
    /// <param name="transaction">a valid SqlTransaction, or 'null'</param>
    /// <param name="commandType">the CommandType (stored procedure,
text, etc.)</param>
    /// <param name="commandText">the stored procedure name or T-SQL
command</param>
    /// <param name="commandParameters">an array of SqlParameter to be
associated with the command or 'null' if no parameters are required</param>

```

```

        /// <param name="connectionOwnership">indicates whether the connection
parameter was provided by the caller, or created by SqlHelper</param>
        /// <returns>SqlDataReader containing the results of the command</returns>
        private static SqlDataReader ExecuteReader(SqlConnection connection,
SqlTransaction transaction, CommandType commandType, string commandText,
SqlParameter[] commandParameters, SqlConnectionOwnership connectionOwnership)
        {
            //create a command and prepare it for execution
            SqlCommand cmd = new SqlCommand();
            PrepareCommand(cmd, connection, transaction, commandType,
commandText, commandParameters);

            //create a reader
            SqlDataReader dr;

            // call ExecuteReader with the appropriate CommandBehavior
            if (connectionOwnership == SqlConnectionOwnership.External)
            {
                dr = cmd.ExecuteReader();
            }
            else
            {
                dr =
cmd.ExecuteReader(CommandBehavior.CloseConnection);
            }

            // detach the SqlParameter from the command object, so they can be
used again.
            cmd.Parameters.Clear();

            return dr;
        }

        /// <summary>
        /// Execute a SqlCommand (that returns a resultset and takes no parameters)
against the database specified in
        /// the connection string.
        /// </summary>
        /// <remarks>
        /// e.g.:
        /// SqlDataReader dr = ExecuteReader(connString,
CommandType.StoredProcedure, "GetOrders");
        /// </remarks>
        /// <param name="connectionString">a valid connection string for a
SqlConnection</param>
        /// <param name="commandType">the CommandType (stored procedure,
text, etc.)</param>
        /// <param name="commandText">the stored procedure name or T-SQL
command</param>
        /// <returns>a SqlDataReader containing the resultset generated by the
command</returns>
        public static SqlDataReader ExecuteReader(string connectionString,
CommandType commandType, string commandText)
        {
            //pass through the call providing null for the set of SqlParameter

```

```

        return ExecuteReader(connectionString, commandType,
commandText, (SqlParameter[])null);
    }

    /// <summary>
    /// Execute a SqlCommand (that returns a resultset) against the database
specified in the connection string
    /// using the provided parameters.
    /// </summary>
    /// <remarks>
    /// e.g.:
    /// SqlDataReader dr = ExecuteReader(connString,
CommandType.StoredProcedure, "GetOrders", new SqlParameter("@prodid", 24));
    /// </remarks>
    /// <param name="connectionString">a valid connection string for a
SqlConnection</param>
    /// <param name="commandType">the CommandType (stored procedure,
text, etc.)</param>
    /// <param name="commandText">the stored procedure name or T-SQL
command</param>
    /// <param name="commandParameters">an array of SqlParameter used to
execute the command</param>
    /// <returns>a SqlDataReader containing the resultset generated by the
command</returns>
    public static SqlDataReader ExecuteReader(string connectionString,
CommandType commandType, string commandText, params SqlParameter[]
commandParameters)
    {
        //create & open a SqlConnection
        SqlConnection cn = new SqlConnection(connectionString);
        cn.Open();

        try
        {
            //call the private overload that takes an internally owned
connection in place of the connection string
            return ExecuteReader(cn, null, commandType, commandText,
commandParameters, SqlConnectionOwnership.Internal);
        }
        catch
        {
            //if we fail to return the SqlDatReader, we need to close the
connection ourselves
            cn.Close();
            throw;
        }
    }

    /// <summary>
    /// Execute a stored procedure via a SqlCommand (that returns a resultset)
against the database specified in
    /// the connection string using the provided parameter values. This method
will query the database to discover the parameters for the
    /// stored procedure (the first time each stored procedure is called), and
assign the values based on parameter order.

```



```

        /// </summary>
        /// <remarks>
        /// This method provides no access to output parameters or the stored
        procedure's return value parameter.
        ///
        /// e.g.:
        /// SqlDataReader dr = ExecuteReader(connString, "GetOrders", 24, 36);
        /// </remarks>
        /// <param name="connectionString">a valid connection string for a
        SqlConnection</param>
        /// <param name="spName">the name of the stored procedure</param>
        /// <param name="parameterValues">an array of objects to be assigned as
        the input values of the stored procedure</param>
        /// <returns>a SqlDataReader containing the resultset generated by the
        command</returns>
        public static SqlDataReader ExecuteReader(string connectionString, string
        spName, params object[] parameterValues)
        {
            //if we receive parameter values, we need to figure out where they go
            if ((parameterValues != null) && (parameterValues.Length > 0))
            {
                //pull the parameters for this stored procedure from the
                parameter cache (or discover them & populate the cache)
                SqlParameter[] commandParameters =
                SqlHelperParameterCache.GetSpParameterSet(connectionString, spName);

                //assign the provided values to these parameters based on
                parameter order
                AssignParameterValues(commandParameters,
                parameterValues);

                //call the overload that takes an array of SqlParameter
                return ExecuteReader(connectionString,
                CommandType.StoredProcedure, spName, commandParameters);
            }
            //otherwise we can just call the SP without params
            else
            {
                return ExecuteReader(connectionString,
                CommandType.StoredProcedure, spName);
            }
        }

        /// <summary>
        /// Execute a SqlCommand (that returns a resultset and takes no parameters)
        against the provided SqlConnection.
        /// </summary>
        /// <remarks>
        /// e.g.:
        /// SqlDataReader dr = ExecuteReader(conn,
        CommandType.StoredProcedure, "GetOrders");
        /// </remarks>
        /// <param name="connection">a valid SqlConnection</param>
        /// <param name="commandType">the CommandType (stored procedure,
        text, etc.)</param>

```

```

        /// <param name="commandText">the stored procedure name or T-SQL
command</param>
        /// <returns>a SqlDataReader containing the resultset generated by the
command</returns>
        public static SqlDataReader ExecuteReader(SqlConnection connection,
CommandType commandType, string commandText)
        {
            //pass through the call providing null for the set of SqlParameter
return ExecuteReader(connection, commandType, commandText,
(SqlParameter[])null);
        }

        /// <summary>
        /// Execute a SqlCommand (that returns a resultset) against the specified
SqlConnection
        /// using the provided parameters.
        /// </summary>
        /// <remarks>
        /// e.g.:
        /// SqlDataReader dr = ExecuteReader(conn,
CommandType.StoredProcedure, "GetOrders", new SqlParameter("@prodid", 24));
        /// </remarks>
        /// <param name="connection">a valid SqlConnection</param>
        /// <param name="commandType">the CommandType (stored procedure,
text, etc.)</param>
        /// <param name="commandText">the stored procedure name or T-SQL
command</param>
        /// <param name="commandParameters">an array of SqlParameter used to
execute the command</param>
        /// <returns>a SqlDataReader containing the resultset generated by the
command</returns>
        public static SqlDataReader ExecuteReader(SqlConnection connection,
CommandType commandType, string commandText, params SqlParameter[]
commandParameters)
        {
            //pass through the call to the private overload using a null transaction
value and an externally owned connection
            return ExecuteReader(connection, (SqlTransaction)null,
commandType, commandText, commandParameters, SqlConnectionOwnership.External);
        }

        /// <summary>
        /// Execute a stored procedure via a SqlCommand (that returns a resultset)
against the specified SqlConnection
        /// using the provided parameter values. This method will query the database
to discover the parameters for the
        /// stored procedure (the first time each stored procedure is called), and
assign the values based on parameter order.
        /// </summary>
        /// <remarks>
        /// This method provides no access to output parameters or the stored
procedure's return value parameter.
        ///
        /// e.g.:
        /// SqlDataReader dr = ExecuteReader(conn, "GetOrders", 24, 36);

```

```

    /// </remarks>
    /// <param name="connection">a valid SqlConnection</param>
    /// <param name="spName">the name of the stored procedure</param>
    /// <param name="parameterValues">an array of objects to be assigned as
the input values of the stored procedure</param>
    /// <returns>a SqlDataReader containing the resultset generated by the
command</returns>
    public static SqlDataReader ExecuteReader(SqlConnection connection, string
spName, params object[] parameterValues)
    {
        //if we receive parameter values, we need to figure out where they go
        if ((parameterValues != null) && (parameterValues.Length > 0))
        {
            SqlParameter[] commandParameters =
SqlHelper.ParameterCache.GetSpParameterSet(connection.ConnectionString, spName);

            AssignParameterValues(commandParameters,
parameterValues);

            return ExecuteReader(connection,
CommandType.StoredProcedure, spName, commandParameters);
        }
        //otherwise we can just call the SP without params
        else
        {
            return ExecuteReader(connection,
CommandType.StoredProcedure, spName);
        }
    }

    /// <summary>
    /// Execute a SqlCommand (that returns a resultset and takes no parameters)
against the provided SqlTransaction.
    /// </summary>
    /// <remarks>
    /// e.g.:
    /// SqlDataReader dr = ExecuteReader(trans,
CommandType.StoredProcedure, "GetOrders");
    /// </remarks>
    /// <param name="transaction">a valid SqlTransaction</param>
    /// <param name="commandType">the CommandType (stored procedure,
text, etc.)</param>
    /// <param name="commandText">the stored procedure name or T-SQL
command</param>
    /// <returns>a SqlDataReader containing the resultset generated by the
command</returns>
    public static SqlDataReader ExecuteReader(SqlTransaction transaction,
CommandType commandType, string commandText)
    {
        //pass through the call providing null for the set of SqlParameter
return ExecuteReader(transaction, commandType, commandText,
(SqlParameter[])null);
    }

    /// <summary>

```

```

        /// Execute a SqlCommand (that returns a resultset) against the specified
SqlConnection
        /// using the provided parameters.
        /// </summary>
        /// <remarks>
        /// e.g.:
        /// SqlDataReader dr = ExecuteReader(trans,
CommandType.StoredProcedure, "GetOrders", new SqlParameter("@prodid", 24));
        /// </remarks>
        /// <param name="transaction">a valid SqlConnection</param>
        /// <param name="commandType">the CommandType (stored procedure,
text, etc.)</param>
        /// <param name="commandText">the stored procedure name or T-SQL
command</param>
        /// <param name="commandParameters">an array of SqlParameter used to
execute the command</param>
        /// <returns>a SqlDataReader containing the resultset generated by the
command</returns>
        public static SqlDataReader ExecuteReader(SqlConnection transaction,
CommandType commandType, string commandText, params SqlParameter[]
commandParameters)
        {
            //pass through to private overload, indicating that the connection is
owned by the caller
            return ExecuteReader(transaction.Connection, transaction,
commandType, commandText, commandParameters, SqlConnectionOwnership.External);
        }

        /// <summary>
        /// Execute a stored procedure via a SqlCommand (that returns a resultset)
against the specified
        /// SqlConnection using the provided parameter values. This method will
query the database to discover the parameters for the
        /// stored procedure (the first time each stored procedure is called), and
assign the values based on parameter order.
        /// </summary>
        /// <remarks>
        /// This method provides no access to output parameters or the stored
procedure's return value parameter.
        ///
        /// e.g.:
        /// SqlDataReader dr = ExecuteReader(trans, "GetOrders", 24, 36);
        /// </remarks>
        /// <param name="transaction">a valid SqlConnection</param>
        /// <param name="spName">the name of the stored procedure</param>
        /// <param name="parameterValues">an array of objects to be assigned as
the input values of the stored procedure</param>
        /// <returns>a SqlDataReader containing the resultset generated by the
command</returns>
        public static SqlDataReader ExecuteReader(SqlConnection transaction,
string spName, params object[] parameterValues)
        {
            //if we receive parameter values, we need to figure out where they go
            if ((parameterValues != null) && (parameterValues.Length > 0))
            {

```

```

        SqlParameter[] commandParameters =
SqlHelperParameterCache.GetSpParameterSet(transaction.Connection.ConnectionString,
spName);

        AssignParameterValues(commandParameters,
parameterValues);

        return ExecuteReader(transaction,
CommandType.StoredProcedure, spName, commandParameters);
    }
    //otherwise we can just call the SP without params
    else
    {
        return ExecuteReader(transaction,
CommandType.StoredProcedure, spName);
    }
}

#endregion ExecuteReader

#region ExecuteScalar

    /// <summary>
    /// Execute a SqlCommand (that returns a 1x1 resultset and takes no
parameters) against the database specified in
    /// the connection string.
    /// </summary>
    /// <remarks>
    /// e.g.:
    /// int orderCount = (int)ExecuteScalar(connString,
CommandType.StoredProcedure, "GetOrderCount");
    /// </remarks>
    /// <param name="connectionString">a valid connection string for a
SqlConnection</param>
    /// <param name="commandType">the CommandType (stored procedure,
text, etc.)</param>
    /// <param name="commandText">the stored procedure name or T-SQL
command</param>
    /// <returns>an object containing the value in the 1x1 resultset generated by
the command</returns>
    public static object ExecuteScalar(string connectionString, CommandType
commandType, string commandText)
    {
        //pass through the call providing null for the set of SqlParameter
return ExecuteScalar(connectionString, commandType,
commandText, (SqlParameter[])null);
    }

    /// <summary>
    /// Execute a SqlCommand (that returns a 1x1 resultset) against the database
specified in the connection string
    /// using the provided parameters.
    /// </summary>
    /// <remarks>
    /// e.g.:

```

```

        /// int orderCount = (int)ExecuteScalar(connString,
SqlCommand.StoredProcedure, "GetOrderCount", new SqlParameter("@prodid", 24));
        /// </remarks>
        /// <param name="connectionString">a valid connection string for a
SqlConnection</param>
        /// <param name="commandType">the CommandType (stored procedure,
text, etc.)</param>
        /// <param name="commandText">the stored procedure name or T-SQL
command</param>
        /// <param name="commandParameters">an array of SqlParameter used to
execute the command</param>
        /// <returns>an object containing the value in the 1x1 resultset generated by
the command</returns>
        public static object ExecuteScalar(string connectionString, CommandType
commandType, string commandText, params SqlParameter[] commandParameters)
        {
            //create & open a SqlConnection, and dispose of it after we are done.
            using (SqlConnection cn = new SqlConnection(connectionString))
            {
                cn.Open();

                //call the overload that takes a connection in place of the
connection string
                return ExecuteScalar(cn, commandType, commandText,
commandParameters);
            }
        }

        /// <summary>
        /// Execute a stored procedure via a SqlCommand (that returns a 1x1
resultset) against the database specified in
        /// the connection string using the provided parameter values. This method
will query the database to discover the parameters for the
        /// stored procedure (the first time each stored procedure is called), and
assign the values based on parameter order.
        /// </summary>
        /// <remarks>
        /// This method provides no access to output parameters or the stored
procedure's return value parameter.
        ///
        /// e.g.:
        /// int orderCount = (int)ExecuteScalar(connString, "GetOrderCount", 24, 36);
        /// </remarks>
        /// <param name="connectionString">a valid connection string for a
SqlConnection</param>
        /// <param name="spName">the name of the stored procedure</param>
        /// <param name="parameterValues">an array of objects to be assigned as
the input values of the stored procedure</param>
        /// <returns>an object containing the value in the 1x1 resultset generated by
the command</returns>
        public static object ExecuteScalar(string connectionString, string spName,
params object[] parameterValues)
        {
            //if we receive parameter values, we need to figure out where they go
            if ((parameterValues != null) && (parameterValues.Length > 0))

```

```

        {
            //pull the parameters for this stored procedure from the
parameter cache (or discover them & populate the cache)
            SqlParameter[] commandParameters =
SqlHelperParameterCache.GetSpParameterSet(connectionString, spName);

            //assign the provided values to these parameters based on
parameter order
            AssignParameterValues(commandParameters,
parameterValues);

            //call the overload that takes an array of SqlParameter
return ExecuteScalar(connectionString,
CommandType.StoredProcedure, spName, commandParameters);
        }
        //otherwise we can just call the SP without params
        else
        {
            return ExecuteScalar(connectionString,
CommandType.StoredProcedure, spName);
        }
    }

    /// <summary>
    /// Execute a SqlCommand (that returns a 1x1 resultset and takes no
parameters) against the provided SqlConnection.
    /// </summary>
    /// <remarks>
    /// e.g.:
    /// int orderCount = (int)ExecuteScalar(conn,
CommandType.StoredProcedure, "GetOrderCount");
    /// </remarks>
    /// <param name="connection">a valid SqlConnection</param>
    /// <param name="commandType">the CommandType (stored procedure,
text, etc.)</param>
    /// <param name="commandText">the stored procedure name or T-SQL
command</param>
    /// <returns>an object containing the value in the 1x1 resultset generated by
the command</returns>
    public static object ExecuteScalar(SqlConnection connection, CommandType
commandType, string commandText)
    {
        //pass through the call providing null for the set of SqlParameter
return ExecuteScalar(connection, commandType, commandText,
(SqlParameter[])null);
    }

    /// <summary>
    /// Execute a SqlCommand (that returns a 1x1 resultset) against the specified
SqlConnection
    /// using the provided parameters.
    /// </summary>
    /// <remarks>
    /// e.g.:

```

```

        /// int orderCount = (int)ExecuteScalar(conn,
CommandType.StoredProcedure, "GetOrderCount", new SqlParameter("@prodid", 24));
        /// </remarks>
        /// <param name="connection">a valid SqlConnection</param>
        /// <param name="commandType">the CommandType (stored procedure,
text, etc.)</param>
        /// <param name="commandText">the stored procedure name or T-SQL
command</param>
        /// <param name="commandParameters">an array of SqlParameter used to
execute the command</param>
        /// <returns>an object containing the value in the 1x1 resultset generated by
the command</returns>
        public static object ExecuteScalar(SqlConnection connection, CommandType
commandType, string commandText, params SqlParameter[] commandParameters)
        {
            //create a command and prepare it for execution
            SqlCommand cmd = new SqlCommand();
            PrepareCommand(cmd, connection, (SqlConnection)null,
commandType, commandText, commandParameters);

            //execute the command & return the results
            object retval = cmd.ExecuteScalar();

            // detach the SqlParameter from the command object, so they can be
used again.
            cmd.Parameters.Clear();
            return retval;
        }

        /// <summary>
        /// Execute a stored procedure via a SqlCommand (that returns a 1x1
resultset) against the specified SqlConnection
        /// using the provided parameter values. This method will query the database
to discover the parameters for the
        /// stored procedure (the first time each stored procedure is called), and
assign the values based on parameter order.
        /// </summary>
        /// <remarks>
        /// This method provides no access to output parameters or the stored
procedure's return value parameter.
        ///
        /// e.g.:
        /// int orderCount = (int)ExecuteScalar(conn, "GetOrderCount", 24, 36);
        /// </remarks>
        /// <param name="connection">a valid SqlConnection</param>
        /// <param name="spName">the name of the stored procedure</param>
        /// <param name="parameterValues">an array of objects to be assigned as
the input values of the stored procedure</param>
        /// <returns>an object containing the value in the 1x1 resultset generated by
the command</returns>
        public static object ExecuteScalar(SqlConnection connection, string spName,
params object[] parameterValues)
        {
            //if we receive parameter values, we need to figure out where they go

```



```

        if ((parameterValues != null) && (parameterValues.Length > 0))
        {
            //pull the parameters for this stored procedure from the
parameter cache (or discover them & populate the cache)
            SqlParameter[] commandParameters =
SqlHelperParameterCache.GetSpParameterSet(connection.ConnectionString, spName);

            //assign the provided values to these parameters based on
parameter order
            AssignParameterValues(commandParameters,
parameterValues);

            //call the overload that takes an array of SqlParameter
return ExecuteScalar(connection,
CommandType.StoredProcedure, spName, commandParameters);
        }
        //otherwise we can just call the SP without params
        else
        {
            return ExecuteScalar(connection,
CommandType.StoredProcedure, spName);
        }
    }

    /// <summary>
    /// Execute a SqlCommand (that returns a 1x1 resultset and takes no
parameters) against the provided SqlTransaction.
    /// </summary>
    /// <remarks>
    /// e.g.:
    /// int orderCount = (int)ExecuteScalar(trans,
CommandType.StoredProcedure, "GetOrderCount");
    /// </remarks>
    /// <param name="transaction">a valid SqlTransaction</param>
    /// <param name="commandType">the CommandType (stored procedure,
text, etc.)</param>
    /// <param name="commandText">the stored procedure name or T-SQL
command</param>
    /// <returns>an object containing the value in the 1x1 resultset generated by
the command</returns>
    public static object ExecuteScalar(SqlTransaction transaction, CommandType
commandType, string commandText)
    {
        //pass through the call providing null for the set of SqlParameter
return ExecuteScalar(transaction, commandType, commandText,
(SqlParameter[])null);
    }

    /// <summary>
    /// Execute a SqlCommand (that returns a 1x1 resultset) against the specified
SqlTransaction
    /// using the provided parameters.
    /// </summary>
    /// <remarks>
    /// e.g.:

```

```

        /// int orderCount = (int)ExecuteScalar(trans,
CommandType.StoredProcedure, "GetOrderCount", new SqlParameter("@prodid", 24));
        /// </remarks>
        /// <param name="transaction">a valid SqlTransaction</param>
        /// <param name="commandType">the CommandType (stored procedure,
text, etc.)</param>
        /// <param name="commandText">the stored procedure name or T-SQL
command</param>
        /// <param name="commandParameters">an array of SqlParameter used to
execute the command</param>
        /// <returns>an object containing the value in the 1x1 resultset generated by
the command</returns>
        public static object ExecuteScalar(SqlTransaction transaction, CommandType
commandType, string commandText, params SqlParameter[] commandParameters)
        {
            //create a command and prepare it for execution
            SqlCommand cmd = new SqlCommand();
            PrepareCommand(cmd, transaction.Connection, transaction,
commandType, commandText, commandParameters);

            //execute the command & return the results
            object retval = cmd.ExecuteScalar();

            // detach the SqlParameter from the command object, so they can be
used again.
            cmd.Parameters.Clear();
            return retval;
        }

        /// <summary>
        /// Execute a stored procedure via a SqlCommand (that returns a 1x1
resultset) against the specified
        /// SqlTransaction using the provided parameter values. This method will
query the database to discover the parameters for the
        /// stored procedure (the first time each stored procedure is called), and
assign the values based on parameter order.
        /// </summary>
        /// <remarks>
        /// This method provides no access to output parameters or the stored
procedure's return value parameter.
        ///
        /// e.g.:
        /// int orderCount = (int)ExecuteScalar(trans, "GetOrderCount", 24, 36);
        /// </remarks>
        /// <param name="transaction">a valid SqlTransaction</param>
        /// <param name="spName">the name of the stored procedure</param>
        /// <param name="parameterValues">an array of objects to be assigned as
the input values of the stored procedure</param>
        /// <returns>an object containing the value in the 1x1 resultset generated by
the command</returns>
        public static object ExecuteScalar(SqlTransaction transaction, string spName,
params object[] parameterValues)
        {
            //if we receive parameter values, we need to figure out where they go
            if ((parameterValues != null) && (parameterValues.Length > 0))

```

```

        {
            //pull the parameters for this stored procedure from the
parameter cache (or discover them & populate the cache)
            SqlParameter[] commandParameters =
SqlHelperParameterCache.GetSpParameterSet(transaction.Connection.ConnectionString,
spName);

            //assign the provided values to these parameters based on
parameter order
            AssignParameterValues(commandParameters,
parameterValues);

            //call the overload that takes an array of SqlParameter
return ExecuteScalar(transaction,
CommandType.StoredProcedure, spName, commandParameters);
        }
        //otherwise we can just call the SP without params
        else
        {
            return ExecuteScalar(transaction,
CommandType.StoredProcedure, spName);
        }
    }

#endregion ExecuteScalar

#region ExecuteXmlReader

    /// <summary>
    /// Execute a SqlCommand (that returns a resultset and takes no parameters)
against the provided SqlConnection.
    /// </summary>
    /// <remarks>
    /// e.g.:
    /// XmlReader r = ExecuteXmlReader(conn,
CommandType.StoredProcedure, "GetOrders");
    /// </remarks>
    /// <param name="connection">a valid SqlConnection</param>
    /// <param name="commandType">the CommandType (stored procedure,
text, etc.)</param>
    /// <param name="commandText">the stored procedure name or T-SQL
command using "FOR XML AUTO"</param>
    /// <returns>an XmlReader containing the resultset generated by the
command</returns>
    public static XmlReader ExecuteXmlReader(SqlConnection connection,
CommandType commandType, string commandText)
    {
        //pass through the call providing null for the set of SqlParameter
return ExecuteXmlReader(connection, commandType, commandText,
(SqlParameter[])null);
    }

    /// <summary>
    /// Execute a SqlCommand (that returns a resultset) against the specified
SqlConnection

```

```

        /// using the provided parameters.
        /// </summary>
        /// <remarks>
        /// e.g.:
        /// XmlReader r = ExecuteXmlReader(conn,
CommandType.StoredProcedure, "GetOrders", new SqlParameter("@prodid", 24));
        /// </remarks>
        /// <param name="connection">a valid SqlConnection</param>
        /// <param name="commandType">the CommandType (stored procedure,
text, etc.)</param>
        /// <param name="commandText">the stored procedure name or T-SQL
command using "FOR XML AUTO"</param>
        /// <param name="commandParameters">an array of SqlParameter used to
execute the command</param>
        /// <returns>an XmlReader containing the resultset generated by the
command</returns>
        public static XmlReader ExecuteXmlReader(SqlConnection connection,
CommandType commandType, string commandText, params SqlParameter[]
commandParameters)
        {
            //create a command and prepare it for execution
            SqlCommand cmd = new SqlCommand();
            PrepareCommand(cmd, connection, (SqlTransaction)null,
commandType, commandText, commandParameters);

            //create the DataAdapter & DataSet
            XmlReader retval = cmd.ExecuteXmlReader();

            // detach the SqlParameter from the command object, so they can be
used again.
            cmd.Parameters.Clear();
            return retval;
        }

        /// <summary>
        /// Execute a stored procedure via a SqlCommand (that returns a resultset)
against the specified SqlConnection
        /// using the provided parameter values. This method will query the database
to discover the parameters for the
        /// stored procedure (the first time each stored procedure is called), and
assign the values based on parameter order.
        /// </summary>
        /// <remarks>
        /// This method provides no access to output parameters or the stored
procedure's return value parameter.
        ///
        /// e.g.:
        /// XmlReader r = ExecuteXmlReader(conn, "GetOrders", 24, 36);
        /// </remarks>
        /// <param name="connection">a valid SqlConnection</param>
        /// <param name="spName">the name of the stored procedure using "FOR
XML AUTO"</param>
        /// <param name="parameterValues">an array of objects to be assigned as
the input values of the stored procedure</param>

```

```

        /// <returns>an XmlReader containing the resultset generated by the
command</returns>
        public static XmlReader ExecuteXmlReader(SqlConnection connection, string
spName, params object[] parameterValues)
        {
            //if we receive parameter values, we need to figure out where they go
            if ((parameterValues != null) && (parameterValues.Length > 0))
            {
                //pull the parameters for this stored procedure from the
parameter cache (or discover them & populate the cache)
                SqlParameter[] commandParameters =
SqlHelper.ParameterCache.GetSpParameterSet(connection.ConnectionString, spName);

                //assign the provided values to these parameters based on
parameter order
                AssignParameterValues(commandParameters,
parameterValues);

                //call the overload that takes an array of SqlParameter
return ExecuteXmlReader(connection,
CommandType.StoredProcedure, spName, commandParameters);
            }
            //otherwise we can just call the SP without params
            else
            {
                return ExecuteXmlReader(connection,
CommandType.StoredProcedure, spName);
            }
        }

        /// <summary>
        /// Execute a SqlCommand (that returns a resultset and takes no parameters)
against the provided SqlTransaction.
        /// </summary>
        /// <remarks>
        /// e.g.:
        /// XmlReader r = ExecuteXmlReader(trans,
CommandType.StoredProcedure, "GetOrders");
        /// </remarks>
        /// <param name="transaction">a valid SqlTransaction</param>
        /// <param name="commandType">the CommandType (stored procedure,
text, etc.)</param>
        /// <param name="commandText">the stored procedure name or T-SQL
command using "FOR XML AUTO"</param>
        /// <returns>an XmlReader containing the resultset generated by the
command</returns>
        public static XmlReader ExecuteXmlReader(SqlTransaction transaction,
CommandType commandType, string commandText)
        {
            //pass through the call providing null for the set of SqlParameter
return ExecuteXmlReader(transaction, commandType, commandText,
(SqlParameter[])null);
        }

        /// <summary>

```

```

        /// Execute a SqlCommand (that returns a resultset) against the specified
SqlTransaction
        /// using the provided parameters.
        /// </summary>
        /// <remarks>
        /// e.g.:
        /// XmlReader r = ExecuteXmlReader(trans,
CommandType.StoredProcedure, "GetOrders", new SqlParameter("@prodid", 24));
        /// </remarks>
        /// <param name="transaction">a valid SqlTransaction</param>
        /// <param name="commandType">the CommandType (stored procedure,
text, etc.)</param>
        /// <param name="commandText">the stored procedure name or T-SQL
command using "FOR XML AUTO"</param>
        /// <param name="commandParameters">an array of SqlParameter used to
execute the command</param>
        /// <returns>an XmlReader containing the resultset generated by the
command</returns>
        public static XmlReader ExecuteXmlReader(SqlTransaction transaction,
CommandType commandType, string commandText, params SqlParameter[]
commandParameters)
        {
            //create a command and prepare it for execution
            SqlCommand cmd = new SqlCommand();
            PrepareCommand(cmd, transaction.Connection, transaction,
commandType, commandText, commandParameters);

            //create the DataAdapter & DataSet
            XmlReader retval = cmd.ExecuteXmlReader();

            // detach the SqlParameter from the command object, so they can be
used again.
            cmd.Parameters.Clear();
            return retval;
        }

        /// <summary>
        /// Execute a stored procedure via a SqlCommand (that returns a resultset)
against the specified
        /// SqlTransaction using the provided parameter values. This method will
query the database to discover the parameters for the
        /// stored procedure (the first time each stored procedure is called), and
assign the values based on parameter order.
        /// </summary>
        /// <remarks>
        /// This method provides no access to output parameters or the stored
procedure's return value parameter.
        ///
        /// e.g.:
        /// XmlReader r = ExecuteXmlReader(trans, "GetOrders", 24, 36);
        /// </remarks>
        /// <param name="transaction">a valid SqlTransaction</param>
        /// <param name="spName">the name of the stored procedure</param>
        /// <param name="parameterValues">an array of objects to be assigned as
the input values of the stored procedure</param>

```

```

        /// <returns>a dataset containing the resultset generated by the
command</returns>
        public static XmlReader ExecuteXmlReader(SqlTransaction transaction,
string spName, params object[] parameterValues)
        {
            //if we receive parameter values, we need to figure out where they go
            if ((parameterValues != null) && (parameterValues.Length > 0))
            {
                //pull the parameters for this stored procedure from the
parameter cache (or discover them & populate the cache)
                SqlParameter[] commandParameters =
SqlHelperParameterCache.GetSpParameterSet(transaction.Connection.ConnectionString,
spName);

                //assign the provided values to these parameters based on
parameter order
                AssignParameterValues(commandParameters,
parameterValues);

                //call the overload that takes an array of SqlParameter
return ExecuteXmlReader(transaction,
CommandType.StoredProcedure, spName, commandParameters);
            }
            //otherwise we can just call the SP without params
            else
            {
                return ExecuteXmlReader(transaction,
CommandType.StoredProcedure, spName);
            }
        }

        #endregion ExecuteXmlReader
    }

    /// <summary>
    /// SqlHelperParameterCache provides functions to leverage a static cache of
procedure parameters, and the
    /// ability to discover parameters for stored procedures at run-time.
    /// </summary>
    public sealed class SqlHelperParameterCache
    {
        #region private methods, variables, and constructors

        //Since this class provides only static methods, make the default constructor
private to prevent
        //instances from being created with "new SqlHelperParameterCache()".
        private SqlHelperParameterCache() {}

        private static Hashtable paramCache = Hashtable.Synchronized(new
Hashtable());

        /// <summary>
        /// resolve at run time the appropriate set of SqlParameter for a stored
procedure

```

```

        /// </summary>
        /// <param name="connectionString">a valid connection string for a
SqlConnection</param>
        /// <param name="spName">the name of the stored procedure</param>
        /// <param name="includeReturnValueParameter">whether or not to include
their return value parameter</param>
        /// <returns></returns>
        private static SqlParameter[] DiscoverSpParameterSet(string
connectionString, string spName, bool includeReturnValueParameter)
        {
            using (SqlConnection cn = new SqlConnection(connectionString))
            using (SqlCommand cmd = new SqlCommand(spName,cn))
            {
                cn.Open();
                cmd.CommandType = CommandType.StoredProcedure;

                SqlCommandBuilder.DeriveParameters(cmd);

                if (!includeReturnValueParameter)
                {
                    cmd.Parameters.RemoveAt(0);
                }

                SqlParameter[] discoveredParameters = new
SqlParameter[cmd.Parameters.Count];;

                cmd.Parameters.CopyTo(discoveredParameters, 0);

                return discoveredParameters;
            }
        }

        //deep copy of cached SqlParameter array
        private static SqlParameter[] CloneParameters(SqlParameter[]
originalParameters)
        {
            SqlParameter[] clonedParameters = new
SqlParameter[originalParameters.Length];

            for (int i = 0, j = originalParameters.Length; i < j; i++)
            {
                clonedParameters[i] =
(SqlParameter)((ICloneable)originalParameters[i]).Clone();
            }

            return clonedParameters;
        }

#endregion private methods, variables, and constructors

#region caching functions

        /// <summary>
        /// add parameter array to the cache
        /// </summary>

```



```

        /// <param name="connectionString">a valid connection string for a
SqlConnection</param>
        /// <param name="commandText">the stored procedure name or T-SQL
command</param>
        /// <param name="commandParameters">an array of SqlParameter to be
cached</param>
        public static void CacheParameterSet(string connectionString, string
commandText, params SqlParameter[] commandParameters)
        {
            string hashKey = connectionString + ":" + commandText;

            paramCache[hashKey] = commandParameters;
        }

        /// <summary>
        /// retrieve a parameter array from the cache
        /// </summary>
        /// <param name="connectionString">a valid connection string for a
SqlConnection</param>
        /// <param name="commandText">the stored procedure name or T-SQL
command</param>
        /// <returns>an array of SqlParameter</returns>
        public static SqlParameter[] GetCachedParameterSet(string connectionString,
string commandText)
        {
            string hashKey = connectionString + ":" + commandText;

            SqlParameter[] cachedParameters =
(SqlParameter[])paramCache[hashKey];

            if (cachedParameters == null)
            {
                return null;
            }
            else
            {
                return CloneParameters(cachedParameters);
            }
        }

#endregion caching functions

#region Parameter Discovery Functions

        /// <summary>
        /// Retrieves the set of SqlParameter appropriate for the stored procedure
        /// </summary>
        /// <remarks>
        /// This method will query the database for this information, and then store it in
a cache for future requests.
        /// </remarks>
        /// <param name="connectionString">a valid connection string for a
SqlConnection</param>
        /// <param name="spName">the name of the stored procedure</param>
        /// <returns>an array of SqlParameter</returns>

```

```

        public static SqlParameter[] GetSpParameterSet(string connectionString,
string spName)
        {
            return GetSpParameterSet(connectionString, spName, false);
        }

        /// <summary>
        /// Retrieves the set of SqlParameter objects appropriate for the stored procedure
        /// </summary>
        /// <remarks>
        /// This method will query the database for this information, and then store it in
a cache for future requests.
        /// </remarks>
        /// <param name="connectionString">a valid connection string for a
SqlConnection</param>
        /// <param name="spName">the name of the stored procedure</param>
        /// <param name="includeReturnValueParameter">a bool value indicating
whether the return value parameter should be included in the results</param>
        /// <returns>an array of SqlParameter objects</returns>
        public static SqlParameter[] GetSpParameterSet(string connectionString,
string spName, bool includeReturnValueParameter)
        {
            string hashKey = connectionString + ":" + spName +
(includeReturnValueParameter ? ":include ReturnValue Parameter":"" );

            SqlParameter[] cachedParameters;

            cachedParameters = (SqlParameter[])paramCache[hashKey];

            if (cachedParameters == null)
            {
                cachedParameters = (SqlParameter[])(paramCache[hashKey]
= DiscoverSpParameterSet(connectionString, spName, includeReturnValueParameter));
            }

            return CloneParameters(cachedParameters);
        }

        #endregion Parameter Discovery Functions
    }
}

```

This concludes DataTier

Chapter 5

Develop Business Classes

Business Layer classes can be downloaded from the following link in Dropbox

<https://www.dropbox.com/scl/fi/yig29kykigzmu1b3jio9j/DowBAL.rar?rlkey=k1f2t1xiynqqd4yg38a9259r9&st=puqzr6s8&dl=0>

The following classes are added to the business layer:

```
using System.Collections.Generic;
using System.Web.UI.WebControls;
using System;
using System.Linq;
using DowBEntity;
using DowDAL;
```

```
namespace DowBAL
{
    public class PopDlIs
    {
        public void BindDropDowns(DropDownList ddl, List<PopDDL> lst)
        {
            try
            {
                ddl.Items.Clear();
                ddl.DataTextField = "Text";
                ddl.DataValueField = "Value";
                ddl.DataSource = lst;
                ddl.DataBind();
            }
            catch (Exception)
            {
            }
        }
        public void BindStaticDropDowns(DropDownList ddl)
        {
            try
            {
                ddl.Items.Clear();
                ddl.Items.Add(new ListItem("Direct Debit", "D"));
                ddl.Items.Add(new ListItem("Bacs", "B"));
                ddl.Items.Add(new ListItem("Cheque on Invoice", "C"));
            }
            catch (Exception)
            {
            }
        }
        public void BindStaticTech(DropDownList ddl)
        {
            try
            {
                ddl.Items.Clear();
```

```

        ddl.Items.Add(new ListItem("WLR3 Only", "W"));
        ddl.Items.Add(new ListItem("LLU Only", "L"));

    }
    catch (Exception)
    {

    }
}
public List<PopDDL> GetExtProp()
{
    List<PopDDL> result = new List<PopDDL>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        var list = (from c in db.productExtProperties select new { c.EpId, c.EpDescription
    }).ToList();
        result.Add(new PopDDL { Text = "-- Select an Option --", Value = "0" });

        foreach (var row in list)
        {
            result.Add(new PopDDL { Text = row.EpDescription, Value =
row.EpId.ToString() });
        }
    }

    return result;
}
public List<PopDDL> GetExtOpt(int epid)
{
    List<PopDDL> result = new List<PopDDL>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        var list = (from c in db.productExtPrOptions where c.EpId==epid select new {
c.EpOptId, c.EpOPtDescr }).ToList();
        // result.Add(new PopDDL { Text = "-- Select an Option --", Value = "0" });

        foreach (var row in list)
        {
            result.Add(new PopDDL { Text = row.EpOPtDescr, Value =
row.EpOPtDescr.ToString() });
        }
    }

    return result;
}
public List<PopDDL> GetVatCodes()
{
    List<PopDDL> result = new List<PopDDL>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        var list = (from c in db.VatCodes select new { c.Id, c.Description }).ToList();
        // result.Add(new PopDDL { Text = "-- Select an Option --", Value = "0" });
    }
}

```

```

        foreach (var row in list)
        {
            result.Add(new PopDDL { Text = row.Description, Value = row.Id.ToString() });
        }
    }

    return result;
}
public List<PopDDL> GetYesNo()
{
    List<PopDDL> result = new List<PopDDL>();
    result.Add(new PopDDL { Text = " ", Value = "C" });
    result.Add(new PopDDL { Text = "Yes", Value = "Y" });
    result.Add(new PopDDL { Text = "No", Value = "N" });

    return result;
}

public List<PopDDL> GetUser()
{
    List<PopDDL> result = new List<PopDDL>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        var list = (from c in db.tblUsers where c.IsRemoved == false select new { c.UserId,
c.UserName }).ToList();
        result.Add(new PopDDL { Text = "", Value = "0" });

        foreach (var row in list)
        {
            result.Add(new PopDDL { Text = row.UserName, Value = row.UserId.ToString()
});
        }
        result.Add(new PopDDL { Text = " ", Value = null });
    }

    return result;
}

public List<PopDDL> GetUserCompany()
{
    List<PopDDL> result = new List<PopDDL>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        var list = (from c in db.tblUsers where c.IsRemoved == false select new { c.UserId,
c.LastName }).ToList();
        result.Add(new PopDDL { Text = "", Value = "0" });

        foreach (var row in list)
        {

```

```

        result.Add(new PopDDL { Text = row.LastName, Value = row.UserId.ToString()
});
    }
    result.Add(new PopDDL { Text = " ", Value = null });
}

return result;
}
public List<PopDDL> GetAddType()
{
    List<PopDDL> result = new List<PopDDL>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        var list = (from c in db.tbluser_add_types select new { c.adid, c.adtype }).ToList();
        result.Add(new PopDDL { Text = "", Value = "0" });

        foreach (var row in list)
        {
            result.Add(new PopDDL { Text = row.adtype, Value = row.adid.ToString() });
        }
        result.Add(new PopDDL { Text = " ", Value = null });
    }

return result;
}

public List<PopDDL> GetNoteType()
{
    List<PopDDL> result = new List<PopDDL>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        var list = (from c in db.NoteTypes where c.IsActive == true select new { c.Id,
c.Name }).ToList();

        foreach (var row in list)
        {
            result.Add(new PopDDL { Text = row.Name, Value = row.Id.ToString() });
        }
    }

return result;
}
public List<PopDDL> GetTaskStatus()

```

```

    {
        List<PopDDL> result = new List<PopDDL>();
        using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        {
            var list = (from c in db.TaskStatus select new { c.Id, c.TStatusName }).ToList();

            foreach (var row in list)
            {
                result.Add(new PopDDL { Text = row.TStatusName, Value = row.Id.ToString()
});
            }
        }

        return result;
    }
    public List<PopDDL> GetQueue()
    {
        List<PopDDL> result = new List<PopDDL>();
        using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        {
            result.Add(new PopDDL { Text = "", Value = "0" });
            var list = (from c in db.ActionQueues where c.IsActive == true select new { c.Id,
c.Name }).ToList();

            foreach (var row in list)
            {
                result.Add(new PopDDL { Text = row.Name, Value = row.Id.ToString() });
            }
        }

        return result;
    }
    public List<PopDDL> GetCountries()
    {
        List<PopDDL> result = new List<PopDDL>();
        using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        {
            result.Add(new PopDDL { Text = "", Value = "0" });
            var list = (from c in db.countries where c.IsBlocked == false select new { c.id,
c.countryname }).ToList();
            result.Add(new PopDDL { Text = "-- Select a Country --", Value = "0" });

            foreach (var row in list)
            {
                result.Add(new PopDDL { Text = row.countryname, Value = row.id.ToString() });
            }
        }
    }

```

```

        return result;
    }
    public List<PopDDL> GetproductGroupbyServices(int cat)
    {
        List<PopDDL> result = new List<PopDDL>();
        using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        {
            var list = (from c in db.ProductGroups join d in db.ProductCats on c.PCatId equals
d.PCatId where d.Stype == cat select new { c.pgid, c.pgroupname }).ToList();
            result.Add(new PopDDL { Text = "-- Select a Product Group --", Value = "0" });

            foreach (var row in list)
            {
                result.Add(new PopDDL { Text = row.pgroupname, Value = row.pgid.ToString()
});
            }
        }

        return result;
    }
    public List<PopDDL> GetproductGroupbyCat(int cat)
    {
        List<PopDDL> result = new List<PopDDL>();
        using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        {
            var list = (from c in db.ProductGroups where c.PCatId==cat select new { c.pgid,
c.pgroupname }).ToList();
            result.Add(new PopDDL { Text = "-- Select a Product Group --", Value = "0" });

            foreach (var row in list)
            {
                result.Add(new PopDDL { Text = row.pgroupname, Value = row.pgid.ToString()
});
            }
        }

        return result;
    }
    public List<PopDDL> GetproductGroups()
    {
        List<PopDDL> result = new List<PopDDL>();
        using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        {
            var list = (from c in db.ProductGroups select new { c.pgid, c.pgroupname
}).ToList();
            result.Add(new PopDDL { Text = "-- Select a Product Group --", Value = "0" });

            foreach (var row in list)
            {
                result.Add(new PopDDL { Text = row.pgroupname, Value = row.pgid.ToString()
});
            }
        }
    }

```



```

    }
}

return result;
}
public List<PopDDL> GetStores()
{
    List<PopDDL> result = new List<PopDDL>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        var list = (from c in db.ProductCats where c.Stype==1 select new { c.PCatId,
c.PcatDesc }).ToList();
        result.Add(new PopDDL { Text = "-- Select a Store --", Value = "0" });

        foreach (var row in list)
        {
            result.Add(new PopDDL { Text = row.PcatDesc, Value = row.PCatId.ToString()
});
        }
    }

    return result;
}

public List<PopDDL> GetProductCats()
{
    List<PopDDL> result = new List<PopDDL>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        var list = (from c in db.ProductCats select new { c.PCatId, c.PcatDesc }).ToList();
        result.Add(new PopDDL { Text = "-- Select a Product Category --", Value = "0" });

        foreach (var row in list)
        {
            result.Add(new PopDDL { Text = row.PcatDesc, Value = row.PCatId.ToString()
});
        }
    }

    return result;
}

public List<PopDDL> GetProductCatbyType(int id)
{
    List<PopDDL> result = new List<PopDDL>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

```

```

        var list = (from c in db.ProductCats where c.Stype==id select new { c.PCatId,
c.PcatDesc }).ToList();
        result.Add(new PopDDL { Text = "-- Select a Product Category --", Value = "0" });

        foreach (var row in list)
        {
            result.Add(new PopDDL { Text = row.PcatDesc, Value = row.PCatId.ToString()
});
        }
    }
}

```

```

return result;
}
public List<PopDDL> GetProductbyUseridType(int UserId,int type)
{
    List<PopDDL> result = new List<PopDDL>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        List<EteSellerProduct> actpr = new Products().GetSellerProducts(UserId, type);
        var list = (from c in actpr select new { c.ProductId, c.ProductDesc });
        //var list = (from c in db.Products join d in db.SellersProducts on c.ProductId
equals d.productid select new { c.ProductId, c.ProductName }).ToList();
        if (type == 1)
        {
            result.Add(new PopDDL { Text = "-- Select a Product --", Value = "0" });
        }
        else
        {
            result.Add(new PopDDL { Text = "-- Select a Service --", Value = "0" });
        }
        foreach (var row in list)
        {
            result.Add(new PopDDL { Text = row.ProductDesc.ToString(), Value =
row.ProductId.ToString() });
        }
    }
}

```

```

return result;
}
public List<PopDDL> GetProductbyUserid(int UserId)
{
    List<PopDDL> result = new List<PopDDL>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        var list = (from c in db.Products join d in db.SellersProducts on c.ProductId equals
d.productid select new { c.ProductId, c.ProductName }).ToList();
        result.Add(new PopDDL { Text = "-- Select a Product --", Value = "0" });
    }
}

```

```

        foreach (var row in list)
        {
            result.Add(new PopDDL { Text = row.ProductName.ToString(), Value =
row.ProductId.ToString() });
        }
    }

    return result;
}
public List<PopDDL> GetProduct()
{
    List<PopDDL> result = new List<PopDDL>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        var list = (from c in db.Products select new { c.ProductId, c.ProductName
}).ToList();
        result.Add(new PopDDL { Text = "-- Select a Product --", Value = "0" });

        foreach (var row in list)
        {
            result.Add(new PopDDL { Text = row.ProductName.ToString(), Value =
row.ProductId.ToString() });
        }
    }

    return result;
}
public List<PopDDL> Getproductsbycat(int id)
{
    List<PopDDL> result = new List<PopDDL>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        var list = (from c in db.Products where c.ProductGroupCategory==id select new {
c.ProductId, c.ProductName }).ToList();
        result.Add(new PopDDL { Text = "-- Select an Option --", Value = "0" });

        foreach (var row in list)
        {
            result.Add(new PopDDL { Text = row.ProductName, Value =
row.ProductId.ToString() });
        }
    }

    return result;
}
public List<PopDDL> Getproductsbygroup(int id)
{
    List<PopDDL> result = new List<PopDDL>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))

```

```

    {
        var list = (from c in db.Products where c.ProductGroupId == id select new {
c.ProductId, c.ProductName }).ToList();
        result.Add(new PopDDL { Text = "-- Select an Option --", Value = "0" });

        foreach (var row in list)
        {
            result.Add(new PopDDL { Text = row.ProductName, Value =
row.ProductId.ToString() });
        }
    }

    return result;
}
public List<PopDDL> Gettbluom()
{
    List<PopDDL> result = new List<PopDDL>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        var list = (from c in db.tbluoms select new { c.UomId, c.UomShort }).ToList();
        result.Add(new PopDDL { Text = "-- Select an UOM --", Value = "0" });

        foreach (var row in list)
        {
            result.Add(new PopDDL { Text = row.UomShort, Value = row.UomId.ToString()
});
        }
    }

    return result;
}
public List<PopDDL> GetLegalstr()
{
    List<PopDDL> result = new List<PopDDL>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        var list = (from c in db.Legal_structures select new { c.Isid, c.Isdescription
}).ToList();
        result.Add(new PopDDL { Text = "-- Select an Item --", Value = "0" });

        foreach (var row in list)
        {
            result.Add(new PopDDL { Text = row.Isdescription, Value = row.Isid.ToString()
});
        }
    }

    return result;
}
public List<PopDDL> GetBusinessType()
{
    List<PopDDL> result = new List<PopDDL>();

```

```

        using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        {
            var list = (from c in db.Business_Types select new { c.BTId, c.BTDescription
        }).ToList();
            result.Add(new PopDDL { Text = "-- Select a Type --", Value = "0" });

            foreach (var row in list)
            {
                result.Add(new PopDDL { Text = row.BTDescription, Value =
row.BTId.ToString() });
            }

            return result;
        }
        public List<PopDDL> Getdeltimes()
        {
            List<PopDDL> result = new List<PopDDL>();
            using (var db = new DowPortalDataContext(Configuration.ConnectionString))
            {
                var list = (from c in db.tblDeltimes select new { c.DelTimeType,
c.DelTimeDescription }).ToList();
                result.Add(new PopDDL { Text = "-- Select a Timeframe --", Value = "0" });

                foreach (var row in list)
                {
                    result.Add(new PopDDL { Text = row.DelTimeDescription, Value =
row.DelTimeType.ToString() });
                }

                return result;
            }
        }
        public List<PopDDL> Gettbltblpacking()
        {
            List<PopDDL> result = new List<PopDDL>();
            using (var db = new DowPortalDataContext(Configuration.ConnectionString))
            {
                var list = (from c in db.tblpackings select new { c.PackID, c.PackShort }).ToList();
                result.Add(new PopDDL { Text = "-- Select a Packing --", Value = "0" });

                foreach (var row in list)
                {
                    result.Add(new PopDDL { Text = row.PackShort, Value = row.PackID.ToString()
        });
                }

                return result;
            }
        }
        public List<PopDDL> GetRoles(int roleid)
        {

```

```

List<PopDDL> result = new List<PopDDL>();
using (var db = new DowPortalDataContext(Configuration.ConnectionString))
{
    var list = (from c in db.tblRoles where c.RoleId >= roleid select new { c.RoleId,
c.RoleType }).ToList();

    foreach (var row in list)
    {
        result.Add(new PopDDL { Text = row.RoleType, Value = row.RoleId.ToString()
});
    }
}

return result;
}

public List<PopDDL> GetBrands()
{
    List<PopDDL> result = new List<PopDDL>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        var list = (from c in db.Brands select new { c.Id, c.BrandName }).ToList();

        foreach (var row in list)
        {
            result.Add(new PopDDL { Text = row.BrandName, Value = row.Id.ToString() });
        }
    }

    return result;
}

public List<PopDDL> GetCurrency()
{
    List<PopDDL> result = new List<PopDDL>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        var list = (from c in db.Currencies select new { c.CId, c.CDescription }).ToList();

        foreach (var row in list)
        {
            result.Add(new PopDDL { Text = row.CDescription, Value = row.CId.ToString()
});
        }
    }

    return result;
}

```

```

}
public List<PopDDL> GetGroups(int UserId, int GroupId)
{
    List<PopDDL> result = new List<PopDDL>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        if (GroupId == 1)
        {
            var list = (from c in db.tblPermissionGroups where c.GroupId > 1 select new {
c.GroupId, c.GroupName }).ToList();
            foreach (var row in list)
            {
                result.Add(new PopDDL { Text = row.GroupName, Value =
row.GroupId.ToString() });
            }
        }
        else
        {
            var list = (from c in db.tblPermissionGroups where c.GroupId > 1 && c.UserId
== UserId || (c.GroupId >= GroupId && c.UserId == 1) select new { c.GroupId, c.GroupName
}).ToList();

            foreach (var row in list)
            {
                result.Add(new PopDDL { Text = row.GroupName, Value =
row.GroupId.ToString() });
            }
        }
    }

    return result;
}
public List<PopDDL> GetDepts()
{
    List<PopDDL> result = new List<PopDDL>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        var list = (from c in db.Departments select new { c.DeptId, c.DeptName }).ToList();

        foreach (var row in list)
        {
            result.Add(new PopDDL { Text = row.DeptName, Value = row.DeptId.ToString()
});
        }
    }

    return result;
}
public List<PopDDL> GetDealers()
{
    List<PopDDL> result = new List<PopDDL>();

```

```

        using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        {
            var list = (from c in db.Dealers /*where c.IsActive==true*/ select new { c.DealerId,
c.CompanyName }).ToList();

            foreach (var row in list)
            {
                result.Add(new PopDDL { Text = row.CompanyName, Value =
row.DealerId.ToString() });
            }
        }

        return result;
    }
    public List<PopDDL> GetCancelReason()
    {
        List<PopDDL> result = new List<PopDDL>();
        using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        {
            var list = (from c in db.CancellationReasons select new { c.Id, c.Name }).ToList();
            result.Add(new PopDDL { Text = "", Value = "0" });
            foreach (var row in list)
            {
                result.Add(new PopDDL { Text = row.Name, Value = row.Id.ToString() });
            }
        }

        return result;
    }
}
}

```

```

using System;
using System.IO;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using DowBEntity;
using DowDAL;
using System.Xml;
using System.Xml.Serialization;
using System.Xml.XmlConfiguration;

```

```

//
namespace DowBAL

```



```

{
public class Products
{
    DDLogTracer objlog = new DDLogTracer();

    public decimal GetNextProductId()
    {
        decimal result = 0;
        using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        {

            //GetNextProductIdResult lst = db.GetNextProductId().SingleOrDefault();
            //result =(decimal) lst.NextProductId;

        }

        return result;
    }
    public int GetNextProductGroupId()
    {
        int result = 0;
        using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        {

            //GetNextProductGroupIdResult lst =
db.GetNextProductGroupId().SingleOrDefault();
            //result =(int) lst.NextProductId;

        }

        return result;
    }
    public int GetNextProductCatId()
    {
        int result = 0;
        using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        {

            //GetNextProductCatIdResult lst = db.GetNextProductCatId().SingleOrDefault();
            //result = (int)lst.NextProductId;

        }

        return result;
    }
    public int GetProductCatIdfromGId(int Id)
    {
        int result = 0;
        using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        {

```

```

        result = (from c in db.ProductGroups where c.pgid == Id select
c.PCatId).SingleOrDefault();

    }

    return result;
}
public int GetNextSrNo(decimal PslId)
{

    int result = 1;
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        int dt = (from a in db.productSellerImages where a.PSID == PslId orderby a.srno
descending select a.srno).FirstOrDefault();
        result = dt + 1;

    }

    return result;
}
public decimal GetNextPSId()
{

    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        var dt = (from a in db.productSellers orderby a.PSID descending select
a.PSID).FirstOrDefault();
        if (dt > 0)
        {
            dt++;
        }
        else
        {
            dt = 1;
        }
        //var no = (from b in db.productSellerstocks where b.PSID == dt select
b.PSID).FirstOrDefault();
        return dt;
    }
}
public int GetNextUCId()
{

    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        var dt = (from a in db.UserCarts orderby a.UCId descending select
a.UCId).FirstOrDefault();
        if (dt > 0)
        {
            dt++;
        }
        else
        {

```

```

        dt = 1;
    }
    return dt;
}
}
public int GetCartItems(int UserID)
{
    int Ret = 0;
    //try
    {
        using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        {
            Ret = (from c in db.UserCarts where c.UserId == UserID && c.Active select
c).Count();

        }
    }
    //catch (Exception exc){
    //    objlog.Trace_Error(5, "MLB", "Bal.Products", "GetCartItems", exc.Message);

    //    throw exc;
    //}
    return Ret;

}
public int GetSLItems(int UserID, bool stype)
{
    int Ret = 0;
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        Ret = (from c in db.UserShortlists where c.UserId == UserID && c.Active &&
c.SLType==stype select c).Count();

    }
    return Ret;
}

public List<EteMyUserCart> GetCartbyUserId(int userid)
{
    List<EteMyUserCart> usercart = new List<EteMyUserCart>();
    //try
    {
        using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        {
            List<GetMyUserCartResult> user = db.GetMyUserCart(userid).ToList();
            foreach (var data in user)
            {
                if (data.CurrencyId == 0) { data.CurrencyId = 1; }
                string ShCurrency = (from c in db.Currencies where c.CId == data.CurrencyId
select c.CShortCode).SingleOrDefault();
                var ImagePath = (from d in db.productSellerImages where d.PSID ==
data.PSID select d.ImagePath).FirstOrDefault();
                if (ShCurrency == null) { ShCurrency = ""; }
                usercart.Add(new EteMyUserCart
                {

```



```

        PriceUomId = (int)data.PriceUomId,
        PricePackId = (int)data.PricePackId,
        createdon = data.createdon,
        UCId = data.PSId,
        PayId = 0,
        ShCurrency = ShCurrency,
        Qty = 1,
        Amt = (decimal)data.Price,
        ImagePath = ImagePath,
    });
}

}
}
//catch (Exception exc)
//{
//    objlog.Trace_Error(5, "MLB", "Bal.Products", "GetShortListbyUserId",
exc.Message);

//    throw exc;
//}
return usercart;
}
public bool DeleteShortList(decimal psid,int userid, bool stype)
{
    bool ret=false;
    List<EteMyUserCart> usercart = new List<EteMyUserCart>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        var user = (from c in db.UserShortlists where c.PSId == psid && c.UserId ==
userid && c.SLType == stype select c).ToList();
        foreach (var data in user)
        {
            data.Active = false;
        }
        db.SubmitChanges();
        db.Transaction.Commit();
        ret = true;
    }
    return ret;
}
public List<mapids> GetPerSetsForProduct(int GroupId, int type)
{
    List<mapids> lst = new List<mapids>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        //int istatus = 1;
        db.Connection.Open();
        List<ManageProductsResult> List = db.ManageProducts(GroupId, type).ToList();
        foreach (var row in List)
        {

```

```

    {
        //if (row.view == "true")
        //{
        //    istatus = 1;
        //}
        lst.Add(new mapids
        {
            mpageid = row.Pageld.ToString(),
            status = "1",
            buttonid = "",
            pageid = Convert.ToInt32(row.Pageld),
        });
    }
}

return lst;
}
public Eteproduct GetProduct(decimal id,string Curl)
{
    Eteproduct _user = new Eteproduct();
    try {
        using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        {

            db.Connection.Open();
            int pgid = (new tblPagess()).GetPageld(Curl);
            var data = (from c in db.Products where c.ProductId == id select
c).SingleOrDefault();

            _user.ProductId = data.ProductId;
            _user.ProductName=data.ProductName;
            _user.ProductGroupId=data.ProductGroupId;
            _user.ProductGroupCategory=(int)data.ProductGroupCategory;
            _user.Price=(decimal)data.Price;
            _user.CurrencyId=(int)data.CurrencyId;
            _user.AnalysisCode=data.AnalysisCode.ToString();
            _user.CreatedOn=(DateTime)data.CreatedOn;
            _user.UpdatedOn = (DateTime)data.UpdatedOn;
            _user.UserId=(int)data.UserId;
            //_user.PrId=data.PrId;
            _user.MinOrder=(int)data.MinOrder;
            _user.DelTlme=(int)data.DelTlme;
            _user.DealerId = data.DealerId;

            return _user;
        }
    }
    catch (Exception exc)

```

```

    {
        objlog.Trace_Error(5, "MLB", "Bal.Products", "GetProduct", exc.Message);
        throw exc;
    }
}
public List<EteproductImages> GetProductSellerImage(decimal id, string Curl)
{
    List<EteproductImages> _user = new List<EteproductImages>();
    try {
        using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        {
            db.Connection.Open();
            int pgid = (new tblPagess()).GetPagelId(Curl);
            var data = (from c in db.productSellerImages where c.PSID == id select
c).ToList();
            if (data.Count() > 0)
            {
                foreach (var pics in data)
                {
                    //byte[] bytes = (byte[])pics.Images.ToArray();
                    //string base64String = Convert.ToBase64String(bytes, 0, bytes.Length);
                    //string imageUrl = "data:image/png;base64," + base64String;

                    _user.Add(new EteproductImages
                    {
                        ProductId = pics.ProductId,
                        Images = pics.Images,
                        ImageUrls = pics.ImageURL,
                        ImagePath = pics.ImagePath,
                        srno = pics.srno,
                    });
                }
            }
        }
    }
    catch (Exception exc)
    {
        objlog.Trace_Error(5, "MLB", "Bal.Products", "GetProduct", exc.Message);
        throw exc;
    }
    return _user;
}
public EteproductSeller GetProductSeller(decimal id, string Curl)
{
    EteproductSeller _user = new EteproductSeller();
    try {
        using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        {
            db.Connection.Open();

```

```

        int pgid = (new tblPagess()).GetPageld(Curl);
        var data = (from c in db.productSellers where c.PSId == id select
c).SingleOrDefault();
        if (data.LinkPSId == null) { data.LinkPSId = 0; }
        _user.SellerDescription = data.SellerDescription;
        _user.PSId = data.PSId;
        _user.Price = (decimal)data.Price;
        _user.CurrencyId = (int)data.CurrencyId;
        _user.PriceUnits = (int)data.PriceUnits;
        _user.PriceUomId = (int)data.PriceUomId;
        _user.PricePackId = (int)data.PricePackId;
        _user.KeyWords = data.KeyWords;
        _user.CreatedOn = data.CreatedOn;
        _user.UpdatedOn = data.UpdatedOn;
        _user.UserId = data.UserId;
        _user.Prid = data.Prid;
        _user.ProductId = data.ProductId;
        _user.MinOrder = (int)data.MinOrder;
        _user.MinOrderUom = (int)data.MinOrderUom;
        _user.DelTlme = (int)data.DelTlme;
        _user.DelTimeType = (int)data.DelTimeType;
        _user.LinkPSId = (decimal)data.LinkPSId;
        _user.ProductDescription = data.ProductDescription;
        _user.DiscountLink = (int)data.DiscountLink;
        _user.BrandName = data.BrandName;
        _user.MaterialCare = data.MaterialCare;
        _user.MaterialType = data.MaterialType;
        _user.Manufacturer = data.Manufacturer;
        _user.Length = data.Length;
        _user.MadeinCountry = (int)data.MadeinCountry;
        _user.Location = data.Location;
        _user.GeoLocation = data.GeoLocation;

    }

    }
    catch (Exception exc)
    {
        objlog.Trace_Error(5, "MLB", "Bal.Products", "GetProduct", exc.Message);
        throw exc;
    }
    return _user;
}
public List<EteproductSeller> GetProductSellers(string Curl)
{
    List<EteproductSeller> _user = new List<EteproductSeller>();
    try {
        using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        {

            db.Connection.Open();
            int pgid = (new tblPagess()).GetPageld(Curl);
            var user = (from c in db.productSellers select c).ToList();

```



```

foreach (var data in user)
{
    _user.Add(new EteproductSeller
    {PSId=data.PSId,
      ProductId = data.ProductId,
      SellerId = data.SellerId,
      SellerDescription = data.SellerDescription,
      Price = (decimal)data.Price,
      CurrencyId = (int)data.CurrencyId,
      PriceUnits = (int)data.PriceUnits,
      PriceUomId = (int)data.PriceUomId,
      PricePackId = (int)data.PricePackId,
      KeyWords = data.KeyWords,
      CreatedOn = data.CreatedOn,
      UpdatedOn = data.UpdatedOn,
      UserId = data.UserId,
      Prid = data.Prid,
      MinOrder = (int)data.MinOrder,
      MinOrderUom = (int)data.MinOrderUom,
      DelTIme = (int)data.DelTIme,
      DelTimeType = (int)data.DelTimeType,
      Location=data.Location,
      GeoLocation=data.GeoLocation,

    });
}

}

}

catch (Exception exc)
{
    objlog.Trace_Error(5, "MLB", "Bal.Products", "GetProduct", exc.Message);
    throw exc;
}

return _user;
}

public List<Eteproductbuyers> GetProductBuyers(int Prid,int Groupid,string Curl)
{
    List<Eteproductbuyers> _user = new List<Eteproductbuyers>();
    try {
        using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        {
            string RedirectStr = "";
            db.Connection.Open();
            int pgid = (new tblPagess()).GetPageId(Curl);
            if (Groupid == 3)
            {
                if (Prid == 1)
                {
                    RedirectStr =
                    "~/ManageProductsource.aspx?pageid=28&PageType=BuyQuote&source=Home&PSId=";

```

```

    }
    else
    {
        RedirectStr =
"~/ManageProductsourced.aspx?pageid=28&PageType=SellQuote&source=Home&PSId=";

    }
}
else
{
    if (Groupid != 3)
    {
        if (Prid == 1)
        {
            RedirectStr =
"~/ManageProductsource.aspx?pageid=27&PageType=Buy&source=Home&PSId=";
        }
        else
        {
            RedirectStr =
"~/ManageProductsourced.aspx?pageid=27&PageType=Sell&source=Home&PSId=";
        }
    }
}
}
var user = (from c in db.productSellers where c.Prid == Prid orderby c.CreatedOn
select c).ToList();
foreach (var data in user)
{
    RedirectStr = RedirectStr+data.PSId.ToString();
    string currency = (from a in db.Currencies where a.CId == data.CurrencyId
select a.CDescription).SingleOrDefault();
    string priceuom = (from b in db.tbluoms where b.UomId == data.PriceUomId
select b.UomShort).SingleOrDefault();
    string pricepack = (from d in db.tblpackings where d.PackID ==
data.PricePackId select d.PackShort).SingleOrDefault();
    string minuom = (from e in db.tbluoms where e.UomId == data.MinOrderUom
select e.UomShort).SingleOrDefault();
    string Pack = (from f in db.tblpackings where f.PackID == data.PricePackId
select f.PackShort).SingleOrDefault();
    string deltime = (from g in db.tbldeltimes where g.DelTimeType ==
data.DelTimeType select g.DelTimeDescription).SingleOrDefault();
    _user.Add(new Eteproductbuyers
{PSId=data.PSId,
    ProductId = data.ProductId,
    SellerId = data.SellerId,
    SellerDescription = data.SellerDescription,
    Price =(decimal) data.Price,
    Currency = currency,
    PriceUnits = (int)data.PriceUnits,
    PriceUom = priceuom,
    PricePack = pricepack,
    KeyWords = data.KeyWords,

    UpdatedOn = data.UpdatedOn,

```

```

        MinOrder = (int)data.MinOrder,
        MinOrderUom =data.MinOrder.ToString()+ minuom,
        DelTime = (int)data.DelTime,
        DelTimeType = data.DelTime.ToString() + deltime,
        RedirctString=RedirectStr,
    });
}

}

}

catch (Exception exc)
{
    objlog.Trace_Error(5, "MLB", "Bal.Products", "GetProduct", exc.Message);
    throw exc;
}
return _user;
}
public List<Eteproductbuyers> GetProductBuyersByPID(decimal Pid,int Prid, string
Curl)
{
    List<Eteproductbuyers> _user = new List<Eteproductbuyers>();
    try {
        using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        {
            db.Connection.Open();
            int pgid = (new tblPagess()).GetPageId(Curl);
            var user = (from c in db.productSellers where c.Prid == Prid && c.ProductId==Pid
select c).ToList();
            foreach (var data in user)
            {
                string currency = (from a in db.Currencies where a.CId == data.CurrencyId
select a.CShortCode).SingleOrDefault();
                string priceuom = (from b in db.tbluoms where b.UomId == data.PriceUomId
select b.UomShort).SingleOrDefault();
                string pricepack = (from d in db.tblpackings where d.PackID ==
data.PricePackId select d.PackShort).SingleOrDefault();
                string minuom = (from e in db.tbluoms where e.UomId == data.MinOrderUom
select e.UomShort).SingleOrDefault();
                string Pack = (from f in db.tblpackings where f.PackID == data.PricePackId
select f.PackShort).SingleOrDefault();
                string deltime = (from g in db.tbldeltimes where g.DelTimeType ==
data.DelTimeType select g.DelTimeDescription).SingleOrDefault();
                string SellerName = (from j in db.tblUsers where j.UserId == data.SellerId select
j.LastName).SingleOrDefault();
                string ImageURL = "";
                System.Data.Linq.Binary Image = (from h in db.productSellerImages where
h.ProductId == data.ProductId && h.SellerId == data.SellerId select
h.Images).FirstOrDefault();
                if (Image != null)
                {
                    byte[] bytes = (byte[])Image.ToArray();

```

```

        string base64String = Convert.ToBase64String(bytes, 0, bytes.Length);
        ImageURL = "data:image/png;base64," + base64String;
    }
    _user.Add(new Eteproductbuyers
    {
        PSId = data.PSId,
        ProductId = data.ProductId,
        SellerId = data.SellerId,
        SellerDescription = data.SellerDescription,
        Price = (decimal)data.Price,
        Currency = currency+":" + data.Price.ToString(),
        PriceUnits = (int)data.PriceUnits,
        PriceUom = priceuom,
        PricePack = pricepack,
        KeyWords = data.KeyWords,
        ImageURL=ImageURL,
        UpdatedOn = data.UpdatedOn,
        MinOrder = (int)data.MinOrder,
        MinOrderUom = data.MinOrder.ToString() + minuom,
        DelTIme = (int)data.DelTIme,
        DelTimeType = data.DelTIme.ToString() + deltime,
        SellerName = SellerName,
    });
}

}

}
catch (Exception exc)
{
    objlog.Trace_Error(5, "MLB", "Bal.Products", "GetProduct", exc.Message);
    throw exc;
}
return _user;
}
public List<Eteproduct> GetUserOtherServices(int userid, decimal psid)
{
    List<Eteproduct> ep = new List<Eteproduct>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var lst = db.GetUserOtherServices(userid, psid).ToList();
        foreach (var dt in lst)
        {
            string pname = (from c in db.menulists where c.pageid == dt.productid select
c.pagename).FirstOrDefault();

            ep.Add(new Eteproduct
            {
                ProductId=dt.productid,
                Price=dt.PrdPrice,
                PriceUomId=dt.PrdUnit,

```

```

        ProductName = pname,
    });
}
}

return ep;
}
public List<ProductProductSeller> GetProductSellerbySearch(string search,int stp)
{
    List<ProductProductSeller> _user = new List<ProductProductSeller>();
    //try
    {
        using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        {
            db.Connection.Open();
            int pgid = (new tblPagess()).GetPageId("");
            List<GetProductSellerbySearchResult> user =
            db.GetProductSellerbySearch(search).ToList();
            foreach (var data in user)
            {
                var grp = (from e in db.menulists where e.pageid == data.ProductGroupId
                select e).SingleOrDefault();
                string GroupName = grp.pagename;
                string CatName = (from d in db.ProductCats where d.PCatId == grp.parentid
                select d.PcatDesc).FirstOrDefault();
                if (CatName == null) { CatName = ""; }
                int stype = (int)grp.stype;
                string currency = (from a in db.Currencies where a.CId == data.CurrencyId
                select a.CShortCode).SingleOrDefault();
                string priceuom = (from b in db.tbluoms where b.UomId == data.PriceUomId
                select b.UomShort).SingleOrDefault();
                string pricepack = (from d in db.tblpackings where d.PackID ==
                data.PricePackId select d.PackShort).SingleOrDefault();
                string minuom = (from e in db.tbluoms where e.UomId == data.MinOrderUom
                select e.UomShort).SingleOrDefault();
                if (minuom == null) { minuom = priceuom; }
                string Pack = (from f in db.tblpackings where f.PackID == data.PricePackId
                select f.PackShort).SingleOrDefault();
                string deltime = (from g in db.tbldeltimes where g.DelTimeType ==
                data.DelTimeType select g.DelTimeDescription).SingleOrDefault();
                string SellerName = (from j in db.tblUsers where j.UserId == data.SellerId
                select j.LastName).SingleOrDefault();
                string ImageURL = "";
                string ImagePath = "";
                int imgdata = (from j in db.productSellerImages where j.PSID == data.PSID
                select j).Count();
                if (imgdata > 0)
                {
                    var Imagedata = (from h in db.productSellerImages where h.PSID ==
                    data.PSID select h.ImagePath).ToList();
                    ImagePath = Imagedata[0];
                    if (imgdata > 1)
                    {
                        ImageURL = Imagedata[1];
                    }
                }
            }
        }
    }
}

```

```

    }
}
//System.Data.Linq.Binary Image = Imagedata.Images; ;
//if (ImagePath == "" || ImagePath == null)
//{
//    ImagePath = "images/progess.gif";
//}
//if (Image != null)
//{
//    byte[] bytes = (byte[])Image.ToArray();
//    string base64String = Convert.ToBase64String(bytes, 0, bytes.Length);
//    ImageURL = "data:image/png;base64," + base64String;
//}
if (data.LinkPSId == 0) { data.LinkPSId = data.PSId; }
if (stp == stype)
{
    _user.Add(new ProductProductSeller
    {
        PSId = data.PSId,
        ProductId = data.ProductId,
        ProductName = data.ProductName,

        SellerId = data.SellerId,
        SellerDescription = data.SellerDescription,
        ProductGroupId = data.ProductGroupId,
        CurrencyId = (int)data.CurrencyId,
        Price = (decimal)data.Price,
        CurPrice = currency + ":" + data.Price.ToString(),
        PriceUnits = (int)data.PriceUnits,
        QtyUOM = data.MinOrder.ToString() + priceuom,
        PriceUomId = (int)data.PriceUomId,
        PricePackId = (int)data.PricePackId,
        PricePack = pricepack,
        KeyWords = data.KeyWords,
        ImageURL = ImageURL,
        UpdatedOn = data.UpdatedOn,
        CreatedOn = data.CreatedOn,
        MinOrder = (int)data.MinOrder,
        MinOrderUom = (int)data.MinOrderUom,
        DelTIme = (int)data.DelTIme,
        DelTimeType = (int)data.DelTimeType,
        DTTImeType = data.DelTIme.ToString() + deltime,
        SellerName = SellerName,
        UserId = data.UserId,
        PrId = data.PrId,
        CatName = CatName,
        GroupName = GroupName,
        //Images=Image,

        ImagePath = ImagePath,
        LinkPSId = (decimal)data.LinkPSId,
        ProductDescription = data.ProductDescription,
        DiscountLink = (int)data.DiscountLink,
        BrandName = data.BrandName,
    }
}

```

```

        MaterialCare = data.MaterialCare,
        MaterialType = data.MaterialType,
        Manufacturer = data.Manufacturer,
        Length = data.Length,
        MadeinCountry = (int)data.MadeinCountry,
    });
    }
}

}

}
//catch (Exception exc)
//{
//    objlog.Trace_Error(5, "MLB", "Bal.Products", "GetProduct", exc.Message);
//    throw exc;

//}
return _user;
}
public List<ProductProductSeller> GetProductSellerbyCatId(int catId, int Prid, string
Curl)
{
    List<ProductProductSeller> _user = new List<ProductProductSeller>();
    //try
    {
        using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        {

            db.Connection.Open();
            int pgid = (new tblPagess()).GetPagelId(Curl);
            List<GetProductSellerbyCatIdResult> user =
            db.GetProductSellerbyCatId(catId).ToList();
            foreach (var data in user)
            {
                var grp = (from e in db.ProductGroups where e.pgid == data.ProductGroupld
                select e).SingleOrDefault();
                string GroupName = grp.pgroupdescription;
                string CatName = (from d in db.ProductCats where d.PCatId == grp.PCatId
                select d.PcatDesc).SingleOrDefault();

                string currency = (from a in db.Currencies where a.CId == data.Currencyld
                select a.CShortCode).SingleOrDefault();
                string priceuom = (from b in db.tbluoms where b.UomId == data.PriceUomld
                select b.UomShort).SingleOrDefault();
                string pricepack = (from d in db.tblpackings where d.PackID ==
                data.PricePackId select d.PackShort).SingleOrDefault();
                string minuom = (from e in db.tbluoms where e.UomId == data.MinOrderUom
                select e.UomShort).SingleOrDefault();
                if (minuom == null) { minuom = priceuom; }
                string Pack = (from f in db.tblpackings where f.PackID == data.PricePackId
                select f.PackShort).SingleOrDefault();
            }
        }
    }
}

```

```

        string deltime = (from g in db.tblDeltimes where g.DelTimeType ==
data.DelTimeType select g.DelTimeDescription).SingleOrDefault();
        string SellerName = (from j in db.tblUsers where j.UserId == data.SellerId
select j.LastName).SingleOrDefault();
        string ImageURL = "";
        string ImagePath = "";
        int imgdata = (from j in db.productSellerImages where j.PSID == data.PSID
select j).Count();
        if (imgdata > 0)
        {
            var Imagedata = (from h in db.productSellerImages where h.PSID ==
data.PSID select h).FirstOrDefault();
            ImagePath = Imagedata.ImagePath;
        }
        //System.Data.Linq.Binary Image = Imagedata.Images; ;
        //if (ImagePath == "" || ImagePath == null)
        //{
        //    ImagePath = "images/progress.gif";
        //}
        //if (Image != null)
        //{
        //    byte[] bytes = (byte[])Image.ToArray();
        //    string base64String = Convert.ToBase64String(bytes, 0, bytes.Length);
        //    ImageURL = "data:image/png;base64," + base64String;
        //}
        if (data.LinkPSId == null) { data.LinkPSId = data.PSID; }
        _user.Add(new ProductProductSeller
        {
            PSId = data.PSID,
            ProductId = data.ProductId,
            ProductName = data.ProductName,

            SellerId = data.SellerId,
            SellerDescription = data.SellerDescription,
            ProductGroupId = data.ProductGroupId,
            CurrencyId = (int)data.CurrencyId,
            Price = (decimal)data.Price,
            CurPrice = currency + ":" + data.Price.ToString(),
            PriceUnits = (int)data.PriceUnits,
            QtyUOM = data.MinOrder.ToString() + priceuom,
            PriceUomId = (int)data.PriceUomId,
            PricePackId = (int)data.PricePackId,
            PricePack = pricepack,
            KeyWords = data.KeyWords,
            ImageURL = ImageURL,
            UpdatedOn = data.UpdatedOn,
            CreatedOn = data.CreatedOn,
            MinOrder = (int)data.MinOrder,
            MinOrderUom = (int)data.MinOrderUom,
            DelTime = (int)data.DelTime,
            DelTimeType = (int)data.DelTimeType,
            DTTIMEType = data.DelTime.ToString() + deltime,
            SellerName = SellerName,
            UserId = data.UserId,
            PrId = data.PrId,

```



```

        CatName = CatName,
        GroupName = GroupName,
        //Images=Image,

        ImagePath = ImagePath,
        LinkPSId = (decimal)data.LinkPSId,
        ProductDescription = data.ProductDescription,
        DiscountLink = (int)data.DiscountLink,
        BrandName = data.BrandName,
        MaterialCare = data.MaterialCare,
        MaterialType = data.MaterialType,
        Manufacturer = data.Manufacturer,
        Length = data.Length,
        MadeinCountry = (int)data.MadeinCountry,

    });
}

}

}
//catch (Exception exc)
//{
//    objlog.Trace_Error(5, "MLB", "Bal.Products", "GetProduct", exc.Message);
//    throw exc;

//}
return _user;
}

public List<ProductProductSeller> GetProductSellerbyProductId(int ProductId, string
Curl)
{
    List<ProductProductSeller> _user = new List<ProductProductSeller>();
    //try
    {
        using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        {

            db.Connection.Open();
            int pgid = (new tblPagess()).GetPagelId(Curl);
            List<GetProductSellerbyProductIdResult> user =
            db.GetProductSellerbyProductId(ProductId).ToList();
            foreach (var data in user)
            {
                var grp= (from e in db.ProductGroups where e.pgid == data.ProductGroupId
                select e).SingleOrDefault();
                string GroupName =grp.pgroupdescription;
                string CatName = (from d in db.ProductCats where d.PCatId == grp.PCatId
                select d.PcatDesc).SingleOrDefault();

                string currency = (from a in db.Currencies where a.CId == data.CurrencyId
                select a.CShortCode).SingleOrDefault();

```

```

        string priceuom = (from b in db.tbluoms where b.UomId == data.PriceUomId
select b.UomShort).SingleOrDefault();
        string pricepack = (from d in db.tblpackings where d.PackID ==
data.PricePackId select d.PackShort).SingleOrDefault();
        string minuom = (from e in db.tbluoms where e.UomId == data.MinOrderUom
select e.UomShort).SingleOrDefault();
        if (minuom == null) { minuom = priceuom; }
        string Pack = (from f in db.tblpackings where f.PackID == data.PricePackId
select f.PackShort).SingleOrDefault();
        string deltime = (from g in db.tbldeltimes where g.DelTimeType ==
data.DelTimeType select g.DelTimeDescription).SingleOrDefault();
        string SellerName = (from j in db.tblUsers where j.UserId == data.SellerId select
j.LastName).SingleOrDefault();
        string ImageURL = "";
        string ImagePath = "";
        List<EteproductSellerStocks> stk = new
Products().GetSellerStocksbyPslId((int)data.PSId).ToList();
        string sizes = "";
        foreach (var stk1 in stk)
        {
            if (stk1.Stock > 0) {
                var sz = (from l in db.productExtPrOptions where l.EpOptId ==
stk1.EpOptId select l.EpOPtDescr).FirstOrDefault();
                sizes = sizes + "/" + sz;
            }
        }
        int imgdata = (from j in db.productSellerImages where j.PSID == data.PSId
select j).Count();
        if (imgdata > 0)
        {
            var Imagedata = (from h in db.productSellerImages where h.PSID ==
data.PSId select h).FirstOrDefault();
            ImagePath = Imagedata.ImagePath;
        }
        //System.Data.Linq.Binary Image = Imagedata.Images; ;
        //if (ImagePath == "" || ImagePath == null)
        //{
        //    ImagePath = "images/progess.gif";
        //}
        //if (Image != null)
        //{
        //    byte[] bytes = (byte[])Image.ToArray();
        //    string base64String = Convert.ToBase64String(bytes, 0, bytes.Length);
        //    ImageURL = "data:image/png;base64," + base64String;
        //}
        if (data.LinkPSId == null) { data.LinkPSId = data.PSId; }
        _user.Add(new ProductProductSeller
        {
            PSId = data.PSId,
            ProductId = data.ProductId,
            ProductName = data.ProductName,

            SellerId = data.SellerId,
            SellerDescription = data.SellerDescription,
            ProductGroupId=data.ProductGroupId,

```

```

CurrencyId=(int)data.CurrencyId,
Price = (decimal)data.Price,
CurPrice = currency + ":" + data.Price.ToString(),
PriceUnits = (int)data.PriceUnits,
QtyUOM = data.MinOrder.ToString()+ priceuom,
PriceUomId = (int)data.PriceUomId,
PricePackId = (int)data.PricePackId,
PricePack = pricepack,
KeyWords = data.KeyWords,
ImageURL = ImageURL,
UpdatedOn = data.UpdatedOn,
CreatedOn = data.CreatedOn,
MinOrder = (int)data.MinOrder,
MinOrderUom = (int)data.MinOrderUom,
DelTime = (int)data.DelTime,
DelTimeType = (int)data.DelTimeType,
DTTimeType = data.DelTime.ToString() + deltime,
SellerName = SellerName,
UserId = data.UserId,
PrId=data.PrId,
CatName=CatName,
GroupName=GroupName,
//Images=Image,

ImagePath = ImagePath,
LinkPSId = (decimal)data.LinkPSId,
ProductDescription = data.ProductDescription,
DiscountLink = (int)data.DiscountLink,
BrandName = data.BrandName,
MaterialCare = data.MaterialCare,
MaterialType = data.MaterialType,
Manufacturer = data.Manufacturer,
Length = data.Length,
MadeinCountry = (int)data.MadeinCountry,
sizeavl=sizes,
});
}

}

}
//catch (Exception exc)
//{{
//  objlog.Trace_Error(5, "MLB", "Bal.Products", "GetProduct", exc.Message);
//  throw exc;

//}}
return _user;
}
public List<Eteproductbuyers> GetProductBuyersByPSID(decimal PSId, string Curl)
{
    List<Eteproductbuyers> _user = new List<Eteproductbuyers>();
    try {
        using (var db = new DowPortalDataContext(Configuration.ConnectionString))

```

```

{
    db.Connection.Open();
    int pgid = (new tblPagess()).GetPagelD(Curl);
    var user = (from c in db.productSellers where c.PSId == PSId select c).ToList();
    foreach (var data in user)
    {
        string currency = (from a in db.Currencies where a.CId == data.CurrencyId
select a.CShortCode).SingleOrDefault();
        string priceuom = (from b in db.tbluoms where b.UomId == data.PriceUomId
select b.UomShort).SingleOrDefault();
        string pricepack = (from d in db.tblpackings where d.PackID ==
data.PricePackId select d.PackShort).SingleOrDefault();
        string minuom = (from e in db.tbluoms where e.UomId == data.MinOrderUom
select e.UomShort).SingleOrDefault();
        string Pack = (from f in db.tblpackings where f.PackID == data.PricePackId
select f.PackShort).SingleOrDefault();
        string deltime = (from g in db.tbldeltimes where g.DelTimeType ==
data.DelTimeType select g.DelTimeDescription).SingleOrDefault();
        string SellerName = (from j in db.tblUsers where j.UserId == data.SellerId select
j.LastName).SingleOrDefault();
        string ImageURL = "";
        System.Data.Linq.Binary Image = (from h in db.productSellerImages where
h.ProductId == data.ProductId && h.SellerId == data.SellerId select
h.Images).FirstOrDefault();
        if (Image != null)
        {
            byte[] bytes = (byte[])Image.ToArray();
            string base64String = Convert.ToBase64String(bytes, 0, bytes.Length);
            ImageURL = "data:image/png;base64," + base64String;
        }
        var prd = (from j in db.Products where j.ProductId == data.ProductId select
j).SingleOrDefault();
        _user.Add(new Eteproductbuyers
        {
            PSId = data.PSId,
            ProductId = data.ProductId,
            SellerId = data.SellerId,
            SellerDescription = data.SellerDescription,
            Price = (decimal)data.Price,
            Currency = currency + ":" + data.Price.ToString(),
            PriceUnits = (int)data.PriceUnits,
            PriceUom = priceuom,
            PricePack = pricepack,
            KeyWords = data.KeyWords,
            ImageURL = ImageURL,
            UpdatedOn = data.UpdatedOn,
            MinOrder = (int)data.MinOrder,
            MinOrderUom = data.MinOrder.ToString() + minuom,
            DelTime = (int)data.DelTime,
            DelTimeType = data.DelTime.ToString() + deltime,
            SellerName = SellerName,
            ProductName=prd.ProductName,
        });
    }
}

```

```

    }
    return _user;
}
catch (Exception exc)
{
    objlog.Trace_Error(5, "MLB", "Bal.Products", "GetProduct", exc.Message);
    throw exc;
}
}

public List<ProductProductSeller> GetProductSellerbySellerId(int userid,int stype)
{
    List<ProductProductSeller> _user = new List<ProductProductSeller>();
    //try
    {
        using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        {
            db.Connection.Open();
            //int pgid = (new tblPagess()).GetPageld("");
            List<GetProductSellerbySellerIdResult> user =
            db.GetProductSellerbySellerId(userid, stype).ToList();
            //var user = (from c in db.productSellers join d in db.Products on c.ProductId
            equals d.ProductId where c.UserId == userid && d.Stype==stype select new{ c,d}).ToList();
            foreach (var data in user)
            {
                //var data=data1.c;
                //var data2=data1.d;
                var grp = (from e in db.menulists where e.pageid == data.ProductGroupld
                select e).SingleOrDefault();
                string GroupName = grp.pagename;
                string CatName = data.PcatDesc; //(from d in db.ProductCats where
                d.Pcatld == grp.Pcatld select d.PcatDesc).SingleOrDefault();

                string currency = (from a in db.Currencies where a.Cld == data.Currencyld
                select a.CShortCode).SingleOrDefault();
                string priceuom = (from b in db.tbluoms where b.Uomld == data.PriceUomld
                select b.UomShort).SingleOrDefault();
                string pricepack = (from d in db.tblpackings where d.PackID ==
                data.PricePackld select d.PackShort).SingleOrDefault();
                string minuom = (from e in db.tbluoms where e.Uomld == data.MinOrderUom
                select e.UomShort).SingleOrDefault();
                if (minuom == null) { minuom = priceuom; }
                string Pack = (from f in db.tblpackings where f.PackID == data.PricePackld
                select f.PackShort).SingleOrDefault();
                string deltime = (from g in db.tbldeltimes where g.DelTimeType ==
                data.DelTimeType select g.DelTimeDescription).SingleOrDefault();
                string SellerName = (from j in db.tblUsers where j.UserId == data.Sellerld
                select j.LastName).SingleOrDefault();
                string ImageURL = "";
                string ImagePath = "";
            }
        }
    }
}

```

```

        int imgdata = (from j in db.productSellerImages where j.PSID == data.PSID
select j).Count();
        if (imgdata > 0)
        {
            var Imagedata = (from h in db.productSellerImages where h.PSID ==
data.PSID select h.ImagePath).FirstOrDefault();
            ImagePath = Imagedata;
        }
        int stock=0;
        List<EteproductSellerStocks> stk = new
Products().GetSellerStocksbyPslId((int)data.PSID).ToList();
        foreach (var dt in stk)
        {
            stock = stock + dt.Stock;
        }
        //System.Data.Linq.Binary Image = Imagedata.Images; ;
        //if (ImagePath == "" || ImagePath == null)
        //{
        //    ImagePath = "images/progess.gif";
        //}
        //if (Image != null)
        //{
        //    byte[] bytes = (byte[])Image.ToArray();
        //    string base64String = Convert.ToBase64String(bytes, 0, bytes.Length);
        //    ImageURL = "data:image/png;base64," + base64String;
        //}
        if (data.LinkPSId == 0) { data.LinkPSId = data.PSID; }
        _user.Add(new ProductProductSeller
        {
            PSId = data.PSID,
            ProductId = data.ProductId,
            ProductName = data.ProductName,

            SellerId = data.SellerId,
            SellerDescription = data.SellerDescription,
            ProductGroupId = data.ProductGroupId,
            CurrencyId = (int)data.CurrencyId,
            Price = (decimal)data.Price,
            CurPrice = currency + ":" + data.Price.ToString(),
            PriceUnits = (int)data.PriceUnits,
            QtyUOM = data.MinOrder.ToString() + priceuom,
            PriceUomId = (int)data.PriceUomId,
            PricePackId = (int)data.PricePackId,
            PricePack = pricepack,
            KeyWords = data.KeyWords,
            ImageURL = ImageURL,
            UpdatedOn = data.UpdatedOn,
            CreatedOn = data.CreatedOn,
            MinOrder = (int)data.MinOrder,
            MinOrderUom = (int)data.MinOrderUom,
            DelTIme = (int)data.DelTIme,
            DelTimeType = (int)data.DelTimeType,
            DTTImeType = data.DelTIme.ToString() + deltime,
            SellerName = SellerName,
            UserId = data.UserId,
        }
    
```

```

        Prid = data.Prid,
        CatName = CatName,
        GroupName = GroupName,
        //Images=Image,

        ImagePath = ImagePath,
        LinkPSId = (decimal)data.LinkPSId,
        ProductDescription = data.ProductDescription,
        DiscountLink = (int)data.DiscountLink,
        BrandName = data.BrandName,
        MaterialCare = data.MaterialCare,
        MaterialType = data.MaterialType,
        Manufacturer = data.Manufacturer,
        Length = data.Length,
        MadeinCountry = (int)data.MadeinCountry,
        TStock = stock,
    });
}

}

}
//catch (Exception exc)
//{
//    objlog.Trace_Error(5, "MLB", "Bal.Products", "GetProduct", exc.Message);
//    throw exc;

//}
return _user;
}

public List<Eteproductbuyers> GetProductBuyersForPSID(decimal PSId, string Curl)
{
    List<Eteproductbuyers> _user = new List<Eteproductbuyers>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        int pgid = (new tblPagess()).GetPagelId(Curl);
        var user = (from c in db.productSellers where c.PSId == PSId select c).ToList();
        foreach (var data in user)
        {
            string currency = (from a in db.Currencies where a.CId == data.CurrencyId
            select a.CShortCode).SingleOrDefault();
            string priceuom = (from b in db.tbluoms where b.UomId == data.PriceUomId
            select b.UomShort).SingleOrDefault();
            string pricepack = (from d in db.tblpackings where d.PackID ==
            data.PricePackId select d.PackShort).SingleOrDefault();
            string minuom = (from e in db.tbluoms where e.UomId == data.MinOrderUom
            select e.UomShort).SingleOrDefault();
            string Pack = (from f in db.tblpackings where f.PackID == data.PricePackId
            select f.PackShort).SingleOrDefault();

```

```

        string deltime = (from g in db.tblDelTimes where g.DelTimeType ==
data.DelTimeType select g.DelTimeDescription).SingleOrDefault();
        string SellerName = (from j in db.tblUsers where j.UserId == data.SellerId select
j.LastName).SingleOrDefault();
        string ImageURL = "";
        System.Data.Linq.Binary Image = (from h in db.productSellerImages where
h.ProductId == data.ProductId && h.SellerId == data.SellerId select
h.Images).FirstOrDefault();
        if (Image != null)
        {
            byte[] bytes = (byte[])Image.ToArray();
            string base64String = Convert.ToBase64String(bytes, 0, bytes.Length);
            ImageURL = "data:image/png;base64," + base64String;
        }
        if (data.CurrencyId == 0) { data.CurrencyId = 1; }
        if (data.PricePackId == 0) { data.PricePackId = 1; }

        var prd = (from j in db.Products where j.ProductId == data.ProductId select
j).SingleOrDefault();
        _user.Add(new Eteproductbuyers
        {
            PSId = data.PSId,
            ProductId = data.ProductId,
            SellerId = data.SellerId,
            SellerDescription = data.SellerDescription,
            Price = (decimal)data.Price,
            Currency = data.CurrencyId.ToString(),
            PriceUnits = (int)data.PriceUnits,
            PriceUom =data.PriceUomId.ToString(),
            PricePack = data.PricePackId.ToString(),
            KeyWords = data.KeyWords,
            ImageURL = ImageURL,
            UpdatedOn = data.UpdatedOn,
            MinOrder = (int)data.MinOrder,
            MinOrderUom = data.MinOrderUom.ToString() ,
            DelTime = (int)data.DelTime,
            DelTimeType = data.DelTimeType.ToString(),
            SellerName = SellerName,
            ProductName = prd.ProductName,
        });
    }

}

return _user;
}
public List<EteProductGroup> GetProductExtProps(int pid)
{
    List<EteProductGroup> dt = new List<EteProductGroup>();

    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        var data = (from c in db.productExtProperties where c.EpId == pid select
c).ToList();
    }
}

```



```

        foreach (var row in data)
        {
            dt.Add(new EteProductGroup
            {
                ProductId = row.EpId,
                ProductName=row.EpDescription,
                isActive=row.active,
            });
        }
    }
    return dt;
}
public List<EteproductImages> GetProductExtColors(int pid)
{
    List<EteproductImages> dt = new List<EteproductImages>();

    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        var data = (from c in db.productColorOptions where c.active select c).ToList();
        foreach (var row in data)
        {
            string imageUrl = "";
            if(row.ImageURL!=null){ imageUrl=row.ImageURL; }
            else if(row.Images!=null)
            {
                byte[] bytes = (byte[])row.Images.ToArray();
                string base64String = Convert.ToBase64String(bytes, 0, bytes.Length);
                imageUrl = "data:image/png;base64," + base64String;
            }
            else
            {

            }

            dt.Add(new EteproductImages
            {
                ProductId = row.ColorId,
                Images = row.Images,
                ImageUrls = imageUrl,
                ImagePath = row.ImagePath,
                Descr=row.ColorDescr,
            });
        }
    }
    return dt;
}
public List<EteProductGroup> GetProductExtPropts(int pid)
{
    List<EteProductGroup> dt = new List<EteProductGroup>();

    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

```

```

        var data = (from c in db.productExtPrOptions where c.Epld == pid select
c).ToList();
        foreach (var row in data)
        {
            dt.Add(new EteProductGroup
            {
                ProductId = row.Epld,
                pgid=row.EpOptId,
                ProductName = row.EpOPtDescr,
                isActive = row.active,
            });
        }
    }
    return dt;
}
public List<EteProductGroup> GeproductSellerExtProperties(int psid,int pid, int epid)
{
    List<EteProductGroup> dt = new List<EteProductGroup>();

    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        var data = (from c in db.productSellerExtProperties where c.PSId==psid && c.Epld
== pid && c.EpOptId == epid select c).ToList();
        foreach (var row in data)
        {
            dt.Add(new EteProductGroup
            {ProductId=row.PSId,
                PCatId= row.Epld,
                pgid = row.EpOptId,
                ProductName = row.Remark,
                isActive = row.active,
            });
        }
    }
    return dt;
}
public List<EteproductSellerServices> GeproductSellerServicesbyPSID(int psid)
{
    List<EteproductSellerServices> dt = new List<EteproductSellerServices>();

    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        var data = (from c in db.productSellerServices where c.PSId == psid select
c).ToList();
        foreach (var row in data)
        {
            dt.Add(new EteproductSellerServices
            {
                PSId = row.PSId,
                productid=row.productid,
                Remark = row.Remark,
                PrdPrice = row.PrdPrice,
                PrdUnit = row.PrdUnit,
                active = row.active,
            });
        }
    }
}

```

```

    }
    }
    return dt;
}

public List<EteProductGroup> GeproductSellerExtPropertiesbyPSID(int psid)
{
    List<EteProductGroup> dt = new List<EteProductGroup>();

    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        var data = (from c in db.productSellerExtProperties where c.PSId == psid select
c).ToList();
        foreach (var row in data)
        {
            dt.Add(new EteProductGroup
            {
                ProductId = row.PSId,
                PCatId = row.EpId,
                pgid = row.EpOptId,
                ProductName = row.Remark,
                isActive = row.active,
            });
        }
    }
    return dt;
}

public List<Eteproduct> GetProductServices(string Curl)
{
    List<Eteproduct> _user = new List<Eteproduct>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        db.Connection.Open();
        int pgid = (new tblPagess()).GetPageld(Curl);
        var users = (from c in db.Products where c.Stype == 2 orderby
c.ProductGroupCategory, c.ProductGroupId ascending select c).ToList();
        foreach (var data in users)
        {
            var pnos = (from d in db.productSellers where d.ProductId == data.ProductId
select d).Count();
            var catname = (from e in db.ProductCats where e.PCatId ==
data.ProductGroupCategory select e.PcatDesc).FirstOrDefault();
            var gname = (from f in db.ProductGroups where f.pgid == data.ProductGroupId
select f.pgroupdescription).FirstOrDefault();
            _user.Add(new Eteproduct
            {
                ProductId = data.ProductId,
                ProductName = data.ProductName,
                CatName = catname,
                GroupName = gname,
                ProductGroupId = data.ProductGroupId,
                ProductGroupCategory = (int)data.ProductGroupCategory,

                Type = pnos.ToString(),
            });
        }
    }
}

```

```

        Price = (decimal)data.Price,

        //Freq = data.Freq,

        //Period = data.Period,
        CurrencyId = (int)data.CurrencyId,
        AnalysisCode = data.AnalysisCode.ToString(),
        CreatedOn =(DateTime) data.CreatedOn,
        UpdatedOn =( DateTime)data.UpdatedOn,
        UserId =(int) data.UserId,
        //PrId = data.PrId,
        MinOrder = (int)data.MinOrder,
        DelTime = (int)data.DelTime,
        Check = true,

    });
}

return _user;
}
}

public List<Eteproduct> GetProductbyGroup(int id, string Curl)
{
    List<Eteproduct> _user = new List<Eteproduct>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        db.Connection.Open();
        int pgid = (new tblPagess()).GetPagelId(Curl);
        var users = (from c in db.Products where c.ProductGroupId == id orderby
c.ProductGroupCategory,c.ProductGroupId ascending select c).ToList();
        foreach (var data in users)
        {
            var pnos = (from d in db.productSellers where d.ProductId == data.ProductId
select d).Count();
            var catname = (from e in db.ProductCats where e.PCatId ==
data.ProductGroupCategory select e.PcatDesc).FirstOrDefault();
            var gname = (from f in db.ProductGroups where f.pgid == data.ProductGroupId
select f.pgroupdescription).FirstOrDefault();
            _user.Add(new Eteproduct
            {
                ProductId = data.ProductId,
                ProductName = data.ProductName,
                CatName = catname,
                GroupName = gname,
                ProductGroupId = data.ProductGroupId,
                ProductGroupCategory = (int)data.ProductGroupCategory,

                Type = pnos.ToString(),

                Price = (decimal)data.Price,

```

```

        //Freq = data.Freq,

        //Period = data.Period,
        CurrencyId = (int)data.CurrencyId,
        AnalysisCode = data.AnalysisCode.ToString(),
        CreatedOn = (DateTime)data.CreatedOn,
        UpdatedOn = (DateTime)data.UpdatedOn,
        UserId =(int) data.UserId,
        //PrId = data.PrId,
        MinOrder = (int)data.MinOrder,
        DelTime = (int)data.DelTime,
        Check=true,

    });
}

return _user;
}
}
public List<Eteproduct> GetProducts( string Curl)
{
    List<Eteproduct> _user = new List<Eteproduct>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        db.Connection.Open();
        int pgid = (new tblPagess()).GetPagelId(Curl);
        var users = (from c in db.Products select c).ToList();
        foreach (var data in users)
        {
            string GroupName = (from e in db.ProductGroups where e.pgid ==
data.ProductGroupId select e.pgrouppdescription).SingleOrDefault();
            string CatName = (from d in db.ProductCats where d.PCatId ==
data.ProductGroupCategory select d.PcatDesc).SingleOrDefault();
            _user.Add(new Eteproduct
            {
                ProductId = data.ProductId,
                ProductName = data.ProductName,

                ProductGroupId = data.ProductGroupId,
                ProductGroupCategory = (int)data.ProductGroupCategory,

                //Type = data.Type,

                Price = (decimal)data.Price,

                //Freq = data.Freq,

                //Period = data.Period,
                CurrencyId = (int)data.CurrencyId,
                AnalysisCode = data.AnalysisCode.ToString(),
                CreatedOn = (DateTime)data.CreatedOn,
                UpdatedOn = (DateTime)data.UpdatedOn,
                UserId =(int) data.UserId,
            });
        }
    }
}

```

```

        PrId = data.PrId,
        MinOrder = (int)data.MinOrder,
        DelTIme = (int)data.DelTIme,
        Check = true,
        CatName = CatName,
        GroupName = GroupName,
        imageurl=data.lbtnimage,
    });
}

return _user;
}
}
public List<Eteproduct> GetProductbySellerId(int Sid, string Curl)
{
    List<Eteproduct> _user = new List<Eteproduct>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        db.Connection.Open();
        int pgid = (new tblPagess()).GetPagelId(Curl);
        var users = db.GetProductbySellerId(Sid);
        foreach (var data in users)
        {
            _user.Add(new Eteproduct
            {
                ProductId = data.ProductId,
                ProductName = data.ProductName,

                ProductGroupId = data.ProductGroupId,
                ProductGroupCategory =(int) data.ProductGroupCategory,

                //Type = data.Type,

                Price = (decimal)data.Price,

                //Freq = data.Freq,

                //Period = data.Period,
                CurrencyId = (int)data.CurrencyId,
                AnalysisCode = data.KeyWords,
                CreatedOn = data.CreatedOn,
                UpdatedOn = data.UpdatedOn,
                UserId = data.UserId,
                //PrId = data.PrId,
                MinOrder = (int)data.MinOrder,
                DelTIme = (int)data.DelTIme,
                Check = true,

            });
        }

        return _user;
    }
}

```

```

}
public List<Eteproductcat> GetProductCats(string Curl)
{
    List<Eteproductcat> _user = new List<Eteproductcat>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        int pgid = (new tblPagess()).GetPagelId(Curl);
        var users = (from c in db.ProductCats select c).ToList();
        foreach (var data in users)
        {
            var cnt = (from d in db.ProductGroups where d.PCatId == data.PCatId select
d).Count();
            _user.Add(new Eteproductcat
            {
                pcatid =Convert.ToInt16( data.PCatId),
                pcatdesc = data.PcatDesc,
                lbtnimage=data.lbtnimage,
                CreatedOn=Convert.ToDateTime( data.CreatedOn),
                active=true,
                GCount=cnt,
                Stype=Convert.ToInt16( data.Stype),
            });
        }

        return _user;
    }
}
public Eteproduct GetColorByColId(int colid)
{
    Eteproduct col =new Eteproduct();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var data = (from c in db.productColorOptions where c.ColorId == colid select
c).SingleOrDefault();
        col.ProductId = colid;
        col.ProductName = data.ColorDescr;
        col.imageurl = data.ImagePath;
    }
    return col;
}
public string GetSizeByOptId(int OptId)
{
    string col = "";
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var data = (from c in db.productExtPrOptions where c.EpOptId == OptId &&
c.EpId==1 select c.EpOPtDescr).SingleOrDefault();
        col = data.ToString();
    }
}

```

```

    }
    return col;
}

public Eteproductcat GetProductCat(int id)
{
    Eteproductcat _user = new Eteproductcat();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        db.Connection.Open();
        var data = (from c in db.ProductCats where c.PCatId == id select
c).SingleOrDefault();
        if (data != null)
        {
            _user.pcatid = Convert.ToInt16(data.PCatId);
            _user.pcatdesc = data.PcatDesc;
            _user.lbtnimage = data.lbtnimage;
            _user.Stype =(int) data.Stype;
        }

    }
    return _user;
}

public Eteproductcat GetProductCatbypid(int id)
{
    Eteproductcat _user = new Eteproductcat();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        db.Connection.Open();
        // int pgid = (new tblPagess()).GetPageId(Curl);
        var prod=(from d in db.Products where d.ProductId==id select
d).SingleOrDefault();
        var data = (from c in db.ProductCats where c.PCatId ==
prod.ProductGroupCategory select c).SingleOrDefault();
        if (data != null)
        {
            _user.pcatid = Convert.ToInt16(data.PCatId);
            _user.pcatdesc = data.PcatDesc;
            _user.lbtnimage = data.lbtnimage;
        }

    }

    return _user;
}

public List<EteProductGroup> GetProductGroups(string Curl)
{
    List<EteProductGroup> _user = new List<EteProductGroup>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

```



```

        isActive = true,
        createdOn = dt.CreatedOn,
        lbtnimage = dt.lbtnimage,
        UserId = (int)users.UserId,
        PNos = pnos,
        Catname = pcat,
        Price = (decimal)dt.Price,
    });
}
}

return _user;
}
}
public List<EteproductSellerStocks> GetSellerStocksbyPsId(int PsId)
{
    List<EteproductSellerStocks> Lst = new List<EteproductSellerStocks>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var col = (from c in db.productSellerStocks where c.PSId == PsId select
c).ToList();
        foreach (var dt in col)
        {
            Lst.Add(new EteproductSellerStocks
            {
                PSId = dt.PSId,
                EpId = dt.EpId,
                EpOptId = dt.EpOptId,
                colid = dt.colid,
                Stock = dt.Stock,
                Remark = dt.Remark,
                active = dt.active,
            }
        );
    }
}
return Lst;
}
public int GetSellerStocksbyPsIdCol(int PsId,int ColId ,int EpId, int EpOptId)
{
    int Lst =0;
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var seller = (from c in db.productSellerStocks where c.PSId == PsId && c.EpId ==
EpId && c.EpOptId == EpOptId && c.colid == ColId select c.Stock).SingleOrDefault();
        Lst=seller;
    }
return Lst;
}

```

```

}

public List<EteSellerColorSizes> GetSellerColorSizes(int UserId)
{
    List<EteSellerColorSizes> Lst = new List<EteSellerColorSizes>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var col = (from c in db.v_sellercolorsizes where c.userid == UserId select
c).ToList();
        foreach (var dt in col)
        {
            Lst.Add(new EteSellerColorSizes
            {
                userid = dt.userid,
                ColorId = dt.ColorId,
                ColorDescr = dt.ColorDescr,
                Images = dt.Images,
                ImagePath = dt.ImagePath,
                ImageURL = dt.ImageURL,
                ImageThumb = dt.ImageThumb,
                EpDescription = dt.EpDescription,
                epid = dt.epid,
                EpOptId = dt.EpOptId,
                EpOPtDescr = dt.EpOPtDescr,
                Stock=0,
            }
            );
        }
    }
    return Lst;
}

public List<EteSellerSizes> GetSellerSizes(int UserId)
{
    List<EteSellerSizes> Lst = new List<EteSellerSizes>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var col = (from c in db.SellersSizes join d in db.productExtPrOptions on c.Epld
equals d.Epld
                where c.EpOptId==d.EpOptId && c.userid == UserId && c.active select
new{ d.Epld,d.EpOptId,d.EpOPtDescr,c.CreatedOn}).ToList();
        foreach (var dt in col)
        {
            Lst.Add(new EteSellerSizes
            {
                Epld = dt.Epld,
                SellerId = UserId,
                EpOptId=dt.EpOptId,
                EpOptDescr = dt.EpOPtDescr,
                CreatedOn=dt.CreatedOn,
            }
            );
        }
    }
}

```

```

    }
    return Lst;
}
public List<Eteproduct> GetSellerProductServices( int userid)
{
    List<Eteproduct> Lst = new List<Eteproduct>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var col = (from c in db.SellersProducts join d in db.Products on c.productid equals
d.ProductId where d.ProductId==2 && c.userid == userid select d).ToList();
        foreach (var dt in col)
        {
            Lst.Add(new Eteproduct
            {
                ProductId = dt.ProductId,
                ProductName = dt.ProductName,
                ProductGroupCategory = (int)dt.ProductGroupCategory,
                ProductGroupId = dt.ProductGroupId,
                Price = (decimal)dt.Price,
                imageurl = dt.lbtnimage,
                CreatedOn = (DateTime)dt.CreatedOn,

            }
            );
        }
    }
    return Lst;
}
public List<Eteproduct> GetSellerServicesbypsid(decimal psid)
{
    List<Eteproduct> Lst = new List<Eteproduct>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var col = (from c in db.productSellerServices join d in db.Products on c.productid
equals d.ProductId where d.Stype == 2 && c.PSID == psid select d).ToList();
        foreach (var dt in col)
        {
            Lst.Add(new Eteproduct
            {
                ProductId = dt.ProductId,
                ProductName = dt.ProductName,
                ProductGroupCategory = (int)dt.ProductGroupCategory,
                ProductGroupId = dt.ProductGroupId,
                Price = (decimal)dt.Price,
                imageurl = dt.lbtnimage,
                CreatedOn = (DateTime)dt.CreatedOn,

            }
            );
        }
    }
}

```

```

    return Lst;
}

public List<Eteproduct> GetSellerProductsbygrid(int grid,int userid)
{
    List<Eteproduct> Lst = new List<Eteproduct>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var col = (from c in db.SellersProducts join d in db.Products on c.productid equals
d.ProductId where d.ProductGroupId == grid && c.userid==userid select d).ToList();
        foreach (var dt in col)
        {
            Lst.Add(new Eteproduct
            {
                ProductId = dt.ProductId,
                ProductName = dt.ProductName,
                ProductGroupCategory = (int)dt.ProductGroupCategory,
                ProductGroupId=dt.ProductGroupId,
                Price=(decimal)dt.Price,
                imageurl=dt.lbtnimage,
                CreatedOn = (DateTime)dt.CreatedOn,

            }
            );
        }
    }
    return Lst;
}

public bool UpdSellerServices(decimal psid,decimal prdid, int userid, bool chk)
{
    bool ret = false;
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var col = (from c in db.productSellerServices where c.PSId== psid && c.productid
== prdid select c).FirstOrDefault();
        if (col != null)
        {
            if (col.active && !chk)
            {
                //make it inactive
                col.active = false;
                db.SubmitChanges();
                db.Transaction.Commit();
                ret = true;
            }
        }
    }
    else
        if (chk)
        {
            SellersProduct _cs = new SellersProduct

```

```

        {
            productid = prdid,
            userid = userid,
            uid = userid,
            CreatedOn = DateTime.Today,
            active = true,
        };
        db.SellersProducts.InsertOnSubmit(_cs);

        db.SubmitChanges();
        //db.Transaction.Commit();

        ret = true;
    }
}

return ret;
}
public bool UpdSellerProduct(decimal prdid, int userid, bool chk)
{
    bool ret = false;
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var col = (from c in db.SellersProducts where c.userid == userid &&
c.productid==prdid select c).FirstOrDefault();
        if (col != null)
        {
            if(col.active && !chk)
            {
                //make it inactive
                col.active=false;
                db.SubmitChanges();
                //db.Transaction.Commit();
                ret = true;
            }
        }
        else
        if (chk)
        {
            SellersProduct _cs = new SellersProduct
            {
                productid = prdid,
                userid = userid,
                uid = userid,
                CreatedOn = DateTime.Today,
                active = true,
            };
            db.SellersProducts.InsertOnSubmit(_cs);

            db.SubmitChanges();
        }
    }
}

```

```

        //db.Transaction.Commit();

        ret = true;
    }
}

return ret;
}
public bool UpdSellerColors(int colid, int userid, bool chk)
{
    bool ret = false;
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var col = (from c in db.SellersColors where c.userid == userid && c.ColorId ==
colid select c).FirstOrDefault();
        if (col != null)
        {
            if (col.active && !chk)
            {
                //make it inactive
                col.active = false;
                db.SubmitChanges();
                db.Transaction.Commit();
                ret = true;
            }
        }
        else
            if (chk)
            {
                {
                    SellersColor _cs = new SellersColor
                    {
                        ColorId = colid,
                        userid = userid,
                        uid = userid,
                        CreatedOn = DateTime.Today,
                        active = true,
                    };
                    db.SellersColors.InsertOnSubmit(_cs);

                    db.SubmitChanges();
                    //db.Transaction.Commit();

                    ret = true;
                }
            }
    }

    return ret;
}
}

```

```

public bool UpdSellerSizes(int sizid, int userid, bool chk)
{
    bool ret = false;
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var col = (from c in db.SellersSizes where c.userid == userid && c.EpOptId ==
sizid select c).FirstOrDefault();
        if (col != null)
        {
            if (col.active && !chk)
            {
                //make it inactive
                col.active = false;
                db.SubmitChanges();
                //db.Transaction.Commit();
                ret = true;
            }
        }
        else
        {
            if (chk)
            {
                {
                    SellersSize _cs = new SellersSize
                    {
                        EpOptId = sizid,
                        userid = userid,
                        uid = userid,
                        CreatedOn = DateTime.Today,
                        active = true,
                    };
                    db.SellersSizes.InsertOnSubmit(_cs);

                    db.SubmitChanges();
                    //db.Transaction.Commit();

                    ret = true;
                }
            }
        }
    }

    return ret;
}

public List<EteSellerProduct> GetSellerProducts(int UserId,int stype)
{
    List<EteSellerProduct> Lst = new List<EteSellerProduct>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var col = (from c in db.SellersProducts join d in db.Products on c.productid equals
d.ProductId join e in db.ProductCats on d.ProductGroupCategory equals e.PCatId where
c.userid == UserId && e.Stype==stype select d).ToList();
        foreach (var dt in col)
    }
}

```



```

    {
        Lst.Add(new EteSellerProduct
        {
            ProductId = dt.ProductId,
            SellerId = UserId,
            ProductDesc = dt.ProductName,
            pcatid = (int)dt.ProductGroupCategory,
            pgid=dt.ProductGroupId,

        }
        );
    }
}
return Lst;
}

public List<EteSellerColors> GetSellerColorOptions(int UserId)
{
    List<EteSellerColors> Lst = new List<EteSellerColors>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        List<GetSellerColorOptionsResult> col =
db.GetSellerColorOptions(UserId).ToList();
        foreach(var dt in col)
        {
            Lst.Add(new EteSellerColors
            {ColorId=dt.ColorId,
            SellerId=UserId,
            ColorDesc=dt.ColorDesc,
            //Images=dt.Images,
            ImagePath=dt.ImagePath,
            //ImageUrl=dt.ImageURL,
            //ImageThumb=dt.ImageThumb,
            active=dt.active,
            usercolactive=(bool)dt.sellercolact,
            CreatedOn=(DateTime)dt.CreatedOn,
            }
            );
        }
    }
    return Lst;
}

public List<EteSellerColors> GetSellerColors(int UserId)
{
    List<EteSellerColors> Lst = new List<EteSellerColors>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var col=(from c in db.SellersColors join d in db.productColorOptions on c.ColorId
equals d.ColorId where c.userid==UserId && c.active select d).ToList();
        foreach (var dt in col)
        {

```

```

        Lst.Add(new EteSellerColors
        {ColorId=dt.ColorId,
        SellerId=UserId,
        ColorDesc=dt.ColorDescr,
        Images=dt.Images,
        ImagePath=dt.ImagePath,
        ImageURL=dt.ImageURL,
        ImageThumb=dt.ImageThumb,

        }
        );
    }
}
return Lst;
}
public List<EteProductGroup> GetProductsList(string Curl, int pcatid)
{
    List<EteProductGroup> _user = new List<EteProductGroup>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        db.Connection.Open();
        int pgid = (new tblPagess()).GetPagelId(Curl);
        var users = (from c in db.ProductGroups where c.PCatId == pcatid orderby
c.pgroupname select c).ToList();
        var pcat = (from e in db.ProductCats where e.PCatId == pcatid select
e.PcatDesc).SingleOrDefault();
        foreach (var data in users)
        {

            var prods = (from c in db.Products where c.ProductGroupId == data.pgid
orderby c.ProductName select c).ToList();
            foreach (var dt in prods)
            {
                var pnos = (from d in db.productSellers where d.ProductId == dt.ProductId
select d).Count();
                _user.Add(new EteProductGroup
                {

                    pgid = (int)dt.ProductId,
                    PCatId = dt.ProductGroupId,
                    pgroupname = data.pgroupname,
                    pgroupdescription = dt.ProductName,
                    isActive = true,
                    createdOn = dt.CreatedOn,
                    lbtnimage = dt.lbtnimage,
                    UserId = (int)data.UserId,
                    PNos = pnos,
                    Catname = pcat,
                    Price=(decimal)dt.Price,

                });
            }
        }
    }
}

```

```

        return _user;
    }
}
public List<EteProductGroup> GetProductsListbypid(string Curl, decimal pid)
{
    List<EteProductGroup> _user = new List<EteProductGroup>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        db.Connection.Open();
        int pgid = (new tblPagess()).GetPagelId(Curl);
        //foreach (var data in users)
        {

            var prods = (from c in db.Products where c.ProductId == pid orderby
c.ProductName select c).ToList();
            foreach (var dt in prods)
            {
                var users = (from c in db.ProductGroups where c.pgid == dt.ProductGroupId
orderby c.pgroupname select c).SingleOrDefault();
                var pcat = (from e in db.ProductCats where e.PCatId ==
dt.ProductGroupCategory select e.PcatDesc).SingleOrDefault();
                var pnos = (from d in db.productSellers where d.ProductId == dt.ProductId
select d).Count();
                _user.Add(new EteProductGroup
                {

                    pgid = (int)dt.ProductId,
                    PCatId = dt.ProductGroupId,
                    pgroupname = users.pgroupname,
                    pgroupdescription = dt.ProductName,
                    isActive = true,
                    createdOn = dt.CreatedOn,
                    lbtnimage = dt.lbtnimage,
                    UserId = (int)users.UserId,
                    PNos = pnos,
                    Catname = pcat,
                    Price = (decimal)dt.Price,

                });
            }
        }

        return _user;
    }
}
public List<EteProductGroup> GetProductGroupsbyCatId(string Curl, int pcatid)
{
    List<EteProductGroup> _user = new List<EteProductGroup>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        db.Connection.Open();

```

```

int pgid = (new tblPagess()).GetPagelId(Curl);
var users = (from c in db.ProductGroups where c.PCatId == pcatid orderby
c.PCatId,pgid select c ).ToList();

foreach (var data in users)
{
    _user.Add(new EteProductGroup
    {
        pgid = data.pgid,
        PCatId = data.PCatId,
        pgroupname = data.pgroupname,
        pgroupdescription = "",
        isActive = true,
        createdOn = data.createdOn,
        lbtnimage = data.lbtnimage,
        UserId = (int)data.UserId,
    });
    var prods = (from c in db.Products where c.ProductGroupId == data.pgid
orderby c.ProductName select c).ToList();
    foreach(var dt in prods){
        var pnos = (from d in db.productSellers where d.ProductId == dt.ProductId
select d).ToList();

        _user.Add(new EteProductGroup
        {

            pgid =(int) dt.ProductId,
            PCatId = dt.ProductGroupId,
            pgroupname = data.pgroupname,
            pgroupdescription = dt.ProductName,
            isActive = true,
            createdOn = dt.CreatedOn,
            lbtnimage = dt.lbtnimage,
            UserId = (int)data.UserId,
        });
    }
}

return _user;
}
}
public EteProductGroup GetProductGroupbyId(string Curl, int pgid)
{
    EteProductGroup _user = new EteProductGroup();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        db.Connection.Open();
        int pageid = (new tblPagess()).GetPagelId(Curl);
        var users = (from c in db.ProductGroups where c.pgid == pgid select
c).SingleOrDefault();

```

```

        var cat = (from d in db.ProductCats where d.PCatId == users.PCatId select
d.PcatDesc).SingleOrDefault();
        _user.PCatId = Convert.ToInt16(users.PCatId);
        _user.pgroupname = users.pgroupname;
        _user.pgroupdescription = users.pgroupdescription;
        _user.pgid = users.pgid;
        _user.lbtnimage = users.lbtnimage;
        _user.pgid = pgid;
        _user.isActive = Convert.ToBoolean( users.isActive);
        _user.Catname = cat;

        return _user;
    }
}
public bool ProductNameExists(string name)
{
    bool exists = false;

    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        db.Connection.Open();

        var _cust = (from c in db.Products where c.ProductName == name select
c).SingleOrDefault();

        if (_cust != null)
        {
            exists = true;
        }

    }

    return exists;
}
public List<GetSearchFinalProductsResult> GetSearchFinalProducts(string SrchStrng)
{
    List<GetSearchFinalProductsResult> _user = new
List<GetSearchFinalProductsResult>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        _user = db.GetSearchFinalProducts(SrchStrng).ToList();
        foreach (var data in _user)
        {
            byte[] bytes = (byte[])data.sellerimage.ToArray();
            string base64String = Convert.ToBase64String(bytes, 0, bytes.Length);
            data.ImageURL = "data:image/png;base64," + base64String;

        }
        return _user;
    }
}

public string InsertPhoto(EteproductImages data)

```

```

{
    string Photold = "", Thumbld="";
    bool result = false;
    if (data.PSId == 0) { return ""; }
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();
        int srno = GetNextSrNo(data.PSId);
        Photold = "images/vendor/" + Convert.ToInt64(data.PSId).ToString() + "-" +
data.ProductId.ToString() + "-" + srno.ToString()+".png";
        Thumbld = Photold+"thumb/" + Photold;
        try
        {
            productSellerImage _cs = new productSellerImage
            {
                PSID=data.PSId,
                ProductId = data.ProductId,
                SellerId=data.SellerId,
                srno = srno,
                Images=data.Images,
                ImageURL=data.ImageUrls,
                ImagePath = Photold,
                ImageThumb = Photold,

            };
            db.productSellerImages.InsertOnSubmit(_cs);

            db.SubmitChanges();
            db.Transaction.Commit();

            result = true;
        }
        catch (Exception exc)
        {
            db.Transaction.Rollback();
            throw exc;
        }
    }
    if (result) {
        return Photold;
    }
    else { return ""; }
}
public bool InsertproductSellerServices(EteproductSellerServices data)
{
    bool result = false;
    //Eteproduct _user = new Eteproduct();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try

```

```

    {
        productSellerService _cs = new productSellerService
        {
            PSId=data.PSId,
            productid = data.productid,
            Remark=data.Remark,
            PrdPrice=data.PrdPrice,
            PrdUnit=data.PrdUnit,
            active=data.active,
            userid=data.userid,

        };
        db.productSellerServices.InsertOnSubmit(_cs);

        db.SubmitChanges();
        db.Transaction.Commit();

        result = true;

    }
    catch (Exception exc)
    {
        db.Transaction.Rollback();
        throw exc;
    }
}
return result;
}
}

public bool Insert(Eteproduct data)
{
    bool result = false;
    //Eteproduct _user = new Eteproduct();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {
            Product _cs = new Product
            {
                ProductId =(int) GetNextProductId(),
                ProductName = data.ProductName,

                ProductGroupId = data.ProductGroupId,
                ProductGroupCategory = data.ProductGroupCategory,

                //Type = data.Type,

                Price = (decimal)data.Price,

                //Freq = data.Freq,

                //Period = data.Period,
            }

```

```

        CurrencyId = (int)data.CurrencyId,
        AnalysisCode =( data.AnalysisCode),
        CreatedOn = DateTime.Now,
        UpdatedOn = DateTime.Now,
        UserId = data.UserId,
        //PrId = data.PrId,
        MinOrder = (int)data.MinOrder,
        DelTime = (int)data.DelTime,

};
db.Products.InsertOnSubmit(_cs);

db.SubmitChanges();
db.Transaction.Commit();

    result = true;
}
catch (Exception exc)
{
    db.Transaction.Rollback();
    throw exc;
}
}

return result;
}
//public bool InsertProductSeller(EteproductSeller data)
//{
//    bool result = false;
//    //Eteproduct _user = new Eteproduct();
//    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
//    {
//        db.Connection.Open();
//        db.Transaction = db.Connection.BeginTransaction();

//        try
//        {
//            productSeller _cs = new productSeller
//            {
//                PSId = data.PSId,
//                ProductId = data.ProductId,
//                SellerId = data.SellerId,
//                SellerDescription = data.SellerDescription,
//                Price = data.Price,
//                CurrencyId = data.CurrencyId,
//                PriceUnits = data.PriceUnits,
//                PriceUomId = data.PriceUomId,
//                PricePackId = data.PricePackId,
//                KeyWords = data.KeyWords,
//                CreatedOn = DateTime.Now,
//                UpdatedOn = data.UpdatedOn,

```



```

//      UserId = data.UserId,
//      PrId = data.PrId,
//      MinOrder = data.MinOrder,
//      MinOrderUom = data.MinOrderUom,
//      DelTime = data.DelTime,
//      DelTimeType = data.DelTimeType,
//      LinkPSId = data.LinkPSId,
//      ProductDescription = data.ProductDescription,
//      DiscountLink = data.DiscountLink,
//      BrandName = data.BrandName,
//      MaterialCare = data.MaterialCare,
//      MaterialType = data.MaterialType,
//      Manufacturer = data.Manufacturer,
//      Length = data.Length,
//      MadeinCountry = data.MadeinCountry,
//      Location=data.Location,
//      GeoLocation=data.GeoLocation,

//      };
//      db.productSellers.InsertOnSubmit(_cs);

//      db.SubmitChanges();
//      db.Transaction.Commit();
//      result = true;

//      }
//      catch (Exception exc)
//      {
//          db.Transaction.Rollback();
//          throw exc;
//      }
//      }

//      return result;
//}
public bool InsertGroup(EteProductGroup data)
{
    bool result = false;
    //Eteproduct _user = new Eteproduct();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {
            ProductGroup _cs = new ProductGroup
            {
                pgid = GetNextProductGroupId(),
                pgroupname = data.pgroupname,

                PCatId = data.PCatId,

```

```

        pgroupdescription = data.pgroupdescription,
        //Period = data.Period,
        createdOn = DateTime.Now,
        UserId = data.UserId,
        isActive=true,
        lbtnimage="#",
    };
    db.ProductGroups.InsertOnSubmit(_cs);

    db.SubmitChanges();
    db.Transaction.Commit();

    result = true;
}
catch (Exception exc)
{
    db.Transaction.Rollback();
    throw exc;
}
}

return result;
}
public bool InsertproductExtProperties(EteProductGroup data)
{
    bool result = false;
    //Eteproduct _user = new Eteproduct();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {
            productExtProperty _cs = new productExtProperty
            {
                EpId = GetNextProductCatId(),
                EpDescription = data.pgroupname,
                active=data.isActive,
            };
            db.productExtProperties.InsertOnSubmit(_cs);

            db.SubmitChanges();
            db.Transaction.Commit();

            result = true;
        }
        catch (Exception exc)
        {
            db.Transaction.Rollback();
            throw exc;
        }
    }
}

```

```

    return result;
}
public bool InsertproductExtPrOptions(EteProductGroup data)
{
    bool result = false;
    //Eteproduct _user = new Eteproduct();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {
            productExtPrOption _cs = new productExtPrOption
            {
                EpId =(int) data.ProductId,
                EpOptId=data.pgId,
                EpOPtDescr = data.pgrouppname,
                active = data.isActive,
            };
            db.productExtPrOptions.InsertOnSubmit(_cs);

            db.SubmitChanges();
            db.Transaction.Commit();

            result = true;
        }
        catch (Exception exc)
        {
            db.Transaction.Rollback();
            throw exc;
        }
    }

    return result;
}
public bool InsertproductSellerExtProperties(EteProductGroup data)
{
    bool result = false;
    //Eteproduct _user = new Eteproduct();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {
            productSellerExtProperty _cs = new productSellerExtProperty
            {PSId=data.ProductId,
                EpId = data.PCatId,
                EpOptId = data.pgId,
                Remark = data.pgrouppname,
                active = data.isActive,
            };

```

```

};
db.productSellerExtProperties.InsertOnSubmit(_cs);

db.SubmitChanges();
db.Transaction.Commit();

    result = true;
}
catch (Exception exc)
{
    db.Transaction.Rollback();
    throw exc;
}
}

return result;
}

public bool InsertCat(Eteproductcat data)
{
    bool result = false;
    //Eteproduct _user = new Eteproduct();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {
            ProductCat _cs = new ProductCat
            {
                PCatId = GetNextProductCatId(),
                PcatDesc = data.pcatdesc,

                //Period = data.Period,
                CreatedOn = DateTime.Now,
                UserId = data.UserID,

                lbtnimage = "#",
            };
            db.ProductCats.InsertOnSubmit(_cs);

            db.SubmitChanges();
            db.Transaction.Commit();

            result = true;
        }
        catch (Exception exc)
        {
            db.Transaction.Rollback();
            throw exc;
        }
    }
}

```

```

        return result;
    }
    public bool UpdateGroup(EteProductGroup data)
    {
        bool result = false;
        EteProductGroup odata = new EteProductGroup();
        using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        {
            db.Connection.Open();
            db.Transaction = db.Connection.BeginTransaction();

            try
            {

                var _user = db.ProductGroups.Where(e => e.pgid ==
data.pgid).SingleOrDefault();

                if (_user == null)
                {
                    throw (new ApplicationException("Unable to find the User record"));
                }

                _user.pgroupname = data.pgroupname;
                _user.PCatId = data.PCatId;
                _user.pgroupdescription = data.pgroupdescription;

                db.SubmitChanges();
                db.Transaction.Commit();
                result = true;

            }

            catch (Exception exc)
            {
                db.Transaction.Rollback();

                throw exc;
            }
        }
        return result;
    }
    public bool UpdateCat(Eteproductcat data)
    {
        bool result = false;
        Eteproductcat odata = new Eteproductcat();
        using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        {
            db.Connection.Open();
            db.Transaction = db.Connection.BeginTransaction();

            try
            {

```

```

        var _user = db.ProductCats.Where(e => e.PCatId ==
data.pcatid).SingleOrDefault();

        if (_user == null)
        {
            throw (new ApplicationException("Unable to find the User record"));
        }

        _user.PCatId = data.pcatid;
        _user.UserId = data.UserID;
        _user.PcatDesc = data.pcatdesc;

        db.SubmitChanges();
        db.Transaction.Commit();
        result = true;

    }

    catch (Exception exc)
    {
        db.Transaction.Rollback();

        throw exc;
    }
}
return result;
}
public bool Update(Eteproduct data,bool upd=true)
{
    bool result = false;
    //string mess;
    Eteproduct odata = new Eteproduct();

    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {
            if (upd)
            {
                var _user = db.Products.Where(e => e.ProductId ==
data.ProductId).SingleOrDefault();

                if (_user == null)
                {
                    throw (new ApplicationException("Unable to find the User record"));
                }

                _user.ProductName = data.ProductName;

```

```

_user.ProductGroupId = data.ProductGroupId;
_user.ProductGroupCategory = data.ProductGroupCategory;
_user.Price = (decimal)data.Price;
//_user.SellerId = data.SellerId;
_user.CurrencyId = (int)data.CurrencyId;
_user.AnalysisCode =( data.AnalysisCode);
//_user.CreatedOn = data.CreatedOn;
_user.UpdatedOn = DateTime.Now;
_user.UserId = data.UserId;
//_user.PrlId = data.PrlId;
_user.MinOrder = (int)data.MinOrder;
_user.DeITIme = (int)data.DeITIme;

```

```

var seller = (from c in db.productSellers where c.ProductId == data.ProductId
&& c.SellerId == data.SellerId select c).SingleOrDefault();

```

```

        seller.SellerDescription = data.ProductName;
        seller.SellerId = data.SellerId;
        seller.KeyWords = data.AnalysisCode;
        seller.Price = data.Price;
        seller.MinOrder = data.MinOrder;
        seller.DeITIme = data.DeITIme;
        seller.PrlId = Convert.ToInt16(data.PrlId);
        seller.UpdatedOn = DateTime.Now;
        seller.UserId = data.UserId;
        seller.DeITIme = data.DeITIme;
        seller.DeITimeType = data.DeITimeType;
    }
    else
    {
        if (!upd)
        {
            productSeller _seller = new productSeller
            {
                PSId = GetNextPSId(),
                ProductId = data.ProductId,
                SellerId=data.SellerId,
                Price = data.Price,
                MinOrder = data.MinOrder,
                DeITIme = data.DeITIme,
                PrId = Convert.ToInt16(data.PrlId),
                SellerDescription = data.ProductName,
                UpdatedOn = DateTime.Now,
                UserId = data.UserId,
                CurrencyId = data.CurrencyId,
                CreatedOn = DateTime.Now,
                KeyWords = data.AnalysisCode,
                PriceUnits = data.PriceUnits,
                PriceUomId = data.PriceUomId,
                PricePackId = data.PricePackId,
                MinOrderUom = data.MinOrderUom,
                DeITimeType = data.DeITimeType,
            };

            db.productSellers.InsertOnSubmit(_seller);

```

```

        }
    }
    db.SubmitChanges();
    db.Transaction.Commit();
    result = true;

}
catch (Exception exc)
{
    db.Transaction.Rollback();

    throw exc;
}

//mess = BalAuditTrail.GetAuditXml("Product", odata, data);
//EteAuditTrail at = new EteAuditTrail();
//at = new EteAuditTrail
//{
//    CreatedOn = odata.createdOn,
//    UserId = data.UId,
//    ObjectId = data.PagId,
//    Category = "Edit",
//    Message = mess,
//    AuditDate = DateTime.Now,
//};
//bool res = new BalAuditTrail().Insert(at);
}

return result;
}
public bool UpdateProductSeller(EteproductSeller data)
{
    bool result = false;
    //string mess;
    Eteproduct odata = new Eteproduct();

    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {
            var _user = db.productSellers.Where(e => e.PSId ==
data.PSId).SingleOrDefault();

            if (_user == null)
            {
                throw (new ApplicationException("Unable to find the User record"));
            }
        }
    }
}

```



```
_user.PSId = data.PSId;
_user.ProductId = data.ProductId;
_user.SellerId = data.SellerId;
_user.SellerDescription = data.SellerDescription;
_user.Price = data.Price;
_user.CurrencyId = data.CurrencyId;
_user.PriceUnits = data.PriceUnits;
_user.PriceUomId = data.PriceUomId;
_user.PricePackId = data.PricePackId;
_user.KeyWords = data.KeyWords;
//_user.CreatedOn = data.CreatedOn;
_user.UpdatedOn = data.UpdatedOn;
_user.UserId = data.UserId;
_user.PriId = data.PriId;
_user.MinOrder = data.MinOrder;
_user.MinOrderUom = data.MinOrderUom;
_user.DeITime = data.DeITime;
_user.DeITimeType = data.DeITimeType;
```

```
    _user.LinkPSId=data.LinkPSId;
    _user.ProductDescription = data.ProductDescription;
    _user.DiscountLink=data.DiscountLink;
    _user.BrandName=data.BrandName;
    _user.MaterialCare=data.MaterialCare;
    _user.MaterialType =data.MaterialType;
    _user.Length=data.Length;
    _user.MadeinCountry = data.MadeinCountry;
```

```
db.SubmitChanges();
db.Transaction.Commit();
result = true;
```

```
}
catch (Exception exc)
{
    db.Transaction.Rollback();

    throw exc;
}
```

```
//mess = BalAuditTrail.GetAuditXml("ProductSeller", odata, data);
//EteAuditTrail at = new EteAuditTrail();
//at = new EteAuditTrail
//{
//    CreatedOn = odata.createdOn,
//    UserId = data.UId,
//    Objectid = data.Pageld,
//    Category = "Edit",
//    Message = mess,
//    AuditDate = DateTime.Now,
//};
//bool res = new BalAuditTrail().Insert(at);
```

```
}
```

```

    return result;
}
public bool UpdateSellerProduct(decimal ProductId, int SellerId)
{

    bool result = false;
    //string mess;
    Eteproduct odata = new Eteproduct();

    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {

            var _user = db.SellerProducts.Where(e => e.ProductId == ProductId &&
e.SellerId==SellerId).SingleOrDefault();

            if (_user == null)
            {
                SellerProduct _cs = new SellerProduct
                {
                    ProductId = ProductId,
                    SellerId = SellerId,

                };
                db.SellerProducts.InsertOnSubmit(_cs);
            }
            else
            {
                _user.ProductId = ProductId;
                _user.SellerId = SellerId;
            }

            db.SubmitChanges();
            db.Transaction.Commit();
            result = true;

        }
        catch (Exception exc)
        {
            db.Transaction.Rollback();

            throw exc;
        }

        //mess = BalAuditTrail.GetAuditXml("ProductSeller", odata, data);
        //EteAuditTrail at = new EteAuditTrail();
        //at = new EteAuditTrail
        //{

```

```

        // CreatedOn = odata.createdOn,
        // UserId = data.UId,
        // Objectld = data.Pageld,
        // Category = "Edit",
        // Message = mess,
        // AuditDate = DateTime.Now,
        //};
        //bool res = new BalAuditTrail().Insert(at);
    }

    return result;
}
public List<EteCurrency> GetProductCurrencys(string Curl)
{
    List<EteCurrency> _user = new List<EteCurrency>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        int pgid = (new tblPagess()).GetPageld(Curl);
        var users = (from c in db.Currencies select c).ToList();
        foreach (var data in users)
        {
            _user.Add(new EteCurrency
            {
                CId = data.CId,
                CShortCode = data.CShortCode,
                CDescription = data.CDescription,
                createdon = data.createdon,
                updatedon = data.updatedon,
                userid = data.userid,
            });
        }

        return _user;
    }
}
public EteCurrency GetProductCurrencybyld(int id)
{
    EteCurrency _user = new EteCurrency();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var data = (from c in db.Currencies where c.CId == id select c).SingleOrDefault();

        _user.CId = data.CId;
        _user.CShortCode = data.CShortCode ;
        _user.CDescription = data.CDescription ;
        _user.createdon = data.createdon ;
        _user.updatedon = data.updatedon ;
    }
}

```

```

        _user.userid      =      data.userid      ;

    }
    return _user;
}

public List<Etetblpacking> GetProducttblpackings(string Curl)
{
    List<Etetblpacking> _user = new List<Etetblpacking>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        db.Connection.Open();
        int pgid = (new tblPagess()).GetPagelId(Curl);
        var users = (from c in db.tblpackings select c).ToList();
        foreach (var data in users)
        {
            _user.Add(new Etetblpacking
            {
                PackID = data.PackID,

                PackShort = data.PackShort,

                PackDescription = data.PackDescription,

                CreatedOn = data.CreatedOn,

                UpdatedOn = data.UpdatedOn,

                UserId = data.UserId,

            });
        }

        return _user;
    }
}

public Etetblpacking GetProducttblpackingbyId(int id)
{
    Etetblpacking _user = new Etetblpacking();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        db.Connection.Open();
        var data = (from c in db.tblpackings where c.PackID== id select
c).SingleOrDefault();

        _user.PackID = data.PackID;

        _user.PackShort = data.PackShort;

        _user.PackDescription = data.PackDescription;
    }
}

```

```

        _user.CreatedOn = data.CreatedOn;

        _user.UpdatedOn = data.UpdatedOn;

        _user.UserId = data.UserId;
    }

    return _user;
}
public List<Etetbluom> GetProducttbluoms(string Curl)
{
    List<Etetbluom> _user = new List<Etetbluom>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        db.Connection.Open();
        int pgid = (new tblPagess()).GetPageId(Curl);
        var users = (from c in db.tbluoms select c).ToList();
        foreach (var data in users)
        {
            _user.Add(new Etetbluom
            {

                UomId = data.UomId,
                UomShort = data.UomShort,
                UomDescription = data.UomDescription,
                CreatedOn = data.CreatedOn,
                UpdatedOn = data.UpdatedOn,
                UserId = data.UserId,

            });
        }

        return _user;
    }
}
public Etetbldeltime GetProductbldeltimebyId(int id)
{
    Etetbldeltime _user = new Etetbldeltime();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        db.Connection.Open();
        var data = (from c in db.tbldeltimes where c.DelTimeType == id select
c).SingleOrDefault();

        _user.DelTimeType = data.DelTimeType;
        _user.DelTimeDescription = data.DelTimeDescription;
        _user.CreatedOn = data.CreatedOn;
        _user.UpdatedOn = data.UpdatedOn;
        _user.UserId = data.UserId;
    }
}

```

```

    }

    return _user;
}
public List<Etetbldeltime> GetProducttbldeltimes(string Curl)
{
    List<Etetbldeltime> _user = new List<Etetbldeltime>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        db.Connection.Open();
        int pgid = (new tblPagess()).GetPageld(Curl);
        var users = (from c in db.tbldeltimes select c).ToList();
        foreach (var data in users)
        {
            _user.Add(new Etetbldeltime
            {

                DelTimeType = data.DelTimeType,
                DelTimeDescription = data.DelTimeDescription,
                CreatedOn = data.CreatedOn,
                UpdatedOn = data.UpdatedOn,
                UserId = data.UserId,

            });
        }

        return _user;
    }
}
public Etetbluom GetProducttbluombyld(int id)
{
    Etetbluom _user = new Etetbluom();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        db.Connection.Open();
        var data = (from c in db.tbluoms where c.UomId == id select c).SingleOrDefault();

        _user.UomId = data.UomId;
        _user.UomShort = data.UomShort;
        _user.UomDescription = data.UomDescription;
        _user.CreatedOn = data.CreatedOn;
        _user.UpdatedOn = data.UpdatedOn;
        _user.UserId = data.UserId;

    }

    return _user;
}

```

```

}
public List<EteCountry> GetCountryS(string Curl)
{
    List<EteCountry> _user = new List<EteCountry>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        int pgid = (new tblPagess()).GetPageld(Curl);
        var users = (from c in db.countries select c).ToList();
        foreach (var data in users)
        {
            _user.Add(new EteCountry
            {
                id = data.id,
                shortcd = data.shortcd,
                countryname = data.countryname,
                IsBlocked = data.IsBlocked,
                CFlag = data.CFlag,

            });
        }

        return _user;
    }
}
public EteCountry GetCountrySbyId(int id)
{
    EteCountry _user = new EteCountry();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var data = (from c in db.countries where c.id == id select c).SingleOrDefault();

        _user.id = data.id;
        _user.shortcd = data.shortcd;
        _user.countryname = data.countryname;
        _user.IsBlocked = data.IsBlocked;
        _user.CFlag = data.CFlag;
    }

    return _user;
}
public bool InsertDelTimes(EtetblDelttime data)
{
    bool result = false;
    //Eteproduct _user = new Eteproduct();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

```

```

        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();
        GetNextProductDelTimesResult lst =
db.GetNextProductDelTimes().SingleOrDefault();
        int DelTimeTypes =(int) lst.NextDelTimeid;

        try
        {
            tblvertime _cs = new tblvertime
            {
                DelTimeType=DelTimeTypes,

                DelTimeDescription = data.DelTimeDescription,
                CreatedOn = DateTime.Now,
                UpdatedOn = data.UpdatedOn,
                UserId = data.UserId,

            };
            db.tblvertimes.InsertOnSubmit(_cs);

            db.SubmitChanges();
            db.Transaction.Commit();

            result = true;
        }
        catch (Exception exc)
        {
            db.Transaction.Rollback();
            throw exc;
        }
    }

    return result;
}
public bool UpdateDelTimes(Etettblvertime data)
{
    bool result = false;
    Etettblvertime odata = new Etettblvertime();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {

            var _user = db.tblvertimes.Where(e => e.DelTimeType ==
data.DelTimeType).SingleOrDefault();

            if (_user == null)
            {
                throw (new ApplicationException("Unable to find the User record"));
            }

```



```

        _user.DelTimeType = data.DelTimeType;
        _user.DelTimeDescription = data.DelTimeDescription;
        // _user.CreatedOn = data.CreatedOn;
        _user.UpdatedOn = data.UpdatedOn;
        _user.UserId = data.UserId;

        db.SubmitChanges();
        db.Transaction.Commit();
        result = true;
    }

    catch (Exception exc)
    {
        db.Transaction.Rollback();

        throw exc;
    }
}
return result;
}
public bool InsertUOM(Etetbluom data)
{
    bool result = false;
    //Eteproduct _user = new Eteproduct();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();
        GetNextProductUOMResult lst = db.GetNextProductUOM().SingleOrDefault();
        int UomIds = (int)lst.NextUOMid;

        try
        {
            tbluom _cs = new tbluom
            {
                UomId = UomIds,
                UomShort = data.UomShort,
                UomDescription = data.UomDescription,
                CreatedOn = DateTime.Now,
                UpdatedOn = data.UpdatedOn,
                UserId = data.UserId,
            };
            db.tbluoms.InsertOnSubmit(_cs);

            db.SubmitChanges();
            db.Transaction.Commit();

            result = true;
        }
        catch (Exception exc)
        {

```

```

        db.Transaction.Rollback();
        throw exc;
    }
}

return result;
}
public bool InsertShortList(EteUserShortList data, bool stype)
{
    bool result = false;
    //Eteproduct _user = new Eteproduct();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {
            var user = (from c in db.UserShortlists where c.UserId == data.UserId &&
c.PSId == data.PSId select c).SingleOrDefault();
            if (user == null)
            {
                UserShortlist _cs = new UserShortlist
                {
                    UserId = data.UserId,
                    PSId = data.PSId,
                    UId = data.UId,
                    CreatedOn = DateTime.Now,
                    Active = data.Active,
                    SLType = stype,
                };
                db.UserShortlists.InsertOnSubmit(_cs);

                db.SubmitChanges();
                db.Transaction.Commit();
            }
            result = true;
        }
        catch (Exception exc)
        {
            db.Transaction.Rollback();
            throw exc;
        }
    }

    return result;
}
public bool InsertCartList(EteUserCart data)
{
    bool result = false;
    //Eteproduct _user = new Eteproduct();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

```

```

        try
        {
            var user = (from c in db.UserCarts where c.UCId == data.UCId select
c).SingleOrDefault();
            if (user == null)
            {
                UserCart _cs = new UserCart
                {UCId=GetNextUCId(),
                UserId = data.UserId,
                PSId = data.PSId,
                UId = data.UId,
                CreatedOn = DateTime.Now,
                Active = data.Active,
                PayId=data.PayId,

                };
                db.UserCarts.InsertOnSubmit(_cs);

                db.SubmitChanges();
                db.Transaction.Commit();
            }
            result = true;
        }
        catch (Exception exc)
        {
            db.Transaction.Rollback();
            throw exc;
        }
    }

    return result;
}

public bool UpdateUOM(Etetbluom data)
{
    bool result = false;
    Etetbldetime odata = new Etetbldetime();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {
            var _user = db.tbluoms.Where(e => e.UomId == data.UomId).SingleOrDefault();

            if (_user == null)
            {
                throw (new ApplicationException("Unable to find the User record"));
            }

            _user.UomId = data.UomId;
            _user.UomShort = data.UomShort;

```

```

        _user.UomDescription = data.UomDescription;
        //_user.CreatedOn = data.CreatedOn;
        _user.UpdatedOn = data.UpdatedOn;
        _user.UserId = data.UserId;

        db.SubmitChanges();
        db.Transaction.Commit();
        result = true;
    }

    catch (Exception exc)
    {
        db.Transaction.Rollback();

        throw exc;
    }
}
return result;
}
public bool InsertCurrency(EteCurrency data)
{
    bool result = false;
    //Eteproduct _user = new Eteproduct();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();
        GetNextProductCurrencyResult lst =
db.GetNextProductCurrency().SingleOrDefault();
        int UomIds = (int)lst.NextCurrencyid;

        try
        {
            Currency _cs = new Currency
            {
                CId = data.CId,
                CShortCode = data.CShortCode,
                CDescription = data.CDescription,
                createdon = DateTime.Now,
                updatedon = data.updatedon,
                userid = data.userid,
            };

            db.Currencies.InsertOnSubmit(_cs);

            db.SubmitChanges();
            db.Transaction.Commit();

            result = true;
        }
        catch (Exception exc)

```

```

        {
            db.Transaction.Rollback();
            throw exc;
        }
    }

    return result;
}
public bool UpdateCurrency(EteCurrency data)
{
    bool result = false;
    EteCurrency odata = new EteCurrency();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {
            var _user = db.Currencies.Where(e => e.CId == data.CId).SingleOrDefault();

            if (_user == null)
            {
                throw (new ApplicationException("Unable to find the User record"));
            }

            _user.CId = data.CId;
            _user.CShortCode = data.CShortCode;
            _user.CDescription = data.CDescription;
            //_user.createdon = data.createdon;
            _user.updatedon = data.updatedon;
            _user.userid = data.userid;

            db.SubmitChanges();
            db.Transaction.Commit();
            result = true;

        }

        catch (Exception exc)
        {
            db.Transaction.Rollback();

            throw exc;
        }
    }
    return result;
}
public bool InsertPacking(EtePacking data)
{
    bool result = false;
    //EteProduct _user = new EteProduct();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))

```

```

    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();
        GetNextProductPackingResult lst =
db.GetNextProductPacking().SingleOrDefault();
        int UomIds = (int)lst.NextPackID;

        try
        {
            tblpacking _cs = new tblpacking
            {
                PackID = data.PackID,

                PackShort = data.PackShort,

                PackDescription = data.PackDescription,

                CreatedOn = DateTime.Now,

                UpdatedOn = data.UpdatedOn,

                UserId = data.UserId,

            };
            db.tblpackings.InsertOnSubmit(_cs);

            db.SubmitChanges();
            db.Transaction.Commit();

            result = true;
        }
        catch (Exception exc)
        {
            db.Transaction.Rollback();
            throw exc;
        }
    }

    return result;
}
public bool UpdatePacking(Etetblpacking data)
{
    bool result = false;
    Etetblpacking odata = new Etetblpacking();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {

            var _user = db.tblpackings.Where(e => e.PackID ==
data.PackID).SingleOrDefault();

```

```

        if (_user == null)
        {
            throw (new ApplicationException("Unable to find the User record"));
        }

        _user.PackID = data.PackID;

        _user.PackShort = data.PackShort;

        _user.PackDescription = data.PackDescription;

        //_user.CreatedOn = data.CreatedOn;

        _user.UpdatedOn = data.UpdatedOn;

        _user.UserId = data.UserId;

        db.SubmitChanges();
        db.Transaction.Commit();
        result = true;
    }

    catch (Exception exc)
    {
        db.Transaction.Rollback();

        throw exc;
    }
}
return result;
}
public bool InsertproductSellerStocks(EteproductSellerStocks data, bool upd = true)
{
    bool result = false;
    //string mess;
    Eteproduct odata = new Eteproduct();

    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        //try
        {
            if (upd)
            {
                var seller = (from c in db.productSellerStocks where c.PSId == data.PSId &&
c.EpId == data.EpId && c.EpOptId == data.EpOptId && c.colid==data.colid select
c).SingleOrDefault();
                if (seller != null) {

```

```

seller.PSId = data.PSId;
seller.EpId = data.EpId;
seller.EpOptId = data.EpOptId;
seller.colId = data.colId;
seller.Stock = data.Stock;
seller.Remark = data.Remark;
seller.active = data.active;
}
else
{
    productSellerStock _seller = new productSellerStock
    {
        PSId = data.PSId,
        EpId = data.EpId,
        EpOptId = data.EpOptId,
        colId = data.colId,
        Stock = data.Stock,
        Remark = data.Remark,
        active = data.active,
    };

    db.productSellerStocks.InsertOnSubmit(_seller);
}
else
{
    if (!upd)
    {
        productSellerStock _seller = new productSellerStock
        {
            PSId = data.PSId,
            EpId = data.EpId,
            EpOptId = data.EpOptId,
            colId = data.colId,
            Stock = data.Stock,
            Remark = data.Remark,
            active = data.active,
        };

        db.productSellerStocks.InsertOnSubmit(_seller);
    }
    db.SubmitChanges();
    db.Transaction.Commit();
    result = true;
}
//catch (Exception exc)
//{
//    db.Transaction.Rollback();

//    throw exc;
//}

```



```

        //mess = BalAuditTrail.GetAuditXml("Product", odata, data);
        //EteAuditTrail at = new EteAuditTrail();
        //at = new EteAuditTrail
        //{
        //    CreatedOn = odata.createdOn,
        //    UserId = data.UId,
        //    Objectid = data.Pageld,
        //    Category = "Edit",
        //    Message = mess,
        //    AuditDate = DateTime.Now,
        //};
        //bool res = new BalAuditTrail().Insert(at);
    }

    return result;
}
public bool InsertProductSeller(EteproductSeller data, bool upd = true)
{
    bool result = false;
    //string mess;
    Eteproduct odata = new Eteproduct();

    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        //try
        {
            if (upd)
            {
                var seller = (from c in db.productSellers where c.PSId==data.PSId select
c).SingleOrDefault();

                seller.SellerDescription = data.SellerDescription;
                seller.SellerId = data.SellerId;
                seller.KeyWords = data.AnalysisCode;
                seller.Price = data.Price;
                seller.MinOrder = data.MinOrder;
                seller.DelTIme = data.DelTIme;
                seller.Prid = data.Prid;
                seller.UpdatedOn = DateTime.Now;
                seller.UserId = data.UserId;
                seller.DelTIme = data.DelTIme;
                seller.DelTimeType = data.DelTimeType;
                seller.LinkPSId = data.LinkPSId;
                seller.ProductDescription = data.ProductDescription;
                seller.DiscountLink = data.DiscountLink;
                seller.BrandName = data.BrandName;
            }
        }
    }
}

```

```

seller.MaterialCare = data.MaterialCare;
seller.MaterialType = data.MaterialType;
seller.Length = data.Length;
seller.MadeinCountry = data.MadeinCountry;
seller.Location = data.Location;
seller.GeoLocation = data.GeoLocation;
seller.Manufacturer = data.Manufacturer;
}
else
{
    if (!upd)
    {
        productSeller _seller = new productSeller
        {
            PSId = GetNextPSId(),
            ProductId = data.ProductId,
            SellerId = data.SellerId,
            Price = data.Price,
            MinOrder = data.MinOrder,
            DelTime = data.DelTime,
            PrId = data.PrId,
            SellerDescription = data.SellerDescription,
            UpdatedOn = DateTime.Now,
            UserId = data.UserId,
            CurrencyId = data.CurrencyId,
            CreatedOn = DateTime.Now,
            Keywords = data.AnalysisCode,
            PriceUnits = data.PriceUnits,
            PriceUomId = data.PriceUomId,
            PricePackId = data.PricePackId,
            MinOrderUom = data.MinOrderUom,
            DelTimeType = data.DelTimeType,
            LinkPSId=0,
            ProductDescription = data.SellerDescription,
            DiscountLink=data.DiscountLink,
            BrandName=data.BrandName,
            MaterialCare=data.MaterialCare,
            MaterialType = data.MaterialType,
            Length=data.Length,
            MadeinCountry=data.MadeinCountry,
            Location=data.Location,
            GeoLocation=data.GeoLocation,
            Manufacturer=data.Manufacturer,
        };

        db.productSellers.InsertOnSubmit(_seller);
    }
}
db.SubmitChanges();
db.Transaction.Commit();
result = true;
}
//catch (Exception exc)
//{

```

```

// db.Transaction.Rollback();

// throw exc;
//}

//mess = BalAuditTrail.GetAuditXml("Product", odata, data);
//EteAuditTrail at = new EteAuditTrail();
//at = new EteAuditTrail
//{
// CreatedOn = odata.createdOn,
// UserId = data.UId,
// ObjectId = data.PagId,
// Category = "Edit",
// Message = mess,
// AuditDate = DateTime.Now,
//};
//bool res = new BalAuditTrail().Insert(at);

}

return result;
}
public bool InsertEmail(EteTblUser_emails data)
{
bool result = false;
//Eteproduct _user = new Eteproduct();
using (var db = new DowPortalDataContext(Configuration.ConnectionString))
{
db.Connection.Open();
db.Transaction = db.Connection.BeginTransaction();
int nextid = (from c in db.tblUser_Emails orderby c.EmailId descending select
c.EmailId).FirstOrDefault();
try
{
tblUser_Email _cs = new tblUser_Email
{
EmailId = nextid+1,
EMSentUserId = data.EMSentUserId,
EMRcvdUserId = data.EMRcvdUserId,
EmSubject = data.EmSubject,
EmBody = data.EmBody,
CreatedOn = DateTime.Now,

};
db.tblUser_Emails.InsertOnSubmit(_cs);

db.SubmitChanges();
db.Transaction.Commit();

result = true;
}
catch (Exception exc)
{

```

```

        db.Transaction.Rollback();
        throw exc;
    }
}

return result;
}
}
}

```

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using DowBEntity;
using DowDAL;
namespace DowBAL
{
    public class Roles
    {
        public static bool RoleNameExists(int Id)
        {
            bool exists = false;

            using (var db = new DowPortalDataContext(Configuration.ConnectionString))
            {
                db.Connection.Open();

                var _cust = (from c in db.Roles where c.Id == Id select c).SingleOrDefault();

                if (_cust != null)
                {
                    exists = true;
                }
            }

            return exists;
        }
        public static int GetRolebyGroupId(int GroupId)
        {
            int roldid = 0;

            using (var db = new DowPortalDataContext(Configuration.ConnectionString))
            {
                db.Connection.Open();

```

```

        var _cust = (from c in db.tblPermissionGroups where c.GroupId == GroupId select
c).SingleOrDefault();

        if (_cust != null)
        {
            roldid =Convert.ToInt32( _cust.RoleId);
        }

    }

    return roldid;
}
//public int? GetNextRoleId()
//{

//    int? result = 0;
//    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
//    {
//        GetNextRoleIdResult lst = db.GetNextRoleId().SingleOrDefault();
//        result = lst.NextRoleId;
//    }

//    return result;
//}

public bool Insert(EteRole data)
{
    bool result = false;
    EteRole _user = new EteRole();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {
            Role _cs = new Role
            {
                Id = data.Id,
                Name = data.Name,
                IsActive = data.IsActive,
                CreatedOn = data.CreatedOn,
                //UpdatedOn = data.CreatedOn
            };
            db.Roles.InsertOnSubmit(_cs);

            db.SubmitChanges();
            db.Transaction.Commit();

            result = true;
        }
        catch (Exception exc)

```

```

        {
            db.Transaction.Rollback();
            throw exc;
        }
    }

    return result;
}
public bool Update(EteRole data)
{
    bool result = false;

    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {
            var _user = db.Roles.Where(e => e.Id == data.Id).SingleOrDefault();

            if (_user == null)
            {
                throw (new ApplicationException("Unable to find the User record"));
            }

            _user.Id = data.Id;
            _user.Name = data.Name;
            _user.IsActive = data.IsActive;

            //_user.UpdatedOn = data.UpdatedOn;

            db.SubmitChanges();
            db.Transaction.Commit();

            result = true;
        }
        catch (Exception exc)
        {
            db.Transaction.Rollback();

            throw exc;
        }
    }

    return result;
}
public EteRole getRole(string name)
{
    EteRole result = new EteRole();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
    }
}

```

```

var users = (from c in db.Roles where c.Name == name select c).ToList();
foreach (var data in users)
{
    result = new EteRole
    {
        Id = data.Id,
        Name = data.Name,
        IsActive = data.IsActive,

        CreatedOn = data.CreatedOn
    };
    break;
}
}

return result;
}

public List<EteRole> GetRoles()
{
    List<EteRole> _user = new List<EteRole>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();

        var users = (from c in db.Roles select c).ToList();
        foreach (var data in users)
        {
            _user.Add(new EteRole
            {
                Id = data.Id,
                Name = data.Name,
                IsActive = data.IsActive,
                CreatedOn = data.CreatedOn
            });
        }

        return _user;
    }
}

}

}

****

using System;
using System.IO;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using DowBEntity;

```

```

using DowDAL;
using System.Xml;
using System.Xml.Serialization;
using System.Xml.XmlConfiguration;
namespace DowBAL
{

    public class Scratchpads
    {

        public bool ScratchpadNameExists(string name)
        {
            bool exists = false;

            using (var db = new DowPortalDataContext(Configuration.ConnectionString))
            {

                db.Connection.Open();

                var _cust = (from c in db.Scratchpads where c.Content == name select
c).SingleOrDefault();

                if (_cust != null)
                {
                    exists = true;
                }

            }

            return exists;
        }

        public bool Insert(EteScratchpad data)
        {
            bool result = false;
            EteScratchpad _user = new EteScratchpad();
            using (var db = new DowPortalDataContext(Configuration.ConnectionString))
            {

                db.Connection.Open();
                db.Transaction = db.Connection.BeginTransaction();

                try
                {
                    Scratchpad _cs = new Scratchpad
                    {
                        Id = data.Id,
                        CreatedOn = data.CreatedOn,
                        UserId = data.UserId,
                        CallingPageId = data.CallingPageId,
                        SubjectId = data.SubjectId,
                        Content = data.Content,
                    };
                }
            }
        }
    }
}

```



```

        db.Scratchpads.InsertOnSubmit(_cs);

        db.SubmitChanges();
        db.Transaction.Commit();

        result = true;
    }
    catch (Exception exc)
    {
        db.Transaction.Rollback();
        throw exc;
    }
}

return result;
}
public bool Update(EteScratchpad data)
{

    bool result = false;
    string mess;
    EteScratchpad odata = new EteScratchpad();

    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {

            var _user = db.Scratchpads.Where(e => e.Id == data.Id).SingleOrDefault();

            if (_user == null)
            {
                throw (new ApplicationException("Unable to find the User record"));
            }

            odata = new EteScratchpad
            {
                Id = _user.Id,
                CreatedOn = _user.CreatedOn,
                UserId = _user.UserId,
                CallingPageId = _user.CallingPageId,
                SubjectId = _user.SubjectId,
                Content = _user.Content,
            };

            if (odata.CreatedOn == null) { odata.CreatedOn = DateTime.Today; }
            _user.Id = data.Id;
            _user.CreatedOn = data.CreatedOn;
            _user.UserId = data.UserId;
            _user.CallingPageId = data.CallingPageId;
            _user.SubjectId = data.SubjectId;
            _user.Content = data.Content;

```

```

        db.SubmitChanges();
        db.Transaction.Commit();
        result = true;
    }
    catch (Exception exc)
    {
        db.Transaction.Rollback();

        throw exc;
    }
    mess = BalAuditTrail.GetAuditXml("Scratchpad", odata, data);
    EteAuditTrail at = new EteAuditTrail();
    at = new EteAuditTrail
    {
        CreatedOn = odata.CreatedOn,
        UserId = data.UId,
        ObjectId = data.Id,
        Category = "Edit",
        Message = mess,
        AuditDate = DateTime.Today,
        PageId = data.PageId,
    };
    bool res = new BalAuditTrail().Insert(at);
}

return result;
}
public bool DeleteRecord(int Id)
{
    bool result = false;
    string mess;
    EteScratchpad odata = new EteScratchpad();
    EteScratchpad data = new EteScratchpad();

    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {
            var _user = db.Scratchpads.Where(e => e.Id == Id).SingleOrDefault();

            if (_user == null)
            {
                throw (new ApplicationException("Unable to find the User record"));
            }
            odata = new EteScratchpad
            {

```

```

        Id = _user.Id,
        CreatedOn = _user.CreatedOn,
        UserId = _user.UserId,
        CallingPageId = _user.CallingPageId,
        SubjectId = _user.SubjectId,
        Content = _user.Content,
    };
    if (odata.CreatedOn == null) { odata.CreatedOn = DateTime.Today; }

    db.Scratchpads.DeleteOnSubmit(_user);
    db.SubmitChanges();
    db.Transaction.Commit();
    result = true;
}
catch (Exception exc)
{
    db.Transaction.Rollback();

    throw exc;
}
mess = BalAuditTrail.GetAuditXml("Scratchpad", odata, data);
EteAuditTrail at = new EteAuditTrail();
at = new EteAuditTrail
{
    CreatedOn = odata.CreatedOn,
    UserId = odata.UserId,
    ObjectId = odata.Id,
    Category = "Delete",
    Message = mess,
    AuditDate = DateTime.Today,
    PageId = 132,
};
bool res = new BalAuditTrail().Insert(at);
}

return result;
}
public bool DeleteAll(int UserId)
{
    bool result = false;
    string mess;
    EteScratchpad odata = new EteScratchpad();
    EteScratchpad data = new EteScratchpad();

    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {

```

```

var _users = db.Scratchpads.Where(e => e.UserId == UserId).ToList();

if (_users == null)
{
    throw (new ApplicationException("Unable to find the User record"));
}
foreach (var _user in _users)
{
    odata = new EteScratchpad
    {
        Id = _user.Id,
        CreatedOn = _user.CreatedOn,
        UserId = _user.UserId,
        CallingPageId = _user.CallingPageId,
        SubjectId = _user.SubjectId,
        Content = _user.Content,
    };
    mess = BalAuditTrail.GetAuditXml("Scratchpad", odata, data);
    EteAuditTrail at = new EteAuditTrail();
    at = new EteAuditTrail
    {
        CreatedOn = odata.CreatedOn,
        UserId = odata.UserId,
        Objectid = data.Id,
        Category = "Delete",
        Message = mess,
        AuditDate = DateTime.Today,
        PageId = 132,
    };
    bool res = new BalAuditTrail().Insert(at);
}
if (odata.CreatedOn == null) { odata.CreatedOn = DateTime.Today; }

db.Scratchpads.DeleteAllOnSubmit(_users);
db.SubmitChanges();
db.Transaction.Commit();
result = true;
}
catch (Exception exc)
{
    db.Transaction.Rollback();

    throw exc;
}

}

return result;
}
public EteScratchpad getScratchpad(int name)
{

```

```

EteScratchpad result = new EteScratchpad();
using (var db = new DowPortalDataContext(Configuration.ConnectionString))
{
    db.Connection.Open();
    var users = (from c in db.Scratchpads where c.Id == name select c).ToList();
    foreach (var data in users)
    {
        result = new EteScratchpad
        {
            Id = data.Id,
            CreatedOn = data.CreatedOn,
            UserId = data.UserId,
            CallingPagelId = data.CallingPagelId,
            SubjectId = data.SubjectId,
            Content = data.Content,

        };
        break;
    }
}

return result;
}
public List<EteScratchpad> GetScratchpads(string Curl)
{
    List<EteScratchpad> _user = new List<EteScratchpad>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        db.Connection.Open();
        int pgid = (new tblPagess()).GetPagelId(Curl);
        var users = (from c in db.Scratchpads select c).ToList();
        foreach (var data in users)
        {
            _user.Add(new EteScratchpad
            {
                Id = data.Id,
                CreatedOn = data.CreatedOn,
                UserId = data.UserId,
                CallingPagelId = data.CallingPagelId,
                SubjectId = data.SubjectId,
                Content = data.Content,

                PagelId = pgid,
            });
        }

        return _user;
    }
}
public List<EteScratchpad> GetScratchpadsByUsers(string Curl, int DealerId, int
GroupId)
{
    List<EteScratchpad> _user = new List<EteScratchpad>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))

```

```

{
    db.Connection.Open();
    int pgid = (new tblPagess()).GetPagelD(Curl);
    var users = (from c in db.Scratchpads select c).ToList();
    foreach (var data in users)
    {
        _user.Add(new EteScratchpad
        {
            Id = data.Id,
            CreatedOn = data.CreatedOn,
            UserId = data.UserId,
            CallingPagelD = data.CallingPagelD,
            SubjectId = data.SubjectId,
            Content = data.Content,

            PagelD = pgid,
        });
    }

    return _user;
}
}

public List<EteScratchpad> GetMyScratchpadsList(string Curl, int UserId, int GroupId)
{
    List<EteScratchpad> _user = new List<EteScratchpad>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        int pgid = (new tblPagess()).GetPagelD(Curl);
        if (UserId == 1)
        {
            var users = (from c in db.Scratchpads select c).ToList();
            foreach (var data in users)
            {
                _user.Add(new EteScratchpad
                {
                    Id = data.Id,
                    CreatedOn = data.CreatedOn,
                    UserId = data.UserId,
                    CallingPagelD = data.CallingPagelD,
                    SubjectId = data.SubjectId,
                    Content = data.Content,
                    PagelD = pgid,
                });
            }
        }
        else
        {
            var users = (from c in db.Scratchpads where c.UserId==UserId select
c).ToList();
            foreach (var data in users)
            {
                _user.Add(new EteScratchpad

```

```

        {
            Id = data.Id,
            CreatedOn = data.CreatedOn,
            UserId = data.UserId,
            CallingPageId = data.CallingPageId,
            SubjectId = data.SubjectId,
            Content = data.Content,
            PageId = pgid,
        });
    }
}
return _user;
}
}
public String GetScratchpadName(int Id)
{
    String name = "";
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();

        var _cust = (from c in db.Scratchpads where c.Id == Id select c).SingleOrDefault();

        if (_cust != null)
        {
            name = _cust.Content;
        }
        else
        {
            name = "";
        }
    }

    return name;
}
}
}

```

```

using System;
using System.IO;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using DowBEntity;
using DowDAL;
using System.Xml;
using System.Xml.Serialization;
using System.Xml.XmlConfiguration;
namespace DowBAL
{

```

```

public class StaffHolidays
{

    //public int? GetNextStaffHolidayId()
    //{

    //    int? result = 0;
    //    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    //    {
    //        GetNextStaffHolidayIdResult lst = db.GetNextStaffHolidayId().SingleOrDefault();
    //        result = lst.NextStaffHolidayId;

    //    }

    //    return result;
    //}
    public bool StaffHolidayNameExists(string name)
    {
        bool exists = false;

        using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        {

            db.Connection.Open();

            var _cust = (from c in db.StaffHolidays where c.Description == name select
c).SingleOrDefault();

            if (_cust != null)
            {
                exists = true;
            }

        }

        return exists;
    }
    public bool Insert(EteStaffHoliday data)
    {
        bool result = false;
        EteStaffHoliday _user = new EteStaffHoliday();
        using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        {
            db.Connection.Open();
            db.Transaction = db.Connection.BeginTransaction();

            try
            {
                StaffHoliday _cs = new StaffHoliday
                {
                    //Id = data.Id,

```



```

        CreatedOn = data.CreatedOn,
        BMSUserId = data.BMSUserId,
        StartDate = data.StartDate,
        EndDate = data.EndDate,
        ActualDays = data.ActualDays,
        RequestedDate = data.RequestedDate,
        Status = data.Status,
        DecisionDate = data.DecisionDate,
        DecisionById = data.DecisionById,
        Description = data.Description,
        RejectionReason = data.RejectionReason,
        BalanceBefore = data.BalanceBefore,
        BalanceAfter = data.BalanceAfter,
    };
    db.StaffHolidays.InsertOnSubmit(_cs);

    db.SubmitChanges();
    db.Transaction.Commit();

    result = true;
}
catch (Exception exc)
{
    db.Transaction.Rollback();
    throw exc;
}
}

return result;
}
public bool Update(EteStaffHoliday data)
{
    bool result = false;
    string mess;
    EteStaffHoliday odata = new EteStaffHoliday();

    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {
            var _user = db.StaffHolidays.Where(e => e.Id == data.Id).SingleOrDefault();

            if (_user == null)
            {
                throw (new ApplicationException("Unable to find the User record"));
            }

            odata = new EteStaffHoliday
            {
                Id = data.Id,
            }
        }
    }
}

```

```

        CreatedOn = data.CreatedOn,
        BMSUserId = data.BMSUserId,
        StartDate = data.StartDate,
        EndDate = data.EndDate,
        ActualDays = data.ActualDays,
        RequestedDate = data.RequestedDate,
        Status = data.Status,
        DecisionDate = data.DecisionDate,
        DecisionByld = data.DecisionByld,
        Description = data.Description,
        RejectionReason = data.RejectionReason,
        BalanceBefore = data.BalanceBefore,
        BalanceAfter = data.BalanceAfter,
    };
    if (odata.CreatedOn == null) { odata.CreatedOn = DateTime.Today; }
    _user.Id = data.Id;
    _user.CreatedOn = data.CreatedOn;
    _user.BMSUserId = data.BMSUserId;
    _user.StartDate = data.StartDate;
    _user.EndDate = data.EndDate;
    _user.ActualDays = data.ActualDays;
    _user.RequestedDate = data.RequestedDate;
    _user.Status = data.Status;
    _user.DecisionDate = data.DecisionDate;
    _user.DecisionByld = data.DecisionByld;
    _user.Description = data.Description;
    _user.RejectionReason = data.RejectionReason;
    _user.BalanceBefore = data.BalanceBefore;
    _user.BalanceAfter = data.BalanceAfter;

    db.SubmitChanges();
    db.Transaction.Commit();
    result = true;
}
catch (Exception exc)
{
    db.Transaction.Rollback();

    throw exc;
}
mess = BalAuditTrail.GetAuditXml("StaffHoliday", odata, data);
EteAuditTrail at = new EteAuditTrail();
at = new EteAuditTrail
{
    CreatedOn = odata.CreatedOn,
    UserId = data.UId,
    Objectld = data.Id,
    Category = "Edit",
    Message = mess,
    AuditDate = DateTime.Today,
    Pageld = data.Pageld,
};
bool res = new BalAuditTrail().Insert(at);
}

```

```

        return result;
    }
    public EteStaffHoliday getStaffHoliday(int name)
    {
        EteStaffHoliday result = new EteStaffHoliday();
        using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        {
            db.Connection.Open();
            var users = (from c in db.StaffHolidays where c.Id == name select c).ToList();
            foreach (var data in users)
            {
                result = new EteStaffHoliday
                {
                    Id = data.Id,
                    CreatedOn = data.CreatedOn,
                    BMSUserId = data.BMSUserId,
                    StartDate = data.StartDate,
                    EndDate = data.EndDate,
                    ActualDays = data.ActualDays,
                    RequestedDate = data.RequestedDate,
                    Status = data.Status,
                    DecisionDate = data.DecisionDate,
                    DecisionByld = data.DecisionByld,
                    Description = data.Description,
                    RejectionReason = data.RejectionReason,
                    BalanceBefore = data.BalanceBefore,
                    BalanceAfter = data.BalanceAfter,
                };
                break;
            }
        }

        return result;
    }
    public List<EteStaffHoliday> GetStaffHolidays(string Curl)
    {
        //status 1.Pending,2.Approved,3.Rejected
        string statusname = "";
        List<EteStaffHoliday> _user = new List<EteStaffHoliday>();
        using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        {
            string UserName="", DecisionUserName="";

            db.Connection.Open();
            int pgid = (new tblPagess()).GetPageld(Curl);
            var users = (from c in db.StaffHolidays select c).ToList();
            foreach (var data in users)
            {
                UserName = (from u in db.tblUsers where u.UserId == data.BMSUserId select
                u.UserName).SingleOrDefault();
                DecisionUserName = (from u in db.tblUsers where u.UserId ==
                data.DecisionByld select u.UserName).SingleOrDefault();
                if (data.Status == 1) { statusname = "Pending"; } else if (data.Status == 2) {
                statusname = "Approved"; } else if (data.Status == 3) { statusname = "Rejected"; }
            }
        }
    }

```

```

        _user.Add(new EteStaffHoliday
        {
            Id = data.Id,
            CreatedOn = data.CreatedOn,
            BMSUserId = data.BMSUserId,
            StartDate = data.StartDate,
            EndDate = data.EndDate,
            ActualDays = data.ActualDays,
            RequestedDate = data.RequestedDate,
            Status = data.Status,
            DecisionDate = data.DecisionDate,
            DecisionById = data.DecisionById,
            Description = data.Description,
            RejectionReason = data.RejectionReason,
            BalanceBefore = data.BalanceBefore,
            BalanceAfter = data.BalanceAfter,
            PageId = pgid,
            UserName = UserName,
            DecisionUserName = DecisionUserName,
            StatusName=statusname,

        });
    }

    return _user;
}

public String GetStaffHolidayName(int Id)
{
    String name = "";
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        db.Connection.Open();

        var _cust = (from c in db.StaffHolidays where c.Id == Id select
c).SingleOrDefault();

        if (_cust != null)
        {
            name = _cust.Description;
        }
        else
        {
            name = "";
        }
    }

    return name;
}
}

```

```

}
                                     ****

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using DowBEntity;
using DowDAL;
namespace DowBAL
{

    public class tblGrpMappings
    {
        public static bool tblGrpMappingNameExists(int GroupId, int PageId)
        {
            bool exists = false;

            using (var db = new DowPortalDataContext(Configuration.ConnectionString))
            {

                db.Connection.Open();

                var _cust = (from c in db.tblGrpMappings where c.GroupId == GroupId & c.PageId
                == PageId select c).SingleOrDefault();

                if (_cust != null)
                {
                    exists = true;
                }

            }

            return exists;
        }
        //public int? GetNexttblGrpMappingId()
        //{
        //    int? result = 0;
        //    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        //    {
        //        GetNexttblGrpMappingIdResult lst =
        //        db.GetNexttblGrpMappingId().SingleOrDefault();
        //        result = lst.NexttblGrpMappingId;
        //    }

        //    return result;
        //}

        public bool Insert(EtettblGrpMapping data)
        {
            bool result = false;
            EtettblGrpMapping _user = new EtettblGrpMapping();

```

```

using (var db = new DowPortalDataContext(Configuration.ConnectionString))
{
    db.Connection.Open();
    db.Transaction = db.Connection.BeginTransaction();

    try
    {
        var _Grp = (from c in db.tblGrpMappings where c.GroupId == data.GroupId &
c.PageId == data.PageId select c).SingleOrDefault();

        if (_Grp == null)
        {
            tblGrpMapping _cs = new tblGrpMapping
            {
                MapId = data.MapId,
                GroupId = data.GroupId,
                PageId = data.PageId,
                view = data.view,
                add = data.add,
                Edit = data.Edit,
                Delete = data.Delete,
                CreatedOn = data.CreatedOn,

            };
            db.tblGrpMappings.InsertOnSubmit(_cs);

            db.SubmitChanges();
            db.Transaction.Commit();

            result = true;
        }
        else
        {
            throw (new ApplicationException("Data already exists"));
        }
    }
    catch (Exception exc)
    {
        db.Transaction.Rollback();
        throw exc;
    }
}

return result;
}

public bool UpdatePageList(List<EtetblGrpMapping> data)
{
    bool result = false;
    EtetblGrpMapping odata=new EtetblGrpMapping();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))

```

```

    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {
            foreach (var row in data)
            {
                var _user = db.tblGrpMappings.Where(e => e.MapId ==
row.MapId).SingleOrDefault();
                if (_user == null)
                {
                    throw (new ApplicationException("Unable to find the User record"));
                }
                odata = new EtetblGrpMapping
{MapId=_user.MapId,view=_user.view,PageId=_user.PageId,CreatedOn=_user.CreatedOn
};
                _user.view = row.view;
                db.SubmitChanges();

            }
            db.Transaction.Commit();
            result = true;
        }
        catch (Exception exc)
        {
            db.Transaction.Rollback();

            throw exc;
        }
    }

    return result;
}
public bool Update(EtetblGrpMapping data)
{
    bool result = false;

    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {

            var _user = db.tblGrpMappings.Where(e => e.MapId ==
data.MapId).SingleOrDefault();

            if (_user == null)
            {
                throw (new ApplicationException("Unable to find the User record"));
            }

```

```

        _user.MapId = data.MapId;
        _user.GroupId = data.GroupId;
        _user.PagId = data.PagId;
        _user.view = data.view;
        _user.add = data.add;
        _user.Edit = data.Edit;
        _user.Delete = data.Delete;
        _user.CreatedOn = data.CreatedOn;

        db.SubmitChanges();
        db.Transaction.Commit();

        result = true;
    }
    catch (Exception exc)
    {
        db.Transaction.Rollback();

        throw exc;
    }
}

return result;
}
public EtetblGrpMapping gettblGrpMapping(int GroupId)
{
    EtetblGrpMapping result = new EtetblGrpMapping();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var users = (from c in db.tblGrpMappings where c.GroupId == GroupId select
c).ToList();
        foreach (var data in users)
        {
            result = new EtetblGrpMapping
            {
                MapId = data.MapId,
                GroupId = data.GroupId,
                PagId = data.PagId,
                view = data.view,
                add = data.add,
                Edit = data.Edit,
                Delete = data.Delete,
                CreatedOn = data.CreatedOn,

            };
            break;
        }
    }

    return result;
}
public List<Eteev_tblGrpMapping> GettblGrpMappingDesc()

```



```

{
    List<Etev_tblGrpMapping> _user = new List<Etev_tblGrpMapping>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();

        var users = (from c in db.v_tblGrpMappings select c).ToList();
        foreach (var data in users)
        {
            _user.Add(new Etev_tblGrpMapping
            {
                MapId = Convert.ToInt32(data.MapId),
                GroupId = data.GroupId,
                GroupName = data.GroupName,
                PageId = data.PageId,
                PageName = data.PageName,
                view = data.view,
                add = data.add,
                Edit = data.Edit,
                Delete = data.Delete,
                CreatedOn = data.CreatedOn,

            });
        }

        return _user;
    }
}

```

```

public List<Etev_tblGrpMapping> GettblGrpMappingDescByGroupId(int GroupId)
{
    List<Etev_tblGrpMapping> _user = new List<Etev_tblGrpMapping>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();

        var users = (from c in db.v_tblGrpMappings where c.GroupId == GroupId select
c).ToList();
        foreach (var data in users)
        {
            _user.Add(new Etev_tblGrpMapping
            {
                MapId = Convert.ToInt32(data.MapId),
                GroupId = data.GroupId,
                GroupName = data.GroupName,
                PageId = data.PageId,
                PageName = data.PageName,
                view = data.view,
                add = data.add,
                Edit = data.Edit,
                Delete = data.Delete,
            });
        }
    }
}

```

```

        CreatedOn = data.CreatedOn,
        ParentId = data.ParentId,
        level = data.level,
    });
}

return _user;
}
}

public bool InsertBulk(List<EtetblGrpMapping> datas)
{
    bool result = false;
    int res;
    EtetblGrpMapping _user = new EtetblGrpMapping();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {
            foreach (var data in datas)
            {
                //res = db.AddNewPageMapping(data.MapId, data.GroupId, data.view);
                //tblGrpMapping _cs = new tblGrpMapping
                //{
                //    MapId = data.MapId,
                //    GroupId = data.GroupId,
                //    PageId = data.PageId,
                //    view = data.view,
                //    add = data.add,
                //    Edit = data.Edit,
                //    Delete = data.Delete,
                //    CreatedOn = data.CreatedOn,
                //};
                //db.tblGrpMappings.InsertOnSubmit(_cs);
                //db.SubmitChanges();
            }
            db.Transaction.Commit();

            result = true;
        }
        catch (Exception exc)
        {
            db.Transaction.Rollback();
            throw exc;
        }
    }
}

```

```

    return result;
}

public List<mapids> GetPerSetsForGroup(int GroupId,int type)
{
    List<mapids> lst = new List<mapids>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        //int istatus = 1;
        db.Connection.Open();
        List<ManagePermissionsResult> List = db.ManagePermissions(GroupId,
type).ToList();
        foreach (var row in List)
        {
            if (Convert.ToBoolean(row.view) == true)
            {
                //if (row.view == "true")
                //{
                //    istatus = 1;
                //}
                lst.Add(new mapids
                {
                    mpageid = row.MapId.ToString(),
                    status = "1",
                    buttonid = "",
                    pageid = Convert.ToInt32(row.PagId),
                });
            }
        }
    }

    return lst;
}

public List<mapids> GetLeftButtonsPermission(int pageid, int groupid)
{
    List<mapids> lst = new List<mapids>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        List<GetLeftButtonsPermissionResult> List = db.GetLeftButtonsPermission(pageid,
groupid).ToList();
        foreach (var row in List)
        {
            if (row.show == 1)
            {
                lst.Add(new mapids
                {
                    mpageid = "",
                    status = "1",
                    buttonid = row.pagebtnid.ToString(),
                });
            }
        }
    }
}

```

```

        return lst;
    }
}

}

****

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using DowBEntity;
using DowDAL;
namespace DowBAL
{
    public class tblLeftButtons
    {
        public static bool tblLeftButtonsNameExists(int Id)
        {
            bool exists = false;

            using (var db = new DowPortalDataContext(Configuration.ConnectionString))
            {
                db.Connection.Open();

                var _cust = (from c in db.tblLeftButtons where c.lbtnid == Id select
c).SingleOrDefault();

                if (_cust != null)
                {
                    exists = true;
                }
            }

            return exists;
        }
        //public int? GetNexttblLeftButtonsId()
        //{
            // int? result = 0;
            // using (var db = new DowPortalDataContext(Configuration.ConnectionString))
            // {
                // GetNexttblLeftButtonsIdResult lst =
db.GetNexttblLeftButtonsId().SingleOrDefault();
                // result = lst.NexttblLeftButtonsId;

            // }
        }
    }
}

```

```

// return result;
//}

public bool Insert(EtetblLeftButtons data)
{
    bool result = false;
    EtetblLeftButtons _user = new EtetblLeftButtons();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {
            tblLeftButton _cs = new tblLeftButton
            {
                lbtnid = data.lbtnid,
                lbtnName = data.lbtnName,
                lbtnImage = data.lbtnImage,
                lbtnUrl = data.lbtnUrl,
                lbtnCode = data.lbtnCode,
                lbtnPriority = data.lbtnPriority,
                lbtnLevel = data.lbtnLevel,
                isactive = data.isactive,
                lbtnType = data.lbtnType,
            };
            db.tblLeftButtons.InsertOnSubmit(_cs);

            db.SubmitChanges();
            db.Transaction.Commit();

            result = true;
        }
        catch (Exception exc)
        {
            db.Transaction.Rollback();
            throw exc;
        }
    }

    return result;
}

public bool Update(EtetblLeftButtons data)
{
    bool result = false;

    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try

```

```

    {
        var _user = db.tblLeftButtons.Where(e => e.lbtnid ==
data.lbtnid).SingleOrDefault();

        if (_user == null)
        {
            throw (new ApplicationException("Unable to find the User record"));
        }

        _user.lbtnid = data.lbtnid;
        _user.lbtnName = data.lbtnName;
        _user.lbtnImage = data.lbtnImage;
        _user.lbtnUrl = data.lbtnUrl;
        _user.lbtnCode = data.lbtnCode;
        _user.lbtnPriority = data.lbtnPriority;
        _user.lbtnLevel = data.lbtnLevel;
        _user.isactive = data.isactive;
        _user.lbtnType = data.lbtnType;

        db.SubmitChanges();
        db.Transaction.Commit();

        result = true;
    }
    catch (Exception exc)
    {
        db.Transaction.Rollback();

        throw exc;
    }
}

return result;
}
public EtetblLeftButtons gettblLeftButtons(string name)
{
    EtetblLeftButtons result = new EtetblLeftButtons();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var users = (from c in db.tblLeftButtons where c.lbtnName == name select
c).ToList();
        foreach (var data in users)
        {
            result = new EtetblLeftButtons
            {
                lbtnid = data.lbtnid,
                lbtnName = data.lbtnName,
                lbtnImage = data.lbtnImage,
                lbtnUrl = data.lbtnUrl,
                lbtnCode = data.lbtnCode,
                lbtnPriority = data.lbtnPriority,
            }
        }
    }
}

```

```

        lbtnLevel = data.lbtnLevel,
        isactive = data.isactive,
        lbtnType = data.lbtnType,
    };
    break;
}
}

return result;
}
public List<EtetblLeftButtons> GettblLeftButtons()
{
    List<EtetblLeftButtons> _user = new List<EtetblLeftButtons>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();

        var users = (from c in db.tblLeftButtons select c).ToList();
        foreach (var data in users)
        {
            _user.Add(new EtetblLeftButtons
            {
                lbtnid = data.lbtnid,
                lbtnName = data.lbtnName,
                lbtnImage = data.lbtnImage,
                lbtnUrl = data.lbtnUrl,
                lbtnCode = data.lbtnCode,
                lbtnPriority = data.lbtnPriority,
                lbtnLevel = data.lbtnLevel,
                isactive = data.isactive,
                lbtnType = data.lbtnType,
            });
        }

        return _user;
    }
}

}
}

****

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using DowBEntity;
using DowDAL;
namespace DowBAL

```

```

{
public class tblPageButtonMappings
{
    public static bool tblPageButtonMappingNameExists(int Id)
    {
        bool exists = false;

        using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        {

            db.Connection.Open();

            var _cust = (from c in db.tblPageButtonMappings where c.pagebtnid == Id select
c).SingleOrDefault();

            if (_cust != null)
            {
                exists = true;
            }

        }

        return exists;
    }
    //public int? GetNexttblPageButtonMappingId()
    //{

    //    int? result = 0;
    //    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    //    {
    //        GetNexttblPageButtonMappingIdResult lst =
db.GetNexttblPageButtonMappingId().SingleOrDefault();
    //        result = lst.NexttblPageButtonMappingId;

    //    }

    //    return result;
    //}

    public bool Insert(EtetblPageButtonMapping data)
    {
        bool result = false;
        EtetblPageButtonMapping _user = new EtetblPageButtonMapping();
        using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        {
            db.Connection.Open();
            db.Transaction = db.Connection.BeginTransaction();

            try
            {
                tblPageButtonMapping _cs = new tblPageButtonMapping
                {

```



```

        pagebtnid = data.pagebtnid,
        pageid = data.pageid,
        lbtnid = data.lbtnid,
        show = data.show,
        priority = data.priority,
        mapid =(int) data.mapid,
        urlpageid = data.urlpageid,
        groupid = data.groupid,
        matype = data.matype,
        mapdescription = data.mapdescription,
        isactive =1

};
db.tblPageButtonMapings.InsertOnSubmit(_cs);

db.SubmitChanges();
db.Transaction.Commit();

    result = true;
}
catch (Exception exc)
{
    db.Transaction.Rollback();
    throw exc;
}
}

return result;
}
public bool UpdateButtonList(List<EtetblPageButtonMapping> data)
{
    bool result = false;

    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {
            foreach (var row in data)
            {
                var _user = db.tblPageButtonMapings.Where(e => e.pagebtnid ==
row.pagebtnid).SingleOrDefault();

                if (_user == null)
                {
                    throw (new ApplicationException("Unable to find the User record"));
                }

                _user.show = row.show;

                db.SubmitChanges();
            }
        }
    }
}

```

```

        db.Transaction.Commit();
        result = true;
    }
    catch (Exception exc)
    {
        db.Transaction.Rollback();

        throw exc;
    }
}

return result;
}
public bool Update(EtctlPageButtonMapping data)
{
    bool result = false;

    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {
            var _user = db.tblPageButtonMappings.Where(e => e.pagebtnid ==
data.pagebtnid).SingleOrDefault();

            if (_user == null)
            {
                throw (new ApplicationException("Unable to find the User record"));
            }

            _user.pagebtnid = data.pagebtnid;
            _user.pageid = data.pageid;
            _user.lbtnid = data.lbtnid;
            _user.show = data.show;
            _user.priority = data.priority;
            db.SubmitChanges();
            db.Transaction.Commit();

            result = true;
        }
        catch (Exception exc)
        {
            db.Transaction.Rollback();

            throw exc;
        }
    }

    return result;
}
}

```

```

public bool GetDataExists(int btnid,int pageid,int groupid)
{
    bool exists = true;
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var users = (from c in db.tblPageButtonMapings where c.groupid == groupid &&
c.lbtnid==btnid && c.pageid==pageid select c).ToList();
        if(users.Count==0){
            exists = false;
        }
    }
    return exists;
}

public EtetblPageButtonMapping gettblPageButtonMapping(int pagebtnid)
{
    EtetblPageButtonMapping result = new EtetblPageButtonMapping();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var users = (from c in db.tblPageButtonMapings where c.pagebtnid == pagebtnid
select c).ToList();
        foreach (var data in users)
        {
            result = new EtetblPageButtonMapping
            {
                pagebtnid = data.pagebtnid,
                pageid = data.pageid,
                lbtnid = data.lbtnid,
                show = data.show,
                priority = data.priority,
            };
            break;
        }
    }

    return result;
}

public List<EtetblPageButtonMapping> GettblPageButtonMapping()
{
    List<EtetblPageButtonMapping> _user = new List<EtetblPageButtonMapping>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        db.Connection.Open();

        var users = (from c in db.tblPageButtonMapings select c).ToList();
        foreach (var data in users)
        {
            _user.Add(new EtetblPageButtonMapping
            {
                pagebtnid = data.pagebtnid,
                pageid = data.pageid,
                lbtnid = data.lbtnid,
            }
        }
    }
}

```

```

        show = data.show,
        priority = data.priority,
    });
}

return _user;
}
}
public List<EtetblPageButtonMapping> GettblPageButtonMappingGroupld(int Groupid)
{
    List<EtetblPageButtonMapping> _user = new List<EtetblPageButtonMapping>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        db.Connection.Open();

        var users = (from c in db.tblPageButtonMappings where c.groupid == Groupid
select c).ToList();
        foreach (var data in users)
        {
            _user.Add(new EtetblPageButtonMapping
            {
                pagebtnid = data.pagebtnid,
                mapid = data.mapid,
                pageid = data.pageid,
                lbtnid = data.lbtnid,
                show = data.show,
                priority = data.priority,
                urlpageid =(int) data.urlpageid,

                groupid = data.groupid,
                maptype = (int)data.maptype,
                mapdescription = data.mapdescription,
                isactive =(int) data.isactive,
            });
        }

        return _user;
    }
}
public List<EtetblPageButtonMapping> GettblPageButtonMappingbyPageId(int pageid)
{
    List<EtetblPageButtonMapping> _user = new List<EtetblPageButtonMapping>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        db.Connection.Open();

        var users = (from c in db.v_tblPageButtonMappings where c.pageid==pageid select
c).ToList();

```

```

foreach (var data in users)
{
    _user.Add(new EtetblPageButtonMapping
    {
        pagebtnid = data.pagebtnid,
        pageid = data.pageid,
        lbtnid = data.lbtnid,
        show = data.show,
        priority = data.priority,
        lbtnName = data.lbtnName,
    });
}

return _user;
}
}

public bool InsertBulk(List<EtetblPageButtonMapping> datas)
{
    bool result = false;
    int res;
    EtetblPageButtonMapping _user = new EtetblPageButtonMapping();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {
            foreach (var data in datas)
            {
                res = db.AddButtonMapping(data.pagebtnid, data.show, data.groupid);
                //tblPageButtonMapping _cs = new tblPageButtonMapping
                //{
                //    pagebtnid = data.pagebtnid,
                //    pageid = data.pageid,
                //    lbtnid = data.lbtnid,
                //    show = data.show,
                //    priority = data.priority,
                //};
                //db.tblPageButtonMappings.InsertOnSubmit(_cs);

                //db.SubmitChanges();
            }
            db.Transaction.Commit();

            result = true;
        }
        catch (Exception exc)
        {
            db.Transaction.Rollback();
            throw exc;
        }
    }
}

```

```

        return result;
    }
}

}

****

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using DowBEntity;
using DowDAL;
namespace DowBAL
{
    public class tblPagess
    {
        public static bool tblPagesNameExists(int Id)
        {
            bool exists = false;

            using (var db = new DowPortalDataContext(Configuration.ConnectionString))
            {
                db.Connection.Open();

                var _cust = (from c in db.tblPages where c.PagelId == Id select
c).SingleOrDefault();

                if (_cust != null)
                {
                    exists = true;
                }
            }

            return exists;
        }

        //public int? GetNexttblPagesId()
        //{
        //    int? result = 0;
        //    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        //    {
        //        GetNexttblPagesIdResult lst = db.GetNexttblPagesId().SingleOrDefault();
        //        result = lst.NexttblPagesId;
        //    }

        //    return result;
        //}
    }
}

```

```

public int GetPageld(string Purl)
{
    int rtnval = 0;
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        //var _cust = (from c in db.tblPages where c.PageUrl == Purl select c);
        var _cust = (from c in db.tblPages.Where(u => u.PageUrl.Contains(Purl)) select
c).ToList();

        foreach (var data in _cust)
        {
            rtnval = data.Pageld;
            return rtnval;
        }

    }

    return rtnval;
}

public bool Insert(EtetblPages data)
{
    bool result = false;
    EtetblPages _user = new EtetblPages();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {
            tblPage _cs = new tblPage
            {
                Pageld = data.Pageld,
                PageName = data.PageName,
                PageUrl = data.PageUrl,
                PageCode = data.PageCode,
                ParentId = data.ParentId,
                Priority = data.Priority,
                level = data.level,
                isMenuItem = data.isMenuItem,
            };
            db.tblPages.InsertOnSubmit(_cs);

            db.SubmitChanges();
            db.Transaction.Commit();

            result = true;
        }
    }
}

```

```

        catch (Exception exc)
        {
            db.Transaction.Rollback();
            throw exc;
        }
    }

    return result;
}
public bool Update(EtetblPages data)
{
    bool result = false;

    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {
            var _user = db.tblPages.Where(e => e.Pageld ==
data.Pageld).SingleOrDefault();

            if (_user == null)
            {
                throw (new ApplicationException("Unable to find the User record"));
            }

            _user.Pageld = data.Pageld;
            _user.PageName = data.PageName;
            _user.PageUrl = data.PageUrl;
            _user.PageCode = data.PageCode;
            _user.ParentId = data.ParentId;
            _user.Priority = data.Priority;
            _user.level = data.level;
            _user.isMenuItem = data.isMenuItem;

            db.SubmitChanges();
            db.Transaction.Commit();

            result = true;
        }
        catch (Exception exc)
        {
            db.Transaction.Rollback();

            throw exc;
        }
    }

    return result;
}

```



```

public string GettblPageNamebyId(int id)
{
    EtetblPages result = new EtetblPages();
    string strpagename = "";
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var users = (from c in db.tblPages where c.Pageld == id select
c).SingleOrDefault();
        //foreach (var data in users)
        //{
        if (users != null)
        {
            strpagename = users.PageName;
        }
        //return strpagename;
        //break;
        //}
    }

    return strpagename;
}
public EtetblPages GettblPagebyId(int id)
{
    EtetblPages result = new EtetblPages();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var users = (from c in db.tblPages where c.Pageld == id select c).ToList();
        foreach (var data in users)
        {
            result = new EtetblPages
            {
                Pageld = data.Pageld,
                PageName = data.PageName,
                PageUrl = data.PageUrl,
                PageCode = data.PageCode,
                ParentId = data.ParentId,
                Priority = data.Priority,
                level = data.level,
                isMenuItem = data.isMenuItem,

            };
            break;
        }
    }

    return result;
}
public EtetblPages GettblPage(string name)
{
    EtetblPages result = new EtetblPages();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();

```

```

var users = (from c in db.tblPages where c.PageName == name select c).ToList();
foreach (var data in users)
{
    result = new EtetblPages
    {
        PageId = data.PageId,
        PageName = data.PageName,
        PageUrl = data.PageUrl,
        PageCode = data.PageCode,
        ParentId = data.ParentId,
        Priority = data.Priority,
        level = data.level,
        isMenuItem = data.isMenuItem,

    };
    break;
}
}

return result;
}
public List<EtetblPages> GettblPages()
{
    List<EtetblPages> _user = new List<EtetblPages>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        db.Connection.Open();

        var users = (from c in db.tblPages select c).ToList();
        foreach (var data in users)
        {
            _user.Add(new EtetblPages
            {
                PageId = data.PageId,
                PageName = data.PageName,
                PageUrl = data.PageUrl,
                PageCode = data.PageCode,
                ParentId = data.ParentId,
                Priority = data.Priority,
                level = data.level,
                isMenuItem = data.isMenuItem,

            });
        }

        return _user;
    }
}
public List<EteMenuList> GetMenuListbystype(int stype)
{
    List<EteMenuList> _user = new List<EteMenuList>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

```

```

        db.Connection.Open();

        var users = (from c in db.menulists where c.stype==stype && c.active==1 select
c).ToList();
        foreach (var data in users)
        {
            _user.Add(new EteMenuList
            {
                pageid = data.pageid,
                pagename = data.pagename,
                pageurl = data.pageurl,
                parentid = data.parentid,
                priority = data.priority,
                level = data.level,
                ismenuitem = data.ismenuitem,
                active = data.active,
                imagepath = data.imagepath,
                Stype = data.stype,
                Mlevel = data.mlevel,
                BtnImg=data.btnimg,
            });
        }

        return _user;
    }
}

```

```

}
}

```

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using DowBEntity;
using DowDAL;
namespace DowBAL
{
    public class tblPermissionGroups
    {
        public static bool tblPermissionGroupNameExists(int GroupId)
        {
            bool exists = false;

            using (var db = new DowPortalDataContext(Configuration.ConnectionString))
            {

                db.Connection.Open();

```

```

        var _cust = (from c in db.tblPermissionGroups where c.GroupId == GroupId select
c).SingleOrDefault();

        if (_cust != null)
        {
            exists = true;
        }

    }

    return exists;
}
public bool tblPermissionGroupNameExistsbyName(string GroupName)
{
    bool exists = false;

    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        db.Connection.Open();

        var _cust = (from c in db.tblPermissionGroups where c.GroupName ==
GroupName select c).ToList();

        if (_cust.Count > 0)
        {
            exists = true;
        }

    }

    return exists;
}
//public int? GetNexttblPermissionGroupId()
//{

//    int? result = 0;
//    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
//    {
//        GetNexttblPermissionGroupIdResult lst =
db.GetNexttblPermissionGroupId().SingleOrDefault();
//        result = lst.NexttblPermissionGroupId;

//    }

//    return result;
//}

public int Insert(EtblPermissionGroup data)
{
    int result = 0;
    EtblPermissionGroup _user = new EtblPermissionGroup();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))

```

```

{
    db.Connection.Open();
    db.Transaction = db.Connection.BeginTransaction();

    try
    {
        tblPermissionGroup _cs = new tblPermissionGroup
        {
            GroupId = data.GroupId,
            GroupName = data.GroupName,
            Status = data.Status,
            UserId = data.UserId,
            RoleId = 3,
        };
        db.tblPermissionGroups.InsertOnSubmit(_cs);

        db.SubmitChanges();
        db.Transaction.Commit();
        var _cust = (from c in db.tblPermissionGroups where c.GroupName ==
data.GroupName select c).SingleOrDefault();

        result = _cust.GroupId;
    }
    catch (Exception exc)
    {
        db.Transaction.Rollback();
        throw exc;
    }
}

return result;
}
public bool Update(EtblPermissionGroup data)
{
    bool result = false;

    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {
            var _user = db.tblPermissionGroups.Where(e => e.GroupId ==
data.GroupId).SingleOrDefault();

            if (_user == null)
            {
                throw (new ApplicationException("Unable to find the User record"));
            }

            _user.GroupId = data.GroupId;
            _user.GroupName = data.GroupName;

```

```

        _user.Status = data.Status;
        //_user.UserId = data.UserId;
        db.SubmitChanges();
        db.Transaction.Commit();

        result = true;
    }
    catch (Exception exc)
    {
        db.Transaction.Rollback();

        throw exc;
    }
}

return result;
}
public EtetblPermissionGroup GettblPermissionGroup(int name)
{
    EtetblPermissionGroup result = new EtetblPermissionGroup();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var users = (from c in db.tblPermissionGroups where c.GroupId == name select
c).ToList();
        foreach (var data in users)
        {
            result = new EtetblPermissionGroup
            {
                GroupId = data.GroupId,
                GroupName = data.GroupName,
                Status = data.Status,
                UserId=data.UserId,
            };
            break;
        }
    }

    return result;
}
public List<EtetblPermissionGroup> GettblPermissionGroups()
{
    List<EtetblPermissionGroup> _user = new List<EtetblPermissionGroup>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        db.Connection.Open();

        var users = (from c in db.tblPermissionGroups select c).ToList();
        foreach (var data in users)
        {
            _user.Add(new EtetblPermissionGroup
            {
                GroupId = data.GroupId,

```

```
        GroupName = data.GroupName,
        Status = data.Status,
        UserId=data.UserId,
    });
}

return _user;
}
}

}
```

```
using System;
using System.Collections.Generic;
using System.Threading.Tasks;
using System.Diagnostics;
using System.Data;
using System.Xml.Linq;
using Microsoft.VisualBasic;
using System.Collections;
using System.Linq;
using DowBEntity;
using DowDAL;
using System.Data.Linq;
using System.Data.SqlClient;
using System.Text;
using System.Net;
using System.Drawing;
using System.IO;
```

```
namespace DowBAL
{
    public class tblUsers
    {

        public bool UserNameExists(string name)
        {
            bool exists = false;

            using (var db = new
DowPortalDataContext(Configuration.ConnectionString))
            {

                db.Connection.Open();

                var _cust = (from c in db.tblUsers where c.UserName
== name select c).SingleOrDefault();

                if (_cust != null)
```

```

        {
            exists = true;
        }
    }

    return exists;
}
public string getusername(int UserId)
{
    string UserName="";
    using (var db = new
DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var users = (from c in db.tblUsers where c.UserId ==
UserId select c.UserName).SingleOrDefault();
        if (users!=null) { UserName = users; }
    }

    return UserName;
}
public string getuserpage(string username)
{
    string Userpage = "";
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        GetUserPageResult lst= db.GetUserPage(username).SingleOrDefault();
        Userpage = lst.GroupPage;
    }

    return Userpage;
}

public int GetDealerIDByUserid(int UserId)
{
    int DealerID = 0;
    using (var db = new
DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var users = (from c in db.tblUsers where c.UserId ==
UserId select c.DealerId).SingleOrDefault();
        if (users != 0) { DealerID = users; }
    }

    return DealerID;
}

public int? GetNextUserId()

```



```

    {
        int? result = 0;
        using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        {
            GetNextUserIdResult lst = db.GetNextUserId().SingleOrDefault();
            result = lst.NextUserId;
        }
        return result;
    }
}

public static EtetblUser CheckLogin(string username, string
password)
{
    EtetblUser result = null;

    string pword = password;
    result = (new EtetblUser { UserName = "" });

    try
    {
        using (var db = new
DowPortalDataContext(Configuration.ConnectionString))
        {
            result = (from x in db.tblUsers where
x.UserName == username && x.Password == password select new EtetblUser {
UserName = x.UserName }).SingleOrDefault();
        }
    }
    catch (Exception exc)
    {
        throw exc;
    }
    return result;
}

public bool Insert(EtetblUser data)
{
    bool result = false;
    EtetblUser _user = new EtetblUser();
    using (var db = new
DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();
        int userid=(int)GetNextUserId();

        try
        {
            tblUser _cs = new tblUser
            { UserId=userid,
            UserName = data.UserName,
            Password = data.Password,

```

```

        FirstName = data.FirstName,
        LastName = data.LastName,
        EmailID = data.EmailID,
        ContactNum = data.ContactNum,
        CreatedOn = data.CreatedOn,
        Status = data.Status,
        RoleID = data.RoleID,
        GroupId = data.GroupId,
        IsRemoved = data.IsRemoved,
        Department = data.Department,
        Region = data.Region,
        UpdatedOn = data.UpdatedOn,
        DealerId=data.DealerId,
    };
    db.tblUsers.InsertOnSubmit(_cs);

    db.SubmitChanges();
    db.Transaction.Commit();
    var COid = (from a in db.tblUsers where
a.UserName == data.UserName select a.UserId).FirstOrDefault();
    string mess;
    EteTblUser odata = new EteTblUser();
    mess = BalAuditTrail.GetAuditXml("TblUser",
odata, data);

    EteAuditTrail at = new EteAuditTrail();
    at = new EteAuditTrail
    {
        CreatedOn = data.CreatedOn,
        UserId = data.UId,
        ObjectID = Convert.ToInt32(COid), //
data.Id,

        Category = "Insert",
        Message = mess,
        AuditDate = DateTime.Today,
        PageId =163,// data.Pageid,
    };
    bool res = new BalAuditTrail().Insert(at);

    result = true;
}
catch (Exception exc)
{
    db.Transaction.Rollback();
    throw exc;
}
}

return result;
}
public bool InsertAddress(Etetbluser_address data)
{
    bool result = false;
    EteTblUser _user = new EteTblUser();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

```

```

db.Connection.Open();
db.Transaction = db.Connection.BeginTransaction();
int userid = (int)GetNextUserId();

try
{
    tbluser_address _cs = new tbluser_address
    {
        userid = data.userid ,
        address_type=data.address_type,
        address1=data.address1,
        address2 = data.address2,
        address3 = data.address3,
        City=data.city,
        country=data.country,
        ContactNo = data.ContactNo,
    };
    db.tbluser_addresses.InsertOnSubmit(_cs);

    db.SubmitChanges();
    db.Transaction.Commit();
    //var COid = (from a in db.tbluser_addresses where a.UserName ==
data.UserName select a.UserId).FirstOrDefault();
    //string mess;
    //EteTblUser odata = new EteTblUser();
    //mess = BalAuditTrail.GetAuditXml("tbluser_address", odata, data);
    //EteAuditTrail at = new EteAuditTrail();
    //at = new EteAuditTrail
    //{
    //    CreatedOn = data.CreatedOn,
    //    UserId = data.UId,
    //    ObjectId = Convert.ToInt32(COid), // data.Id,
    //    Category = "Insert",
    //    Message = mess,
    //    AuditDate = DateTime.Today,
    //    PageId = 163, // data.Pageid,
    //};
    //bool res = new BalAuditTrail().Insert(at);

    result = true;
}
catch (Exception exc)
{
    db.Transaction.Rollback();
    throw exc;
}

return result;
}
public bool Update(EteTblUser data)
{
    EteTblUser odata = new EteTblUser();
    bool result = false;

```

```

        string mess;
        using (var db = new
DowPortalDataContext(Configuration.ConnectionString))
        {
            db.Connection.Open();
            db.Transaction = db.Connection.BeginTransaction();

//try
            {
                var _user = db.tblUsers.Where(e =>
e.UserName == data.UserName).SingleOrDefault();

                if (_user == null)
                {
                    throw (new
ApplicationException("Unable to find the User record"));
                }
                odata=new EtetblUser
                {
                    UserId = _user.UserId,
                    UserName = _user.UserName,
                    Password = _user.Password,
                    FirstName = _user.FirstName,
                    LastName = _user.LastName,
                    EmailID = _user.EmailID,
                    ContactNum = _user.ContactNum,
                    CreatedOn = _user.CreatedOn,
                    Status = _user.Status,
                    RoleID = _user.RoleID,
                    GroupId = _user.GroupId,
                    IsRemoved = _user.IsRemoved,
                    Department = _user.Department,
                    Region = _user.Region,
                    UpdatedOn = _user.UpdatedOn,
                    DealerId = _user.DealerId,
                };
                if (odata.CreatedOn == null) { odata.CreatedOn
= DateTime.Today; }
                if (data.CreatedOn == null) { data.CreatedOn =
odata.CreatedOn; }

                //_user.UserId = data.UserId;

                _user.UserName = data.UserName;

                _user.Password = data.Password;

                _user.FirstName = data.FirstName;

                _user.LastName = data.LastName;

                _user.EmailID = data.EmailID;

                _user.ContactNum = data.ContactNum;
            }
        }
    }
}

```

```

        _user.CreatedOn = data.CreatedOn;

        _user.Status = data.Status;

        _user.RoleID = data.RoleID;

        _user.GroupId = data.GroupId;

        _user.IsRemoved = data.IsRemoved;

        _user.Department = data.Department;

        _user.Region = data.Region;

        _user.UpdatedOn = data.UpdatedOn;
        _user.DealerId = data.DealerId;

        db.SubmitChanges();
        db.Transaction.Commit();

        result = true;
    }
    //catch (Exception exc)
    //{
    //    db.Transaction.Rollback();

    //    throw exc;
    //}
//try
    {
        mess =BalAuditTrail.GetAuditXml("tblUser", odata,
data);

        EteAuditTrail at = new EteAuditTrail();
        at = new EteAuditTrail {
            CreatedOn =
Convert.ToDateTime(odata.CreatedOn), UserId = data.UId,
            ObjectID = data.UserId,
            Category = "Edit",
            Message = mess,
            AuditDate = DateTime.Today,
            PageId = data.PageId,
        };
        bool res = new BalAuditTrail().Insert(at);
    }
    //catch (Exception exc)
    //{

    //    throw exc;
    //}

```

```

        return result;
    }
    public EtetblUser getuser(string UserName)
    {
        EtetblUser result = new EtetblUser();
        using (var db = new
DowPortalDataContext(Configuration.ConnectionString))
        {
            db.Connection.Open();
            var users = (from c in db.tblUsers where c.UserName
== UserName select c).ToList();
            foreach (var row in users)
            {
                result = new EtetblUser
                {
                    UserId = row.UserId,
                    UserName = row.UserName,
                    Password = row.Password,
                    FirstName = row.FirstName,
                    LastName = row.LastName,
                    EmailID = row.EmailID,
                    ContactNum = row.ContactNum,
                    CreatedOn = row.CreatedOn,
                    Status = row.Status,
                    RoleID = row.RoleID,
                    GroupId = row.GroupId,
                    IsRemoved = row.IsRemoved,
                    Department = row.Department,
                    Region = row.Region,
                    UpdatedOn = row.UpdatedOn,
                    DealerId = row.DealerId,
                };
                break;
            }
        }
        return result;
    }
}

public EtetblUser getuserbyid(int UserId)
{
    EtetblUser result = new EtetblUser();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var users = (from c in db.tblUsers where c.UserId == UserId select c).ToList();
        foreach (var row in users)
        {
            result = new EtetblUser
            {
                UserId = row.UserId,
                UserName = row.UserName,
                Password = row.Password,
                FirstName = row.FirstName,
                LastName = row.LastName,
                EmailID = row.EmailID,
            };
        }
    }
}

```

```

        ContactNum = row.ContactNum,
        CreatedOn = row.CreatedOn,
        Status = row.Status,
        RoleID = row.RoleID,
        GroupId = row.GroupId,
        IsRemoved = row.IsRemoved,
        Department = row.Department,
        Region = row.Region,
        UpdatedOn = row.UpdatedOn,
        DealerId = row.DealerId,
    };
    break;
}
}

return result;
}
public List<EtetblUser> getsearchuser(string UserName)
{
    List<EtetblUser> result = new List<EtetblUser>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
//        var users = (from c in db.tblUsers where c.UserName == UserName select
c).ToList();
        List<GetSearchFinalUsersResult> users =
db.GetSearchFinalUsers(UserName).ToList();

        foreach (var row in users)
        {
            //SqlConnection connection = new SqlConnection(Configuration.ConnectionString);
            //connection.Open();
            //string sql = "SELECT name='~\\images\\FILE'+convert(varchar(5),UserId)+'-
0'+'.jpg',ContentType='IMAGE/JPEG', company_logo FROM tbluser_profile WHERE UserId
= @PID ";
            // SqlCommand cmd = new SqlCommand(sql, connection);
            // cmd.CommandType = CommandType.Text;
            // cmd.Parameters.AddWithValue("@PID", row.UserId);

            // SqlDataReader sqlDataReader = cmd.ExecuteReader();
            // string imageUrl="";
            // while (sqlDataReader.Read())
            // {
            //     byte[] bytes = (byte[])sqlDataReader["company_logo"];
            //     string base64String = Convert.ToBase64String(bytes, 0, bytes.Length);
            //     imageUrl = "data:image/png;base64," + base64String;
            // }
            System.Data.Linq.Binary Image = (from d in db.tbluser_profiles where d.userid
== row.UserId select d.Company_logo).SingleOrDefault();
            byte[] bytes = (byte[])Image.ToArray();
            string base64String = Convert.ToBase64String(bytes, 0, bytes.Length);
            string imageUrl = "data:image/png;base64," + base64String;
            result.Add(new EtetblUser
            {
                UserId = row.UserId,

```

```

        UserName = row.UserName,
        Password = row.Password,
        FirstName = row.FirstName,
        LastName = row.LastName,
        EmailID = row.EmailID,
        ContactNum = row.ContactNum,
        CreatedOn = row.CreatedOn,
        Status = row.Status,
        RoleID = row.RoleID,
        GroupId = row.GroupId,
        IsRemoved = row.IsRemoved,
        Department = row.Department,
        Region = row.Region,
        UpdatedOn = row.UpdatedOn,
        DealerId = row.DealerId,
        ImageURL=ImageUrl,
    });
    break;
}
}
return result;
}

public string GetUserAuditXml(EtetblUser oldd, EtetblUser newd)
{
    string Atrial = "";
    Atrial = BalAuditTrail.GetAuditXml("tblUser", oldd, newd);
    return Atrial;
}
public List<EtetblUser> getusers(string Curl)
{
    List<EtetblUser> _user = new List<EtetblUser>();
    using (var db = new
DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        int pgid = (new tblPagess()).GetPageId(Curl);
        string DealerName="",GroupName="";
        var users = (from c in db.tblUsers orderby
c.UserName select c).ToList();

        foreach (var row in users)
        {
            EteDealer getdealer = (new
Dealers()).getdealer(row.DealerId);
            if (getdealer != null)
            {
                DealerName =
getdealer.CompanyName;
            }

            EtetblPermissionGroup GettblPermissionGroup
= (new tblPermissionGroups()).GettblPermissionGroup(row.GroupId??0);
            if (GettblPermissionGroup != null)

```



```

        {
            GettblPermissionGroup.GroupName;
        }
        GroupName =

        _user.Add(new EtetblUser
        {
            UserId = row.UserId,
            UserName = row.UserName,
            Password = row.Password,
            FirstName = row.FirstName,
            LastName = row.LastName,
            EmailID = row.EmailID,
            ContactNum = row.ContactNum,
            CreatedOn = row.CreatedOn,
            Status = row.Status,
            RoleID = row.RoleID,
            GroupId = row.GroupId,
            IsRemoved = row.IsRemoved,
            Department = row.Department,
            Region = row.Region,
            UpdatedOn = row.UpdatedOn,
            DealerId=row.DealerId,
            DealerName=DealerName,
            PageId = pgid,
            GroupName=GroupName,
        });
    }
    return _user;
}
}
public List<EtetblUser> getusersbyusers(string Curl, int DealerId, int
GroupId)
{
    List<EtetblUser> _user = new List<EtetblUser>();
    using (var db = new
DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        int pgid = (new tblPagess()).GetPageId(Curl);
        string GroupName = "", DealerName="";
        var users = (from c in db.tblUsers where c.DealerId ==
DealerId select c).ToList();

        foreach (var row in users)
        {
            if (row.GroupId >= GroupId)
            {
                EtetblPermissionGroup
GettblPermissionGroup = (new
tblPermissionGroups()).GettblPermissionGroup(row.GroupId??0);
                if (GettblPermissionGroup != null)

```

```

        {
            GroupName =
        }
        EteDealer getdealer = (new
    if (getdealer != null)
    {
        DealerName =
    }
    _user.Add(new EteblUser
    {
        UserId = row.UserId,
        UserName = row.UserName,
        Password = row.Password,
        FirstName = row.FirstName,
        LastName = row.LastName,
        EmailID = row.EmailID,
        ContactNum = row.ContactNum,
        CreatedOn = row.CreatedOn,
        Status = row.Status,
        RoleID = row.RoleID,
        GroupId = row.GroupId,
        IsRemoved = row.IsRemoved,
        Department = row.Department,
        Region = row.Region,
        UpdatedOn = row.UpdatedOn,
        DealerId = row.DealerId,
        DealerName=DealerName,
        PageId = pgid,
        GroupName = GroupName,
    });
    }
}
return _user;
}
}
}
public List<Etebluser_address> getusersaddressbyuserid(int UserId, string Curl)
{
    List<Etebluser_address> _user = new List<Etebluser_address>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        int pgid = (new tblPagess()).GetPageId(Curl);
        var users = (from c in db.tbluser_addresses where c.userid == UserId select
c).ToList();
        string address_typedesc="";
        bool IsDefault;
        foreach (var data in users)
        {

```

```

        address_typedesc = (from e in db.tbluser_add_types where e.adid ==
data.address_type select e.adtype).SingleOrDefault();
        string countryname = (from d in db.countries where d.id==data.country
select d.countryname).SingleOrDefault();
        if (address_typedesc == null) { address_typedesc = ""; }
        if (countryname == null) { countryname = ""; }
        if ( data.IsDefault==false) { IsDefault = false; } else { IsDefault = true; }

        _user.Add(new EteTbluser_address
        {
            userid=data.userid,
            address_type=data.address_type,
            address1=data.address1,
            address2=data.address2,
            address3=data.address3,
            pincode=data.pincode,
            city=data.City,
            country=(int)data.country,
            countryname = countryname,
            ContactNo=data.ContactNo,
            address_typedesc = address_typedesc,
            IsDefault=(bool)IsDefault,
        });
    }
}

return _user;
}
public List<EteTblUser_emails> getuserEmailsbyuserid(int UserId, string Curl)
{
    List<EteTblUser_emails> _user = new List<EteTblUser_emails>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        int pgid = (new tblPagess()).GetPagelId(Curl);
        string EMSentUserName ,          EMRcvdUserName = "";
        var users = (from c in db.tblUser_Emails orderby c.CreatedOn descending
select c).ToList();

        foreach (var data in users)
        {
            EMSentUserName = (from d in db.tblUsers where
d.UserId==data.EMSentUserId select d.FirstName).SingleOrDefault();
            EMRcvdUserName = (from d in db.tblUsers where d.UserId ==
data.EMRcvdUserId select d.FirstName).SingleOrDefault();

            _user.Add(new EteTblUser_emails
            {EmailId=data.EmailId,
            UserId = data.UserId,
            EMSentUserId = data.EMSentUserId,
            EMRcvdUserId = data.EMRcvdUserId,
            EmSubject = data.EmSubject,
            EmBody=data.EmBody,
            CreatedOn=data.CreatedOn,

```

```

        EMSentUserName = EMSentUserName,
        EMRcvdUserName = EMRcvdUserName,

    });
}
}

return _user;
}
public List<EtetblUserImages> getuserImagesbyuserid(int UserId, string Curl)
{
    List<EtetblUserImages> _user = new List<EtetblUserImages>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        db.Connection.Open();
        int pgid = (new tblPagess()).GetPageId(Curl);
        SqlConnection connection = new SqlConnection(Configuration.ConnectionString);
        connection.Open();
        string sql = "SELECT name='~\\images\\FILE'+convert(varchar(5),UserId)+'-
'+convert(varchar(5),srno)+' .jpg',ContentType='IMAGE/JPEG', Images FROM tblUserImages
WHERE UserId = @PID ";
        SqlCommand cmd = new SqlCommand(sql, connection);
        cmd.CommandType = CommandType.Text;
        cmd.Parameters.AddWithValue("@PID", UserId);
        int srno=0;
        SqlDataReader sqlDataReader = cmd.ExecuteReader();

        while (sqlDataReader.Read())
        {
            srno++;

            byte[] bytes = (byte[])sqlDataReader["Images"];
            string base64String = Convert.ToBase64String(bytes, 0, bytes.Length);
            string imageUrl = "data:image/png;base64," + base64String;
            var users = (from c in db.tblUserImages where c.UserId == UserId && c.srno==srno
select c).FirstOrDefault();

            _user.Add(new EtetblUserImages
            {
                UserId = UserId,
                srno = srno,
                Images = users.Images,
                imageUrl = imageUrl,
            });
        }
    }

    // foreach (var data in users)
    // {
    //     string imageUrl = "";
    //     if (data.Images != null)
    //     {
    //         byte[] bytes = (byte[])data.Images.ToArray();
    //         string base64String = Convert.ToBase64String(bytes, 0, bytes.Length);
    //         imageUrl = "data:image/png;base64," + base64String;
    //     }
    // }

```

```

//    }

//    _user.Add(new EtetblUserImages
//    {UserId=data.UserId,
//      srno=data.srno,
//      Images=data.Images,
//      ImageUrl = ImageURL,
//    });
//  }
//}

return _user;
}

public DataTable GetUserImagesDTbyUid(int UserId,string Curl="")
{
    DataTable dt = new DataTable();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        int pgid = (new tblPagess()).GetPagelId(Curl);
        SqlConnection connection = new SqlConnection(Configuration.ConnectionString);
        connection.Open();
        string sql = "SELECT name='\\images\\FILE'+convert(varchar(5),UserId)+'-
'+convert(varchar(5),srno)+'.jpg',ContentType='IMAGE/JPEG', Images FROM tblUserImages
WHERE UserId = @PID ";
        SqlCommand cmd = new SqlCommand(sql, connection);
        cmd.CommandType = CommandType.Text;
        cmd.Parameters.AddWithValue("@PID", UserId);
        SqlDataAdapter sda = new SqlDataAdapter();
        sda.SelectCommand = cmd;
        sda.Fill(dt);

        return dt;
    }
}

public bool InsertLog(tblLogIn data)
{
    bool result = false;
    tblLogIn _user = new tblLogIn();
    using (var db = new
DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Connection.BeginTransaction();

        try
        {
            tblLogIn _cs = new tblLogIn
            {
                LogInId = data.LogInId,

                LogInTime = data.LogInTime,

```

```

        LogOutTime = data.LogOutTime,
        UserId = data.UserId,
        LoginStatus = data.LoginStatus,
        SourceIp = data.SourceIp,
        MachineName = data.MachineName,
        Description = data.Description,
        City = data.City,
        Country = data.Country

    };
    db.tblLogIns.InsertOnSubmit(_cs);

    db.SubmitChanges();
    db.Transaction.Commit();

    result = true;
}
catch (Exception exc)
{

    db.Transaction.Rollback();
    throw exc;
}
}

return result;
}

public string GetIP()
{
    string strHostName = "";
    strHostName = System.Net.Dns.GetHostName();
    IPHostEntry ipEnTry =
System.Net.Dns.GetHostEntry(strHostName);
    IPAddress[] addr = ipEnTry.AddressList;
    return (addr[addr.Length - 1].ToString());
}

public string GetMachineName()
{
    string clientPCName="";
    return clientPCName;
}

public bool InsertProfile(Etetbluser_profile data)
{

```

```

bool result = false;
Etetbluser_profile _user = new Etetbluser_profile();
using (var db = new DowPortalDataContext(Configuration.ConnectionString))
{
    db.Connection.Open();
    db.Transaction = db.Connection.BeginTransaction();

    try
    {
        tbluser_profile _cs = new tbluser_profile
        {
            userid = data.userid,
            Business_Type = data.Business_Type,
            Legal_Structure = data.Legal_Structure,
            Year_Established = data.Year_Established,
            Number_of_Employees = data.Number_of_Employees,
            Company_Description = data.Company_Description,
            Company_Website_URL = data.Company_Website_URL,
            Turnover_per_Year = data.Turnover_per_Year,
            CID = data.CID,
            Quality_Assurance_Accreditation = data.Quality_Assurance_Accreditation,
            Registration_Number = data.Registration_Number,
            D_U_N_S_Number = data.D_U_N_S_Number,
            Specializing_in = data.Specializing_in,
            Monthly_Production_Capacity = data.Monthly_Production_Capacity,
            UOMId = data.UOMId,
            Company_logo = data.Company_logo,
            CreatedOn = data.CreatedOn,
            UpdatedOn = data.UpdatedOn,
            uid = data.uid,
        };
        db.tbluser_profiles.InsertOnSubmit(_cs);

        db.SubmitChanges();
        db.Transaction.Commit();
        var COid = (from a in db.tblUsers where a.UserId == data.userid select
a.UserId).FirstOrDefault();
        string mess;
        Etetbluser_profile odata = new Etetbluser_profile();
        mess = BalAuditTrail.GetAuditXml("tbluser_profile", odata, data);
        EteAuditTrail at = new EteAuditTrail();
        at = new EteAuditTrail
        {
            CreatedOn = data.CreatedOn,
            UserId = data.userid,
            Objectid = Convert.ToInt32(COid), // data.Id,
            Category = "Insert",
            Message = mess,
            AuditDate = DateTime.Today,
            Pageid = 163, // data.Pageid,
        };
        bool res = new BalAuditTrail().Insert(at);

        result = true;
    }
}

```

```

    }
    catch (Exception exc)
    {
        db.Transaction.Rollback();
        throw exc;
    }
}

return result;
}
public bool UpdateProfile(Etetbluser_profile data)
{
    Etetbluser_profile odata = new Etetbluser_profile();
    bool result = false;
    string mess;
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        //try
        {

            var _user = db.tbluser_profiles.Where(e => e.userid ==
data.userid).SingleOrDefault();

            if (_user == null)
            {
                result = InsertProfile(data);
            }
            else
            {
                odata = new Etetbluser_profile
                {
                    userid = data.userid,
                    Business_Type = data.Business_Type,
                    Legal_Structure = data.Legal_Structure,
                    Year_Established = data.Year_Established,
                    Number_of_Employees = data.Number_of_Employees,
                    Company_Description = data.Company_Description,
                    Company_Website_URL = data.Company_Website_URL,
                    Turnover_per_Year = data.Turnover_per_Year,
                    CID = data.CID,
                    Quality_Assurance_Accreditation = data.Quality_Assurance_Accreditation,
                    Registration_Number = data.Registration_Number,
                    D_U_N_S_Number = data.D_U_N_S_Number,
                    Specializing_in = data.Specializing_in,
                    Monthly_Production_Capacity = data.Monthly_Production_Capacity,
                    UOMId = data.UOMId,
                    Company_logo = data.Company_logo,
                    CreatedOn = data.CreatedOn,
                    UpdatedOn = data.UpdatedOn,
                    uid = data.uid,

                };
            }
        }
    }
}

```



```

if (odata.CreatedOn == null) { odata.CreatedOn = DateTime.Today; }
if (data.CreatedOn == null) { data.CreatedOn = odata.CreatedOn; }

    _user.userid = data.userid;
    _user.Business_Type = data.Business_Type;
    _user.Legal_Structure = data.Legal_Structure;
    _user.Year_Established = data.Year_Established;
    _user.Number_of_Employees = data.Number_of_Employees;
    _user.Company_Description = data.Company_Description;
    _user.Company_Website_URL = data.Company_Website_URL;
    _user.Turnover_per_Year = data.Turnover_per_Year;
    _user.CID = data.CID;
    _user.Quality_Assurance_Accreditation = data.Quality_Assurance_Accreditation;
    _user.Registration_Number = data.Registration_Number;
    _user.D_U_N_S_Number = data.D_U_N_S_Number;
    _user.Specializing_in = data.Specializing_in;
    _user.Monthly_Production_Capacity = data.Monthly_Production_Capacity;
    _user.UOMId = data.UOMId;
    _user.Company_logo = data.Company_logo;
    _user.CreatedOn = data.CreatedOn;
    _user.UpdatedOn = data.UpdatedOn;
    _user.uid = data.uid;
    db.SubmitChanges();
    db.Transaction.Commit();

    result = true;
}
}
//catch (Exception exc)
//{
//    db.Transaction.Rollback();

//    throw exc;
//}
}
//try
{
    mess = BalAuditTrail.GetAuditXml("tblUser", odata, data);
    EteAuditTrail at = new EteAuditTrail();
    at = new EteAuditTrail
    {
        CreatedOn = Convert.ToDateTime(odata.CreatedOn),
        UserId = data.userid,
        ObjectId = data.userid,
        Category = "Edit",
        Message = mess,
        AuditDate = DateTime.Today,
        PageId = data.PageId,
    };
    bool res = new BalAuditTrail().Insert(at);
}
//catch (Exception exc)
//{

```

```

// throw exc;
//}

return result;
}
public Etetbluser_profile getuserprofilebyid(int UserId)
{
    Etetbluser_profile result = new Etetbluser_profile();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var data = (from c in db.tbluser_profiles where c.userid == UserId select
c).SingleOrDefault();
        if (data != null)
        {
            result = new Etetbluser_profile
            {
                userid = data.userid,
                Business_Type = data.Business_Type,
                Legal_Structure = data.Legal_Structure,
                Year_Established = data.Year_Established,
                Number_of_Employees = data.Number_of_Employees,
                Company_Description = data.Company_Description,
                Company_Website_URL = data.Company_Website_URL,
                Turnover_per_Year = data.Turnover_per_Year,
                CID = data.CID,
                Quality_Assurance_Accreditation = data.Quality_Assurance_Accreditation,
                Registration_Number = data.Registration_Number,
                D_U_N_S_Number = data.D_U_N_S_Number,
                Specializing_in = data.Specializing_in,
                Monthly_Production_Capacity = data.Monthly_Production_Capacity,
                UOMId = data.UOMId,
                Company_logo = data.Company_logo,
                CreatedOn = data.CreatedOn,
                UpdatedOn = data.UpdatedOn,
                uid = data.uid,

            };

        }
    }

    return result;
}
public bool InsertPhoto(Etetbluser_profile data)
{
    bool result = false;
    //Eteproduct _user = new Eteproduct();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();
    }
}

```

```

        var user = (from c in db.tbluser_profiles where c.userid == data.userid select
c).SingleOrDefault();
        if (user != null)
        {
            user.Company_logo = data.Company_logo;
        }
        else
        {
            tbluser_profile _cs = new tbluser_profile
            {
                userid = data.userid,
                CreatedOn=DateTime.Now,
                Company_logo = data.Company_logo,
                UpdatedOn = DateTime.Now,
            };
            db.tbluser_profiles.InsertOnSubmit(_cs);
        }
        db.SubmitChanges();
        db.Transaction.Commit();

        result = true;
    }

    return result;
}
public bool InsertImages(EtetblUserImages data)
{
    bool result = false;
    //Eteproduct _user = new Eteproduct();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();
        int srno = (from c in db.tblUserImages where c.UserId == data.UserId orderby
c.srno descending select c.srno).FirstOrDefault();
        srno++;
        tblUserImage _cs = new tblUserImage
        {
            UserId = data.UserId,
            srno=srno,
            CreatedOn = DateTime.Now,
            Images = data.Images,
        };
        db.tblUserImages.InsertOnSubmit(_cs);

        db.SubmitChanges();
        db.Transaction.Commit();

        result = true;
    }
}

```

```

        return result;
    }
}

}

****

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using DowBEntity;
using DowDAL;
namespace DowBAL
{

    public class UserPrivileges
    {
        public static bool UserPrivilegeNameExists(int Id)
        {
            bool exists = false;

            using (var db = new DowPortalDataContext(Configuration.ConnectionString))
            {

                db.Connection.Open();

                var _cust = (from c in db.UserPrivileges where c.Privileged == Id select
c).SingleOrDefault();

                if (_cust != null)
                {
                    exists = true;
                }

            }

            return exists;
        }
        //public int? GetNextUserPrivileged()
        //{
        //    int? result = 0;
        //    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        //    {
        //        GetNextUserPrivilegedResult lst =
db.GetNextUserPrivileged().SingleOrDefault();
        //        result = lst.NextUserPrivileged;

        //    }

        //    return result;
        //}
    }
}

```

```

public bool Insert(EteUserPrivilege data)
{
    bool result = false;
    EteUserPrivilege _user = new EteUserPrivilege();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {
            UserPrivilege _cs = new UserPrivilege
            {
                PrivilegedId = data.PrivilegedId,
                UserId = data.UserId,
                //IsActive = data.IsActive,
                CreatedOn = data.CreatedOn,
                //UpdatedOn = data.CreatedOn
            };
            db.UserPrivileges.InsertOnSubmit(_cs);

            db.SubmitChanges();
            db.Transaction.Commit();

            result = true;
        }
        catch (Exception exc)
        {
            db.Transaction.Rollback();
            throw exc;
        }
    }

    return result;
}

public bool Update(EteUserPrivilege data)
{
    bool result = false;

    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {
            var _user = db.UserPrivileges.Where(e => e.PrivilegedId ==
data.PrivilegedId).SingleOrDefault();

            if (_user == null)
            {
                throw (new ApplicationException("Unable to find the User record"));
            }
        }
    }
}

```

```

        _user.Privileged = data.Privileged;
        _user.UserId = data.UserId;
        //_user.IsActive = data.IsActive;

        //_user.UpdatedOn = data.UpdatedOn;

        db.SubmitChanges();
        db.Transaction.Commit();

        result = true;
    }
    catch (Exception exc)
    {
        db.Transaction.Rollback();

        throw exc;
    }
}

return result;
}
public EteUserPrivilege getUserPrivilege(int name)
{
    EteUserPrivilege result = new EteUserPrivilege();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var users = (from c in db.UserPrivileges where c.Privileged == name select
c).ToList();
        foreach (var data in users)
        {
            result = new EteUserPrivilege
            {
                Privileged = data.Privileged,
                UserId = data.UserId,
                //IsActive = data.IsActive,

                CreatedOn = data.CreatedOn
            };
            break;
        }
    }

    return result;
}
public List<EteUserPrivilege> GetUserPrivilege()
{
    List<EteUserPrivilege> _user = new List<EteUserPrivilege>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        db.Connection.Open();

```

```

var users = (from c in db.UserPrivileges select c).ToList();
foreach (var data in users)
{
    _user.Add(new EteUserPrivilege
    {
        Privileged = data.Privileged,
        UserId = data.UserId,
        //IsActive = data.IsActive,
        CreatedOn = data.CreatedOn
    });
}

return _user;
}
}

```

```

}
}

```

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using DowBEntity;
using DowDAL;
namespace DowBAL
{
    public class UserRoles
    {
        public static bool UserRoleNameExists(int Id)
        {
            bool exists = false;

            using (var db = new DowPortalDataContext(Configuration.ConnectionString))
            {
                db.Connection.Open();

                var _cust = (from c in db.UserRoles where c.RoleId == Id select
c).SingleOrDefault();

                if (_cust != null)
                {
                    exists = true;
                }
            }

            return exists;
        }
    }
}

```

```

}
//public int? GetNextUserRoleId()
//{

// int? result = 0;
// using (var db = new DowPortalDataContext(Configuration.ConnectionString))
// {
//     GetNextUserRoleIdResult lst = db.GetNextUserRoleId().SingleOrDefault();
//     result = lst.NextUserRoleId;

// }

// return result;
//}

public bool Insert(EteUserRole data)
{
    bool result = false;
    EteUserRole _user = new EteUserRole();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {
            {
                UserRole _cs = new UserRole
                {
                    UserId = data.UserId,
                    RoleId = data.RoleId,
                    //BMSUser = data.BMSUser,
                    //Role = data.Role,
                    CreatedOn = data.CreatedOn,
                    //UpdatedOn = data.CreatedOn
                };
                db.UserRoles.InsertOnSubmit(_cs);

                db.SubmitChanges();
                db.Transaction.Commit();

                result = true;
            }
            catch (Exception exc)
            {
                {
                    db.Transaction.Rollback();
                    throw exc;
                }
            }
        }

        return result;
    }
}

public bool Update(EteUserRole data)
{
    bool result = false;

```



```

using (var db = new DowPortalDataContext(Configuration.ConnectionString))
{
    db.Connection.Open();
    db.Transaction = db.Connection.BeginTransaction();

    try
    {
        var _user = db.UserRoles.Where(e => e.RoleId ==
data.Role).SingleOrDefault();

        if (_user == null)
        {
            throw (new ApplicationException("Unable to find the User record"));
        }

        _user.UserId = data.UserId;
        _user.RoleId = data.RoleId;
        //_user.IsActive = data.IsActive;

        //_user.UpdatedOn = data.UpdatedOn;

        db.SubmitChanges();
        db.Transaction.Commit();

        result = true;
    }
    catch (Exception exc)
    {
        db.Transaction.Rollback();

        throw exc;
    }
}

return result;
}
public EteUserRole getUserRole(int name)
{
    EteUserRole result = new EteUserRole();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var users = (from c in db.UserRoles where c.RoleId == name select c).ToList();
        foreach (var data in users)
        {
            result = new EteUserRole
            {
                RoleId = data.RoleId,
                UserId = data.UserId,
                //IsActive = data.IsActive,

                CreatedOn = data.CreatedOn
            }
        }
    }
}

```

```

        };
        break;
    }
}

return result;
}
public List<EteUserRole> GetUserRole()
{
    List<EteUserRole> _user = new List<EteUserRole>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        db.Connection.Open();

        var users = (from c in db.UserRoles select c).ToList();
        foreach (var data in users)
        {
            _user.Add(new EteUserRole
            {
                RoleId = data.RoleId,
                UserId = data.UserId,
                //IsActive = data.IsActive,
                CreatedOn = data.CreatedOn
            });
        }

        return _user;
    }
}
}
}

```

```

using System;
using System.IO;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using DowBEntity;
using DowDAL;
using System.Xml;
using System.Xml.Serialization;
using System.Xml.XmlConfiguration;
namespace DowBAL
{

    public class ActionTasks
    {

```

```

//definition for callingpageid=1-customer,2-order

public bool ActionTaskNameExists(string name)
{
    bool exists = false;

    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();

        var _cust = (from c in db.ActionTasks where c.Summary == name select
c).SingleOrDefault();

        if (_cust != null)
        {
            exists = true;
        }
    }

    return exists;
}

public bool Insert(EteActionTask data)
{
    bool result = false;
    EteActionTask _user = new EteActionTask();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();
        if (data.ActionedById == 0) { data.ActionedById = null; }
        if (data.OwnedById == 0) { data.OwnedById = data.RaisedById; }
        if (data.QueueId == 0) { data.QueueId = 1; }
        try
        {
            ActionTask _cs = new ActionTask
            {
                Id = data.Id,
                CreatedOn = data.CreatedOn,
                QueueId = data.QueueId,
                OwnedById = data.OwnedById,
                RaisedById = data.RaisedById,
                ActionedById = data.ActionedById,
                RaisedDate = data.RaisedDate,
                ActionedDate = data.ActionedDate,
                Status = data.Status,
                Summary = data.Summary,
                Details = data.Details,
                SubjectId = data.SubjectId,
                CallingPageId = data.CallingPageId,
                DueDate = data.DueDate,
                TaskType = data.TaskType,
                TaskOutcome = data.TaskOutcome,
                OutcomeNotes = data.OutcomeNotes,
            }

```

```

        CLI = data.CLI,
        RelateTo = data.RelateTo,
        RelateToURL = data.RelateToURL,
    };
    db.ActionTasks.InsertOnSubmit(_cs);

    db.SubmitChanges();
    db.Transaction.Commit();
    int id = (from d in db.ActionTasks where d.QueueId == data.QueueId &&
d.SubjectId == data.SubjectId && d.CallingPageId == data.CallingPageId && d.Summary ==
data.Summary select d.Id).FirstOrDefault();
    string mess;
    EteActionTask odata = new EteActionTask();

    odata.Status = '0';
    //odata.CreatedOn = "";
    odata.TaskType = '0';
    mess = BalAuditTrail.GetAuditXml("ActionTask", odata, data);
    EteAuditTrail at = new EteAuditTrail();
    at = new EteAuditTrail
    {
        CreatedOn = data.CreatedOn,
        UserId = data.UId,
        ObjectId = id, // data.SubjectId,
        Category = "Insert",
        Message = mess,
        AuditDate = DateTime.Today,
        PageId = data.PageId,
    };
    bool res = new BalAuditTrail().Insert(at);
    result = true;
}
catch (Exception exc)
{
    db.Transaction.Rollback();
    throw exc;
}
}

return result;
}
public bool Update(EteActionTask data)
{
    bool result = false;
    string mess;
    EteActionTask odata = new EteActionTask();

    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try

```

```

{
    var _user = db.ActionTasks.Where(e => e.Id == data.Id).SingleOrDefault();

    if (_user == null)
    {
        throw (new ApplicationException("Unable to find the User record"));
    }

    odata = new EteActionTask
    {
        Id = _user.Id,
        CreatedOn = _user.CreatedOn,
        QueueId = _user.QueueId,
        OwnedById = _user.OwnedById,
        RaisedById = _user.RaisedById,
        ActionedById = _user.ActionedById,
        RaisedDate = _user.RaisedDate,
        ActionedDate = _user.ActionedDate,
        Status = _user.Status,
        Summary = _user.Summary,
        Details = _user.Details,
        //SubjectId = _user.SubjectId,
        //CallingPageId = _user.CallingPageId,
        DueDate = _user.DueDate,
        TaskType = _user.TaskType,
        TaskOutcome = _user.TaskOutcome,
        OutcomeNotes = _user.OutcomeNotes,
        CLI = _user.CLI,
        RelateTo = _user.RelateTo,
        RelateToURL = _user.RelateToURL,
    };

    if (odata.CreatedOn == null) { odata.CreatedOn = DateTime.Today; }
    if (data.QueueId == 0) { data.QueueId = 1; }
    //_user.Id = data.Id;
    _user.CreatedOn = data.CreatedOn;
    _user.QueueId = data.QueueId;
    _user.OwnedById = data.OwnedById;
    _user.RaisedById = data.RaisedById;
    _user.ActionedById = data.ActionedById;
    _user.RaisedDate = data.RaisedDate;
    _user.ActionedDate = data.ActionedDate;
    _user.Status = data.Status;
    _user.Summary = data.Summary;
    _user.Details = data.Details;
    //_user.SubjectId = data.SubjectId;
    //_user.CallingPageId = data.CallingPageId;
    _user.DueDate = data.DueDate;
    _user.TaskType = data.TaskType;
    _user.TaskOutcome = data.TaskOutcome;
    _user.OutcomeNotes = data.OutcomeNotes;
    _user.CLI = data.CLI;
    _user.RelateTo = data.RelateTo;
    _user.RelateToURL = data.RelateToURL;
}

```

```

        db.SubmitChanges();
        db.Transaction.Commit();
        result = true;
    }
    catch (Exception exc)
    {
        db.Transaction.Rollback();

        throw exc;
    }
    mess = BalAuditTrail.GetAuditXml("ActionTask", odata, data);
    EteAuditTrail at = new EteAuditTrail();
    at = new EteAuditTrail
    {
        CreatedOn = odata.CreatedOn,
        UserId = data.UId,
        Objectid = data.Id,
        Category = "Edit",
        Message = mess,
        AuditDate = DateTime.Today,
        PageId = data.PageId,
    };
    bool res = new BalAuditTrail().Insert(at);
}

return result;
}
private string GetRelateTo(int SubjectID, int CallingPageID)
{
    string Result = "";
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        if (CallingPageID == 2)
        {
            // var CustDetails = (from d in db.Customers where d.Id == SubjectID select
            d.ExternalReferenceNumber).SingleOrDefault();
            // if (CustDetails != null) { Result = "Customer: (" + CustDetails + ")"; }
            //}
            //else if (CallingPageID == 3)
            //{
            // var CustOrdDetails = (from d in db.CustomerOrders where d.Id == SubjectID
            select d.OrderReferenceString).SingleOrDefault();
            // if (CustOrdDetails != null) { Result = "CustomerOrder: (" + CustOrdDetails +
            ")"; }
            }
        }
    }
    return Result;
}
public EteActionTask getActionTask(int Id)
{
    EteActionTask result = new EteActionTask();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))

```

```

{
    string stname = "";
    db.Connection.Open();
    var users = (from c in db.ActionTasks where c.Id == Id select c).ToList();
    foreach (var data in users)
    {
        if (data.OwnedById == null)
        {
            if (data.RaisedById != null)
            {
                data.OwnedById = data.RaisedById;
            }
            else
            {
                data.OwnedById = 0;
                data.RaisedById = 0;
            }
        }
        string OwnName = new tblUsers().getusername((int)data.OwnedById);
        var st = (from d in db.TaskStatus where d.Id == data.Status select
d.TStatusName).SingleOrDefault();
        if (st != null) { stname = st; }
        if (data.Status=='A' && data.DueDate<DateTime.Now)
        {
            data.Status = 'O';
        }
        result = new EteActionTask
        {
            Id = data.Id,
            CreatedOn = data.CreatedOn,
            QueueId = data.QueueId,
            OwnedById = data.OwnedById,
            RaisedById = data.RaisedById,
            ActionedById = data.ActionedById,
            RaisedDate = data.RaisedDate,
            ActionedDate = data.ActionedDate,
            Status = data.Status,
            Summary = data.Summary,
            Details = data.Details,
            SubjectId = data.SubjectId,
            CallingPageId = data.CallingPageId,
            DueDate = data.DueDate,
            TaskType = data.TaskType,
            TaskOutcome = data.TaskOutcome,
            OutcomeNotes = data.OutcomeNotes,
            CLI = data.CLI,
            StatusName = stname,
            OwnedByName = OwnName,
            RelateTo = GetRelateTo(data.SubjectId, data.CallingPageId),
            RelateToURL = data.RelateToURL,
        };
        break;
    }
}

```

```

return result;
}
public List<EteActionTask> GetActionTasks(string Curl, int OrdId, int callingpage)
{
    List<EteActionTask> _user = new List<EteActionTask>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        string sname = "";
        db.Connection.Open();
        int pgid = (new tblPagess()).GetPagelId(Curl);
        var users = (from c in db.ActionTasks where c.SubjectId == OrdId &&
c.CallingPagelId == callingpage select c).ToList();
        foreach (var data in users)
        {
            if (data.OwnedById == null)
            {
                if (data.RaisedById != null)
                {
                    data.OwnedById = data.RaisedById;
                }
                else
                {
                    data.OwnedById = 0;
                    data.RaisedById = 0;
                }
            }
            string OwnName = new tblUsers().getusername((int)data.OwnedById);
            var st = (from d in db.TaskStatus where d.Id == data.Status select
d.TStatusName).SingleOrDefault();
            if (st != null) { sname = st; }
            _user.Add(new EteActionTask
            {
                Id = data.Id,
                CreatedOn = data.CreatedOn,
                QueueId = data.QueueId,
                OwnedById = data.OwnedById,
                RaisedById = data.RaisedById,
                ActionedById = data.ActionedById,
                RaisedDate = data.RaisedDate,
                ActionedDate = data.ActionedDate,
                Status = data.Status,
                Summary = data.Summary,
                Details = data.Details,
                SubjectId = data.SubjectId,
                CallingPagelId = data.CallingPagelId,
                DueDate = data.DueDate,
                TaskType = data.TaskType,
                TaskOutcome = data.TaskOutcome,
                OutcomeNotes = data.OutcomeNotes,
                CLI = data.CLI,
                StatusName = sname,
                PagelId = pgid,
                OwnedByName = OwnName,
                RelateTo = data.RelateTo,
            });
        }
    }
}

```



```

        RelateToURL = data.RelateToURL,
    });
}

return _user;
}
}
public bool GetAllActionTasks()
{
    bool ret = false;
    List<EteActionTask> _user = new List<EteActionTask>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        //string stname = "";
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();
        //var usr = (from a in db.tblUser1s where a.bmsuserid == null select a).ToList();
        //foreach (var dt in usr)
        //{
        //    var bms = (from b in db.BmsUsrOrgs where b.UserName == dt.UserName
select b).SingleOrDefault();
        //    if(bms !=null){
        //        dt.bmsuserid = bms.Id;
        //    }
        //}
        //db.SubmitChanges();
        //db.Transaction.Commit();
        //db.Transaction = db.Connection.BeginTransaction();

        //    int pgid = (new tblPagess()).GetPageld(Curl);
        var users = (from c in db.ActionTasks where c.ActionedByld==null ||
c.OwnedByld==null select c).ToList();
        foreach (var data in users)
        {

        }

        db.SubmitChanges();
        db.Transaction.Commit();

        return ret;
    }
}
public List<EteActionTask> GetActionTasksByQueue(string Curl, int QueueId)
{
    List<EteActionTask> _user = new List<EteActionTask>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        string stname = "";
        db.Connection.Open();
        int pgid = (new tblPagess()).GetPageld(Curl);
    }
}

```

```

        var users = (from c in db.ActionTasks where c.QueueId == QueueId & c.Status !=
'C' select c).ToList();
        foreach (var data in users)
        {
            var st = (from d in db.TaskStatus where d.Id == data.Status select
d.TStatusName).SingleOrDefault();
            if (data.OwnedById == null)
            {
                if (data.RaisedById != null)
                {
                    data.OwnedById = data.RaisedById;
                }
                else
                {
                    data.OwnedById = 0;
                    data.RaisedById = 0;
                }
            }
            string OwnName = new tblUsers().getusername((int)data.OwnedById);
            int ActbyID = 0;
            if (data.ActionedById != null) { ActbyID = (int)data.ActionedById; }

            string ActionedName = new tblUsers().getusername(ActbyID);
            if (st != null) { stname = st; }
            _user.Add(new EteActionTask
            {
                Id = data.Id,
                CreatedOn = data.CreatedOn,
                QueueId = data.QueueId,
                OwnedById = data.OwnedById,
                RaisedById = data.RaisedById,
                ActionedById = data.ActionedById,
                RaisedDate = data.RaisedDate,
                ActionedDate = data.ActionedDate,
                Status = data.Status,
                Summary = data.Summary,
                Details = data.Details,
                SubjectId = data.SubjectId,
                CallingPageId = data.CallingPageId,
                DueDate = data.DueDate,
                TaskType = data.TaskType,
                TaskOutcome = data.TaskOutcome,
                OutcomeNotes = data.OutcomeNotes,
                CLI = data.CLI,
                StatusName = stname,
                PageId = pgid,
                OwnedByName = OwnName,
                ActionedByName = ActionedName,
                RelateTo = data.RelateTo,
                RelateToURL = data.RelateToURL,
            });
        }

        return _user;
    }

```

```

}
public List<EteActionTask> GetActionTasksByUser(string Curl, int UserId)
{
    List<EteActionTask> _user = new List<EteActionTask>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        string stname = "";
        db.Connection.Open();
        int pgid = (new tblPagess()).GetPagelId(Curl);
        //var users = (from c in db.ActionTasks where c.RaisedById == UserId select
c).ToList();
        var users = (from c in db.ActionTasks where c.OwnedById == UserId select
c).ToList();
        foreach (var data in users)
        {
            if (data.OwnedById == null) data.OwnedById = data.RaisedById;
            string OwnName = new tblUsers().getusername((int)data.OwnedById);
            if (data.DueDate < DateTime.Today && data.Status=='A') { data.Status = 'O'; }
            var st = (from d in db.TaskStatus where d.Id == data.Status select
d.TStatusName).SingleOrDefault();
            if (st != null) { stname = st; }
            _user.Add(new EteActionTask
            {
                Id = data.Id,
                CreatedOn = data.CreatedOn,
                QueueId = data.QueueId,
                OwnedById = data.OwnedById,
                RaisedById = data.RaisedById,
                ActionedById = data.ActionedById,
                RaisedDate = data.RaisedDate,
                ActionedDate = data.ActionedDate,
                Status = data.Status,
                Summary = data.Summary,
                Details = data.Details,
                SubjectId = data.SubjectId,
                CallingPagelId = data.CallingPagelId,
                DueDate = data.DueDate,
                TaskType = data.TaskType,
                TaskOutcome = data.TaskOutcome,
                OutcomeNotes = data.OutcomeNotes,
                CLI = data.CLI,
                StatusName = stname,
                PagelId = pgid,
                OwnedByName = OwnName,
                RelateTo = data.RelateTo,
                RelateToURL = data.RelateToURL,
            });
        }

        return _user;
    }
}
//public List<EteActionTask> GetActionTasksByFilter(string Curl, int dealerid, string
StatusId, int RoleId, int QueueId, int UserId)
//{

```

```

// try
// {
//     List<EteActionTask> _user = new List<EteActionTask>();
//     using (var db = new DowPortalDataContext(Configuration.ConnectionString))
//     {
//         //string stname = "";
//         db.Connection.Open();
//         int pgid = (new tblPagess()).GetPageld(Curl);
//         //         var users = (from c in db.ActionTasks where c.RaisedByld == Userld
select c).ToList();
//         List<GetActionTasksByFilterResult> users =
db.GetActionTasksByFilter(dealerid, Statusld, Roleld, QueueId, Userld).ToList();

//         foreach (var data in users)
//         {
//             if (data.OwnedByld == null)
//             {
//                 if (data.RaisedByld != null)
//                 {
//                     data.OwnedByld = data.RaisedByld;
//                 }
//                 else
//                 {
//                     data.OwnedByld = 0;
//                     data.RaisedByld = 0;
//                 }
//             }

//             //string OwnName = new tblUsers().getusername((int)data.OwnedByld);
//             //var st = (from d in db.TaskStatus where d.Id == data.Status select
d.TStatusName).SingleOrDefault();
//             //if (st != null) { stname = st; }
//             _user.Add(new EteActionTask
//             {
//                 Id = data.Id,//
//                 CreatedOn = data.CreatedOn,
//                 QueueId = data.QueueId,
//                 OwnedByld = data.OwnedByld,
//                 RaisedByld = data.RaisedByld,
//                 ActionedByld = data.ActionedByld,
//                 RaisedDate = data.RaisedDate,//
//                 ActionedDate = data.ActionedDate,//
//                 Status = data.Status,
//                 Summary = data.Summary,//
//                 Details = data.Details,
//                 SubjectId = data.SubjectId,
//                 CallingPageld = data.CallingPageld,
//                 DueDate = data.DueDate,//
//                 TaskType = data.TaskType,
//                 TaskOutcome = data.TaskOutcome,
//                 OutcomeNotes = data.OutcomeNotes,
//                 CLI = data.CLI,
//                 StatusName = data.StatusName,//
//                 Pageld = pgid,
//                 OwnedByName = data.OwnedByName,//

```

```

//          // RelateTo = data.RelateTo,
//          // RelateToURL = data.RelateToURL,

//      });
//      }

//      return _user;
//      }
//      }
//      catch (Exception exc)
//      {
//          return null;
//      }
//      }
//  }

public int GetActionTaskCount(int userid)
{
    int ActionTaskCounts = 0;

    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();

        //var FlgCount = (from c in db.Flags select c);
        var ActionTskCnts = (from c in db.ActionTasks where c.OwnedById == userid &&
c.Status == 'A' select c);
        if (ActionTskCnts != null)
        {
            ActionTaskCounts = ActionTskCnts.Count();
        }
        return ActionTaskCounts;
    }
}

public List<EteActionTask> GetAllActionTasks(string Curl, int dealerid, int groupid)
{
    List<EteActionTask> _user = new List<EteActionTask>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        string stname = "";
        db.Connection.Open();
        int pgid = (new tblPagess()).GetPagelId(Curl);
        if (groupid == 1)
        {
            var user1 = (from c in db.ActionTasks select c).ToList();

            var users = user1;
            foreach (var data in users)
            {
                var st = (from d in db.TaskStatus where d.Id == data.Status select
d.TStatusName).SingleOrDefault();
                if (data.OwnedById == null) { data.OwnedById = 0; } // data.RaisedById; }
                if (data.RaisedById == null) { data.RaisedById = 0; } // data.RaisedById; }
                if (data.QueueId == null) { data.QueueId = 1; }
                if (data.DueDate == null) { data.DueDate = data.RaisedDate; }
            }
        }
    }
}

```

```

int ActionedByUserId = data.ActionedById ?? default(int);
string OwnName = new tblUsers().getusername((int)data.OwnedById);
//string ActionedName = new tblUsers().getusername((int)data.ActionedById);
string ActionedName = new tblUsers().getusername(ActionedByUserId);
if (st != null) { stname = st; }
_user.Add(new EteActionTask
{
    Id = data.Id,
    CreatedOn = data.CreatedOn,
    QueueId = data.QueueId,
    OwnedById = data.OwnedById,
    RaisedById = data.RaisedById,
    ActionedById = data.ActionedById,
    RaisedDate = data.RaisedDate,
    ActionedDate = data.ActionedDate,
    Status = data.Status,
    Summary = data.Summary,
    Details = data.Details,
    SubjectId = data.SubjectId,
    CallingPageId = data.CallingPageId,
    DueDate = data.DueDate,
    TaskType = data.TaskType,
    TaskOutcome = data.TaskOutcome,
    OutcomeNotes = data.OutcomeNotes,
    CLI = data.CLI,
    StatusName = stname,
    PageId = pgid,
    OwnedByName = OwnName,
    ActionedByName = ActionedName,
    RelateTo = data.RelateTo,
    RelateToURL = data.RelateToURL,
});
}
}
else
{
    var user1 = (from c in db.ActionTasks select c).ToList();
    var users = user1;
    foreach (var data in users)
    {
        var st = (from d in db.TaskStatus where d.Id == data.Status select
d.TStatusName).SingleOrDefault();
        if (st != null) { stname = st; }
        string OwnName = new tblUsers().getusername((int)data.OwnedById);
        var usr = (from e in db.tblUsers where e.UserId == data.RaisedById select
e.DealerId).SingleOrDefault();
        if (usr == dealerid)
        {
            _user.Add(new EteActionTask
            {
                Id = data.Id,
                CreatedOn = data.CreatedOn,
                QueueId = data.QueueId,
                OwnedById = data.OwnedById,
                RaisedById = data.RaisedById,

```

```

        ActionedById = data.ActionedById,
        RaisedDate = data.RaisedDate,
        ActionedDate = data.ActionedDate,
        Status = data.Status,
        Summary = data.Summary,
        Details = data.Details,
        SubjectId = data.SubjectId,
        CallingPageId = data.CallingPageId,
        DueDate = data.DueDate,
        TaskType = data.TaskType,
        TaskOutcome = data.TaskOutcome,
        OutcomeNotes = data.OutcomeNotes,
        CLI = data.CLI,
        StatusName = sname,
        PageId = pgid,
        OwnedByName = OwnName,
        RelateTo = data.RelateTo,
        RelateToURL = data.RelateToURL,
    });
    }
}

return _user;
}
}
public String GetActionTaskName(int Id)
{
    String name = "";
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();

        var _cust = (from c in db.ActionTasks where c.Id == Id select c).SingleOrDefault();

        if (_cust != null)
        {
            name = _cust.Summary;
        }
        else
        {
            name = "";
        }
    }

    return name;
}
}
}

using System;

```

```

using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using DowBEntity;
using DowDAL;
namespace DowBAL
{
    public class AuditTrails
    {
        public bool Insert(EteAuditTrail data)
        {
            bool result = false;
            EteAuditTrail _user = new EteAuditTrail();
            using (var db = new DowPortalDataContext(Configuration.ConnectionString))
            {
                db.Connection.Open();
                db.Transaction = db.Connection.BeginTransaction();

                try
                {
                    AuditTrail _cs = new AuditTrail
                    {
                        Id = data.Id,
                        CreatedOn = Convert.ToDateTime( data.CreatedOn),
                        UserId = data.UserId,
                        Objectid = data.Objectid,
                        Category = data.Category,
                        Message = data.Message,
                        ExtraInfo = data.ExtraInfo,
                        AuditDate = data.AuditDate,
                    };
                    db.AuditTrails.InsertOnSubmit(_cs);

                    db.SubmitChanges();
                    db.Transaction.Commit();

                    result = true;
                }
                catch (Exception exc)
                {
                    db.Transaction.Rollback();
                    throw exc;
                }
            }

            return result;
        }
    }
}

using System;

```

```

using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Xml;
using System.Xml.Serialization;
using System.Xml.XmlConfiguration;
using System.IO;
using DowBEntity;
using DowBAL;
using DowDAL;
using System.Data;
//
using System.Text.RegularExpressions;
//using Microsoft.Practices.EnterpriseLibrary.Validation;
//using Microsoft.Practices.EnterpriseLibrary.Validation.Validators;

//using System.Web.Script.Serialization;

namespace DowBAL
{
    public class BalAuditTrail
    {
        DDLogTracer objlog = new DDLogTracer();

        public bool Insert(EteAuditTrail data)
        {
            bool result = false;
            EteAuditTrail _user = new EteAuditTrail();
            using (var db = new DowPortalDataContext(Configuration.ConnectionString))
            {
                db.Connection.Open();
                db.Transaction = db.Connection.BeginTransaction();
                try
                {
                    {
                        DateTime dt = Convert.ToDateTime("2000/01/01");
                        if (data.UserId == 0) { data.UserId = 1; }
                        if (data.CreatedOn < dt) data.CreatedOn = DateTime.Today;
                        AuditTrail _cs = new AuditTrail
                        {
                            Id = data.Id,
                            CreatedOn = data.CreatedOn ?? DateTime.Now, //? null
                            :System.DateTime.Now, //data.CreatedOn;
                            UserId = data.UserId,
                            Objectid = data.Objectid,
                            Category = data.Category,
                            Message = data.Message,
                            ExtraInfo = data.ExtraInfo,
                            AuditDate = data.AuditDate,
                            Pageid = data.Pageid,
                        };
                    };
                    db.AuditTrails.InsertOnSubmit(_cs);
                }
            }
        }
    }
}

```

```

        db.SubmitChanges();
        db.Transaction.Commit();

        result = true;
    }
    catch (Exception exc)
    {
        db.Transaction.Rollback();
        throw exc;
    }
}

return result;
}
public static string GetAuditXml(string tab_name, object oldd, object newd)
{
    DDLogTracer objlog = new DDLogTracer();
    string Atrial = "", OldValue, NewValue;
    string name;
    List<BalAuditTrail.DataXml> dxml = new List<BalAuditTrail.DataXml>();
    try
    {
        using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        {
            db.Connection.Open();
            var _cols = (from c in db.v_allcolumns where c.tab_name == tab_name select
c).ToList();
            foreach (var row in _cols)
            {
                name = row.col_name;

                OldValue = GetPropValue(oldd, name);
                NewValue = GetPropValue(newd, name);
                if (OldValue != null && NewValue != null)
                {
                    if (OldValue != NewValue)
                    {
                        dxml.Add(new BalAuditTrail.DataXml { DName = name, OValue =
OldValue, NValue = NewValue, });
                    }
                }
            }
            Atrial = XMLFromObject(dxml);
            return Atrial;
        }
    }
    catch (Exception exc)
    {
        objlog.Trace_Error(4, "b2bportal", "MangeUser", "Load", "Exception:" +
exc.ToString());
    }

    return Atrial;
}
public static string GetPropValue(object target, string propName)

```

```

{
    string result;
    try
    {

        result = target.GetType().GetProperty(propName).GetValue(target).ToString();
        if (result == null) { result = ""; }
        result = result.Replace( "[0x00]", " ");

        result = (result.Contains("01/01/0001") ? "" : result);
        return result;
    }
    catch
    {
        return "";
    }
}
}
public static string XMLFromObject(object o)
{
    string str;
    XmlSerializer serializer = new XmlSerializer(o.GetType());
    StringWriter sw = new StringWriter();
    XmlTextWriter tw = new XmlTextWriter(sw);
    serializer.Serialize(tw, o);
    str = sw.ToString();
    sw.Close();
    tw.Close();
    return str;
}
public static List<DataXml> ObjectToXML(string XmlOfAnObject, Type ObjectType)
{
    try
    {
        StringReader StrReader = new StringReader(Regex.Replace(XmlOfAnObject,
        "\\p{C}+", "")); // new StringReader(XmlOfAnObject.Replace("\x00", "[0x00]")); //StringReader
        StrReader = new StringReader(XmlOfAnObject);
        XmlSerializer serializer = new XmlSerializer(ObjectType);
        XmlTextReader XmlReader = new XmlTextReader(StrReader);
        List<DataXml> AnObject = (List<DataXml>)serializer.Deserialize(XmlReader);
        XmlReader.Close();
        StrReader.Close();
        return AnObject;
    }
    catch (Exception exc)
    {
        return null;
    }
}
}
public static List<ClassXml> ClasstToXML(string XmlOfAnObject, Type ObjectType)
{
    StringReader StrReader = new StringReader(XmlOfAnObject);
    XmlSerializer serializer = new XmlSerializer(ObjectType);
    XmlTextReader XmlReader = new XmlTextReader(StrReader);
    List<ClassXml> AnObject = (List<ClassXml>)serializer.Deserialize(XmlReader);
    XmlReader.Close();
}
}

```

```

        StrReader.Close();
        return AnObject;
    }
    public List<EteAuditTrailXml> GetLogTrail(int UserId,int groupid)
    {
        List<EteAuditTrailXml> lst = new List<EteAuditTrailXml>();
        List<DataXml> dlst = new List<DataXml>();
        string UserName = "";
        using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        {
            db.Connection.Open();
            List<tblLogIn> _cust = new List<tblLogIn>();
            {
                if (groupid == 1)
                {
                    _cust = (from c in db.tblLogIns orderby c.LogInTime select c).ToList();
                }
                else
                {
                    _cust = (from c in db.tblLogIns where c.UserId == UserId orderby
c.LogInTime select c).ToList();
                }
            }

            foreach (var row in _cust)
            {
                UserName = new tblUsers().getusername(Convert.ToInt32(row.UserId));

                lst.Add(new EteAuditTrailXml
                {
                    Id = row.LogInId,
                    CreatedOn = Convert.ToDateTime(row.LogInTime),
                    UserId = row.UserId,
                    ObjectId = Convert.ToInt32(row.LoginStatus),
                    Category = "Security",
                    AuditDate = Convert.ToDateTime(row.LogInTime), //row.LogOutTime,
                    DName = "",
                    OValue = "",
                    NValue = "",
                    UserName = UserName,
                    ObjectName = "Login Aceess",
                }
                );
            }

        }

    }
    }
    return lst;
}
public List<EteAuditTrailXml> GetAuditTrail(int PageId, int UserId,int Id,int groupid)
{

```

```

List<EteAuditTrailXml> lst = new List<EteAuditTrailXml>();
try
{
    List<DataXml> dlst = new List<DataXml>();
    string UserName = "";
    string Name = "";
    string _ref="";
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        List<GetAuditTrailbyPageIdResult> _cust = new
List<GetAuditTrailbyPageIdResult>(); //AuditTrail

        _cust = db.GetAuditTrailbyPageId(Id, PageId).ToList();
        {
            if (PageId == 0 && groupid == 1)
            {
                // _cust = (from c in db.AuditTrails orderby c.Id descending select
c).ToList();

            }
            else if (PageId != 0 && UserId != 0) //==1 original
            {

                if (PageId == 177 || PageId == 179 || PageId == 204||PageId==37)
                {
                    // _cust = (from c in db.AuditTrails where (c.PageId == 177 || c.PageId ==
179 || c.PageId == 204 || PageId == 37) && c.ObjectId == Id orderby c.Id descending select
c).ToList();

                    // _ref = (from d in db.CustomerOrders where d.Id == Id select
d.OrderReferenceString).FirstOrDefault();

                }
                else if (PageId == 131 || PageId == 172)
                {
                    /// _cust = (from c in db.AuditTrails where (c.PageId == 131 || c.PageId
== 172 ) && c.ObjectId == Id orderby c.Id descending select c).ToList();

                    var _task = (from a in db.ActionTasks where a.Id == Id select
a).FirstOrDefault();

                    if (_task.CallingPageId == 1 ||( _task.CallingPageId >= 30 &&
_task.CallingPageId <= 36) ) {
                        // _cust = (from c in db.AuditTrails where (c.PageId == 1 || (c.PageId
>= 30 && c.PageId <= 36)) && c.ObjectId == Id orderby c.Id descending select c).ToList();
                        _ref = Id.ToString();// (from d in db.Customers where d.Id ==
_task.SubjectId select d.ExternalReferenceNumber).FirstOrDefault();
                    }
                    else
                    {
                        // _cust = (from c in db.AuditTrails where (c.PageId == PageId &&
c.ObjectId == Id) orderby c.Id descending select c).ToList();
                        _ref = Id.ToString();// (from d in db.CustomerOrders where d.Id ==
_task.SubjectId select d.OrderReferenceString).FirstOrDefault();
                    }
                }
            }
        }
    }
}

```

```

else if (Pageld == 62 || Pageld == 171) {
    //_cust = (from c in db.AuditTrails where (c.Pageld == 62 || c.Pageld ==
171) && c.ObjectId == Id orderby c.Id descending select c).ToList();
    //_ref = (from d in db.Customers where d.Id == Id select
d.ExternalReferenceNumber).FirstOrDefault();
}
else{
    //_cust = (from c in db.AuditTrails where c.Pageld == Pageld &&
c.ObjectId==Id orderby c.Id descending select c).ToList();
    _ref=Id.ToString();
}
}

//else if (Pageld != 0 && UserId != 1)
//{
//    if (Pageld == 177 || Pageld == 179 || Pageld == 204 || Pageld == 37)
//    {
//        //_cust = (from c in db.AuditTrails where (c.Pageld == 177 || c.Pageld ==
179 || c.Pageld == 204 || Pageld == 37) && c.UserId == UserId && c.ObjectId == Id orderby
c.Id descending select c).ToList();

//        //_ref = (from d in db.CustomerOrders where d.Id == Id select
d.OrderReferenceString).FirstOrDefault();
//    }
//    else if (Pageld == 131 || Pageld == 172)
//    {
//        //_cust = (from c in db.AuditTrails where (c.Pageld == 131 || c.Pageld ==
172) && c.UserId == UserId && c.ObjectId == Id orderby c.Id descending select c).ToList();
//        var _task = (from a in db.ActionTasks where a.Id == Id select
a).FirstOrDefault();

//        if (_task.CallingPageld == 1 ||( _task.CallingPageld >= 30 &&
_task.CallingPageld <= 36) )
//        {
//            //_ref = (from d in db.Customers where d.Id == _task.SubjectId select
d.ExternalReferenceNumber).FirstOrDefault();
//        }
//        else
//        {
//            //_ref = (from d in db.CustomerOrders where d.Id == _task.SubjectId
select d.OrderReferenceString).FirstOrDefault();
//        }

//    }
//    else if (Pageld == 62 || Pageld == 171)
//    {
//        //_cust = (from c in db.AuditTrails where (c.Pageld == 62 || c.Pageld ==
171) && c.UserId == UserId && c.ObjectId == Id orderby c.Id descending select c).ToList();
//    }
//    else
//    {

//        //_cust = (from c in db.AuditTrails where c.Pageld == Pageld && c.UserId
== UserId && c.ObjectId == Id orderby c.Id descending select c).ToList();

```

```

//      _ref = (from d in db.Customers where d.Id == Id select
d.ExternalReferenceNumber).FirstOrDefault();
//  }
//}

foreach (var row in _cust)
{
    Name = (from d in db.tblPages where d.Pageld == row.Pageld select
d.PageName).SingleOrDefault();

    if (row.Message.Replace(" ", "").Length <= 0)
    {
        continue;
    }
    //row.Message = row.Message.Replace("\x00", "[0x00]");//ravi to remove
null value

    dlst = (new BalAuditTrail()).GETListfromXml(row.Message);

    UserName = new tblUsers().getusername(Convert.ToInt32(row.UserId));
    foreach (var rxml in dlst)
    {
        if (rxml.OValue != rxml.NValue)
        {
            rxml.OValue = (rxml.OValue.Contains("01/01/0001") ? "" :
(rxml.OValue == "0x00" ? "" : rxml.OValue));
            rxml.OValue = rxml.OValue.Replace( "[\0]", "");
            if ((rxml.DName.Contains("Date") ||
rxml.DName.Contains("CreatedOn")) && rxml.OValue.Length > 10)
            { rxml.OValue = rxml.OValue.Substring(0,10);} else{ rxml.OValue =
rxml.OValue;}

            if (rxml.OValue == null) { rxml.OValue = " "; }
            lst.Add(new EteAuditTrailXml
            {
                Id = row.Id,
                CreatedOn = row.CreatedOn,
                UserId = row.UserId,
                Objectld = row.Objectld,
                Category = row.Category,
                AuditDate = row.AuditDate,
                DName = rxml.DName,
                OValue = (rxml.OValue.Contains("01/01/0001") ? "" :
(rxml.OValue=="\0"?"":rxml.OValue)),
                NValue = rxml.NValue,
                UserName = UserName,
                ObjectName = Name + " Id:" + _ref,// row.Objectld.ToString(),
            }
        );
    }
}
}
}

```

```

    }
    }
}
catch (Exception exc)
{
    lst = null;
}
return lst;
}
}

public List<DataXml> GEtListfromXml(string xmldata)
{
    List<DataXml> lst = new List<DataXml>();
    try
    {
        Type ObjectType = lst.GetType();
        xmldata = xmldata.Replace("\x00", "");
        lst = ObjectToXML(xmldata, ObjectType);

    }
    catch (Exception exc)
    {
        lst = null;
    }
    return lst;
}

public List<ClassXml> GetClassfromXml(string xmldata)
{
    List<ClassXml> lst = new List<ClassXml>();
    Type ObjectType = lst.GetType();

    lst = ClassToXML(xmldata, ObjectType);
    return lst;
}

public class DataXml
{
    public string DName { get; set; }
    public string OValue { get; set; }
    public string NValue { get; set; }
}

public class ClassXml
{
    public string DName { get; set; }
    public string DValue { get; set; }
}

public class EteAuditTrailXml
{
    public int Id { get; set; }
    public System.DateTime CreatedOn { get; set; }
    public System.Nullable<int> UserId { get; set; }
    public System.Nullable<int> ObjectId { get; set; }
}

```



```

    public string Category { get; set; }
    public string UserName { get; set; }
    public System.Nullable<System.DateTime> AuditDate { get; set; }
    public string DName { get; set; }
    public string OValue { get; set; }
    public string NValue { get; set; }
    public string ObjectName { get; set; }
}
public static XmlDocument stringtoXML(string StringXML)
{
    XmlDocument XmlDoc = new XmlDocument();
    XmlDoc.LoadXml(StringXML);
    return XmlDoc;
}
private static XmlElement GetXMLElement(string xml)
{
    XmlDocument doc = new XmlDocument();
    doc.LoadXml(xml);
    return doc.DocumentElement;
}
private static System.Xml.Linq.XElement GetXElement(string xml)
{
    System.Xml.Linq.XElement doc = new System.Xml.Linq.XElement(xml);
    // doc.LoadXml(xml);
    return doc;
}
public static List<qadata> ObjectToXMLQA(string XmlOfAnObject, Type ObjectType)
{
    StringReader StrReader = new StringReader(XmlOfAnObject);
    XmlSerializer serializer = new XmlSerializer(ObjectType);
    XmlTextReader XmlReader = new XmlTextReader(StrReader);
    List<qadata> AnObject = (List<qadata>)serializer.Deserialize(XmlReader);
    XmlReader.Close();
    StrReader.Close();
    return AnObject;
}
}
}

```

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Xml;
using DowBEntity;
using DowDAL;
using System.Security.Cryptography;
using System.IO;
//using Microsoft.Practices.EnterpriseLibrary.Validation;
//using Microsoft.Practices.EnterpriseLibrary.Validation.Validators;

```

```

namespace DowBAL
{
    public class BalCommon
    {
        public static bool IsValidEmail(string email)
        {
            try
            {
                var mail = new System.Net.Mail.MailAddress(email);
                return true;
            }
            catch
            {
                return false;
            }
        }

        public static XmlDocument stringtoXML(string StringXML)
        {
            XmlDocument XmlDoc = new XmlDocument();
            XmlDoc.LoadXml(StringXML);
            return XmlDoc;
        }

        public string encrypt(string encryptString)
        {
            string EncryptionKey = "0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ";
            byte[] clearBytes = Encoding.Unicode.GetBytes(encryptString);
            using (Aes encryptor = Aes.Create())
            {
                Rfc2898DeriveBytes pdb = new Rfc2898DeriveBytes(EncryptionKey, new byte[] {
                    0x49, 0x76, 0x61, 0x6e, 0x20, 0x4d, 0x65, 0x64, 0x76, 0x65, 0x64, 0x65, 0x76
                });
                encryptor.Key = pdb.GetBytes(32);
                encryptor.IV = pdb.GetBytes(16);
                using (MemoryStream ms = new MemoryStream())
                {
                    using (CryptoStream cs = new CryptoStream(ms, encryptor.CreateEncryptor(),
                        CryptoStreamMode.Write))
                    {
                        cs.Write(clearBytes, 0, clearBytes.Length);
                        cs.Close();
                    }
                    encryptString = Convert.ToBase64String(ms.ToArray());
                }
            }
            return encryptString;
        }

        public string Decrypt(string cipherText)
        {
            string EncryptionKey = "0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ";
            //cipherText = cipherText.Replace(" ", "+");
            byte[] cipherBytes = Convert.FromBase64String(cipherText);

```

```

using (Aes encryptor = Aes.Create())
{
    Rfc2898DeriveBytes pdb = new Rfc2898DeriveBytes(EncryptionKey, new byte[] {
        0x49, 0x76, 0x61, 0x6e, 0x20, 0x4d, 0x65, 0x64, 0x76, 0x65, 0x64, 0x65, 0x76
    });
    encryptor.Key = pdb.GetBytes(32);
    encryptor.IV = pdb.GetBytes(16);
    using (MemoryStream ms = new MemoryStream())
    {
        using (CryptoStream cs = new CryptoStream(ms, encryptor.CreateDecryptor(),
            CryptoStreamMode.Write))
        {
            cs.Write(cipherBytes, 0, cipherBytes.Length);
            cs.Close();
        }
        cipherText = Encoding.Unicode.GetString(ms.ToArray());
    }
}
return cipherText;
}

public static string Encode(string password)
{
    try
    {
        byte[] encData_byte = new byte[password.Length];
        encData_byte = System.Text.Encoding.UTF8.GetBytes(password);
        string encodedData = Convert.ToBase64String(encData_byte);
        return encodedData;
    }
    catch (Exception ex)
    {
        throw new Exception("Error in base64Encode" + ex.Message);
    }
}

public string EncryptPwd(string pwd)
{
    string opwd = "";
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        opwd = db.EncryptData(pwd);
    }

    return opwd;
}

public string Decode(string encodedData)
{
    System.Text.UTF8Encoding encoder = new System.Text.UTF8Encoding();
    System.Text.Decoder utf8Decode = encoder.GetDecoder();
    byte[] todecode_byte = Convert.FromBase64String(encodedData);
    int charCount = utf8Decode.GetCharCount(todecode_byte, 0,
    todecode_byte.Length);
    char[] decoded_char = new char[charCount];
    utf8Decode.GetChars(todecode_byte, 0, todecode_byte.Length, decoded_char, 0);
}

```

```

        string result = new String(decoded_char);
        return result;
    }
    //public class RequiredStringValidatorAttribute : ValueValidatorAttribute
    //{
    //    protected override Validator DoCreateValidator(Type targetType)
    //    {
    //        return new RequiredStringValidator(this.MessageTemplate, this.Negated);
    //    }
    //}
    //public class RequiredStringValidator : ValueValidator<string>
    //{
    //    public RequiredStringValidator(string messageTemplate, bool negated)
    //        : base(messageTemplate, null, negated)
    //    {
    //    }

    //    protected override void DoValidate(string objectToValidate, object currentTarget,
    string key, ValidationResults validationResults)
    //    {
    //        if (string.IsNullOrEmpty(objectToValidate) != Negated)
    //        {
    //            LogValidationResult(validationResults, GetMessage(objectToValidate, key),
    currentTarget, key);
    //        }
    //    }

    //    protected override string DefaultNegatedMessageTemplate
    //    {
    //        get
    //        {
    //            return "Field cannot have a value.";
    //        }
    //    }

    //    protected override string DefaultNonNegatedMessageTemplate
    //    {
    //        get
    //        {
    //            return "Field is required.";
    //        }
    //    }
    //}
}
}

```

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using DowBEntity;
using DowDAL;
using System.Data;

```

```

namespace DowBAL
{
    public class BALCreateUser
    {
        DALUser UserDb = null;

        public BALCreateUser()
        {
            UserDb = new DALUser();
        }

        public string CreateUser(CreateUser crtUser, bool status, int Role, int GrouptId, int
        UserId, int loginId)
        {
            return UserDb.CreateUser(crtUser, status, Role, GrouptId, UserId, loginId);
        }
        public bool DisableUser(int UserId, bool status)
        {
            return UserDb.DisableUser(UserId, status);
        }
        public bool UpdateUser(CreateUser crtUser, string pwd, int GrouptId, int UserId, int
        loginId)
        {
            return UserDb.UpdateUser(crtUser, pwd, GrouptId, UserId, loginId);
        }
        public DataSet GetUserInfo(string col, string exp, int Pgno, int psize)
        {
            DALUser Dusr = new DALUser();
            return Dusr.GetUserInfo(col, exp, Pgno, psize);
        }

        public DataSet GetUserList(string username)
        {
            DALUser Dusr = new DALUser();
            return Dusr.GetUserList(username);
        }

        public DataTable LoginGuest(string uname, string pwd, string IPAddress, string
        MachineName, string City, string Country)
        {
            //pwd = new BalCommon().encrypt(pwd);
            return UserDb.LoginGuest(uname, pwd, MachineName, IPAddress, City, Country);
        }

        public DataTable Login(string uname, string pwd, string IPAddress, string
        MachineName, string City, string Country)
        {
            //pwd = new BalCommon().encrypt(pwd);
            return UserDb.Login(uname, pwd, MachineName, IPAddress, City, Country);
        }
        public DataTable LoginDet(string uname, string pwd, string IPAddress, string
        MachineName, string City, string Country)
        {

```

```

        //pwd = new BalCommon().encrypt(pwd);
        return UserDb.Login(uname, pwd, MachineName, IPAddress, City, Country);
    }
    public bool LogOut(int UserId, int LoginId)
    {
        return UserDb.LogOut(UserId, LoginId);
    }
    public bool RemoveUser(int UserId, bool status)
    {
        return UserDb.RemoveUser(UserId, status);
    }
    public DataSet GetSourceIp(string col, string exp, int Pgno, int psize)
    {
        DALUser Dusr = new DALUser();
        return Dusr.GetSourceIp(col, exp, Pgno, psize);
    }
    public DataSet GetAllPersets()
    {
        return UserDb.GetAllPermissionsets();
    }
    public DataSet GetPersets_Grp(int GrpId)
    {
        return UserDb.GetPersets_Grp(GrpId);
    }
    public CreateUser GetUser(int UserId)
    {
        return UserDb.GetUser(UserId);
    }
}
}

```

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using DowDAL;
using System.Data;

```

```

namespace DowBAL

```

```

{
    public class BalMenus
    {
        DalMenus clstreev = null;

        public BalMenus()
        {
            clstreev = new DalMenus();// ClsTreeView();
        }
        public DataTable SSP_GET_HIERARCHY()
        {
            return clstreev.SSP_GET_HIERARCHY();
        }
    }
}

```

```

    }

    public DataTable GetMenus(int GroupId)
    {
        return clstreev.GetMenus(GroupId);
    }
    public DataTable GetNewMenus(int GroupId)
    {
        return clstreev.GetNewMenus(GroupId);
    }

    public DataTable Permission_Group(int GroupId, int type)
    {
        return clstreev.Permission_Group(GroupId,type);
    }
    public DataTable Products_Group(int GroupId, int type)
    {
        return clstreev.Products_Group(GroupId, type);
    }
    public DataSet BindDynamic_DDL(int lbtnid, int loginid, int lbtntype)
    {
        return clstreev.BindDynamic_DDL(lbtnid, loginid, lbtntype);
    }

    public DataSet GetPageButtonDetails(int pagebtnid)
    {
        return clstreev.GetPageButtonDetails(pagebtnid);
    }
    public DataSet GetLeftButtonsPermision(int pageid, int groupid)
    {
        return clstreev.GetLeftButtonsPermision(pageid, groupid);
    }
    public DataSet GetLeftButtonsProducts(int pageid, int groupid)
    {
        return clstreev.GetLeftButtonsPermision(pageid, groupid);
    }
    public DataTable GetLeftButtons(int pageid,int groupid )
    {
        return clstreev.GetLeftButtons(pageid, groupid);
    }
    public DataTable GetAllDeviceByParentId(int ParentId)
    {
        return clstreev.GetAllDeviceByParentId(ParentId);
    }
    public DataSet GetAllDeviceByParentIdSorting(string col, string exp, int Pgno, int psize,
int ParentId)
    {
        return clstreev.GetAllDeviceByParentIdSorting(col, exp, Pgno, psize, ParentId);
    }
}
}

****

using System;
using System.Collections.Generic;

```

```

using System.Linq;
using System.Text;
using System.Threading.Tasks;
using DowBEntity;
using DowDAL;
namespace DowBAL
{

    public class BalRecentPlacess
    {
        public static bool recentplacesExists(int Id)
        {
            bool exists = false;

            using (var db = new DowPortalDataContext(Configuration.ConnectionString))
            {

                db.Connection.Open();

                var _cust = (from c in db.RecentPlaces where c.rp_id == Id select
c).SingleOrDefault();

                if (_cust != null)
                {
                    exists = true;
                }

            }

            return exists;
        }
        public bool Insert(EteRecentPlaces data)
        {
            bool result = false;
            EteRecentPlaces _user = new EteRecentPlaces();
            using (var db = new DowPortalDataContext(Configuration.ConnectionString))
            {
                db.Connection.Open();
                db.Transaction = db.Connection.BeginTransaction();

                try
                {
                    RecentPlace _cs = new RecentPlace
                    {
                        rp_id = data.rp_id,
                        rp_type = data.rp_type,
                        rp_heading = data.rp_heading,
                        rp_url = data.rp_url,
                        rp_date = data.rp_date,
                        rp_user = data.rp_user,
                    };

                    db.RecentPlaces.InsertOnSubmit(_cs);
                }
            }
        }
    }
}

```



```

        db.SubmitChanges();
        db.Transaction.Commit();

        result = true;
    }
    catch (Exception exc)
    {
        db.Transaction.Rollback();
        throw exc;
    }
}

return result;
}

public bool StoreRecentPlaces(EteRecentPlaces data)
{
    bool result = false;

    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {
            var _user = db.RecentPlaces.Where(e => e.rp_type == data.rp_type &
e.rp_heading == data.rp_heading & e.rp_url == data.rp_url).SingleOrDefault();

            if (_user == null)
            {
                RecentPlace _cs = new RecentPlace
                {
                    rp_id = data.rp_id,
                    rp_type = data.rp_type,
                    rp_heading = data.rp_heading,
                    rp_url = data.rp_url,
                    rp_date = data.rp_date,
                    rp_user = data.rp_user,

                };
                db.RecentPlaces.InsertOnSubmit(_cs);
            }
            else
            {
                // _user.rp_id = data.rp_id;
                _user.rp_type = data.rp_type;
                _user.rp_heading = data.rp_heading;
                _user.rp_url = data.rp_url;
                _user.rp_date = data.rp_date;
                _user.rp_user = data.rp_user;
            }
        }
        db.SubmitChanges();
        db.Transaction.Commit();
    }
}

```

```

        result = true;
    }
    catch (Exception exc)
    {
        db.Transaction.Rollback();

        throw exc;
    }
}

return result;
}
public EteRecentPlaces getRecentPlaces(string type, string heading, string url)
{
    EteRecentPlaces result = new EteRecentPlaces();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var users = (from c in db.RecentPlaces where c rp_type == type & c rp_heading
== heading & c rp_url == url select c).ToList();
        foreach (var data in users)
        {
            result = new EteRecentPlaces
            {
                rp_id = data rp_id,
                rp_type = data rp_type,
                rp_heading = data rp_heading,
                rp_url = data rp_url,
                rp_date = data rp_date,
                rp_user = data rp_user,

            };
            break;
        }
    }

    return result;
}
public List<EteRecentPlaces> GetRecentPlacess(int userid)
{
    List<EteRecentPlaces> _user = new List<EteRecentPlaces>();

    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        db.Connection.Open();
        var users = (from c in db.RecentPlaces where c rp_user == userid select
c).OrderByDescending(c => c rp_type).ThenByDescending(c => c rp_date).ToList();
        foreach (var data in users)
            _user.Add(new EteRecentPlaces
            {
                rp_id = data rp_id,
                rp_type = data rp_type,
                rp_heading = data rp_heading,

```

```

        rp_url = data.rp_url,
        rp_date = data.rp_date,
        rp_user = data.rp_user,
    });
}

return _user;
}
public bool DeleteRecentPlaces(int userid)
{
    bool result = false;

    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        try
        {
            var _ord = db.RecentPlaces.Where(e => e.rp_user == userid);

            if (_ord != null)
            {
                db.Transaction = db.Connection.BeginTransaction();
                //int res = db.DeleteCompleteDraft(userid);
                int res = db.DeleteRecentPlaces(userid);
                db.Transaction.Commit();
            }
        }

        catch (Exception exc)
        {
            db.Transaction.Rollback();
            throw exc;
        }
    }

    return result;
}
}
}

```

```

using System;
using System.IO;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using DowBEntity;
using DowDAL;
using System.Xml;
using System.Xml.Serialization;
using System.Xml.XmlConfiguration;
namespace DowBAL
{

```

```

public class BalVatCodes
{

    public bool Insert(EteVatCode data)
    {
        bool result = false;
        EteVatCode _user = new EteVatCode();
        using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        {
            db.Connection.Open();
            db.Transaction = db.Connection.BeginTransaction();

            try
            {
                VatCode _cs = new VatCode
                {
                    Id = data.Id,
                    Code = data.Code,
                    Rate = data.Rate,
                    Description = data.Description,
                    IsDefault = data.IsDefault,
                    IsActive = data.IsActive,
                    CreatedOn = data.CreatedOn,
                };
                db.VatCodes.InsertOnSubmit(_cs);

                db.SubmitChanges();
                db.Transaction.Commit();

                result = true;
            }
            catch (Exception exc)
            {
                db.Transaction.Rollback();
                throw exc;
            }
        }

        return result;
    }

    public bool Update(EteVatCode data)
    {
        bool result = false;

        EteVatCode odata = new EteVatCode();

        using (var db = new DowPortalDataContext(Configuration.ConnectionString))
        {
            db.Connection.Open();
            db.Transaction = db.Connection.BeginTransaction();

```

```

try
{
    var _user = db.VatCodes.Where(e => e.Id == data.Id).SingleOrDefault();

    if (_user == null)
    {
        throw (new ApplicationException("Unable to find the User record"));
    }
    _user.Id = data.Id;
    _user.Code = data.Code;
    _user.Rate = data.Rate;
    _user.Description = data.Description;
    _user.IsDefault = data.IsDefault;
    _user.IsActive = data.IsActive;
    _user.CreatedOn = data.CreatedOn;

    db.SubmitChanges();
    db.Transaction.Commit();
    result = true;
}
catch (Exception exc)
{
    db.Transaction.Rollback();

    throw exc;
}

}

return result;
}
public EteVatCode getVatCodeById(int Id)
{
    EteVatCode result = new EteVatCode();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var users = (from c in db.VatCodes where c.Id == Id select c).ToList();
        foreach (var data in users)
        {
            result = new EteVatCode
            {
                Id = data.Id,
                Code = data.Code,
                Rate = data.Rate,
                Description = data.Description,
                IsDefault = data.IsDefault,
                IsActive = data.IsActive,
                CreatedOn = data.CreatedOn,
            }
        }
    }
}

```

```

        };
        break;
    }
}

return result;
}

public List<EteVatCode> GetVatCodes()
{
    List<EteVatCode> _user = new List<EteVatCode>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        db.Connection.Open();

        var users = (from c in db.VatCodes select c).ToList();
        foreach (var data in users)
        {
            _user.Add(new EteVatCode
            {
                Id = data.Id,
                Code = data.Code,
                Rate = data.Rate,
                Description = data.Description,
                IsDefault = data.IsDefault,
                IsActive = data.IsActive,
                CreatedOn = data.CreatedOn,
            });
        }

        return _user;
    }
}

public String GetVatCodeById(int Id)
{
    String name = "";
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        db.Connection.Open();

        var _cust = (from c in db.VatCodes where c.Id == Id select c).SingleOrDefault();

        if (_cust != null)
        {
            name = _cust.Description;
        }
        else
        {
            name = "";
        }
    }
}

```

```

        return name;
    }
}

}

****

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using DowBEntity;
using DowDAL;

namespace DowBAL
{
    public class Branding
    {
        public EteBrand getbrand(String brandName)
        {
            EteBrand result = new EteBrand();
            using (var db = new DowPortalDataContext(Configuration.ConnectionString))
            {
                db.Connection.Open();
                var users = (from c in db.Brands where c.BrandName == brandName select
c).ToList();
                foreach (var row in users)
                {
                    result = new EteBrand
                    {
                        Id = row.Id,
                        ShortCode = row.ShortCode,
                        BrandName = row.BrandName,
                        OpalCredentialId = row.OpalCredentialId,
                        DefaultInvoiceMask = row.DefaultInvoiceMask,
                        EmailInvoiceMask = row.EmailInvoiceMask,
                        BusinessInvoiceMask = row.BusinessInvoiceMask,
                        BusinessEmailInvoiceMask = row.BusinessEmailInvoiceMask,
                        FromEmailAddress = row.FromEmailAddress,
                        ForwardLinksRoot = row.ForwardLinksRoot,
                        FilePath = row.FilePath,
                        CreatedOn = row.CreatedOn
                    };
                    break;
                }
            }

            return result;
        }
        public EteBrand getbrand(int id)
        {

```

```

EteBrand result = new EteBrand();
using (var db = new DowPortalDataContext(Configuration.ConnectionString))
{
    db.Connection.Open();
    var users = (from c in db.Brands where c.Id == id select c).ToList();
    foreach (var row in users)
    {
        result = new EteBrand
        {
            Id = row.Id,
            ShortCode = row.ShortCode,
            BrandName = row.BrandName,
            OpalCredentialId = row.OpalCredentialId,
            DefaultInvoiceMask = row.DefaultInvoiceMask,
            EmailInvoiceMask = row.EmailInvoiceMask,
            BusinessInvoiceMask = row.BusinessInvoiceMask,
            BusinessEmailInvoiceMask = row.BusinessEmailInvoiceMask,
            FromEmailAddress = row.FromEmailAddress,
            ForwardLinksRoot = row.ForwardLinksRoot,
            FilePath = row.FilePath,
            CreatedOn = row.CreatedOn

        };
        break;
    }
}

return result;
}
public string getbrandName(int id)
{
    string result = "";
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var users = (from c in db.Brands where c.Id == id select c).ToList();
        foreach (var row in users)
        {

            result = row.BrandName;

            break;
        }
    }

    return result;
}
public List<EteBrand> getbrands()
{
    List<EteBrand> _user = new List<EteBrand>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

```



```

db.Connection.Open();

var users = (from c in db.Brands select c).ToList();
foreach (var row in users)
{
    _user.Add(new EteBrand
    {
        Id = row.Id,
        ShortCode = row.ShortCode,
        BrandName = row.BrandName,
        OpalCredentialId = row.OpalCredentialId,
        DefaultInvoiceMask = row.DefaultInvoiceMask,
        EmailInvoiceMask = row.EmailInvoiceMask,
        BusinessInvoiceMask = row.BusinessInvoiceMask,
        BusinessEmailInvoiceMask = row.BusinessEmailInvoiceMask,
        FromEmailAddress = row.FromEmailAddress,
        ForwardLinksRoot = row.ForwardLinksRoot,
        FilePath = row.FilePath,
        CreatedOn = row.CreatedOn

    });
}

return _user;
}
}
//public List<EteDealerBrand> getdealerbrand(int dealerid)
//{
// List<EteDealerBrand> _user = new List<EteDealerBrand>();
// using (var db = new DowPortalDataContext(Configuration.ConnectionString))
// {

//     db.Connection.Open();

//     //if (RoleID == 1)
//     //{
//         var Brandings = (from c in db.v_dealerbrands where c.DealerId == dealerid
select c).ToList();
//     //}
//     //else if (RoleID == 3 ||RoleID == 3 )
//     //{
//         var users = (from c in db.v_dealerbrands where c.DealerId == dealerid select
c).ToList();
//     //}else
//     //{
//         var users = (from c in db.v_dealerbrands where c.DealerId == dealerid select
c).ToList();
//     //}

//     var users = (from c in db.v_dealerbrands where c.DealerId == dealerid select
c).ToList();
//     foreach (var row in users)
//     {

```

```

//     _user.Add(new EteDealerBrand
//     {
//         DealerId = row.DealerId,
//         BrandId = row.BrandId,
//         ShortCode = row.ShortCode,
//         BrandName = row.BrandName,
//         FilePath = row.FilePath
//     });
// }

//     return _user;
// }
//}
public bool DeleteDealerBrand(int dealerid)
{
    bool result = false;
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();
        var users = (from c in db.Brands where c.Id == dealerid select c).ToList();
        db.Brands.DeleteAllOnSubmit(users);
        db.Transaction.Commit();
        result = true;
        return result;
    }
}
public bool Update(EteDealerBrand data)
{
    bool result = false;

    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {
            var _user = db.DealerBrands.Where(e => e.DealerId == data.DealerId).ToList();

            if (_user == null)
            {
                throw (new ApplicationException("Unable to find the User record"));
            }
            db.DealerBrands.DeleteAllOnSubmit(_user);
            db.Transaction.Commit();

            db.Transaction = db.Connection.BeginTransaction();

            DealerBrand _cs = new DealerBrand
            {
                DealerId = data.DealerId,
                BrandId = data.BrandId,
                CreatedTime = DateTime.Now
            }
        }
    }
}

```

```

};
db.DealerBrands.InsertOnSubmit(_cs);

db.SubmitChanges();
db.Transaction.Commit();
result = true;
}
catch (Exception exc)
{
    db.Transaction.Rollback();

    throw exc;
}
}

return result;
}
public bool InsertDealerBrand(List<EteDealerBrand> Lst)
{
    bool result = false;
    DealerBrand _user = new DealerBrand();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        if (Lst.Count > 0)
        {
            db.Connection.Open();
            db.Transaction = db.Connection.BeginTransaction();
            var _usr = db.DealerBrands.Where(e => e.DealerId == Lst[0].DealerId).ToList();
            try
            {
                if (_usr.Count > 0 && _usr != null)
                {
                    db.DealerBrands.DeleteAllOnSubmit(_usr);
                    db.Transaction.Commit();
                    db.Transaction = db.Connection.BeginTransaction();
                }
                foreach (var row in Lst)
                {
                    DealerBrand _cs = new DealerBrand
                    {
                        DealerId = row.DealerId,
                        BrandId = row.BrandId,
                        CreatedTime = DateTime.Now
                    };

                    db.DealerBrands.InsertOnSubmit(_cs);
                }
                db.SubmitChanges();
                db.Transaction.Commit();
            }
        }
    }
}

```

```

        result = true;
    }

    catch (Exception exc)
    {
        db.Transaction.Rollback();
        throw exc;
    }
}

return result;
}
}
}
}
}
}
}

****

```

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using DowBEntity;
using DowDAL;
using System.Data;
using System.Xml;
namespace DowBAL
{
    public class ClsPermissioGroup
    {
        DALUser UserDb = null;
        DalPermissionGroups pg = null;
        public ClsPermissioGroup()
        {
            UserDb = new DALUser();
            pg = new DalPermissionGroups();
        }
        public DataTable GetAllPermissionGroups()
        {
            return UserDb.GetAllPermissionGroups();
        }
        public UserPermissions GetPersets_Page(string UserName, string PageUrl)
        {
            DalPermissionGroups Uper = new DalPermissionGroups();
            return Uper.GetPersets_Page(UserName, PageUrl);
        }
        public List<UserPermissions> GetPersets_usr(string UserName)
        {
            DalPermissionGroups Uper = new DalPermissionGroups();
            return Uper.GetPersets_usr(UserName);
        }
        public UserPermissions GetUserLandingPage(string UserName)
        {

```

```

        DalPermissionGroups Uper = new DalPermissionGroups();
        return Uper.GetUserLandingPage(UserName);
    }
    public DataTable GetPerSetsForGroup()
    {
        return pg.GetPerSetsForGroup();
    }
    public DataTable PopulateDynamicTree(int GrpId)
    {
        return pg.PopulateDynamicTree(GrpId);
    }
    public string CreatePermissionGrp(XmlDocument xdoc, string GrpName,int UserId,int
LoginId)
    {
        return pg.CreatePermissionGrp(xdoc, GrpName,UserId,LoginId);
    }
    public bool UpdatePermissionGrp(XmlDocument xdoc, int GrpId, int UserId, int LoginId)
    {
        return pg.UpdatePermissionGrp(xdoc, GrpId,UserId,LoginId);
    }
}
}

```

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Xml;
using System.Xml.Serialization;
using System.Xml.XmlConfiguration;
using System.IO;
using DowBEntity;
using DowBAL;
using DowDAL;
using System.Data;

```

```

using System.Collections;

```

```

namespace DowBAL
{

```

```

    public class Common
    {
        public static int GetDupRecCount(string TableName, string IdColName, Hashtable
colvalues, string ColName)
        {
            return DowDAL.Common.GetDupRecCount(TableName, IdColName, colvalues,
ColName);
        }
        public static DataSet GetDDIData(string TableName, string ValueColumn, string
TextColumn, Hashtable colvalues)
        {

```

```

        return DowDAL.Common.GetDDIDData(TableName, ValueColumn, TextColumn,
colvalues);
    }

```

```

public static string XMLFromObject(object o)
{
    string str;
    XmlSerializer serializer = new XmlSerializer(o.GetType());
    StringWriter sw = new StringWriter();
    XmlTextWriter tw = new XmlTextWriter(sw);
    serializer.Serialize(tw, o);
    str = sw.ToString();
    sw.Close();
    tw.Close();
    return str;
}

```

```

public static List<DataXml> ObjectToXML(string XmlOfAnObject, Type ObjectType)
{
    StringReader StrReader = new StringReader(XmlOfAnObject);
    XmlSerializer serializer = new XmlSerializer(ObjectType);
    XmlTextReader XmlReader = new XmlTextReader(StrReader);
    List<DataXml> AnObject = (List<DataXml>)serializer.Deserialize(XmlReader);
    XmlReader.Close();
    StrReader.Close();
    return AnObject;
}

```

```

}
public List<BalAuditTrail> GetAuditTrail(int PageId)
{
    List<BalAuditTrail> lst = new List<BalAuditTrail>();
    List<DataXml> dlst = new List<DataXml>();
    string UserName = "";
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();

        var _cust = (from c in db.AuditTrails where c.ObjectId == PageId select c).ToList();
        foreach (var row in _cust)
        {
            dlst = (new Common().GETListfromXml(row.Message));
            UserName = new tblUsers().getusername(Convert.ToInt32(row.UserId));
            foreach (var rxml in dlst)
            {
                lst.Add(new BalAuditTrail
                {
                    Id = row.Id,
                    CreatedOn = row.CreatedOn,
                    UserId = row.UserId,
                    ObjectId = row.ObjectId,
                    Category = row.Category,
                    AuditDate = row.AuditDate,
                    DName = rxml.DName,
                    OValue = rxml.OValue,
                    NValue = rxml.NValue,
                });
            }
        }
    }
}

```

```
        UserName = UserName,
    }
    );
}
}

}

return lst;
}
public List<Common.DataXml> GEtListfromXml(string xmldata)
{
    Object o;
    List<Common.DataXml> lst = new List<Common.DataXml>();
    Type ObjectType = lst.GetType();
    o = Common.ObjectToXML(xmldata, ObjectType);
    lst = Common.ObjectToXML(xmldata, ObjectType);
    return lst;
}
public class DataXml
{
    public string DName { get; set; }
    public string OValue { get; set; }
    public string NValue { get; set; }
}
public class BalAuditTrail
{
    public int Id { get; set; }

    public System.DateTime CreatedOn { get; set; }

    public System.Nullable<int> UserId { get; set; }

    public System.Nullable<int> ObjectId { get; set; }

    public string Category { get; set; }
    public string UserName { get; set; }
    public System.Nullable<System.DateTime> AuditDate { get; set; }
    public string DName { get; set; }
    public string OValue { get; set; }
    public string NValue { get; set; }
}
}
}
}
}
}

****
```

```
using System.Collections.Generic;
using System;
using System.Threading.Tasks;
using System.Diagnostics;
using System.Data;
```

```
using System.Xml.Linq;
using Microsoft.VisualBasic;
using System.Collections;
using System.Linq;
using System.Configuration;
using System.Text;
```

```
namespace DowBAL
```

```
{
    sealed public class Configuration
    {
        private Configuration()
        {
        }
        internal static string ConnectionString
        {
            get
            {
                return ConfigurationSettings.AppSettings["DowConectingString"].ToString();
                //string constring =
                ConfigurationSettings.AppSettings["DowConectingString"].ToString();
                //return constring;
            }
        }
    }
}
```

```
using Microsoft.VisualBasic;
using System;
using System.Collections;
using System.Collections.Generic;
using System.Configuration;
using System.Data;
using System.Diagnostics;
using System.IO;
using System.Web;
```

```
namespace DowBAL
```

```
{
    public class DDLogTracer
    {
        //public enum LogLevel
        //{
        //    INFO = 0,
        //    DEBUG = 1,
        //    DB = 2,
        //    EXC = 3,
        //    CRIT = 4
        //}
```



```

public string TraceThreadId;
private static string logDirectory;
private static StreamWriter sw;
//Private Sub ssNew()
// Try
//   logDirectory = (ConfigurationManager.AppSettings("strlogpath").ToString +
"logconfig.err")
//   If Not File.Exists(logDirectory) Then
//       Dim fs As FileStream = File.Create(logDirectory)
//       fs.Close()
//   End If
//   sw = File.AppendText(logDirectory)
//   Catch exc As Exception

// End Try
//End Sub

private string GetLotType(int TL_Type)
{
    string TR_Num = null;

    if ((TL_Type == 0))
    {
        TR_Num = "INFO";
    }
    else if ((TL_Type == 1))
    {
        TR_Num = "DEBUG";
    }
    else if ((TL_Type == 2))
    {
        TR_Num = "DATABASE";
    }
    else if ((TL_Type == 3))
    {
        TR_Num = "DETAILED";
    }
    else if ((TL_Type == 4))
    {
        TR_Num = "CRITICAL";
    }
    else if ((TL_Type == 5))
    {
        TR_Num = "EXCEPTION";
    }
    else
    {
        TR_Num = "ERROR";
    }
    return TR_Num;
}

public void Trace_Log(int TL_Type, string ModuleName, string Page, string Block,
string Log,string path="")
{
    if ((ConfigurationSettings.AppSettings["level" + TL_Type.ToString()] == "1"))

```

```

    {
        string Log_Write_String = string.Format("{0} - [{1}]:[{2}]:[{3}]:[{4}]:{5}",
DateTime.Now.ToString("yyyy-MM-dd HH:mm:ss.ffffff"), GetLotType(TL_Type),
ModuleName, Page, Block, Log);
        WriteLogFile(Log_Write_String, path);
    }
}
public void Trace_Error(int TL_Type, string ModuleName, string Page, string Block,
string ExceptionMessage,string path="")
{
    if ((ConfigurationSettings.AppSettings["level5"] == "1"))
    {
        string Error_Write_String = string.Format("{0} - [{1}]:[{2}]:[{3}]:[{4}]:{5}",
DateTime.Now.ToString("yyyy-MM-dd HH:mm:ss.ffffff"), GetLotType(TL_Type),
ModuleName, Page, Block, ExceptionMessage);
        return;
        WriteErrorFile(Error_Write_String, path);
    }
}

private void WriteLogFile(string Log_Content,string path)
{
    string Logpath = ConfigurationSettings.AppSettings["strlogpath"].ToString();
    string filepath = ""; // (ConfigurationSettings.AppSettings["strlogpath"].ToString() +
(DateTime.Now.ToString("yyyy\\\\"MM\\\\"dd") + "\\");

    if (Logpath == "" | Logpath.Length <= 0)
    {
        filepath =path+ ("\\Logs\\" + (DateTime.Now.ToString("yyyy\\\\"MM\\\\"dd") +
"\\"));
    }
    else
    {
        filepath = (Logpath );//+ (DateTime.Now.ToString("yyyy\\\\"MM\\\\"dd") + "\\");
    }

    if (!Directory.Exists(filepath))
    {
        Directory.CreateDirectory(filepath);
    }
    string strLogFileName = (DateTime.Now.Hour.ToString("D2")) + "_WS_" +
TraceThreadId + "." + (ConfigurationSettings.AppSettings["strLogFileName"]);
    strLogFileName = filepath + "\\" + strLogFileName;
    StreamWriter sw = null;

    if ((!File.Exists(strLogFileName)))
    {
        sw = File.CreateText(strLogFileName);
    }
    else
    {
        sw = File.AppendText(strLogFileName);
    }
}

```

```

        sw.WriteLine(Log_Content);
        sw.Close();
    }

    private void WriteErrorFile(string Error_Content,string path)
    {
        string Logpath = ConfigurationSettings.AppSettings["strlogpath"].ToString();
        string filepath = ""; // (ConfigurationSettings.AppSettings["strlogpath"].ToString() +
(DateTime.Now.ToString("yyyy\\\\" + "MM\\\\" + "dd") + "\\\\"));

        if (Logpath == "" | Logpath.Length <= 0)
        {
            filepath = path + ("\\Log\\"); // + (DateTime.Now.ToString("yyyy\\\\" + "MM\\\\" + "dd") +
"\\\\"));
        }
        else
        {
            filepath = (Logpath); // + (DateTime.Now.ToString("yyyy\\\\" + "MM\\\\" + "dd") + "
\\\\"));
        }
        if (!Directory.Exists(filepath))
        {
            Directory.CreateDirectory(filepath);
        }
        string strErrorFileName = (DateTime.Now.Day.ToString("D2")) +
(DateTime.Now.Month.ToString("D2")) + "_WS_" + TraceThreadid + "." +
(ConfigurationSettings.AppSettings["strErrorFileName"]);
        strErrorFileName = filepath + "\\\" + strErrorFileName;
        StreamWriter sw = null;

        if ((!File.Exists(strErrorFileName)))
        {
            sw = File.CreateText(strErrorFileName);
        }
        else
        {
            sw = File.AppendText(strErrorFileName);
        }
        sw.WriteLine(Error_Content);
        sw.Close();
    }
    private long checkfilesizefunction(long filesizetocheck)
    {
        try
        {
            long filesize = filesizetocheck;
            long getfilesizefromconfig =
long.Parse(ConfigurationSettings.AppSettings["filesize"].ToString());
            string getfilesizeextension =
ConfigurationSettings.AppSettings["FileSizeFormat"].ToString();
            if ((getfilesizeextension.ToUpper() == "KB"))
            {
                long chkfilesize = (filesize / 1024);
                return chkfilesize;
            }
        }
    }

```

```

    }
    else if ((getfilesizeextension.ToUpper() == "MB"))
    {
        long chkfilesize = (filesize / (1024 * 1024));
        return chkfilesize;
    }
    else if ((getfilesizeextension.ToUpper() == "GB"))
    {
        long chkfilesize = (filesize / (1024 * (1024 * 1024)));
        return chkfilesize;
    }
    return filesize;
}
catch (Exception ex)
{
    return 0;
}
}
}
}

```

```

using System;
using System.IO;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using DowBEntity;
using DowDAL;
using System.Xml;
using System.Xml.Serialization;
using System.Xml.XmlConfiguration;
namespace DowBAL
{
    public class Dealers
    {
        public int? GetNextDealerId()
        {
            int? result = 0;
            using (var db = new DowPortalDataContext(Configuration.ConnectionString))
            {
                //GetNextDealerIdResult lst = db.GetNextDealerId().SingleOrDefault();
                //result = lst.NextDealerId;
            }

            return result;
        }
        public bool DealerNameExists(string name)
        {
            bool exists = false;

```

```

using (var db = new DowPortalDataContext(Configuration.ConnectionString))
{
    db.Connection.Open();

    var _cust = (from c in db.Dealers where c.CompanyName == name select
c).SingleOrDefault();

    if (_cust != null)
    {
        exists = true;
    }
}

return exists;
}
public bool Insert(EteDealer data)
{
    bool result = false;
    EteDealer _user = new EteDealer();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {
            Dealer _cs = new Dealer
            {
                DealerId = data.DealerId,
                CompanyName = data.CompanyName,
                IsActive = data.IsActive,
                UserNamePrefix = data.UserNamePrefix,
                AutoProcessCommissions = data.AutoProcessCommissions,
                DealerOperationProfileId = data.DealerOperationProfileId,
                CreatedOn = data.CreatedOn,
                UpdatedOn = data.CreatedOn,
                EMailString = data.EMailString,
            };
            db.Dealers.InsertOnSubmit(_cs);

            db.SubmitChanges();
            db.Transaction.Commit();
            var COid = (from a in db.Dealers where a.CompanyName ==
data.CompanyName select a.DealerId).FirstOrDefault();
            string mess;
            EteDealer odata = new EteDealer();
            mess = BalAuditTrail.GetAuditXml("Dealer", odata, data);
            EteAuditTrail at = new EteAuditTrail();
            at = new EteAuditTrail
            {

```

```

        CreatedOn = data.CreatedOn,
        UserId = data.UId,
        ObjectId = Convert.ToInt32(COid), // data.Id,
        Category = "Insert",
        Message = mess,
        AuditDate = DateTime.Today,
        PageId = 166, // data.Pageid,
    };
    bool res = new BalAuditTrail().Insert(at);

    result = true;
}
catch (Exception exc)
{
    db.Transaction.Rollback();
    throw exc;
}
}

return result;
}
public bool Update(EteDealer data)
{
    bool result = false;
    string mess;
    EteDealer odata = new EteDealer();

    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {
            var _user = db.Dealers.Where(e => e.DealerId ==
data.DealerId).SingleOrDefault();

            if (_user == null)
            {
                throw (new ApplicationException("Unable to find the User record"));
            }

            odata = new EteDealer
            {
                DealerId = _user.DealerId,
                CompanyName = _user.CompanyName,
                IsActive = _user.IsActive,
                UserNamePrefix = _user.UserNamePrefix,
                AutoProcessCommissions = _user.AutoProcessCommissions,
                DealerOperationProfileId = _user.DealerOperationProfileId,
                CreatedOn = _user.CreatedOn,
                EMailString = data.EMailString,
            };

```

```

        if (odata.CreatedOn == null) { odata.CreatedOn = DateTime.Today; }
        _user.DealerId = data.DealerId;
        _user.CompanyName = data.CompanyName;
        _user.IsActive = data.IsActive;
        _user.UserNamePrefix = data.UserNamePrefix;
        _user.AutoProcessCommissions = data.AutoProcessCommissions;
        _user.DealerOperationProfileId = data.DealerOperationProfileId;
        _user.EMailString = data.EMailString;
        _user.UpdatedOn = data.UpdatedOn;

        db.SubmitChanges();
        db.Transaction.Commit();
        result = true;
    }
    catch (Exception exc)
    {
        db.Transaction.Rollback();

        throw exc;
    }
    mess = BalAuditTrail.GetAuditXml("Dealer", odata, data);
    EteAuditTrail at = new EteAuditTrail();
    at = new EteAuditTrail
    {
        CreatedOn = odata.CreatedOn,
        UserId = data.UId,
        ObjectId = data.DealerId,
        Category = "Edit",
        Message = mess,
        AuditDate = DateTime.Today,
        PageId = data.PageId,
    };
    bool res = new BalAuditTrail().Insert(at);
}

return result;
}
public EteDealer getdealer(int name)
{
    EteDealer result = new EteDealer();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var users = (from c in db.Dealers where c.DealerId == name select c).ToList();
        foreach (var data in users)
        {
            result = new EteDealer
            {
                DealerId = data.DealerId,
                CompanyName = data.CompanyName,
                IsActive = data.IsActive,
                UserNamePrefix = data.UserNamePrefix,
                AutoProcessCommissions = data.AutoProcessCommissions,
            }
        }
    }
}

```

```

        DealerOperationProfileId = data.DealerOperationProfileId,
        EMailString = data.EMailString,
        CreatedOn = data.CreatedOn
    };
    break;
}
}

return result;
}
public List<EteDealer> GetDealers(string Curl)
{
    List<EteDealer> _user = new List<EteDealer>();
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        db.Connection.Open();
        int pgid = (new tblPagess()).GetPagId(Curl);
        var users = (from c in db.Dealers select c).ToList();
        foreach (var data in users)
        {
            _user.Add(new EteDealer
            {
                DealerId = data.DealerId,
                CompanyName = data.CompanyName,
                IsActive = data.IsActive,
                UserNamePrefix = data.UserNamePrefix,
                AutoProcessCommissions = data.AutoProcessCommissions,
                DealerOperationProfileId = data.DealerOperationProfileId,
                CreatedOn = data.CreatedOn,
                EMailString = data.EMailString,
                PagId = pgid,
            });
        }

        return _user;
    }
}

public String GetDealerName(int Id)
{
    String name = "";
    using (var db = new DowPortalDataContext(Configuration.ConnectionString))
    {

        db.Connection.Open();

        var _cust = (from c in db.Dealers where c.DealerId == Id select
c).SingleOrDefault();

        if (_cust != null)
        {

            name = _cust.CompanyName;
        }
    }
}

```



```

        else
        {
            name = "";
        }
    }

    return name;
}

//public class Common.DataXml
//{
//    public string DName { get; set; }
//    public string OValue { get; set; }
//    public string NValue { get; set; }
//}
//public static string GetXMLFromObject(object o)
//{
//    string str;
//    XmlSerializer serializer = new XmlSerializer(o.GetType());
//    StringWriter sw = new StringWriter();
//    XmlTextWriter tw = new XmlTextWriter(sw);
//    serializer.Serialize(tw, o);
//    str= sw.ToString();
//    sw.Close();
//    tw.Close();
//    return str;
//}
//public static Object ObjectToXML(string XmlOfAnObject, Type ObjectType)
//{
//    StringReader StrReader = new StringReader(XmlOfAnObject);
//    XmlSerializer serializer = new XmlSerializer(ObjectType);
//    XmlTextReader XmlReader = new XmlTextReader(StrReader);
//    Object AnObject = serializer.Deserialize(XmlReader);
//    XmlReader.Close();
//    StrReader.Close();
//    return AnObject;
//}
}
}

****

using System;
using System.Configuration;
using System.IO;
using System.Net;
using System.Net.Mail;

namespace DowBEntity
{
    public class Email
    {
        public String toList{ get; set; } //Separated by ;
    }
}

```

```

public String ccList { get; set; }
public String bccList { get; set; }
public String messageBody { get; set; }
public String attachmentFilePath { get; set; }
public bool isBodyHTML { get; set; }

public String subject { get; set; }

public String fromEmail { get; set; }

public void sendEmail()
{
    MailMessage mail = new MailMessage();
    //--- Commented by Syed >>>>> Smtplib SmtplibServer = new
SmtplibClient("smtp.gmail.com");
    string SmtplibClientServer =
ConfigurationSettings.AppSettings["SmtplibClientServer"].ToString(); //"mail.eze-talk.com";
    SmtplibClient SmtplibServer = new SmtplibClient(SmtplibClientServer);//SmtplibClient
SmtplibServer = new SmtplibClient("mail.eze-talk.com");
    string mailFrom = ConfigurationSettings.AppSettings["mailFrom"].ToString();
//"provisioning@eze-talk.com";
    mail.From = new MailAddress(mailFrom);//mail.From = new
MailAddress("provisioning@eze-talk.com");
    mail.Subject = subject;
    mail.Body = messageBody;
    mail.To.Add(this.toList);
    //mail.CC.Add(this.ccList???"");
    mail.Bcc.Add(this.bccList);
    if (attachmentFilePath != null && attachmentFilePath.Length > 0)
    {
        mail.Attachments.Add(new Attachment(File.OpenWrite(attachmentFilePath),
attachmentFilePath));
    }
    mail.IsBodyHtml = true;

    /*mail.To.Add("santosh.kapse@gmail.com");
    mail.Subject = "Test Mail";
    mail.Body = "This is for testing SMTP mail from GMAIL";*/
    int SmtplibPort
=Convert.ToInt32(ConfigurationSettings.AppSettings["SmtplibPort"].ToString()); // 25;
    SmtplibServer.Port = SmtplibPort;
    string SmtplibServerUser =
ConfigurationSettings.AppSettings["SmtplibServerUser"].ToString(); //"b2bportal\Synverse";
    string SmtplibServerPassword =
ConfigurationSettings.AppSettings["SmtplibServerPassword"].ToString(); //"aK4Hhi4M8W9V";
    SmtplibServer.Credentials = new System.Net.NetworkCredential(SmtplibServerUser,
SmtplibServerPassword); //SmtplibServer.Credentials = new
System.Net.NetworkCredential("b2bportal\Synverse", "aK4Hhi4M8W9V");
    ///("santosh.kapse@gmail.com", "KA11!!se");
    SmtplibServer.EnableSsl = false;

    //SmtplibServer.Send(mail);
    //marked for setting up email by ravi
    /*
    * To resolve the OutStanding Orders Issue.

```

<http://stackoverflow.com/questions/20906077/gmail-error-the-smtp-server-requires-a-secure-connection-or-the-client-was-not>

```
*/  
  
/*var client = new SmtClient("auth.smtp.1and1.co.uk", 25)  
{  
    Credentials = new NetworkCredential("haroon.mohammed@synverse.com",  
"nooman@2012"),  
    EnableSsl = true  
};  
  
MailMessage mm = new MailMessage(fromEmail, toList);  
mm.Subject = subject;  
mm.Body = messageBody;  
if (attachmentFilePath != null && attachmentFilePath.Length > 0)  
{  
    mm.Attachments.Add(new Attachment(File.OpenWrite(attachmentFilePath),  
attachmentFilePath));  
}  
mm.IsBodyHtml = true;  
client.Send(mm);*/  
}  
}
```

This Concludes Business Logic Tier

Chapter 6

First, these are a few files to be added to the app_code folder

****Common.cs*****

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Security;
using DowBAL;
using DowBEntity;
using System.Text;
using System.IO;
using System.Collections;
using System.Web.UI.WebControls;
using System.Data;
using DowDAL;
using System.Data.SqlClient;
using System.Web.UI.HtmlControls;
using System.Security.Principal;
using System.Net;
using System.Security.Cryptography;
//using LogTracer;
/// <summary>
/// Summary description for Common
/// </summary>

public class Common
{
    public Common()
    {
        //
        // TODO: Add constructor logic here
        //
    }
}
```

```
}
```

```
public static string tblMstNetGrp = "Group";  
public static string colMstGrp_Val = "GroupID";  
public static string colMstGrp_txt = "GroupName";
```

```
//public static string strQryNetwork = "select a.NetviewId,ObjAttrValue from  
ObjectNetworkView a,Objectattributes b,ObjectNetworkAttributes c where  
a.ObjectId=b.ObjId and a.NetviewId=c.NetviewId and b.ObjAttrId=c.objAttrId and  
(b.objAttrName)='Name' and ISNULL(ISREMOVED,0)=0 and a.NetviewID <> 1 and  
ObjAttrValue Like '%@$#%' and a.NetviewId not in (select Netviewid from Corousel) order  
by b.objAttrName";
```

```
DDLogTracer objlog = new DDLogTracer();  
public static string EncryptQueryString(string inputText, string key, string salt)  
{  
    byte[] plainText = Encoding.UTF8.GetBytes(inputText);  
  
    using (RijndaelManaged rijndaelCipher = new RijndaelManaged())  
    {  
        PasswordDeriveBytes secretKey = new  
PasswordDeriveBytes(Encoding.ASCII.GetBytes(key), Encoding.ASCII.GetBytes(salt));  
        using (ICryptoTransform encryptor =  
rijndaelCipher.CreateEncryptor(secretKey.GetBytes(32), secretKey.GetBytes(16)))  
        {  
            using (MemoryStream memoryStream = new MemoryStream())  
            {  
                using (CryptoStream cryptoStream = new CryptoStream(memoryStream,  
encryptor, CryptoStreamMode.Write))  
                {  
                    cryptoStream.Write(plainText, 0, plainText.Length);  
                    cryptoStream.FlushFinalBlock();  
                    string base64 = Convert.ToBase64String(memoryStream.ToArray());  
  
                    // Generate a string that won't get screwed up when passed as a query string.
```

```

        string urlEncoded = HttpUtility.UrlEncode(base64);
        return urlEncoded;
    }
}
}
}
}

public static string DecryptQueryString(string inputText, string key, string salt)
{
    byte[] encryptedData = Convert.FromBase64String(inputText);

    PasswordDeriveBytes secretKey = new
    PasswordDeriveBytes(Encoding.ASCII.GetBytes(key), Encoding.ASCII.GetBytes(salt));

    using (RijndaelManaged rijndaelCipher = new RijndaelManaged())
    {
        using (ICryptoTransform decryptor =
        rijndaelCipher.CreateDecryptor(secretKey.GetBytes(32), secretKey.GetBytes(16)))
        {
            using (MemoryStream memoryStream = new MemoryStream(encryptedData))
            {
                using (CryptoStream cryptoStream = new CryptoStream(memoryStream,
                decryptor, CryptoStreamMode.Read))
                {
                    byte[] plainText = new byte[encryptedData.Length];
                    cryptoStream.Read(plainText, 0, plainText.Length);
                    string utf8 = Encoding.UTF8.GetString(plainText);
                    return utf8;
                }
            }
        }
    }
}
}
}
}
}

```



```

StringBuilder sbuilder = new StringBuilder();
ClsPermissioGroup BalPerGrp = new ClsPermissioGroup();
string ClsHLighter = string.Empty;

List<UserPermissions> UserPerSets = BalPerGrp.GetPersets_usr(userName);
if (UserPerSets.Count > 0)
{
    var getViewPer = from Pset in UserPerSets
        where Pset.ParentId == 0
        //&& Pset.View == true

        select Pset;
    sbuilder.Append("<span class='inbar'>");
    sbuilder.Append("<ul>");
    foreach (UserPermissions Uprs in getViewPer)
    {

        if (Uprs.PageUrl != "No-Url")
        {

            if (Uprs.View == true)
            {

                if (HighlightMenuItem(ReqFileName, UserPerSets, Uprs.Hir.ToString(),
Uprs.PageUrl))
                {
                    ClsHLighter = "class ='navhome'";
                }
            }
        }
    }
}

```



```
        sbuilder.Append("<li id='li" + Uprs.PageName.ToString() + "" +
ClsHLighter.ToString() + " runat='server'><a href='" + Uprs.PageUrl + "'><span>" +
Uprs.PageName.ToString() + "</span></a></li>");
```

```
    }
```

```
}
```

```
else
```

```
{
```

```
    var Childs = from Pset in UserPerSets
```

```
        where Pset.Hir.StartsWith(Uprs.Hir) && Pset.View == true &&
Pset.ParentId != 0 && Pset.PageUrl != "No-Url"
```

```
        orderby Pset.PagId ascending
```

```
        select Pset;
```

```
    var PgLanding = Childs.Take(1);
```

```
    foreach (UserPermissions pglan in PgLanding)
```

```
    {
```

```
        if (HighlightMenuItem(ReqFileName, UserPerSets, Uprs.Hir.ToString(),
Uprs.PageUrl) || ReqFileName.ToLower() == "adduser.aspx" || ReqFileName.ToLower() ==
"edituser.aspx")
```

```
        {
```

```
            ClsHLighter = "class = 'navhome'";
```

```
        }
```

```
        sbuilder.Append("<li id='li" + Uprs.PageName.ToString() + "" +
ClsHLighter.ToString() + " runat='server'><a href='" + pglan.PageUrl + "'><span>" +
Uprs.PageName.ToString() + "</span></a></li>");
```

```
    }
```

```
}
```

```
ClsHLighter = string.Empty;
```

```

    }

    //<li id="liEvents" runat="server"> <a
href="Events.aspx"><span>Events</span></a></li>

    //<li id="liNetworksetUp" runat="server"><a
href="NetworksetUp.aspx"><span>Network Setup</span></a></li>

    //<li id="liViewGlobal" runat="server"><a
href="ViewGlobal.aspx"><span>Administration</span></a></li>
    }

    sbuilder.Append("<ul>");
    sbuilder.Append("</span>");

    return sbuilder.ToString();

}

public static bool HighlightMenuItem(string Reqpath, List<UserPermissions> userPerSets,
string strHir, string PageUrl)
{

    if (Reqpath.ToString() == PageUrl)
    {
        return true;
    }
    else
    {
        int i = 0;
        var Childs = from Pset in userPerSets
            where Pset.PageUrl == Reqpath
            select Pset;
        foreach (UserPermissions Uper in Childs)
        {

            if (Uper.Hir.Contains(strHir + " *"))

```

```
    {  
  
        i = i + 1;  
    }  
  
}  
  
if (i == 0)  
{  
    return false;  
}  
else  
{  
    return true;  
}  
}
```

```
public static string GetPermissionSets(string FileName)  
{  
  
    string Perset = string.Empty;  
    string Hiddenfield = string.Empty;  
  
    ClsPermissioGroup cgroup = new ClsPermissioGroup();  
    ClsLoggedInUser Loginuser =  
(ClsLoggedInUser)HttpContext.Current.Session["LoggedInUser_Specs"];  
  
    UserPermissions UserPer = cgroup.GetPersets_Page(Loginuser.UserName,  
FileName);  
  
    if (UserPer.Pageld != 0)
```

```
{
    Hiddenfield = Hiddenfield + " <input type ='hidden' id='hdenSiteacc' value='" + "True"
+ "' />";
}
else
{
    Hiddenfield = Hiddenfield + " <input type ='hidden' id='hdenSiteacc' value='" +
"False" + "' />";
}

if (UserPer.View)
{
    Hiddenfield = Hiddenfield + " <input type ='hidden' id='hdenPerSetView' value='" +
"View" + "' />";
}
else
{
    Hiddenfield = Hiddenfield + " <input type ='hidden' id='hdenPerSetView' value='" + ""
+ "' />";
}

if (UserPer.Add)
{
    Hiddenfield = Hiddenfield + " <input type ='hidden' id='hdenPerSetADD' value='" +
"Add" + "' />";
}
else
{
    Hiddenfield = Hiddenfield + " <input type ='hidden' id='hdenPerSetADD' value='" + ""
+ "' />";
}

if (UserPer.Edit)
{
    Hiddenfield = Hiddenfield + " <input type ='hidden' id='hdenPerSetEdit' value='" +
"Edit" + "' />";
}
```

```

    }
    else
    {
        Hiddenfield = Hiddenfield + " <input type ='hidden' id='hdenPerSetEdit' value='" + "" +
"/>";
    }
    if (UserPer.Delete)
    {
        Hiddenfield = Hiddenfield + " <input type ='hidden' id='hdenPerSetDelete' value='" +
"Delete" + "" />";
    }
    else
    {
        Hiddenfield = Hiddenfield + " <input type ='hidden' id='hdenPerSetDelete' value='" +
"" + "" />";
    }
    return Hiddenfield;
}

```

```

    public static int GetDupRecCount(string TableName, string IdColName, Hashtable
colvalues, string ColName)
    {
        return DowBAL.Common.GetDupRecCount(TableName, IdColName, colvalues,
ColName);
    }

    public static void BindDropDownList(DropDownList SrcDdl, string TableName, string
ValueColumn, string TextColumn, Hashtable ColVal)
    {
        DataSet ds = DowBAL.Common.GetDDIData(TableName, ValueColumn, TextColumn,
ColVal);
        if (ds != null)
        {
            if (ds.Tables[0].Rows.Count > 0)

```

```

    {
        SrcDdl.DataSource = ds;
        SrcDdl.DataValueField = "colVal";
        SrcDdl.DataTextField = "ColTxt";
        SrcDdl.DataBind();
    }
}
}
public static Dictionary<string, string> GetDDIData(string Query)
{

    DataSet ds = SqlHelper.ExecuteDataset(ConnectionHelper.DowConectingString,
CommandType.Text, Query);
    Dictionary<string, string> DDLITEMS = new Dictionary<string, string>();
    if (ds.Tables[0].Rows.Count > 0)
    {
        foreach (DataRow Dr in ds.Tables[0].Rows)
        {
            DDLITEMS.Add(Dr[0].ToString(), Dr[1].ToString());
        }
    }
    return DDLITEMS;
}
public static Dictionary<string, string> GetDDIData(string SpName, string SearchStr)
{

    SqlParameter[] SqlParams = { new SqlParameter { ParameterName =
"@SearchParam", Value = SearchStr, SqlDbType = SqlDbType.VarChar, Size = 100 } };
    DataSet ds = SqlHelper.ExecuteDataset(ConnectionHelper.DowConectingString,
CommandType.StoredProcedure, SpName, SqlParams);

    Dictionary<string, string> DDLITEMS = new Dictionary<string, string>();
    if (ds.Tables[0].Rows.Count > 0)

```

```

    {
        foreach (DataRow Dr in ds.Tables[0].Rows)
        {
            DDLITEMS.Add(Dr[0].ToString(), Dr[1].ToString());
        }
    }
    return DDLITEMS;
}

public static DataSet GetDDI(string SpName)
{

    SqlParameter[] SqlParams = { new SqlParameter { ParameterName =
"@SearchParam", Value = "", SqlDbType = SqlDbType.VarChar, Size = 100 } };

    DataSet ds = SqlHelper.ExecuteDataset(ConnectionHelper.DowConectingString,
CommandType.StoredProcedure, SpName, SqlParams);

    //Dictionary<string, string> DDLITEMS = new Dictionary<string, string>();
    //if (ds.Tables[0].Rows.Count > 0)
    //{
    //    foreach (DataRow Dr in ds.Tables[0].Rows)
    //    {
    //        DDLITEMS.Add(Dr[0].ToString(), Dr[1].ToString());
    //    }
    //}
    //return DDLITEMS;
    return ds;
}

public static int DeleteGroup(int GrpId)
{
    SqlParameter[] Params =
    {
        new SqlParameter{ParameterName="@GrpId",Value = GrpId
,SqlDbType=SqlDbType.Int}

```

```
};  
    return SqlHelper.ExecuteNonQuery(ConnectionHelper.DowConectingString,  
CommandType.StoredProcedure, "deletegroup", Params);  
}
```

```
public static string ConvertUniqueIDToAddress(UInt32 argID)
```

```
{  
    string hexID;  
    char[] idCharArray = new char[16];  
    string strByte;  
    byte byteVAlue;  
    string strPrgAddress;  
  
    // convert to hex  
    hexID = argID.ToString("X").PadLeft(8, '0');  
  
    // Add STX char  
    idCharArray[0] = Convert.ToChar(0x02);  
  
    // Add unique ID  
    for (int i = 0; i < 4; i++)  
    {  
        strByte = hexID.Substring((i * 2), 2);  
        byteVAlue = Convert.ToByte(strByte, 16);  
        idCharArray[(i + 1)] = Convert.ToChar(byteVAlue);  
    }  
  
    // fill the rest of the address with full stops  
    for (int i = 5; i < 16; i++)  
    {  
        idCharArray[i] = '.';  
    }  
}
```



```

    strPrgAddress = new string(idCharArray);

    return strPrgAddress;
}

#region Dynamic Menus
/*Web method that populates the Dynamic Menus*/
//[System.Web.Services.WebMethod]
public static string Main_Menu(int GroupId)
{
    BalMenus objclsper = new BalMenus();
    DataTable table = objclsper.GetMenus(GroupId);
    DataRow[] parentMenus = table.Select("parentid = 0");
    var sb = new StringBuilder();
    string unorderedList = Generate_MainMenu(parentMenus, table, sb, "0");
    //lft_panel.InnerHtml = unorderedList;
    return unorderedList;
}

/*Recursive method that populates the Dynamic Menus*/
private static string Generate_MainMenu(DataRow[] menu, DataTable table, StringBuilder
sb, string level)
{
    string Flines = "";
    if (level == "0")
    {
        Flines = String.Format(@"<ul id='ULevel_{0}'>", level);
    }
    else if (level == "1")
    {

```

```
Flines = String.Format(@"<ul id='ULevel_{0}' style='top: 29px; box-shadow: rgb(170, 170, 170) 1px 1px 5px; -webkit-box-shadow: rgb(170, 170, 170) 1px 1px 5px; visibility: visible; left: 0px; width: 171px; display: none;'>", level);
```

```
}
```

```
else if (level == "2")
```

```
{
```

```
Flines = String.Format(@"<ul id='ULevel_{0}' style='display: none; top: 0px; box-shadow: rgb(170, 170, 170) 1px 1px 5px; -webkit-box-shadow: rgb(170, 170, 170) 1px 1px 5px; visibility: visible;'>", level);
```

```
}
```

```
sb.AppendLine(Flines);
```

```
string styleclsname = string.Empty;
```

```
string ConcatValue = string.Empty;
```

```
if (menu.Length > 0)
```

```
{
```

```
foreach (DataRow dr in menu)
```

```
{
```

```
string mapid = dr["MapId"].ToString();
```

```
string menuText = dr["PageName"].ToString();
```

```
string menuUrl = dr["PageUrl"].ToString();
```

```
string pageid = dr["pageId"].ToString();
```

```
string page_veiw = dr["view"].ToString();
```

```
string line = "";
```

```
if (page_veiw.ToLower() == "true")
```

```
{
```

```
if (menuUrl.ToLower() != "no-url")
```

```
{
```

```
string pageurl = menuUrl;
```

```
if (dr["level"] == "0")
```

```

        {
            line = String.Format(@" <li id='pgid_{2}' style='padding-left:20px;'><a
href='{0}'>{1}</a></li>", pageurl, menuText, pageid);
        }
        else
        {
            line = String.Format(@" <li id='pgid_{2}'><a href='{0}'>{1}</a></li>",
pageurl, menuText, pageid);
            //<li><a href="#">Residential</a></li>
        }

    }
    else
    {

        if (dr["level"] == "0")
        {
            line = String.Format(@"<li style='z-index:{1};'><a href='#' style='cursor:
default; padding-right: 23px;'>{0}</a>", menuText, pageid);

        }
        else
        {
            line = String.Format(@"<li style='z-index:{1};'><a href='#'
style='cursor:default;'>{0}</a>", menuText, pageid);

        }
    }
}

sb.Append(line);
string pid = dr["pageId"].ToString();
DataRow[] subMenu = table.Select(String.Format("parentid = {0}", pid));
if (subMenu.Length > 0)

```

```

        {
            var subMenuBuilder = new StringBuilder();
            sb.Append(Generate_MainMenu(subMenu, table, subMenuBuilder,
dr["level"].ToString()));
        }
        sb.Append("</li>");
    }
}

sb.Append("</ul>");
string test = sb.ToString();
return sb.ToString();
}
#endregion

```

```
#region Dynamic LeftNavBar
```

```
/*Web method that populates the Dynamic Menus*/
```

```
//[System.Web.Services.WebMethod]
```

```
public static string BindDynamic_DDL(int lbtid, int loginid, int lbtntype)
```

```

{
    var sb = new StringBuilder();
    BalMenus objclsper = new BalMenus();
    DataSet ds = objclsper.BindDynamic_DDL(lbtid, loginid, lbtntype);
    DataTable dtdll = ds.Tables[0];

    if (ds != null)
    {
        if (ds.Tables[0].Rows.Count > 0)
        {
            //SrcDdl.DataSource = ds;
            //SrcDdl.DataValueField = "ID";
            //SrcDdl.DataTextField = "TEXT";

```

```

        //SrcDdl.DataBind();

        foreach (DataRow drnew in dtdll.Rows)
        {
            //--- commented ASP.net Dropdown dynamic generation code

            // sb.Append(string.Format(@"<asp:ListItem
Value='{0}'>{1}</asp:ListItem>", drnew["ID"], drnew["TEXT"]));//;drnew["ID"].ToString();drnew["
TEXT"].ToString();

            sb.Append(string.Format(@"<option>{0}</option>",
drnew["TEXT"]));//;drnew["ID"].ToString();drnew["TEXT"].ToString();

        }
    }
}

return sb.ToString();
}

public static string LeftNaveBar(int pageid, int userid, int groupid)
{
    BalMenus objclsper = new BalMenus();
    DataTable table = objclsper.GetLeftButtons(pageid, groupid);
    string unorderedList = Generate_LeftNaveBar(table, userid);
    return unorderedList;

    //1= Action
    //2= WorkPlace
    //3=Filter (Common values)
    //4=Filter (user based values)
}

private static string Generate_LeftNaveBar(DataTable table, int userid)
{
    var sb = new StringBuilder();

    //string Flines = String.Format(@"<div id='LeftSideBar' style='float: left; width: 15%;
background-color: lightblue; display: none;' runat='server'>");

    sb.AppendLine("");

    string line0 = "", line1 = "", line2 = "", line3 = "", ActionButtonStart = "", End = "",
FilterStart = "", WorkPlaceStart = "";

```

```

if (table.Rows.Count > 0)
{
    ActionButtonStart = "<ul class='nav'><li class='active'><span class='menu-
text'>ACTIONS</span></li>";
    WorkPlaceStart = "<ul class='nav'><li
class='active'><span>WORKPLACE</span></li>";
    FilterStart = "<ul class='nav'><li class='active'><span>FILTERS</span></li>";
    End = "</ul>";

    foreach (DataRow dr in table.Rows)
    {
        string buttonid = dr["lbtnid"].ToString();
        string buttonname = dr["lbtnname"].ToString();
        string buttonImage = dr["lbtnimage"].ToString();
        string buttonurl = dr["lbtnurl"].ToString();
        string pageurl = dr["pageurl"].ToString();
        string buttontype = dr["lbtnatype"].ToString();
        string buttonveiw = dr["show"].ToString();
        string buttoncode = dr["lbtncode"].ToString();

        if (buttontype == "1" || buttontype == "2")
        {
            if (buttonveiw.ToLower() == "1")
            {
                line0 = string.Format(@"<li><a id='{3}' href='{0}'><i class='{1}'></i><span>
{2}</span></a></li>", pageurl, buttonImage, buttonname, buttoncode);
            }
            else
            {
                line0 = string.Format(@"<li class='disable'><a href='{0}'><i
class='{1}'></i><span> {2}</span></a></li>", pageurl, buttonImage, buttonname);
            }
        }
    }
}

```

```

if (buttontype == "1")
{
    line1 = line1 + System.Environment.NewLine + line0;
}
else
{
    line2 = line2 + System.Environment.NewLine + line0;
}
}

else if (buttontype == "3" || buttontype == "4")
{
    //<form><select> // </select></form>
    //--- commented ASP.net Dropdown dynamic generation code
    //string dropdwon1 = string.Format("<asp:DropDownList ID='DDL_{0}'
runat='server'>", buttonname);
    //string ddlitems = BindDynamic_DDL(Convert.ToInt32(buttonid), userid,
Convert.ToInt32(buttontype)) + "</asp:DropDownList>";

    string dropdwon1 = "<form><select>" +
BindDynamic_DDL(Convert.ToInt32(buttonid), userid, Convert.ToInt32(buttontype)) +
"</select></form>";

if (buttonveiw.ToLower() == "1")
{

    line3 = line3 + System.Environment.NewLine +
string.Format("@<li><span>{0} :</span>{1}</li>", buttonname, dropdwon1);
    //BindDynamic_DDL("DDL_" + buttonname, buttonid, buttontype, 1);
}
else
{

    //line3 = line3 + System.Environment.NewLine + string.Format("@<li><a
href='#'><span>{0} :</span><asp:DropDownList ID='DDL_{0}'
runat='server'></asp:DropDownList></a></li>", buttonname);

```

```
        line3 = line3 + System.Environment.NewLine + string.Format(@"<li  
class='disable'><span>{0} :</span>{1}</li>", buttonname, dropdwon1);
```

```
    }
```

```
}
```

```
}
```

```
if (line1.Length > 0)
```

```
{
```

```
    sb.AppendLine(ActionButtonStart);
```

```
    sb.AppendLine(line1);
```

```
    sb.AppendLine(End);
```

```
}
```

```
if (line2.Length > 0)
```

```
{
```

```
    sb.AppendLine(WorkPlaceStart);
```

```
    sb.AppendLine(line2);
```

```
    sb.AppendLine(End);
```

```
}
```

```
if (line3.Length > 0)
```

```
{
```

```
    sb.AppendLine(FilterStart);
```

```
    sb.AppendLine(line3);
```

```
    sb.AppendLine(End);
```

```
}
```

```
}
```

```
return sb.ToString();
```

```
}
```

```
#endregion
```



```

#region Dynamic TopBar
/*Web method that populates the Dynamic Menus*/
[System.Web.Services.WebMethod]
public static string TopBarBar(string userid)
{
    // BalMenus objclsper = new BalMenus();
    // //D\ DataTable table = objclsper.GetLeftButtons(userid);
    // //D\ string unorderedList = Generate_TopBar();

    // string unorderedList = Generate_TopBar(userid);
    // return unorderedList;

}

public static string Generate_bottommenu()
{
    DDLogTracer objlog = new DDLogTracer();

    string line1 = "";
    var sb = new StringBuilder();
    List<Eteproductcat> et = new Products().GetProductCats("").ToList();

    line1 = "<ul class='nav-main'>";

    sb.AppendLine(line1);

    foreach (var row in et)
    {
        if (row.GCount > 0)
        {
            line1 = "<li class='nav-main-item'>";

```

```

        sb.AppendLine(line1);
        line1 = "<a class='listing-caption span10' href='OrderSearchList.aspx?id=" +
row.pcatid + ">" + row.pcatdesc + "</a>";
        sb.AppendLine(line1);
        line1 = "</li>\n";

        sb.AppendLine(line1);
        Generate_BottomSubMenu(row.pcatid,sb);
    }
}
line1 = "</ul>\n";
sb.AppendLine(line1);

return sb.ToString();

```

```

}
public static string Generate_BottomSubMenu(int id,StringBuilder sb)
{
    DDLogTracer objlog = new DDLogTracer();

    string line1 = "";
    //var sb = new StringBuilder();
    List<EteProductGroup> et = new Products().GetProductGroups("").ToList();
    et = (from c in et where c.PCatId==id orderby c.PCatId ascending select c).ToList();
    int catid = id;
    int cpgid = 0;// et1.pgid;
    line1 = "<div id='btab" + catid.ToString() + "' class='tabs_holder ' runat='server'
style='display:block'>\n"; //
    sb.AppendLine(line1);

    foreach (var row in et)
    {
        int catid1 = row.PCatId;

```

```

int cpgid1 = row.pgid;
if (catid1 != catid)
{
    line1 = "</div>";
    sb.AppendLine(line1);
    line1 = "<div id='btabs' + catid1.ToString() + "' class='pure-u-4-5 tabs_holder'
runat='server' style='display:block'>\n";
    sb.AppendLine(line1);
    catid = row.PCatId;
    cpgid = row.pgid;

    line1 = "<a href='GOrderSearchList.aspx?pid=' + cpgid1.ToString() + "' class='nav-
main-link span2' onmouseover='javascript:ShowSubMenu(" + cpgid1 + ")'>" +
row.pgroupdescription + "</a>";
    sb.AppendLine(line1);
}
else
{
    line1 = "<a href='GOrderSearchList.aspx?pid=' + cpgid1.ToString() + "' class='nav-
main-link span2'>" + row.pgroupdescription + "</a>";

    sb.AppendLine(line1);

    line1 = "<div id='bsubtab' + cpgid1.ToString() + "' class='pure-u-4-5' runat='server'
style='display:block'>\n";
    sb.AppendLine(line1);
    // line1 ="<br style='clear: left'>\n";
    //sb.AppendLine(line1);

    //for inner menu
    //List<Eteproduct> ep = new Products().GetProductbyGroup(cpgid1, "").ToList();
    //foreach (var dt in ep)
    //{

```

```
        // line1 = "<a href='OrderSearchList.aspx?pid=" + dt.ProductId.ToString() + "  
class='tabsHeaderList__item span2'>" + dt.ProductName + "</a>";
```

```
        // sb.AppendLine(line1);
```

```
    //}
```

```
    //inner mene ends
```

```
    line1 = "</div>";
```

```
    sb.AppendLine(line1);
```

```
    //line1 = "</div>";
```

```
    //sb.AppendLine(line1);
```

```
    }
```

```
}
```

```
line1 = "</div>";
```

```
sb.AppendLine(line1);
```

```
//objlog.Trace_Log(1, "Generate_Menu", "1", "menu", sb.ToString());
```

```
return sb.ToString();
```

```
}
```

```
public static string Generate_newmenu()
```

```
{
```

```
    DDLogTracer objlog = new DDLogTracer();
```

```
    int i = 0;
```

```
    string line1 = "";
```

```
    var sb = new StringBuilder();
```

```
    List<EteMenuList> Lst = new tbIPagess().GetMenuListbystype(1);
```

```
    List<EteMenuList> et = (from c in Lst where c.Stype == 1 && c.level == 1 orderby  
c.priority ascending select c).ToList();
```

```
    line1 = "<ul class='nav navbar-nav'>";
```

```

sb.AppendLine(line1);

foreach (var row in et)
{
    //if (row.GCount > 0)
    {
        line1 = "<li class='dropdown mega-menu-fullwidth'>";
        sb.AppendLine(line1);

        line1 = "<a class='dropdown-toggle' href='javascript:void(0);' data-
hover='dropdown' data-toggle='dropdown'>" + row.pagename + "</a>";
        sb.AppendLine(line1);

        line1 = "<ul class='dropdown-menu'>";
        sb.AppendLine(line1);

        sb.AppendLine("<li>");
        sb.AppendLine("<div class='mega-menu-content'>");
        sb.AppendLine("<div class='container'>");
        sb.AppendLine("<div class='row'>");
        sb.AppendLine("<div class='col-md-2 col-sm-6'>");

        List<EteMenuList> line = (from c in Lst where c.parentid == row.pageid orderby
c.priority select c).ToList();

        foreach (var ln in line)
        {
            i = 1;

            sb.AppendLine("<h3 class='mega-menu-heading'>" + ln.pagename + "</h3>");
            sb.AppendLine("<ul class='list-unstyled style-list'>");

            List<EteMenuList> ep = (from c in Lst where c.parentid == ln.pageid orderby
c.priority select c).ToList();

            foreach (var dt in ep)
            {
                sb.AppendLine("<li><a href='"+dt.pageurl+"'>" + dt.pagename + "</a></li>");
                i = i + 1;

                if (i > 16)

```

```

        {
            sb.AppendLine("</ul>");
            sb.AppendLine("</div>");
            sb.AppendLine("<div class='col-md-2 col-sm-6'>");
            sb.AppendLine("<ul class='list-unstyled style-list'>"); //checking for ul

                i = 0;
            } //endif for 1>15

        } //for each product
        sb.AppendLine("</ul>");
        sb.AppendLine("</div>");
        sb.AppendLine("<div class='col-md-2 col-sm-6'>");
        sb.AppendLine("<ul class='list-unstyled style-list'>");
        i = 0;
        //sb.AppendLine("</ul>");
    } //for each group
    i = 0;
    sb.AppendLine("</div>");
    sb.AppendLine("</div>");
    sb.AppendLine("</div>");
    sb.AppendLine("</div>");
    line1 = "</li>\n";
    sb.AppendLine(line1);
    sb.AppendLine("</ul>");
} //for endif
// sb.AppendLine("</ul>");

} //for each cat
sb.AppendLine("</ul>");
//line1 = "</li>\n";

```

```

        //sb.AppendLine(line1);
        //sb.AppendLine("</ul>");
        //objlog.Trace_Log(1, "Generate_newmenu", "1", "common", sb.ToString());

        return sb.ToString();

    }

    public static string Generate_newmenuold()
    {
        DDLogTracer objlog = new DDLogTracer();
        int i = 0;
        string line1 = "";
        var sb = new StringBuilder();
        List<Eteproductcat> et = new Products().GetProductCats("").ToList();
        List<EteProductGroup> et1 = new Products().GetProductGroups("").ToList();
        et = (from c in et where c.Stype==1 orderby c.pcatid ascending select c).ToList();

        line1 = "<ul class='nav navbar-nav'>";

        sb.AppendLine(line1);

        foreach (var row in et)
        {
            if (row.GCount > 0)
            {
                line1 = "<li class='dropdown mega-menu-fullwidth'>";
                sb.AppendLine(line1);

                line1 = "<a class='dropdown-toggle' href='javascript:void(0);' data-hover='dropdown' data-toggle='dropdown'>" + row.pcatdesc + "</a>";
                sb.AppendLine(line1);

                line1 = "<ul class='dropdown-menu'>";
                sb.AppendLine(line1);
                sb.AppendLine("<li>");
            }
        }
    }

```

```

sb.AppendLine("<div class='mega-menu-content'>");
sb.AppendLine("<div class='container'>");
sb.AppendLine("<div class='row'>");
sb.AppendLine("<div class='col-md-2 col-sm-6'>");

List<EteProductGroup> line=(from c in et1 where c.PCatId==row.pcatid orderby
c.pgid select c).ToList();

foreach(var ln in line)
{
    i = 1;

    sb.AppendLine("<h3 class='mega-menu-heading'>" + ln.pgrouppdescription +
"</h3>");

    sb.AppendLine("<ul class='list-unstyled style-list'>");

    List<Eteproduct> ep = new Products().GetProductbyGroup(ln.pgid, "").ToList();

    foreach (var dt in ep)
    {
        sb.AppendLine("<li><a href='ShoppingList.aspx?pgid=" + dt.ProductId + ">" +
dt.ProductName + "</a></li>");

        i = i + 1;

        if (i > 16)
        {
            sb.AppendLine("</ul>");
            sb.AppendLine("</div>");
            sb.AppendLine("<div class='col-md-2 col-sm-6'>");
            sb.AppendLine("<ul class='list-unstyled style-list'>");

            i = 0;
        } //endif for 1>15
    } //for each product

    sb.AppendLine("</ul>");
} //for each group

```



```

        sb.AppendLine("</div>");
        sb.AppendLine("</div>");
        sb.AppendLine("</div>");
        sb.AppendLine("</div>");
        line1 = "</li>\n";
        sb.AppendLine(line1);
        sb.AppendLine("</ul>");
    } //for endif
    sb.AppendLine("</ul>");

} //for each cat
sb.AppendLine("</ul>");
//line1 = "</li>\n";

//sb.AppendLine(line1);
//sb.AppendLine("</ul>");
objlog.Trace_Log(1, "Generate_topmenu", "1", "common", sb.ToString());

return sb.ToString();

}

public static string Generate_servicemenuold()
{
    DDLogTracer objlog = new DDLogTracer();
    int i = 0;
    string line1 = "";
    var sb = new StringBuilder();
    List<Eteproductcat> et = new Products().GetProductCats("").ToList();
    List<EteProductGroup> et1 = new Products().GetProductGroups("").ToList();
    et = (from c in et where c.Stype ==2 orderby c.pcatid ascending select c).ToList();

    line1 = "<ul class='nav navbar-nav'>";

```

```

sb.AppendLine(line1);

foreach (var row in et)
{
    if (row.GCount > 0)
    {
        line1 = "<li class='dropdown mega-menu-fullwidth'>";
        sb.AppendLine(line1);

        line1 = "<a class='dropdown-toggle' href='javascript:void(0);' data-hover='dropdown' data-toggle='dropdown'>" + row.pcatdesc + "</a>";
        sb.AppendLine(line1);

        line1 = "<ul class='dropdown-menu'>";
        sb.AppendLine(line1);

        sb.AppendLine("<li>");
        sb.AppendLine("<div class='mega-menu-content'>");
        sb.AppendLine("<div class='container'>");
        sb.AppendLine("<div class='row'>");
        sb.AppendLine("<div class='col-md-2 col-sm-6'>");

        List<EteProductGroup> line = (from c in et1 where c.PCatId == row.pcatid orderby c.pgid select c).ToList();

        foreach (var ln in line)
        {
            i = 1;

            sb.AppendLine("<h3 class='mega-menu-heading'>" + ln.pgroupdescription +
"</h3>");

            sb.AppendLine("<ul class='list-unstyled style-list'>");

            List<Eteproduct> ep = new Products().GetProductbyGroup(ln.pgid, "").ToList();

            foreach (var dt in ep)
            {
                sb.AppendLine("<li><a href='ShoppingList.aspx?pgid=" + dt.ProductId + "'>" +
dt.ProductName + "</a></li>");

                i = i + 1;
            }
        }
    }
}

```

```

        if (i > 15)
        {
            sb.AppendLine("</ul>");
            sb.AppendLine("</div>");
            sb.AppendLine("<div class='col-md-2 col-sm-6'>");
            sb.AppendLine("<ul class='list-unstyled style-list'>");

                i = 0;
            } //endif for 1>15
        } //for each product

            sb.AppendLine("</ul>");
        } //for each group

            sb.AppendLine("</div>");
            sb.AppendLine("</div>");
            sb.AppendLine("</div>");
            sb.AppendLine("</div>");
            line1 = "</li>\n";
            sb.AppendLine(line1);
            sb.AppendLine("</ul>");
        } //for endif
        sb.AppendLine("</ul>");

    } //for each cat
    sb.AppendLine("</ul>");
    //line1 = "</li>\n";

    //sb.AppendLine(line1);
    //sb.AppendLine("</ul>");
    objlog.Trace_Log(1, "Generate_servicesmenu", "1", "common", sb.ToString());

```

```

return sb.ToString();

}

public static string Generate_servicemenu()
{
    DDLogTracer objlog = new DDLogTracer();
    int i = 1;
    string line1 = "";
    var sb = new StringBuilder();
    List<EteMenuList> Lst = new tblPagess().GetMenuListbystype(2);
    var et = (from c in Lst where c.level == 0 orderby c.priority ascending select c).Take(8);

    line1 = "\n<ul class='nav navbar-nav'>";

    sb.AppendLine(line1);
    foreach (var row in et)
    {
        //if (row.GCount > 0)
        {
            line1 = "<li class='dropdown mega-menu-fullwidth'>";
            sb.AppendLine(line1);

            line1 = "<a class='dropdown-toggle' href='javascript:void(0);' data-hover='dropdown' data-toggle='dropdown'>" + row.pagename + "</a>";
            sb.AppendLine(line1);

            line1 = "<ul class='dropdown-menu'>";
            sb.AppendLine(line1);

            sb.AppendLine("<li>");

            sb.AppendLine("<div class='mega-menu-content'>");

            sb.AppendLine("<div class='container'>");

            sb.AppendLine("<div class='row'>");

            sb.AppendLine("<div class='col-md-3 col-sm-6'>");

            List<EteMenuList> line = (from c in Lst where c.parentid == row.pageid orderby c.priority select c).ToList();

```

```

foreach (var ln in line)
{
    i = i + 2;
    sb.AppendLine("<h3 class='mega-menu-heading'" + ln.pagename + "</h3>");
    sb.AppendLine("<ul class='list-unstyled style-list'">");
    List<EteMenuList> ep = (from c in Lst where c.parentid == ln.pageid orderby
c.priority select c).ToList();
    foreach (var dt in ep)
    {
        sb.AppendLine("<li><a href='" + dt.pageurl + "'"> + dt.pagename + "</a></li>");
        i = i + 1;

    } //var dt in ep
    sb.AppendLine("</ul>");
    if (i > 10)
    {
        //sb.AppendLine("</ul>");
        sb.AppendLine("</div>");
        sb.AppendLine("<div class='col-md-3 col-sm-6'">");
        //sb.AppendLine("<ul class='list-unstyled style-list'">");
        i = 0;
    } //endif for 1>15

    //sb.AppendLine("</div>");
    //sb.AppendLine("<div class='col-md-2 col-sm-6'">");
    //sb.AppendLine("<ul class='list-unstyled style-list'">");

    //sb.AppendLine("</ul>");
} //var ln in line

sb.AppendLine("</div>");
sb.AppendLine("</div>");

```

```

        sb.AppendLine("</div>");
        sb.AppendLine("</div>");
        line1 = "</li>\n";
        sb.AppendLine(line1);
        sb.AppendLine("</ul>");
    } //for endif
    //    sb.AppendLine("</ul>");
} //for each cat
//second round

//second round complete
sb.AppendLine("</ul>");
//line1 = "</li>\n";

//sb.AppendLine(line1);
//sb.AppendLine("</ul>");
objlog.Trace_Log(1, "Generate_topmenu", "1", "common", sb.ToString());

return sb.ToString();
}

public static string Generate_servicemenu1()
{
    DDLogTracer objlog = new DDLogTracer();
    int i = 0;
    string line1 = "";
    var sb = new StringBuilder();
    List<Eteproductcat> et = new Products().GetProductCats("").ToList();
    List<EteProductGroup> et1 = new Products().GetProductGroups("").ToList();
    et = (from c in et where c.Stype == 2 orderby c.pcatid ascending select c).ToList();

```

```

line1 = "<ul class='nav navbar-nav'>";

sb.AppendLine(line1);

foreach (var row in et)
{
    if (row.GCount > 0)
    {
        line1 = "<li class='dropdown mega-menu-fullwidth'>";
        sb.AppendLine(line1);

        line1 = "<a class='dropdown-toggle' href='javascript:void(0);' data-hover='dropdown' data-toggle='dropdown'>" + row.pcatdesc + "</a>";
        sb.AppendLine(line1);

        line1 = "<ul class='dropdown-menu'>";
        sb.AppendLine(line1);

        sb.AppendLine("<li>");
        sb.AppendLine("<div class='mega-menu-content'>");
        sb.AppendLine("<div class='container'>");
        sb.AppendLine("<div class='row'>");
        sb.AppendLine("<div class='col-md-2 col-sm-6'>");

        List<EteProductGroup> line = (from c in et1 where c.PCatId == row.pcatid orderby c.pgjid select c).ToList();

        foreach (var ln in line)
        {
            i = 1;

            sb.AppendLine("<h3 class='mega-menu-heading'>" + ln.pgroupdescription + "</h3>");

            sb.AppendLine("<ul class='list-unstyled style-list'>");

            List<Eteproduct> ep = new Products().GetProductbyGroup(ln.pgjid, "").ToList();

            foreach (var dt in ep)
            {
                sb.AppendLine("<li><a href='ServicesList.aspx?sid=" + dt.ProductId + "'>" + dt.ProductName + "</a></li>");

                i = i + 1;
            }
        }
    }
}

```

```

        if (i > 15)
        {
            sb.AppendLine("</ul>");
            sb.AppendLine("</div>");
            sb.AppendLine("<div class='col-md-2 col-sm-6'>");
            sb.AppendLine("<ul class='list-unstyled style-list'>");

                i = 0;
            } //endif for 1>15
        } //for each product

            sb.AppendLine("</ul>");
        } //for each group

        sb.AppendLine("</div>");
        sb.AppendLine("</div>");
        sb.AppendLine("</div>");
        sb.AppendLine("</div>");
        line1 = "</li>\n";
        sb.AppendLine(line1);
    } //for endif
} //for each cat
sb.AppendLine("</ul>");
line1 = "</li>\n";

sb.AppendLine(line1);
sb.AppendLine("</ul>");
objlog.Trace_Log(1, "Generate_servicemenu", "1", "common", sb.ToString());

return sb.ToString();

```



```

}
public static string Generate_topmenu()
{
    DDLogTracer objlog = new DDLogTracer();

    string line1 = "";
    var sb = new StringBuilder();
    List<Eteproductcat> et = new Products().GetProductCats("").ToList();
    var et1 = (from c in et where c.pcatid < 11 select c).ToList();
    line1 = "<ul class='nav-main'>";

    sb.AppendLine(line1);

    foreach (var row in et1)
    {
        if (row.GCount > 0)
        {
            line1 = "<li class='nav-main-item'>";
            sb.AppendLine(line1);

            line1 = "<a class='nav-main-link app-header-tab'
onmouseover='\"javascript:ShowMenu(\"tab\" + row.pcatid + \"')\"
href='OrderSearchList.aspx?id=\" + row.pcatid + \">\" + row.pcatdesc + \"</a>\";
//onmouseout='\"javascript:hideDiv(\"tab\"+ row.pcatid + \"')\"\"
            sb.AppendLine(line1);

            line1 = "</li>\n";
            sb.AppendLine(line1);
        }
    }
    line1 = "</ul>\n";
    sb.AppendLine(line1);

    //objlog.Trace_Log(1, "Generate_topmenu", "1", "common", sb.ToString());
}

```

```

return sb.ToString();

}

public static string Generate_Menu()
{
    DDLogTracer objlog = new DDLogTracer();
    string line1 = "\n";
    var sb = new StringBuilder();
    List<EteProductGroup> et = new Products().GetProductGroups("").ToList();
    et = (from c in et orderby c.PCatId ascending select c).ToList();
    var et1 = (from c in et select c).FirstOrDefault();
    int catid = et1.PCatId;
    int cpgid = 0;// et1.pgid;

    line1 = "\n<div id='tab" + catid.ToString() + "' class='col-xs-4 col-sm-4 col-md-4 item-title
ActiveLi' runat='server' style='display:block'>\n";
    sb.AppendLine(line1);

    foreach (var row in et)
    {
        int catid1 = row.PCatId;
        int cpgid1 = row.pgid;
        if (catid1 != catid)
        {
            line1 = "</div>";
            sb.AppendLine(line1);

            line1 = "<div id='tab" + catid1.ToString() + "' class='col-xs-4 col-sm-4 col-md-4
item-title ' runat='server' style='display:none'>\n"; // class='tabs_holder'
            sb.AppendLine(line1);

            catid = row.PCatId;
            cpgid = row.pgid;

            line1 = "<a href='GOrderSearchList.aspx?pid=" + cpgid1.ToString() + "' class='col-
xs-12 col-sm-6 col-md-6 item-title sub-heading '
onmouseover='\"javascript:showTable('subtab" + cpgid1 + "')\" >" + row.pgroupdescription +

```

```

"</a>"; //onmouseover="\javascript:showDiv('subtab" + cpgid1 + ")\"
onmouseout="\javascript:hideDiv('subtab" + cpgid1 + ")\"

        //line1 = "<a href='GOrderSearchList.aspx?pid=" + cpgid1.ToString() + "\"
class='nav-main-link span2 ' >" + row.pgroupdescription + "</a>";

        sb.AppendLine(line1);

    }

    else

    {

        line1 = "<a href='GOrderSearchList.aspx?pid=" + cpgid1.ToString() + "\" class='col-
xs-12 col-sm-6 col-md-6 item-title sub-heading'
onmouseover="\javascript:showTable('subtab" + cpgid1 + ")\">" + row.pgroupdescription +
"</a>";

        //<tr><th> </th>";

        sb.AppendLine(line1);

        sb.Append("<table id='subtab" + cpgid1.ToString() + "\" class='table'
style='display:none'><tr>");

        //line1 = "<div id='subtab" + cpgid1.ToString() + "\" class='class='col-xs-4 col-sm-4
col-md-4' runat='server' style='display:block'>\n";

        //sb.AppendLine(line1);

        // line1 = "<br style='clear: left'>\n";

        //sb.AppendLine(line1);

        //for inner menu

        List<Eteproduct> ep = new Products().GetProductbyGroup(cpgid1, "").ToList();

        foreach (var dt in ep)

        {

            line1 = "<td><a href='OrderSearchList.aspx?pid=" + dt.ProductId.ToString() + "\"
class='tabsHeaderList__item'>" + dt.ProductName + "</a></td>"; // span2

            sb.AppendLine(line1);

        }

        //inner mene ends

        line1 = "</tr></table>";//</div>";

```

```

        sb.AppendLine(line1);
        //line1 = "</div>";
        //sb.AppendLine(line1);

    }
}
line1 = "</div>";
sb.AppendLine(line1);

//objlog.Trace_Log(1, "Generate_Menu", "1", "menu", sb.ToString());

return sb.ToString();

}
public static string Generate_TopBarMenu(string user, int RoleId, int DealerId, int UserId)
{
    DDLogTracer objlog = new DDLogTracer();

    //    CustomerOrders co = new CustomerOrders();
    int dftcnt = new Products().GetSLItems(UserId, true);//
    co.GetCustomerDraftOrderNos(RoleId, DealerId, UserId);
    Flags FlgCnt = new Flags();
    int Saleoff = new Products().GetSLItems(UserId, false); ;
    ActionTasks ActionTsk = new ActionTasks();
    int CartItems = new Products().GetCartItems(UserId);
    var sb = new StringBuilder();
    sb.AppendLine("");
    string line1 = "", line2 = "";
    string BuyOffers = string.Empty;
    string SaleOffers = string.Empty;
    string MyCarts = string.Empty;
    string MyProducts = string.Empty;
    string LoggedInUser = string.Empty;

```

```

string logoutbutton = string.Empty;
string loginbutton = string.Empty;
String Profile = string.Empty;
string registerbutton = string.Empty;
string DealerName;
DealerName = new Dealers().GetDealerName(DealerId);
// user = user;//+ " (" + DealerName + ")";
line1 = "<ul class='pull-right'>";
BuyOffers = string.Format("<li style='border-right: none;'><a
href='BSOffers.aspx?pageid=23&stype=true' class='draft'><button type='button' class='btn
btn-primary'>WishList({0})</a></li>", dftcnt);
SaleOffers = string.Format("<li style='border-right: none;'><a
href='BSOffers.aspx?pageid=23&stype=false' class='draft'>Sale Offers({0})</a></li>",
Saleoff);
MyCarts = string.Format("<li style='border-right: none;'><a
href='MyCart.aspx?pageid=129' class='draft'><button type='button' class='btn btn-
primary'>My Cart({0})</a></li>", CartItems);
//MyProducts = string.Format("<li><a href='{0}' class='btn btn-primary'>My
Products</a></li>", "ProductSearchList.aspx?pageid=26");
MyProducts = string.Format("<li style='border-right: none;'><a
href='ProductSearchList.aspx?pageid=26'><button type='button' class='btn btn-primary'><i
class='fa fa-user'></i>&nbsp;&nbsp;&nbsp;My Products</button></a></li>");
logoutbutton = string.Format("<li style='border-right: none;'><a
href='logout.aspx'><button type='button' class='ddsmoothmenu'><i class='fa fa-power-
off'></i>&nbsp;&nbsp;&nbsp;Log Out</button></a></li>");
// logoutbutton = string.Format("<li style='border-top: 5px;'><a class='btn btn-primary'
href='logout.aspx'><i class='fa fa-power-off'></i>&nbsp;&nbsp;&nbsp;log out</a></li>");
//loginbutton = string.Format("<li style='border-right: none;'><a class='ddsmoothmenu'
href='login.aspx'><i class='fa fa-user'></i>&nbsp;&nbsp;&nbsp;login</a></li>");
registerbutton = string.Format("<li style='border-right: none;'><a class='btn btn-primary'
href='ManageUser.aspx?pageid=4'><i class='fa fa-
users'></i>&nbsp;&nbsp;&nbsp;Register</a></li>");
// Profile = string.Format("<li style='border-right: none;'><a
href='UserDetail.aspx?pageid=25&ld=" + UserId + "'><button type='button'
class=ddsmoothmenu'><i class='fa fa-user'></i>&nbsp;&nbsp;&nbsp;Profile</button></a></li>");
//LoggedInUser = string.Format("<li style='border-right: none;'><a href='Login.aspx'
class='ddsmoothmenu'><i class='fa fa-user'></i>&nbsp;&nbsp;&nbsp;Login</button></a></li>");
registerbutton = string.Format("<li style='border-right: none;'><a
href='ManageUser.aspx?pageid=4'

```

```
class='ddsmoothmenu'>&nbsp;Register&nbsp;</button></a></li>");// <i class='fa fa-  
users'></i>
```

```
    LoggedInUser = string.Format("<li><a href='#' class='ddsmoothmenu' style='line-  
height:15px' >{0}</a></li>", "Welcome " + user);
```

```
    Profile = string.Format("<li style='border-right: none;'><a  
href='UserDetail.aspx?pageid=25&Id=" + UserId + "' class='ddsmoothmenu'  
></i>&nbsp;Profile&nbsp;</button></a></li>");// <i class='fa fa-user'>
```

```
    logoutbutton = string.Format("<li style='border-right: none;'><a href='logout.aspx'  
class='ddsmoothmenu' >&nbsp;Logout&nbsp;</button></a></li>");// <i class='fa fa-power-  
off'></i>
```

```
    MyCarts = string.Format("<li style='border-right: none;'><a  
href='MyCart.aspx?pageid=129' class='ddsmoothmenu'> Cart({0})</a></li>", CartItems);
```

```
    BuyOffers = string.Format("<li style='border-right: none;'><a  
href='MyWishList.aspx?pageid=23&stype=true'  
class='ddsmoothmenu'>WishList({0})</a></li>", dftcnt);
```

```
    line2 = "</ul>";
```

```
sb.AppendLine(line1);
```

```
//sb.AppendLine(" ");
```

```
if (user != "Guest")
```

```
{
```

```
    // sb.AppendLine(BuyOffers);
```

```
    //sb.AppendLine(SaleOffers);
```

```
    // sb.AppendLine(MyCarts);
```

```
    //sb.AppendLine(MyProducts);
```

```
    sb.AppendLine(LoggedInUser);
```

```
    sb.AppendLine(Profile);
```

```
    sb.Append(MyCarts);
```

```
    sb.Append(BuyOffers);
```

```
    //sb.AppendLine(loginbutton);
```

```
    sb.AppendLine(logoutbutton);
```

```
    //sb.AppendLine(registerbutton);
```

```
}
```

```
else
```

```
{
```

```

        sb.AppendLine(LoggedInUser);
        sb.AppendLine(loginbutton);
        sb.AppendLine(registerbutton);

    }
    sb.AppendLine(line2);
    //objlog.Trace_Log(1, "mainmenu", "1", "common", sb.ToString());
    return sb.ToString();
}

public static string Generate_TopBar(string user, int RoleId, int DealerId, int UserId)
{
    // CustomerOrders co = new CustomerOrders();
    int dfcnt = new Products().GetSLItems(UserId,true);//
    co.GetCustomerDraftOrderNos(RoleId, DealerId, UserId);
    Flags FlgCnt = new Flags();
    int Saleoff = new Products().GetSLItems(UserId, false); ;
    ActionTasks ActionTsk = new ActionTasks();
    int CartItems = new Products().GetCartItems(UserId);
    var sb = new StringBuilder();
    sb.AppendLine("");
    string line1 = "", line2 = "";
    string BuyOffers = string.Empty;
    string SaleOffers = string.Empty;
    string MyCarts = string.Empty;
    string MyProducts = string.Empty;
    string LoggedInUser = string.Empty;
    string logoutbutton = string.Empty;
    string loginbutton = string.Empty;
    String Profile = string.Empty;
    string registerbutton = string.Empty;
    string DealerName;
    DealerName = new Dealers().GetDealerName(DealerId);

```

```

// user = user;//+ " (" + DealerName + ")";

line1 = "<ul class='pull-right'>";

BuyOffers = string.Format("<li style='border-right: none;'><a
href='BSOffers.aspx?pageid=23&stype=true' class='draft'><button type='button' class='btn
btn-primary'>WishList({0})</a></li>", dftcnt);

SaleOffers = string.Format("<li style='border-right: none;'><a
href='BSOffers.aspx?pageid=23&stype=false' class='draft'>Sale Offers({0})</a></li>",
Saleoff);

MyCarts = string.Format("<li style='border-right: none;'><a
href='MyCart.aspx?pageid=129' class='draft'><button type='button' class='btn btn-
primary'>My Cart({0})</a></li>", CartItems);

//MyProducts = string.Format("<li><a href='{0}' class='btn btn-primary'>My
Products</a></li>", "ProductSearchList.aspx?pageid=26");

MyProducts = string.Format("<li style='border-right: none;'><a
href='ProductSearchList.aspx?pageid=26'><button type='button' class='btn btn-primary'><i
class='fa fa-user'></i>&nbsp;&nbsp;&nbsp;My Products</button></a></li>");

LoggedInUser = string.Format("<li><a href='#' class='Test'>{0}</a></li>", "Welcome " +
user);

logoutbutton = string.Format("<li style='border-right: none;'><a
href='logout.aspx'><button type='button' class='btn btn-primary'><i class='fa fa-power-
off'></i>&nbsp;&nbsp;&nbsp;Log Out</button></a></li>");

// logoutbutton = string.Format("<li style='border-top: 5px;'><a class='btn btn-primary'
href='logout.aspx'><i class='fa fa-power-off'></i>&nbsp;&nbsp;&nbsp;log out</a></li>");

loginbutton = string.Format("<li style='border-right: none;'><a class='btn btn-primary'
href='login.aspx'><i class='fa fa-user'></i>&nbsp;&nbsp;&nbsp;login</a></li>");

registerbutton = string.Format("<li style='border-right: none;'><a class='btn btn-primary'
href='ManageUser.aspx?pageid=4'><i class='fa fa-
users'></i>&nbsp;&nbsp;&nbsp;Register</a></li>");

Profile = string.Format("<li style='border-right: none;'><a
href='UserDetail.aspx?pageid=25&ld="+UserId+"><button type='button' class='btn btn-
primary'><i class='fa fa-user'></i>&nbsp;&nbsp;&nbsp;Profile</button></a></li>");

line2 = "</ul>";

sb.AppendLine(line1);

//sb.AppendLine("
");

if(user!="Guest"){

sb.AppendLine(LoggedInUser);

sb.AppendLine(BuyOffers);

//sb.AppendLine(SaleOffers);

```



```

        sb.AppendLine(MyCarts);
        //sb.AppendLine(MyProducts);
    sb.AppendLine(Profile);
    sb.AppendLine(logoutbutton);
    }
    else
    {
        sb.AppendLine(LoggedInUser);
        sb.AppendLine(loginbutton);
        sb.AppendLine(registerbutton);

    }
    sb.AppendLine(line2);
    return sb.ToString();
}
#endregion
public static string New_Menu(int GroupId)
{
    BalMenus objclsper = new BalMenus();
    DataTable table = objclsper.GetNewMenus(GroupId);
    DataRow[] parentMenus = table.Select("parentid = 0");
    var sb = new StringBuilder();
    string unorderedList = Generate_MainMenu(parentMenus, table, sb, "0");
    //lft_panel.InnerHtml = unorderedList;
    return unorderedList;
}

public static string ShowMessage(string Msg)
{
    var sb = new StringBuilder();
    sb.Append("<a href='#' style='text-align: left;'><p>&nbsp;&nbsp;&nbsp;<i class='fa fa-check-circle'></i>&nbsp;&nbsp;&nbsp;" + Msg + "</p></a>");
    return sb.ToString();
}

```

```
}
```

```
public static void ShowMessageNew(string message, int type, bool bshow, Label lblname)
{
    string Msg = "";

    if (type == 1) { Msg = "<i class='fa fa-check-circle' style='color: #1fa67a;'></i> " +
message; } else { Msg = "<i class='fa fa-times-circle' style='color: #CD0E0E;'></i> " +
message; }

    lblname.Text = Msg;
    lblname.Visible = bshow;
    lblname.Parent.Visible = bshow;
}
```

```
public static void ShowMessageHtml(string message, int type, bool bshow,
HtmlGenericControl div)
{

    string Msg = "";

    if (bshow == true)
    {
        if (type == 1)
        {
            Msg = "<i class='fa fa-check-circle'></i> " + message;
            div.Attributes["class"] = "alert alert-success";
        }
        else
        {
            Msg = "<i class='fa fa-times-circle'></i> " + message;
            div.Attributes["class"] = "alert alert-error";
        }
    }
}
```

```

        div.Attributes.CssStyle.Add("display", "block;");
    }
    else
    {
        div.Attributes.CssStyle.Add("display", "none;");
    }

    div.Attributes["class"] = "alert-error";
    div.InnerHtml = Msg;
}

```

#region Permission Group

```

public static string GetLeftButtonsPermission(int pageid, int groupid)
{
    var sb = new StringBuilder();
    BalMenus objclsper = new BalMenus();
    DataSet ds = objclsper.GetLeftButtonsPermission(pageid, groupid);
    DataTable pagebtns = ds.Tables[0];

    if (ds != null)
    {
        if (ds.Tables[0].Rows.Count > 0)
        {

            foreach (DataRow drnew in pagebtns.Rows)
            {
                string pagebtnid = drnew["pagebtnid"].ToString();
                string buttonName = drnew["lbtnname"].ToString();
                string imgUrl = "";

                string pagebuttonid = "button-" + pagebtnid;

```

```

        if (drnew["lbtnimage"] != null || drnew["lbtnimage"] != "")
        {
            imgUrl = string.Format("<i class='{0}'></i>", drnew["lbtnimage"]);
        }
        else
        {
            imgUrl = "";
        }

        sb.Append(string.Format("@<label class='checkbox inline'><input
type='checkbox' id='{0}' value='{0}'>{2} {1}</label>", pagebuttonid, buttonName,
imgUrl));//;drnew["ID"].ToString();drnew["TEXT"].ToString();
    }
}
}
return sb.ToString();
}

public static string GetLeftButtonsProducts(int pageid, int groupid)
{
    var sb = new StringBuilder();
    BalMenus objclsper = new BalMenus();
    DataSet ds = objclsper.GetLeftButtonsProducts(pageid, groupid);
    DataTable pagebtns = ds.Tables[0];

    if (ds != null)
    {
        if (ds.Tables[0].Rows.Count > 0)
        {

            foreach (DataRow drnew in pagebtns.Rows)
            {
                string pagebtnid = drnew["pagebtnid"].ToString();
                string buttonName = drnew["lbtnname"].ToString();
            }
        }
    }
}

```

```

string imgUrl = "";

string pagebuttonid = "button-" + pagebtnid;
if (drnew["lbtnimage"] != null || drnew["lbtnimage"] != "")
{
    imgUrl = string.Format("<i class='{0}'></i>", drnew["lbtnimage"]);
}
else
{
    imgUrl = "";
}

sb.Append(string.Format("@<label class='checkbox inline'><input
type='checkbox' id='{0}' value='{0}'>{2} {1}</label>", pagebuttonid, buttonName,
imgUrl));;//drnew["ID"].ToString();drnew["TEXT"].ToString();
    }
}
}
return sb.ToString();
}
public static string Permission_Group(int GroupId, int type)
{
    //if (type == "NEW")
    //{
    //}
    //else if (type == "EDIT")
    //{
    //}
    BalMenus objclsper = new BalMenus();
    DataTable table = objclsper.Permission_Group(GroupId, type);
    DataRow[] parentMenus = table.Select("parentid = 0");

    var sb = new StringBuilder();

```

```

        string unorderedList = Generate_Permission_Group(parentMenus, table, sb, "0",
GroupId);
        //lft_panel.InnerHtml = unorderedList;
        return unorderedList;
    }
    /*Recursive method that populates the Dynamic Permissions*/
    public static string Products_Group(int GroupId, int type)
    {
        //if (type == "NEW")
        //{
        //}
        //else if (type == "EDIT")
        //{
        //}

        BalMenus objclsper = new BalMenus();
        DataTable table = objclsper.Products_Group(GroupId, type);
        DataRow[] parentMenus = table.Select("parentid = 0");

        var sb = new StringBuilder();
        string unorderedList = Generate_Products_Group(parentMenus, table, sb, "0",
GroupId);
        //lft_panel.InnerHtml = unorderedList;
        return unorderedList;
    }

    private static string Generate_Permission_Group(DataRow[] menu, DataTable table,
StringBuilder sb, string level, int groupid)
    {
        string styleclsname = string.Empty;
        string ConcatValue = string.Empty;
        if (menu.Length > 0)
        {
            foreach (DataRow dr in menu)
            {

```

```

string mapid = dr["MapId"].ToString();
string menuText = dr["PageName"].ToString();
string menuUrl = dr["PageUrl"].ToString();
string pageid = dr["pageId"].ToString();
string page_veiw = dr["view"].ToString();

string FullAccordion = "", accordion = "", accordionclose = "";

if (dr["MapId"] != null)
{
    string hrefid = "mapid-" + mapid;// "mapid" + mapid + "-pageid" + pageid;

    string LeftButtons = GetLeftButtonsPermission(Convert.ToInt32(pageid),
groupID);//string.Format("<label class='checkbox inline'><input type='checkbox'
id='inlineCheckbox1' value='option1'>Left Side Button For {0} page </label>", menuText);

    if (LeftButtons != "" && LeftButtons != "\n\r")
    {
        accordion = string.Format("<div class='accordion-group'><div
class='accordion-heading'><a class='accordion-toggle' data-toggle='collapse' data-
parent='#accordion2' href='#{0}'>{1}</a></div><div id='{0}' class='accordion-body collapse'
style='height: 0px;'><div class='accordion-inner'>", hrefid, menuText);

        accordionclose = "</div></div></div>";
    }
    else
    {
        accordion = string.Format("<div class='accordion-group'><div
class='accordion-heading'><a class='accordion-toggle' data-toggle='collapse' data-
parent='#accordion2' href='#{0}'>{1}</a></div><div id='{0}' class='accordion-body collapse'
style='height: 0px;'>", hrefid, menuText);

        accordionclose = "</div></div>";
    }

    if (menuUrl.ToLower() != "no-url")
    {

```

```

        FullAccordion = accordion + LeftButtons;
    }
    else
    {
        FullAccordion = accordion;
    }

    sb.Append(FullAccordion);
    string pid = dr["pageId"].ToString();
    DataRow[] subMenu = table.Select(String.Format("parentid = {0}", pid));
    if (subMenu.Length > 0)
    {
        var subMenuBuilder = new StringBuilder();
        sb.Append(Generate_Permission_Group(subMenu, table, subMenuBuilder,
        dr["level"].ToString(), groupid));

    }
    sb.Append(accordionclose);
}
}
}

return sb.ToString();
}

private static string Generate_Products_Group(DataRow[] menu, DataTable table,
StringBuilder sb, string level, int groupid)
{
    string styleclsname = string.Empty;
    string ConcatValue = string.Empty;
    if (menu.Length > 0)
    {
        foreach (DataRow dr in menu)

```



```

{
    string mapid = dr["pageId"].ToString();// dr["MapId"].ToString();
    string menuText = dr["PageName"].ToString();
    string menuUrl = dr["PageName"].ToString();
    string pageid = dr["pageId"].ToString();
    string page_veiw = "true";

    string FullAccordion = "", accordion = "", accordionclose = "";

    if (dr["pageid"] != null)
    {
        string hrefid = "mapid-" + pageid;// "mapid" + mapid + "-pageid" + pageid;

        string LeftButtons = GetLeftButtonsProducts(Convert.ToInt32(pageid),
groupid);//string.Format("<label class='checkbox inline'><input type='checkbox'
id='inlineCheckbox1' value='option1'>Left Side Button For {0} page </label>", menuText);

        if (LeftButtons != "" && LeftButtons != "\n\r")
        {
            accordion = string.Format("<div class='accordion-group'><div
class='accordion-heading'><a class='accordion-toggle' data-toggle='collapse' data-
parent='#accordion2' href='#{0}'>{1}</a></div><div id='{0}' class='accordion-body collapse'
style='height: 0px;'><div class='accordion-inner'>", hrefid, menuText);

            accordionclose = "</div></div></div>";
        }
        else
        {
            accordion = string.Format("<div class='accordion-group'><div
class='accordion-heading'><a class='accordion-toggle' data-toggle='collapse' data-
parent='#accordion2' href='#{0}'>{1}</a></div><div id='{0}' class='accordion-body collapse'
style='height: 0px;'>", hrefid, menuText);

            accordionclose = "</div></div>";
        }

        if (menuUrl.ToLower() != "no-url")

```

```

    {

        FullAccordion = accordion + LeftButtons;
    }
    else
    {
        FullAccordion = accordion;
    }

    sb.Append(FullAccordion);
    string pid = dr["pageId"].ToString();
    DataRow[] subMenu = table.Select(String.Format("parentid = {0}", pid));
    if (subMenu.Length > 0)
    {
        var subMenuBuilder = new StringBuilder();
        sb.Append(Generate_Products_Group(subMenu, table, subMenuBuilder,
dr["level"].ToString(), groupid));

    }
    sb.Append(accordionclose);
}
}
}

return sb.ToString();
}
/* -----Edit Permission Group----- */
public static string EditLeftButtonsPermission(int pageid, int groupid)
{
    var sb = new StringBuilder();
    BalMenus objclsper = new BalMenus();
    DataSet ds = objclsper.GetLeftButtonsPermission(pageid, groupid);
    DataTable pagebtns = ds.Tables[0];

```

```

if (ds != null)
{
    if (ds.Tables[0].Rows.Count > 0)
    {

        foreach (DataRow drnew in pagebtns.Rows)
        {
            string buttonId = "pagebtnid-" + drnew["pagebtnid"].ToString();
            string buttonName = drnew["lbtnname"].ToString();
            string imgUrl = "";
            if (drnew["lbtnimage"] != null || drnew["lbtnimage"] != "")
            {
                imgUrl = string.Format("<i class='{0}'></i>", drnew["lbtnimage"]);
            }
            else
            {
                imgUrl = "";
            }

            sb.Append(string.Format("@<label class='checkbox inline'><input
type='checkbox' id='{0}' value='{0}'>{2} {1}</label>", buttonId, buttonName,
imgUrl));//;drnew["ID"].ToString();drnew["TEXT"].ToString();
        }
    }
}
return sb.ToString();
}

public static string EditPermission_Group(int GroupId)
{
    BalMenus objclsper = new BalMenus();
    DataTable table = objclsper.Permission_Group(GroupId, 0);
    DataRow[] parentMenus = table.Select("parentid = 0");

```

```

var sb = new StringBuilder();

string unorderedList = EditGenerate_Permission_Group(parentMenus, table, sb, "0",
GroupId);

//lft_panel.InnerHtml = unorderedList;

return unorderedList;
}

/*Recursive method that populates the Dynamic Permissions*/

private static string EditGenerate_Permission_Group(DataRow[] menu, DataTable table,
StringBuilder sb, string level, int groupid)
{
string styleclsname = string.Empty;
string ConcatValue = string.Empty;
if (menu.Length > 0)
{
foreach (DataRow dr in menu)
{
string mapid = dr["MapId"].ToString();
string menuText = dr["PageName"].ToString();
string menuUrl = dr["PageUrl"].ToString();
string pageid = dr["pageId"].ToString();
string page_veiw = dr["view"].ToString();
string IsPageView = "";

if (page_veiw.ToLower() == "true")
{
IsPageView = "accordion-toggle collapsed";
}
else
{
IsPageView = "accordion-toggle";
}
}
}
}

```

```

string FullAccordion = "", accordion = "", accordionclose = "";

if (dr["MapId"] != null)
{
    string hrefid = "mapid" + mapid + "-pageid" + pageid;

    string LeftButtons = EditLeftButtonsPermission(Convert.ToInt32(pageid),
groupid);//string.Format("<label class='checkbox inline'><input type='checkbox'
id='inlineCheckbox1' value='option1'>Left Side Button For {0} page </label>", menuText);

    if (LeftButtons != "" && LeftButtons != "\n\r")
    {
        accordion = string.Format("<div class='accordion-group'><div
class='accordion-heading'><a class='{2}' data-toggle='collapse' data-parent='#accordion2'
href='#{0}'>{1}</a></div><div id='{0}' class='accordion-body collapse' style='height:
0px;'><div class='accordion-inner'>", hrefid, menuText, IsPageView);

        accordionclose = "</div></div></div>";
    }
    else
    {
        accordion = string.Format("<div class='accordion-group'><div
class='accordion-heading'><a class='{2}' data-toggle='collapse' data-parent='#accordion2'
href='#{0}'>{1}</a></div><div id='{0}' class='accordion-body collapse' style='height: 0px;'>",
hrefid, menuText, IsPageView);

        accordionclose = "</div></div>";
    }

    if (menuUrl.ToLower() != "no-url")
    {
        FullAccordion = accordion + LeftButtons;
    }
    else
    {
        FullAccordion = accordion;
    }
}

```

```

    }

    sb.Append(FullAccordion);
    string pid = dr["pageId"].ToString();
    DataRow[] subMenu = table.Select(String.Format("parentid = {0}", pid));
    if (subMenu.Length > 0)
    {
        var subMenuBuilder = new StringBuilder();
        sb.Append(EditGenerate_Permission_Group(subMenu, table,
subMenuBuilder, dr["level"].ToString(), groupid));

    }
    sb.Append(accordionclose);
}
}
}

return sb.ToString();
}

/* ----- */
#endregion

}

**** ExceptionUtility.cs ****
using System;
using System.IO;
using System.Web;

// Create our own utility for exceptions
public sealed class ExceptionUtility

```

```

{
// All methods are static, so this can be private
private ExceptionUtility()
{}

// Log an Exception
public static void LogException(Exception exc, string source)
{
// Include enterprise logic for logging exceptions
// Get the absolute path to the log file
string logFile = "App_Data/ErrorLog.txt";
logFile = HttpContext.Current.Server.MapPath(logFile);

// Open the log file for Append and write the log
StreamWriter sw = new StreamWriter(logFile, true);
sw.WriteLine("***** {0} *****", DateTime.Now);
if (exc.InnerException != null)
{
sw.Write("Inner Exception Type: ");
sw.WriteLine(exc.InnerException.GetType().ToString());
sw.Write("Inner Exception: ");
sw.WriteLine(exc.InnerException.Message);
sw.Write("Inner Source: ");
sw.WriteLine(exc.InnerException.Source);
if (exc.InnerException.StackTrace != null)
{
sw.WriteLine("Inner Stack Trace: ");
sw.WriteLine(exc.InnerException.StackTrace);
}
}
sw.Write("Exception Type: ");
sw.WriteLine(exc.GetType().ToString());
}
}

```

```

        sw.WriteLine("Exception: " + exc.Message);
        sw.WriteLine("Source: " + source);
        sw.WriteLine("Stack Trace: ");
        if (exc.StackTrace != null)
        {
            sw.WriteLine(exc.StackTrace);
            sw.WriteLine();
        }
        sw.Close();
    }

    // Notify System Operators about an exception
    public static void NotifySystemOps(Exception exc)
    {
        // Include code for notifying IT system operators
    }
}

***** ExceptionLogging *****
using System;
using context = System.Web.HttpContext;
using System.Configuration;
using System.Data.SqlClient;
using System.Data;

/// <summary>
/// Summary description for ExceptionLogging
/// article by Vithal Wadje

/// </summary>
public static class ExceptionLogging
{

```



```

private static String exepurl;
static SqlConnection con;
private static void connecttion()
{
    string constr =
ConfigurationManager.ConnectionStrings["DowConectingString"].ToString();
    con = new SqlConnection(constr);
    con.Open();
    if (con.State == ConnectionState.Closed) { con.Open(); }
}
public static void SendExcepToDB(Exception exdb)
{

    connection();
    exepurl = context.Current.Request.Url.ToString();
    SqlCommand com = new SqlCommand("ExceptionLoggingToDataBase", con);
    com.CommandType = CommandType.StoredProcedure;
    com.Parameters.AddWithValue("@ExceptionMsg", exdb.Message.ToString());
    com.Parameters.AddWithValue("@ExceptionType", exdb.GetType().Name.ToString());
    com.Parameters.AddWithValue("@ExceptionURL", exepurl);
    com.Parameters.AddWithValue("@ExceptionSource", exdb.StackTrace.ToString());
    com.ExecuteNonQuery();

}

}

**** FieldValidator ****
using System;
using System.Collections.Generic;
using System.Linq;

```

```
using System.Web;
using System.Web.UI.WebControls;

/// <summary>
/// Summary description for FieldValidator
/// </summary>
public class FieldValidator
{
    public FieldValidator()
    {
        //
        // TODO: Add constructor logic here
        //
    }

    public bool ValidateNumeric(TextBox txtbox)
    {
        bool blnResult = false;

        System.Text.RegularExpressions.Regex r = new
System.Text.RegularExpressions.Regex("[0-9]+$");
        blnResult = r.IsMatch(txtbox.Text);
        return blnResult;
    }

    public bool ValidateAlphanumeric(TextBox txtbox)
    {
        string str = txtbox.Text;
        str = str.Replace(" ", String.Empty);

        bool blnResult = false;

        System.Text.RegularExpressions.Regex r = new
System.Text.RegularExpressions.Regex("[a-zA-Z0-9]+$");
        blnResult = r.IsMatch(str);
        return blnResult;
    }
}
```

```

public bool ValidateAlpha(TextBox txtbox)
{
    string str = txtbox.Text;
    str = str.Replace(" ", String.Empty);

    bool blnResult = false;

    System.Text.RegularExpressions.Regex r = new
System.Text.RegularExpressions.Regex("[a-zA-Z]+$");

    blnResult = r.IsMatch(str);

    return blnResult;
}
}

```

***** FileUpDown.cs *****

```

using System;
using System.Collections.Generic;
using System.Data;
using System.IO;
using System.Linq;
using System.Web;
using System.Windows.Forms;
using DowBEntity;
using DowBAL;
using System.Web.UI.WebControls;
/// <summary>
/// Summary description for Downloader
/// </summary>
public class FileUpDown
{
    public FileUpDown()
    {
        //
        // TODO: Add constructor logic here
        //
    }
}

```

```
}
```

```
public static void SaveAttachment(SaveFileDialog objSfd, int Id)
```

```
{
```

```
    if (Id != 0)
```

```
    {
```

```
        EteNote notes = new EteNote();
```

```
        notes = new Notes().getNote(Id);
```

```
        byte[] objData;
```

```
        objData =
```

```
notes.AttachmentData.ToArray();//(byte[])objRow["AttachmentData"].ToString();
```

```
        string filename = notes.AttachmentFilename;
```

```
        string filenameWithoutPath = Path.GetFileName(filename);
```

```
        //File.WriteAllBytes(filename, objData);
```

```
        StreamFileToBrowser(filenameWithoutPath, objData);
```

```
    }
```

```
}
```

```
public static void StreamFileToBrowser(string sFileName, byte[] fileBytes)
```

```
{
```

```
    System.Web.HttpContext context = System.Web.HttpContext.Current;
```

```
    context.Response.Clear();
```

```
    context.Response.ClearHeaders();
```

```
    context.Response.ClearContent();
```

```
    context.Response.AppendHeader("content-length", fileBytes.Length.ToString());
```

```
    context.Response.ContentType = GetMimeTypeByFileName(sFileName);
```

```
    context.Response.AppendHeader("content-disposition", "attachment; filename=" +  
sFileName);
```

```

context.Response.BinaryWrite(fileBytes);
// use this instead of response.end to avoid thread aborted exception (known issue):
context.ApplicationInstance.CompleteRequest();
}
public static string GetMimeTypeByFileName(string sFileName)
{
    string sMime = "application/octet-stream";

    string sExtension = System.IO.Path.GetExtension(sFileName);
    if (!string.IsNullOrEmpty(sExtension))
    {
        sExtension = sExtension.Replace(".", "");
        sExtension = sExtension.ToLower();

        if (sExtension == "xls" || sExtension == "xlsx")
        {
            sMime = "application/ms-excel";
        }
        else if (sExtension == "doc" || sExtension == "docx")
        {
            sMime = "application/msword";
        }
        else if (sExtension == "ppt" || sExtension == "pptx")
        {
            sMime = "application/ms-powerpoint";
        }
        else if (sExtension == "rtf")
        {
            sMime = "application/rtf";
        }
        else if (sExtension == "zip")
        {

```

```
    sMime = "application/zip";
}
else if (sExtension == "mp3")
{
    sMime = "audio/mpeg";
}
else if (sExtension == "bmp")
{
    sMime = "image/bmp";
}
else if (sExtension == "gif")
{
    sMime = "image/gif";
}
else if (sExtension == "jpg" || sExtension == "jpeg")
{
    sMime = "image/jpeg";
}
else if (sExtension == "png")
{
    sMime = "image/png";
}
else if (sExtension == "tiff" || sExtension == "tif")
{
    sMime = "image/tiff";
}
else if (sExtension == "txt")
{
    sMime = "text/plain";
}
}
```

```

    return sMime;
}

public static EteFileUpDown GetUploadFile(FileUpload FUpload)
{
    EteFileUpDown df = new EteFileUpDown();
    // Read the file and convert it to Byte Array
    df.AttachmentFilename = FUpload.PostedFile.FileName; //string filePath =
FUpload.PostedFile.FileName;
    string filename = Path.GetFileName(FUpload.PostedFile.FileName);
    string ext = Path.GetExtension(filename);
    df.AttachmentType = ext;
    string contenttype = String.Empty;
    contenttype = GetMimeTypeByFileName(filename);

    if (contenttype != String.Empty)
    {

        Stream fs = FUpload.PostedFile.InputStream;
        df.AttachmentLength = (Int32)fs.Length;
        BinaryReader br = new BinaryReader(fs);
        Byte[] bytes = br.ReadBytes((Int32)fs.Length);
        df.AttachmentData = bytes;

    }
    else
    {
        return null;
    }
}

```

```

        return df;
    }
}
***** Navigations.cs *****
using DowBAL;
using DowBEntity;
using System;
using System.Collections.Generic;
using System.Data;
using System.Linq;
using System.Text;
using System.Text.RegularExpressions;
using System.Web;
using System.Web.UI.HtmlControls;
using System.Web.UI.WebControls;

/// <summary>
/// Summary description for Navigations
/// </summary>
public class Navigations
{

    //public bool IsLeftBar = false;
    public Navigations()
    {
        //
        // TODO: Add constructor logic here
        //
    }
    public static bool Generate_Color_sizeStocks(int UserId,int PsId, Table tbl)
    {
        DDLogTracer objlog = new DDLogTracer();

```



```

bool ret = false;
TableRow tr = new TableRow();
TableCell td = new TableCell();
List<EteSellerSizes> Lsizes = new Products().GetSellerSizes(UserId);
List<EteSellerColors> LColors = new Products().GetSellerColors(UserId);
TableHeaderRow thr = new TableHeaderRow();
td.Text = "Colors";
thr.Cells.Add(td);
td = new TableCell();
td.Text = " ";
thr.Cells.Add(td);

foreach (var siz in Lsizes)
{
    td = new TableCell();
    td.Text = "Size";
    thr.Cells.Add(td);
    td = new TableCell();
    td.Text = "Stock";
    thr.Cells.Add(td);
}
tbl.Rows.Add(thr);
foreach (var col in LColors)
{
    td = new TableCell();
    string img = "<img src=\"" + col.ImagePath + "\" runat='server' /> ";
    td.Text = img;
    tr.Cells.Add(td);
    td = new TableCell();
    td.Text = col.ColorDesc;
    tr.Cells.Add(td);
}

```

```

        foreach (var siz in Lsizes)
        {
            int stk = new Products().GetSellerStocksbyPsIdCol(PsId,
col.ColorId,siz.EpId,siz.EpOptId);
            td = new TableCell();

            td.Text = siz.EpOptDescr;

            tr.Cells.Add(td);
            TextBox tb = new TextBox();
            td = new TableCell();
            tb.Text = stk.ToString();
            tb.Width = 100;
            tb.ID = "TextBox_" + col.ColorId.ToString() + "_" + siz.EpOptId;
            td.Controls.Add(tb);
            tr.Cells.Add(td);
        }
        tbl.Rows.Add(tr);
        tr = new TableRow();
    }
    objlog.Trace_Log(1, "home", "1", "menu", "\n" + tbl.ToString());

    ret = true;
    return ret;
}
public static bool Generate_Color_sizes(int UserId, Table tbl)
{
    DDLogTracer objlog = new DDLogTracer();
    bool ret = false;
    TableRow tr = new TableRow();
    TableCell td = new TableCell();
    List<EteSellerSizes> Lsizes = new Products().GetSellerSizes(UserId);
    List<EteSellerColors> LColors = new Products().GetSellerColors(UserId);

```

```

TableHeaderRow thr = new TableHeaderRow();
td.Text = "Colors";
thr.Cells.Add(td);
td = new TableCell();
td.Text = " ";
thr.Cells.Add(td);

foreach (var siz in Lsizes)
{
    td = new TableCell();
    td.Text = "Size";
    thr.Cells.Add(td);
    td = new TableCell();
    td.Text = "Stock";
    thr.Cells.Add(td);
}
tbl.Rows.Add(thr);
foreach (var col in LColors)
{
    td = new TableCell();
    string img = "<img src=\"" + col.ImagePath + "\" runat='server' /> ";
    td.Text=img;
    tr.Cells.Add(td);
    td = new TableCell();

    td.Text = col.ColorDesc;
    tr.Cells.Add(td);
    foreach (var siz in Lsizes)
    {
        td = new TableCell();

```

```

        td.Text = siz.EpOptDescr;

        tr.Cells.Add(td);
        TextBox tb = new TextBox();
        td = new TableCell();
        tb.Text = "0";
        tb.Width = 100;
        tb.ID = "TextBox_" + col.ColorId.ToString() + "_" + siz.EpOptId;
        td.Controls.Add(tb);
        tr.Cells.Add(td);
    }
    tbl.Rows.Add(tr);
    tr = new TableRow();
}
objlog.Trace_Log(1, "home", "1", "menu", "\n" + tbl.ToString());

ret = true;
return ret;
}

public static bool Generate_LeftNaveBar(int pageid, int userid, int groupid, Table tbl,
EventHandler EventHandl)
{
    bool IsLeftBar = false;
    BalMenus objclsper = new BalMenus();
    DataTable table = objclsper.GetLeftButtons(pageid, 1);//groupid set to 1
    var sb = new StringBuilder();
    sb.AppendLine("");
    TableRow tr = new TableRow();
    TableCell td = new TableCell();
    IsLeftBar = false;
    if (table.Rows.Count > 0)
    {
        IsLeftBar = true;

```

```
bool Action = true;
bool WorkPlace = true;
bool Filterheading = true;

foreach (DataRow dr in table.Rows)
{

    string buttonid = dr["lbtnid"].ToString();
    string buttonname = dr["lbtnname"].ToString();
    string buttonImage = dr["lbtnimage"].ToString();
    string buttonurl = dr["lbtnurl"].ToString();
    string pageurl = dr["pageurl"].ToString();
    string buttontype = dr["lbtntype"].ToString();
    string buttonveiw = dr["show"].ToString();
    string buttoncode = dr["lbtncode"].ToString();
    string pagebuttonid = dr["pagebtnid"].ToString();

    if (buttontype == "1")
    {
        if (buttonveiw.ToLower() == "1")
        {

            tr = new TableRow();
            td = new TableCell();

            if (Action == true)
            {

                td.Text = "ACTION";
                td.CssClass = "customer";
```

```

        tr.Cells.Add(td);
        tbl.Rows.Add(tr);
        Action = false;
    }

    tr = new TableRow();
    td = new TableCell();
    td.CssClass = "span2 list";

    if (dr["maptype"] == "2")
    {

        td.Text = ButtonHTMLScript(dr); //string.Format("<div
id='div_permissions'><a href='#' onclick='RadXML();'><span><i class='{0}'></i>
{1}</span><a></div>", buttonImage, buttonname); //"<i class="" + buttonImage + ""></i> " +
buttonname;
    }

    else
    {
        LinkButton Ink = new LinkButton();
        Ink.ID = pagebuttonid;
        Ink.Text = "<i class="" + buttonImage + ""></i> " + buttonname;
        Ink.CommandName = pageid.ToString();
        Ink.CommandArgument = "buttoncode:" + buttoncode + "~buttonname:" +
buttonname + "~buttonurl:" + buttonurl + "~pageurl:" + pageurl;
        Ink.Click += new System.EventHandler(EventHandl);
        td.Controls.Add(Ink);
    }

    tr.Cells.Add(td);
    tbl.Rows.Add(tr);

```

```

    }
else
{

    tr = new TableRow();
    td = new TableCell();

    if (Action == true)
    {

        td.Text = "ACTION";
        td.CssClass = "customer";
        tr.Cells.Add(td);
        tbl.Rows.Add(tr);
        Action = false;
    }
    tr = new TableRow();
    td = new TableCell();
    td.CssClass = "disable";
    LinkButton Ink = new LinkButton();
    Ink.ID = pagebuttonid;
    Ink.Enabled = false;
    Ink.Text = "<i class=\"" + buttonImage + "\"></i> " + buttonname;
    Ink.CommandName = pageid.ToString();
    Ink.CommandArgument = "buttoncode:" + buttoncode + "~buttonname:" +
buttonname + "~buttonurl:" + buttonurl + "~pageurl:" + pageurl;
    //Ink.Click += new System.EventHandler(EventHandl[0]);
    td.Controls.Add(Ink);
    tr.Cells.Add(td);
    tbl.Rows.Add(tr);
}

```

```
}  
else if (buttontype == "2")  
{  
    if (buttonveiw.ToLower() == "1")  
    {  
  
        tr = new TableRow();  
        td = new TableCell();  
  
        if (WorkPlace == true)  
        {  
            if (Action == false)  
            {  
                td.Text = "<br/>";  
                tr.Cells.Add(td);  
                tbl.Rows.Add(tr);  
                tr = new TableRow();  
                td = new TableCell();  
            }  
            td.Text = "Product Category";// "WORK PLACE";  
            td.CssClass = "customer";  
            tr.Cells.Add(td);  
            tbl.Rows.Add(tr);  
            WorkPlace = false;  
        }  
  
        tr = new TableRow();  
        td = new TableCell();  
        td.CssClass = "span2 list";  
  
        if (dr["maptypes"] == "2")  
        {
```



```

        td.Text = ButtonHTMLScript(dr); //string.Format("<div
id='div_permissions'><a href='#' onclick='RadXML();'><span><i class='{0}'></i>
{1}</span><a></div>", buttonImage, buttonname); //"<i class="" + buttonImage + ""></i> " +
buttonname;
    }

    else
    {
        LinkButton Ink = new LinkButton();
        Ink.ID = pagebuttonid;
        Ink.Text = "<i class="" + buttonImage + ""></i> " + buttonname;
        Ink.CommandName = pageid.ToString();
        Ink.CommandArgument = "buttoncode:" + buttoncode + "~buttonname:" +
buttonname + "~buttonurl:" + buttonurl + "~pageurl:" + pageurl;
        Ink.Click += new System.EventHandler(EventHandl);
        td.Controls.Add(Ink);
    }
    tr.Cells.Add(td);
    tbl.Rows.Add(tr);

}

else
{

    tr = new TableRow();
    td = new TableCell();

    if (WorkPlace == true)
    {
        if (Action == false)
        {
            td.Text = "<br/>";

```

```

        tr.Cells.Add(td);
        tbl.Rows.Add(tr);
        tr = new TableRow();
        td = new TableCell();
    }
    td.Text = "WORK PLACE";
    td.CssClass = "customer";
    tr.Cells.Add(td);
    tbl.Rows.Add(tr);
    Workplace = false;
}
tr = new TableRow();
td = new TableCell();
td.CssClass = "disable";
LinkButton Ink = new LinkButton();
Ink.ID = pagebuttonid;
Ink.Enabled = false;
Ink.Text = "<i class=\"" + buttonImage + "></i> " + buttonname;
Ink.CommandName = pageid.ToString();
Ink.CommandArgument = "buttoncode:" + buttoncode + "~buttonname:" +
buttonname + "~buttonurl:" + buttonurl + "~pageurl:" + pageurl;
Ink.Click += new System.EventHandler(EventHandl);
td.Controls.Add(Ink);
tr.Cells.Add(td);
tbl.Rows.Add(tr);
}
}
else if (buttontype == "3" || buttontype == "4")
{

    tr = new TableRow();
    td = new TableCell();

```

```
if (Filterheading == true)
{
    if (Action == false)
    {
        td.Text = "<br/>";
        tr.Cells.Add(td);
        tbl.Rows.Add(tr);
        tr = new TableRow();
        td = new TableCell();
    }
    td.Text = "FILTER";
    td.CssClass = "customer"; //customer
    tr.Cells.Add(td);
    tbl.Rows.Add(tr);
    Filterheading = false;
}
```

```
tr = new TableRow();
td = new TableCell();
td.Text = buttonname + ":<br/>";
td.CssClass = "dllheadings";
tr.Cells.Add(td);
tbl.Rows.Add(tr);
```

```
td = new TableCell();
tr = new TableRow();
tr.CssClass = "dlllist";
td.CssClass = "span2 list";
```

```
DropDownList ddl = new DropDownList();
ddl.SelectedIndexChanged += new System.EventHandler(EventHandl);
```

```

ddl.AutoPostBack = true;
ddl.EnableViewState = true;

//Don't remove this, this required to identify the
//Dropdown in CustSerachList
ddl.Attributes.Add("LISTTYPE", buttonname);

td.Controls.Add(ddl);
tr.Cells.Add(td);

BindDynamic_DDL(ddl, Convert.ToInt32(buttonid), userid,
Convert.ToInt32(buttontype));

if (buttonveiw.ToLower() == "0")
{
    ddl.Enabled = false;
    td.CssClass = "span2 list";
}
tbl.Rows.Add(tr);
}
}
}
return IsLeftBar;
}

public static string StatusWiseRedirect(int statusid, string param)
{

switch (statusid.ToString())

```

```

{
    case "1":
        return ("ManageOrder.aspx?pageid=177&PageType=Edit&" + param);

    case "2":
        return ("ManageOrder.aspx?pageid=177&PageType=Edit&" + param);

    case "3":
        return ("ManageOrder.aspx?pageid=177&PageType=Edit&" + param);

    case "17":
        return ("ManageOrder.aspx?pageid=179&PageType=Edit&" + param);

    case "9":
        return ("ManageOrder.aspx?pageid=204&PageType=Edit&" + param); //179

    case "11":
        return ("ManageOrder.aspx?pageid=204&PageType=Edit&" + param);

    default:
        return ("ManageOrder.aspx?pageid=179&PageType=Edit&" + param);

}
}
private static string ButtonHTMLScript(DataRow dr)
{
    string buttonid = dr["lbtnid"].ToString();
    string buttonname = dr["lbtnname"].ToString();
    string buttonImage = dr["lbtnimage"].ToString();
    string buttonurl = dr["lbtnurl"].ToString();
    string pageurl = dr["pageurl"].ToString();
    string buttontype = dr["lbtntype"].ToString();
}

```

```

string buttonveiw = dr["show"].ToString();
string buttoncode = dr["lbtncode"].ToString();
string pagebuttonid = dr["pagebtnid"].ToString();
string mapdescription = dr["mapdescription"].ToString();
string result = "";

    result = string.Format(mapdescription, buttonImage, buttonname); // "<i class=" +
buttonImage + "></i> " + buttonname;

    return result;
}

public static void BindDynamic_DDL(DropDownList SrcDdl, int lbtnid, int loginid, int
lbtntype)
{

    BalMenus objclsper = new BalMenus();
    DataSet ds = objclsper.BindDynamic_DDL(lbtnid, loginid, lbtntype);
    if (ds != null)
    {
        if (ds.Tables[0].Rows.Count > 0)
        {
            SrcDdl.DataSource = ds;
            SrcDdl.DataValueField = "ID";
            SrcDdl.DataTextField = "TEXT";
            SrcDdl.DataBind();

            // By Syed on 2015-Jan-01 to display "All" option in all dropdowns
            SrcDdl.Items.Insert(0, new ListItem("All", "0"));
            SrcDdl.SelectedIndex = 0;

        }
    }
}

```

```
public static DataSet GetPageButtonDetails(int pagebtnid)
{
    BalMenus objclsper = new BalMenus();
    DataSet ds = objclsper.GetPageButtonDetails(pagebtnid);
    return ds;
}
```

```
}
```

```
***** DDLogTracer.cs *****
```

```
using Microsoft.VisualBasic;
using System;
using System.Collections;
using System.Collections.Generic;
using System.Configuration;
using System.Data;
using System.Diagnostics;
using System.IO;
using System.Web;
```

```
//namespace LogTracer
//{{
public class DDLogTracer
{
    //public enum LogLevel
    //{
    //    INFO = 0,
    //    DEBUG = 1,
```

```

// DB = 2,
// EXC = 3,
// CRIT = 4
//}
public string TraceThreadId;
private static string logDirectory;
private static StreamWriter sw;
System.Web.HttpContext context = System.Web.HttpContext.Current;

//Private Sub ssNew()
// Try
//     logDirectory = (ConfigurationManager.AppSettings("strlogpath").ToString +
"logconfig.err")
//     If Not File.Exists(logDirectory) Then
//         Dim fs As FileStream = File.Create(logDirectory)
//         fs.Close()
//     End If
//     sw = File.AppendText(logDirectory)
// Catch exc As Exception

// End Try
//End Sub
public static void StreamFileToDisk(string sFileName, byte[] fileBytes)
{
    System.Web.HttpContext context = System.Web.HttpContext.Current;
    context.Response.Clear();

    context.Response.ClearHeaders();
    context.Response.ClearContent();
    context.Response.AppendHeader("content-length", fileBytes.Length.ToString());
    context.Response.ContentType = GetMimeTypeByFileName(sFileName);
    context.Response.AppendHeader("content-disposition", "attachment; filename=" +
sFileName);

```



```
context.Response.BinaryWrite(fileBytes);
// use this instead of response.end to avoid thread aborted exception (known issue):
context.ApplicationInstance.CompleteRequest();
}
public static string GetMimeTypeByFileName(string sFileName)
{
    string sMime = "application/octet-stream";

    string sExtension = System.IO.Path.GetExtension(sFileName);
    if (!string.IsNullOrEmpty(sExtension))
    {
        sExtension = sExtension.Replace(".", "");
        sExtension = sExtension.ToLower();

        if (sExtension == "xls" || sExtension == "xlsx")
        {
            sMime = "application/ms-excel";
        }
        else if (sExtension == "doc" || sExtension == "docx")
        {
            sMime = "application/msword";
        }
        else if (sExtension == "ppt" || sExtension == "pptx")
        {
            sMime = "application/ms-powerpoint";
        }
        else if (sExtension == "rtf")
        {
            sMime = "application/rtf";
        }
        else if (sExtension == "zip")
        {
```

```
    sMime = "application/zip";
}
else if (sExtension == "mp3")
{
    sMime = "audio/mpeg";
}
else if (sExtension == "bmp")
{
    sMime = "image/bmp";
}
else if (sExtension == "gif")
{
    sMime = "image/gif";
}
else if (sExtension == "jpg" || sExtension == "jpeg")
{
    sMime = "image/jpeg";
}
else if (sExtension == "png")
{
    sMime = "image/png";
}
else if (sExtension == "tiff" || sExtension == "tif")
{
    sMime = "image/tiff";
}
else if (sExtension == "txt")
{
    sMime = "text/plain";
}
}
```

```
    return sMime;
}

private string GetLotType(int TL_Type)
{
    string TR_Num = null;

    if ((TL_Type == 0))
    {
        TR_Num = "INFO";
    }
    else if ((TL_Type == 1))
    {
        TR_Num = "DEBUG";
    }
    else if ((TL_Type == 2))
    {
        TR_Num = "DATABASE";
    }
    else if ((TL_Type == 3))
    {
        TR_Num = "DETAILED";
    }
    else if ((TL_Type == 4))
    {
        TR_Num = "CRITICAL";
    }
    else if ((TL_Type == 5))
    {
```

```

        TR_Num = "EXCEPTION";
    }
    else
    {
        TR_Num = "ERROR";
    }
    return TR_Num;
}

public void Trace_Log(int TL_Type, string ModuleName, string Page, string Block,
string Log,string path="")
{
    if ((ConfigurationSettings.AppSettings["level" + TL_Type.ToString()] == "1"))
    {
        string Log_Write_String = string.Format("{0} - [{1}]:[{2}]:[{3}]:[{4}]:{5}",
DateTime.Now.ToString("yyyy-MM-dd HH:mm:ss.ffffff"), GetLotType(TL_Type),
ModuleName, Page, Block, Log);
        WriteLogFile(Log_Write_String,path);
    }
}

public void Trace_Error(int TL_Type, string ModuleName, string Page, string Block,
string ExceptionMessage,string path="")
{
    if ((ConfigurationSettings.AppSettings["level5"] == "1"))
    {
        string Error_Write_String = string.Format("{0} - [{1}]:[{2}]:[{3}]:[{4}]:{5}",
DateTime.Now.ToString("yyyy-MM-dd HH:mm:ss.ffffff"), GetLotType(TL_Type),
ModuleName, Page, Block, ExceptionMessage);
        WriteErrorFile(Error_Write_String,path);
    }
}

private void WriteLogFile(string Log_Content,string path)
{

```

```

string Logpath = ConfigurationSettings.AppSettings["strlogpath"].ToString();
string filepath = ""; // (ConfigurationSettings.AppSettings["strlogpath"] +
(DateTime.Now.ToString("yyyy\\\\" + "MM\\\\" + "dd") + "\\\\"));

if (Logpath == "" | Logpath.Length <= 0)
{
    filepath =path+ ("\\Logs\\" + (DateTime.Now.ToString("yyyy\\\\" + "MM\\\\" + "dd") +
"\\\\"));
}
else
{
    filepath = (Logpath );//+ (DateTime.Now.ToString("yyyy\\\\" + "MM\\\\" + "dd") + "\\\\"));
}
filepath = context.Server.MapPath(filepath);
if (!Directory.Exists(filepath))
{
    Directory.CreateDirectory(filepath);
}

string strLogFileName = (DateTime.Now.Hour.ToString("D2")) + "_WS_" +
TraceThreadId + "." + (ConfigurationSettings.AppSettings["strLogFileName"]);
strLogFileName = filepath + "\\" + strLogFileName;
StreamWriter sw = null;

if ((!File.Exists(strLogFileName)))
{
    sw = File.CreateText(strLogFileName);
}
else
{
    sw = File.AppendText(strLogFileName);
}

sw.WriteLine(Log_Content);

```

```

        sw.Close();

    }

    private void WriteErrorFile(string Error_Content,string path)
    {
        string Logpath = ConfigurationSettings.AppSettings["strlogpath"].ToString();
        string filepath = ""; // (ConfigurationSettings.AppSettings["strlogpath"] +
(DateTime.Now.ToString("yyyy\\\\" + "MM\\\\" + "dd") + "\\");

        if (Logpath == "" | Logpath.Length <= 0)
        {
            filepath = path+"\\Log\\"; // + (DateTime.Now.ToString("yyyy\\\\" + "MM\\\\" + "dd") + "\\");
        }
        else
        {
            filepath = (Logpath ); // + (DateTime.Now.ToString("yyyy\\\\" + "MM\\\\" + "dd") + "\\");
        }
        filepath = context.Server.MapPath(filepath);

        if (!Directory.Exists(filepath))
        {
            Directory.CreateDirectory(filepath);
        }

        string strErrorFileName = (DateTime.Now.Day.ToString("D2")) +
(DateTime.Now.Month.ToString("D2")) + "_WS_" + TraceThreadid + "." +
(ConfigurationSettings.AppSettings["strErrorFileName"]);

        strErrorFileName = filepath + "\\" + strErrorFileName;

        StreamWriter sw = null;

        if ((!File.Exists(strErrorFileName)))
        {
            sw = File.CreateText(strErrorFileName);

```

```

    }
    else
    {
        sw = File.AppendText(strErrorFileName);
    }
    sw.WriteLine(Error_Content);
    sw.Close();
}
private long checkfilesizefunction(long filesizetocheck)
{

    try
    {
        long filesize = filesizetocheck;

        long getfilesizefromconfig =
long.Parse(ConfigurationSettings.AppSettings["filesize"]);

        string getfilesizeextension =
ConfigurationSettings.AppSettings["FileSizeFormat"].ToString();
        if ((getfilesizeextension.ToUpper() == "KB"))
        {
            long chkfilesize = (filesize / 1024);
            return chkfilesize;
        }
        else if ((getfilesizeextension.ToUpper() == "MB"))
        {
            long chkfilesize = (filesize / (1024 * 1024));
            return chkfilesize;
        }
        else if ((getfilesizeextension.ToUpper() == "GB"))
        {
            long chkfilesize = (filesize / (1024 * (1024 * 1024)));
            return chkfilesize;
        }
    }
}

```

```
        return filesize;
    }
    catch (Exception ex)
    {
        return 0;
    }
}
}
//}
```

This concludes common files in the application

Chapter 7

Develop Master Windows to inherit during development

Each window will have two files, one .aspx and its corresponding .cs file

```
*****ShoppingMasterPage.master*****
```

```
<%@ Master Language="C#" AutoEventWireup="true"
CodeFile="ShoppingMasterPage.master.cs" Inherits="ShoppingMasterPage" %>
<!DOCTYPE html>
<!--[if !IE]><!--> <html lang="en"> <!--<![endif]-->
<head>
    <title>DateOrWed Shopping</title>
    <!-- META -->
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-
fit=yes">
    <meta name="description" content="Fstore Bootstrap HTML5 eCommerce Template">
    <meta name="author" content="Craftdzine">

    <!-- FAVICON -->
<link rel="shortcut icon" href="shopping/img/favicon.png"/>

    <!-- BOOTSTRAP -->
<link rel="stylesheet" href="shopping/css/bootstrap.min.css"/>
<link rel="stylesheet" href="shopping/css/font-awesome.min.css"/>
<link rel="stylesheet" href="font/fontawesome/css/all.css"/>
<link rel="stylesheet" href="shopping/css/style.css"/>
<link rel="stylesheet" href="shopping/css/animate.css"/>
<link rel="stylesheet" href="shopping/css/magnific-popup.css"/>
<link rel="stylesheet" href="plugins/FlexSlider/jquery.flexslider.css"/>
<link rel="stylesheet" href="plugins/owl-carousel/owl.carousel.css"/>
<link rel="stylesheet" href="plugins/owl-carousel/owl.theme.css"/>
<link rel="stylesheet" type="text/css" href="shopping/css/slick.css"/>
<link rel="stylesheet" type="text/css" href="shopping/css/slick-theme.css"/>
<link rel="stylesheet" href="css/style.css"/>
```

```
<!-- WEB FONTS -->

<link href='http://fonts.googleapis.com/css?family=Open+Sans:300,400,600,700'
rel='stylesheet' type='text/css'>

<link href="http://fonts.googleapis.com/css?family=Raleway:500,800" rel="stylesheet"
type="text/css" media="all" />

<!-- JQUERY LIBS -->

<script type="text/javascript" src="shopping/js/jquery.min.js"></script>

<!-- BOOTSTRAP -->

<script type="text/javascript" src="shopping/js/bootstrap.min.js"></script>

<!-- OWL CAROUSEL -->

<script type="text/javascript" src="plugins/owl-carousel/owl.carousel.min.js"></script>

<!-- FLEXSLIDER -->

<script type="text/javascript" src="plugins/FlexSlider/jquery.flexslider-min.js"></script>

<!-- MAGNIFIC POPUP-->

<script type="text/javascript" src="shopping/js/jquery.magnific-popup.min.js"></script>

<!-- MAGNIFIC POPUP -->

<script type="text/javascript" src="shopping/js/magnific-popup.min.js"></script>

<!-- slick-->

<script type="text/javascript" src="shopping/js/slick.js"></script>

<!-- slick-slider-->

<script type="text/javascript" src="shopping/js/slick-slider.js"></script>

<!-- plugins-->

<script type="text/javascript" src="shopping/js/plugins.min.html"></script>

<!-- CUSTOM-->

<script type="text/javascript" src="shopping/js/custom.js"></script>

<!--[if lt IE 9]>

<script src="respond.js"></script>

<script src="html5shiv.js"></script>

<![endif]-->

<asp:ContentPlaceHolder ID="CPH_MainMaster_Header" runat="server">
```

```
</asp:ContentPlaceholder>
</head>
<body>
<form id="form1" runat="server" >
    <asp:ScriptManager ID="ScriptManager1" runat="server">
    </asp:ScriptManager>

<!-- Site Wrapper -->
<div class="wrapper">

<!-- Header -->
<div class="header header-static">
    <!-- Topbar -->
    <div class="topbar">

        <div class="container">
            <div class="row">
                <div class="col-sm-6">
                    <!-- Topbar Navigation -->
                    <ul class="topbar-left">
                        <li>
                            <a>admin@dateorwed.com</a>
                        </li>
                        <li>
                            <a>+91 - 7893642582</a>
                        </li>

                    </ul><!--topbar-left-->
                </div>
                <div id="account" class="col-sm-6" runat="server">
                    <ul class="list-inline topbar-right pull-right">
                        <li><a href="#">Welcome Guest</a></li>
```

```
                <li><a href="MyWishList.aspx"><i class="fas fa-lock"></i>Wishlist
(0)</a></li>
```

```
                <li><a href="MLB_Login.aspx">Login</a> | <a
href="ManageUser.aspx">Register</a></li>
```

```
            </ul>
```

```
        </div>
```

```
</div>
```

```
<div class="collapse navbar-collapse" id="navbarSupportedContent-4">
```

```
    <div class="search-head text-center">
```

```
        <div class="row">
```

```
            <div class="col-lg-12">
```

```
                <div class="row">
```

```
                    <div class="col-lg-2 col-md-2 col-sm-12 p-0 bg-white mt0">
```

```
                        <a class="navbar-brand" href="Home.aspx">
```

```
                            
```

```
                        </a>
```

```
                    </div>
```

```
                <div class="col-lg-2 col-md-2 col-sm-12 p-0">
```

```
<asp:DropDownList id="ddl_location" runat="server" Width="190px" Height="50px">
```

```
    <asp:ListItem Value="0">--- Select Store ---</asp:ListItem>
```

```
    <asp:ListItem Value="1"> Bridal Fashion </asp:ListItem>
```

```
    <asp:ListItem Value="2"> Groom Shopping </asp:ListItem>
```

```
    <asp:ListItem Value="3"> Kids Shopping </asp:ListItem>
```

```
    <asp:ListItem Value="4"> Gadgets </asp:ListItem>
```

```
</asp:DropDownList>
```

```
    </div>
```

```
        <div class="col-lg-5 col-md-5 col-sm-12 p-0">
```

```
<asp:TextBox ID="txt_SearchString" runat="server" class="form-control search-slt"
Width="570px" Height="37px"
onkeydown="javascript:TextName_OnKeyDown(event)"></asp:TextBox>

    </div>

    <div class="col-lg-2 col-md-2 col-sm-12 p-0">

        <button id="Button1" type="button" class="btn btn-secondary wrn-btn
mt-0 ml-0" onclick="JavaScript:GetURLSring1()" ><i class='fa fa-
search'></i>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;Search</button>

    </div>

</div>

</div>

</div>

<script type="text/javascript">
    function GetURLSring1() {
//        alert('GetURLSring-1');
        var fullURL = (location.pathname + location.search).substr(1);

        var loc=document.getElementById("<%=ddl_location.ClientID%>").value;
        var EnCodedFullURL = encodeURIComponent(fullURL);
        var SearchString =
document.getElementById("<%=txt_SearchString.ClientID%>").value;

        var url = 'ShoppingList.aspx?SrchStrng=' + encodeURIComponent(SearchString) +
'&cat=' + encodeURIComponent(loc);

        //$(location).attr('href', url);
        window.location.href = url;
        event.preventDefault();
    }
</script>
    </div>

</div>

</div><!--container-->

</div>
```

```

<!-- End Topbar -->

<!-- Navbar -->
<div class="navbar navbar-default mega-menu" role="navigation">
  <div class="container-fluid">
    <div class="navbar-header">
      <button type="button" class="navbar-toggle" data-toggle="collapse" data-
target=".cd-navbar-collapse">
        <span class="sr-only">Toggle navigation</span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
      </button>
      <a class="navbar-brand" href="Home.aspx">
        
      </a--%>
    </div>
  </div>
  <!-- Shopping Cart -->
  <div id="shopcart" class="shop-badge badge-icons pull-right" runat="server">
    <a href="#"><i class="fa fa-shopping-bag"></i></a>
    <span class="badge badge-sea rounded-x">
      <asp:Label ID="cartnos" runat="server"
Text="Label"></asp:Label></span>
    <div id="cart" runat="server" class="badge-open">

    </div>
  </div>
  <!-- End Shopping Cart -->
<%--menu strts--%>

```

```
<div id="menu" class="collapse navbar-collapse cd-navbar-collapse"
runat="server">
```

```
<!-- Nav Menu -->
```

```
<%--<div>
```

```
<ul class='nav navbar-nav'>
```

```
<li class='dropdown mega-menu-fullwidth'>
```

```
<a class='dropdown-toggle' href='javascript:void(0);' data-hover='dropdown' data-
toggle='dropdown'>Bride Fashion</a>
```

```
<ul class='dropdown-menu'>
```

```
<li>
```

```
<div class='mega-menu-content'>
```

```
<div class='container'>
```

```
<div class='row'>
```

```
<div class='col-md-2 col-sm-6'>
```

```
<h3 class='mega-menu-heading'>Clothing</h3>
```

```
<ul class='list-unstyled style-list'>
```

```
<li><a href='/ShoppingList.aspx?pid=42'>Blazers</a></li>
```

```
<li><a href='/ShoppingList.aspx?pid=43'>Blouses</a></li>
```

```
<li><a href='/ShoppingList.aspx?pid=44'>Coats</a></li>
```

```
<li><a href='/ShoppingList.aspx?pid=45'>Coverups</a></li>
```

```
<li><a href='/ShoppingList.aspx?pid=46'>Fashion Hudies</a></li>
```

```
<li><a href='/ShoppingList.aspx?pid=47'>Hosiery</a></li>
```

```
<li><a href='/ShoppingList.aspx?pid=48'>Indo western</a></li>
```

```
<li><a href='/ShoppingList.aspx?pid=49'>Jackets</a></li>
```

```
<li><a href='/ShoppingList.aspx?pid=50'>Jeans</a></li>
```

```
<li><a href='/ShoppingList.aspx?pid=51'>Jeggins</a></li>
```

```
<li><a href='/ShoppingList.aspx?pid=52'>Jumpsuits</a></li>
```

```
<li><a href='/ShoppingList.aspx?pid=53'>Kurtaset</a></li>
```

```
<li><a href='/ShoppingList.aspx?pid=54'>Kurtees</a></li>
```

```
<li><a href='/ShoppingList.aspx?pid=55'>Leggins</a></li>
```

```
<li><a href='/ShoppingList.aspx?pid=56'>Lehengas</a></li>
```

```
<li><a href='/ShoppingList.aspx?pid=57'>Lingerie</a></li>
```

```
</ul>
```

</div>

<div class='col-md-2 col-sm-6'>

<ul class='list-unstyled style-list'>

Mix and Match

Overalls

Pants

Rompers

Salwar Khameez

Sarees

Shorts

Skirts

Socks

Suiting

Sweaters

SweatShirts

Swim Suits

Tees

Tops

Vests

</div>

<div class='col-md-2 col-sm-6'>

<h3 class='mega-menu-heading'>Handbags & Wallets</h3>

<ul class='list-unstyled style-list'>

Clutches & Evening Bags

Crossbody Bags

Fashion Backpacks

Hobo Bags

Satchels

Shoulder Bags

Top-Handle Bags

Totes


```
<li><a href='/ShoppingList.aspx?pid=82'>Wallets</a></li>
<li><a href='/ShoppingList.aspx?pid=83'>Wristlets</a></li>
</ul>
```

```
</div>
```

```
<div class='col-md-2 col-sm-6'>
```

```
<ul class='list-unstyled style-list'>
```

```
<h3 class='mega-menu-heading'>Jewelry</h3>
```

```
<ul class='list-unstyled style-list'>
```

```
<li><a href='/ShoppingList.aspx?pid=84'>Anklets</a></li>
```

```
<li><a href='/ShoppingList.aspx?pid=85'>Bracelets</a></li>
```

```
<li><a href='/ShoppingList.aspx?pid=86'>Brooches & Pins</a></li>
```

```
<li><a href='/ShoppingList.aspx?pid=87'>Earrings</a></li>
```

```
<li><a href='/ShoppingList.aspx?pid=88'>Jewelry Sets</a></li>
```

```
<li><a href='/ShoppingList.aspx?pid=89'>Necklaces</a></li>
```

```
<li><a href='/ShoppingList.aspx?pid=90'>Rings</a></li>
```

```
<li><a href='/ShoppingList.aspx?pid=91'>Wedding & Engagement</a></li>
```

```
<li><a href='/ShoppingList.aspx?pid=92'>Smart Jewelry</a></li>
```

```
<li><a href='/ShoppingList.aspx?pid=93'>Body Jewelry</a></li>
```

```
</ul>
```

```
</div>
```

```
<div class='col-md-2 col-sm-6'>
```

```
<ul class='list-unstyled style-list'>
```

```
<h3 class='mega-menu-heading'>shoes</h3>
```

```
<ul class='list-unstyled style-list'>
```

```
<li><a href='/ShoppingList.aspx?pid=94'>Athletic</a></li>
```

```
<li><a href='/ShoppingList.aspx?pid=95'>Boots</a></li>
```

```
<li><a href='/ShoppingList.aspx?pid=96'>Fashion Sneakers</a></li>
```

```
<li><a href='/ShoppingList.aspx?pid=97'>Flats</a></li>
```

```
<li><a href='/ShoppingList.aspx?pid=98'>Loafers & Slip-Ons</a></li>
```

```
<li><a href='/ShoppingList.aspx?pid=99'>Mules & Clogs</a></li>
```

```
<li><a href='/ShoppingList.aspx?pid=100'>Outdoor</a></li>
```

```
<li><a href='/ShoppingList.aspx?pid=101'>Oxfords</a></li>
```

Pumps
Sandals
Slippers
Work & Safety

</div>

<div class='col-md-2 col-sm-6'>
<ul class='list-unstyled style-list'>
<h3 class='mega-menu-heading'>watches</h3>
<ul class='list-unstyled style-list'>
Wrist Watches
Watch Bands
Pocket Watches
Smartwatches
Certified Pre-Owned

</div>

<div class='col-md-2 col-sm-6'>
<ul class='list-unstyled style-list'>
<h3 class='mega-menu-heading'>Accessories</h3>
<ul class='list-unstyled style-list'>
Belts
Sunglasses & Eyewear Accessories
Scarves & Wraps
Gloves & Mittens
Hats & Caps
Earmuffs
Handbag Accessories
Keyrings & Keychains
Wallets, Card Cases & Money Organizers
Special Occasion Accessories

</div>

```
</div>
<div class='col-md-2 col-sm-6'>
<ul class='list-unstyled style-list'>
</div>
</div>
</div>
</div>
</li>

</ul>
<li class='dropdown mega-menu-fullwidth'>
<a class='dropdown-toggle' href='javascript:void(0);' data-hover='dropdown' data-
toggle='dropdown'>Women Fashion</a>
<ul class='dropdown-menu'>
<li>
<div class='mega-menu-content'>
<div class='container'>
<div class='row'>
<div class='col-md-2 col-sm-6'>
<h3 class='mega-menu-heading'>Clothing</h3>
<ul class='list-unstyled style-list'>
<li><a href='/ShoppingList.aspx?pid=121'>Blazers</a></li>
<li><a href='/ShoppingList.aspx?pid=122'>Blouses</a></li>
<li><a href='/ShoppingList.aspx?pid=123'>Coats</a></li>
<li><a href='/ShoppingList.aspx?pid=124'>Coverups</a></li>
<li><a href='/ShoppingList.aspx?pid=125'>Fashion Hudies</a></li>
<li><a href='/ShoppingList.aspx?pid=126'>Hosiery</a></li>
<li><a href='/ShoppingList.aspx?pid=127'>Indo western</a></li>
<li><a href='/ShoppingList.aspx?pid=128'>Jackets</a></li>
<li><a href='/ShoppingList.aspx?pid=129'>Jeans</a></li>
<li><a href='/ShoppingList.aspx?pid=130'>Jeggins</a></li>
<li><a href='/ShoppingList.aspx?pid=131'>Jumpsuits</a></li>
<li><a href='/ShoppingList.aspx?pid=132'>Kurtaset</a></li>
```

```
<li><a href='/ShoppingList.aspx?pid=133'>Kurtees</a></li>
<li><a href='/ShoppingList.aspx?pid=134'>Leggins</a></li>
<li><a href='/ShoppingList.aspx?pid=135'>Lehengas</a></li>
<li><a href='/ShoppingList.aspx?pid=136'>Lengerie</a></li>
</ul>
</div>
```

```
<div class='col-md-2 col-sm-6'>
<ul class='list-unstyled style-list'>
<li><a href='/ShoppingList.aspx?pid=137'>Mix and Match</a></li>
<li><a href='/ShoppingList.aspx?pid=138'>Overalls</a></li>
<li><a href='/ShoppingList.aspx?pid=139'>Pants</a></li>
<li><a href='/ShoppingList.aspx?pid=140'>Rompers</a></li>
<li><a href='/ShoppingList.aspx?pid=141'>Salwar Khameez</a></li>
<li><a href='/ShoppingList.aspx?pid=142'>Sarees</a></li>
<li><a href='/ShoppingList.aspx?pid=143'>Shorts</a></li>
<li><a href='/ShoppingList.aspx?pid=144'>Skirts</a></li>
<li><a href='/ShoppingList.aspx?pid=145'>Socks</a></li>
<li><a href='/ShoppingList.aspx?pid=146'>Suiting</a></li>
<li><a href='/ShoppingList.aspx?pid=147'>Sweaters</a></li>
<li><a href='/ShoppingList.aspx?pid=148'>SweatShirts</a></li>
<li><a href='/ShoppingList.aspx?pid=149'>Swim Suits</a></li>
<li><a href='/ShoppingList.aspx?pid=150'>Tees</a></li>
<li><a href='/ShoppingList.aspx?pid=151'>Tops</a></li>
<li><a href='/ShoppingList.aspx?pid=152'>Vests</a></li>
</ul>
</div>
```

```
<div class='col-md-2 col-sm-6'>
<ul class='list-unstyled style-list'>
<h3 class='mega-menu-heading'>Handbags & Wallets</h3>
<ul class='list-unstyled style-list'>
<li><a href='/ShoppingList.aspx?pid=153'>Clutches & Evening Bags</a></li>
<li><a href='/ShoppingList.aspx?pid=154'>Crossbody Bags</a></li>
```

- Fashion Backpacks
- Hobo Bags
- Satchels
- Shoulder Bags
- Top-Handle Bags
- Totes
- Wallets
- Wristlets

</div>

<div class='col-md-2 col-sm-6'>

<ul class='list-unstyled style-list'>

<h3 class='mega-menu-heading'>Jewelry</h3>

<ul class='list-unstyled style-list'>

- Anklets
- Bracelets
- Brooches & Pins
- Earrings
- Jewelry Sets
- Necklaces
- Rings
- Wedding & Engagement
- Smart Jewelry
- Body Jewelry

</div>

<div class='col-md-2 col-sm-6'>

<ul class='list-unstyled style-list'>

<h3 class='mega-menu-heading'>shoes</h3>

<ul class='list-unstyled style-list'>

- Athletic
- Boots

- Fashion Sneakers
- Flats
- Loafers & Slip-Ons
- Mules & Clogs
- Outdoor
- Oxfords
- Pumps
- Sandals
- Slippers
- Work & Safety

</div>

<div class='col-md-2 col-sm-6'>

<ul class='list-unstyled style-list'>

<h3 class='mega-menu-heading'>watches</h3>

<ul class='list-unstyled style-list'>

Wrist Watches

Watch Bands

Pocket Watches

Smartwatches

Certified Pre-Owned

</div>

<div class='col-md-2 col-sm-6'>

<ul class='list-unstyled style-list'>

<h3 class='mega-menu-heading'>Accessories</h3>

<ul class='list-unstyled style-list'>

Belts

Sunglasses & Eyewear Accessories

Scarves & Wraps

Gloves & Mittens

Hats & Caps

- Earmuffs
- Handbag Accessories
- Keyrings & Keychains
- Wallets, Card Cases & Money Organizers
- Special Occasion Accessories

</div>

<div class='col-md-2 col-sm-6'>

<ul class='list-unstyled style-list'>

</div>

</div>

</div>

<li class='dropdown mega-menu-fullwidth'>

Groom Fashion

<ul class='dropdown-menu'>

<div class='mega-menu-content'>

<div class='container'>

<div class='row'>

<div class='col-md-2 col-sm-6'>

<h3 class='mega-menu-heading'>Clothing</h3>

<ul class='list-unstyled style-list'>

Blazers

Coats

Coverups

Fashion Hoodies

Hosiery

- Indo western
- Jackets
- Jeans
- Jumpsuits
- Mix and Match
- Overalls
- Pants
- Rompers
- Shorts
- Skirts
- Socks

</div>

<div class='col-md-2 col-sm-6'>

<ul class='list-unstyled style-list'>

- Suiting
- Sweaters
- SweatShirts
- Tees
- Vests

</div>

<div class='col-md-2 col-sm-6'>

<ul class='list-unstyled style-list'>

<h3 class='mega-menu-heading'>Handbags & Wallets</h3>

<ul class='list-unstyled style-list'>

- Clutches & Evening Bags
- Crossbody Bags
- Fashion Backpacks
- Hobo Bags
- Satchels
- Shoulder Bags

Top-Handle Bags
Totes
Wallets
Wristlets

</div>

<div class='col-md-2 col-sm-6'>
<ul class='list-unstyled style-list'>
<h3 class='mega-menu-heading'>Jewelry</h3>
<ul class='list-unstyled style-list'>
Anklets
Bracelets
Brooches & Pins
Earrings
Jewelry Sets
Necklaces
Rings
Wedding & Engagement
Smart Jewelry
Body Jewelry

</div>

<div class='col-md-2 col-sm-6'>
<ul class='list-unstyled style-list'>
<h3 class='mega-menu-heading'>shoes</h3>
<ul class='list-unstyled style-list'>
Athletic
Boots
Fashion Sneakers
Flats
Loafers & Slip-Ons
Mules & Clogs

```
<li><a href='/ShoppingList.aspx?pid=247'>Outdoor</a></li>
<li><a href='/ShoppingList.aspx?pid=248'>Oxfords</a></li>
<li><a href='/ShoppingList.aspx?pid=249'>Sandals</a></li>
<li><a href='/ShoppingList.aspx?pid=250'>Slippers</a></li>
<li><a href='/ShoppingList.aspx?pid=251'>Work & Safety</a></li>
</ul>
</div>
```

```
<div class='col-md-2 col-sm-6'>
<ul class='list-unstyled style-list'>
<h3 class='mega-menu-heading'>watches</h3>
<ul class='list-unstyled style-list'>
<li><a href='/ShoppingList.aspx?pid=252'>Wrist Watches</a></li>
<li><a href='/ShoppingList.aspx?pid=253'>Watch Bands</a></li>
<li><a href='/ShoppingList.aspx?pid=254'>Pocket Watches</a></li>
<li><a href='/ShoppingList.aspx?pid=255'>Smartwatches</a></li>
<li><a href='/ShoppingList.aspx?pid=256'>Certified Pre-Owned</a></li>
</ul>
</div>
```

```
<div class='col-md-2 col-sm-6'>
<ul class='list-unstyled style-list'>
<h3 class='mega-menu-heading'>Accessories</h3>
<ul class='list-unstyled style-list'>
<li><a href='/ShoppingList.aspx?pid=257'>Belts</a></li>
<li><a href='/ShoppingList.aspx?pid=258'>Sunglasses & Eyewear Accessories</a></li>
<li><a href='/ShoppingList.aspx?pid=259'>Scarves & Wraps</a></li>
<li><a href='/ShoppingList.aspx?pid=260'>Gloves & Mittens</a></li>
<li><a href='/ShoppingList.aspx?pid=261'>Hats & Caps</a></li>
<li><a href='/ShoppingList.aspx?pid=262'>Earmuffs</a></li>
<li><a href='/ShoppingList.aspx?pid=263'>Handbag Accessories</a></li>
<li><a href='/ShoppingList.aspx?pid=264'>Keyrings & Keychains</a></li>
<li><a href='/ShoppingList.aspx?pid=265'>Wallets, Card Cases & Money
Organizers</a></li>
<li><a href='/ShoppingList.aspx?pid=266'>Special Occasion Accessories</a></li>
```


</div>

<div class='col-md-2 col-sm-6'>

<ul class='list-unstyled style-list'>

</div>

</div>

</div>

</div>

<li class='dropdown mega-menu-fullwidth'>

Men Fashion

<ul class='dropdown-menu'>

<div class='mega-menu-content'>

<div class='container'>

<div class='row'>

<div class='col-md-2 col-sm-6'>

<h3 class='mega-menu-heading'>Clothing</h3>

<ul class='list-unstyled style-list'>

Blazers

Coats

Coverups

Fashion Hudies

Hosiery

Indo western

Jackets

Jeans

Jumpsuits

Mix and Match

Overalls

Pants
Rompers
Shorts
Skirts
Socks

</div>

<div class='col-md-2 col-sm-6'>
<ul class='list-unstyled style-list'>
Suiting
Sweaters
SweatShirts
Tees
Vests

</div>

<div class='col-md-2 col-sm-6'>
<ul class='list-unstyled style-list'>
<h3 class='mega-menu-heading'>Handbags & Wallets</h3>
<ul class='list-unstyled style-list'>
Clutches & Evening Bags
Crossbody Bags
Fashion Backpacks
Hobo Bags
Satchels
Shoulder Bags
Top-Handle Bags
Totes
Wallets
Wristlets

</div>

```
<div class='col-md-2 col-sm-6'>
<ul class='list-unstyled style-list'>
<h3 class='mega-menu-heading'>Jewelry</h3>
<ul class='list-unstyled style-list'>
<li><a href='/ShoppingList.aspx?pid=298'>Anklets</a></li>
<li><a href='/ShoppingList.aspx?pid=299'>Bracelets</a></li>
<li><a href='/ShoppingList.aspx?pid=300'>Brooches & Pins</a></li>
<li><a href='/ShoppingList.aspx?pid=301'>Earrings</a></li>
<li><a href='/ShoppingList.aspx?pid=302'>Jewelry Sets</a></li>
<li><a href='/ShoppingList.aspx?pid=303'>Necklaces</a></li>
<li><a href='/ShoppingList.aspx?pid=304'>Rings</a></li>
<li><a href='/ShoppingList.aspx?pid=305'>Wedding & Engagement</a></li>
<li><a href='/ShoppingList.aspx?pid=306'>Smart Jewelry</a></li>
<li><a href='/ShoppingList.aspx?pid=307'>Body Jewelry</a></li>
</ul>
</div>
```

```
<div class='col-md-2 col-sm-6'>
<ul class='list-unstyled style-list'>
<h3 class='mega-menu-heading'>shoes</h3>
<ul class='list-unstyled style-list'>
<li><a href='/ShoppingList.aspx?pid=308'>Athletic</a></li>
<li><a href='/ShoppingList.aspx?pid=309'>Boots</a></li>
<li><a href='/ShoppingList.aspx?pid=310'>Fashion Sneakers</a></li>
<li><a href='/ShoppingList.aspx?pid=311'>Flats</a></li>
<li><a href='/ShoppingList.aspx?pid=312'>Loafers & Slip-Ons</a></li>
<li><a href='/ShoppingList.aspx?pid=313'>Mules & Clogs</a></li>
<li><a href='/ShoppingList.aspx?pid=314'>Outdoor</a></li>
<li><a href='/ShoppingList.aspx?pid=315'>Oxfords</a></li>
<li><a href='/ShoppingList.aspx?pid=316'>Sandals</a></li>
<li><a href='/ShoppingList.aspx?pid=317'>Slippers</a></li>
<li><a href='/ShoppingList.aspx?pid=318'>Work & Safety</a></li>
</ul>
```

</div>

<div class='col-md-2 col-sm-6'>

<ul class='list-unstyled style-list'>

<h3 class='mega-menu-heading'>watches</h3>

<ul class='list-unstyled style-list'>

Wrist Watches

Watch Bands

Pocket Watches

Smartwatches

Certified Pre-Owned

</div>

<div class='col-md-2 col-sm-6'>

<ul class='list-unstyled style-list'>

<h3 class='mega-menu-heading'>Accessories</h3>

<ul class='list-unstyled style-list'>

Belts

Sunglasses & Eyewear Accessories

Scarves & Wraps

Gloves & Mittens

Hats & Caps

Earmuffs

Handbag Accessories

Keyrings & Keychains

Wallets, Card Cases & Money Organizers

Special Occasion Accessories

</div>

<div class='col-md-2 col-sm-6'>

<ul class='list-unstyled style-list'>

</div>

</div>

</div>

</div>

<li class='dropdown mega-menu-fullwidth'>

Gadgets & Home Living

<ul class='dropdown-menu'>

<div class='mega-menu-content'>

<div class='container'>

<div class='row'>

<div class='col-md-2 col-sm-6'>

<h3 class='mega-menu-heading'>TV & Electronics</h3>

<ul class='list-unstyled style-list'>

Air Conditioners

Air Coolers

Air Purifiers

Audio Devices

Camera

Desktops

Fans

Fridge

Geysers

Irons

Laptops

Microwave Ovens

Mixer & Grinders

Mobile Accessories

Mobiles

Office Electronis

</div>

<div class='col-md-2 col-sm-6'>

<ul class='list-unstyled style-list'>

TVs

Video Devices

Washing Machines

Water Purifiers

Wet Grinders

</div>

<div class='col-md-2 col-sm-6'>

<ul class='list-unstyled style-list'>

<h3 class='mega-menu-heading'>Kitchen Appliances</h3>

<ul class='list-unstyled style-list'>

Casseroles

Cooking Tools

Cookware

Cutlery

Dinner Sets

Gas Stoves

Glasses & Jugs

Induction CookTops

Jars & Containers

Lunch Boxes

Pans & Tawas

Pressure Cookers

Thermos & Flasks

Water Bottles

</div>

<div class='col-md-2 col-sm-6'>

<ul class='list-unstyled style-list'>

Furniture & Fixtures</h3>

<ul class='list-unstyled style-list'>

Beds

Sofas

Matters

Pillows

Tables

Chairs

Dining Table & Chairs Set

Recliners

Coffee Tables

WardRobes

Book Shelves

Shoe Racks

Cabinets

Bedside Tables

Bed Sheets

Towels

</div>

<div class='col-md-2 col-sm-6'>

<ul class='list-unstyled style-list'>

Pillow Covers

Curtains

Carpets

Rugs

Bathroom Mats

Sofa covers

</div>

<div class='col-md-2 col-sm-6'>

<ul class='list-unstyled style-list'>

Home & Living</h3>

<ul class='list-unstyled style-list'>

Kitchen Mats

Kitchen Aprons

Duvet Covers

Bath Robes

Table Clothes

Curtain Accessories

Shower Curtains

</div>

<div class='col-md-2 col-sm-6'>

<ul class='list-unstyled style-list'>

<ul class='list-unstyled style-list'>

Silver Items

Mixed Metal Items

Stainless Steel Items

Iron Items

</div>

<div class='col-md-2 col-sm-6'>

<ul class='list-unstyled style-list'>

</div>

</div>

</div>

</div>

<li class='dropdown mega-menu-fullwidth'>

Wedding Materials

```
<ul class='dropdown-menu'>
<li>
<div class='mega-menu-content'>
<div class='container'>
<div class='row'>
<div class='col-md-2 col-sm-6'>
<h3 class='mega-menu-heading'>Wedding Materials</h3>
<ul class='list-unstyled style-list'>
<li><a href='/ShoppingList.aspx?pid=402'>Wedding Items</a></li>
</ul>
</div>
<div class='col-md-2 col-sm-6'>
<ul class='list-unstyled style-list'>
</div>
</div>
</div>
</div>
</li>

</ul>
</ul>

</div> --%>

</div>
<%--<div>
```

```
<ul class='nav navbar-nav'>
<li class='dropdown mega-menu-fullwidth'>
<a class='dropdown-toggle' href='javascript:void(0);' data-hover='dropdown' data-
toggle='dropdown'>Bride Fashion</a>
<ul class='dropdown-menu'>
<li>
<div class='mega-menu-content'>
<div class='container'>
<div class='row'>
<div class='col-md-2 col-sm-6'>
<h3 class='mega-menu-heading'>Clothing</h3>
<ul class='list-unstyled style-list'>
<li><a href='ShoppingList.aspx?pgid=42'>Blazers</a></li>
<li><a href='ShoppingList.aspx?pgid=43'>Blouses</a></li>
<li><a href='ShoppingList.aspx?pgid=44'>Coats</a></li>
<li><a href='ShoppingList.aspx?pgid=45'>Coverups</a></li>
<li><a href='ShoppingList.aspx?pgid=46'>Fashion Hudies</a></li>
<li><a href='ShoppingList.aspx?pgid=47'>Hosiery</a></li>
<li><a href='ShoppingList.aspx?pgid=48'>Indo western</a></li>
<li><a href='ShoppingList.aspx?pgid=49'>Jackets</a></li>
<li><a href='ShoppingList.aspx?pgid=50'>Jeans</a></li>
<li><a href='ShoppingList.aspx?pgid=51'>Jeggins</a></li>
<li><a href='ShoppingList.aspx?pgid=52'>Jumpsuits</a></li>
<li><a href='ShoppingList.aspx?pgid=53'>Kurtaset</a></li>
<li><a href='ShoppingList.aspx?pgid=54'>Kurtees</a></li>
<li><a href='ShoppingList.aspx?pgid=55'>Leggins</a></li>
<li><a href='ShoppingList.aspx?pgid=56'>Lehengas</a></li>
</ul>
</div>
<div class='col-md-2 col-sm-6'>
<ul class='list-unstyled style-list'>
<li><a href='ShoppingList.aspx?pgid=57'>Lengerie</a></li>
```

- [Mix and Match](ShoppingList.aspx?pgid=58)
- [Overalls](ShoppingList.aspx?pgid=59)
- [Pants](ShoppingList.aspx?pgid=60)
- [Rompers](ShoppingList.aspx?pgid=61)
- [Salwar Khameez](ShoppingList.aspx?pgid=62)
- [Sarees](ShoppingList.aspx?pgid=63)
- [Shorts](ShoppingList.aspx?pgid=64)
- [Skirts](ShoppingList.aspx?pgid=65)
- [Socks](ShoppingList.aspx?pgid=66)
- [Suiting](ShoppingList.aspx?pgid=67)
- [Sweaters](ShoppingList.aspx?pgid=68)
- [SweatShirts](ShoppingList.aspx?pgid=69)
- [Swim Suits](ShoppingList.aspx?pgid=70)
- [Tees](ShoppingList.aspx?pgid=71)
- [Tops](ShoppingList.aspx?pgid=72)
- [Vests](ShoppingList.aspx?pgid=73)

</div>

<div class='col-md-2 col-sm-6'>

<ul class='list-unstyled style-list'> - - - - - - - - - -

</div>

<div class='col-md-2 col-sm-6'>

<h3 class='mega-menu-heading'>Jewelry</h3>

<ul class='list-unstyled style-list'>

Anklets

Bracelets

Brooches & Pins

Earrings

Jewelry Sets

Necklaces

Rings

Wedding & Engagement

Smart Jewelry

Body Jewelry

</div>

<div class='col-md-2 col-sm-6'>

<h3 class='mega-menu-heading'>shoes</h3>

<ul class='list-unstyled style-list'>

Athletic

Boots

Fashion Sneakers

Flats

Loafers & Slip-Ons

Mules & Clogs

Outdoor

Oxfords

Pumps

Sandals

Slippers

Work & Safety

<h3 class='mega-menu-heading'>watches</h3>

<ul class='list-unstyled style-list'>

Wrist Watches

Watch Bands

Pocket Watches

Smartwatches

Certified Pre-Owned

</div>

<div class='col-md-2 col-sm-6'>

<h3 class='mega-menu-heading'>Accessories</h3>

<ul class='list-unstyled style-list'>

Belts

Sunglasses & Eyewear Accessories

Scarves & Wraps

Gloves & Mittens

Hats & Caps

Earmuffs

Handbag Accessories

Keyrings & Keychains

Wallets, Card Cases & Money Organizers

Special Occasion Accessories

</div>

</div>

</div>

</div>

<li class='dropdown mega-menu-fullwidth'>

Women Fashion

<ul class='dropdown-menu'>

<div class='mega-menu-content'>

<div class='container'>

<div class='row'>

<div class='col-md-2 col-sm-6'>

<h3 class='mega-menu-heading'>Clothing</h3>

<ul class='list-unstyled style-list'>

Blazers

Blouses

Coats

Coverups

Fashion Hoodies

Hosiery

Indo western

Jackets

Jeans

Jeggins

Jumpsuits

Kurtaset

Kurtees

Leggins

Lehengas

</div>

<div class='col-md-2 col-sm-6'>

<ul class='list-unstyled style-list'>

Lingerie

Mix and Match

Overalls

Pants

- Rompers
- Salwar Khameez
- Sarees
- Shorts
- Skirts
- Socks
- Suiting
- Sweaters
- SweatShirts
- Swim Suits
- Tees
- Tops
- Vests

</div>

<h3 class='mega-menu-heading'>Handbags & Wallets</h3>

<ul class='list-unstyled style-list'>

- Clutches & Evening Bags
- Crossbody Bags
- Fashion Backpacks
- Hobo Bags
- Satchels
- Shoulder Bags
- Top-Handle Bags
- Totes
- Wallets
- Wristlets

</div>

<div class='col-md-2 col-sm-6'>

<h3 class='mega-menu-heading'>Jewelry</h3>

- Anklets
- Bracelets
- Brooches & Pins
- Earrings
- Jewelry Sets
- Necklaces
- Rings
- Wedding & Engagement
- Smart Jewelry
- Body Jewelry

</div>

<div class='col-md-2 col-sm-6'>

- Athletic
- Boots
- Fashion Sneakers
- Flats
- Loafers & Slip-Ons
- Mules & Clogs
- Outdoor
- Oxfords
- Pumps
- Sandals
- Slippers
- Work & Safety

- Wrist Watches

- Watch Bands
- Pocket Watches
- Smartwatches
- Certified Pre-Owned

</div>

<h3 class='mega-menu-heading'>Accessories</h3>

<ul class='list-unstyled style-list'>

- Belts
- Sunglasses & Eyewear Accessories
- Scarves & Wraps
- Gloves & Mittens
- Hats & Caps
- Earmuffs
- Handbag Accessories
- Keyrings & Keychains
- Wallets, Card Cases & Money Organizers
- Special Occasion Accessories

</div>

</div>

</div>

</div>

<li class='dropdown mega-menu-fullwidth'>

Groom Fashion

<ul class='dropdown-menu'>


```
<div class='mega-menu-content'>
<div class='container'>
<div class='row'>
<div class='col-md-2 col-sm-6'>
<h3 class='mega-menu-heading'>Clothing</h3>
<ul class='list-unstyled style-list'>
<li><a href='ShoppingList.aspx?pgid=200'>Blazers</a></li>
<li><a href='ShoppingList.aspx?pgid=201'>Coats</a></li>
<li><a href='ShoppingList.aspx?pgid=202'>Coverups</a></li>
<li><a href='ShoppingList.aspx?pgid=203'>Fashion Hudies</a></li>
<li><a href='ShoppingList.aspx?pgid=204'>Hosiery</a></li>
<li><a href='ShoppingList.aspx?pgid=205'>Indo western</a></li>
<li><a href='ShoppingList.aspx?pgid=206'>Jackets</a></li>
<li><a href='ShoppingList.aspx?pgid=207'>Jeans</a></li>
<li><a href='ShoppingList.aspx?pgid=208'>Jumpsuits</a></li>
<li><a href='ShoppingList.aspx?pgid=209'>Mix and Match</a></li>
<li><a href='ShoppingList.aspx?pgid=210'>Overalls</a></li>
<li><a href='ShoppingList.aspx?pgid=211'>Pants</a></li>
<li><a href='ShoppingList.aspx?pgid=212'>Rompers</a></li>
<li><a href='ShoppingList.aspx?pgid=213'>Shorts</a></li>
<li><a href='ShoppingList.aspx?pgid=214'>Skirts</a></li>
</ul>
</div>
<div class='col-md-2 col-sm-6'>
<ul class='list-unstyled style-list'>
<li><a href='ShoppingList.aspx?pgid=215'>Socks</a></li>
<li><a href='ShoppingList.aspx?pgid=216'>Suiting</a></li>
<li><a href='ShoppingList.aspx?pgid=217'>Sweaters</a></li>
<li><a href='ShoppingList.aspx?pgid=218'>SweatShirts</a></li>
<li><a href='ShoppingList.aspx?pgid=219'>Tees</a></li>
<li><a href='ShoppingList.aspx?pgid=220'>Vests</a></li>
</ul>
```

>Handbags & Wallets</h3>

<ul class='list-unstyled style-list'>

Clutches & Evening Bags

Crossbody Bags

Fashion Backpacks

Hobo Bags

Satchels

Shoulder Bags

Top-Handle Bags

Totes

Wallets

Wristlets

</div>

<div class='col-md-2 col-sm-6'>

>Jewelry</h3>

<ul class='list-unstyled style-list'>

Anklets

Bracelets

Brooches & Pins

Earrings

Jewelry Sets

Necklaces

Rings

Wedding & Engagement

Smart Jewelry

Body Jewelry

>shoes</h3>

<ul class='list-unstyled style-list'>

Athletic

Boots

- Fashion Sneakers
- Flats
- Loafers & Slip-Ons
- Mules & Clogs
- Outdoor
- Oxfords
- Sandals
- Slippers
- Work & Safety

</div>

<div class='col-md-2 col-sm-6'>

<h3 class='mega-menu-heading'>watches</h3>

<ul class='list-unstyled style-list'>

- Wrist Watches
- Watch Bands
- Pocket Watches
- Smartwatches
- Certified Pre-Owned

<h3 class='mega-menu-heading'>Accessories</h3>

<ul class='list-unstyled style-list'>

- Belts
- Sunglasses & Eyewear Accessories
- Scarves & Wraps
- Gloves & Mittens
- Hats & Caps
- Earmuffs
- Handbag Accessories
- Keyrings & Keychains
- Wallets, Card Cases & Money Organizers
- Special Occasion Accessories

</div>

</div>

</div>

</div>

<li class='dropdown mega-menu-fullwidth'>

Men Fashion

<ul class='dropdown-menu'>

<div class='mega-menu-content'>

<div class='container'>

<div class='row'>

<div class='col-md-2 col-sm-6'>

<h3 class='mega-menu-heading'>Clothing</h3>

<ul class='list-unstyled style-list'>

Blazers

Coats

Coverups

Fashion Hudies

Hosiery

Indo western

Jackets

Jeans

Jumpsuits

Mix and Match

Overalls

Pants

Rompers

Shorts

Skirts

</div>

<div class='col-md-2 col-sm-6'>

<ul class='list-unstyled style-list'>

Socks

Suiting

Sweaters

SweatShirts

Tees

Vests

<h3 class='mega-menu-heading'>Handbags & Wallets</h3>

<ul class='list-unstyled style-list'>

Clutches & Evening Bags

Crossbody Bags

Fashion Backpacks

Hobo Bags

Satchels

Shoulder Bags

Top-Handle Bags

Totes

Wallets

Wristlets

</div>

<div class='col-md-2 col-sm-6'>

<h3 class='mega-menu-heading'>Jewelry</h3>

<ul class='list-unstyled style-list'>

Anklets

Bracelets

Brooches & Pins

- Earrings
- Jewelry Sets
- Necklaces
- Rings
- Wedding & Engagement
- Smart Jewelry
- Body Jewelry

</div>

<div class='col-md-2 col-sm-6'>

<h3 class='mega-menu-heading'>shoes</h3>

<ul class='list-unstyled style-list'>

- Athletic
- Boots
- Fashion Sneakers
- Flats
- Loafers & Slip-Ons
- Mules & Clogs
- Outdoor
- Oxfords
- Sandals
- Slippers
- Work & Safety

<h3 class='mega-menu-heading'>watches</h3>

<ul class='list-unstyled style-list'>

- Wrist Watches
- Watch Bands
- Pocket Watches
- Smartwatches
- Certified Pre-Owned

</div>

<div class='col-md-2 col-sm-6'>

<h3 class='mega-menu-heading'>Accessories</h3>

<ul class='list-unstyled style-list'>

Belts

Sunglasses & Eyewear Accessories

Scarves & Wraps

Gloves & Mittens

Hats & Caps

Earmuffs

Handbag Accessories

Keyrings & Keychains

Wallets, Card Cases & Money Organizers

Special Occasion Accessories

</div>

</div>

</div>

</div>

<li class='dropdown mega-menu-fullwidth'>

Gadgets

<ul class='dropdown-menu'>

<div class='mega-menu-content'>

<div class='container'>

<div class='row'>

<div class='col-md-2 col-sm-6'>

<h3 class='mega-menu-heading'>TV & Electronics</h3>

```
<ul class='list-unstyled style-list'>
<li><a href='ShoppingList.aspx?pgid=334'>Air Conditioners</a></li>
<li><a href='ShoppingList.aspx?pgid=335'>Air Coolers</a></li>
<li><a href='ShoppingList.aspx?pgid=336'>Air Purifiers</a></li>
<li><a href='ShoppingList.aspx?pgid=337'>Audio Devices</a></li>
<li><a href='ShoppingList.aspx?pgid=338'>Camera</a></li>
<li><a href='ShoppingList.aspx?pgid=339'>Desktops</a></li>
<li><a href='ShoppingList.aspx?pgid=340'>Fans</a></li>
<li><a href='ShoppingList.aspx?pgid=341'>Fridge</a></li>
<li><a href='ShoppingList.aspx?pgid=342'>Geysers</a></li>
<li><a href='ShoppingList.aspx?pgid=343'>Irons</a></li>
<li><a href='ShoppingList.aspx?pgid=344'>Laptops</a></li>
<li><a href='ShoppingList.aspx?pgid=345'>Microwave Ovens</a></li>
<li><a href='ShoppingList.aspx?pgid=346'>Mixer & Grinders</a></li>
<li><a href='ShoppingList.aspx?pgid=347'>Mobile Accessories</a></li>
<li><a href='ShoppingList.aspx?pgid=348'>Mobiles</a></li>
</ul>
</div>
```

```
<div class='col-md-2 col-sm-6'>
<ul class='list-unstyled style-list'>
<li><a href='ShoppingList.aspx?pgid=349'>Office Electronics</a></li>
<li><a href='ShoppingList.aspx?pgid=350'>TVs</a></li>
<li><a href='ShoppingList.aspx?pgid=351'>Video Devices</a></li>
<li><a href='ShoppingList.aspx?pgid=352'>Washing Machines</a></li>
<li><a href='ShoppingList.aspx?pgid=353'>Water Purifiers</a></li>
<li><a href='ShoppingList.aspx?pgid=354'>Wet Grinders</a></li>
</ul>
```

<h3 class='mega-menu-heading'>Kitchen Appliances</h3>

```
<ul class='list-unstyled style-list'>
<li><a href='ShoppingList.aspx?pgid=355'>Casseroles</a></li>
<li><a href='ShoppingList.aspx?pgid=356'>Cooking Tools</a></li>
<li><a href='ShoppingList.aspx?pgid=357'>Cookware</a></li>
```

- Cutlery
- Dinner Sets
- Gas Stoves
- Glasses & Jugs
- Induction CookTops
- Jars & Containers
- Lunch Boxes
- Pans & Tawas
- Pressure Cookers
- Thermos & Flasks
- Water Bottles

- Beds - Sofas - Matters - Pillows - Tables - Chairs - Dining Table & Chairs Set - Recliners - Coffee Tables - WardRobes - Book Shelves - Shoe Racks - Cabinets - Bedside Tables - Bed Sheets </div> <div class='col-md-2 col-sm-6'>

- [Towels](ShoppingList.aspx?pgid=384)
- [Pillow Covers](ShoppingList.aspx?pgid=385)
- [Curtains](ShoppingList.aspx?pgid=386)
- [Carpets](ShoppingList.aspx?pgid=387)
- [Rugs](ShoppingList.aspx?pgid=388)
- [Bathroom Mats](ShoppingList.aspx?pgid=389)
- [Sofa covers](ShoppingList.aspx?pgid=390)

Home & Living

- [Kitchen Mats](ShoppingList.aspx?pgid=391)
- [Kitchen Aprons](ShoppingList.aspx?pgid=392)
- [Duvet Covers](ShoppingList.aspx?pgid=393)
- [Bath Robes](ShoppingList.aspx?pgid=394)
- [Table Clothes](ShoppingList.aspx?pgid=395)
- [Curtain Accessories](ShoppingList.aspx?pgid=396)
- [Shower Curtains](ShoppingList.aspx?pgid=397)

</div>

</div>

</div>

</div>

<li class='dropdown mega-menu-fullwidth'>

Metal Appliances

<ul class='dropdown-menu'>

<div class='mega-menu-content'>

<div class='container'>

```
<div class='row'>
<div class='col-md-2 col-sm-6'>
<h3 class='mega-menu-heading'>Metal Appliances</h3>
<ul class='list-unstyled style-list'>
<li><a href='ShoppingList.aspx?pgid=398'>Silver Items</a></li>
<li><a href='ShoppingList.aspx?pgid=399'>Mixed Metal Items</a></li>
<li><a href='ShoppingList.aspx?pgid=400'>Stainless Steel Items</a></li>
<li><a href='ShoppingList.aspx?pgid=401'>Iron Items</a></li>
</ul>
</div>
</div>
</div>
</div>
</div>
</li>

</ul>
<li class='dropdown mega-menu-fullwidth'>
<a class='dropdown-toggle' href='javascript:void(0);' data-hover='dropdown' data-
toggle='dropdown'>Wedding Materials</a>
<ul class='dropdown-menu'>
<li>
<div class='mega-menu-content'>
<div class='container'>
<div class='row'>
<div class='col-md-2 col-sm-6'>
<h3 class='mega-menu-heading'>Wedding Materials</h3>
<ul class='list-unstyled style-list'>
<li><a href='ShoppingList.aspx?pgid=402'>Wedding Items</a></li>
</ul>
</div>
</div>
</div>
</div>
</div>
```


</div>--%>

<!-- End Nav Menu -->

<%-- menu ends--%>

</div>

</div>

<!-- End Navbar -->

</div>

<!-- End Header-->

<!-- Start Parallax -->

```

<asp:ContentPlaceHolder ID="ContentMainPage" runat="server">
</asp:ContentPlaceHolder>
    <!-- Top Logo/Messages Div/Section Start -->
    <asp:ContentPlaceHolder ID="CPH_TopMessages" runat="server">
        <div class="container-fluid">

            <div class="row-fluid">
                <div class="span2" id="dvlogo" runat="server" style="display: none;">

                    </div>

                <div class="span10">
                    <div class="row">
                        <div id="dvMsgBox" class="span12 ordersave" runat="server"
style="display: block;">
                            <asp:Label ID="Label1" runat="server" Text=""></asp:Label>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </asp:ContentPlaceHolder>
    <!-- Top Logo/Messages Div/Section End -->

    <div class="container-fluid">
        <div class="row-fluid">

            <!-- Left Bar Div/Section Start -->
            <!-- Left Bar Div/Section End -->

            <!-- Main Content Div/Section Start -->
            <asp:ContentPlaceHolder ID="CPH_MainContent" runat="server">

```



```

        </asp:ContentPlaceHolder>
        <!-- Main Content Div/Section End -->
    </div>

</div>

<section class="home-main-contant-style3 bg-white">
    <div class="parallax-block parallaxBg">
        <div class="container">
            <div class="parallax-block-text">
                <p>We are "Dateorwed.com - Shopping & Services" <span class="color-green">Thanks for checking out our profile.</span> <br> Thank you! </p>
            </div>
            <div class="pre-btn-group">
                <a class="btn btn-product mr15" href="#">Shop Now</a>
                <a class="btn btn-product" href="#">get a free quote</a>
            </div>
        </div>
    </div>
</section>
<!-- End Parallax -->

<!-- START NEW -->
<section class="home-main-contant-style3 bg-white">
    <div class="container">
        <div class="widget">
            <h6 class="text-uppercase bottom-line">Our Brand</h6>
        </div>
        <div class="row">
            <div class="brand slider">
                <div class="col-md-2">

```

```
                <a href="#"></a>
            </div>
        <!--
            <div class="col-md-2">
                <a href="#"></a>
            </div>
            <div class="col-md-2">
                <a href="#"></a>
            </div>
            <div class="col-md-2">
                <a href="#"></a>
            </div>
            <div class="col-md-2">
                <a href="#"></a>
            </div>
            <div class="col-md-2">
                <a href="#"></a>
            </div>
            <div class="col-md-2">
                <a href="#"></a>
            </div>
            <div class="col-md-2">
                <a href="#"></a>
            </div>
            <div class="col-md-2">
                <a href="#"></a>
            </div>
```

```
<div class="col-md-2">
  <a href="#"></a>
</div>--%>
</div>
</div>
</div>
</div>
</section>
<!-- END NEW -->
```

```
<!-- Start Suvbscribe -->
<div class="news-subscribe">
  <div class="container">
    <div class="row">
      <div class="col-md-4">
        <h3>subscribe to <strong>newsletter</strong></h3>
      </div>
      <div class="col-md-4 newsletter-form-block">
        <div class="input-group">
          <input type="text" class="form-control" placeholder="Email your
email...">
          <span class="input-group-btn">
            <button class="btn" type="button"><i class="fa fa-envelope-o"></i></button>
          </span>
        </div>
      </div>
    </div>
    <div class="col-md-4 col-xs-12">
      <div class="social-footer">
        <ul class="social-icons social-icons-simple">
          <li><a
href="https://www.facebook.com/profile.php?id=100077266105117"><i class="fab fa-facebook"></i></a></li>
```

```
                <li><a href="https://twitter.com/dateorwed"><i class="fab fa-
twitter"></i></a></li>

                <li><a href="#"><i class="fab fa-tumblr"></i></a></li>

                <li><a href="https://instagram.com/admin.dateorwed"><i class="fab
fa-instagram"></i></a></li>

                <li><a href="www.linkedin.com/in/administrator-dateorwedorwed-
780a58231"><i class="fab fa-linkedin"></i></a></li>

            </ul>

        </div>

    </div>

</div>

</div>

</div>

</div>

</div>

</div>

<!--End Suvbscribe -->

<asp:ContentPlaceHolder ID="ContentPlaceHolder1" runat="server">

    <div class="container-fluid">

        <div class="row-fluid">    <%-- <div class="row-fluid" style="padding-top:
20px;">--%>

            <div class="span2" id="Div1" runat="server" style="display: none;">

                <%--
                
                --%>

            </div>

            <div class="span10">

                <div class="row">

                    <div id="Div2" class="span12 ordersave" runat="server" style="display:
block;">

                        <asp:Label ID="Label2" runat="server" Text=""></asp:Label>

                    </div>

                </div>

            </div>

        </div>

    </div>

</div>

</div>
```

```
</asp:ContentPlaceholder>
```

```
<!-- START FOOTER -->
```

```
<footer class="footer">
```

```
  <div class="container">
```

```
    <div class="row">
```

```
      <!-- CONTACT WIDGET -->
```

```
      <div class="col-md-2 col-sm-2 col-xs-12">
```

```
        <div class="widget">
```

```
          <h6 class="montserrat text-uppercase">&nbsp;</h6>
```

```
          
```

```
        </div>
```

```
      </div>
```

```
      <!-- END CONTACT WIDGET -->
```

```
    <!-- CONTACT WIDGET -->
```

```
    <div class="col-md-3 col-sm-6 col-xs-12">
```

```
      <div class="widget">
```

```
        <h6 class="text-uppercase bottom-line">Contact Us</h6>
```

```
        <address>
```

```
          <p>H.No: 123/44 ABC Colony, Hyderabad, Telangana, India</p><br>
```

```
          <p>Phone: (+91) 98765-43210</p>
```

```
          <p>E-mail: <a
```

```
href="mailto:name@yoursitename.com">name@yoursitename.com</a></p>
```

```
        </address>
```

```
      </div>
```

```
    </div>
```

```
    <!-- END CONTACT WIDGET -->
```

```
  <!-- CATEGORIES WIDGET -->
```

```
  <div class="col-md-2 col-sm-5 col-xs-12">
```

```
    <div class="widget">
```

```
<h6 class="montserrat text-uppercase bottom-line">Users</h6>
<ul class="icons-list">
  <li>
    <a href="#">My Profile</a>

  </li>
</ul>
</div>
</div>
<!-- END CATEGORIES WIDGET -->
```

```
<!-- CATEGORIES WIDGET -->
<div class="col-md-2 col-sm-5 col-xs-12">
  <div class="widget">
    <h6 class="montserrat text-uppercase bottom-line">About Us</h6>
    <ul class="icons-list">
      <li><a href="#">About Us</a></li>
      <li><a href="#">FAQs</a></li>
      <li><a href="#">Feedback</a></li>
    </ul>
  </div>
</div>
<!-- END CATEGORIES WIDGET -->
```

```
<!-- RECENT POSTS WIDGET -->
<div class="col-md-3 col-sm-6 col-xs-12">
  <div class="widget">
    <h6 class="text-uppercase bottom-line">Service Providers</h6>
    <ul class="icons-list">
      <li><a href="#">Become a Service Provider</a></li>
```

```
        <li><a href="#">Dashboard</a></li>
        <li><a href="#">Emergency Contacts</a></li>
    </ul>
</div>
</div>
<!-- END RECENT POSTS WIDGET -->
```

```
</div>
```

```
<hr style="border-top: 1px solid #ddd; padding-bottom: 10px;">
```

```
<div class="row">
    <!-- CONTACT WIDGET -->
    <div class="col-md-3 col-sm-6 col-xs-12">
        <div class="widget">
            <h6 class="text-uppercase bottom-line">Top Brands</h6>
            <ul class="icons-list">
                <li><a href="#">Biba</a></li>
                <li><a href="#">Rain & Rainbow</a></li>
                <li><a href="#">Aks</a></li>
                <li><a href="#">Soch</a></li>
                <li><a href="#">Global Desi</a></li>
            </ul>
        </div>
    </div>
</div>
<!-- END CONTACT WIDGET -->
```

```
<!-- CATEGORIES WIDGET -->
<div class="col-md-3 col-sm-6 col-xs-12">
    <div class="widget">
```

```
<h6 class="montserrat text-uppercase bottom-line">Categories</h6>
<ul class="icons-list">
  <li><a href="#">Fashion <span class="pull-right">12</span></a></li>
  <li><a href="#">Blogger <span class="pull-right">11</span></a></li>
  <li><a href="#">Sport <span class="pull-right">10</span></a></li>
  <li><a href="#">Photography <span class="pull-
right">14</span></a></li>
  <li><a href="#">Popular <span class="pull-right">15</span></a></li>
</ul>
</div>
</div>
<!-- END CATEGORIES WIDGET -->
```

```
<!-- CATEGORIES WIDGET -->
<div class="col-md-3 col-sm-6 col-xs-12">
  <div class="widget">
    <h6 class="montserrat text-uppercase bottom-line">Quick Links</h6>
    <ul class="icons-list">
      <li><a href="#">About</a></li>
      <li><a href="#">Home</a></li>
      <li><a href="#">New Products</a></li>
      <li><a href="#">Teams</a></li>
      <li><a href="#">Popular</a></li>
    </ul>
  </div>
</div>
<!-- END CATEGORIES WIDGET -->
```

```
<!-- RECENT POSTS WIDGET -->
<div id="popprods" class="col-md-3 col-sm-6 col-xs-12" runat="server">
  <div class="widget">
```


<h6 class="text-uppercase bottom-line">Popular Products</h6>

<ul class="recent-posts">

<div class="widget-posts-image">

</div>

<div class="widget-posts-body">

<h6 class="widget-posts-title">Printed Summer

Dress</h6>

<div class="widget-posts-meta">₹5220.45</div>

</div>

<div class="widget-posts-image">

</div>

<div class="widget-posts-body">

<h6 class="widget-posts-title">Printed Summer

Dress</h6>

<div class="widget-posts-meta">₹2300.45</div>

</div>

<div class="widget-posts-image">

</div>

<div class="widget-posts-body">

<h6 class="widget-posts-title">Printed Summer

Dress</h6>

<div class="widget-posts-meta">₹1100.45</div>

</div>


```
</div>
```

```
</div>
```

```
<!-- END RECENT POSTS WIDGET -->
```

```
</div>
```

```
<!-- CONTACT WIDGET -->
```

```
<div class="copyright">
```

```
<div class="text-center">
```

```
<p>© 2022 Copyrights All Rights Reserved @ <a target="_blank"  
href="Home.aspx">dateorwed.com</a></p>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</footer>
```

```
<!-- END FOOTER-->
```

```
<asp:ContentPlaceHolder ID="ContentFooter" runat="server">
```

```
</asp:ContentPlaceHolder>
```

```
</form>
```

```
</body>
```

```
</html>
```

```
*****ShoppingMasterPage.master.cs *****
```

```
using System;
```

```
using System.Collections.Generic;
```

```
using System.Linq;
```

```
using System.Web;
```

```
using System.Web.UI;
```

```
using System.Web.UI.WebControls;
```

```

using DowBEntity;
using DowBAL;
using System.Text;
using DowDAL;
using System.Data;
using System.Web.Security;

public partial class ShoppingMasterPage : System.Web.UI.MasterPage
{
    string UserName;
    DDLogTracer objlog = new DDLogTracer();

    protected void Page_Load(object sender, EventArgs e)
    {
        string logpath = Server.MapPath("~/log/" + "Log" +
Convert.ToString(DateTime.Today.Day) + ".txt");
        Session["Logpath"] = logpath;
        string user="";

        StringBuilder sb = new StringBuilder();
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];

        if (Clsusr != null)
        {
            UserName = Clsusr.UserName;
            int dftcnt = new Products().GetSLItems(Convert.ToInt16( Clsusr.UserId), true);
            sb.Append("<ul class='list-inline topbar-right pull-right'>\n");
            sb.Append("<li><a href='UserDetail.aspx'> Welcome " + UserName + "</a></li>\n");
            sb.Append("<li><a href='MyWishList.aspx'><i class='fas fa-hand-holding-heart'></i>Wishlist (" + Convert.ToString(dftcnt) + ")</a></li>\n");
            sb.Append("<li><a href='logout.aspx'><i class='fas fa-lock'></i>Logout</a> | <a href='UserDetail.aspx'><i class='fas fa-user'></i>Profile</a></li>");
            account.InnerHtml = sb.ToString();
        }
    }
}

```

```

}
else
{
    HttpCookie myCookie = Request.Cookies["DateOrWedCookie"];
    if (myCookie != null)
    {
        user = myCookie.Value.ToString();
    }
    else
    {
        user = "Guest";
        //Response.Redirect("MLB_login.aspx");
    }

    guest_login(user);

    //    Response.Redirect("MLB_Login.aspx");
}

    PopDlls ObjPD = new PopDlls();
    ObjPD.BindDropDowns(ddl_location, (new PopDlls()).GetStores());
    sb = new StringBuilder();
    decimal amt = 0;
    //if (!IsPostBack)
    if (Clsusr != null)
    {
        menu.InnerHtml = Common.Generate_newmenu();
        List<EteMyUserCart> Lst = (new
Products()).GetCartbyUserId(Convert.ToInt16(Clsusr.UserId));
        if (Lst.Count == 0)
        {
            shopcart.InnerHtml = "<i class='fab fa-opencart'></i>";
        }
    }

```

```

cartnos.Text = Convert.ToString(Lst.Count());
sb.Append("\n <ul class='list-unstyled mCustomScrollbar' data-mcs-theme='minimal-
dark'> ");
foreach (var list in Lst)
{
    sb.Append("    <li> \n");
    sb.Append("        <img src='" + list.ImagePath + "' alt='"> \n");
    sb.Append("        <button type='button' class='close'>x</button> \n");
    sb.Append("        <div class='overflow-h'> \n");
    sb.Append("            <span>" + list.SellerDescription + "</span> \n");
    sb.Append("            <small>1 x Rs." + list.Price + "</small> \n");
    sb.Append("        </div> \n");
    sb.Append("    </li> \n");
    amt = amt + (decimal)(list.Price);
}

sb.Append("    <li> \n");
sb.Append("    <div class='subtotal'>\n");
sb.Append("    <div class='overflow-h mb10'> \n");
sb.Append("    <span>Subtotal</span> \n");
sb.Append("    <span class='pull-right subtotal-cost'>Rs." + (amt) + "</span>
\n");
sb.Append("    </div> \n");
sb.Append("    <div class='row'> \n");
sb.Append("    <div class='col-xs-6'>");
sb.Append("    <a href='MyCart.aspx' class='btn btn-product cd-cart-btn'>View
Cart</a> \n");
sb.Append("    </div> \n");
sb.Append("    <div class='col-xs-6'>");
sb.Append("    <a href='Checkout.aspx?PageType=Pay&Amt=" + amt + "' class='btn
btn-product cd-cart-btn pull-right'>Checkout</a> \n");
sb.Append("    </div> \n");

```

```

sb.Append(" </div> \n");
sb.Append(" </div> \n");
sb.Append(" </li> \n");
sb.Append(" </ul> \n");
//objlog.Trace_Log(1, "Cart_menu", "1", "ShoppingMasterpage", sb.ToString());

cart.InnerHtml = sb.ToString();
sb = new StringBuilder();
sb.Append(" <div class='widget mb30'> ");
sb.Append(" <h6 class='text-uppercase bottom-line'>Top Rated</h6> ");
sb.Append(" </div> ");

//List<ProductProductSeller> top1 = new Products().GetProductSellerbySearch("",
1);

//var top3 = (from c in top1 select c).Take(3);
//foreach (var tr in top3)
//{
// sb.Append(" <div class='small-thumb-proudct mb30'> ");
// sb.Append(" <div class='thumb-product'> ");
// sb.Append(" <img class='thumb-product-img img-responsive' src="" +
tr.ImagePath + "" alt=""> ");
// //sb.Append(" <div class='thumb-product-inner'> ");
// sb.Append(" <span class='thumb-product-cat'>" + tr.CatName + "</span>
");
// sb.Append(" <h6><a href='shoppingdetail.aspx?pid=" + tr.PSId + "'>" +
tr.ProductDescription + "</a></h6> ");
// sb.Append(" </div> ");
// sb.Append(" <ul class='list-inline overflow-h'> ");
// sb.Append(" <li class='thumb-product-price line-through'>" + tr.CurPrice +
"</li> ");
// sb.Append(" <li class='thumb-product-price'>" + tr.CurPrice + "</li> ");
// sb.Append(" </ul> ");
// //sb.Append(" </div> ");
// sb.Append(" </div> ");
//}

```

```

        //popprods.InnerHtml = sb.ToString();
    }
}
protected void guest_login(string user)
{

    string City = "Hyderabad";// _City;
    string Country = "India";// _Country;

    BALCreateUser objlogin = new BALCreateUser();

    DataTable status_login = objlogin.LoginGuest(user, user,
Request.ServerVariables["REMOTE_ADDR"], "", City, Country);

    string login_status = status_login.Rows[0]["UserStatus"].ToString();

    ClsLoggedInUser objLoggedInUser;
    if (login_status == "True")
    {
        objLoggedInUser = new ClsLoggedInUser();

        objLoggedInUser.UserId = status_login.Rows[0]["UserId"].ToString();
        objLoggedInUser.UserName = status_login.Rows[0]["UserName"].ToString();
        objLoggedInUser.GroupId = status_login.Rows[0]["GroupId"].ToString();
        objLoggedInUser.LoginStatus = "True";
        objLoggedInUser.LoginID = Convert.ToInt32(status_login.Rows[0]["LoginId"]);
        objLoggedInUser.DealerId = Convert.ToInt32(status_login.Rows[0]["DealerId"]);
        objLoggedInUser.RoleId = Convert.ToInt32(status_login.Rows[0]["RoleId"]);
        objLoggedInUser.BrandId = 1; //Convert.ToInt32(status_login.Rows[0]["BrandId"]);

        Session["LoggedUser_Specs"] = objLoggedInUser;

        FormsAuthentication.SetAuthCookie(objLoggedInUser.UserName, false);
        HttpCookie myCookie = new HttpCookie("DateOrWedCookie");
    }
}

```

```
DateTime now = DateTime.Now;
myCookie.Value = objLoggedInUser.UserName;
myCookie.Expires = now.AddYears(1);
```

```
Response.Cookies.Add(myCookie);
```

```
}
```

```
}
```

```
}
```

```
***** ServicesMasterPage.master *****
```

```
<%@ Master Language="C#" AutoEventWireup="true"
CodeFile="ServicesMasterPage.master.cs" Inherits="ServicesMasterPage" %>
```

```
<!DOCTYPE html>
```

```
<!--[if !IE]><!--> <html lang="en"> <!--<![endif]-->
```

```
<head>
```

```
<title>DateOrWed Shopping</title>
```

```
<!-- META -->
```

```
<meta charset="utf-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
```

```
<meta name="description" content="Fstore Bootstrap HTML5 eCommerce Template">
```

```
<meta name="author" content="Craftdzine">
```

```
<!-- FAVICON -->
```

```
<link rel="shortcut icon" href="shopping/img/favicon.png"/>
```

```
<link rel="stylesheet" href="shopping/css/bootstrap.min.css"/>
```

```
<link rel="stylesheet" href="shopping/css/font-awesome.min.css"/>
```

```
<link rel="stylesheet" href="shopping/font/fontawesome/css/all.css"/>
```

```
<link rel="stylesheet" href="shopping/css/magnific-popup.css"/>
```

```
<link rel="stylesheet" href="plugins/FlexSlider/jquery.flexslider.css"/>
```

```
<link rel="stylesheet" href="plugins/owl-carousel/owl.carousel.css"/>
```



```

<link rel="stylesheet" href="plugins/owl-carousel/owl.theme.css"/>
<link rel="stylesheet" type="text/css" href="shopping/css/slick.css"/>
<link rel="stylesheet" type="text/css" href="shopping/css/slick-theme.css"/>
<link rel="stylesheet" href="shopping/css/style.css"/>
<link rel="stylesheet" href="shopping/css/animate.css"/>
<link rel="stylesheet" href="font/fontawesome/css/all.css">
<link href="css/style.css" rel="stylesheet">
  <script type="text/javascript" src="shopping/js/jquery.min.js"></script>
  <!-- BOOTSTRAP -->
  <script type="text/javascript" src="shopping/js/bootstrap.min.js"></script>
  <!-- OWL CAROUSEL -->
  <script type="text/javascript" src="plugins/owl-carousel/owl.carousel.min.js"></script>
  <!-- FLEXSLIDER -->
  <script type="text/javascript" src="plugins/FlexSlider/jquery.flexslider-min.js"></script>
  <!-- MAGNIFIC POPUP-->
  <script type="text/javascript" src="shopping/js/jquery.magnific-popup.min.js"></script>
  <!-- MAGNIFIC POPUP -->
  <script type="text/javascript" src="shopping/js/magnific-popup.min.js"></script>
  <!-- slick-->
  <script type="text/javascript" src="shopping/js/slick.js"></script>
  <!-- slick-slider-->
  <script type="text/javascript" src="shopping/js/slick-slider.js"></script>
  <!-- plugins-->
  <script type="text/javascript" src="shopping/js/plugins.min.html"></script>
  <!-- CUSTOM-->
  <script type="text/javascript" src="shopping/js/custom.js"></script>
<%-- <script type="text/javascript">
  import { mdbContainer, mdbRow, mdbIcon, mdbNavbar, mdbNavbarBrand,
mdbNavbarToggler, mdbNavbarNav, mdbNavItem, mdbInput, mdbDropdown,
mdbDropdownToggler, mdbDropdownMenu, mdbDropdownItem } from 'mdbvue';
  export default {
    name: 'MegaMenuPage',
    components: {

```

```
    mdbContainer,  
    mdbRow,  
    mdbIcon,  
    mdbNavbar,  
    mdbNavbarBrand,  
    mdbNavbarToggler,  
    mdbNavbarNav,  
    mdbNavItem,  
    mdbInput,  
    mdbDropdown,  
    mdbDropdownToggle,  
    mdbDropdownMenu,  
    mdbDropdownItem  
  }  
};  
</script>--%>  
<script type="text/javascript">  
  $(function() {  
  
    var $btn = $('#btnTop');  
    var $home = $('#topSection');  
    var startpoint = $home.scrollTop() + $home.height();  
  
    $(window).on('scroll', function() {  
      if($(window).scrollTop() > startpoint) {  
        $btn.show();  
      } else {  
        $btn.hide();  
      }  
    });  
  });  
</script>
```

```
<!-- BOOTSTRAP -->
```

```
<!-- WEB FONTS -->
```

```
<link href='http://fonts.googleapis.com/css?family=Open+Sans:300,400,600,700'  
rel='stylesheet' type='text/css'>
```

```
<link href="http://fonts.googleapis.com/css?family=Raleway:500,800" rel="stylesheet"  
type="text/css" media="all" />
```

```
<asp:ContentPlaceHolder ID="CPH_MainMaster_Header" runat="server">
```

```
</asp:ContentPlaceHolder>
```

```
</head>
```

```
<body>
```

```
<form id="form1" runat="server" >
```

```
    <asp:ScriptManager ID="ScriptManager1" runat="server">
```

```
    </asp:ScriptManager>
```

```
<!-- Site Wrapper -->
```

```
<div class="wrapper">
```

```
<!-- Header -->
```

```
<div class="header header-static">
```

```
    <!-- Topbar -->
```

```
    <div class="topbar">
```

```
        <div class="container">
```

```
            <div class="row">
```

```
                <div class="col-sm-6">
```

```
                    <!-- Topbar Navigation -->
```

```
                    <ul class="topbar-left">
```

```
<li>
  <a>admin@dateorwed.com</a>
</li>
<li>
  <a>+91 - 7893642582</a>
</li>
```

```
</ul><!--topbar-left-->
```

```
</div>
```

```
<div id="account" class="col-sm-6" runat="server">
```

```
<ul class="list-inline topbar-right pull-right">
```

```
<li><a href="#">Welcome Guest</a></li>
```

```
<li><a href="MyWishList.aspx"><i class="fas fa-lock"></i>Wishlist
(0)</a></li>
```

```
<li><a href="MLB_Login.aspx">Login</a> | <a
href="ManageUser.aspx">Register</a></li>
```

```
</ul>
```

```
</div>
```

```
</div>
```

```
<div class="collapse navbar-collapse" id="navbarSupportedContent-4">
```

```
<div class="search-head text-center">
```

```
<div class="row">
```

```
<div class="col-lg-12">
```

```
<div class="row">
```

```
<div class="col-lg-2 col-md-2 col-sm-12 p-0 bg-white">
```

```
<a class="navbar-brand" href="Home.aspx">
```

```

```

```
</a>
```

```
</div>
```

```
<div class="col-lg-2 col-md-2 col-sm-10 p-1 mr-0">
```

```
<asp:DropDownList id="ddl_location" runat="server" Width="190px" Font-Size="Large" Height="52px">
```

```
<asp:ListItem Value="0">--- Select Location ---</asp:ListItem>
```

```
<asp:ListItem Value="1"> Hyderabad </asp:ListItem>
```

```
<asp:ListItem Value="2"> Rangareddy </asp:ListItem>
```

```
<asp:ListItem Value="3"> Medchal </asp:ListItem>
```

```
<asp:ListItem Value="4"> Patancheru </asp:ListItem>
```

```
</asp:DropDownList>
```

```
</div>
```

```
<div class="col-lg-5 col-md-5 col-sm-12 p-0">
```

```
<asp:TextBox ID="txt_SearchString" runat="server" CssClass="form-control search-stl" Width="570px" Height="37px" ></asp:TextBox>
```

```
</div>
```

```
<div class="col-lg-2 col-md-2 col-sm-8 p-0">
```

```
<button id="Button1" type="button" class="btn btn-secondary wrn-btn mt-0 ml-0" onclick="JavaScript.GetURLSrting1()" ><i class='fa fa-search'></i>&nbsp;&nbsp;&nbsp;Search</button>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<script type="text/javascript">
```

```
function GetURLSrting1() {
```

```
//      alert('GetURLSrting-1');
```

```
var fullURL = (location.pathname + location.search).substr(1);
```

```
var loc = document.getElementById("<%=ddl_location.ClientID%>").value;
```

```
var EnCodedFullURL = encodeURIComponent(fullURL);
```

```
var SearchString = document.getElementById("<%=txt_SearchString.ClientID%>").value;
```

```
var url = 'ServicesList.aspx?SrchStrng=' + encodeURIComponent(SearchString) + '&loc=' +
encodeURIComponent(loc);
//$(location).attr('href', url);
window.location.href = url;
event.preventDefault();
}
</script>
</div>
</div>
```

```
</div><!--container-->
</div>
```

```
<!-- End Topbar -->
```

```
<!-- Navbar -->
```

```
<div class="navbar navbar-default mega-menu" role="navigation">
  <div class="container-fluid">
    <div class="navbar-header">
      <button type="button" class="navbar-toggle" data-toggle="collapse" data-
target=".cd-navbar-collapse">
        <span class="sr-only">Toggle navigation</span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
      </button>
      <%--
        <a class="navbar-brand" href="Home.aspx">
          
        </a--%>
      </div>
    </div>
  <!-- Shopping Cart -->
```

```
<div class="shop-badge badge-icons pull-right">
  <a href="#"><i class="fa fa-shopping-bag"></i></a>
  <span class="badge badge-sea rounded-x">
    <asp:Label ID="cartnos" runat="server" Text="(0)"></asp:Label></span>
    <div id="cart" runat="server" class="badge-open">
```

```
</div>
```

```
</div>
```

```
<!--End Shopping Cart-->
```

```
<!--menu strts-->
```

```
<div id="menu" class="collapse navbar-collapse cd-navbar-collapse pull-left"
runat="server">
```

```
<!-- Nav Menu -->
```

```
<div>
```

```
</div>
```

```
</div>
```

<!-- End Nav Menu -->

<%-- menu ends--%>

</div>

<div class="col-md-3" id="sidebar">

<%--<div class="m-list"><h5><i class="fas fa-list"></i> Services Offered</h5></div>--%>

<div id="menu1" runat="server" class="list-group panel" style="display:none">

 <i class='fas fa-window-restore'></i> Services Offered <i class='fas fa-chevron-right' style='font-size: 12px;'></i>

<div class='collapse' id='menuoth'>

 <i class='fas fa-vihara'></i> Venue <i class='fas fa-chevron-right' style='font-size: 12px;'></i>

<div class='collapse' id='menu406'>

<i class='fas '></i>A/c Venues

<i class='fas '></i>Open Air Venues

<i class='fas '></i>Venues with Catering

<i class='fas '></i>High cost venues

<i class='fas '></i>Low cost venues

<i class='fas '></i>Banquet Halls

</div>

[<i class='fas fa-pizza-slice'></i> Food <i class='fas fa-chevron-right' style='font-size: 12px;'></i>](#menu404)

<div class='collapse' id='menu404'>

</div>

[<i class='fas fa-holly-berry'></i> Decors <i class='fas fa-chevron-right' style='font-size: 12px;'></i>](#menu405)

<div class='collapse' id='menu405'>

</div>

[<i class='fas fa-birthday-cake'></i> Cakes & Bakery Items <i class='fas fa-chevron-right' style='font-size: 12px;'></i>](#menu409)

<div class='collapse' id='menu409'>

</div>

[<i class='fas fa-music'></i> Music <i class='fas fa-chevron-right' style='font-size: 12px;'></i>](#menu407)

<div class='collapse' id='menu407'>

<i class='fas '>></i>Marriage Music

<i class='fas '>></i>Orchestra

</div>

 <i class='fas fa-female'></i> Makeup <i class='fas fa-chevron-right' style='font-size: 12px;'></i>

<div class='collapse' id='menu419'>

<i class='fas '>></i>Makeup

</div>

 <i class='fas fa-allergies'></i> Mehendi <i class='fas fa-chevron-right' style='font-size: 12px;'></i>

<div class='collapse' id='menu420'>

<i class='fas '>></i>Mehendi

</div>

 <i class='fas fa-video'></i> Videographer <i class='fas fa-chevron-right' style='font-size: 12px;'></i>

<div class='collapse' id='menu415'>

<i class='fas '>></i>Videographer

</div>

 <i class='fas fa-camera-retro'></i> Photographer <i class='fas fa-chevron-right' style='font-size: 12px;'></i>

<div class='collapse' id='menu414'>

<i class='fas '>></i>Photographer

</div>

 <i class='fas fa-address-card'></i> Wedding Invitation Cards <i class='fas fa-chevron-right' style='font-size: 12px;'></i>

```
<div class='collapse' id='menu417'>
```

```
<a href='/ServicesList.aspx?sid=470' class='list-group-item' data-parent='#menu417'><i class='fas ' ></i>Wedding Invitation Cards</a>
```

```
</div>
```

```
<a href='#menu411' class='list-group-item collapsed' data-toggle='collapse' data-parent='#sidebar' aria-expanded='false'> <span class='hidden-sm-down'><i class='fas fa-tshirt'></i> Dress & Attire <i class='fas fa-chevron-right' style='font-size: 12px;'></i></span></a>
```

```
<div class='collapse' id='menu411'>
```

```
<a href='/ServicesList.aspx?sid=463' class='list-group-item' data-parent='#menu411'><i class='fas ' ></i>Dress & Attire</a>
```

```
</div>
```

```
<a href='#menu413' class='list-group-item collapsed' data-toggle='collapse' data-parent='#sidebar' aria-expanded='false'> <span class='hidden-sm-down'><i class='fas fa-gem'></i> Jewelry <i class='fas fa-chevron-right' style='font-size: 12px;'></i></span></a>
```

```
<div class='collapse' id='menu413'>
```

```
<a href='/ServicesList.aspx?sid=466' class='list-group-item' data-parent='#menu413'><i class='fas ' ></i>Jewelry Rentals</a>
```

```
</div>
```

```
<a href='#menu410' class='list-group-item collapsed' data-toggle='collapse' data-parent='#sidebar' aria-expanded='false'> <span class='hidden-sm-down'><i class='fas fa-gift'></i> Favors & gifts <i class='fas fa-chevron-right' style='font-size: 12px;'></i></span></a>
```

```
<div class='collapse' id='menu410'>
```

```
<a href='/ServicesList.aspx?sid=462' class='list-group-item' data-parent='#menu410'><i class='fas ' ></i>Favors & gifts</a>
```

```
</div>
```

```
<a href='#menu412' class='list-group-item collapsed' data-toggle='collapse' data-parent='#sidebar' aria-expanded='false'> <span class='hidden-sm-down'><i class='fas fa-renren'></i> Wedding item Rentals <i class='fas fa-chevron-right' style='font-size: 12px;'></i></span></a>
```

```
<div class='collapse' id='menu412'>
```

```
<a href='/ServicesList.aspx?sid=464' class='list-group-item' data-parent='#menu412'><i class='fas ' ></i>Wedding item Rentals</a>
```

```
<a href='/ServicesList.aspx?sid=465' class='list-group-item' data-parent='#menu412'><i class='fas ' ></i>Rentals</a>
```

```
</div>
```

```
<a href='#menu416' class='list-group-item collapsed' data-toggle='collapse' data-parent='#sidebar' aria-expanded='false'> <span class='hidden-sm-down'><i class='fas fa-
```

[globe-asia](#) > </i> Wedding Planner <i class='fas fa-chevron-right' style='font-size: 12px;'></i>

<div class='collapse' id='menu416'>

<i class='fas ' ></i>Wedding Planner

</div>

 <i class='fas fa-bus-alt'></i> Transport <i class='fas fa-chevron-right' style='font-size: 12px;'></i>

<div class='collapse' id='menu408'>

<i class='fas ' ></i>Local Travel Planning

<i class='fas ' ></i>Internatonal Travel Planning

</div>

 <i class='fas fa-hotel'></i> Travel Agents (Hotel, Travel) <i class='fas fa-chevron-right' style='font-size: 12px;'></i>

<div class='collapse' id='menu418'>

<i class='fas ' ></i>Travel Agents (Hotel, Travel)

</div>

 <i class='fas fa-praying-hands'></i> Recital Material Packing <i class='fas fa-chevron-right' style='font-size: 12px;'></i>

<div class='collapse' id='menu421'>

<i class='fas ' ></i>Recital Material Packing

</div>

 <i class='fas fa-user-friends'></i> Honey Moon Packages <i class='fas fa-chevron-right' style='font-size: 12px;'></i>

<div class='collapse' id='menu422'>

<i class='fas ' ></i>Honey Moon Packages

</div>

</div>

</div>

</div>

<%-- menu ends--%>

</div>

</div>

<!-- End Navbar -->

<!-- End Header-->

<!-- Start Parallax -->

<asp:ContentPlaceHolder ID="ContentMainPage" runat="server">

</asp:ContentPlaceHolder>

<!-- Top Logo/Messages Div/Section Start -->

<asp:ContentPlaceHolder ID="CPH_TopMessages" runat="server">

<div class="container-fluid">

<div class="row-fluid">

<div class="span10">

<div class="row">

<div id="dvMsgBox" class="span12 ordersave" runat="server" style="display: block;">

<asp:Label ID="Label1" runat="server" Text=""></asp:Label>

</div>

</div>

</div>

</div>

```
</div>
</asp:ContentPlaceHolder>
<!-- Top Logo/Messages Div/Section End -->
```

```
<div class="container-fluid">
  <div class="row-fluid">

    <!-- Left Bar Div/Section Start -->
    <!-- Left Bar Div/Section End -->

    <!-- Main Content Div/Section Start -->
    <asp:ContentPlaceHolder ID="CPH_MainContent" runat="server">

      </asp:ContentPlaceHolder>
      <!-- Main Content Div/Section End -->
    </div>
  </div>
```

```
</div>
```

```
<%-- <section class="home-main-contant-style3 "> <%--bg-white--%>
  <div class="parallax-block parallaxBg">
    <div class="container">
      <div class="parallax-block-text">
        <p>We are "DateOrWed.com" <span class="color-green">Thanks for
checking out our profile.</span> <br> Thank you! </p>
      </div>
      <div class="pre-btn-group">
        <a class="btn btn-product mr15" href="#">Shop Now</a>
        <a class="btn btn-product" href="#">get a free quote</a>
      </div>
    </div>
  </div>
</div>
```

```
<%-- </section>--%>
```

```
<!-- End Parallax -->
```

```
<!-- START NEW -->
```

```
<section class="home-main-contant-style3 bg-white">
```

```
<div class="container">
```

```
<div class="widget">
```

```
<h6 class="text-uppercase bottom-line">Our Brand</h6>
```

```
</div>
```

```
<div class="row">
```

```
<div class="brand slider">
```

```
<div class="col-md-2">
```

```
<a href="#"></a>
```

```
</div>
```

```
<div class="col-md-2">
```

```
<a href="#"></a>
```

```
</div>
```

```
<div class="col-md-2">
```

```
<a href="#"></a>
```

```
</div>
```

```
<div class="col-md-2">
```

```
<a href="#"></a>
```

```
</div>
```

```
<div class="col-md-2">
```

```
<a href="#"></a>
```

```
</div>
```

```
<div class="col-md-2">
```

```
        <a href="#"></a>
    </div>
    <div class="col-md-2">
        <a href="#"></a>
    </div>
    <div class="col-md-2">
        <a href="#"></a>
    </div>
    <div class="col-md-2">
        <a href="#"></a>
    </div>
    <div class="col-md-2">
        <a href="#"></a>
    </div>
</div>
</div>
</div>
</div>
</section>
<!-- END NEW -->
```

```
<!-- Start Suvbscribe -->
<div class="news-subscribe">
    <div class="container">
        <div class="row">
            <div class="col-md-4">
                <h3>subscribe to <strong>newsletter</strong></h3>
            </div>
            <div class="col-md-4 newsletter-form-block">
                <div class="input-group">
```



```

        <input type="text" class="form-control" placeholder="Email your
email...">
        <span class="input-group-btn">
            <button class="btn" type="button"><i class="fa fa-envelope-
o"></i></button>
        </span>
    </div>
</div>
<div class="col-md-4 col-xs-12">
    <div class="social-footer">
        <ul class="social-icons social-icons-simple">
            <li><a
href="https://www.facebook.com/profile.php?id=100077266105117"><i class="fab fa-
facebook"></i></a></li>
            <li><a href="https://twitter.com/dateorwed"><i class="fab fa-
twitter"></i></a></li>
            <li><a href="#"><i class="fab fa-tumblr"></i></a></li>
            <li><a href="https://instagram.com/admin.dateorwed"><i class="fab
fa-instagram"></i></a></li>
            <li><a href="www.linkedin.com/in/administrator-dateorwedorwed-
780a58231"><i class="fab fa-linkedin"></i></a></li>
        </ul>
    </div>
</div>
</div>
</div>
</div>
</div>
<!--End Suvbscribe -->
<asp:ContentPlaceHolder ID="ContentPlaceHolder1" runat="server">
    <div class="container-fluid">
        <div class="row-fluid">
            <%-- <div class="row-fluid" style="padding-top:
20px;"-->%>
                <div class="span2" id="Div1" runat="server" style="display: none;">

```

```

<%--          
--%>

    </div>
    <div class="span10">
        <div class="row">
            <div id="Div2" class="span12 ordersave" runat="server" style="display:
block;">
                <asp:Label ID="Label2" runat="server" Text=""></asp:Label>
            </div>
        </div>
    </div>
</div>
</div>
</div>
</asp:ContentPlaceHolder>

```

```

<!-- START FOOTER -->
<footer class="footer">
    <div class="container">
        <div class="row">
            <!-- CONTACT WIDGET -->
            <div class="col-md-2 col-sm-2 col-xs-12">
                <div class="widget">
                    <h6 class="montserrat text-uppercase">&nbsp;</h6>
                    
                </div>
            </div>
            <!-- END CONTACT WIDGET -->

            <!-- CONTACT WIDGET -->
            <div class="col-md-3 col-sm-6 col-xs-12">
                <div class="widget">

```

```
<h6 class="text-uppercase bottom-line">Contact Us</h6>
```

```
<address>
```

```
<p>H.No: 2-95, Shanthinagar, Hyderabad, Telangana, India</p><br>
```

```
<p>Phone: (+91) 7893642582</p>
```

```
<p>E-mail: <a href="mailto:admin@dateorwed.com">admin@dateorwed.com</a></p>
```

```
</address>
```

```
</div>
```

```
</div>
```

```
<!-- END CONTACT WIDGET -->
```

```
<!-- CATEGORIES WIDGET -->
```

```
<div class="col-md-2 col-sm-5 col-xs-12">
```

```
<div class="widget">
```

```
<h6 class="montserrat text-uppercase bottom-line">Users</h6>
```

```
<ul class="icons-list">
```

```
<li>
```

```
<a href="#">My Profile</a>
```

```
</li>
```

```
</ul>
```

```
</div>
```

```
</div>
```

```
<!-- END CATEGORIES WIDGET -->
```

```
<!-- CATEGORIES WIDGET -->
```

```
<div class="col-md-2 col-sm-5 col-xs-12">
```

```
<div class="widget">
```

```
<h6 class="montserrat text-uppercase bottom-line">About Us</h6>
```

```
<ul class="icons-list">
```

```
<li><a href="#">About Us</a></li>
```

```
<li><a href="#">FAQs</a></li>
```

```
        <li><a href="#">Feedback</a></li>
    </ul>
</div>
</div>
<!-- END CATEGORIES WIDGET -->
```

```
<!-- RECENT POSTS WIDGET -->
<div class="col-md-3 col-sm-6 col-xs-12">
    <div class="widget">
        <h6 class="text-uppercase bottom-line">Service Providers</h6>
        <ul class="icons-list">
            <li><a href="#">Become a Service Provider</a></li>
            <li><a href="#">Dashboard</a></li>
            <li><a href="#">Emergency Contacts</a></li>
        </ul>
    </div>
</div>
<!-- END RECENT POSTS WIDGET -->
```

```
</div>
```

```
<hr style="border-top: 1px solid #ddd; padding-bottom: 10px;">
```

```
<div class="row">
    <!-- CONTACT WIDGET -->
    <div class="col-md-3 col-sm-6 col-xs-12">
        <div class="widget">
            <h6 class="text-uppercase bottom-line">Top Brands</h6>
            <ul class="icons-list">
```

```
<li><a href="#">Biba</a></li>
<li><a href="#">Rain & Rainbow</a></li>
<li><a href="#">Aks</a></li>
<li><a href="#">Soch</a></li>
<li><a href="#">Global Desi</a></li>
</ul>
</div>
</div>
<!-- END CONTACT WIDGET -->
```

```
<!-- CATEGORIES WIDGET -->
<div class="col-md-3 col-sm-6 col-xs-12">
  <div class="widget">
    <h6 class="montserrat text-uppercase bottom-line">Categories</h6>
    <ul class="icons-list">
      <li><a href="#">Fashion <span class="pull-right">12</span></a></li>
      <li><a href="#">Blogger <span class="pull-right">11</span></a></li>
      <li><a href="#">Sport <span class="pull-right">10</span></a></li>
      <li><a href="#">Photography <span class="pull-
right">14</span></a></li>
      <li><a href="#">Popular <span class="pull-right">15</span></a></li>
    </ul>
  </div>
</div>
<!-- END CATEGORIES WIDGET -->
```

```
<!-- CATEGORIES WIDGET -->
<div class="col-md-3 col-sm-6 col-xs-12">
  <div class="widget">
    <h6 class="montserrat text-uppercase bottom-line">Quick Links</h6>
    <ul class="icons-list">
      <li><a href="#">About</a></li>
```

```
<li><a href="#">Home</a></li>
<li><a href="#">New Products</a></li>
<li><a href="#">Teams</a></li>
<li><a href="#">Popular</a></li>
</ul>
</div>
</div>
<!-- END CATEGORIES WIDGET -->
```

```
<!-- RECENT POSTS WIDGET -->
```

```
<div class="col-md-3 col-sm-6 col-xs-12">
```

```
<div class="widget">
```

```
<h6 class="text-uppercase bottom-line">Popular Products</h6>
```

```
<ul class="recent-posts">
```

```
<li>
```

```
<div class="widget-posts-image">
```

```
<a href="#"></a>
```

```
</div>
```

```
<div class="widget-posts-body">
```

```
<h6 class="widget-posts-title"><a href="#">Printed Summer  
Dress</a></h6>
```

```
<div class="widget-posts-meta">₹5220.45</div>
```

```
</div>
```

```
</li>
```

```
<li>
```

```
<div class="widget-posts-image">
```

```
<a href="#"></a>
```

```
</div>
```

```
<div class="widget-posts-body">
```

```
<h6 class="widget-posts-title"><a href="#">Printed Summer  
Dress</a></h6>
```

```
<div class="widget-posts-meta">₹2300.45</div>
```

```
        </div>
    </li>
    <li>
        <div class="widget-posts-image">
            <a href="#"></a>
        </div>
        <div class="widget-posts-body">
            <h6 class="widget-posts-title"><a href="#">Printed Summer
Dress</a></h6>
            <div class="widget-posts-meta">₹1100.45</div>
        </div>
    </li>
</ul>
</div>
</div>
<!-- END RECENT POSTS WIDGET -->
```

```
</div>
```

```
<!-- CONTACT WIDGET -->
```

```
<div class="copyright">
```

```
    <div class="text-center">
```

```
        <p>© 2019 Copyrights All Rights Reserved @ <a target="_blank"
href="Home.aspx">DateOrWed.com</a></p>
```

```
    </div>
```

```
</div>
```

```
</div>
```

```
</footer>
```

```
<!-- END FOOTER-->
```

```
<asp:ContentPlaceHolder ID="ContentFooter" runat="server">
```

```
</asp:ContentPlaceholder>
```

```
</form>
```

```
</body>
```

```
</html>
```

```
***** ServicesMasterPage.master.cs*****
```

```
using System;
```

```
using System.Collections.Generic;
```

```
using System.Linq;
```

```
using System.Web;
```

```
using System.Web.UI;
```

```
using System.Web.UI.WebControls;
```

```
using DowBEntity;
```

```
using DowBAL;
```

```
using System.Text;
```

```
using DowDAL;
```

```
using System.Data;
```

```
using System.Web.Security;
```

```
public partial class ServicesMasterPage : System.Web.UI.MasterPage
```

```
{
```

```
    string UserName;
```

```
    DDLogTracer objlog = new DDLogTracer();
```

```
    protected void Page_Load(object sender, EventArgs e)
```

```
    {
```

```
        string logpath = Server.MapPath("~/log/" + "Log" +  
Convert.ToString(DateTime.Today.Day) + ".txt");
```

```
        Session["Logpath"] = logpath;
```

```
        StringBuilder sb = new StringBuilder();
```

```
        menu.InnerHtml = Common.Generate_servicemenu();
```



```

ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];

string user;
if (Clsusr != null)
{
    UserName = Clsusr.UserName;
    int dftcnt = new Products().GetSLItems(Convert.ToInt16(Clsusr.UserId), true);
    sb.Append("<ul class='list-inline topbar-right pull-right'\>\n");
    sb.Append("<li><a href='UserDetail.aspx'\> Welcome " + UserName + "</a></li>\n");
    sb.Append("<li><a href='MyWishList.aspx'\><i class='fas fa-hand-holding-heart'\></i>Wishlist (" + Convert.ToString(dftcnt) + ")</a></li>\n");
    sb.Append("<li><a href='MyItinerary.aspx'\><i class='fas fa-calendar-alt'\></i>Itinerary (" + "0" + ")</a></li>\n");
    sb.Append("<li><a href='logout.aspx'\><i class='fas fa-lock'\></i>Logout</a> | <a href='UserDetail.aspx'\><i class='fas fa-user'\></i>Profile</a></li>");
    account.InnerHtml = sb.ToString();

    //      Response.Redirect("MLB_Login.aspx");
}
PopDlIs ObjPD = new PopDlIs();
//ObjPD.BindDropDowns(ddl_location, (new PopDlIs()).GetProductCats());
sb = new StringBuilder();
decimal amt = 0;
if (Clsusr != null)
{
    List<EteMyUserCart> Lst = (new
Products()).GetCartbyUserId(Convert.ToInt16(Clsusr.UserId));
    cartnos.Text = Convert.ToString(Lst.Count());
    sb.Append("\n <ul class='list-unstyled mCustomScrollbar' data-mcs-theme='minimal-dark'\> ");
    foreach (var list in Lst)
    {

```

```

        sb.Append("        <li> \n");
        sb.Append("        <img src=\"" + list.ImagePath + "\" alt=\"" + list.ImageAlt + "\"> \n");
        sb.Append("        <button type='button' class='close'>x</button> \n");
        sb.Append("        <div class='overflow-h'> \n");
        sb.Append("        <span>" + list.SellerDescription + "</span> \n");
        sb.Append("        <small>1 x Rs." + list.Price + "</small> \n");
        sb.Append("        </div> \n");
        sb.Append("    </li> \n");
    amt = amt + (decimal)(list.Price);

}

sb.Append("    <li> \n");
sb.Append("    <div class='subtotal'>\n");
sb.Append("    <div class='overflow-h mb10'> \n");
sb.Append("    <span>Subtotal</span> \n");
sb.Append("    <span class='pull-right subtotal-cost'>Rs." + (amt) + "</span> \n");
sb.Append("    </div> \n");
sb.Append("    <div class='row'> \n");
sb.Append("    <div class='col-xs-6'>");
sb.Append("    <a href='MyCart.aspx' class='btn btn-product cd-cart-btn'>View
Cart</a> \n");
sb.Append("    </div> \n");
sb.Append("    <div class='col-xs-6'>");
sb.Append("    <a href='Checkout.aspx?PageType=Pay&Amt=" + amt + "' class='btn
btn-product cd-cart-btn pull-right'>Checkout</a> \n");
sb.Append("    </div> \n");
sb.Append("    </div> \n");
sb.Append("    </div> \n");
sb.Append("    </li> \n");
sb.Append("    </ul> \n");

```

```

//objlog.Trace_Log(1, "Cart_menu", "1", "ShoppingMasterpage", sb.ToString());

cart.InnerHtml = sb.ToString();

}

else
{
    HttpCookie myCookie = Request.Cookies["DateOrWedCookie"];
    if (myCookie != null)
    {
        user = myCookie.Value.ToString();
    }
    else
    {
        user = "Guest";
        //Response.Redirect("MLB_login.aspx");
    }

    guest_login(user);

}
}
protected void guest_login(string user)
{

    string City = "Hyderabad";// _City;
    string Country = "India";// _Country;

    BALCreateUser objlogin = new BALCreateUser();

    DataTable status_login = objlogin.LoginGuest(user, user,
Request.ServerVariables["REMOTE_ADDR"], "", City, Country);

```

```
string login_status = status_login.Rows[0]["UserStatus"].ToString();
```

```
ClsLoggedInUser objLoggedInUser;
```

```
if (login_status == "True")
```

```
{
```

```
    objLoggedInUser = new ClsLoggedInUser();
```

```
    objLoggedInUser.UserId = status_login.Rows[0]["UserId"].ToString();
```

```
    objLoggedInUser.UserName = status_login.Rows[0]["UserName"].ToString();
```

```
    objLoggedInUser.GroupId = status_login.Rows[0]["GroupId"].ToString();
```

```
    objLoggedInUser.LoginStatus = "True";
```

```
    objLoggedInUser.LoginID = Convert.ToInt32(status_login.Rows[0]["LoginId"]);
```

```
    objLoggedInUser.DealerId = Convert.ToInt32(status_login.Rows[0]["DealerId"]);
```

```
    objLoggedInUser.RoleId = Convert.ToInt32(status_login.Rows[0]["RoleId"]);
```

```
    objLoggedInUser.BrandId = 1; //Convert.ToInt32(status_login.Rows[0]["BrandId"]);
```

```
    Session["LoggedUser_Specs"] = objLoggedInUser;
```

```
    FormsAuthentication.SetAuthCookie(objLoggedInUser.UserName, false);
```

```
    HttpCookie myCookie = new HttpCookie("DateOrWedCookie");
```

```
    DateTime now = DateTime.Now;
```

```
    myCookie.Value = objLoggedInUser.UserName;
```

```
    myCookie.Expires = now.AddYears(1);
```

```
    Response.Cookies.Add(myCookie);
```

```
}
```

```
}
```

```
}
```

*****VendorMasterPage.master *****

```
<%@ Master Language="C#" AutoEventWireup="true"
CodeFile="VendorMasterPage.master.cs" Inherits="VendorMasterPage" %>
```

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
<meta charset="utf-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
```

```
<meta http-equiv="x-ua-compatible" content="ie=edge">
```

```
<title>MLB | Vendor Management</title>
```

```
<!-- Favicon -->
```

```
<link rel="icon" type="image/gif" href="img/favicon.png">
```

```
<!-- Font Awesome -->
```

```
<link rel="stylesheet" href="Vendor/css/all.css">
```

```
<!-- Bootstrap core CSS -->
```

```
<link rel="stylesheet" href="Vendor/css/bootstrap.min.css">
```

```
<!-- Material Design Bootstrap -->
```

```
<link rel="stylesheet" href="Vendor/css/mdb.min.css">
```

```
<!-- Your custom styles (optional) -->
```

```
<link rel="stylesheet" type="text/css" href="Vendor/css/style.css">
```

```
<style>
```

```
</style>
```

```
</head>
```

```
<body class="fixed-sn white-skin">
```

```
<!-- Main Navigation -->
```

```
<header>
```

```
<!-- Sidebar navigation -->
```

```
<div id="slide-out" class="side-nav sn-bg-4 fixed">
```

```

<ul class="custom-scrollbar">

<!-- Logo -->
<li class="logo-sn waves-effect py-3">
  <div class="text-center">
    <a href="#" class="pl-0"></a>
  </div>
</li>

<!-- Search Form -->
<li>
  <form class="search-form" role="search">
    <div class="md-form mt-0 waves-light">
      <input type="text" class="form-control py-2" placeholder="Search">
    </div>
  </form>
</li>

<!-- Side navigation links -->
<li>
  <ul class="collapsible collapsible-accordion">
    <li class="active">
      <a href="VendorHome.aspx" class="collapsible-header waves-effect"><i class="w-
fa fas fa-tachometer-alt"></i>Dashboard</a>
    </li>
    <li>
      <a class="collapsible-header waves-effect arrow-r">
        <i class="w-fa fas fa-shopping-cart"></i>Shopping Vendor<i class="fas fa-angle-
down rotate-icon"></i>
      </a>
      <div class="collapsible-body">
        <ul>
          <li>

```

```
</a>
    <a href="vendorproductsAdd.aspx" class="waves-effect">Select Categories
</a>
    <li>
        <a href="vendorcolorsAdd.aspx" class="waves-effect">Select Colors </a>
    </li>
    <li>
        <a href="vendorsizesAdd.aspx" class="waves-effect">Select Sizes </a>
    </li>
    <li>
        <a href="vendorproducts.aspx" class="waves-effect">Products
Management</a>
    </li>
    <li>
        <a href="PostProducts.aspx" class="waves-effect">Post Products </a>
    </li>
    <li>
        <a href="ManageOrder.aspx" class="waves-effect">Order Management</a>
    </li>
    <li>
        <a href="ManagePayments.aspx" class="waves-effect">Payments</a>
    </li>
    <li>
        <a href="ManageLogistics.aspx" class="waves-effect">Logistic</a>
    </li>
    <li>
        <a href="ManageOrderRet.aspx" class="waves-effect">Order returned</a>
    </li>
    <li>
        <a href="ManageCommunications.aspx" class="waves-
effect">Communication</a>
    </li>
</ul>
</div>
```

```
</li>
<li>
  <a class="collapsible-header waves-effect arrow-r">
    <i class="w-fa fas fa-user-cog"></i>Service Vendor<i class="fas fa-angle-down
rotate-icon"></i>
  </a>
  <div class="collapsible-body">
    <ul>
      <li>
        <a href="VendorServicesAdd.aspx" class="waves-effect">Select Service Types
</a>
      </li>
      <li>
        <a href="vendorServices.aspx" class="waves-effect">Services
Management</a>
      </li>
      <li>
        <a href="PostServices.aspx" class="waves-effect">Add Services </a>
      </li>
      <li>
        <a href="ManageOrder.aspx" class="waves-effect">Order Management</a>
      </li>
      <li>
        <a href="ManagePayments.aspx" class="waves-effect">Payments</a>
      </li>
      <li>
        <a href="ManageOrderRet.aspx" class="waves-effect">Order Cancellations</a>
      </li>
      <li>
        <a href="ManageCommunications.aspx" class="waves-
effect">Communication</a>
      </li>
    </ul>
  </div>
```



```
</li>
<li>
  <a class="collapsible-header waves-effect arrow-r">
    <i class="w-fa fas fa-tasks"></i>Business Listing Vendor<i class="fas fa-angle-
down rotate-icon"></i>
  </a>
  <div class="collapsible-body">
    <ul>
      <li>
        <a href="#" class="waves-effect">Add the Details Business</a>
      </li>
      <li>
        <a href="#" class="waves-effect">Product Catalog</a>
      </li>
      <li>
        <a href="#" class="waves-effect">Contact Detail</a>
      </li>
    </ul>
  </div>
</li>
</ul>
</li>
</ul>
</li>
<!-- Side navigation links -->

</ul>
<div class="sidenav-bg mask-strong"></div>
</div>
<!-- Sidebar navigation -->

<!-- Navbar -->
<nav class="navbar fixed-top navbar-expand-lg scrolling-navbar double-nav">

<!-- SideNav slide-out button -->
```

```

<div class="float-left">
  <a href="#" data-activates="slide-out" class="button-collapse"><i class="fas fa-bars"></i></a>
</div>

<!-- Breadcrumb -->
<div class="breadcrumb-dn mr-auto">
  <p>Dashboard</p>
</div>

<div class="d-flex change-mode">

  <div class="ml-auto mb-0 mr-3 change-mode-wrapper">
    <button class="btn btn-outline-black btn-sm" id="dark-mode">Change Mode</button>
  </div>

  <!-- Navbar links -->
  <ul class="nav navbar-nav nav-flex-icons ml-auto">

    <!-- Dropdown -->
    <li class="nav-item dropdown notifications-nav">
      <a class="nav-link dropdown-toggle waves-effect" id="navbarDropdownMenuLink"
        data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
        <span class="badge red">3</span> <i class="fas fa-bell"></i>
        <span class="d-none d-md-inline-block">Notifications</span>
      </a>
      <div class="dropdown-menu dropdown-primary" aria-labelledby="navbarDropdownMenuLink">
        <a class="dropdown-item" href="#">
          <i class="far fa-money-bill-alt mr-2" aria-hidden="true"></i>
          <span>New order received</span>
          <span class="float-right"><i class="far fa-clock" aria-hidden="true"></i> 13
min</span>
        </a>

```

```
<a class="dropdown-item" href="#">
  <i class="far fa-money-bill-alt mr-2" aria-hidden="true"></i>
  <span>New order received</span>
  <span class="float-right"><i class="far fa-clock" aria-hidden="true"></i> 33
min</span>
</a>
<a class="dropdown-item" href="#">
  <i class="fas fa-chart-line mr-2" aria-hidden="true"></i>
  <span>Your campaign is about to end</span>
  <span class="float-right"><i class="far fa-clock" aria-hidden="true"></i> 53
min</span>
</a>
</div>
</li>
<li class="nav-item">
  <a class="nav-link waves-effect"><i class="fas fa-envelope"></i> <span
class="clearfix d-none d-sm-inline-block">Contact</span></a>
</li>
<li class="nav-item">
  <a class="nav-link waves-effect"><i class="far fa-comments"></i> <span
class="clearfix d-none d-sm-inline-block">Support</span></a>
</li>
<li class="nav-item dropdown">
  <a class="nav-link dropdown-toggle waves-effect" href="#" id="userDropdown" data-
toggle="dropdown" aria-haspopup="true" aria-expanded="false">
    <i class="fas fa-user"></i> <span class="clearfix d-none d-sm-inline-
block">Profile</span>
  </a>
  <div class="dropdown-menu dropdown-menu-right" aria-
labelledby="userDropdown">
    <a class="dropdown-item" href="#">Log Out</a>
    <a class="dropdown-item" href="#">My account</a>
  </div>
</li>
```

```
</ul>
```

```
<!-- Navbar links -->
```

```
</div>
```

```
</nav>
```

```
<!-- Navbar -->
```

```
<!-- Fixed button -->
```

```
<div class="fixed-action-btn clearfix d-none d-xl-block" style="bottom: 45px; right: 24px;">
```

```
<a class="btn-floating btn-lg red">
```

```
<i class="fas fa-pencil-alt"></i>
```

```
</a>
```

```
<ul class="list-unstyled">
```

```
<li><a class="btn-floating red"><i class="fas fa-star"></i></a></li>
```

```
<li><a class="btn-floating yellow darken-1"><i class="fas fa-user"></i></a></li>
```

```
<li><a class="btn-floating green"><i class="fas fa-envelope"></i></a></li>
```

```
<li><a class="btn-floating blue"><i class="fas fa-shopping-cart"></i></a></li>
```

```
</ul>
```

```
</div>
```

```
<!-- Fixed button -->
```

```
</header>
```

```
<!-- Main Navigation -->
```

```
<!-- Main layout -->
```

```
<div class="container-fluid">
```

```
<!-- Section: Intro -->
```

```
<section class="mt-md-4 pt-md-2 mb-5 pb-4">
```

```
<!-- Grid row -->
```

```
<div class="row">
```

```
<!-- Grid column -->
```

```
<div class="col-xl-3 col-md-6 mb-xl-0 mb-4">
```

```
<!-- Card -->
```

```
<div class="card card-cascade cascading-admin-card">
```

```
<!-- Card Data -->
```

```
<div class="admin-up">
```

```
<i class="far fa-money-bill-alt primary-color mr-3 z-depth-2"></i>
```

```
<div class="data">
```

```
<p class="text-uppercase">sales</p>
```

```
<h4 class="font-weight-bold dark-grey-text">2000$</h4>
```

```
</div>
```

```
</div>
```

```
<!-- Card content -->
```

```
<div class="card-body card-body-cascade">
```

```
<div class="progress mb-3">
```

```
<div class="progress-bar bg-primary" role="progressbar" style="width: 25%" aria-  
valuenow="25" aria-valuemin="0" aria-valuemax="100"></div>
```

```
</div>
```

```
<p class="card-text">Better than last week (25%)</p>
```

```
</div>
```

```
</div>
```

```
<!-- Card -->
```

```
</div>
```

```
<!-- Grid column -->
```

```
<!-- Grid column -->
```

```
<div class="col-xl-3 col-md-6 mb-xl-0 mb-4">
```

```
<!-- Card -->
```

```
<div class="card card-cascade cascading-admin-card">
```

```
<!-- Card Data -->
```

```
<div class="admin-up">
```

```
<i class="fas fa-chart-line warning-color mr-3 z-depth-2"></i>
```

```
<div class="data">
```

```
<p class="text-uppercase">subscriptions</p>
```

```
<h4 class="font-weight-bold dark-grey-text">200</h4>
```

```
</div>
```

```
</div>
```

```
<!-- Card content -->
```

```
<div class="card-body card-body-cascade">
```

```
<div class="progress mb-3">
```

```
<div class="progress-bar red accent-2" role="progressbar" style="width: 25%"  
aria-valuenow="25" aria-valuemin="0" aria-valuemax="100"></div>
```

```
</div>
```

```
<p class="card-text">Worse than last week (25%)</p>
```

```
</div>
```

```
</div>
```

```
<!-- Card -->
```

```
</div>
```

```
<!-- Grid column -->
```

```
<!-- Grid column -->
```

```
<div class="col-xl-3 col-md-6 mb-md-0 mb-4">
```

```
<!-- Card -->
```

```
<div class="card card-cascade cascading-admin-card">
```

```
<!-- Card Data -->
```

```
<div class="admin-up">
```

```
<i class="fas fa-chart-pie light-blue lighten-1 mr-3 z-depth-2"></i>
```

```
<div class="data">
```

```
<p class="text-uppercase">traffic</p>
```

```
<h4 class="font-weight-bold dark-grey-text">20000</h4>
```

```
</div>
```

```
</div>
```

```
<!-- Card content -->
```

```
<div class="card-body card-body-cascade">
```

```
<div class="progress mb-3">
```

```
<div class="progress-bar red accent-2" role="progressbar" style="width: 75%"  
aria-valuenow="75" aria-valuemin="0" aria-valuemax="100"></div>
```

```
</div>
```

```
<p class="card-text">Worse than last week (75%)</p>
```

```
</div>
```

```
</div>
```

```
<!-- Card -->
```

```
</div>
```

```
<!-- Grid column -->
```

```
<!-- Grid column -->
```

```
<div class="col-xl-3 col-md-6 mb-0">
```

```

<!-- Card -->
<div class="card card-cascade cascading-admin-card">

  <!-- Card Data -->
  <div class="admin-up">
    <i class="fas fa-chart-bar red accent-2 mr-3 z-depth-2"></i>
    <div class="data">
      <p class="text-uppercase">organic traffic</p>
      <h4 class="font-weight-bold dark-grey-text">2000</h4>
    </div>
  </div>

  <!-- Card content -->
  <div class="card-body card-body-cascade">
    <div class="progress mb-3">
      <div class="progress-bar bg-primary" role="progressbar" style="width: 25%" aria-
valuenow="25" aria-valuemin="0" aria-valuemax="100"></div>
    </div>
    <p class="card-text">Better than last week (25%)</p>
  </div>

</div>
<!-- Card -->

</div>
<!-- Grid column -->

</div>
<!-- Grid row -->

</section>
<!-- Section: Intro -->

```



```
<!-- Section: Main panel -->
```

```
<section class="mb-5">
```

```
<!-- Card -->
```

```
<div class="card card-cascade narrower">
```

```
<!-- Section: Chart -->
```

```
<section>
```

```
<!-- Grid row -->
```

```
<div class="row">
```

```
<!-- Grid column -->
```

```
<div class="col-xl-5 col-lg-12 mr-0 pb-2">
```

```
<!-- Card image -->
```

```
<div class="view view-cascade gradient-card-header light-blue lighten-1">
```

```
<h2 class="h2-responsive mb-0 font-weight-500">Purchased</h2>
```

```
</div>
```

```
<!-- Card content -->
```

```
<div class="card-body card-body-cascade pb-0">
```

```
<!-- Panel data -->
```

```
<div class="row py-3 pl-4">
```

```
<!-- First column -->
```

```
<div class="col-md-6">
```

```
<!-- Date select -->
```

```
<p class="lead"><span class="badge info-color p-2">Data range</span></p>
```

```
<select class="mdb-select colorful-select dropdown-info md-form">
  <option value="" disabled selected>Choose time period</option>
  <option value="1">Today</option>
  <option value="2">Yesterday</option>
  <option value="3">Last 7 days</option>
  <option value="3">Last 30 days</option>
  <option value="3">Last week</option>
  <option value="3">Last month</option>
</select>
```

```
<!-- Date pickers -->
```

```
<p class="lead pt-3 pb-4"><span class="badge info-color p-2">Custom
date</span></p>
```

```
<div class="md-form">
```

```
<input placeholder="Selected date" type="text" id="from" class="form-control
datepicker">
```

```
<label for="date-picker-example">From</label>
```

```
</div>
```

```
<div class="md-form">
```

```
<input placeholder="Selected date" type="text" id="to" class="form-control
datepicker">
```

```
<label for="date-picker-example">To</label>
```

```
</div>
```

```
</div>
```

```
<!-- First column -->
```

```
<!-- Second column -->
```

```
<div class="col-md-6 text-center pl-xl-2 my-md-0 my-3">
```

```
<!-- Summary -->
```

```
<p>Total sales: <strong>2000$</strong>
```

```
<button type="button" class="btn btn-info btn-sm p-2" data-toggle="tooltip"
data-placement="top"
```

```
        title="Total sales in the given period"><i class="fas fa-  
question"></i></button>
```

```
    </p>
```

```
    <p>Average sales: <strong>100$</strong>
```

```
        <button type="button" class="btn btn-info btn-sm p-2 mr-0" data-  
toggle="tooltip" data-placement="top"
```

```
        title="Average daily sales in the given period"><i class="fas fa-  
question"></i></button>
```

```
    </p>
```

```
    <!-- Change chart -->
```

```
        <span class="min-chart my-4" id="chart-sales" data-percent="76"><span  
class="percent"></span></span>
```

```
    <h5>
```

```
        <span class="badge red accent-2 p-2">Change <i class="fas fa-arrow-circle-  
up ml-1"></i></span>
```

```
        <button type="button" class="btn btn-info btn-sm p-2" data-toggle="tooltip"  
data-placement="top"
```

```
        title="Percentage change compared to the same period in the past"><i  
class="fas fa-question"></i>
```

```
    </button>
```

```
    </h5>
```

```
</div>
```

```
<!-- Second column -->
```

```
</div>
```

```
<!-- Panel data -->
```

```
</div>
```

```
<!-- Card content -->
```

```
</div>
```

```
<!-- Grid column -->
```

```
<!-- Grid column -->
<div class="col-xl-7 col-lg-12 mb-4 pb-2">

  <!-- Chart -->
  <div class="view view-cascade gradient-card-header blue-gradient">

    <canvas id="lineChart" height="175"></canvas>

  </div>

</div>

<!-- Grid column -->

</div>

<!-- Grid row -->

</section>
<!-- Section: Chart -->

<!-- Section: Table -->
<section>

  <!-- Top Table UI -->
  <div class="table-ui p-2 mb-3 mx-4 mb-5">

    <!-- Grid row -->
    <div class="row">

      <!-- Grid column -->
      <div class="col-xl-3 col-md-6">

        <!-- Name -->
```

```
<select class="mdb-select colorful-select dropdown-info mx-2 md-form mt-3 md-  
dropdown">
```

```
<option value="" disabled selected>Bulk actions</option>
```

```
<option value="1">Delete</option>
```

```
<option value="2">Export</option>
```

```
<option value="3">Change segment</option>
```

```
</select>
```

```
</div>
```

```
<!-- Grid column -->
```

```
<!-- Grid column -->
```

```
<div class="col-xl-3 col-md-6">
```

```
<!-- Blue select -->
```

```
<select class="mdb-select colorful-select dropdown-info mx-2 md-form mt-3 md-  
dropdown">
```

```
<option value="" disabled selected>Show only</option>
```

```
<option value="1">All <span> (2000)</span></option>
```

```
<option value="2">Never opened <span> (200)</span></option>
```

```
<option value="3">Opened but unanswered <span> (1800)</span></option>
```

```
<option value="4">Answered <span> (200)</span></option>
```

```
<option value="5">Unsubscribed <span> (50)</span></option>
```

```
</select>
```

```
</div>
```

```
<!-- Grid column -->
```

```
<!-- Grid column -->
```

```
<div class="col-xl-3 col-md-6">
```

```
<!-- Blue select -->
```

```
<select class="mdb-select colorful-select dropdown-info mx-2 md-form mt-3 md-  
dropdown">
```

```
<option value="" disabled selected>Filter segments</option>
```

```
<option value="1">Contacts in no segments <span> (100)</span></option>
```

```
<option value="1">Segment 1 <span> (2000)</span></option>
```

```
<option value="2">Segment 2 <span> (1000)</span></option>
```

```
<option value="3">Segment 3 <span> (4000)</span></option>
```

```
</select>
```

```
</div>
```

```
<!-- Grid column -->
```

```
<!-- Grid column -->
```

```
<div class="col-xl-3 col-md-6">
```

```
<form class="form-inline mt-2 ml-2">
```

```
<div class="form-group md-form mt-2">
```

```
<input class="form-control w-100" type="text" placeholder="Search">
```

```
</div>
```

```
<button class="btn btn-sm btn-primary ml-2 mr-0 mb-md-0 mb-4 px-2"><i  
class="fas fa-search"></i></button>
```

```
</form>
```

```
</div>
```

```
<!-- Grid column -->
```

```
</div>
```

```
<!-- Grid row -->
```

```
</div>
```

```
<!-- Top Table UI -->
```

```
</section>
```

<!--Section: Table-->

</div>

<!-- Card -->

</section>

<!-- Section: Main panel -->

</div>

<!-- Main layout -->

<!-- Footer -->

<asp:ContentPlaceHolder ID="CPH_TopMessages" runat="server">

</asp:ContentPlaceHolder>

<asp:ContentPlaceHolder ID="CPH_MainMaster_Header" runat="server">

</asp:ContentPlaceHolder>

<!-- Main layout -->

<!-- Main layout -->

<!-- Footer -->

<asp:ContentPlaceHolder ID="ContentMain" runat="server">

</asp:ContentPlaceHolder>

<asp:ContentPlaceHolder ID="ContentPlaceHolderfooter" runat="server">

<footer class="page-footer pt-0 mt-5 rgba-stylish-light">

<!-- Copyright -->

```
<div class="footer-copyright py-3 text-center">
  <div class="container-fluid">
    © 2019 Copyright: <a href="#" target="_blank"> Date Or Wed </a>
  </div>
</div>
```

```
</footer>
<!-- Footer -->
</asp:ContentPlaceHolder>
```

```
<!-- SCRIPTS -->
<!-- JQuery -->
<script src="Vendor/js/jquery-3.3.1.min.js"></script>
<!-- Bootstrap tooltips -->
<script type="text/javascript" src="Vendor/js/popper.min.js"></script>
<!-- Bootstrap core JavaScript -->
<script type="text/javascript" src="Vendor/js/bootstrap.js"></script>
<!-- MDB core JavaScript -->
<script type="text/javascript" src="Vendor/js/mdb.min.js"></script>
```

```
<!-- Initializations -->
<script>

  // SideNav Initialization
  $(".button-collapse").sideNav();

  var container = document.querySelector('.custom-scrollbar');
  Ps.initialize(container, {
    wheelSpeed: 2,
    wheelPropagation: true,
    minScrollbarLength: 20
  });
```



```
// Data Picker Initialization
$('.datepicker').pickadate();

// Material Select Initialization
$(document).ready(function () {
    $('.mdb-select').material_select();
});

// Tooltips Initialization
$(function () {
    $('[data-toggle="tooltip"]').tooltip()
})

</script>

<!-- Charts -->
<script>

// Small chart
$(function () {
    $('#min-chart#chart-sales').easyPieChart({
        barColor: "#FF5252",
        onStep: function (from, to, percent) {
            $(this.el).find('.percent').text(Math.round(percent));
        }
    });
});

// Main chart
var ctxL = document.getElementById("lineChart").getContext('2d');
var myLineChart = new Chart(ctxL, {
```

```
type: 'line',
data: {
  labels: ["January", "February", "March", "April", "May", "June", "July"],
  datasets: [{
    label: "My First dataset",
    fillColor: "#fff",
    backgroundColor: 'rgba(255, 255, 255, .3)',
    borderColor: 'rgba(255, 255, 255)',
    data: [0, 10, 5, 2, 20, 30, 45],
  }]
},
options: {
  legend: {
    labels: {
      fontColor: "#fff",
    }
  },
  scales: {
    xAxes: [{
      gridLines: {
        display: true,
        color: "rgba(255,255,255,.25)"
      },
      ticks: {
        fontColor: "#fff",
      },
    }],
    yAxes: [{
      display: true,
      gridLines: {
        display: true,
        color: "rgba(255,255,255,.25)"
      }
    }]
```

```
    },
    ticks: {
        fontColor: "#fff",
    },
}],
}
}
});
```

```
$(function () {
    $('#dark-mode').on('click', function (e) {

        e.preventDefault();

        $('footer, .card').toggleClass('dark-card-admin');
        $('body, .navbar').toggleClass('white-skin navy-blue-skin');
        $(this).toggleClass('white text-dark btn-outline-black');
        $('body').toggleClass('dark-bg-admin');
        $('h6, .card, p, td, th, i, li a, h4, input, label').not('#slide-out i, #slide-out a,
.dropdown-item i, .dropdown-item').toggleClass('text-white');
        $('.btn-dash').toggleClass('grey blue').toggleClass('lighten-3 darken-3');
        $('.gradient-card-header').toggleClass('white black lighten-4');
        $('.list-panel a').toggleClass('navy-blue-bg-a text-white').toggleClass('list-group-
border');

    });
});
```

```
</script>
```

```
</body>
```

```
</html>
```

```
*****VendorMasterPage.master.cs *****
```

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBEntity;
using DowBAL;
using System.Text;
using DowDAL;
using System.Data;
using System.Web.UI.WebControls;
using System.Web.Security;

public partial class VendorMasterPage : System.Web.UI.MasterPage
{
    string UserName;
    protected void Page_Load(object sender, EventArgs e)
    {
        Session["Pid"] = "1";
        Session["Pgid"] = "0";
        StringBuilder sb = new StringBuilder();
        //menu.InnerHtml = Common.Generate_newmenu();
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        string user;
        if (Clsusr != null)
        {
            UserName = Clsusr.UserName;
        }
        else
        {

```

```

    HttpCookie myCookie = Request.Cookies["DateOrWedCookie"];
    if (myCookie != null)
    {
        user = myCookie.Value.ToString();
    }
    else
    {
        user = "Guest";
    }
    guest_login(user);

}

}

protected void guest_login(string user)
{

    string City = "Hyderabad";// _City;
    string Country = "India";// _Country;

    BALCreateUser objlogin = new BALCreateUser();
    DataTable status_login = objlogin.LoginGuest(user, user,
Request.ServerVariables["REMOTE_ADDR"].ToString(), "", City, Country);
    string login_status = status_login.Rows[0]["UserStatus"].ToString();

    ClsLoggedInUser objLoggedInUser;
    if (login_status == "True")
    {
        objLoggedInUser = new ClsLoggedInUser();

        objLoggedInUser.UserId = status_login.Rows[0]["UserId"].ToString();

```

```
objLoggedInUser.UserName = status_login.Rows[0]["UserName"].ToString();
objLoggedInUser.GroupId = status_login.Rows[0]["GroupId"].ToString();
objLoggedInUser.LoginStatus = "True";
objLoggedInUser.LoginID = Convert.ToInt32(status_login.Rows[0]["LoginId"]);
objLoggedInUser.DealerId = Convert.ToInt32(status_login.Rows[0]["DealerId"]);
objLoggedInUser.RoleId = Convert.ToInt32(status_login.Rows[0]["RoleId"]);
objLoggedInUser.BrandId = 1; //Convert.ToInt32(status_login.Rows[0]["BrandId"]);
```

```
Session["LoggedUser_Specs"] = objLoggedInUser;
```

```
FormsAuthentication.SetAuthCookie(objLoggedInUser.UserName, false);
```

```
HttpCookie myCookie = new HttpCookie("DateOrWedCookie");
```

```
DateTime now = DateTime.Now;
```

```
myCookie.Value = objLoggedInUser.UserName;
```

```
myCookie.Expires = now.AddYears(1);
```

```
Response.Cookies.Add(myCookie);
```

```
}
```

```
}
```

```
}
```

```
*****AdminMasterPage.master*****
```

```
<%@ Master Language="C#" AutoEventWireup="true"  
CodeFile="AdminMasterPage.master.cs" Inherits="AdminMasterPage" %>
```

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
<meta charset="utf-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
```

```
<meta http-equiv="x-ua-compatible" content="ie=edge">
```

```
<title>MLB | Vendor Management</title>
<!-- Favicon -->
<link rel="icon" type="image/gif" href="img/favicon.png">
<!-- Font Awesome -->
<link rel="stylesheet" href="Vendor/css/all.css">
<!-- Bootstrap core CSS -->
<link rel="stylesheet" href="Vendor/css/bootstrap.min.css">
<!-- Material Design Bootstrap -->
<link rel="stylesheet" href="Vendor/css/mdb.min.css">
<!-- Your custom styles (optional) -->
<link rel="stylesheet" type="text/css" href="Vendor/css/style.css">
<style>

</style>
</head>

<body class="fixed-sn white-skin">

<!-- Main Navigation -->
<header>

<!-- Sidebar navigation -->
<div id="slide-out" class="side-nav sn-bg-4 fixed">
  <ul class="custom-scrollbar">

    <!-- Logo -->
    <li class="logo-sn waves-effect py-3">
      <div class="text-center">
        <a href="#" class="pl-0"></a>
      </div>
    </li>
```

```
<!-- Search Form -->
```

```
<li>
```

```
<form class="search-form" role="search">
```

```
<div class="md-form mt-0 waves-light">
```

```
<input type="text" class="form-control py-2" placeholder="Search">
```

```
</div>
```

```
</form>
```

```
</li>
```

```
<!-- Side navigation links -->
```

```
<li>
```

```
<ul class="collapsible collapsible-accordion">
```

```
<li class="active">
```

```
<a href="AdminHome.aspx" class="collapsible-header waves-effect"><i class="w-fa  
fas fa-tachometer-alt"></i>Dashboard</a>
```

```
</li>
```

```
<li>
```

```
<a class="collapsible-header waves-effect arrow-r">
```

```
<i class="w-fa fas fa-shopping-cart"></i>Shopping Vendor<i class="fas fa-angle-  
down rotate-icon"></i>
```

```
</a>
```

```
<div class="collapsible-body">
```

```
<ul>
```

```
<li>
```

```
<a href="vendorproductsAddUser.aspx" class="waves-effect">Select  
Categories </a>
```

```
<li>
```

```
<a href="vendorcolorsAddUser.aspx" class="waves-effect">Select Colors </a>
```

```
</li>
```

```
<li>
```

```
<a href="vendorsizesAddUser.aspx" class="waves-effect">Select Sizes </a>
```

```
</li>
```

```
<li>
```



```
        <a href="vendorproductsUser.aspx" class="waves-effect">Products
Management</a>
    </li>
    <li>
        <a href="PostProductsUser.aspx" class="waves-effect">Post Products </a>
    </li>
    <li>
        <a href="ManageOrderUser.aspx" class="waves-effect">Order
Management</a>
    </li>
    <li>
        <a href="ManagePaymentsUser.aspx" class="waves-effect">Payments</a>
    </li>
    <li>
        <a href="ManageLogisticsUser.aspx" class="waves-effect">Logistic</a>
    </li>
    <li>
        <a href="ManageOrderRetUser.aspx" class="waves-effect">Order returned</a>
    </li>
    <li>
        <a href="ManageCommunicationsUser.aspx" class="waves-
effect">Communication</a>
    </li>
</ul>
</div>
</li>
<li>
    <a class="collapsible-header waves-effect arrow-r">
        <i class="w-fa fas fa-user-cog"></i>Service Vendor<i class="fas fa-angle-down
rotate-icon"></i>
    </a>
    <div class="collapsible-body">
        <ul>
            <li>
```

Types

Management

Add Services

Management

Payments

Cancellations

effect">Communication

</div>

<i class="w-fa fas fa-tasks"></i>Business Listing Vendor<i class="fas fa-angle-down rotate-icon"></i>

<div class="collapsible-body">

```
<ul>
  <li>
    <a href="#" class="waves-effect">Add the Details Business</a>
  </li>
  <li>
    <a href="#" class="waves-effect">Product Catalog</a>
  </li>
  <li>
    <a href="#" class="waves-effect">Contact Detail</a>
  </li>
</ul>
</div>
</li>
</ul>
</li>
<!-- Side navigation links -->

</ul>
<div class="sidenav-bg mask-strong"></div>
</div>
<!-- Sidebar navigation -->

<!-- Navbar -->
<nav class="navbar fixed-top navbar-expand-lg scrolling-navbar double-nav">

  <!-- SideNav slide-out button -->
  <div class="float-left">
    <a href="#" data-activates="slide-out" class="button-collapse"><i class="fas fa-bars"></i></a>
  </div>

  <!-- Breadcrumb -->
  <div class="breadcrumb-dn mr-auto">
```

```
<p>Dashboard</p>
```

```
</div>
```

```
<div class="d-flex change-mode">
```

```
<div class="ml-auto mb-0 mr-3 change-mode-wrapper">
```

```
<button class="btn btn-outline-black btn-sm" id="dark-mode">Change Mode</button>
```

```
</div>
```

```
<!-- Navbar links -->
```

```
<ul class="nav navbar-nav nav-flex-icons ml-auto">
```

```
<!-- Dropdown -->
```

```
<li class="nav-item dropdown notifications-nav">
```

```
<a class="nav-link dropdown-toggle waves-effect" id="navbarDropdownMenuLink" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
```

```
<span class="badge red">3</span> <i class="fas fa-bell"></i>
```

```
<span class="d-none d-md-inline-block">Notifications</span>
```

```
</a>
```

```
<div class="dropdown-menu dropdown-primary" aria-labelledby="navbarDropdownMenuLink">
```

```
<a class="dropdown-item" href="#">
```

```
<i class="far fa-money-bill-alt mr-2" aria-hidden="true"></i>
```

```
<span>New order received</span>
```

```
<span class="float-right"><i class="far fa-clock" aria-hidden="true"></i> 13 min</span>
```

```
</a>
```

```
<a class="dropdown-item" href="#">
```

```
<i class="far fa-money-bill-alt mr-2" aria-hidden="true"></i>
```

```
<span>New order received</span>
```

```
<span class="float-right"><i class="far fa-clock" aria-hidden="true"></i> 33 min</span>
```

```
</a>
```

```
<a class="dropdown-item" href="#">
```

```
<i class="fas fa-chart-line mr-2" aria-hidden="true"></i>
<span>Your campaign is about to end</span>
<span class="float-right"><i class="far fa-clock" aria-hidden="true"></i> 53
min</span>
</a>
</div>
</li>
<li class="nav-item">
  <a class="nav-link waves-effect"><i class="fas fa-envelope"></i> <span
class="clearfix d-none d-sm-inline-block">Contact</span></a>
</li>
<li class="nav-item">
  <a class="nav-link waves-effect"><i class="far fa-comments"></i> <span
class="clearfix d-none d-sm-inline-block">Support</span></a>
</li>
<li class="nav-item dropdown">
  <a class="nav-link dropdown-toggle waves-effect" href="#" id="userDropdown" data-
toggle="dropdown" aria-haspopup="true" aria-expanded="false">
    <i class="fas fa-user"></i> <span class="clearfix d-none d-sm-inline-
block">Profile</span>
  </a>
  <div class="dropdown-menu dropdown-menu-right" aria-
labelledby="userDropdown">
    <a class="dropdown-item" href="#">Log Out</a>
    <a class="dropdown-item" href="#">My account</a>
  </div>
</li>
</ul>
<!-- Navbar links -->
</div>
</nav>
```

```
<!-- Navbar -->
```

```
<!-- Fixed button -->
```

```
<div class="fixed-action-btn clearfix d-none d-xl-block" style="bottom: 45px; right: 24px;">
```

```
<a class="btn-floating btn-lg red">
```

```
<i class="fas fa-pencil-alt"></i>
```

```
</a>
```

```
<ul class="list-unstyled">
```

```
<li><a class="btn-floating red"><i class="fas fa-star"></i></a></li>
```

```
<li><a class="btn-floating yellow darken-1"><i class="fas fa-user"></i></a></li>
```

```
<li><a class="btn-floating green"><i class="fas fa-envelope"></i></a></li>
```

```
<li><a class="btn-floating blue"><i class="fas fa-shopping-cart"></i></a></li>
```

```
</ul>
```

```
</div>
```

```
<!-- Fixed button -->
```

```
</header>
```

```
<!-- Main Navigation -->
```

```
<!-- Main layout -->
```

```
<div class="container-fluid">
```

```
<!-- Section: Intro -->
```

```
<section class="mt-md-4 pt-md-2 mb-5 pb-4">
```

```
<!-- Grid row -->
```

```
<div class="row">
```

```
<!-- Grid column -->
```

```
<div class="col-xl-3 col-md-6 mb-xl-0 mb-4">
```

```
<!-- Card -->
```

```
<div class="card card-cascade cascading-admin-card">
```

```
<!-- Card Data -->
```

```
<div class="admin-up">
```

```
<i class="far fa-money-bill-alt primary-color mr-3 z-depth-2"></i>
```

```
<div class="data">
```

```
<p class="text-uppercase">sales</p>
```

```
<h4 class="font-weight-bold dark-grey-text">2000$</h4>
```

```
</div>
```

```
</div>
```

```
<!-- Card content -->
```

```
<div class="card-body card-body-cascade">
```

```
<div class="progress mb-3">
```

```
<div class="progress-bar bg-primary" role="progressbar" style="width: 25%" aria-  
valuenow="25" aria-valuemin="0" aria-valuemax="100"></div>
```

```
</div>
```

```
<p class="card-text">Better than last week (25%)</p>
```

```
</div>
```

```
</div>
```

```
<!-- Card -->
```

```
</div>
```

```
<!-- Grid column -->
```

```
<!-- Grid column -->
```

```
<div class="col-xl-3 col-md-6 mb-xl-0 mb-4">
```

```
<!-- Card -->
```

```
<div class="card card-cascade cascading-admin-card">
```

```
<!-- Card Data -->
```

```
<div class="admin-up">
```

```
<i class="fas fa-chart-line warning-color mr-3 z-depth-2"></i>
```

```
<div class="data">
```

```
<p class="text-uppercase">subscriptions</p>
```

```
<h4 class="font-weight-bold dark-grey-text">200</h4>
```

```
</div>
```

```
</div>
```

```
<!-- Card content -->
```

```
<div class="card-body card-body-cascade">
```

```
<div class="progress mb-3">
```

```
<div class="progress-bar red accent-2" role="progressbar" style="width: 25%"  
aria-valuenow="25" aria-valuemin="0" aria-valuemax="100"></div>
```

```
</div>
```

```
<p class="card-text">Worse than last week (25%)</p>
```

```
</div>
```

```
</div>
```

```
<!-- Card -->
```

```
</div>
```

```
<!-- Grid column -->
```

```
<!-- Grid column -->
```

```
<div class="col-xl-3 col-md-6 mb-md-0 mb-4">
```

```
<!-- Card -->
```

```
<div class="card card-cascade cascading-admin-card">
```



```
<!-- Card Data -->
```

```
<div class="admin-up">
```

```
  <i class="fas fa-chart-pie light-blue lighten-1 mr-3 z-depth-2"></i>
```

```
  <div class="data">
```

```
    <p class="text-uppercase">traffic</p>
```

```
    <h4 class="font-weight-bold dark-grey-text">20000</h4>
```

```
  </div>
```

```
</div>
```

```
<!-- Card content -->
```

```
<div class="card-body card-body-cascade">
```

```
  <div class="progress mb-3">
```

```
    <div class="progress-bar red accent-2" role="progressbar" style="width: 75%"  
aria-valuenow="75" aria-valuemin="0" aria-valuemax="100"></div>
```

```
  </div>
```

```
  <p class="card-text">Worse than last week (75%)</p>
```

```
</div>
```

```
</div>
```

```
<!-- Card -->
```

```
</div>
```

```
<!-- Grid column -->
```

```
<!-- Grid column -->
```

```
<div class="col-xl-3 col-md-6 mb-0">
```

```
<!-- Card -->
```

```
<div class="card card-cascade cascading-admin-card">
```

```
<!-- Card Data -->
```

```
<div class="admin-up">
```

```
<i class="fas fa-chart-bar red accent-2 mr-3 z-depth-2"></i>
<div class="data">
  <p class="text-uppercase">organic traffic</p>
  <h4 class="font-weight-bold dark-grey-text">2000</h4>
</div>
</div>

<!-- Card content -->
<div class="card-body card-body-cascade">
  <div class="progress mb-3">
    <div class="progress-bar bg-primary" role="progressbar" style="width: 25%" aria-
valuenow="25" aria-valuemin="0" aria-valuemax="100"></div>
  </div>
  <p class="card-text">Better than last week (25%)</p>
</div>

</div>
<!-- Card -->

</div>
<!-- Grid column -->

</div>
<!-- Grid row -->

</section>
<!-- Section: Intro -->

<!-- Section: Main panel -->
<section class="mb-5">

  <!-- Card -->
  <div class="card card-cascade narrower">
```

```
<!-- Section: Chart -->
```

```
<section>
```

```
<!-- Grid row -->
```

```
<div class="row">
```

```
<!-- Grid column -->
```

```
<div class="col-xl-5 col-lg-12 mr-0 pb-2">
```

```
<!-- Card image -->
```

```
<div class="view view-cascade gradient-card-header light-blue lighten-1">
```

```
<h2 class="h2-responsive mb-0 font-weight-500">Purchased</h2>
```

```
</div>
```

```
<!-- Card content -->
```

```
<div class="card-body card-body-cascade pb-0">
```

```
<!-- Panel data -->
```

```
<div class="row py-3 pl-4">
```

```
<!-- First column -->
```

```
<div class="col-md-6">
```

```
<!-- Date select -->
```

```
<p class="lead"><span class="badge info-color p-2">Data range</span></p>
```

```
<select class="mdb-select colorful-select dropdown-info md-form">
```

```
<option value="" disabled selected>Choose time period</option>
```

```
<option value="1">Today</option>
```

```
<option value="2">Yesterday</option>
```

```
<option value="3">Last 7 days</option>
```

```
<option value="3">Last 30 days</option>
```

```
<option value="3">Last week</option>
<option value="3">Last month</option>
</select>
```

```
<!-- Date pickers -->
<p class="lead pt-3 pb-4"><span class="badge info-color p-2">Custom
date</span></p>
<div class="md-form">
  <input placeholder="Selected date" type="text" id="from" class="form-control
datepicker">
  <label for="date-picker-example">From</label>
</div>
<div class="md-form">
  <input placeholder="Selected date" type="text" id="to" class="form-control
datepicker">
  <label for="date-picker-example">To</label>
</div>
```

```
</div>
```

```
<!-- First column -->
```

```
<!-- Second column -->
```

```
<div class="col-md-6 text-center pl-xl-2 my-md-0 my-3">
```

```
<!-- Summary -->
```

```
<p>Total sales: <strong>2000$</strong>
<button type="button" class="btn btn-info btn-sm p-2" data-toggle="tooltip"
data-placement="top"
  title="Total sales in the given period"><i class="fas fa-
question"></i></button>
</p>
<p>Average sales: <strong>100$</strong>
<button type="button" class="btn btn-info btn-sm p-2 mr-0" data-
toggle="tooltip" data-placement="top"
```

```
        title="Average daily sales in the given period"><i class="fas fa-
question"></i></button>
    </p>

    <!-- Change chart -->
    <span class="min-chart my-4" id="chart-sales" data-percent="76"><span
class="percent"></span></span>
    <h5>
        <span class="badge red accent-2 p-2">Change <i class="fas fa-arrow-circle-
up ml-1"></i></span>
        <button type="button" class="btn btn-info btn-sm p-2" data-toggle="tooltip"
data-placement="top"
            title="Percentage change compared to the same period in the past"><i
class="fas fa-question"></i>
        </button>
    </h5>

</div>

<!-- Second column -->

</div>

<!-- Panel data -->

</div>

<!-- Card content -->

</div>

<!-- Grid column -->

<!-- Grid column -->
<div class="col-xl-7 col-lg-12 mb-4 pb-2">

    <!-- Chart -->
    <div class="view view-cascade gradient-card-header blue-gradient">
```

```
<canvas id="lineChart" height="175"></canvas>
```

```
</div>
```

```
</div>
```

```
<!-- Grid column -->
```

```
</div>
```

```
<!-- Grid row -->
```

```
</section>
```

```
<!-- Section: Chart -->
```

```
<!-- Section: Table -->
```

```
<section>
```

```
<!-- Top Table UI -->
```

```
<div class="table-ui p-2 mb-3 mx-4 mb-5">
```

```
<!-- Grid row -->
```

```
<div class="row">
```

```
<!-- Grid column -->
```

```
<div class="col-xl-3 col-md-6">
```

```
<!-- Name -->
```

```
<select class="mdb-select colorful-select dropdown-info mx-2 md-form mt-3 md-  
dropdown">
```

```
<option value="" disabled selected>Bulk actions</option>
```

```
<option value="1">Delete</option>
```

```
<option value="2">Export</option>
```

```
<option value="3">Change segment</option>
```

```
</select>
```

```
</div>
```

```
<!-- Grid column -->
```

```
<!-- Grid column -->
```

```
<div class="col-xl-3 col-md-6">
```

```
<!-- Blue select -->
```

```
<select class="mdb-select colorful-select dropdown-info mx-2 md-form mt-3 md-  
dropdown">
```

```
<option value="" disabled selected>Show only</option>
```

```
<option value="1">All <span> (2000)</span></option>
```

```
<option value="2">Never opened <span> (200)</span></option>
```

```
<option value="3">Opened but unanswered <span> (1800)</span></option>
```

```
<option value="4">Answered <span> (200)</span></option>
```

```
<option value="5">Unsubscribed <span> (50)</span></option>
```

```
</select>
```

```
</div>
```

```
<!-- Grid column -->
```

```
<!-- Grid column -->
```

```
<div class="col-xl-3 col-md-6">
```

```
<!-- Blue select -->
```

```
<select class="mdb-select colorful-select dropdown-info mx-2 md-form mt-3 md-  
dropdown">
```

```
<option value="" disabled selected>Filter segments</option>
```

```
<option value="1">Contacts in no segments <span> (100)</span></option>
```

```
<option value="1">Segment 1 <span> (2000)</span></option>
```

```
<option value="2">Segment 2 <span> (1000)</span></option>
```

```
<option value="3">Segment 3 <span> (4000)</span></option>
```

```
</select>
```

```
</div>
```

```
<!-- Grid column -->
```

```
<!-- Grid column -->
```

```
<div class="col-xl-3 col-md-6">
```

```
<form class="form-inline mt-2 ml-2">
```

```
<div class="form-group md-form mt-2">
```

```
<input class="form-control w-100" type="text" placeholder="Search">
```

```
</div>
```

```
<button class="btn btn-sm btn-primary ml-2 mr-0 mb-md-0 mb-4 px-2"><i  
class="fas fa-search"></i></button>
```

```
</form>
```

```
</div>
```

```
<!-- Grid column -->
```

```
</div>
```

```
<!-- Grid row -->
```

```
</div>
```

```
<!-- Top Table UI -->
```

```
</section>
```

```
<!--Section: Table-->
```

```
</div>
```

```
<!-- Card -->
```

```
</section>
```

```
<!-- Section: Main panel -->
```


</div>

<!-- Main layout -->

<!-- Footer -->

<asp:ContentPlaceHolder ID="CPH_TopMessages" runat="server">

</asp:ContentPlaceHolder>

<asp:ContentPlaceHolder ID="CPH_MainMaster_Header" runat="server">

</asp:ContentPlaceHolder>

<!-- Main layout -->

<!-- Main layout -->

<!-- Footer -->

<asp:ContentPlaceHolder ID="ContentMain" runat="server">

</asp:ContentPlaceHolder>

<asp:ContentPlaceHolder ID="ContentPlaceHolderfooter" runat="server">

<footer class="page-footer pt-0 mt-5 rgba-stylish-light">

<!-- Copyright -->

<div class="footer-copyright py-3 text-center">

<div class="container-fluid">

© 2022 Copyright: Shopping@dateorwed.com

</div>

</div>

</footer>

```
<!-- Footer -->
    </asp:ContentPlaceHolder>

<!-- SCRIPTS -->
<!-- JQuery -->
<script src="Vendor/js/jquery-3.3.1.min.js"></script>
<!-- Bootstrap tooltips -->
<script type="text/javascript" src="Vendor/js/popper.min.js"></script>
<!-- Bootstrap core JavaScript -->
<script type="text/javascript" src="Vendor/js/bootstrap.js"></script>
<!-- MDB core JavaScript -->
<script type="text/javascript" src="Vendor/js/mdb.min.js"></script>

<!-- Initializations -->
<script>

    // SideNav Initialization
    $(".button-collapse").sideNav();

    var container = document.querySelector('.custom-scrollbar');
    Ps.initialize(container, {
        wheelSpeed: 2,
        wheelPropagation: true,
        minScrollbarLength: 20
    });

    // Data Picker Initialization
    $(''.datepicker').pickadate();

    // Material Select Initialization
    $(document).ready(function () {
        $(''.mdb-select').material_select();
    });
</script>
```

```

});

// Tooltips Initialization
$(function () {
    $('[data-toggle="tooltip"]').tooltip()
})

</script>

<!-- Charts -->
<script>

// Small chart
$(function () {
    $('#min-chart#chart-sales').easyPieChart({
        barColor: "#FF5252",
        onStep: function (from, to, percent) {
            $(this.el).find('.percent').text(Math.round(percent));
        }
    });
});

// Main chart
var ctxL = document.getElementById("lineChart").getContext('2d');
var myLineChart = new Chart(ctxL, {
    type: 'line',
    data: {
        labels: ["January", "February", "March", "April", "May", "June", "July"],
        datasets: [{
            label: "My First dataset",
            fillColor: "#fff",
            backgroundColor: 'rgba(255, 255, 255, .3)',

```

```
        borderColor: 'rgba(255, 255, 255)',
        data: [0, 10, 5, 2, 20, 30, 45],
    ]
},
options: {
    legend: {
        labels: {
            fontColor: "#fff",
        }
    },
    scales: {
        xAxes: [{
            gridLines: {
                display: true,
                color: "rgba(255,255,255,.25)"
            },
            ticks: {
                fontColor: "#fff",
            },
        }],
        yAxes: [{
            display: true,
            gridLines: {
                display: true,
                color: "rgba(255,255,255,.25)"
            },
            ticks: {
                fontColor: "#fff",
            },
        }],
    }
}
```

```
});

$(function () {
    $('#dark-mode').on('click', function (e) {

        e.preventDefault();

        $('footer, .card').toggleClass('dark-card-admin');
        $('body, .navbar').toggleClass('white-skin navy-blue-skin');
        $(this).toggleClass('white text-dark btn-outline-black');
        $('body').toggleClass('dark-bg-admin');
        $('h6, .card, p, td, th, i, li a, h4, input, label').not('#slide-out i, #slide-out a,
.dropdown-item i, .dropdown-item').toggleClass('text-white');
        $('.btn-dash').toggleClass('grey blue').toggleClass('lighten-3 darken-3');
        $('.gradient-card-header').toggleClass('white black lighten-4');
        $('.list-panel a').toggleClass('navy-blue-bg-a text-white').toggleClass('list-group-
border');

    });
});
```

```
</script>
```

```
</body>
```

```
</html>
```

```
***** AdminMasterPage.master.cs *****
```

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBEntity;
```

```

using DowBAL;
using System.Text;
using DowDAL;
using System.Data;
using System.Web.UI.WebControls;
using System.Web.Security;

public partial class AdminMasterPage : System.Web.UI.MasterPage
{
    string UserName;
    protected void Page_Load(object sender, EventArgs e)
    {
        Session["Pid"] = "1";
        Session["Pgid"] = "0";
        StringBuilder sb = new StringBuilder();
        //menu.InnerHtml = Common.Generate_newmenu();
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        string user;
        if (Clsusr != null)
        {
            UserName = Clsusr.UserName;
        }
        else
        {
            HttpCookie myCookie = Request.Cookies["DateOrWedCookie"];
            if (myCookie != null)
            {
                user = myCookie.Value.ToString();
            }
            else
            {

```

```

        user = "Guest";
    }
    guest_login(user);

}

}

protected void guest_login(string user)
{

    string City = "Hyderabad";// _City;
    string Country = "India";// _Country;

    BALCreateUser objlogin = new BALCreateUser();
    DataTable status_login = objlogin.LoginGuest(user, user,
Request.ServerVariables["REMOTE_ADDR"].ToString(), "", City, Country);
    string login_status = status_login.Rows[0]["UserStatus"].ToString();

    ClsLoggedInUser objLoggedInUser;
    if (login_status == "True")
    {
        objLoggedInUser = new ClsLoggedInUser();

        objLoggedInUser.UserId = status_login.Rows[0]["UserId"].ToString();
        objLoggedInUser.UserName = status_login.Rows[0]["UserName"].ToString();
        objLoggedInUser.GroupId = status_login.Rows[0]["GroupId"].ToString();
        objLoggedInUser.LoginStatus = "True";
        objLoggedInUser.LoginID = Convert.ToInt32(status_login.Rows[0]["LoginId"]);
        objLoggedInUser.DealerId = Convert.ToInt32(status_login.Rows[0]["DealerId"]);
        objLoggedInUser.RoleId = Convert.ToInt32(status_login.Rows[0]["RoleId"]);
        objLoggedInUser.BrandId = 1; //Convert.ToInt32(status_login.Rows[0]["BrandId"]);
    }
}

```

```
Session["LoggedUser_Specs"] = objLoggedInUser;
```

```
FormsAuthentication.SetAuthCookie(objLoggedInUser.UserName, false);
```

```
HttpCookie myCookie = new HttpCookie("DateOrWedCookie");
```

```
DateTime now = DateTime.Now;
```

```
myCookie.Value = objLoggedInUser.UserName;
```

```
myCookie.Expires = now.AddYears(1);
```

```
Response.Cookies.Add(myCookie);
```

```
}
```

```
}
```

```
}
```

```
*****BListingMasterPage.master *****
```

```
<%@ Master Language="C#" AutoEventWireup="true"  
CodeFile="BListingMasterPage.master.cs" Inherits="BListingMasterPage" %>
```

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
<meta charset="utf-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
```

```
<meta http-equiv="x-ua-compatible" content="ie=edge">
```

```
<title>Date Or Wed Business Listing</title>
```

```
<link rel="icon" href="img/favicon.png" type="image/gif">
```

```
<!-- Font Awesome -->
```

```
<link rel="stylesheet" href="font/fontawesome/css/all.css">
```

```
<!-- Bootstrap core CSS -->
```

```
<link href="css/bootstrap.min.css" rel="stylesheet">
```



```

<!-- Material Design Bootstrap -->
<link href="css/mdb.min.css" rel="stylesheet">
<!-- Your custom styles (optional) -->
<link href="css/style.css" rel="stylesheet">
</head>

<body>
  <form id="form1" runat="server" >
<%--      <asp:ScriptManager ID="ScriptManager1" runat="server">
          </asp:ScriptManager>      --%>
  <!-- Start your project here-->

  <!--Navbar -->
<nav class="mb-1 navbar fixed-top navbar-expand-lg navbar-light bg-light">
  <a class="navbar-brand" href="Home.aspx"></a>

  <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent-4"
  aria-controls="navbarSupportedContent-4" aria-expanded="false" aria-label="Toggle navigation">
    <span class="navbar-toggler-icon"></span>
</button>
<div class="collapse navbar-collapse" id="navbarSupportedContent-4">
  <ul class="navbar-nav ml-auto">
    <li class="nav-item">
      <a class="nav-link text-center" href="#">
        <i class="fas fa-store-alt"></i><br> Sell</a>
    </li>
    <li class="nav-item">
      <a class="nav-link text-center" href="#">
        <i class="far fa-question-circle"></i><br> Help</a>
    </li>
    <li class="nav-item">
      <a class="nav-link text-center" href="#">

```

```
        <i class="fas fa-envelope-open-text"></i><br> Messages</a>
    </li>
    <li class="nav-item dropdown">
        <a class="nav-link dropdown-toggle text-center" id="navbarDropdownMenuLink-4"
        data-toggle="dropdown"
            aria-haspopup="true" aria-expanded="false">
            <i class="far fa-user-circle"></i><br> Sign In</a>
        <div class="dropdown-menu dropdown-menu-right dropdown-info dropdown-primary"
        aria-labelledby="navbarDropdownMenuLink-4">
            <a class="dropdown-item" href="#">My account</a>
            <a class="dropdown-item" href="#">Log out</a>
        </div>
    </li>
</ul>
</div>
</nav>
<!--.Navbar -->
```

```
<!--Carousel Wrapper-->
```

```
<div id="carousel-example-2" class="carousel slide carousel-fade" data-ride="carousel">
```

```
<!--Slides-->
```

```
<div class="carousel-inner" role="listbox">
```

```
<div class="carousel-item active">
```

```
<div class="view">
```

```

```

```
<div class="mask rgba-black-slight"></div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<!--/.Carousel Wrapper-->
```

```
<!-- Search -->
```

```
<section class="search-sec">
```

```
  <div class="container">
```

```
    <div class="row">
```

```
      <div class="col-lg-12">
```

```
        <div class="row">
```

```
          <div class="col-lg-2 col-md-2 col-sm-6 p-0">
```

```
            <asp:DropDownList ID="ddl_location" runat="server" Height="50px">
```

```
              <asp:ListItem Value="0">--- Select a Location ---</asp:ListItem>
```

```
              <asp:ListItem Value="1">Ameerpet</asp:ListItem>
```

```
              <asp:ListItem Value="2">Secunderabad</asp:ListItem>
```

```
              <asp:ListItem Value="3">Hyderabad</asp:ListItem>
```

```
              <asp:ListItem Value="4">Charminar</asp:ListItem>
```

```
            </asp:DropDownList>
```

```
          </div>
```

```
            <div class="col-lg-5 col-md-5 col-sm-16 p-0">
```

```
              <asp:TextBox ID="txt_SearchString" runat="server" class="form-control search-slt" Height="45px" onkeydown="javascript:TextName_OnKeyDown(event)"></asp:TextBox>
```

```
            </div>
```

```
          <div class="col-lg-2 col-md-2 col-sm-6 p-0">
```

```
            <button id="Button1" type="button" class="btn btn-secondary wrn-btn mt-0 ml-0" onclick="JavaScript:GetURLSring1()" ><i class='fa fa-search'></i>&nbsp;&nbsp;&nbsp;Search</button>
```

```
          </div>
```

```
<script type="text/javascript">
```

```
  function GetURLSring1() {
```

```
    //      alert('GetURLSring-1');
```

```
    var fullURL = (location.pathname + location.search).substr(1);
```

```

var loc=document.getElementById("<%=ddl_location.ClientID%>").value;
var EnCodedFullURL = encodeURIComponent(fullURL);
var SearchString =
document.getElementById("<%=txt_SearchString.ClientID%>").value;
var url = 'ViewSearchResult.aspx?pageid=198' + '&SrchStrng=' +
encodeURIComponent(SearchString) + '&loc=' + encodeURIComponent(loc);
//$(location).attr('href', url);
window.location.href = url;
event.preventDefault();
}
</script>

```

```

<!--
        <div class="col-lg-3 col-md-3 col-sm-12 p-0">
            <select class="form-control search-slt" id="exampleFormControlSelect1">
                <option>-Select Location-</option>
                <option>Hyderabad</option>
                <option>Delhi</option>
                <option>Banglore</option>
            </select>
        </div>
        <div class="col-lg-6 col-md-6 col-sm-12 p-0">
            <input type="text" class="form-control search-slt" placeholder="Enter
Product/Service name">
        </div>
        <div class="col-lg-3 col-md-3 col-sm-12 p-0">
            <button type="button" class="btn btn-secondary wrn-btn mt-0 ml-
0">Search</button>
        </div>
--%>
        </div>
    </div>
</div>
</section>

```

```

<!-- ./Search -->
    <asp:ContentPlaceHolder ID="CPH_TopMessages" runat="server">
        <div class="container-fluid">
            <div class="row-fluid">
                <%-- <div class="row-fluid" style="padding-top:
20px;">--%>
                    <div class="span2" id="dvlogo" runat="server" style="display: none;">
                        <%--
                            
                        --%>
                    </div>
                    <div class="span10">
                        <div class="row">
                            <div id="dvMsgBox" class="span12 ordersave" runat="server"
style="display: block;">
                                <asp:Label ID="Label1" runat="server" Text=""></asp:Label>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </asp:ContentPlaceHolder>
    <asp:ContentPlaceHolder ID="CPH_MainMaster_Header" runat="server">
        </asp:ContentPlaceHolder>

<section class="buynsell pt-5 pb-3">
    <div class="container-fluid">
        <div class="row">
            <div class="col-md-3" id="sidebar">
                <div class="m-list"><h5><i class="fas fa-list"></i> BUSINESS LISTING</h5></div>
                <div class="list-group panel">
                    <a href="#" class="list-group-item collapsed" data-parent="#sidebar"> <span
class="hidden-sm-down"><i class="fas fa-circle"></i> Shopping Vendors</span></a>

```

 <i class="fas fa-circle"></i> Fashion <i class="fas fa-chevron-right" style="font-size: 12px;"></i>

<div class="collapse" id="menu3">

<i class="fas fa-circle"></i> Bride

<i class="fas fa-circle"></i> BrideGroom

<i class="fas fa-circle"></i> Men

<i class="fas fa-circle"></i> Women

<i class="fas fa-circle"></i> Kids

</div>

 <i class="fas fa-circle"></i> TV & Appliances

 <i class="fas fa-circle"></i> Electronics

 <i class="fas fa-circle"></i> Furniture & Furnishing

 <i class="fas fa-circle"></i> Home & Living

 <i class="fas fa-circle"></i> Metal Appliances

 <i class="fas fa-circle"></i> Wedding Material

</div>

<!-- <div class="m-list">

<h5><i class="fas fa-list"></i> MY MARKETS</h5>

Cakes & Bakery Items

Wedding item Rentals

Wedding Planner

Wedding Invitation Cards

```

    <li><a href="#">Travel Agents (Hotel, Travel)</a></li>
    <li><a href="#">Makeup </a></li>
    <li><a href="#">Recital Material Packing</a></li>
    <li><a href="#">Honey Moon Packages</a></li>
    <li class="text-primary"><a class="text-primary" href="#">All Categories</a></li>
  </ul>
</div> -->
</div>
<div class="col-md-9">
  <!--Carousel Wrapper-->
<div id="carousel-example-1z" class="carousel slide carousel-fade" data-ride="carousel">
  <!--Indicators-->
  <ol class="carousel-indicators">
    <li data-target="#carousel-example-1z" data-slide-to="0" class="active"></li>
    <li data-target="#carousel-example-1z" data-slide-to="1"></li>
  </ol>
  <!--/.Indicators-->
  <!--Slides-->
  <div class="carousel-inner" role="listbox">
    <!--First slide-->
    <div class="carousel-item active">
      
    </div>
    <!--/First slide-->
    <!--Second slide-->
    <div class="carousel-item">
      
    </div>
    <!--/Second slide-->
    <div class="carousel-item">

```

```

    
  </div>
<div class="carousel-item">
  
  </div>
</div>
<!--/.Slides-->
<!--Controls-->
<a class="carousel-control-prev" href="#carousel-example-1z" role="button" data-
slide="prev">
  <span class="carousel-control-prev-icon" aria-hidden="true"></span>
  <span class="sr-only">Previous</span>
</a>
<a class="carousel-control-next" href="#carousel-example-1z" role="button" data-
slide="next">
  <span class="carousel-control-next-icon" aria-hidden="true"></span>
  <span class="sr-only">Next</span>
</a>
<!--/.Controls-->
</div>
<!--/.Carousel Wrapper-->
  </div>
</div>
</div>
</section>

<section class="furec pb-3 pt-3">
  <div class="container-fluid">
    <div class="card">
      <h3 class="h3-responsive font-weight-bold pb-0 mb-0 pl-3">Featured
Recommendations</h3>

```



```
<div class="p-3">
<div class="row">
  <div class="col-md-2">
    <div class="card1 p-2 text-center">
      
      <div class="card-body">
        <h6 class="card-title font-weight-bold">Bride Fashion Store</h6>
        <div class="mb-4"></div>
        <a href="#">By: ABC</a>
        <div><a class="btn btn-secondary btn-sm btn-rounded">Get Best Price</a></div>
      </div>
    </div>
  </div>
</div>
```

```
<div class="col-md-2">
  <div class="card1 p-2 text-center">
    
    <div class="card-body">
      <h6 class="card-title font-weight-bold">Bride Fashion Store</h6>
      <div class="mb-4"></div>
      <a href="#">By: ABC</a>
      <div><a class="btn btn-secondary btn-sm btn-rounded">Get Best Price</a></div>
    </div>
  </div>
</div>
```

```
<div class="col-md-2">
  <div class="card1 p-2 text-center">
    
    <div class="card-body">
      <h6 class="card-title font-weight-bold">Bride Fashion Store</h6>
      <div class="mb-4"></div>
```

```
<a href="#">By: ABC</a>
<div><a class="btn btn-secondary btn-sm btn-rounded">Get Best Price</a></div>
</div>
</div>
</div>
```

```
<div class="col-md-2">
  <div class="card1 p-2 text-center">
    
    <div class="card-body">
      <h6 class="card-title font-weight-bold">Bride Fashion Store</h6>
      <div class="mb-4"></div>
      <a href="#">By: ABC</a>
      <div><a class="btn btn-secondary btn-sm btn-rounded">Get Best Price</a></div>
    </div>
  </div>
</div>
```

```
<div class="col-md-2">
  <div class="card1 p-2 text-center">
    
    <div class="card-body">
      <h6 class="card-title font-weight-bold">Bride Fashion Store</h6>
      <div class="mb-4"></div>
      <a href="#">By: ABC</a>
      <div><a class="btn btn-secondary btn-sm btn-rounded">Get Best Price</a></div>
    </div>
  </div>
</div>
```

```
<div class="col-md-2">
  <div class="card1 p-2 text-center">
```


</div>

</div>

</div>

<div class="col-md-9">

<div class="row pb-4">

<div class="col-md-4">

<div class="card1 p-2">

<div class="row">

<div class="col-md-4">

</div>

<div class="col-md-8">

<p class="card-title font-weight-bold">Bride Fashion Store</p>

<p>Bridal Dresses</p>

<p>Best Quality Dresses</p>

<p>Best Quality Bridal Dresses</p>

</div>

</div>

</div>

</div>

<div class="col-md-4">

<div class="card1 p-2">

<div class="row">

<div class="col-md-4">

</div>

<div class="col-md-8">

<p class="card-title font-weight-bold">Women Fashion Store</p>

<p>Best Bridal Dresses</p>

<p>Best Quality Bridal Dresses</p>

```
<p><a href="#">Best Quality Bridal</a></p>
</div>
</div>
</div>
</div>

<div class="col-md-4">
<div class="card1 p-2">
<div class="row">
<div class="col-md-4">

</div>
<div class="col-md-8">
<p class="card-title font-weight-bold">Groom Fashion Store</p>
<p><a href="#">Best Quality Bridal</a></p>
<p><a href="#">Best Quality Bridal Dresses</a></p>
<p><a href="#">Best QualityDresses</a></p>
</div>
</div>
</div>
</div>
</div>

<div class="row pb-4">
<div class="col-md-4">
<div class="card1 p-2">
<div class="row">
<div class="col-md-4">

</div>
<div class="col-md-8">
<p class="card-title font-weight-bold">Women Fashion Store</p>
```

<p>Best Quality Dresses</p>

<p>Best Quality Bridal Dresses</p>

<p>Best Bridal Dresses</p>

</div>

</div>

</div>

</div>

<div class="col-md-4">

<div class="card1 p-2">

<div class="row">

<div class="col-md-4">

</div>

<div class="col-md-8">

<p class="card-title font-weight-bold">Men Fashion Store</p>

<p>Best Quality Bridal Dresses</p>

<p>Quality Bridal Dresses</p>

<p>Best Quality Bridal</p>

</div>

</div>

</div>

</div>

<div class="col-md-4">

<div class="card1 p-2">

<div class="row">

<div class="col-md-4">

</div>

<div class="col-md-8">

<p class="card-title font-weight-bold">Kids Fashion Store</p>

```
<p><a href="#">Best Bridal Dresses</a></p>
<p><a href="#">Best Quality Bridal Dresses</a></p>
<p><a href="#">Best Quality Bridal</a></p>
</div>
</div>
</div>
</div>
</div>

<div class="row">
<div class="col-md-4">
<div class="card1 p-2">
<div class="row">
<div class="col-md-4">

</div>
<div class="col-md-8">
<p class="card-title font-weight-bold">Kids Fashion Store</p>
<p><a href="#">Best Bridal Dresses</a></p>
<p><a href="#">Best Quality Dresses</a></p>
<p><a href="#">Best Quality Bridal Dresses</a></p>
</div>
</div>
</div>
</div>
</div>

<div class="col-md-4">
<div class="card1 p-2">
<div class="row">
<div class="col-md-4">

</div>
```



```
</section>
<asp:ContentPlaceHolder ID="ContentPlaceHolder1" runat="server">
  </asp:ContentPlaceHolder>

<!-- Footer -->
<footer class="page-footer font-small unique-color-dark">

<div style="background-color: #080A52;">
  <div class="container">

    <!-- Grid row-->
    <div class="row py-4 d-flex align-items-center">

      <!-- Grid column -->
      <div class="col-md-6 col-lg-5 text-center text-md-left mb-4 mb-md-0">
        <h6 class="mb-0">Get connected with us on social networks!</h6>
      </div>

      <!-- Grid column -->

      <!-- Grid column -->
      <div class="col-md-6 col-lg-7 text-center text-md-right">

        <!-- Facebook -->
        <a class="fb-ic">
          <i class="fab fa-facebook-f white-text mr-4"> </i>
        </a>

        <!-- Twitter -->
        <a class="tw-ic">
          <i class="fab fa-twitter white-text mr-4"> </i>
        </a>

        <!--Linkedin -->
        <a class="li-ic">
```

```
        <i class="fab fa-linkedin-in white-text mr-4"> </i>
    </a>
    <!--Instagram-->
    <a class="ins-ic">
        <i class="fab fa-instagram white-text"> </i>
    </a>

</div>
<!-- Grid column -->

</div>
<!-- Grid row-->

</div>
</div>

<!-- Footer Links -->
<div class="container text-center text-md-left mt-5">

    <!-- Grid row -->
    <div class="row mt-3">

        <!-- Grid column -->
        <div class="col-md-2 col-lg-2 col-xl-2 mx-auto mb-4">

        </div>
        <!-- Grid column -->

        <!-- Grid column -->
        <div class="col-md-3 col-lg-3 col-xl-3 mx-auto mb-4">
```

```
<!-- Links -->
<h6 class="text-uppercase font-weight-bold">Contact Us</h6>
<hr class="deep-purple accent-2 mb-4 mt-0 d-inline-block mx-auto" style="width:
60px;">
<p>
  <i class="fas fa-envelope mr-3"></i> admin@dateorwed.com</p>
<p>
  <i class="fas fa-phone mr-3"></i> +91 - 7893642582</p>

</div>
```

```
<!-- Grid column -->
```

```
<!-- Grid column -->
```

```
<div class="col-md-2 col-lg-2 col-xl-2 mx-auto mb-4">
```

```
<!-- Links -->
<h6 class="text-uppercase font-weight-bold">Users</h6>
<hr class="deep-purple accent-2 mb-4 mt-0 d-inline-block mx-auto" style="width:
40px;">
<p>
  <a href="#">My Profile</a>
</p>

</div>
```

```
<!-- Grid column -->
```

```
<!-- Grid column -->
```

```
<div class="col-md-2 col-lg-2 col-xl-2 mx-auto mb-md-0 mb-4">
```

```
<!-- Links -->
```

```
<h6 class="text-uppercase font-weight-bold">About Us</h6>
```

```
<hr class="deep-purple accent-2 mb-4 mt-0 d-inline-block mx-auto" style="width: 60px;">
```

```
<p>
```

```
<a href="#">About Us</a></p>
```

```
<p>
```

```
<a href="#">FAQs</a></p>
```

```
<p>
```

```
<a href="#">Feedback</a></p>
```

```
</div>
```

```
<!-- Grid column -->
```

```
<!-- Grid column -->
```

```
<div class="col-md-3 col-lg-3 col-xl-3 mx-auto mb-md-0 mb-4">
```

```
<!-- Links -->
```

```
<h6 class="text-uppercase font-weight-bold">SERVICE PROVIDERS</h6>
```

```
<hr class="deep-purple accent-2 mb-4 mt-0 d-inline-block mx-auto" style="width: 60px;">
```

```
<p>
```

```
<a href="#">Become a Service Provider</a></p>
```

```
<p>
```

```
<a href="#">Dashboard</a></p>
```

```
<p>
```

```
<a href="#">Emergency Contacts</a></p>
```

```
</div>
```

```
<!-- Grid column -->
```

```
</div>
```

```
<!-- Grid row -->
```

```
<div class="border-bottom border-light"></div>
```

```
<!-- Grid row -->
```

```
<div class="row mt-3">
```

```
<!-- Grid column -->
```

```
<div class="col-md-3 col-lg-4 col-xl-3 mx-auto mb-4">
```

```
<!-- Content -->
```

```
<h6 class="text-uppercase font-weight-bold">Top Business Listings</h6>
```

```
<hr class="deep-purple accent-2 mb-4 mt-0 d-inline-block mx-auto" style="width: 100px;">
```

```
<p>
```

```
<a href="#">Shopping Vendors</a></p>
```

```
<p>
```

```
<a href="#">Bride </a></p>
```

```
<p>
```

```
<a href="#">Men </a></p>
```

```
</div>
```

```
<!-- Grid column -->
```

```
<!-- Grid column -->
```

```
<div class="col-md-2 col-lg-2 col-xl-2 mx-auto mb-4">
```

```
<!-- Links -->
```

```
<h6 class="text-uppercase font-weight-bold">&nbsp;</h6>
```

```
<hr class="deep-purple accent-2 mb-4 mt-0 d-inline-block mx-auto border-0" style="width: 60px;">
```

```
<p>
```

```
<a href="#">Women</a>
```

```
</p>
```

```
<p>
```

```
<a href="#">Kids</a>
```

```
</p>
```

```
<p>
```

```
<a href="#">TV & Appliances</a>
```

</p>

</div>

<!-- Grid column -->

<!-- Grid column -->

<div class="col-md-3 col-lg-2 col-xl-2 mx-auto mb-4">

<!-- Links -->

<h6 class="text-uppercase font-weight-bold"> </h6>

<hr class="deep-purple accent-2 mb-4 mt-0 d-inline-block mx-auto border-0" style="width: 60px;">

<p>

Electronics

</p>

<p>

Furniture & Furnishing

</p>

<p>

Home & Living

</p>

</div>

<!-- Grid column -->

<!-- Grid column -->

<div class="col-md-4 col-lg-3 col-xl-3 mx-auto mb-md-0 mb-4">

<!-- Links -->

<h6 class="text-uppercase font-weight-bold"> </h6>

<hr class="deep-purple accent-2 mb-4 mt-0 d-inline-block mx-auto border-0" style="width: 60px;">

<p>

[Metal Appliances](#)

[Wedding Material](#)

[Jewelry](#)

© 2019 Copyright

[Date Or Wed](#) All Rights Reserved

[#topSection](#) *btn-floating btn-large red*

fa fa-arrow-up

```
<asp:ContentPlaceHolder ID="ContentPlaceHolder2" runat="server">
</asp:ContentPlaceHolder>
```

```
<!-- Start your project here-->
```

```
<!-- SCRIPTS -->
```

```
<!-- JQuery -->
```

```
<script type="text/javascript" src="js/jquery-3.4.1.min.js"></script>
```

```
<!-- Bootstrap tooltips -->
```

```
<script type="text/javascript" src="js/popper.min.js"></script>
```

```
<!-- Bootstrap core JavaScript -->
```

```
<script type="text/javascript" src="js/bootstrap.min.js"></script>
```

```
<!-- MDB core JavaScript -->
```

```
<script type="text/javascript" src="js/mdb.min.js"></script>
```

```
<script type="text/javascript">
```

```
$(function() {
```

```
    var $btn = $('#btnTop');
```

```
    var $home = $('#topSection');
```

```
    var startpoint = $home.scrollTop() + $home.height();
```

```
    $(window).on('scroll', function() {
```

```
        if($(window).scrollTop() > startpoint) {
```

```
            $btn.show();
```

```
        } else {
```

```
            $btn.hide();
```

```
        }
```

```
    });
```

```
});
```

```
</script>
```

```
</form>
```



```
</body>
```

```
</html>
```

```
*****BListingMasterPage.master.cs *****
```

```
using System;
```

```
using System.Collections.Generic;
```

```
using System.Linq;
```

```
using System.Web;
```

```
using System.Web.UI;
```

```
using System.Web.UI.WebControls;
```

```
public partial class BListingMasterPage : System.Web.UI.MasterPage
```

```
{
```

```
    protected void Page_Load(object sender, EventArgs e)
```

```
    {
```

```
        string logpath = Server.MapPath("~/log/" + "Log" +  
Convert.ToString(DateTime.Today.Day) + ".txt");
```

```
        Session["Logpath"] = logpath;
```

```
    }
```

```
}
```

```
***** MainMaster.master *****
```

```
<%@ Master Language="C#" AutoEventWireup="true" CodeFile="MainMaster.master.cs"  
Inherits="MainMaster" %>
```

```
<%--<%@ Register Assembly="AjaxControlToolkit" Namespace="AjaxControlToolkit"  
TagPrefix="cc1" %>--%>
```

```
<!DOCTYPE html>
```

```
<html xmlns="http://www.w3.org/1999/xhtml">
```

```
<head>
```

```
    <title>Your shopping Destination</title>
```

```
<meta charset="utf-8"/>
<meta name="viewport" content="width=device-width, initial-scale=1.0" />
<meta name="description" content="The description of my page" />
<link href="OldCss/css/font-awesome.css" type="text/css" rel="stylesheet" />
<link href="OldCss/css/font-awesome.min.css" type="text/css" rel="stylesheet" />
<link href="OldCss/css/menu.css" type="text/css" rel="menu1" />
<link href="OldCss/css/menu.css" type="text/css" rel="stylesheet" />
<link href="OldCss/css/tabstyle.css" type="text/css" rel="stylesheet" />
<link rel="stylesheet" type="text/css" href="OldCss/css/bootstrap.css" />
<link rel="stylesheet" type="text/css" href="OldCss/css/bootstrap.min.css" />
<link rel="stylesheet" type="text/css" href="OldCss/css/style.css" />
<link rel="stylesheet" href="OldCss/css/base.css"/>
<link rel="stylesheet" href="OldCss/css/migrate.css"/>
<link rel="stylesheet" href="OldCss/css/pdp.css" />
<script src="OldCss/js/jquery1.min.js"></script>
<script src="OldCss/js/bootstrap1.min.js"></script>
<script src="OldCss/js/common.js"></script>
<script src="OldCss/js/jquery.countdown.js"></script>
<script type="text/javascript" src="OldCss/js/jquery-1.6.1.min.js"></script>
<script type="text/javascript" src="OldCss/js/ddsmoothmenu.js"></script>
<script type="text/javascript" src="OldCss/js/bootstrap.js"></script>
<script type="text/javascript" src="OldCss/js/bootstrap.min.js"></script>
<script type="text/javascript" src="OldCss/js/jquery.js"></script>
<script type="text/javascript" src="OldCss/js/bootstrap-collapse.js"></script>
<script src="OldCss/js/swiper.js"></script>
<script src="OldCss/js/common.js"></script>
<script src="OldCss/js/jquery.countdown.js"></script>
<%--<script src="js/NewOrder.js"></script>--%>
<script src="OldCss/js/bootstrap.min.js"></script>
<script src="OldCss/js/swiper.min.js"></script>
<script src="OldCss/js/swiper.esm.js"></script>
<script src="OldCss/js/swiper.esm.bundle.js"></script>
```

```

<script type="text/javascript" src="OldCss/js/skinable_tabs.min.js"></script>
<script type="text/javascript" src="OldCss/js/jquery-ui.min.js"></script>
<script type="text/javascript">
    $(' .tabs_holder').skinableTabs({
        effect: 'basic_display',
        skin: 'skin7',
        position: 'top',
        persist_tab: false
    });
</script>
<script type="text/javascript">
    ddsmoothmenu.init({
        mainmenuid: "mainMenu", //menu DIV id
        orientation: 'h', //Horizontal or vertical menu: Set to "h" or "v"
        classname: 'ddsmoothmenu', //class added to menu's outer DIV
        //customtheme: ["#1c5a80", "#18374a"],
        contentsource: "markup" //"markup" or ["container_id", "path_to_menu_file"]
    }) </script>
<script type="text/javascript" src="js/ManagePermissions.js"></script>
<!-- <script type="text/javascript" src="js/NewOrder.js"></script>-->
<script type="text/javascript">
    function showTable(tbl) {
        //alert('table' + tbl);

        if (document.getElementById(tbl).style.display == "none") {
            document.getElementById(tbl).style.display = "table";
        }
        else {
            document.getElementById(tbl).style.display = "none";
        }
    }
}

```

```

function ShowMenu(Div) {

    if (document.getElementById) {

        if (document.getElementById(Div).style.display == "none") {
            document.getElementById(Div).style.display = "table";
        }
        else
        {
            document.getElementById(Div).style.display = "none";
        }

        //var oelement = document.getElementsByClassName("ActiveLi");
        //oelement.style.visibility = 'hide';
        //oelement.style.display = 'none';
        //oelement.classList.remove("ActiveLi");
        //if (document.getElementById(Div).style.display == "none") {
        //    var element = document.getElementById(Div);
        //    element.style.visibility = 'visible';
        //    element.style.display = 'block';
        //    element.classList.add("ActiveLi");
        //}
    }

    document.getElementById(mainMenu).style.display = 'block';
}

function showDiv(Div) {

    //alert('show');

    if (document.getElementById) {

        //LookupMenu('1');

        document.getElementById(Div).style.visibility = 'visible';
        document.getElementById(Div).style.display = 'block';
    }
}

```

```

    }
}
function HideDiv(Div) {
    alert('hide');

    if (document.getElementById) {
        //LookupMenu('1');
        document.getElementById(Div).style.visibility = 'hide';
        document.getElementById(Div).style.display = 'none';
    }
}
</script>
<script type="text/javascript">
    function ShowMenu(Div) {

        if (document.getElementById) {
            if (document.getElementById(Div).style.display == "none") {
                document.getElementById(Div).style.display = "table";
            }
            else
            {
                document.getElementById(Div).style.display = "none";
            }
        }

        //document.getElementById(mainMenu).style.display = 'block';
        //var oelement = document.getElementsByClassName("ActiveLi");
        //oelement.style.visibility = 'hide';
        //oelement.style.display = 'none';
        //oelement.classList.remove("ActiveLi");
        //if (document.getElementById(Div).style.display == "none") {

```

```
// var element = document.getElementById(Div);
// element.style.visibility = 'visible';
// element.style.display = 'block';
// element.classList.add("ActiveLi");
//}
}
```

</script>

<script type="text/javascript">

function hideAll() {

if (document.getElementById) {

 //var element = document.getElementsByTagName("ActiveLi");

 //element.style.visibility = 'hide';

 //element.style.display = 'none';

 //element.classList.remove("ActiveLi");

 //document.getElementById('tab1').style.display = "none";

 //document.getElementById('tab2').style.display = "none";

 //document.getElementById('tab3').style.display = "none";

 //document.getElementById('tab4').style.display = "none";

 //document.getElementById('tab5').style.display = "none";

 //document.getElementById('tab6').style.display = "none";

 //document.getElementById('tab7').style.display = "none";

 //document.getElementById('tab8').style.display = "none";

 //document.getElementById('tab9').style.display = "none";

 //document.getElementById('tab10').style.display = "none";

 //document.getElementById('tab11').style.display = "none";

 //document.getElementById('tab11').style.display = "none";

 //document.getElementById('tab12').style.display = "none";

 //document.getElementById('tab13').style.display = "none";

 //document.getElementById('tab14').style.display = "none";

 //document.getElementById('tab15').style.display = "none";

```
        //document.getElementById('tab16').style.display = "none";
    }
} </script>
<script type="text/javascript" src="js/skinable_tabs.min.js"></script>
<script type="text/javascript" src="js/jquery-ui.min.js"></script>
```

```
<style type="text/css">
```

```
.line {
    /*line-height: 40px;*/
    background: #09F;
    /* word-spacing: 55px; */
    width: 100%;
    margin: 0 auto;
    color: #F5F20C;
    font-family: Arial, Helvetica, sans-serif;
    font-size: 9px;
    font-weight: bold;
}
```

```
.line a {
    /*padding: 10px 0px 10px 0px;*/
    color: #FFF;
    text-decoration: underline;
    font-weight: normal;
}
```

```
.line a:hover {
    text-decoration: none;
}
```

```
.text {
    padding: 3px 0px;
```

```
}  
</style>
```

```
<style type="text/css">
```

```
.line {  
    /*line-height: 40px;*/  
    background: #09F;  
    /* word-spacing: 55px; */  
    width: 100%;  
    margin: 0 auto;  
    color: #F5F20C;  
    font-family: Arial, Helvetica, sans-serif;  
    font-size: 9px;  
    font-weight: bold;  
}
```

```
.line a {  
    /*padding: 10px 0px 10px 0px;*/  
    color: #FFF;  
    text-decoration: underline;  
    font-weight: normal;  
}
```

```
.line a:hover {  
    text-decoration: none;  
}
```

```
.text {  
    padding: 3px 0px;  
}
```

```
</style>
```

```
<asp:ContentPlaceHolder ID="CPH_MainMaster_Header" runat="server">
```



```
</asp:ContentPlaceholder>
```

```
</head>
```

```
<body>
```

```
<form id="form1" runat="server" >
```

```
<div>
```

```
<asp:ScriptManager ID="ScriptManager1" runat="server">
```

```
</asp:ScriptManager>
```

```
<!-- Top login detail bar Div/Section Start -->
```

```
<div>
```

```
<%-- added for test--%>
```

```
<%-- adds for test --%>
```

```
<div id="menu" class="menu app-menu" style="display:block"
onmouseout="javascript:hideAll()" >
```

```
<div class="menu-wrapper">
```

```
</div>
```

```
<%-- these are for extra menus to code--%>
```

```
<%-- <div class="menu-wrapper-align" >--%>
```

```
<div class="pure-g-r">
```

```
<div id="navmain" class="app-common-header-container span14" role="navigation"
runat="server"> <%-- onmouseout="javascript:hideAll"--%>
```

```
<%-- new menu--%>
```

```
<ul class='nav-main'>
```

```
<li class='nav-main-item'>
```

```
<a class='nav-main-link app-header-tab' onmouseover="javascript:ShowMenu('tab1')" href='OrderSearchList.aspx?id=1'>Clothing</a>
```

```
</li>
```

```
<li class='nav-main-item'>
```

```
<a class='nav-main-link app-header-tab' onmouseover="javascript:ShowMenu('tab2')" href='OrderSearchList.aspx?id=2'>Jewellery</a>
```

```
</li>
```

```
<li class='nav-main-item'>
```

```
<a class='nav-main-link app-header-tab' onmouseover="javascript:ShowMenu('tab3')" href='OrderSearchList.aspx?id=3'>Footwear</a>
```

```
</li>
```

```
<li class='nav-main-item'>
```

```
<a class='nav-main-link app-header-tab' onmouseover="javascript:ShowMenu('tab4')" href='OrderSearchList.aspx?id=4'>Makeup</a>
```

```
</li>
```

```
<li class='nav-main-item'>
```

```
<a class='nav-main-link app-header-tab' onmouseover="javascript:ShowMenu('tab5')" href='OrderSearchList.aspx?id=5'>Mehendi</a>
```

```
</li>
```

```
<li class='nav-main-item'>
```

```
<a class='nav-main-link app-header-tab' onmouseover="javascript:ShowMenu('tab6')" href='OrderSearchList.aspx?id=6'>Venues</a>
```

```
</li>
```

```
<li class='nav-main-item'>
```

```
<a class='nav-main-link app-header-tab' onmouseover="javascript:ShowMenu('tab8')" href='OrderSearchList.aspx?id=8'>Video</a>
```

```
</li>
```

```
<li class='nav-main-item'>
```

```
<a class='nav-main-link app-header-tab' onmouseover="javascript:ShowMenu('tab9')"
href='OrderSearchList.aspx?id=9'>Decorations</a>
</li>
```

```
<li class='nav-main-item'>
<a class='nav-main-link app-header-tab' onmouseover="javascript:ShowMenu('tab10')"
href='OrderSearchList.aspx?id=10'>Planners</a>
</li>
```

```
<li class='nav-main-item'>
<a class='nav-main-link app-header-tab' onmouseover="javascript:ShowMenu('tab11')"
href='OrderSearchList.aspx?id=11'>Invitations</a>
</li>
```

```
<li class='nav-main-item'>
<a class='nav-main-link app-header-tab' onmouseover="javascript:ShowMenu('tab12')"
href='OrderSearchList.aspx?id=12'>Catering</a>
</li>
```

```
<li class='nav-main-item'>
<a class='nav-main-link app-header-tab' onmouseover="javascript:ShowMenu('tab13')"
href='OrderSearchList.aspx?id=13'>Sangeeth</a>
</li>
```

```
<li class='nav-main-item'>
<a class='nav-main-link app-header-tab' onmouseover="javascript:ShowMenu('tab14')"
href='OrderSearchList.aspx?id=14'>Accessories</a>
</li>
```

```
<li class='nav-main-item'>
<a class='nav-main-link app-header-tab' onmouseover="javascript:ShowMenu('tab15')"
href='OrderSearchList.aspx?id=15'>Gifts</a>
</li>
```

```
<li class='nav-main-item'>
```

```
<a class='nav-main-link app-header-tab' onmouseover="javascript:ShowMenu('tab16')"
href='OrderSearchList.aspx?id=16'>Travel</a>
```

```
</li>
```

```
</ul>
```

```
</div>
```

```
<!-- $(function() {
    $(document).on('mouseenter.collapse', '[data-toggle=collapse]', function(e) {
        var $this = $(this),
            href, target = $this.attr('data-target') || e.preventDefault() || (href = $this.attr('href')) &&
            href.replace(/.*(?=#[\^s]+$)/, "");
        ,
            option = $(target).hasClass('in') ? 'hide' : "show";
            $('<!-- .tabsholder-collapse').not(target).collapse("hide");
            $(target).collapse(option);
        })
    });-->
```

```
<!--new menu ends-->
```

```
<!-- original mennu-->
```

```
<%-- original menu ends--%>
```

```
</div>
```

```
<%-- </div>
```

```
</div> --%>
```

```
<div class="tabsHeader app-common-header-dropdown">
```

```
<div class="vendorMenu shadow bg-white">
```

```
<div class="frow bg-pattern-light frow bg-white">
```

```
<div id="tabsholder" class="tabs_holder" runat="server" >
```

```
<%--tabsholder starts--%>
```

```
<%--col-xs-12 col-sm-6 col-md-6--%>
```

```
<div id='tab1' class='col-xs-4 col-sm-4 col-md-4 item-title ActiveLi' runat='server'  
style='display:block'>
```

```
<a href='GOrderSearchList.aspx?pid=1' class='col-xs-12 col-sm-6 col-md-6 item-title sub-  
heading' onmouseover="javascript:showTable('subtab1')">Bridal Wear</a>
```

```
<table id='subtab1' class='table' style='display:none'><tr><td><a  
href='OrderSearchList.aspx?pid=1' class='tabsHeaderList__item'>Bridal Dresses</a></td>
```

```
<td><a href='OrderSearchList.aspx?pid=2' class='tabsHeaderList__item'>Bridal  
Sarees</a></td>
```

```
<td><a href='OrderSearchList.aspx?pid=3' class='tabsHeaderList__item'>Gharga  
Choli</a></td>
```

```
<td><a href='OrderSearchList.aspx?pid=4' class='tabsHeaderList__item'>Western  
Dresses</a></td>
```

```
<td><a href='OrderSearchList.aspx?pid=5' class='tabsHeaderList__item'>Indian Traditional  
Wear</a></td>
```

```
<td><a href='OrderSearchList.aspx?pid=6' class='tabsHeaderList__item'>Imported  
Dresses</a></td>
```

<td>Indian Silk Sarees</td>

</tr></table>

Groom Wear

<table id='subtab2' class='table' style='display:none'><tr><td>Mens Dresses</td>

<td>Mens Shirts</td>

<td>Mens Trousers</td>

<td>Mens Coats</td>

<td>Mens Shervani</td>

</tr></table>

Mens Clothing

<table id='subtab36' class='table' style='display:none'><tr><td>Casual Wear</td>

<td>Formal Wear</td>

</tr></table>

Womens Clothing

<table id='subtab37' class='table' style='display:none'><tr><td>Womens Casual Wear</td>

<td>Womens Formal Wear</td>

</tr></table>

</div>

<div id='tab2' class='col-xs-4 col-sm-4 col-md-4 item-title ' runat='server' style='display:none'>

Chains

[Necklesses](GOrderSearchList.aspx?pid=4)

[Rings](GOrderSearchList.aspx?pid=5)

[Anklewear](GOrderSearchList.aspx?pid=6)

[Hipwear](GOrderSearchList.aspx?pid=7)

</div>

<div id='tab3' class='col-xs-4 col-sm-4 col-md-4 item-title ' runat='server' style='display:none'>

[Ladies Footwear](GOrderSearchList.aspx?pid=8)

[Gents Footwear](GOrderSearchList.aspx?pid=9)

</div>

<div id='tab4' class='col-xs-4 col-sm-4 col-md-4 item-title ' runat='server' style='display:none'>

[Groom Makeup](GOrderSearchList.aspx?pid=10)

[Bride Makeup](GOrderSearchList.aspx?pid=11)

</div>

<div id='tab5' class='col-xs-4 col-sm-4 col-md-4 item-title ' runat='server' style='display:none'>

</tr></table>

</div>

<div id='tab6' class='col-xs-4 col-sm-4 col-md-4 item-title ' runat='server' style='display:none'>

</tr></table>

</tr></table>


```
<table id='subtab17' class='table' style='display:none'><tr><td><a href='OrderSearchList.aspx?pid=27' class='tabsHeaderList__item'>High cost venues</a></td>
```

```
</tr></table>
```

```
<a href='GOrderSearchList.aspx?pid=18' class='col-xs-12 col-sm-6 col-md-6 item-title sub-heading' onmouseover="javascript:showTable('subtab18')">Low cost venues</a>
```

```
<table id='subtab18' class='table' style='display:none'><tr><td><a href='OrderSearchList.aspx?pid=28' class='tabsHeaderList__item'>Low cost venues</a></td>
```

```
</tr></table>
```

```
</div>
```

```
<div id='tab8' class='col-xs-4 col-sm-4 col-md-4 item-title ' runat='server' style='display:none'>
```

```
<a href='GOrderSearchList.aspx?pid=20' class='col-xs-12 col-sm-6 col-md-6 item-title sub-heading ' onmouseover="javascript:showTable('subtab20')" >Videographers</a>
```

```
<a href='GOrderSearchList.aspx?pid=21' class='col-xs-12 col-sm-6 col-md-6 item-title sub-heading' onmouseover="javascript:showTable('subtab21')">Photo & Videographers</a>
```

```
<table id='subtab21' class='table' style='display:none'><tr><td><a href='OrderSearchList.aspx?pid=31' class='tabsHeaderList__item'>Photo & Videographers</a></td>
```

```
</tr></table>
```

```
</div>
```

```
<div id='tab9' class='col-xs-4 col-sm-4 col-md-4 item-title ' runat='server' style='display:none'>
```

```
<a href='GOrderSearchList.aspx?pid=22' class='col-xs-12 col-sm-6 col-md-6 item-title sub-heading ' onmouseover="javascript:showTable('subtab22')" >Venue Decorations</a>
```

```
<a href='GOrderSearchList.aspx?pid=23' class='col-xs-12 col-sm-6 col-md-6 item-title sub-heading' onmouseover="javascript:showTable('subtab23')">Reception Decoration</a>
```

```
<table id='subtab23' class='table' style='display:none'><tr><td><a href='OrderSearchList.aspx?pid=33' class='tabsHeaderList__item'>Reception Decoration</a></td>
```

```
</tr></table>
```

```
</div>
```

```
<div id='tab10' class='col-xs-4 col-sm-4 col-md-4 item-title ' runat='server' style='display:none'>
```

Wedding Planners

Destination Wedding Planners

<table id='subtab25' class='table' style='display:none'><tr><td>Destination Wedding Planners</td>

</tr></table>

</div>

<div id='tab11' class='col-xs-4 col-sm-4 col-md-4 item-title ' runat='server' style='display:none'>

Invitations

Invitations with Gifts

<table id='subtab27' class='table' style='display:none'><tr><td>Invitations with Gifts</td>

</tr></table>

</div>

<div id='tab12' class='col-xs-4 col-sm-4 col-md-4 item-title ' runat='server' style='display:none'>

Catering Services

</div>

<div id='tab13' class='col-xs-4 col-sm-4 col-md-4 item-title ' runat='server' style='display:none'>

DJ Services

Sangeeth Services

<table id='subtab30' class='table' style='display:none'><tr><td>Sangeeth Services</td>

</tr></table>

Wedding Music

<table id='subtab31' class='table' style='display:none'><tr><td>Wedding Music</td>

</tr></table>

</div>

<div id='tab14' class='col-xs-4 col-sm-4 col-md-4 item-title ' runat='server' style='display:none'>

Accessories

</div>

<div id='tab15' class='col-xs-4 col-sm-4 col-md-4 item-title ' runat='server' style='display:none'>

Return Gifts

</div>

<div id='tab16' class='col-xs-4 col-sm-4 col-md-4 item-title ' runat='server' style='display:none'>

Local Travel Planning

International Travel Planning

<table id='subtab35' class='table' style='display:none'><tr><td>International Travel Planning</td>

</tr></table>

</div>

<%--tabsholder ends--%>

</div>

</div>

</div>

</div>

</div>

<!-- Top login detail bar Div/Section End -->

<%--<div class="menu app-menu" runat="server" style="display:block"> --%>

<%-- class='menudiv' style="display:none"--%>

<div id="mainMenu" class="ddsmoothmenu" runat="server" style="display:block">

<br style="clear: left"/>

<ul class='pull-right'>

Welcome admin

<li style='border-right: none;'><i class='fa fa-user'></i> Profile

<li style='border-right: none;' > <i class='fa fa-power-off'></i> Logout

</div>

<div id="dv_RecentPlaces" class="ddsmoothmenu" runat="server" style="display:none">

Link1 Link2 Link3 Text to be Displayed

<br style="clear: left"/>

```

        </div>
    <%--</div>--%>
<%-- --new menu system starts
    <!-- Top Main Menu Div/Section Start -->
    <!-- Top Main Menu Div/Section End -->

<%-- ----new menu ends--%>

    <!-- Top Main Menu Div/Section Start -->
    <!-- Top Main Menu Div/Section End -->
    <!-- Top Logo/Messages Div/Section Start -->
    <asp:ContentPlaceHolder ID="CPH_TopMessages" runat="server">
        <div class="container-fluid">

            <div class="row-fluid">    <%-- <div class="row-fluid" style="padding-top:
20px;">--%>

                <div class="span2" id="dvlogo" runat="server" style="display: none;">
    <%--
                    
--%>

                </div>

                <div class="span10">

                    <div class="row">

                        <div id="dvMsgBox" class="span12 ordersave" runat="server"
style="display: block;">

                            <asp:Label ID="Label1" runat="server" Text=""></asp:Label>

                        </div>

                    </div>

                </div>

            </div>

        </asp:ContentPlaceHolder>
    <!-- Top Logo/Messages Div/Section End -->

```

```
<div class="container-fluid">
  <div class="row-fluid">

    <!-- Left Bar Div/Section Start -->
    <asp:ContentPlaceHolder ID="CPH_LeftSideBar" runat="server">
      <div id="LeftSideBar" runat="server" class="span2">
        <asp:Table ID="Table1" runat="server">
          </asp:Table>

        </div>
      </asp:ContentPlaceHolder>
    <!-- Left Bar Div/Section End -->

    <!-- Main Content Div/Section Start -->
    <asp:ContentPlaceHolder ID="CPH_MainContent" runat="server">

      </asp:ContentPlaceHolder>
    <!-- Main Content Div/Section End -->
  </div>

</div>

</div>

</div>
  <div class="app-footer">

    <div id="footer" runat="server">

      <asp:ContentPlaceHolder ID="ContentPlaceHolder1" runat="server">

    </div>

  </div>

</div>
```

```
<div class="homeSection homeCatalog">
```

```
<p class="homeSection__extraTitle">Wedding Shopping gallery</p>
```

```
<p class="homeSection__title">Catalogue of bridal & Groom fashion</p>
```

```
<p class="homeCatalog__text">
```

```
Discover the latest trends in wedding dresses by top designers and bridesmaid dresses. Choose your favourite from our catalogue! </p>
```

```
<div class="homeCatalog__categories">
```

```
<div class="homeCatalog__categoriesItem">
```

```
<span class="app-link" data-href="/GOrderSearchList.aspx?pid=2">
```

```
<i class="icon-catalog-categories icon-catalog-categories-bride-dress block mb40"></i>
```

```
<a href="GOrderSearchList.aspx?pid=2">Wedding Dresses</a>
```

```
</span>
```

```
</div>
```

```
<div class="homeCatalog__categoriesItem">
```

```
<span class="app-link" data-href="/GOrderSearchList.aspx?pid=2">
```

```
<i class="icon-catalog-categories icon-catalog-categories-suit block mb40"></i>
```

```
<a href="GOrderSearchList.aspx?pid=2">Grooms suits</a>
```

```
</span>
```

```
</div>
```

```
<div class="homeCatalog__categoriesItem">
```

```
<span class="app-link" data-href="/GOrderSearchList.aspx?pid=37">
```

```
<i class="icon-catalog-categories icon-catalog-categories-godmum block mb40"></i>
```

```
<a href="OrderSearchList.aspx?pid=37">Bridesmaid Dresses</a>
```

```
</span>
```

```
</div>
```

```
<div class="homeCatalog__categoriesItem">
```

```
<span class="app-link" data-href="/GOrderSearchList.aspx?pid=54">
```

```
<i class="icon-catalog-categories icon-catalog-categories-dress block mb40"></i>
```

```
<a href="GOrderSearchList.aspx?pid=54">Party Dresses</a>
```

```
</span>
```

```
</div>
```

```
<div class="homeCatalog__categoriesItem">
  <span class="app-link" data-href="/OrderSearchList.aspx?id=2">
    <i class="icon-catalog-categories icon-catalog-categories-diamond block
mb40"></i>
    <a href="OrderSearchList.aspx?id=2">Jewellery</a>
  </span>
</div>

<div class="homeCatalog__categoriesItem">
  <span class="app-link" data-href="/OrderSearchList.aspx?id=3">
    <i class="icon-catalog-categories icon-catalog-categories-shoe block
mb40"></i>
    <a href="OrderSearchList.aspx?id=3">Shoes</a>
  </span>
</div>

<div class="homeCatalog__categoriesItem">
  <span class="app-link" data-href="/OrderSearchList.aspx?pid=16">
    <i class="icon-catalog-categories icon-catalog-categories-bra block mb40"></i>
    <a href="OrderSearchList.aspx?pid=16">Lingerie</a>
  </span>
</div>

<div class="homeCatalog__categoriesItem">
  <span class="app-link" data-href="/OrderSearchList.aspx?id=16">
    <i class="icon-catalog-categories icon-catalog-categories-handbag block
mb40"></i>
    <a href="OrderSearchList.aspx?id=16">Accessories</a>
  </span>
</div>
</div>

<span class="btn-outline outline-red mt40 app-link" data-
href="OrderSearchList.aspx?id=1">
  &nbsp;&nbsp;&nbsp;See catalogue&nbsp;&nbsp;&nbsp;
</span>
</div>

<div class="pb25">
```



```
<div class="home-category">
  <div class="wrapper">
    <p class="homeSection__title homeSection__title--md mt45">Wedding Services by
category</p>
    <div class="pure-g mt40">
      <div class="pure-u-1-3">
        <a class="homeCategories__titleCategory homeCategories__titleCategory--
catering" href="wedding-venues">
          Wedding Venues      </a>
        <ul class="pure-g homeCategories__list">
          <li class="pure-u-1-2 homeCategories__listItem">
            <a class="homeCategories__listAnchor" href="FunctionHalls">
              Function Halls      </a>
          </li>
          <li class="pure-u-1-2 homeCategories__listItem">
            <a class="homeCategories__listAnchor" href="hotels">
              Hotels      </a>
          </li>
          <li class="pure-u-1-2 homeCategories__listItem">
            <a class="homeCategories__listAnchor" href="banquet-halls">
              Banquet Halls      </a>
          </li>
          <li class="pure-u-1-2 homeCategories__listItem">
            <a class="homeCategories__listAnchor" href="party-lounge">
              Party Lounge      </a>
          </li>
          <li class="pure-u-1-2 homeCategories__listItem">
            <a class="homeCategories__listAnchor" href="farmhouse">
              Farm Houses      </a>
          </li>
          <li class="pure-u-1-2 homeCategories__listItem">
            <a class="homeCategories__listAnchor" href="temples">
```

```
        Temple Function Halls                </a>
    </li>
</ul>
</div>
<div class="pure-u-1-3">
    <a class="homeCategories__titleCategory homeCategories__titleCategory--
vendors" href="wedding-vendors">
        Wedding Suppliers                </a>
    <ul class="homeCategories__list pure-g">
        <li class="pure-u-1-2 homeCategories__listItem">
            <a class="homeCategories__listAnchor" href="catering">
                Catering                    </a>
            </li>
            <li class="pure-u-1-2 homeCategories__listItem">
                <a class="homeCategories__listAnchor" href="wedding-invitations">
                    Wedding Invitations      </a>
                </li>
                <li class="pure-u-1-2 homeCategories__listItem">
                    <a class="homeCategories__listAnchor" href="wedding-gifts">
                        Wedding Gifts         </a>
                    </li>
                    <li class="pure-u-1-2 homeCategories__listItem">
                        <a class="homeCategories__listAnchor" href="wedding-photography">
                            Wedding Photography </a>
                        </li>
                        <li class="pure-u-1-2 homeCategories__listItem">
                            <a class="homeCategories__listAnchor" href="wedding-music">
                                Wedding Music   </a>
                            </li>
                            <li class="pure-u-1-2 homeCategories__listItem">
                                <a class="homeCategories__listAnchor" href="wedding-transportation">
                                    Wedding Transportation </a>
                                </li>
```


<li class="pure-u-1-2 homeCategories__listItem">

Tent House

<li class="pure-u-1-2 homeCategories__listItem">

Wedding Entertainment

<li class="pure-u-1-2 homeCategories__listItem">

Florists

<li class="pure-u-1-2 homeCategories__listItem">

Wedding Planners

<li class="pure-u-1-2 homeCategories__listItem">

Wedding Videography

<li class="pure-u-1-2 homeCategories__listItem">

Honeymoon

<li class="pure-u-1-2 homeCategories__listItem">

Wedding Decoration

<li class="pure-u-1-2 homeCategories__listItem">

Wedding Cakes

<li class="pure-u-1-2 homeCategories__listItem">

Wedding DJ

<li class="pure-u-1-2 homeCategories__listItem">

Photobooth

<li class="pure-u-1-2 homeCategories__listItem">

Astrologers

<li class="pure-u-1-2 homeCategories__listItem">

Wedding Choreography

</div>

<div class="pure-u-1-3">

<div class="pure-g">

<div class="pure-u-1-2">

Brides

<ul class="homeCategories__list">

<li class="homeCategories__listItem">

Mehndi Design

<li class="homeCategories__listItem">


```
        Bridal Makeup                                </a>
    </li>
        <li class="homeCategories__listItem">
            <a class="homeCategories__listAnchor" href="makeup-salon">
                Makeup Salon                            </a>
        </li>
        <li class="homeCategories__listItem">
            <a class="homeCategories__listAnchor" href="bridal-jewellery">
                Bridal Jewellery                        </a>
        </li>
        <li class="homeCategories__listItem">
            <a class="homeCategories__listAnchor" href="bridal-lehenga">
                Bridal Lehenga                          </a>
        </li>
        <li class="homeCategories__listItem">
            <a class="homeCategories__listAnchor" href="trousseau-packing">
                Trousseau Packing                       </a>
        </li>
    </ul>
</div>
<div class="pure-u-1-2">
    <a class="homeCategories__titleCategory homeCategories__titleCategory--grooms" href="grooms">
        Grooms                                </a>
    <ul class="homeCategories__list">
        <li class="homeCategories__listItem">
            <a class="homeCategories__listAnchor" href="sherwani">
                Sherwani                            </a>
        </li>
        <li class="homeCategories__listItem">
            <a class="homeCategories__listAnchor" href="Coats">
                Coats                                </a>
```

```
                </li>
            </ul>
        </div>
    </div>
</div>
</div>
</div>
</div>
</div>
</div>
```

```
<div class="white pb25">
  <div class="wrapper">
    <p class="footer-chevron-title">Wedding Venues by location</p>
    <ul class="pure-g footer-list">
      <li class="pure-u-1-4">
        <a href="wedding-venues/delhi-ncr">
          Wedding Venues Delhi NCR      </a>
        </li>
      <li class="pure-u-1-4">
        <a href="wedding-venues/mumbai">
          Wedding Venues Mumbai      </a>
        </li>
      <li class="pure-u-1-4">
        <a href="wedding-venues/bangalore">
          Wedding Venues Bangalore      </a>
        </li>
      <li class="pure-u-1-4">
        <a href="wedding-venues/hyderabad">
          Wedding Venues Hyderabad      </a>
        </li>
      <li class="pure-u-1-4">
        <a href="wedding-venues/ap">
          Wedding Venues Andra Pradesh      </a>
```

```
</li>
  <li class="pure-u-1-4">
    <a href="wedding-venues/Telengana">
      Wedding Venues Telengana    </a>
    </li>
  </ul>
</div>
</div>
```

```
</div>
```

```
<div id="app-common-layer" class="modal fade dnone" tabindex="-1" role="dialog"
aria-hidden="true"></div>
```

```
<div id="app-va-modal" class="modal fade dnone" tabindex="-1" role="dialog" aria-
hidden="true"></div>
```

```
</div>
```

```
</div>
```

```
<%--          dateorwed.com Version 1.00.01. <a
href="http://dateorwed.com:699/Heartbeat">Heartbeat</a>
```

```
Copyright © 2018 Ravikirantechnosoft
```

```
--%>
```

```
</asp:ContentPlaceHolder>
```

```
</form>
```

```
</body>
```

```
</html>
```

```
*****MainMaster.master.cs *****
```

```
using System;  
using System.Collections.Generic;  
using System.Linq;  
using System.Web;  
using System.Web.UI;  
using System.Web.UI.WebControls;  
using DowBEntity;  
using DowBAL;  
using System.Text;  
using System.Data;  
using System.Web.UI.HtmlControls;  
using System.Text.RegularExpressions;  
using System.Web.Security;  
//using LogTracer;
```

```
public partial class MainMaster : System.Web.UI.MasterPage  
{
```

```
    public event EventHandler MasterControlClicked;  
    public event EventHandler MasterControlSaveClicked;
```



```

public event EventHandler MasterControlCancelClicked;
public event EventHandler MasterControlURLClicked;
public DDLogTracer objlog = new DDLogTracer();

protected void Page_Load(object sender, EventArgs e)
{
    bool IsMobile = false;
    if (Request.Browser.IsMobileDevice) { IsMobile = true; } else { IsMobile = false; }

    try
    {
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr != null)
        {
            if (!Page.IsPostBack)
            {
                int Groupid = Convert.ToInt16( Clsusr.GroupId);
                mainMenu.InnerHtml = string.Empty;

                dv_RecentPlaces.InnerHtml = string.Empty;
                dv_RecentPlaces.InnerHtml =
Common.BuildRecentPlaces(Convert.ToInt32(Clsusr.UserId));
                string pageid = Request.QueryString["pageid"];
                string fpath = Request.QueryString["fpath"];
                Session["PageType"] = "";
                navmain.InnerHtml = Common.Generate_topmenu();
                tabsholder.InnerHtml = Common.Generate_Menu();
                //TopBar.InnerHtml = Common.Generate_TopBar(Clsusr.UserName,
Clsusr.RoleId, Clsusr.DealerId, Convert.ToInt32(Clsusr.UserId));
                //FindData(1);
                if (IsMobile) {

```

```

        footer.Visible = false;

        mainMenu.InnerHtml = Common.Main_Menu(Convert.ToInt32(Groupid)) +
Common.Generate_TopBarMenu(Clsusr.UserName, Clsusr.RoleId, Clsusr.DealerId,
Convert.ToInt32(Clsusr.UserId));
    }
    else
    {
        mainMenu.InnerHtml = Common.Main_Menu(Convert.ToInt32(Groupid))
+Common.Generate_TopBarMenu(Clsusr.UserName, Clsusr.RoleId, Clsusr.DealerId,
Convert.ToInt32(Clsusr.UserId));

    }
}
}
else
{
    string user;
    HttpCookie myCookie = Request.Cookies["DateOrWedCookie"];
    if (myCookie != null)
    {
        user = myCookie.Value.ToString();
    }
    else
    {
        user = "Guest";
    }

    Session.Clear();
    Session.RemoveAll();
    Session.Abandon();
    //Response.Redirect("MLB_Login.aspx");
    guest_login(user);
}

```

```

    //}

    string script = string.Format("<script type=\"text/javascript\">
LoadxmlPermissions({0}); </script>", "1");

    //ClientScript.RegisterClientScriptBlock(this.GetType(), "myscript", script);
    Page.ClientScript.RegisterClientScriptBlock(this.GetType(), "MakeSure", script);
    //form1.Attributes.Add("onsubmit", "return askBeforeSubmit();");
}
catch (Exception ex)
{
    //Common.ShowMessageHtml(ex.Message, 1, true, this.dvMsgBox);
    ExceptionLogging.SendExcepToDB(ex);

    objlog.Trace_Error(5, "DateOrWed", "MainMaster", "Page_load", ex.Message);

}
}

protected void guest_login(string user)
{

    string City = "Hyderabad";// _City;
    string Country = "India";// _Country;

    BALCreateUser objlogin = new BALCreateUser();

    DataTable status_login = objlogin.LoginGuest(user, user,
Request.ServerVariables["REMOTE_ADDR"].ToString(), "", City, Country);
    string login_status = status_login.Rows[0]["UserStatus"].ToString();

    ClsLoggedInUser objLoggedInUser;

```

```
if (login_status == "True")
{
    objLoggedInUser = new ClsLoggedInUser();

    objLoggedInUser.UserId = status_login.Rows[0]["UserId"].ToString();
    objLoggedInUser.UserName = status_login.Rows[0]["UserName"].ToString();
    objLoggedInUser.GroupId = status_login.Rows[0]["GroupId"].ToString();
    objLoggedInUser.LoginStatus = "True";
    objLoggedInUser.LoginID = Convert.ToInt32(status_login.Rows[0]["LoginId"]);
    objLoggedInUser.DealerId = Convert.ToInt32(status_login.Rows[0]["DealerId"]);
    objLoggedInUser.RoleId = Convert.ToInt32(status_login.Rows[0]["RoleId"]);
    objLoggedInUser.BrandId = 1; //Convert.ToInt32(status_login.Rows[0]["BrandId"]);

    Session["LoggedUser_Specs"] = objLoggedInUser;

    FormsAuthentication.SetAuthCookie(objLoggedInUser.UserName, false);
    HttpCookie myCookie = new HttpCookie("DateOrWedCookie");
    DateTime now = DateTime.Now;
    myCookie.Value = objLoggedInUser.UserName;
    myCookie.Expires = now.AddYears(1);

    Response.Cookies.Add(myCookie);

    ClsPermissioGroup clsperGrp = new ClsPermissioGroup();
    UserPermissions Entuserper =
clsperGrp.GetUserLandingPage(objLoggedInUser.UserName);

    if (Entuserper.Pageld != 0)
    {
        Entuserper.Add = true;
        Entuserper.Edit = true;
        Entuserper.Delete = true;
        Entuserper.View = true;
    }
}
```

```

        //BalRecentPlacess objRemoveRcntPlc = new BalRecentPlacess();

        //bool rst =
objRemoveRcntPlc.DeleteRecentPlaces(Convert.ToInt32(status_login.Rows[0]["UserId"].ToString()));

        //Response.Redirect("~/ " + Entuserper.PageUrl);

    }

}

}

public static string FindData(int id)
{
    try
    {
        List<EteProductGroup> et = new Products().GetProductGroups("").ToList();
        et = (from c in et where c.PCatId == id orderby c.PCatId ascending select c).ToList();

        string js = Newtonsoft.Json.JsonConvert.SerializeObject(et);
        //Common.ShowMessageHtml(js, 1, true, dvMsgBox);
        return js;
    }
    catch (Exception exc)
    {
        DDLogTracer objlog = new DDLogTracer();
        objlog.Trace_Error(4, "DateOrWed", "Ajax_Call", "ResetPermissiomns", "Exception:"
+ exc.ToString());

        ExceptionLogging.SendExcepToDB(exc);

        return "Error: exception - " + exc.ToString();
    }
}
}

```

```
protected void Button1_Click(object sender, EventArgs e)
```

```
{  
    Response.Redirect("~/Login.aspx");
```

```
}
```

```
protected void Button2_Click(object sender, EventArgs e)
```

```
{  
    Response.Redirect("~/Register.aspx");
```

```
}
```

```
public void ShowMessage(string msg)
```

```
{  
    //dvMsgBox.InnerHtml = Common.ShowMessage(msg);  
    //dvMsgBox.Attributes.CssStyle.Add("display", "block;");  
}
```

```
#region Dynamic LeftNavBar
```

```
public static string BindDynamic_DDL(int lbtnid, int loginid, int lbtntype)
```

```
{  
    var sb = new StringBuilder();  
    BalMenus objclsper = new BalMenus();  
    DataSet ds = objclsper.BindDynamic_DDL(lbtnid, loginid, lbtntype);  
    DataTable dtdll = ds.Tables[0];
```

```
    if (ds != null)
```

```
    {
```

```
        if (ds.Tables[0].Rows.Count > 0)
```

```
        {
```

```
            foreach (DataRow drnew in dtdll.Rows)
```

```
            {
```

```
        sb.Append(string.Format(@"<option>{0}</option>",
drnew["TEXT"].ToString()));//;drnew["ID"].ToString();drnew["TEXT"].ToString());
    }
}
}
return sb.ToString();
}
```

#endregion

```
protected void OnMasterSaveClick(EventArgs e)
```

```
{
    if (this.MasterControlSaveClicked != null)
    {
        this.MasterControlSaveClicked(this, e);
    }
}
```

```
//MasterControlClicked
```

```
protected void OnMasterCancelClick(EventArgs e)
```

```
{
    if (this.MasterControlCancelClicked != null)
    {
        this.MasterControlCancelClicked(this, e);
    }
}
```

```
protected void OnMasterControlClick(EventArgs e)
```

```
{
    if (this.MasterControlClicked != null)
    {
        this.MasterControlClicked(this, e);
    }
}
```

```

protected void OnMasterControlURLClick(EventArgs e)
{
    if (this.MasterControlURLClicked != null)
    {
        this.MasterControlURLClicked(this, e);
    }
}

```

```

protected void ImageButton1_Click(object sender, ImageClickEventArgs e)
{
    mainMenu.Attributes.CssStyle.Add("display", "block;");

    //tabsholder.InnerHtml = Common.Generate_Menu();
}
}

```

***** ProductMaster.master *****

```

<%@ Master Language="C#" AutoEventWireup="true" CodeFile="ProductMaster.master.cs"
Inherits="ProductMaster" %>

```

```

<%--<%@ Register Assembly="AjaxControlToolkit" Namespace="AjaxControlToolkit"
TagPrefix="cc1" %>--%>

```

```

<!DOCTYPE html>

```

```

<html xmlns="http://www.w3.org/1999/xhtml">

```

```

<head>

```

```

    <title>Your shopping Destination</title>

```

```

    <meta charset="utf-8"/>

```

```

    <meta name="viewport" content="width=device-width, initial-scale=1.0, shrink-to-fit=yes" />

```

```

    <meta name="description" content="DateOrWed" />

```

```

    <link href="css/font-awesome.css" type="text/css" rel="stylesheet" />

```

```

    <link href="css/font-awesome.min.css" type="text/css" rel="stylesheet" />

```



```
<link rel="stylesheet" type="text/css" href="css/bootstrap.css" />
<link rel="stylesheet" type="text/css" href="css/bootstrap.min.css" />
<link rel="stylesheet" type="text/css" href="css/style.css" />
<%-- <link rel="stylesheet" type="text/css" href="css/style1.css" />--%>
<script type="text/javascript" src="js/jquery-1.6.1.min.js"></script>
<script type="text/javascript" src="js/bootstrap.js"></script>
<script type="text/javascript" src="shopping/js/bootstrap.min.js"></script>
<script type="text/javascript" src="js/jquery.js"></script>
<script type="text/javascript" src="js/bootstrap-collapse.js"></script>
<script type="text/javascript" src="js/bootstrap.min.js"></script>
<script type="text/javascript" src="js/waves.js"></script>
<!-- Font Awesome -->
<link rel="stylesheet" href="Vendor/css/all.css">

<%-- <script src="js/common.js"></script>--%>
<%-- <script src="js/jquery.countdown.js"></script>--%>
<%-- <script src="js/bootstrap.min.js"></script>--%>

<%-- <script type="text/javascript" src="js/ManagePermissions.js"></script>--%>

<%-- <style type="text/css">
    .line {
        /*line-height: 40px;*/
        background: #09F;
        /* word-spacing: 55px; */
        width: 100%;
        margin: 0 auto;
        color: #F5F20C;
```

```
font-family: Arial, Helvetica, sans-serif;
font-size: 9px;
font-weight: bold;
}
```

```
.line a {
  /*padding: 10px 0px 10px 0px;*/
  color: #FFF;
  text-decoration: underline;
  font-weight: normal;
}
```

```
.line a:hover {
  text-decoration: none;
}
```

```
.text {
  padding: 3px 0px;
}
```

```
</style>
```

```
<style type="text/css">
```

```
.line {
  /*line-height: 40px;*/
  background: #09F;
  /* word-spacing: 55px; */
  width: 100%;
  margin: 0 auto;
  color: #F5F20C;
  font-family: Arial, Helvetica, sans-serif;
  font-size: 9px;
  font-weight: bold;
```

```
}
```

```
.line a {  
    /*padding: 10px 0px 10px 0px;*/  
    color: #FFF;  
    text-decoration: underline;  
    font-weight: normal;  
}
```

```
.line a:hover {  
    text-decoration: none;  
}
```

```
.text {  
    padding: 3px 0px;  
}
```

```
</style>--%>
```

```
<asp:ContentPlaceHolder ID="CPH_MainMaster_Header" runat="server">
```

```
</asp:ContentPlaceHolder>
```

```
</head>
```

```
<body>
```

```
    <form id="form1" runat="server" >
```

```
    <!--Navbar -->
```

```
<nav class="mb-1 navbar fixed-top navbar-expand-lg navbar-light bg-light">
```

```
    <a class="navbar-brand" href="VendorHome.aspx"></a>
```

```
<%-- <a class="navbar-brand" href="Shopping.aspx"></a>
```

```
<a class="navbar-brand" href="Services.aspx"></a>
```

```
<a class="navbar-brand" href="BListing.aspx"></a>--%>
```

```
<button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent-4"
```

```
aria-controls="navbarSupportedContent-4" aria-expanded="false" aria-label="Toggle navigation">
```

```
<span class="navbar-toggler-icon"></span>
```

```
</button>
```

```
<div class="collapse navbar-collapse" id="navbarSupportedContent-5">
```

```
<ul class="navbar-nav ml-auto">
```

```
<li class="nav-item">
```

```
<a class="nav-link dropdown-toggle waves-effect" id="navbarDropdownMenuLink" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
```

```
<span class="badge red">3</span> <i class="fas fa-bell"></i>
```

```
<span class="d-none d-md-inline-block">Notifications</span>
```

```
</a>
```

```
<div class="dropdown-menu dropdown-primary pull-left" aria-labelledby="navbarDropdownMenuLink">
```

```
<a class="dropdown-item" href="#">
```

```
<i class="far fa-money-bill-alt mr-2" aria-hidden="true"></i>
```

```
<span>New order received</span>
```

```
<span class="float-right"><i class="far fa-clock" aria-hidden="true"></i> 13 min</span>
```

```
</a>
```

```
<a class="dropdown-item" href="#">
```

```
<i class="far fa-money-bill-alt mr-2" aria-hidden="true"></i>
```

```
<span>New order received</span>
```

```
<span class="float-right"><i class="far fa-clock" aria-hidden="true"></i> 33 min</span>
```

```
</a>
```

```
<a class="dropdown-item" href="#">
```

```
<i class="fas fa-chart-line mr-2" aria-hidden="true"></i>
<span>Your campaign is about to end</span>
<span class="float-right"><i class="far fa-clock" aria-hidden="true"></i> 53
min</span>
</a>
</div>
</li>
<li class="nav-item">
  <a class="nav-link waves-effect"><i class="fas fa-envelope"></i> <span
class="clearfix d-none d-sm-inline-block">Contact</span></a>
</li>
<li class="nav-item">
  <a class="nav-link waves-effect"><i class="far fa-comments"></i> <span
class="clearfix d-none d-sm-inline-block">Support</span></a>
</li>
<li class="nav-item">
  <a class="nav-link dropdown-toggle waves-effect" href="#" id="userDropdown"
data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
    <i class="fas fa-user"></i> <span class="clearfix d-none d-sm-inline-
block">Profile</span>
  </a>
  <div class="dropdown-menu dropdown-menu-right" aria-
labelledby="userDropdown">
    <a class="dropdown-item" href="logout.aspx">Log Out</a>
    <a class="dropdown-item" href="#">My account</a>
  </div>
</li>
</ul>
</div>
</nav>
<!--/.Navbar -->

<div class="col-lg-12">
```

```

<asp:ScriptManager ID="ScriptManager1" runat="server">
</asp:ScriptManager>
<!-- Top login detail bar Div/Section Start -->
<div>
<%-- <div class="collapse navbar-collapse" id="navbarSupportedContent-5">

<div class="span2" id="Div1" runat="server" style="display: block;">
    
</div>

</div>--%>

<asp:ContentPlaceHolder ID="CPH_TopMessages" runat="server">
    <div class="container-fluid">

        <div class="row-fluid">    <%-- <div class="row-fluid" style="padding-top:
20px;">--%>
            <div class="span2" id="dvlogo" runat="server" style="display: none;">
<%--
                
--%>

            </div>
            <div class="span10">
                <div class="row">
                    <div id="dvMsgBox" class="span12 ordersave" runat="server"
style="display: block;">
                        <asp:Label ID="Label1" runat="server" Text=""></asp:Label>
                    </div>
                </div>
            </div>
        </div>
    </div>

```

```
        </div>
    </div>
</asp:ContentPlaceHolder>
<!-- Top Logo/Messages Div/Section End -->
    <!-- Left Bar Div/Section Start -->
        <asp:ContentPlaceHolder ID="CPH_LeftSideBar" runat="server">
<%--
            <div id="LeftSideBar" runat="server" class="span2">
                <asp:Table ID="Table1" runat="server">
                    </asp:Table>

            </div>
--%>
        </asp:ContentPlaceHolder>
    <!-- Left Bar Div/Section End -->

    <!-- Main Content Div/Section Start -->
    <asp:ContentPlaceHolder ID="CPH_MainContent" runat="server">

    </asp:ContentPlaceHolder>

</div>
    </div>
<asp:ContentPlaceHolder ID="ContentPlaceHolder1" runat="server">

    <div id="footer" runat="server">

<div class="app-footer">
    <div class="homeSection homeCatalog">
    <p class="homeSection__extraTitle">Wedding Shopping gallery</p>
    <p class="homeSection__title">Catalogue of bridal & Groom fashion</p>
```

```
<p class="homeCatalog__text">
```

```
    Discover the latest trends in wedding dresses by top designers and bridesmaid dresses. Choose your favourite from our catalogue! </p>
```

```
<div class="homeCatalog__categories">
```

```
    <div class="homeCatalog__categoriesItem">
```

```
        <span class="app-link" data-href="/GOrderSearchList.aspx?pid=2">
```

```
            <i class="icon-catalog-categories icon-catalog-categories-bride-dress block mb40"></i>
```

```
            <a href="GOrderSearchList.aspx?pid=2">Wedding Dresses</a>
```

```
        </span>
```

```
    </div>
```

```
    <div class="homeCatalog__categoriesItem">
```

```
        <span class="app-link" data-href="/GOrderSearchList.aspx?pid=2">
```

```
            <i class="icon-catalog-categories icon-catalog-categories-suit block mb40"></i>
```

```
            <a href="GOrderSearchList.aspx?pid=2">Grooms suits</a>
```

```
        </span>
```

```
    </div>
```

```
    <div class="homeCatalog__categoriesItem">
```

```
        <span class="app-link" data-href="/GOrderSearchList.aspx?pid=37">
```

```
            <i class="icon-catalog-categories icon-catalog-categories-godmum block mb40"></i>
```

```
            <a href="OrderSearchList.aspx?pid=37">Bridesmaid Dresses</a>
```

```
        </span>
```

```
    </div>
```

```
    <div class="homeCatalog__categoriesItem">
```

```
        <span class="app-link" data-href="/GOrderSearchList.aspx?pid=54">
```

```
            <i class="icon-catalog-categories icon-catalog-categories-dress block mb40"></i>
```

```
            <a href="GOrderSearchList.aspx?pid=54">Party Dresses</a>
```

```
        </span>
```

```
    </div>
```

```
    <div class="homeCatalog__categoriesItem">
```

```
        <span class="app-link" data-href="/OrderSearchList.aspx?id=2">
```



```
<i class="icon-catalog-categories icon-catalog-categories-diamond block
mb40"></i>
  <a href="OrderSearchList.aspx?id=2">Jewellery</a>
</span>
</div>
  <div class="homeCatalog__categoriesItem">
    <span class="app-link" data-href="/OrderSearchList.aspx?id=3">
      <i class="icon-catalog-categories icon-catalog-categories-shoe block
mb40"></i>
        <a href="OrderSearchList.aspx?id=3">Shoes</a>
      </span>
    </div>
      <div class="homeCatalog__categoriesItem">
        <span class="app-link" data-href="/OrderSearchList.aspx?pid=16">
          <i class="icon-catalog-categories icon-catalog-categories-bra block mb40"></i>
            <a href="OrderSearchList.aspx?pid=16">Lingerie</a>
          </span>
        </div>
          <div class="homeCatalog__categoriesItem">
            <span class="app-link" data-href="/OrderSearchList.aspx?id=16">
              <i class="icon-catalog-categories icon-catalog-categories-handbag block
mb40"></i>
                <a href="OrderSearchList.aspx?id=16">Accessories</a>
              </span>
            </div>
          </div>
        <span class="btn-outline outline-red mt40 app-link" data-
href="OrderSearchList.aspx?id=1">
          &nbsp;See catalogue&nbsp;
        </span>
      </div>
        <div class="pb25">
          <div class="home-category">
```

```
<div class="wrapper">
  <p class="homeSection__title homeSection__title--md mt45">Wedding Services by
category</p>
  <div class="pure-g mt40">
    <div class="pure-u-1-3">
      <a class="homeCategories__titleCategory homeCategories__titleCategory--
catering" href="wedding-venues">
        Wedding Venues      </a>
      <ul class="pure-g homeCategories__list">
        <li class="pure-u-1-2 homeCategories__listItem">
          <a class="homeCategories__listAnchor" href="FunctionHalls">
            Function Halls      </a>
          </li>
          <li class="pure-u-1-2 homeCategories__listItem">
            <a class="homeCategories__listAnchor" href="hotels">
              Hotels      </a>
            </li>
            <li class="pure-u-1-2 homeCategories__listItem">
              <a class="homeCategories__listAnchor" href="banquet-halls">
                Banquet Halls      </a>
            </li>
            <li class="pure-u-1-2 homeCategories__listItem">
              <a class="homeCategories__listAnchor" href="party-lounge">
                Party Lounge      </a>
            </li>
            <li class="pure-u-1-2 homeCategories__listItem">
              <a class="homeCategories__listAnchor" href="farmhouse">
                Farm Houses      </a>
            </li>
            <li class="pure-u-1-2 homeCategories__listItem">
              <a class="homeCategories__listAnchor" href="temples">
                Temple Function Halls      </a>
            </li>
          </ul>
        </div>
      </div>
    </div>
```


</div>

<div class="pure-u-1-3">

Wedding Suppliers

<ul class="homeCategories__list pure-g">

<li class="pure-u-1-2 homeCategories__listItem">

Catering

<li class="pure-u-1-2 homeCategories__listItem">

Wedding Invitations

<li class="pure-u-1-2 homeCategories__listItem">

Wedding Gifts

<li class="pure-u-1-2 homeCategories__listItem">

Wedding Photography

<li class="pure-u-1-2 homeCategories__listItem">

Wedding Music

<li class="pure-u-1-2 homeCategories__listItem">

Wedding Transportation

<li class="pure-u-1-2 homeCategories__listItem">

```
<a class="homeCategories__listAnchor" href="tent-house">
  Tent House          </a>
</li>
  <li class="pure-u-1-2 homeCategories__listItem">
    <a class="homeCategories__listAnchor" href="wedding-entertainment">
      Wedding Entertainment          </a>
    </li>
      <li class="pure-u-1-2 homeCategories__listItem">
        <a class="homeCategories__listAnchor" href="florists">
          Florists          </a>
        </li>
          <li class="pure-u-1-2 homeCategories__listItem">
            <a class="homeCategories__listAnchor" href="wedding-planners">
              Wedding Planners          </a>
            </li>
              <li class="pure-u-1-2 homeCategories__listItem">
                <a class="homeCategories__listAnchor" href="wedding-videography">
                  Wedding Videography          </a>
                </li>
                  <li class="pure-u-1-2 homeCategories__listItem">
                    <a class="homeCategories__listAnchor" href="honeymoon">
                      Honeymoon          </a>
                    </li>
                      <li class="pure-u-1-2 homeCategories__listItem">
                        <a class="homeCategories__listAnchor" href="wedding-decoration">
                          Wedding Decoration          </a>
                        </li>
                          <li class="pure-u-1-2 homeCategories__listItem">
                            <a class="homeCategories__listAnchor" href="wedding-cakes">
                              Wedding Cakes          </a>
                            </li>
                              <li class="pure-u-1-2 homeCategories__listItem">
```

```
<a class="homeCategories__listAnchor" href="wedding-dj">
  Wedding DJ      </a>
</li>
  <li class="pure-u-1-2 homeCategories__listItem">
    <a class="homeCategories__listAnchor" href="photobooth">
      Photobooth      </a>
    </li>
      <li class="pure-u-1-2 homeCategories__listItem">
        <a class="homeCategories__listAnchor" href="astrologers">
          Astrologers      </a>
        </li>
          <li class="pure-u-1-2 homeCategories__listItem">
            <a class="homeCategories__listAnchor" href="wedding-choreography">
              Wedding Choreography      </a>
            </li>
          </ul>
        </div>
      <div class="pure-u-1-3">
        <div class="pure-g">
          <div class="pure-u-1-2">
            <a class="homeCategories__titleCategory homeCategories__titleCategory--brides" href="brides">
              Brides      </a>
            <ul class="homeCategories__list">
              <li class="homeCategories__listItem">
                <a class="homeCategories__listAnchor" href="mehndi-design">
                  Mehndi Design      </a>
                </li>
                  <li class="homeCategories__listItem">
                    <a class="homeCategories__listAnchor" href="bridal-makeup">
                      Bridal Makeup      </a>
                    </li>
                  </li>
                </ul>
              </div>
            </div>
          </div>
        </div>
      </div>
    </div>
  </div>
</div>
```

```
                <li class="homeCategories__listItem">
                    <a class="homeCategories__listAnchor" href="makeup-salon">
                        Makeup Salon                </a>
                </li>

                <li class="homeCategories__listItem">
                    <a class="homeCategories__listAnchor" href="bridal-jewellery">
                        Bridal Jewellery            </a>
                </li>

                <li class="homeCategories__listItem">
                    <a class="homeCategories__listAnchor" href="bridal-lehenga">
                        Bridal Lehenga              </a>
                </li>

                <li class="homeCategories__listItem">
                    <a class="homeCategories__listAnchor" href="trousseau-packing">
                        Trousseau Packing            </a>
                </li>
            </ul>
        </div>

        <div class="pure-u-1-2">
            <a class="homeCategories__titleCategory homeCategories__titleCategory--
grooms" href="grooms">
                Grooms                </a>
            <ul class="homeCategories__list">

                <li class="homeCategories__listItem">
                    <a class="homeCategories__listAnchor" href="sherwani">
                        Sherwani          </a>
                </li>

                <li class="homeCategories__listItem">
                    <a class="homeCategories__listAnchor" href="Coats">
                        Coats              </a>
                </li>
            </ul>
        </div>
```

```
        </div>
    </div>
</div>
</div>
</div>
</div>
</div>
</div>

<div class="white pb25">
    <div class="wrapper">
        <p class="footer-chevron-title">Wedding Venues by location</p>
        <ul class="pure-g footer-list">
            <li class="pure-u-1-4">
                <a href="wedding-venues/delhi-ncr">
                    Wedding Venues Delhi NCR    </a>
            </li>
            <li class="pure-u-1-4">
                <a href="wedding-venues/mumbai">
                    Wedding Venues Mumbai    </a>
            </li>
            <li class="pure-u-1-4">
                <a href="wedding-venues/bangalore">
                    Wedding Venues Bangalore    </a>
            </li>
            <li class="pure-u-1-4">
                <a href="wedding-venues/hyderabad">
                    Wedding Venues Hyderabad    </a>
            </li>
            <li class="pure-u-1-4">
                <a href="wedding-venues/ap">
                    Wedding Venues Andra Pradesh    </a>
            </li>
            <li class="pure-u-1-4">
```

```
    <a href="wedding-venues/Telengana">
      Wedding Venues Telengana    </a>
  </li>
</ul>
</div>
</div>
```

```
</div>
```

```
<div id="app-common-layer" class="modal fade dnone" tabindex="-1" role="dialog"
aria-hidden="true"></div>
```

```
<div id="app-va-modal" class="modal fade dnone" tabindex="-1" role="dialog" aria-
hidden="true"></div>
```

```
</div>
```

```
</div>
```

```
<%-- shopping.dateorwed.com Version 1.00.01. <a
href="http://DateOrWed.com:699/Heartbeat">Heartbeat</a>
```

```
Copyright © 2018 Ravikirantechnosoft&vtechtechnologies.com.
```

```
--%>
```

```
</asp:ContentPlaceholder>
```



```
</form>
```

```
</body>
```

```
</html>
```

```
***** ProductMaster.master.cs*****
```

```
using System;
```

```
using System.Collections.Generic;
```

```
using System.Linq;
```

```
using System.Web;
```

```
using System.Web.UI;
```

```
using System.Web.UI.WebControls;
```

```
using DowBEntity;
```

```
using DowBAL;
```

```
using System.Text;
```

```
using System.Data;
```

```
using System.Web.UI.HtmlControls;
```

```
using System.Text.RegularExpressions;
```

```
using System.Web.Security;
```

```
//using LogTracer;
```

```
public partial class ProductMaster : System.Web.UI.MasterPage
```

```
{
```

```
    public event EventHandler MasterControlClicked;
```

```
    public event EventHandler MasterControlSaveClicked;
```

```
    public event EventHandler MasterControlCancelClicked;
```

```
    public event EventHandler MasterControlURLClicked;
```

```

public DDLogTracer objlog = new DDLogTracer();

protected void Page_Load(object sender, EventArgs e)
{
    bool IsMobile = false;
    if (Request.Browser.IsMobileDevice) { IsMobile = true; } else { IsMobile = false; }
    string user="";

    try
    {
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr != null)
        {
            if (!Page.IsPostBack)
            {
                int Groupid =Convert.ToInt16( Clsusr.GroupId);

                string pageid = Request.QueryString["pageid"];
                string fpath = Request.QueryString["fpath"];
                Session["PageType"] = "";
                if (IsMobile) {
                }
                else
                {

                }
            }
        }
        else
        {
            HttpCookie myCookie = Request.Cookies["DateOrWedCookie"];

```

```

        if (myCookie != null)
        {
            user = myCookie.Value.ToString();
        }

        Session.Clear();
        Session.RemoveAll();
        Session.Abandon();
        guest_login(user);
    }
    //}

    string script = string.Format("<script type=\"text/javascript\">
LoadxmlPermissions({0}); </script>", "1");
    Page.ClientScript.RegisterClientScriptBlock(this.GetType(), "MakeSure", script);
}
catch (Exception ex)
{
    //Common.ShowMessageHtml(ex.Message, 1, true, this.dvMsgBox);
    ExceptionLogging.SendExcepToDB(ex);

    objlog.Trace_Error(5, "DateOrWed", "MainMaster", "Page_load", ex.Message);

}
}

protected void guest_login(string user)
{

    string City = "Hyderabad";// _City;

```

```
string Country = "India";// _Country;
```

```
BALCreateUser objlogin = new BALCreateUser();
```

```
DataTable status_login = objlogin.LoginGuest(user, user,  
Request.ServerVariables["REMOTE_ADDR"].ToString(), "", City, Country);
```

```
string login_status = status_login.Rows[0]["UserStatus"].ToString();
```

```
ClsLoggedInUser objLoggedInUser;
```

```
if (login_status == "True")
```

```
{
```

```
    objLoggedInUser = new ClsLoggedInUser();
```

```
    objLoggedInUser.UserId = status_login.Rows[0]["UserId"].ToString();
```

```
    objLoggedInUser.UserName = status_login.Rows[0]["UserName"].ToString();
```

```
    objLoggedInUser.GroupId = status_login.Rows[0]["GroupId"].ToString();
```

```
    objLoggedInUser.LoginStatus = "True";
```

```
    objLoggedInUser.LoginID = Convert.ToInt32(status_login.Rows[0]["LoginId"]);
```

```
    objLoggedInUser.DealerId = Convert.ToInt32(status_login.Rows[0]["DealerId"]);
```

```
    objLoggedInUser.RoleId = Convert.ToInt32(status_login.Rows[0]["RoleId"]);
```

```
    objLoggedInUser.BrandId = 1; //Convert.ToInt32(status_login.Rows[0]["BrandId"]);
```

```
    Session["LoggedUser_Specs"] = objLoggedInUser;
```

```
    FormsAuthentication.SetAuthCookie(objLoggedInUser.UserName, false);
```

```
    HttpCookie myCookie = new HttpCookie("DateOrWedCookie");
```

```
    DateTime now = DateTime.Now;
```

```
    myCookie.Value = objLoggedInUser.UserName;
```

```
    myCookie.Expires = now.AddYears(1);
```

```
    Response.Cookies.Add(myCookie);
```

```
    ClsPermissioGroup clsperGrp = new ClsPermissioGroup();
```

```

        UserPermissions Entuserper =
clsperGrp.GetUserLandingPage(objLoggedInUser.UserName);

        if (Entuserper.Pageld != 0)
        {
            Entuserper.Add = true;
            Entuserper.Edit = true;
            Entuserper.Delete = true;
            Entuserper.View = true;

            //BalRecentPlacess objRemoveRcntPlc = new BalRecentPlacess();

            //bool rst =
objRemoveRcntPlc.DeleteRecentPlaces(Convert.ToInt32(status_login.Rows[0]["UserId"].ToString()));

            //Response.Redirect("~/ " + Entuserper.PageUrl);

        }

    }
}

public static string FindData(int id)
{
    try
    {
        List<EteProductGroup> et = new Products().GetProductGroups("").ToList();
        et = (from c in et where c.PCatId == id orderby c.PCatId ascending select c).ToList();

        string js = Newtonsoft.Json.JsonConvert.SerializeObject(et);
        //Common.ShowMessageHtml(js, 1, true, dvMsgBox);

        return js;
    }
    catch (Exception exc)
    {
        DDLogTracer objlog = new DDLogTracer();
    }
}

```

```
objlog.Trace_Error(4, "DateOrWed", "Ajax_Call", "ResetPermissiomns", "Exception:"  
+ exc.ToString());
```

```
ExceptionLogging.SendExcepToDB(exc);
```

```
return "Error: exception - " + exc.ToString();
```

```
}
```

```
}
```

```
protected void Button1_Click(object sender, EventArgs e)
```

```
{
```

```
Response.Redirect("~/Login.aspx");
```

```
}
```

```
protected void Button2_Click(object sender, EventArgs e)
```

```
{
```

```
Response.Redirect("~/Register.aspx");
```

```
}
```

```
public void ShowMessage(string msg)
```

```
{
```

```
//dvMsgBox.InnerHtml = Common.ShowMessage(msg);
```

```
//dvMsgBox.Attributes.CssStyle.Add("display", "block;");
```

```
}
```

```
#region Dynamic LeftNavBar
```

```
public static string BindDynamic_DDL(int lbtnid, int loginid, int lbtntype)
```

```
{
```

```
var sb = new StringBuilder();
```

```
BalMenus objclsper = new BalMenus();
```

```
DataSet ds = objclsper.BindDynamic_DDL(lbtnid, loginid, lbtntype);
```

```

DataTable dtdll = ds.Tables[0];

if (ds != null)
{
    if (ds.Tables[0].Rows.Count > 0)
    {
        foreach (DataRow drnew in dtdll.Rows)
        {
            sb.Append(string.Format(@"<option>{0}</option>",
drnew["TEXT"].ToString());//;drnew["ID"].ToString();drnew["TEXT"].ToString();
        }
    }
}
return sb.ToString();
}

```

#endregion

```

protected void OnMasterSaveClick(EventArgs e)
{
    if (this.MasterControlSaveClicked != null)
    {
        this.MasterControlSaveClicked(this, e);
    }
}
//MasterControlClicked
protected void OnMasterCancelClick(EventArgs e)
{
    if (this.MasterControlCancelClicked != null)
    {
        this.MasterControlCancelClicked(this, e);
    }
}

```

```
protected void OnMasterControlClick(EventArgs e)
{
    if (this.MasterControlClicked != null)
    {
        this.MasterControlClicked(this, e);
    }
}
```

```
protected void OnMasterControlURLClick(EventArgs e)
{
    if (this.MasterControlURLClicked != null)
    {
        this.MasterControlURLClicked(this, e);
    }
}
```

```
protected void ImageButton1_Click(object sender, ImageClickEventArgs e)
{
    //mainMenu.Attributes.CssStyle.Add("display", "block;");

    //tabsholder.InnerHtml = Common.Generate_Menu();
}
}
```

This concludes Master Windows development

Chapter 8

Create Application windows inheriting the master windows
Now let us drive into developing Windows in the application

```
***** Ajax_Call.aspx *****
```

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Ajax_Call.aspx.cs"  
Inherits="Ajax_Call" %>
```

```
<!DOCTYPE html>
```

```
<html xmlns="http://www.w3.org/1999/xhtml">  
<head runat="server">  
  <title></title>  
</head>  
<body>  
  <form id="form1" runat="server">  
    <div>  
  
    </div>  
  </form>  
</body>  
</html>
```

```
***** Ajax_Call.aspx.cs *****
```

```
using DowBAL;  
using DowBEntity;  
using System;  
using System.Collections.Generic;  
using System.Linq;  
using System.Web;  
using System.Web.UI;  
using System.Web.UI.WebControls;  
using System.Web.Script.Serialization;  
//using LogTracer;  
using System.Configuration;  
using System.IO;
```

```
public partial class Ajax_Call : System.Web.UI.Page  
{  
  DDLogTracer objlog = new DDLogTracer();  
  public string TraceThreadId;  
  private static string logDirectory;  
  private static StreamWriter sw;  
  
  protected void Page_Load(object sender, EventArgs e)  
  {  
  
  }  
  [System.Web.Services.WebMethod]  
  public static string FindData(int id)  
  {  
    List<PopDDL> result = new List<PopDDL>();  
  
    try
```

```

    {
        List<EteProductGroup> et = new Products().GetProductGroups("").ToList();
        foreach (var row in et)
        {
            if (row.PCatId == id)
            {
                result.Add(new PopDDL { Text = row.pgroupdescription, Value =
row.pgjid.ToString() });
            }
        }
        string js = Newtonsoft.Json.JsonConvert.SerializeObject(et);
        return js;
    }
    catch (Exception exc)
    {
        DDLogTracer objlog = new DDLogTracer();
        objlog.Trace_Error(4, "DateOrWed", "AjaxCall", "ResetPermissiomns", "Exception:" +
exc.ToString());

        ExceptionLogging.SendExcepToDB(exc);

        return "Error: exception - " + exc.ToString();
    }
}
[System.Web.Services.WebMethod]
public static string ResetMenus(string groupid)
{
    try
    {
        if (groupid.Length <= 0)
        {
            return "Error: groupid not send.";
        }
        else
        {
            List<EteProductGroup> et = new Products().GetProductGroups("").ToList();
            et = (from c in et where c.PCatId == Convert.ToInt16( groupid) orderby c.PCatId
ascending select c).ToList();

            string js = Newtonsoft.Json.JsonConvert.SerializeObject(et);
            return js;
            //return PerGrp;
        }
    }
    catch (Exception exc)
    {
        //DDLogTracer objlog = new DDLogTracer();
        //objlog.Trace_Error(4, "DateOrWed", "AjaxCall", "ResetPermissiomns", "Exception:"
+ exc.ToString());

        return "Error: exception - " + exc.ToString();
    }
}

```

```

    }
}
[System.Web.Services.WebMethod]
public static string ResetPermissions(string groupid)
{
    try
    {
        if (groupid.Length <= 0)
        {
            return "Error: groupid not send.";
        }
        else
        {
            PermissionsGroup[] PerGrp = new PermissionsGroup[] { };
            PerGrp = GetAttributesList(Convert.ToInt32(groupid), 1);
            string js = Newtonsoft.Json.JsonConvert.SerializeObject(PerGrp);
            return js;
            //return PerGrp;
        }
    }
    catch (Exception exc)
    {
        //DDLogTracer objlog = new DDLogTracer();
        //objlog.Trace_Error(4, "DateOrWed", "Ajax_Call", "ResetPermissiomns",
"Exception:" + exc.ToString());

        return "Error: exception - " + exc.ToString();
    }
}

```

```

[System.Web.Services.WebMethod]
public static string ResetProducts(string groupid)
{
    try
    {
        if (groupid.Length <= 0)
        {
            return "Error: ProductID not send.";
        }
        else
        {
            PermissionsGroup[] PerGrp = new PermissionsGroup[] { };
            PerGrp = GetProductsList(Convert.ToInt32(groupid), 1);
            string js = Newtonsoft.Json.JsonConvert.SerializeObject(PerGrp);
            return js;
            //return PerGrp;
        }
    }
    catch (Exception exc)
    {
        //DDLogTracer objlog = new DDLogTracer();

```

```

        //objlog.Trace_Error(4, "DateOrWed", "Ajax_Call", "ResetPermissiomns",
"Exception:" + exc.ToString());

        return "Error: exception - " + exc.ToString();
    }
}
public static PermissionsGroup[] GetAttributesList(int Groupid, int type)
{
    try
    {
        List<PermissionsGroup> Pergrp = new List<PermissionsGroup>();
        List<mapids> lst = new List<mapids>();
        int Pageid = 0;

        lst = new tblGrpMappings().GetPerSetsForGroup(Groupid, type).ToList();
        int lstindex = lst.Count();
        PermissionsGroup[] PerGroup;

        foreach (var Pgrow in lst)
        {

            Pergrp.Add(new PermissionsGroup
            {
                //"mapid-" + mapid;
                mpageid = "mapid-" + Pgrow.mpageid,
                //mpageid = Pgrow.mpageid,
                status = Pgrow.status

            });

            Pageid = Pgrow.pageid;
            List<mapids> blst = new List<mapids>();
            blst = new tblGrpMappings().GetLeftButtonsPermission(Pageid, Groupid).ToList();
            foreach (var Btrow in blst)
            {
                Pergrp.Add(new PermissionsGroup
                {
                    //"button-" + pagebtid;
                    buttonid = "button-" + Btrow.buttonid,
                    //buttonid = Btrow.buttonid,
                    status = Btrow.status

                });
            }
        }
        PerGroup = Pergrp.ToArray();
        return PerGroup;
    }
    catch (Exception exc)
    {
        DDLogTracer objlog = new DDLogTracer();
        objlog.Trace_Error(4, "DateOrWed", "Ajax_Call", "Load", "Exception:" +
exc.ToString());
    }
}

```

```

        return null;
    }
}
public static PermissionsGroup[] GetProductsList(int Groupid, int type)
{
    try
    {
        List<PermissionsGroup> Pergrp = new List<PermissionsGroup>();
        List<mapids> lst = new List<mapids>();
        int Pageid = 0;

        lst = new Products().GetPerSetsForProduct(Groupid, type).ToList();
        int lstindex = lst.Count();
        PermissionsGroup[] PerGroup;

        foreach (var Pgrow in lst)
        {

            Pergrp.Add(new PermissionsGroup
            {
                //"mapid-" + mapid;
                mpageid = "mapid-" + Pgrow.mpageid,
                //mpageid = Pgrow.mpageid,
                status = Pgrow.status

            });

            Pageid = Pgrow.pageid;
            List<mapids> blst = new List<mapids>();
            blst = new tblGrpMappings().GetLeftButtonsPermission(Pageid, Groupid).ToList();
            foreach (var Btrow in blst)
            {
                Pergrp.Add(new PermissionsGroup
                {
                    //"button-" + pagebtnid;
                    buttonid = "button-" + Btrow.buttonid,
                    //buttonid = Btrow.buttonid,
                    status = Btrow.status

                });
            }
        }
        PerGroup = Pergrp.ToArray();
        return PerGroup;
    }
    catch (Exception exc)
    {
        DDLogTracer objlog = new DDLogTracer();
        objlog.Trace_Error(4, "DateOrWed", "Ajax_Call", "Load", "Exception:" +
exc.ToString());

        return null;
    }
}
[System.Web.Services.WebMethod]
public static string InsertPermissions(string XmlAttr, string GroupName)

```

```

{
    try
    {
        if (XmlAttribute.Length <= 2 && GroupName.Length <= 0) { return "Group name or
permissions string is empty."; }
        else
        {
            string strgroupid = InsertGroupName(GroupName);
            int groupid;
            bool res = int.TryParse(strgroupid, out groupid);
            if (res == false)
            {
                // String is not a number.
                return strgroupid;
            }

            List<EtetblGrpMapping> pagelst = new List<EtetblGrpMapping>();
            List<EtetblPageButtonMapping> buttonlst = new List<EtetblPageButtonMapping>();

            int i_pagebtnid, i_mapid, status;
            string failed = "";

            PermissionsGroup[] PerGrp =
Newtonsoft.Json.JsonConvert.DeserializeObject<PermissionsGroup[]>(XmlAttribute);

            for (int i = 0; i < PerGrp.GetUpperBound(0); i++)
            {
                if (PerGrp[i].mpageid != null && PerGrp[i].mpageid != "")
                {
                    string mapid = PerGrp[i].mpageid.ToString().Split('-').Last();

                    i_mapid = Convert.ToInt32(mapid);
                    //i_mapid = Convert.ToInt32(PerGrp[i].mpageid);
                    status = Convert.ToInt32(PerGrp[i].status);

                    pagelst.Add(new EtetblGrpMapping { MapId = i_mapid, view =
Convert.ToBoolean(status), GroupId = groupid });
                }
                else if (PerGrp[i].buttonid != null && PerGrp[i].buttonid != "")
                {
                    string pagebuttonid = PerGrp[i].buttonid.ToString().Split('-').Last();
                    i_pagebtnid = Convert.ToInt32(pagebuttonid);
                    //i_pagebtnid = Convert.ToInt32(PerGrp[i].buttonid);
                    status = Convert.ToInt32(PerGrp[i].status);
                    buttonlst.Add(new EtetblPageButtonMapping { pagebtnid = i_pagebtnid, show
= status, groupid = groupid });
                }
                else
                {
                    failed = failed + System.Environment.NewLine + "pageid:" +
PerGrp[i].mpageid + "|buttonid:" + PerGrp[i].buttonid + "|status:" + PerGrp[i].status;
                }
            }
        }
    }
}

```

```

    }

    bool pageresult = false, buttonresult = false;
    //bool result = false;
    //result=(new tblGrpMappings()).InsertBulk(lst);

    pageresult = new tblGrpMappings().InsertBulk(pagelst);
    buttonresult = new tblPageButtonMapings().InsertBulk(buttonlst);
//.InsertBulk(buttonlst);
    }
    return "Role has been created successfully.";
    }
    catch (Exception exc)
    {
        DDLogTracer objlog = new DDLogTracer();
        objlog.Trace_Error(4, "DateOrWed", "Ajax_Call", "Load", "Exception:" +
exc.ToString());

        return "Error occured:" + exc.ToString();
    }
}

[System.Web.Services.WebMethod]
public static string InsertGroupName(string groupname)
{
    string s = "";

    ClsLoggedInUser Clsusr =
(ClsLoggedInUser)HttpContext.Current.Session["LoggedInUser_Specs"];
    if (Clsusr == null)
    {
        return "Session Has expired.";
    }

    EtetblPermissionGroup etegroup = new EtetblPermissionGroup();
    etegroup.GroupName = groupname;
    etegroup.Status = true;
    etegroup.UserId = Convert.ToInt32(Clsusr.UserId);

    if (new tblPermissionGroups().tblPermissionGroupNameExistsbyName(groupname) ==
false)
    {
        int groupid = new tblPermissionGroups().Insert(etegroup);
        s = groupid.ToString();
    }
    else
    {
        s = "Group name " + groupname + " already exist.";
    }
    return s;
}
protected void AddtoShortList(int psid)

```

```

{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];

    //int psid = Convert.ToInt16(Request.QueryString["pid"]);
    EteUserShortList lst = new EteUserShortList();
    lst.UserId = Convert.ToInt16(Clsusr.UserId);
    lst.PSId = psid;
    lst.UId = Convert.ToInt16(Clsusr.UserId);

    lst.CreatedOn = DateTime.Today;
    lst.Active = true;
    bool result = new Products().InsertShortList(lst, true);

}
[System.Web.Services.WebMethod]
public static string UpdateAttributes(string XmlAttr)
{
    try
    {
        if (XmlAttr.Length <= 2) { return "Permissions string is empty."; }
        else
        {
            List<EteTblGrpMapping> lstgrpmap = new List<EteTblGrpMapping>();
            List<EteTblPageButtonMapping> lstbtn = new List<EteTblPageButtonMapping>();

            int i_pagebtnid, i_mapid, status;
            string failed = "";

            PermissionsGroup[] PerGrp =
Newtonsoft.Json.JsonConvert.DeserializeObject<PermissionsGroup[]>(XmlAttr);

            for (int i = 0; i < PerGrp.GetUpperBound(0); i++)
            {
                if (PerGrp[i].mpageid != null && PerGrp[i].mpageid != "")
                {

                    string mapid = PerGrp[i].mpageid.ToString().Split('-').Last();
                    i_mapid = Convert.ToInt32(mapid);
                    //i_mapid = Convert.ToInt32(PerGrp[i].mpageid);
                    status = Convert.ToInt32(PerGrp[i].status);
                    lstgrpmap.Add(new EteTblGrpMapping { MapId = i_mapid, view =
Convert.ToBoolean(status), PageId = 170, UId = 0 });
                }
                else if (PerGrp[i].buttonid != null && PerGrp[i].buttonid != "")
                {
                    string pagebuttonid = PerGrp[i].buttonid.ToString().Split('-').Last();
                    i_pagebtnid = Convert.ToInt32(pagebuttonid);
                    //i_pagebtnid = Convert.ToInt32(PerGrp[i].buttonid);
                    status = Convert.ToInt32(PerGrp[i].status);
                    lstbtn.Add(new EteTblPageButtonMapping
                    {
                        pagebtnid = i_pagebtnid,
                        show = status,

```



```

        pageid = 170,
        Uid = 0
    });
}
else
{
    failed = failed + System.Environment.NewLine + "pageid:" +
PerGrp[i].mpageid + "|buttonid:" + PerGrp[i].buttonid + "|status:" + PerGrp[i].status;
}

}

bool pageresult = false, buttonresult = false;

pageresult = new tblGrpMappings().UpdatePageList(lstgrpmap);
//UpdateButtonList
buttonresult = new tblPageButtonMapings().UpdateButtonList(lstbtn);
}
return "Role Permissions have been updated successfully.";
}
catch (Exception exe)
{
    DDLogTracer objlog = new DDLogTracer();
    objlog.Trace_Error(4, "DateOrWed", "Ajax_call", "Load", "Exception:" +
exe.ToString());

    return "Error occured:" + exe.ToString();
}
}

private static string[] ConvertToArray(string VolBonList)
{
    string[] Result = { "0" };

    if (VolBonList.Length > 0)
    {
        string RemoveUnwantedString = VolBonList.Replace("ctl00$CPH_MainContent$",
""");
        char[] delimiterChars = { ',' };
        Result = RemoveUnwantedString.Split(delimiterChars);
    }
    return Result;
}

private static string FieldValidation(string actionType, string dealerID, string groupName,
string startDate, string endDate)
{
    DateTime std = Convert.ToDateTime(startDate);
    DateTime etd = Convert.ToDateTime(endDate);

    if (string.IsNullOrEmpty(groupName))
    {
        return "Invlid Group Name.";
    }
}

```

```

    }
    if (etd <= std)
    {
        return "Invlid Date Range.";
    }

    string getValidate = ""; // new
    cmnVolumeMatrixs().sp_GetVolMatrixFieldValidation(actionType == "ADD" ? 1 : 0,
    Convert.ToInt32(dealerID), groupName, startDate, endDate);
    return getValidate;
}

public void Trace_Error(int TL_Type, string ModuleName, string Page, string Block, string
ExceptionMessage, string path = "")
{
    if ((ConfigurationSettings.AppSettings["level5"] == "1"))
    {
        string Error_Write_String = string.Format("{0} - [{1}]:[{2}]:[{3}]:[{4}]:{5}",
        DateTime.Now.ToString("yyyy-MM-dd HH:mm:ss.ffffff"), GetLotType(TL_Type),
        ModuleName, Page, Block, ExceptionMessage);

        WriteErrorFile(Error_Write_String, path);
    }
}

private void WriteErrorFile(string Error_Content, string path)
{
    string Logpath = ConfigurationSettings.AppSettings["pdfOutPut"].ToString();
    string filepath = ""; // (ConfigurationSettings.AppSettings["strlogpath"].ToString() +
(DateTime.Now.ToString("yyyy\\\\" + "MM\\\\" + "dd") + "\\"));

    if (Logpath == "" | Logpath.Length <= 0)
    {
        filepath = path + ("\\Log\\"); // + (DateTime.Now.ToString("yyyy\\\\" + "MM\\\\" + "dd") +
"\\");
    }
    else
    {
        filepath = (Logpath); // + (DateTime.Now.ToString("yyyy\\\\" + "MM\\\\" + "dd") + "
\\");
    }
    filepath = System.IO.Path.GetFileName(filepath);
    if (!Directory.Exists(filepath))
    {
        Directory.CreateDirectory(filepath);
    }
    string strErrorFileName = (DateTime.Now.Day.ToString("D2")) +
(DateTime.Now.Month.ToString("D2")) + "_WS_" + TraceThreadId + "." +
(ConfigurationSettings.AppSettings["strErrorFileName"]);
    strErrorFileName = filepath + "\\\" + strErrorFileName;
    StreamWriter sw = null;

    if ((!File.Exists(strErrorFileName)))
    {
        sw = File.CreateText(strErrorFileName);
    }
    else
    {

```

```

        sw = File.AppendText(strErrorFileName);
    }
    sw.WriteLine(Error_Content);
    sw.Close();
}
private string GetLotType(int TL_Type)
{
    string TR_Num = null;

    if ((TL_Type == 0))
    {
        TR_Num = "INFO";
    }
    else if ((TL_Type == 1))
    {
        TR_Num = "DEBUG";
    }
    else if ((TL_Type == 2))
    {
        TR_Num = "DATABASE";
    }
    else if ((TL_Type == 3))
    {
        TR_Num = "DETAILED";
    }
    else if ((TL_Type == 4))
    {
        TR_Num = "CRITICAL";
    }
    else if ((TL_Type == 5))
    {
        TR_Num = "EXCEPTION";
    }
    else
    {
        TR_Num = "ERROR";
    }
    return TR_Num;
}
}
}
*****AuditTrail.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="AuditTrail.aspx.cs" Inherits="AuditTrail" %>

<%@ Register Assembly="Infragistics4.Web.v14.2, Version=14.2.20142.1028,
Culture=neutral, PublicKeyToken=7dd5c3163f2cd0cb"
Namespace="Infragistics.Web.UI.GridControls" TagPrefix="ig" %>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="CPH_TopMessages" runat="Server">

```

```

<div class="container-fluid">
  <div class="row-fluid" style="padding-top: 20px;">
    <div class="span2" id="dvlogo" runat="server" style="display: none;">
      <%----%>
    </div>
    <div class="span10">
      <div class="row">
        <div id="dvMsgBox" class="span12 ordersave" runat="server" style="display:
none;">
          <asp:Label ID="Label1" runat="server" Text="Label"></asp:Label>
        </div>
      </div>
    </div>
  </div>
</asp:Content>

<asp:Content ID="Content3" ContentPlaceHolderID="CPH_LeftSideBar" runat="Server">
  <div id="LeftSideBar" runat="server" class="span2">
    <asp:Table ID="Table1" runat="server">
    </asp:Table>
  </div>
</asp:Content>

<asp:Content ID="Content4" ContentPlaceHolderID="CPH_MainContent" runat="Server">
  <div class="span10" id="MainContent" runat="server">
    <div id="ShowPageHeader" class="row" style="display: block;" runat="server">
      <div id="PageName" class="span12 customer" runat="server">
      </div>
    </div>

    <div id="Div1" class="row" style="display: block;">
      <div id="Div2" class="span12 customertable">

        <ig:WebDataGrid ID="WDGrid_UserList" runat="server" CssClass="wdglist"
AutoGenerateColumns="False">
          <Columns>

            <%--<ig:BoundDataField DataFieldName="CreatedOn" Key="CreatedOn"
Width="100px" DataFormatString="{0:dd/MM/yyyy}" >
              <Header Text="Created On">
              </Header>
            </ig:BoundDataField>--%>

            <ig:BoundDataField DataFieldName="AuditDate" Key="AuditDate"
Width="100px" DataFormatString="{0:dd/MM/yyyy}">
              <Header Text="On Date">
              </Header>
            </ig:BoundDataField>
            <ig:BoundDataField DataFieldName="UserName" Key="UserName"
Width="100px">
              <Header Text="User Name">
              </Header>

```

```

        </ig:BoundDataField>

        <ig:BoundDataField DataFieldName="Category" Key="Category"
Width="100px">
            <Header Text="Type">
                </Header>
            </ig:BoundDataField>

            <ig:BoundDataField DataFieldName="ObjectName" Key="ObjectName"
Width="200px">
                <Header Text="Object Id">
                    </Header>
                </ig:BoundDataField>
                <ig:BoundDataField DataFieldName="DName" Key="DName">
                    <Header Text="Field">
                        </Header>
                    </ig:BoundDataField>
                    <ig:BoundDataField DataFieldName="OValue" Key="OValue">
                        <Header Text="Previous">
                            </Header>
                        </ig:BoundDataField>
                        <ig:BoundDataField DataFieldName="NValue" Key="NValue">
                            <Header Text="New">
                                </Header>
                            </ig:BoundDataField>

                            <%-- <ig:TemplateDataField Key="Id" Width="60px">
                                <ItemTemplate>
                                    <asp:HyperLink ID="Id" runat="server" CssClass="btn btn-small btn-
primary" Text="Open"
                                        NavigateUrl='<% #Eval("DealerId",
"~/Managedealer.aspx?pageid=165&PageType=Edit&dealerid={0}")%>'>
                                        </asp:HyperLink>
                                    </ItemTemplate>
                                    <Header Text="Open" />
                                </ig:TemplateDataField>--%>
                            </Columns>
                            <Behaviors>
                                <ig:Paging CurrentPageLinkCssClass="Current" PageLinkCssClass="Link"
PagerCssClass="Pager" PagerMode="NumericFirstLast" PageSize="15" QuickPages="3"
FirstPageText="First" LastPageText="Last" NextPageText="Next"
PreviousPageText="Previous" PagerAppearance="bottom">
                                    </ig:Paging>

                                <ig:Filtering>
                                    </ig:Filtering>
                                <ig:Sorting Enabled="true">
                                    </ig:Sorting>
                                </Behaviors>
                            </ig:WebDataGrid>

                        </div>
                    </div>
                </div>

```

</asp:Content>

*****AuditTrail.aspx.cs *****

```
using DowBAL;
using DowBEntity;
using System;
using System.Collections.Generic;
using System.Data;
using System.Linq;
using System.Text.RegularExpressions;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
```

```
public partial class AuditTrail : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr == null)
        {
            Response.Redirect("MLB_Login.aspx");
        }
        string groupid = Clsusr.GroupId;
        string pageid = "6";// Request.QueryString["pageid"];
        string Id = Request.QueryString["Id"];
        ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
Convert.ToInt32(Clsusr.GroupId));
        int frompage = Convert.ToInt32(Request.QueryString["frompage"]);
        BindDetails(Convert.ToInt32(frompage), Clsusr.UserId, pageid, Id, groupid);
    }
    private void BindDetails(int frompage, string userid, string pageid, string Id, string groupid)
    {
        try
        {
            List<BalAuditTrail.EteAuditTrailXml> Lst;

            if (pageid == "157")
            {
                Lst = new BalAuditTrail().GetLogTrail(Convert.ToInt32(userid),
Convert.ToInt32(groupid));
            }
            else
            {
                Lst = new BalAuditTrail().GetAuditTrail(frompage, Convert.ToInt32(userid),
Convert.ToInt32(Id), Convert.ToInt32(groupid));
            }

            if (Lst != null && Lst.Count > 0)
            {
                WDGrid_UserList.DataSource = Lst;
            }
        }
    }
}
```

```

        WDGrid_UserList.DataBind();
        WDGrid_UserList.Visible = true;
    }
    else
    {
        Common.ShowMessageHtml("No Audit Trail available for this screen", 0, true,
dvMsgBox);

        WDGrid_UserList.Visible = false;
    }
}
catch(Exception exc)
{
}
}
private void ManageNavigations(int pageid, int userid, int groupid)
{
    bool IsLeftBar = false;
    IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
myBtnClickHandler);

    if (IsLeftBar == true)
    {
        LeftSideBar.Attributes["class"] = "span2";
        MainContent.Attributes["class"] = "span10";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;"); //style="display:
block;"
        PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
    else
    {
        LeftSideBar.Attributes["class"] = "";
        MainContent.Attributes["class"] = "span12";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
        PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
}

// This is Event Handler
protected void myBtnClickHandler(object sender, EventArgs e)
{
    LinkButton lnk = (LinkButton)sender;
    string ID = lnk.ID;

    DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
    if (ds != null)
    {
        if (ds.Tables[0].Rows.Count > 0)
        {
            DataTable dtdll = ds.Tables[0];

```

```

foreach (DataRow drnew in dtdll.Rows)
{
    switch (drnew["lbtncode"].ToString())
    {
        case "SAVE":
            // Call SAVE Method
            break;
        case "URL":
            Response.Redirect(drnew["pageurl"].ToString());
            break;
        case "AUDIT":
            Response.Redirect(drnew["pageurl"].ToString() + "&frompage=" +
Request.QueryString["pageid"]);
            break;

        case "CANCEL":
            // Call Cancel Method
            break;
        default:
            break;
    }
}
}
}
}
}
}
}
}
}
}
}

```

*****AdminHome.aspx *****

```

<%@ Page Title="" Language="C#" MasterPageFile="~/AdminMasterPage.master"
AutoEventWireup="true" CodeFile="AdminHome.aspx.cs" Inherits="AdminHome" %>

```

```

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
Runat="Server">
    <div class="span10">
        <div class="row">
            <%--<h3>Ravikiran Techno Soft</h3>--%>
            <div id="dvMsgBox" class="span10 ordersave" runat="server" style="display:
block;">
                <asp:Label ID="Label2" runat="server" Text="" Height="10px"></asp:Label>
            </div>
        </div>
    </div></asp:Content>
<asp:Content ID="Content4" ContentPlaceHolderID="CPH_TopMessages" Runat="Server">

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentMain" Runat="Server">
<!-- Main layout -->

    <div class="container-fluid">

```



```

<!-- Section: Intro -->
<section class="mt-md-4 pt-md-2 mb-5 pb-4">

  <!-- Grid row -->
  <div class="row">

    <!-- Grid column -->
    <div class="col-xl-3 col-md-6 mb-xl-0 mb-4">

      <!-- Card -->
      <div class="card card-cascade cascading-admin-card">

        <!-- Card Data -->
        <div class="admin-up">
          <i class="far fa-money-bill-alt primary-color mr-3 z-depth-2"></i>
          <div class="data">
            <p class="text-uppercase">sales</p>
            <h4 class="font-weight-bold dark-grey-text">2000$</h4>
          </div>
        </div>

        <!-- Card content -->
        <div class="card-body card-body-cascade">
          <div class="progress mb-3">
            <div class="progress-bar bg-primary" role="progressbar" style="width: 25%" aria-
            valuenow="25" aria-valuemin="0" aria-valuemax="100"></div>
          </div>
          <p class="card-text">Better than last week (25%)</p>
        </div>

      </div>
      <!-- Card -->

    </div>
    <!-- Grid column -->

    <!-- Grid column -->
    <div class="col-xl-3 col-md-6 mb-xl-0 mb-4">

      <!-- Card -->
      <div class="card card-cascade cascading-admin-card">

        <!-- Card Data -->
        <div class="admin-up">
          <i class="fas fa-chart-line warning-color mr-3 z-depth-2"></i>
          <div class="data">
            <p class="text-uppercase">subscriptions</p>
            <h4 class="font-weight-bold dark-grey-text">200</h4>
          </div>
        </div>

        <!-- Card content -->
        <div class="card-body card-body-cascade">
          <div class="progress mb-3">

```

```

        <div class="progress-bar red accent-2" role="progressbar" style="width: 25%"
aria-valuenow="25" aria-valuemin="0" aria-valuemax="100"></div>
    </div>
    <p class="card-text">Worse than last week (25%)</p>
</div>

<!-- Card -->

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-xl-3 col-md-6 mb-md-0 mb-4">

<!-- Card -->
<div class="card card-cascade cascading-admin-card">

    <!-- Card Data -->
    <div class="admin-up">
        <i class="fas fa-chart-pie light-blue lighten-1 mr-3 z-depth-2"></i>
        <div class="data">
            <p class="text-uppercase">traffic</p>
            <h4 class="font-weight-bold dark-grey-text">20000</h4>
        </div>
    </div>

    <!-- Card content -->
    <div class="card-body card-body-cascade">
        <div class="progress mb-3">
            <div class="progress-bar red accent-2" role="progressbar" style="width: 75%"
aria-valuenow="75" aria-valuemin="0" aria-valuemax="100"></div>
        </div>
        <p class="card-text">Worse than last week (75%)</p>
    </div>

</div>
<!-- Card -->

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-xl-3 col-md-6 mb-0">

<!-- Card -->
<div class="card card-cascade cascading-admin-card">

    <!-- Card Data -->
    <div class="admin-up">
        <i class="fas fa-chart-bar red accent-2 mr-3 z-depth-2"></i>
        <div class="data">
            <p class="text-uppercase">organic traffic</p>
            <h4 class="font-weight-bold dark-grey-text">2000</h4>
        </div>
    </div>

```

```

</div>

<!-- Card content -->
<div class="card-body card-body-cascade">
  <div class="progress mb-3">
    <div class="progress-bar bg-primary" role="progressbar" style="width: 25%" aria-
valuenow="25" aria-valuemin="0" aria-valuemax="100"></div>
  </div>
  <p class="card-text">Better than last week (25%)</p>
</div>

</div>
<!-- Card -->

</div>
<!-- Grid column -->

</div>
<!-- Grid row -->

</section>
<!-- Section: Intro -->

<!-- Section: Main panel -->
<section class="mb-5">

  <!-- Card -->
  <div class="card card-cascade narrower">

    <!-- Section: Chart -->
    <section>

      <!-- Grid row -->
      <div class="row">

        <!-- Grid column -->
        <div class="col-xl-5 col-lg-12 mr-0 pb-2">

          <!-- Card image -->
          <div class="view view-cascade gradient-card-header light-blue lighten-1">
            <h2 class="h2-responsive mb-0 font-weight-500">Purchased</h2>
          </div>

          <!-- Card content -->
          <div class="card-body card-body-cascade pb-0">

            <!-- Panel data -->
            <div class="row py-3 pl-4">

              <!-- First column -->
              <div class="col-md-6">

                <!-- Date select -->
                <p class="lead"><span class="badge info-color p-2">Data range</span></p>
                <select class="mdb-select colorful-select dropdown-info md-form">

```

```

        <option value="" disabled selected>Choose time period</option>
        <option value="1">Today</option>
        <option value="2">Yesterday</option>
        <option value="3">Last 7 days</option>
        <option value="3">Last 30 days</option>
        <option value="3">Last week</option>
        <option value="3">Last month</option>
    </select>

    <!-- Date pickers -->
    <p class="lead pt-3 pb-4"><span class="badge info-color p-2">Custom
date</span></p>
    <div class="md-form">
    <input placeholder="Selected date" type="text" id="from" class="form-control
datepicker">
    <label for="date-picker-example">From</label>
    </div>
    <div class="md-form">
    <input placeholder="Selected date" type="text" id="to" class="form-control
datepicker">
    <label for="date-picker-example">To</label>
    </div>

</div>
<!-- First column -->

<!-- Second column -->
<div class="col-md-6 text-center pl-xl-2 my-md-0 my-3">

    <!-- Summary -->
    <p>Total sales: <strong>2000$</strong>
    <button type="button" class="btn btn-info btn-sm p-2" data-toggle="tooltip"
data-placement="top"
    title="Total sales in the given period"><i class="fas fa-
question"></i></button>
    </p>
    <p>Average sales: <strong>100$</strong>
    <button type="button" class="btn btn-info btn-sm p-2 mr-0" data-
toggle="tooltip" data-placement="top"
    title="Average daily sales in the given period"><i class="fas fa-
question"></i></button>
    </p>

    <!-- Change chart -->
    <span class="min-chart my-4" id="chart-sales" data-percent="76"><span
class="percent"></span></span>
    <h5>
    <span class="badge red accent-2 p-2">Change <i class="fas fa-arrow-circle-
up ml-1"></i></span>
    <button type="button" class="btn btn-info btn-sm p-2" data-toggle="tooltip"
data-placement="top"
    title="Percentage change compared to the same period in the past"><i
class="fas fa-question"></i>
    </button>
    </h5>

```

```

        </div>
        <!-- Second column -->

    </div>
    <!-- Panel data -->

</div>
<!-- Card content -->

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-xl-7 col-lg-12 mb-4 pb-2">

    <!-- Chart -->
    <div class="view view-cascade gradient-card-header blue-gradient">

        <canvas id="lineChart" height="175"></canvas>

    </div>

</div>
<!-- Grid column -->

</div>
<!-- Grid row -->

</section>
<!-- Section: Chart -->

<!-- Section: Table -->
<section>

    <!-- Top Table UI -->
    <div class="table-ui p-2 mb-3 mx-4 mb-5">

        <!-- Grid row -->
        <div class="row">

            <!-- Grid column -->
            <div class="col-xl-3 col-md-6">

                <!-- Name -->
                <select class="mdb-select colorful-select dropdown-info mx-2 md-form mt-3 md-
dropdown">
                    <option value="" disabled selected>Bulk actions</option>
                    <option value="1">Delete</option>
                    <option value="2">Export</option>
                    <option value="3">Change segment</option>
                </select>

            </div>
        </div>
        <!-- Grid column -->

```

```

<!-- Grid column -->
<div class="col-xl-3 col-md-6">

  <!-- Blue select -->
  <select class="mdb-select colorful-select dropdown-info mx-2 md-form mt-3 md-
dropdown">
    <option value="" disabled selected>Show only</option>
    <option value="1">All <span> (2000)</span></option>
    <option value="2">Never opened <span> (200)</span></option>
    <option value="3">Opened but unanswered <span> (1800)</span></option>
    <option value="4">Answered <span> (200)</span></option>
    <option value="5">Unsubscribed <span> (50)</span></option>
  </select>

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-xl-3 col-md-6">

  <!-- Blue select -->
  <select class="mdb-select colorful-select dropdown-info mx-2 md-form mt-3 md-
dropdown">
    <option value="" disabled selected>Filter segments</option>
    <option value="1">Contacts in no segments <span> (100)</span></option>
    <option value="1">Segment 1 <span> (2000)</span></option>
    <option value="2">Segment 2 <span> (1000)</span></option>
    <option value="3">Segment 3 <span> (4000)</span></option>
  </select>

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-xl-3 col-md-6">

  <div class="form-inline mt-2 ml-2">
    <div class="form-group md-form mt-2">
      <input class="form-control w-100" type="text" placeholder="Search">
    </div>
    <button class="btn btn-sm btn-primary ml-2 mr-0 mb-md-0 mb-4 px-2"><i
class="fas fa-search"></i></button>
  </div>

</div>
<!-- Grid column -->

</div>
<!-- Grid row -->

</div>
<!-- Top Table UI -->

</section>

```

```

        <!--Section: Table-->

        </div>
        <!-- Card -->

    </section>
    <!-- Section: Main panel -->

</div>

    <!-- Main layout --></asp:Content>
<asp:Content ID="Content3" ContentPlaceHolderID="ContentPlaceHolderfooter"
Runat="Server">
</asp:Content>

***** AdminHome.aspx.cs *****
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBEntity;
using DowBAL;

public partial class AdminHome : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr != null)
        {
            string msg = Request.QueryString["msg"];
            if (msg != null)
            {
                Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
            }
        }
        else
        {
            Response.Redirect("Login.aspx");
        }
    }
}

***** AdminMasterPage.master *****
<%@ Master Language="C#" AutoEventWireup="true"
CodeFile="AdminMasterPage.master.cs" Inherits="AdminMasterPage" %>
<!DOCTYPE html>
<html lang="en">

```

```

<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
  <meta http-equiv="x-ua-compatible" content="ie=edge">
  <title>MLB | Vendor Management</title>
  <!-- Favicon -->
  <link rel="icon" type="image/gif" href="img/favicon.png">
  <!-- Font Awesome -->
  <link rel="stylesheet" href="Vendor/css/all.css">
  <!-- Bootstrap core CSS -->
  <link rel="stylesheet" href="Vendor/css/bootstrap.min.css">
  <!-- Material Design Bootstrap -->
  <link rel="stylesheet" href="Vendor/css/mdb.min.css">
  <!-- Your custom styles (optional) -->
  <link rel="stylesheet" type="text/css" href="Vendor/css/style.css">
  <style>

  </style>
</head>

<body class="fixed-sn white-skin">

  <!-- Main Navigation -->
  <header>

    <!-- Sidebar navigation -->
    <div id="slide-out" class="side-nav sn-bg-4 fixed">
      <ul class="custom-scrollbar">

        <!-- Logo -->
        <li class="logo-sn waves-effect py-3">
          <div class="text-center">
            <a href="#" class="pl-0"></a>
          </div>
        </li>

        <!-- Search Form -->
        <li>
          <form class="search-form" role="search">
            <div class="md-form mt-0 waves-light">
              <input type="text" class="form-control py-2" placeholder="Search">
            </div>
          </form>
        </li>

        <!-- Side navigation links -->
        <li>
          <ul class="collapsible collapsible-accordion">
            <li class="active">
              <a href="AdminHome.aspx" class="collapsible-header waves-effect"><i class="w-fa
fas fa-tachometer-alt"></i>Dashboard</a>
            </li>
            <li>
              <a class="collapsible-header waves-effect arrow-r">

```



```
        <i class="w-fa fas fa-shopping-cart"></i>Shopping Vendor<i class="fas fa-angle-
down rotate-icon"></i>
    </a>
    <div class="collapsible-body">
        <ul>
            <li>
                <a href="vendorproductsAddUser.aspx" class="waves-effect">Select
Categories </a>
            </li>
            <li>
                <a href="vendorcolorsAddUser.aspx" class="waves-effect">Select Colors </a>
            </li>
            <li>
                <a href="vendorsizesAddUser.aspx" class="waves-effect">Select Sizes </a>
            </li>
            <li>
                <a href="vendorproductsUser.aspx" class="waves-effect">Products
Management</a>
            </li>
            <li>
                <a href="PostProductsUser.aspx" class="waves-effect">Post Products </a>
            </li>
            <li>
                <a href="ManageOrderUser.aspx" class="waves-effect">Order
Management</a>
            </li>
            <li>
                <a href="ManagePaymentsUser.aspx" class="waves-effect">Payments</a>
            </li>
            <li>
                <a href="ManageLogisticsUser.aspx" class="waves-effect">Logistic</a>
            </li>
            <li>
                <a href="ManageOrderRetUser.aspx" class="waves-effect">Order returned</a>
            </li>
            <li>
                <a href="ManageCommunicationsUser.aspx" class="waves-
effect">Communication</a>
            </li>
        </ul>
    </div>
</li>
<li>
    <a class="collapsible-header waves-effect arrow-r">
        <i class="w-fa fas fa-user-cog"></i>Service Vendor<i class="fas fa-angle-down
rotate-icon"></i>
    </a>
    <div class="collapsible-body">
        <ul>
            <li>
                <a href="VendorServicesAddUser.aspx" class="waves-effect">Select Service
Types </a>
            </li>
            <li>
                <a href="vendorServicesUser.aspx" class="waves-effect">Services
Management</a>
            </li>
        </ul>
    </div>
</li>
</ul>
</div>
```

```

        </li>
        <li>
            <a href="PostServicesUser.aspx" class="waves-effect">Add Services </a>
        </li>
        <li>
            <a href="ManageOrderUser.aspx" class="waves-effect">Order
Management</a>
        </li>
        <li>
            <a href="ManagePaymentsUser.aspx" class="waves-effect">Payments</a>
        </li>
        <li>
            <a href="ManageOrderRetUser.aspx" class="waves-effect">Order
Cancellations</a>
        </li>
        <li>
            <a href="ManageCommunicationsUser.aspx" class="waves-
effect">Communication</a>
        </li>
    </ul>
</div>
</li>
<li>
    <a class="collapsible-header waves-effect arrow-r">
        <i class="w-fa fas fa-tasks"></i>Business Listing Vendor<i class="fas fa-angle-
down rotate-icon"></i>
    </a>
    <div class="collapsible-body">
        <ul>
            <li>
                <a href="#" class="waves-effect">Add the Details Business</a>
            </li>
            <li>
                <a href="#" class="waves-effect">Product Catalog</a>
            </li>
            <li>
                <a href="#" class="waves-effect">Contact Detail</a>
            </li>
        </ul>
    </div>
</li>
</ul>
</li>
<!-- Side navigation links -->

</ul>
<div class="sidenav-bg mask-strong"></div>
</div>
<!-- Sidebar navigation -->

<!-- Navbar -->
<nav class="navbar fixed-top navbar-expand-lg scrolling-navbar double-nav">

<!-- SideNav slide-out button -->
<div class="float-left">

```

```

        <a href="#" data-activates="slide-out" class="button-collapse"><i class="fas fa-bars"></i></a>
    </div>

    <!-- Breadcrumb -->
    <div class="breadcrumb-dn mr-auto">
        <p>Dashboard</p>
    </div>

    <div class="d-flex change-mode">

        <div class="ml-auto mb-0 mr-3 change-mode-wrapper">
            <button class="btn btn-outline-black btn-sm" id="dark-mode">Change Mode</button>
        </div>

    <!-- Navbar links -->
    <ul class="nav navbar-nav nav-flex-icons ml-auto">

        <!-- Dropdown -->
        <li class="nav-item dropdown notifications-nav">
            <a class="nav-link dropdown-toggle waves-effect" id="navbarDropdownMenuLink"
data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
                <span class="badge red">3</span> <i class="fas fa-bell"></i>
                <span class="d-none d-md-inline-block">Notifications</span>
            </a>
            <div class="dropdown-menu dropdown-primary" aria-
labelledby="navbarDropdownMenuLink">
                <a class="dropdown-item" href="#">
                    <i class="far fa-money-bill-alt mr-2" aria-hidden="true"></i>
                    <span>New order received</span>
                    <span class="float-right"><i class="far fa-clock" aria-hidden="true"></i> 13
min</span>
                </a>
                <a class="dropdown-item" href="#">
                    <i class="far fa-money-bill-alt mr-2" aria-hidden="true"></i>
                    <span>New order received</span>
                    <span class="float-right"><i class="far fa-clock" aria-hidden="true"></i> 33
min</span>
                </a>
                <a class="dropdown-item" href="#">
                    <i class="fas fa-chart-line mr-2" aria-hidden="true"></i>
                    <span>Your campaign is about to end</span>
                    <span class="float-right"><i class="far fa-clock" aria-hidden="true"></i> 53
min</span>
                </a>
            </div>
        </li>
        <li class="nav-item">
            <a class="nav-link waves-effect"><i class="fas fa-envelope"></i> <span
class="clearfix d-none d-sm-inline-block">Contact</span></a>
        </li>
        <li class="nav-item">
            <a class="nav-link waves-effect"><i class="far fa-comments"></i> <span
class="clearfix d-none d-sm-inline-block">Support</span></a>
        </li>

```

```

        <li class="nav-item dropdown">
            <a class="nav-link dropdown-toggle waves-effect" href="#" id="userDropdown" data-
toggle="dropdown"aria-haspopup="true" aria-expanded="false">
                <i class="fas fa-user"></i> <span class="clearfix d-none d-sm-inline-
block">Profile</span>
            </a>
            <div class="dropdown-menu dropdown-menu-right" aria-
labelledby="userDropdown">
                <a class="dropdown-item" href="#">Log Out</a>
                <a class="dropdown-item" href="#">My account</a>
            </div>
        </li>

    </ul>
    <!-- Navbar links -->

</div>

</nav>
<!-- Navbar -->

<!-- Fixed button -->
<div class="fixed-action-btn clearfix d-none d-xl-block" style="bottom: 45px; right: 24px;">

    <a class="btn-floating btn-lg red">
        <i class="fas fa-pencil-alt"></i>
    </a>

    <ul class="list-unstyled">
        <li><a class="btn-floating red"><i class="fas fa-star"></i></a></li>
        <li><a class="btn-floating yellow darken-1"><i class="fas fa-user"></i></a></li>
        <li><a class="btn-floating green"><i class="fas fa-envelope"></i></a></li>
        <li><a class="btn-floating blue"><i class="fas fa-shopping-cart"></i></a></li>
    </ul>

</div>
<!-- Fixed button -->

</header>
<!-- Main Navigation -->

<!-- Main layout -->

<div class="container-fluid">

    <!-- Section: Intro -->
    <section class="mt-md-4 pt-md-2 mb-5 pb-4">

        <!-- Grid row -->
        <div class="row">

            <!-- Grid column -->
            <div class="col-xl-3 col-md-6 mb-xl-0 mb-4">

```

```

<!-- Card -->
<div class="card card-cascade cascading-admin-card">

  <!-- Card Data -->
  <div class="admin-up">
    <i class="far fa-money-bill-alt primary-color mr-3 z-depth-2"></i>
    <div class="data">
      <p class="text-uppercase">sales</p>
      <h4 class="font-weight-bold dark-grey-text">2000$</h4>
    </div>
  </div>

  <!-- Card content -->
  <div class="card-body card-body-cascade">
    <div class="progress mb-3">
      <div class="progress-bar bg-primary" role="progressbar" style="width: 25%" aria-
valuenow="25" aria-valuemin="0" aria-valuemax="100"></div>
    </div>
    <p class="card-text">Better than last week (25%)</p>
  </div>

</div>
<!-- Card -->

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-xl-3 col-md-6 mb-xl-0 mb-4">

  <!-- Card -->
  <div class="card card-cascade cascading-admin-card">

    <!-- Card Data -->
    <div class="admin-up">
      <i class="fas fa-chart-line warning-color mr-3 z-depth-2"></i>
      <div class="data">
        <p class="text-uppercase">subscriptions</p>
        <h4 class="font-weight-bold dark-grey-text">200</h4>
      </div>
    </div>

    <!-- Card content -->
    <div class="card-body card-body-cascade">
      <div class="progress mb-3">
        <div class="progress-bar red accent-2" role="progressbar" style="width: 25%"
aria-valuenow="25" aria-valuemin="0" aria-valuemax="100"></div>
      </div>
      <p class="card-text">Worse than last week (25%)</p>
    </div>

  </div>
  <!-- Card -->

</div>

```

```

<!-- Grid column -->

<!-- Grid column -->
<div class="col-xl-3 col-md-6 mb-md-0 mb-4">

  <!-- Card -->
  <div class="card card-cascade cascading-admin-card">

    <!-- Card Data -->
    <div class="admin-up">
      <i class="fas fa-chart-pie light-blue lighten-1 mr-3 z-depth-2"></i>
      <div class="data">
        <p class="text-uppercase">traffic</p>
        <h4 class="font-weight-bold dark-grey-text">20000</h4>
      </div>
    </div>

    <!-- Card content -->
    <div class="card-body card-body-cascade">
      <div class="progress mb-3">
        <div class="progress-bar red accent-2" role="progressbar" style="width: 75%"
aria-valuenow="75" aria-valuemin="0" aria-valuemax="100"></div>
      </div>
      <p class="card-text">Worse than last week (75%)</p>
    </div>

  </div>
<!-- Card -->

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-xl-3 col-md-6 mb-0">

  <!-- Card -->
  <div class="card card-cascade cascading-admin-card">

    <!-- Card Data -->
    <div class="admin-up">
      <i class="fas fa-chart-bar red accent-2 mr-3 z-depth-2"></i>
      <div class="data">
        <p class="text-uppercase">organic traffic</p>
        <h4 class="font-weight-bold dark-grey-text">2000</h4>
      </div>
    </div>

    <!-- Card content -->
    <div class="card-body card-body-cascade">
      <div class="progress mb-3">
        <div class="progress-bar bg-primary" role="progressbar" style="width: 25%"
aria-valuenow="25" aria-valuemin="0" aria-valuemax="100"></div>
      </div>
      <p class="card-text">Better than last week (25%)</p>
    </div>
  </div>
</div>

```

```

</div>
<!-- Card -->

</div>
<!-- Grid column -->

</div>
<!-- Grid row -->

</section>
<!-- Section: Intro -->

<!-- Section: Main panel -->
<section class="mb-5">

  <!-- Card -->
  <div class="card card-cascade narrower">

    <!-- Section: Chart -->
    <section>

      <!-- Grid row -->
      <div class="row">

        <!-- Grid column -->
        <div class="col-xl-5 col-lg-12 mr-0 pb-2">

          <!-- Card image -->
          <div class="view view-cascade gradient-card-header light-blue lighten-1">
            <h2 class="h2-responsive mb-0 font-weight-500">Purchased</h2>
          </div>

          <!-- Card content -->
          <div class="card-body card-body-cascade pb-0">

            <!-- Panel data -->
            <div class="row py-3 pl-4">

              <!-- First column -->
              <div class="col-md-6">

                <!-- Date select -->
                <p class="lead"><span class="badge info-color p-2">Data range</span></p>
                <select class="mdb-select colorful-select dropdown-info md-form">
                  <option value="" disabled selected>Choose time period</option>
                  <option value="1">Today</option>
                  <option value="2">Yesterday</option>
                  <option value="3">Last 7 days</option>
                  <option value="3">Last 30 days</option>
                  <option value="3">Last week</option>
                  <option value="3">Last month</option>
                </select>

                <!-- Date pickers -->

```

```

        <p class="lead pt-3 pb-4"><span class="badge info-color p-2">Custom
date</span></p>
        <div class="md-form">
            <input placeholder="Selected date" type="text" id="from" class="form-control
datepicker">
            <label for="date-picker-example">From</label>
        </div>
        <div class="md-form">
            <input placeholder="Selected date" type="text" id="to" class="form-control
datepicker">
            <label for="date-picker-example">To</label>
        </div>

    </div>
    <!-- First column -->

    <!-- Second column -->
    <div class="col-md-6 text-center pl-xl-2 my-md-0 my-3">

        <!-- Summary -->
        <p>Total sales: <strong>2000$</strong>
            <button type="button" class="btn btn-info btn-sm p-2" data-toggle="tooltip"
data-placement="top"
            title="Total sales in the given period"><i class="fas fa-
question"></i></button>
        </p>
        <p>Average sales: <strong>100$</strong>
            <button type="button" class="btn btn-info btn-sm p-2 mr-0" data-
toggle="tooltip" data-placement="top"
            title="Average daily sales in the given period"><i class="fas fa-
question"></i></button>
        </p>

        <!-- Change chart -->
        <span class="min-chart my-4" id="chart-sales" data-percent="76"><span
class="percent"></span></span>
        <h5>
            <span class="badge red accent-2 p-2">Change <i class="fas fa-arrow-circle-
up ml-1"></i></span>
            <button type="button" class="btn btn-info btn-sm p-2" data-toggle="tooltip"
data-placement="top"
            title="Percentage change compared to the same period in the past"><i
class="fas fa-question"></i>
            </button>
        </h5>

    </div>
    <!-- Second column -->

</div>
<!-- Panel data -->

</div>
<!-- Card content -->

```



```

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-xl-7 col-lg-12 mb-4 pb-2">

  <!-- Chart -->
  <div class="view view-cascade gradient-card-header blue-gradient">

    <canvas id="lineChart" height="175"></canvas>

  </div>

</div>
<!-- Grid column -->

</div>
<!-- Grid row -->

</section>
<!-- Section: Chart -->

<!-- Section: Table -->
<section>

  <!-- Top Table UI -->
  <div class="table-ui p-2 mb-3 mx-4 mb-5">

    <!-- Grid row -->
    <div class="row">

      <!-- Grid column -->
      <div class="col-xl-3 col-md-6">

        <!-- Name -->
        <select class="mdb-select colorful-select dropdown-info mx-2 md-form mt-3 md-
dropdown">
          <option value="" disabled selected>Bulk actions</option>
          <option value="1">Delate</option>
          <option value="2">Export</option>
          <option value="3">Change segment</option>
        </select>

      </div>
    <!-- Grid column -->

    <!-- Grid column -->
    <div class="col-xl-3 col-md-6">

      <!-- Blue select -->
      <select class="mdb-select colorful-select dropdown-info mx-2 md-form mt-3 md-
dropdown">
        <option value="" disabled selected>Show only</option>
        <option value="1">All <span> (2000)</span></option>
        <option value="2">Never opened <span> (200)</span></option>
      </select>
    </div>
  </div>
</div>

```

```

    <option value="3">Opened but unanswered <span> (1800)</span></option>
    <option value="4">Answered <span> (200)</span></option>
    <option value="5">Unsubscribed <span> (50)</span></option>
  </select>

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-xl-3 col-md-6">

  <!-- Blue select -->
  <select class="mdb-select colorful-select dropdown-info mx-2 md-form mt-3 md-
dropdown">
    <option value="" disabled selected>Filter segments</option>
    <option value="1">Contacts in no segments <span> (100)</span></option>
    <option value="1">Segment 1 <span> (2000)</span></option>
    <option value="2">Segment 2 <span> (1000)</span></option>
    <option value="3">Segment 3 <span> (4000)</span></option>
  </select>

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-xl-3 col-md-6">

  <form class="form-inline mt-2 ml-2">
    <div class="form-group md-form mt-2">
      <input class="form-control w-100" type="text" placeholder="Search">
    </div>
    <button class="btn btn-sm btn-primary ml-2 mr-0 mb-md-0 mb-4 px-2"><i
class="fas fa-search"></i></button>
  </form>

</div>
<!-- Grid column -->

</div>
<!-- Grid row -->

</div>
<!-- Top Table UI -->

</section>
<!-- Section: Table-->

</div>
<!-- Card -->

</section>
<!-- Section: Main panel -->

</div>

```

```

<!-- Main layout -->

<!-- Footer -->
    <asp:ContentPlaceHolder ID="CPH_TopMessages" runat="server">

        </asp:ContentPlaceHolder>
<asp:ContentPlaceHolder ID="CPH_MainMaster_Header" runat="server">
    </asp:ContentPlaceHolder>

<!-- Main layout -->

<!-- Main layout -->

<!-- Footer -->
    <asp:ContentPlaceHolder ID="ContentMain" runat="server">
    </asp:ContentPlaceHolder>
    <asp:ContentPlaceHolder ID="ContentPlaceHolderfooter" runat="server">

<footer class="page-footer pt-0 mt-5 rgba-stylish-light">

    <!-- Copyright -->
    <div class="footer-copyright py-3 text-center">
        <div class="container-fluid">
            © 2022 Copyright: <a href="#" target="_blank"> Shopping@dateorwed.com </a>
        </div>
    </div>

</footer>
<!-- Footer -->
    </asp:ContentPlaceHolder>

<!-- SCRIPTS -->
<!-- JQuery -->
<script src="Vendor/js/jquery-3.3.1.min.js"></script>
<!-- Bootstrap tooltips -->
<script type="text/javascript" src="Vendor/js/popper.min.js"></script>
<!-- Bootstrap core JavaScript -->
<script type="text/javascript" src="Vendor/js/bootstrap.js"></script>
<!-- MDB core JavaScript -->
<script type="text/javascript" src="Vendor/js/mdb.min.js"></script>

<!-- Initializations -->
<script>

    // SideNav Initialization
    $(".button-collapse").sideNav();

    var container = document.querySelector('.custom-scrollbar');
    Ps.initialize(container, {
        wheelSpeed: 2,
        wheelPropagation: true,
        minScrollbarLength: 20
    });

```

```

// Data Picker Initialization
$('.datepicker').pickadate();

// Material Select Initialization
$(document).ready(function () {
    $('.mdb-select').material_select();
});

// Tooltips Initialization
$(function () {
    $('[data-toggle="tooltip"]').tooltip()
})

</script>

<!-- Charts -->
<script>

// Small chart
$(function () {
    $('.min-chart#chart-sales').easyPieChart({
        barColor: "#FF5252",
        onStep: function (from, to, percent) {
            $(this.el).find('.percent').text(Math.round(percent));
        }
    });
});

// Main chart
var ctxL = document.getElementById("lineChart").getContext('2d');
var myLineChart = new Chart(ctxL, {
    type: 'line',
    data: {
        labels: ["January", "February", "March", "April", "May", "June", "July"],
        datasets: [{
            label: "My First dataset",
            fillColor: "#fff",
            backgroundColor: 'rgba(255, 255, 255, .3)',
            borderColor: 'rgba(255, 255, 255)',
            data: [0, 10, 5, 2, 20, 30, 45],
        }]
    },
    options: {
        legend: {
            labels: {
                fontFamily: "#fff",
            }
        },
        scales: {
            xAxes: [{
                gridLines: {
                    display: true,
                    color: "rgba(255,255,255,.25)"
                },
                ticks: {

```

```

        fontColor: "#fff",
    },
}],
yAxes: [{
    display: true,
    gridLines: {
        display: true,
        color: "rgba(255,255,255,.25)"
    },
    ticks: {
        fontColor: "#fff",
    },
}],
}
});

$(function () {
    $('#dark-mode').on('click', function (e) {

        e.preventDefault();

        $('footer, .card').toggleClass('dark-card-admin');
        $('body, .navbar').toggleClass('white-skin navy-blue-skin');
        $(this).toggleClass('white text-dark btn-outline-black');
        $('body').toggleClass('dark-bg-admin');
        $('h6, .card, p, td, th, i, li a, h4, input, label').not('#slide-out i, #slide-out a,
.dropdown-item i, .dropdown-item').toggleClass('text-white');
        $('.btn-dash').toggleClass('grey blue').toggleClass('lighten-3 darken-3');
        $('.gradient-card-header').toggleClass('white black lighten-4');
        $('.list-panel a').toggleClass('navy-blue-bg-a text-white').toggleClass('list-group-
border');

    });
});

</script>

</body>
</html>

```

```

***** AdminMasterPage.master.cs *****
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBEntity;
using DowBAL;
using System.Text;
using DowDAL;
using System.Data;
using System.Web.UI.WebControls;
using System.Web.Security;

```

```

public partial class AdminMasterPage : System.Web.UI.MasterPage
{
    string UserName;
    protected void Page_Load(object sender, EventArgs e)
    {
        Session["Pid"] = "1";
        Session["Pgid"] = "0";
        StringBuilder sb = new StringBuilder();
        //menu.InnerHtml = Common.Generate_newmenu();
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        string user;
        if (Clsusr != null)
        {
            UserName = Clsusr.UserName;
        }
        else
        {
            HttpCookie myCookie = Request.Cookies["DateOrWedCookie"];
            if (myCookie != null)
            {
                user = myCookie.Value.ToString();
            }
            else
            {
                user = "Guest";
            }
            guest_login(user);
        }
    }
}

protected void guest_login(string user)
{
    string City = "Hyderabad";// _City;
    string Country = "India";// _Country;

    BALCreateUser objlogin = new BALCreateUser();
    DataTable status_login = objlogin.LoginGuest(user, user,
Request.ServerVariables["REMOTE_ADDR"].ToString(), "", City, Country);
    string login_status = status_login.Rows[0]["UserStatus"].ToString();

    ClsLoggedInUser objLogedInUser;
    if (login_status == "True")
    {
        objLogedInUser = new ClsLoggedInUser();

        objLogedInUser.UserId = status_login.Rows[0]["UserId"].ToString();
        objLogedInUser.UserName = status_login.Rows[0]["UserName"].ToString();
        objLogedInUser.GroupId = status_login.Rows[0]["GroupId"].ToString();
        objLogedInUser.LoginStatus = "True";
    }
}

```

```
objLoggedInUser.LoginID = Convert.ToInt32(status_login.Rows[0]["LoginId"]);
objLoggedInUser.DealerId = Convert.ToInt32(status_login.Rows[0]["DealerId"]);
objLoggedInUser.RoleId = Convert.ToInt32(status_login.Rows[0]["RoleId"]);
objLoggedInUser.BrandId = 1; //Convert.ToInt32(status_login.Rows[0]["BrandId"]);
```

```
Session["LoggedUser_Specs"] = objLoggedInUser;
```

```
FormsAuthentication.SetAuthCookie(objLoggedInUser.UserName, false);
HttpCookie myCookie = new HttpCookie("DateOrWedCookie");
DateTime now = DateTime.Now;
myCookie.Value = objLoggedInUser.UserName;
myCookie.Expires = now.AddYears(1);
Response.Cookies.Add(myCookie);
```

```
    }
}
}
```

```
*****BSOffers.aspx *****
```

```
<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="BSOffers.aspx.cs" Inherits="BSOffers" %>
```

```
<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
</asp:Content>
```

```
<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
<div class="container-fluid">
<div class="row-fluid" style="padding-top: 20px;">
<div class="span2" id="dvlogo" runat="server" style="display: block;">
<!----%>
</div>
<div class="span10">
<div class="row">
<div id="dvMsgBox" class="span12 ordersave" runat="server" style="display:
none;">
<asp:Label ID="Label1" runat="server" Text=""></asp:Label>
</div>
</div>
</div>
</div>
</div>
</asp:Content>
```

```
<asp:Content ID="Content4" ContentPlaceHolderID="CPH_LeftSideBar" runat="Server">
<div id="LeftSideBar" runat="server" class="span2">
<asp:Table ID="Table1" runat="server">
</asp:Table>
</div>
</asp:Content>
```

```

<asp:Content ID="Content2" ContentPlaceHolderID="CPH_MainContent" runat="Server">

    <div class="span10" id="MainContent" runat="server">
        <div id="ShowPageHeader" class="row" style="display: block;" runat="server">
            <div id="PageName" class="span12 customer" runat="server">
                </div>
            </div>

            <div id="Div1" class="row" style="display: block;">
                <div id="Div2" class="span12 customertable">

                    <!-- Main Content Comes here-->

                    <%--<asp:ScriptManager ID="ScriptManager1"
runat="server"></asp:ScriptManager>--%>

                    <%--                <ig:WebDataGrid ID="WDGrid_UserList" runat="server" CssClass="wdglist"
AutoGenerateColumns="False">
                        <Columns>

                                <ig:BoundDataField DataFieldName="ProductName" Key="ProductName" >
                                    <Header Text="Product Name">
                                        </Header>
                                    </ig:BoundDataField>

                                <ig:BoundDataField DataFieldName="SellerDescription"
Key="SellerDescription ">
                                    <Header Text="Description">
                                        </Header>
                                    </ig:BoundDataField>
                                <ig:BoundDataField DataFieldName="Price" Key="Price " Width="60px">
                                    <Header Text="Price">
                                        </Header>
                                    </ig:BoundDataField>

                                <ig:BoundDataField DataFieldName="ShCurrency" Key="ShCurrency"
Width="60px">
                                    <Header Text="Currency">
                                        </Header>
                                    </ig:BoundDataField>

                                <ig:BoundDataField DataFieldName="Qty" Key="Qty" Width="60px">
                                    <Header Text="Qty">
                                        </Header>
                                    </ig:BoundDataField>
                                <ig:BoundDataField DataFieldName="Amt" Key="Amt" Width="60px">
                                    <Header Text="Amount">
                                        </Header>
                                    </ig:BoundDataField>

                                <ig:TemplateDataField Key="UCId" Width="80px">

```



```

        <ItemTemplate>
            <asp:HyperLink ID="UCId" runat="server" CssClass="btn btn-small
btn-primary" Text="More info"
                NavigateUrl='<% #Eval("UCId",
"~/ManageProductsource.aspx?pageid=28&PageType=Buy&source=Home&PSId={0}")%>'>
            </asp:HyperLink>
            <asp:LinkButton ID="LinkButton1" runat="server" OnClick="LinkButton1_Click"
                CssClass="btn btn-danger" CommandName='<% #Eval("UCId", "{0}")%>'
                >Delete</asp:LinkButton>

        </ItemTemplate>
        <Header Text="Open" />
    </ig:TemplateDataField>
</Columns>
<Behaviors>
    <ig:Paging CurrentPageLinkCssClass="Current" PageLinkCssClass="Link"
        PagerCssClass="Pager" PagerMode="NumericFirstLast" PageSize="15" QuickPages="3"
        FirstPageText="First" LastPageText="Last" NextPageText="Next"
        PreviousPageText="Previous" PagerAppearance="bottom">
    </ig:Paging>

    <ig:Filtering>
    </ig:Filtering>
    <ig:Sorting Enabled="true">
    </ig:Sorting>
</Behaviors>
</ig:WebDataGrid>
    <asp:SqlDataSource ID="SqlDataSource1" runat="server"
        ConnectionString="<%= $ ConnectionStrings:DowConectingString %>"
        SelectCommand="SELECT [UserId], [UserName], [FirstName], [LastName], [EmailID],
        [ContactNum], [CreatedOn], [Status] FROM [tblUser]"></asp:SqlDataSource>--%>

    </div>
</div>
</div>
</asp:Content>

```

*****BSOffers.aspx.cs *****

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBEntity;
using DowBAL;
using System.Data;

public partial class BSOffers: System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        //if (!IsPostBack)
        //{

```

```

ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
if (Clsusr != null)
{
    string pageid = "50";// Request.QueryString["pageid"];
    ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
Convert.ToInt32(Clsusr.GroupId));
    string msg = Request.QueryString["msg"];
    if (msg != null)
    {
        Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
    }
    if (Clsusr.GroupId == "1")
    {
        Bindetails();
    }
}
else
{
    Response.Redirect("MLB_Login.aspx");
}
}

```

```

private void ManageNavigations(int pageid, int userid, int groupid)
{
    bool IsLeftBar = false;
    IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
myBtnClickHandler);

    if (IsLeftBar == true)
    {
        LeftSideBar.Attributes["class"] = "span2";
        MainContent.Attributes["class"] = "span10";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;");//style="display:
block;"
        PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
    else
    {
        LeftSideBar.Attributes["class"] = "";
        MainContent.Attributes["class"] = "span12";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
        PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
}

```

```

// This is Event Handler
protected void myBtnClickHandler(object sender, EventArgs e)
{
    LinkButton Ink = (LinkButton)sender;
    string ID = Ink.ID;

```

```

DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
if (ds != null)
{
    if (ds.Tables[0].Rows.Count > 0)
    {
        DataTable dtdll = ds.Tables[0];
        foreach (DataRow drnew in dtdll.Rows)
        {
            switch (drnew["lbtncode"].ToString())
            {
                case "SAVE":
                    // Call SAVE Method
                    break;
                case "URL":
                    Response.Redirect(drnew["pageurl"].ToString());
                    break;
                case "AUDIT":
                    Response.Redirect(drnew["pageurl"].ToString() + "&frompage=" +
Request.QueryString["pageid"]);
                    break;
                case "CANCEL":
                    Response.Redirect("Home.aspx?pageid=1");
                    break;
                case "NEW":
                    Response.Redirect("ManageSubgroup.aspx?pageid=4&pagetype=Add");
                    break;
                default:
                    break;
            }
        }
    }
}

protected void LinkButton1_Click(object sender, EventArgs e)
{
    bool stype=Convert.ToBoolean( Request.QueryString["stype"]);
    LinkButton lnkbutton = (LinkButton)sender; ;
    decimal PSID = Convert.ToDecimal(lnkbutton.CommandName);
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    int SellerId = Convert.ToInt16(Clsusr.UserId);
    bool ret = new Products().DeleteShortList(PSID, SellerId,stype);
}

private void Bindetails()
{
    //try
    //{
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        bool stype =Convert.ToBoolean( Request.QueryString["stype"]);
    }
}

```

```

//List<EteMyUserCart> Lst = (new Products()).GetShortListbyUserId(Convert.ToInt16(
Clsusr.UserId),stype);

//if (Lst != null && Lst.Count > 0)
//{
//  WDGrid_UserList.DataSource = Lst;
//  WDGrid_UserList.DataBind();
//  WDGrid_UserList.Visible = true;
//}
//else
//{
//  WDGrid_UserList.Visible = false;
//}
}
//protected override void OnInit(EventArgs e)
//{
//  base.OnInit(e);

//  if (this.Master is MainMaster)
//  {
//    ((MainMaster)this.Master).MasterControlClicked += new
EventHandler(ContentControlClicked);
//  }
//}
private void ContentControlClicked(object sender, EventArgs e)
{
    string pageid = "pageid=" + Session["pageid"].ToString();
    Response.Redirect("~/Managedealer.aspx?PageType=Add&dealername=0&" +
pageid);
}
}

```

*****BussinessHome.aspx *****

```

<%@ Page Title="" Language="C#" MasterPageFile="~/ShoppingMasterPage.master"
AutoEventWireup="true" CodeFile="BussinessHome.aspx.cs" Inherits="BussinessHome"
%>

```

```

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
Runat="Server">
</asp:Content>
<asp:Content ID="Content4" ContentPlaceHolderID="CPH_TopMessages" Runat="Server">
<div class="span10">
<div class="row">
<!--<h3>Ravikiran Techno Soft</h3>-->
<div id="dvMsgBox" class="span10 ordersave" runat="server" style="display:
block;">
<asp:Label ID="Label2" runat="server" Text="" Height="100px"></asp:Label>
</div>
</div>
</div>
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentMainPage" Runat="Server">

```

```

<!-- Main layout -->

<div class="container-fluid">

  <!-- Section: Intro -->
  <section class="mt-md-4 pt-md-2 mb-5 pb-4">

    <!-- Grid row -->
    <div class="row">

      <!-- Grid column -->
      <div class="col-xl-3 col-md-6 mb-xl-0 mb-4">

        <!-- Card -->
        <div class="card card-cascade cascading-admin-card">

          <!-- Card Data -->
          <div class="admin-up">
            <i class="far fa-money-bill-alt primary-color mr-3 z-depth-2"></i>
            <div class="data">
              <p class="text-uppercase">sales</p>
              <h4 class="font-weight-bold dark-grey-text">2000$</h4>
            </div>
          </div>

          <!-- Card content -->
          <div class="card-body card-body-cascade">
            <div class="progress mb-3">
              <div class="progress-bar bg-primary" role="progressbar" style="width: 25%" aria-
valuenow="25" aria-valuemin="0" aria-valuemax="100"></div>
            </div>
            <p class="card-text">Better than last week (25%)</p>
          </div>

        </div>
        <!-- Card -->

      </div>
      <!-- Grid column -->

      <!-- Grid column -->
      <div class="col-xl-3 col-md-6 mb-xl-0 mb-4">

        <!-- Card -->
        <div class="card card-cascade cascading-admin-card">

          <!-- Card Data -->
          <div class="admin-up">
            <i class="fas fa-chart-line warning-color mr-3 z-depth-2"></i>
            <div class="data">
              <p class="text-uppercase">subscriptions</p>
              <h4 class="font-weight-bold dark-grey-text">200</h4>
            </div>
          </div>
        </div>
      </div>
    </div>
  </div>

```

```

    <!-- Card content -->
    <div class="card-body card-body-cascade">
      <div class="progress mb-3">
        <div class="progress-bar red accent-2" role="progressbar" style="width: 25%"
aria-valuenow="25" aria-valuemin="0" aria-valuemax="100"></div>
      </div>
      <p class="card-text">Worse than last week (25%)</p>
    </div>

  <!-- Card -->

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-xl-3 col-md-6 mb-md-0 mb-4">

  <!-- Card -->
  <div class="card card-cascade cascading-admin-card">

    <!-- Card Data -->
    <div class="admin-up">
      <i class="fas fa-chart-pie light-blue lighten-1 mr-3 z-depth-2"></i>
      <div class="data">
        <p class="text-uppercase">traffic</p>
        <h4 class="font-weight-bold dark-grey-text">20000</h4>
      </div>
    </div>

    <!-- Card content -->
    <div class="card-body card-body-cascade">
      <div class="progress mb-3">
        <div class="progress-bar red accent-2" role="progressbar" style="width: 75%"
aria-valuenow="75" aria-valuemin="0" aria-valuemax="100"></div>
      </div>
      <p class="card-text">Worse than last week (75%)</p>
    </div>

  </div>
  <!-- Card -->

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-xl-3 col-md-6 mb-0">

  <!-- Card -->
  <div class="card card-cascade cascading-admin-card">

    <!-- Card Data -->
    <div class="admin-up">
      <i class="fas fa-chart-bar red accent-2 mr-3 z-depth-2"></i>
      <div class="data">

```

```

    <p class="text-uppercase">organic traffic</p>
    <h4 class="font-weight-bold dark-grey-text">2000</h4>
  </div>
</div>

<!-- Card content -->
<div class="card-body card-body-cascade">
  <div class="progress mb-3">
    <div class="progress-bar bg-primary" role="progressbar" style="width: 25%" aria-
valuenow="25" aria-valuemin="0" aria-valuemax="100"></div>
  </div>
  <p class="card-text">Better than last week (25%)</p>
</div>

</div>
<!-- Card -->

</div>
<!-- Grid column -->

</div>
<!-- Grid row -->

</section>
<!-- Section: Intro -->

<!-- Section: Main panel -->
<section class="mb-5">

  <!-- Card -->
  <div class="card card-cascade narrower">

    <!-- Section: Chart -->
    <section>

      <!-- Grid row -->
      <div class="row">

        <!-- Grid column -->
        <div class="col-xl-5 col-lg-12 mr-0 pb-2">

          <!-- Card image -->
          <div class="view view-cascade gradient-card-header light-blue lighten-1">
            <h2 class="h2-responsive mb-0 font-weight-500">Purchased</h2>
          </div>

          <!-- Card content -->
          <div class="card-body card-body-cascade pb-0">

            <!-- Panel data -->
            <div class="row py-3 pl-4">

              <!-- First column -->
              <div class="col-md-6">

```

```

<!-- Date select -->
<p class="lead"><span class="badge info-color p-2">Data range</span></p>
<select class="mdb-select colorful-select dropdown-info md-form">
  <option value="" disabled selected>Choose time period</option>
  <option value="1">Today</option>
  <option value="2">Yesterday</option>
  <option value="3">Last 7 days</option>
  <option value="3">Last 30 days</option>
  <option value="3">Last week</option>
  <option value="3">Last month</option>
</select>

<!-- Date pickers -->
<p class="lead pt-3 pb-4"><span class="badge info-color p-2">Custom
date</span></p>
<div class="md-form">
  <input placeholder="Selected date" type="text" id="from" class="form-control
datepicker">
  <label for="date-picker-example">From</label>
</div>
<div class="md-form">
  <input placeholder="Selected date" type="text" id="to" class="form-control
datepicker">
  <label for="date-picker-example">To</label>
</div>
</div>
<!-- First column -->

<!-- Second column -->
<div class="col-md-6 text-center pl-xl-2 my-md-0 my-3">

  <!-- Summary -->
  <p>Total sales: <strong>2000$</strong>
  <button type="button" class="btn btn-info btn-sm p-2" data-toggle="tooltip"
data-placement="top"
  title="Total sales in the given period"><i class="fas fa-
question"></i></button>
  </p>
  <p>Average sales: <strong>100$</strong>
  <button type="button" class="btn btn-info btn-sm p-2 mr-0" data-
toggle="tooltip" data-placement="top"
  title="Average daily sales in the given period"><i class="fas fa-
question"></i></button>
  </p>

  <!-- Change chart -->
  <span class="min-chart my-4" id="chart-sales" data-percent="76"><span
class="percent"></span></span>
  <h5>
  <span class="badge red accent-2 p-2">Change <i class="fas fa-arrow-circle-
up ml-1"></i></span>
  <button type="button" class="btn btn-info btn-sm p-2" data-toggle="tooltip"
data-placement="top"

```



```

        title="Percentage change compared to the same period in the past"><i
class="fas fa-question"></i>
      </button>
    </h5>

  </div>
  <!-- Second column -->

</div>
<!-- Panel data -->

</div>
<!-- Card content -->

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-xl-7 col-lg-12 mb-4 pb-2">

  <!-- Chart -->
  <div class="view view-cascade gradient-card-header blue-gradient">

    <canvas id="lineChart" height="175"></canvas>

  </div>

</div>
<!-- Grid column -->

</div>
<!-- Grid row -->

</section>
<!-- Section: Chart -->

<!-- Section: Table -->
<section>

  <!-- Top Table UI -->
  <div class="table-ui p-2 mb-3 mx-4 mb-5">

    <!-- Grid row -->
    <div class="row">

      <!-- Grid column -->
      <div class="col-xl-3 col-md-6">

        <!-- Name -->
        <select class="mdb-select colorful-select dropdown-info mx-2 md-form mt-3 md-
dropdown">
          <option value="" disabled selected>Bulk actions</option>
          <option value="1">Delate</option>
          <option value="2">Export</option>
          <option value="3">Change segment</option>

```

```

    </select>

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-xl-3 col-md-6">

    <!-- Blue select -->
    <select class="mdb-select colorful-select dropdown-info mx-2 md-form mt-3 md-
dropdown">
        <option value="" disabled selected>Show only</option>
        <option value="1">All <span> (2000)</span></option>
        <option value="2">Never opened <span> (200)</span></option>
        <option value="3">Opened but unanswered <span> (1800)</span></option>
        <option value="4">Answered <span> (200)</span></option>
        <option value="5">Unsubscribed <span> (50)</span></option>
    </select>

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-xl-3 col-md-6">

    <!-- Blue select -->
    <select class="mdb-select colorful-select dropdown-info mx-2 md-form mt-3 md-
dropdown">
        <option value="" disabled selected>Filter segments</option>
        <option value="1">Contacts in no segments <span> (100)</span></option>
        <option value="1">Segment 1 <span> (2000)</span></option>
        <option value="2">Segment 2 <span> (1000)</span></option>
        <option value="3">Segment 3 <span> (4000)</span></option>
    </select>

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-xl-3 col-md-6">

    <div class="form-inline mt-2 ml-2">
        <div class="form-group md-form mt-2">
            <input class="form-control w-100" type="text" placeholder="Search">
        </div>
        <button class="btn btn-sm btn-primary ml-2 mr-0 mb-md-0 mb-4 px-2"><i
class="fas fa-search"></i></button>
    </div>

</div>
<!-- Grid column -->

</div>
<!-- Grid row -->

```

```

        </div>
        <!-- Top Table UI -->

        </section>
        <!--Section: Table-->

        </div>
        <!-- Card -->

        </section>
        <!-- Section: Main panel -->

    </div>

    <!-- Main layout --></asp:Content>

```

*****BussinessHome.aspx.cs *****

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBAL;
using DowBEntity;
using System.Web.Security;
using System.Data;
public partial class BussinessHome : System.Web.UI.Page
{
    DDLogTracer objlog = new DDLogTracer();
    protected void Page_Load(object sender, EventArgs e)
    {
        try
        {
            bool IsMobile = false;
            if (Request.Browser.IsMobileDevice) { IsMobile = true; } else { IsMobile = false; }
            if (IsMobile)
            {
            }

            ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedUser_Specs"];
            if (Clsusr != null)
            {
                if (Clsusr.GroupId != "8")
                {
                    Response.Redirect("Home.aspx");
                }
            }
            else
            {
                string user;
                HttpCookie myCookie = Request.Cookies["DateOrWedCookie"];
                if (myCookie != null)

```

```

    {
        user = myCookie.Value.ToString();
    }
    else
    {
        user = "Guest";
    }
    guest_login(user);

    //      Response.Redirect("MLB_Login.aspx");
}
string pageid = Request.QueryString["pageid"];
if (pageid == null) { pageid = "1"; }
Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
if (Clsusr != null)
{
}

string msg = Request.QueryString["msg"];
if (msg != null)
{
    Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
}
}
catch (Exception ex)
{
    Common.ShowMessageHtml(ex.Message, 1, true, dvMsgBox);
    ExceptionLogging.SendExcepToDB(ex);

    objlog.Trace_Error(5, "DateOrWed", "OrderSearchList", "Bindetails_Filter",
ex.Message);

}
}

protected void guest_login(string user)
{
    string City = "Hyderabad";// _City;
    string Country = "India";// _Country;

    BALCreateUser objlogin = new BALCreateUser();
    DataTable status_login = objlogin.LoginGuest(user, user,
Request.ServerVariables["REMOTE_ADDR"].ToString(), "", City, Country);
    string login_status = status_login.Rows[0]["UserStatus"].ToString();

    ClsLoggedInUser objLoggedInUser;
    if (login_status == "True")
    {
        objLoggedInUser = new ClsLoggedInUser();
    }
}

```

```

objLogedInUser.UserId = status_login.Rows[0]["UserId"].ToString();
objLogedInUser.UserName = status_login.Rows[0]["UserName"].ToString();
objLogedInUser.GroupId = status_login.Rows[0]["GroupId"].ToString();
objLogedInUser.LoginStatus = "True";
objLogedInUser.LoginID = Convert.ToInt32(status_login.Rows[0]["LoginId"]);
objLogedInUser.DealerId = Convert.ToInt32(status_login.Rows[0]["DealerId"]);
objLogedInUser.RoleId = Convert.ToInt32(status_login.Rows[0]["RoleId"]);
objLogedInUser.BrandId = 1; //Convert.ToInt32(status_login.Rows[0]["BrandId"]);

Session["LoggedUser_Specs"] = objLogedInUser;

FormsAuthentication.SetAuthCookie(objLogedInUser.UserName, false);
HttpCookie myCookie = new HttpCookie("DateOrWedCookie");
DateTime now = DateTime.Now;
myCookie.Value = objLogedInUser.UserName;
myCookie.Expires = now.AddYears(1);

Response.Cookies.Add(myCookie);

ClsPermissioGroup clsperGrp = new ClsPermissioGroup();
UserPermissions Entuserper =
clsperGrp.GetUserLandingPage(objLogedInUser.UserName);

if (Entuserper.Pageld != 0)
{
    Entuserper.Add = true;
    Entuserper.Edit = true;
    Entuserper.Delete = true;
    Entuserper.View = true;
    //BalRecentPlacess objRemoveRcntPlc = new BalRecentPlacess();
    //bool rst =
objRemoveRcntPlc.DeleteRecentPlaces(Convert.ToInt32(status_login.Rows[0]["UserId"].To
String()));
    Response.Redirect("~/BusinessHome.aspx");

}

}

}

}
***** BuyerHome.aspx*****
<%@ Page Title="DateOrWed-Buyer-Home" Language="C#"
MasterPageFile="~/ShoppingMasterPage.master" AutoEventWireup="true"
CodeFile="BuyerHome.aspx.cs" Inherits="BuyerHome" %>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
    <!-- BOOTSTRAP -->
    <link rel="stylesheet" href="shopping/css/bootstrap.min.css">
    <!-- FONT AWESOME -->
    <link rel="stylesheet" href="shopping/css/font-awesome.min.css">
    <!-- STYLE -->
    <link rel="stylesheet" href="shopping/css/style.css">
    <!-- ANIMATE -->
    <link rel="stylesheet" href="shopping/css/animate.css">

```

```

    <!-- MAGNIFIC POPUP -->
    <link rel="stylesheet" href="shopping/css/magnific-popup.css">
    <!-- FLEXSLIDER -->
    <link rel="stylesheet" href="shopping/plugins/FlexSlider/jquery.flexslider.css">
    <!-- OWL CAROUSEL -->
    <link rel="stylesheet" href="shopping/plugins/owl-carousel/owl.carousel.css">
    <!-- OWL CAROUSEL THEME -->
    <link rel="stylesheet" href="shopping/plugins/owl-carousel/owl.theme.css">
    <!-- slick -->
    <link rel="stylesheet" type="text/css" href="shopping/css/slick.css">
    <!-- slick-theme -->
    <link rel="stylesheet" type="text/css" href="shopping/css/slick-theme.css">

    <!-- WEB FONTS -->
    <link href='http://fonts.googleapis.com/css?family=Open+Sans:300,400,600,700'
rel='stylesheet' type='text/css'>
    <link href="http://fonts.googleapis.com/css?family=Raleway:500,800" rel="stylesheet"
property="stylesheet" type="text/css" media="all" />
    <!-- JQUERY LIBS -->
    <script type="text/javascript" src="shopping/js/jquery.min.js"></script>
    <!-- BOOTSTRAP -->
    <script type="text/javascript" src="shopping/js/bootstrap.min.js"></script>
    <!-- OWL CAROUSEL -->
    <script type="text/javascript" src="shopping/plugins/owl-
carousel/owl.carousel.min.js"></script>
    <!-- FLEXSLIDER -->
    <script type="text/javascript" src="shopping/plugins/FlexSlider/jquery.flexslider-
min.js"></script>
    <!-- MAGNIFIC POPUP-->
    <script type="text/javascript" src="shopping/js/jquery.magnific-popup.min.js"></script>
    <!-- MAGNIFIC POPUP -->
    <script type="text/javascript" src="shopping/js/magnific-popup.min.js"></script>
    <!-- slick-->
    <script type="text/javascript" src="shopping/js/slick.js"></script>
    <!-- slick-slider-->
    <script type="text/javascript" src="shopping/js/slick-slider.js"></script>
    <!-- plugins-->
    <script type="text/javascript" src="shopping/js/plugins.min.html"></script>
    <!-- CUSTOM-->
    <script type="text/javascript" src="shopping/js/custom.js"></script>

```

```
</asp:Content>
```

```

<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
<div class="container-fluid">
    <div class="row-fluid"> <%-- style="padding-top: 20px;" --%>
    <div class="span10">
    <div class="row">
        <%--<h3>Ravikiran Techno Soft</h3>--%>
        <div id="dvMsgBox" class="span10 ordersave" runat="server" style="display:
block;">
            <asp:Label ID="Label2" runat="server" Text=""></asp:Label>
        </div>
    </div>
    </div>
</div>

```

```
    </div>
  </div>
</div>
</div>
</asp:Content>
```

```
<asp:Content ID="Content2" ContentPlaceHolderID="ContentMainPage" runat="Server"
Visible="false" >
```

```
  <div class="content_holder">
    <section class="clearfix">
      <div id="megaslider" class="carousel slide" data-ride="carousel" runat="server">
        <ol class="carousel-indicators">
          <li data-target="#mega-slider" data-slide-to="0" class="active"></li>
          <li data-target="#mega-slider" data-slide-to="1"></li>
          <li data-target="#mega-slider" data-slide-to="2"></li>
        </ol>
        <div class="carousel-inner" role="listbox">
          <div class="item active beactive">
            
            <div class="carousel-caption">
              <h2>Welcome to DateOrWed</h2>
              <p>Your online Shopping Destination</p>
            </div>
          </div>
          <div class="item">
            
            <div class="carousel-caption">
              <h2>A Complete Wedding Online Store</h2>
              <p>Fashion Wedding Clothing Specially for Wedding</p>
            </div>
          </div>
          <div class="item">
            
            <div class="carousel-caption">
              <h2>Everything for your wedding</h2>
              <p>A to Z Wedding Portal</p>
            </div>
          </div>
        </div>
        <!-- Controls -->
        <a class="left carousel-control" href="#mega-slider" role="button" data-
slide="prev">
          </a>
        <a class="right carousel-control" href="#mega-slider" role="button" data-
slide="next">
          </a>
        </div>
      </section>
      <!-- Search -->
      <%--<section class="search-sec">
        <div class="container">
          <div class="row">
            <div class="col-lg-12">
```



```

    <h6 class="text-uppercase bottom-line">New Products</h6>
  </div>
  <div class="row">
    <div class="new-product slider">
      <div class="col-md-4 item">
        <div class="products grid-product">
          <div class="product-grid-inner">
            <div class="product-grid-img">
              <ul class="product-labels">
                <li>Sale</li>
              </ul>
              <a href="#" class="product-grid-img-thumbnail">
                
                
              </a>
              <!-- <div class="product-grid-atb text-center"><i class="fa fa-
shopping-cart"></i><a class="" href="#">Add to Cart</a></div> -->
            </div>

            <div class="product-grid-caption text-center">
              <h4><a class="p-grid-title" href="#">Bride Fashion
Dress</a></h4>
              <!-- <span class="price"><span class="old-price">$
150,00</span> $ <span class="special-price">99,99</span></span>
              <ul class="list-inline add-to-wishlist">
                <li class="product-wishlist">
                  <i class="fa fa-heart"></i>
                  <a href="#">Add to Wishlist</a>
                </li>
                <li class="product-wishlist">
                  <i class="fa fa-exchange-alt"></i>
                  <a href="#">Add to Compare</a>
                </li>
              </ul> -->
            </div>
          </div>
        </div>
      </div>
    </div>
    <div class="col-md-4 item">
      <div class="products grid-product">
        <div class="product-grid-inner">
          <div class="product-grid-img">
            <ul class="product-labels">
              <li>Sale</li>
            </ul>
            <a href="#" class="product-grid-img-thumbnail">
              
              
            </a>

```

```
        <!-- <div class="product-grid-atb text-center"><i class="fa fa-  
shopping-cart"></i><a class="" href="#">Add to Cart</a></div> -->  
    </div>
```

```
    <div class="product-grid-caption text-center">  
        <h4><a class="p-grid-title" href="#">Groom Fashion  
Dress</a></h4>
```

```
        <!-- <span class="price"><span class="old-price">$  
150,00</span> $ <span class="special-price">99,99</span></span>  
        <ul class="list-inline add-to-wishlist">  
            <li class="product-wishlist">  
                <i class="fa fa-heart"></i>  
                <a href="#">Add to Wishlist</a>  
            </li>  
            <li class="product-wishlist">  
                <i class="fa fa-exchange-alt"></i>  
                <a href="#">Add to Compare</a>  
            </li>  
        </ul> -->
```

```
    </div>  
</div>  
</div>  
</div>  
<div class="col-md-4 item">  
    <div class="products grid-product">  
        <div class="product-grid-inner">  
            <div class="product-grid-img">  
                <ul class="product-labels">  
                    <li>Sale</li>  
                </ul>  
                <a href="#" class="product-grid-img-thumbnail">  
                      
                      
                </a>  
                <!-- <div class="product-grid-atb text-center"><i class="fa fa-  
shopping-cart"></i><a class="" href="#">Add to Cart</a></div> -->  
            </div>
```

```
    <div class="product-grid-caption text-center">  
        <h4><a class="p-grid-title" href="#">Bridegroom Fashion  
Dress</a></h4>
```

```
        <!-- <span class="price"><span class="old-price">$  
150,00</span> $ <span class="special-price">99,99</span></span>  
        <ul class="list-inline add-to-wishlist">  
            <li class="product-wishlist">  
                <i class="fa fa-heart"></i>  
                <a href="#">Add to Wishlist</a>  
            </li>  
            <li class="product-wishlist">  
                <i class="fa fa-exchange-alt"></i>  
                <a href="#">Add to Compare</a>  
            </li>
```

```

        </ul> -->
    </div>
</div>
</div>
</div>
<div class="col-md-4 item">
    <div class="products grid-product">
        <div class="product-grid-inner">
            <div class="product-grid-img">
                <ul class="product-labels">
                    <li>Sale</li>
                </ul>
                <a href="#" class="product-grid-img-thumbnail">
                    
                    
                </a>
                <!-- <div class="product-grid-atb text-center"><i class="fa fa-
shopping-cart"></i><a class="" href="#">Add to Cart</a></div> -->
            </div>

            <div class="product-grid-caption text-center">
                <h4><a class="p-grid-title" href="#">Printed Summer
Dress</a></h4>
                <!-- <span class="price"><span class="old-price">$
150,00</span> $ <span class="special-price">99,99</span></span>
                <ul class="list-inline add-to-wishlist">
                    <li class="product-wishlist">
                        <i class="fa fa-heart"></i>
                        <a href="#">Add to Wishlist</a>
                    </li>
                    <li class="product-wishlist">
                        <i class="fa fa-exchange-alt"></i>
                        <a href="#">Add to Compare</a>
                    </li>
                </ul> -->
            </div>
        </div>
    </div>
</div>
</div>
</div>
<div class="col-md-4 item">
    <div class="products grid-product">
        <div class="product-grid-inner">
            <div class="product-grid-img">
                <ul class="product-labels">
                    <li>Sale</li>
                </ul>
                <a href="#" class="product-grid-img-thumbnail">
                    
                    

```

```
    </a>
    <!-- <div class="product-grid-atb text-center"><i class="fa fa-
shopping-cart"></i><a class="" href="#">Add to Cart</a></div> -->
</div>
```

```
<div class="product-grid-caption text-center">
<h4><a class="p-grid-title" href="#">Printed Summer
Dress</a></h4>
```

```
    <!-- <span class="price"><span class="old-price">$
150,00</span> $ <span class="special-price">99,99</span></span>
<ul class="list-inline add-to-wishlist">
  <li class="product-wishlist">
    <i class="fa fa-heart"></i>
    <a href="#">Add to Wishlist</a>
  </li>
  <li class="product-wishlist">
    <i class="fa fa-exchange-alt"></i>
    <a href="#">Add to Compare</a>
  </li>
</ul> -->
```

```
    </div>
  </div>
</div>
<div class="col-md-4 item">
  <div class="products grid-product">
    <div class="product-grid-inner">
      <div class="product-grid-img">
        <ul class="product-labels">
          <li>Sale</li>
        </ul>
        <a href="#" class="product-grid-img-thumbnail">
          
          
        </a>
        <!-- <div class="product-grid-atb text-center"><i class="fa fa-
shopping-cart"></i><a class="" href="#">Add to Cart</a></div> -->
      </div>
```

```
<div class="product-grid-caption text-center">
<h4><a class="p-grid-title" href="#">Printed Summer
Dress</a></h4>
```

```
    <!-- <span class="price"><span class="old-price">$
150,00</span> $ <span class="special-price">99,99</span></span>
<ul class="list-inline add-to-wishlist">
  <li class="product-wishlist">
    <i class="fa fa-heart"></i>
    <a href="#">Add to Wishlist</a>
  </li>
  <li class="product-wishlist">
    <i class="fa fa-exchange-alt"></i>
    <a href="#">Add to Compare</a>
```

```

        </li>
    </ul> -->

</div>
</div>
</div>
</div>

</div>
</div>
</div>
<!-- JQUERY LIBS -->
<script type="text/javascript" src="shopping/js/jquery.min.js"></script>
<!-- BOOTSTRAP -->
<script type="text/javascript" src="shopping/js/bootstrap.min.js"></script>
<!-- OWL CAROUSEL -->
<script type="text/javascript" src="shopping/plugins/owl-
carousel/owl.carousel.min.js"></script>
<!-- FLEXSLIDER -->
<script type="text/javascript" src="shopping/plugins/FlexSlider/jquery.flexslider-
min.js"></script>
<!-- MAGNIFIC POPUP-->
<script type="text/javascript" src="shopping/js/jquery.magnific-popup.min.js"></script>
<!-- MAGNIFIC POPUP -->
<script type="text/javascript" src="shopping/js/magnific-popup.min.js"></script>
<!-- slick-->
<script type="text/javascript" src="shopping/js/slick.js"></script>
<!-- slick-slider-->
<script type="text/javascript" src="shopping/js/slick-slider.js"></script>
<!-- plugins-->
<script type="text/javascript" src="shopping/js/plugins.min.html"></script>
<!-- CUSTOM-->
<script type="text/javascript" src="shopping/js/custom.js"></script>
</section>
<!-- END NEW -->

<!-- START BANNER STYLE -->
<section class="home-main-contant-style banner-style3 bg-white">
<div class="container">
<div class="row">
<div class="col-md-4 col-sm-4 col-xs-12">
<div class="row">
<div class="col-md-12 mb30 banner-coll2">
<div class="image">
<a href="#">

</a>
</div>
</div>

```

```
        </div>
        <div class="col-md-12 banner-coll2">
            <div class="image">
                <a href="#">
                    
                </a>
            </div>
        </div>
    </div>
</div>
<div class="col-md-4 col-sm-4 col-xs-12 banner-coll2">
    <div class="image">
        <a href="#">
            
        </a>
    </div>
</div>
<div class="col-md-4 col-sm-4 col-xs-12">
    <div class="row">
        <div class="col-md-12 mb30 banner-coll2">
            <div class="image">
                <a href="#">
                    
                </a>
            </div>
        </div>
        <div class="col-md-12 banner-coll2">
            <div class="image">
                <a href="#">
                    
                </a>
            </div>
        </div>
    </div>
</div>
</div>
</div>
</div>
</section>
<!-- END BANNER STYLE -->
```

```
<!-- START Featured -->
<section class="home-main-contant-style3 col-grid-style bg-white">
    <div class="container">
        <div class="widget">
            <h6 class="text-uppercase bottom-line">Featured Products</h6>
        </div>
        <div class="row clearfix">
```

```

<div class="col-md-3 col-sm-6 col-xs-12">
  <div class="products grid-product mb30">
    <div class="product-grid-inner">
      <div class="product-grid-img">
        <ul class="product-labels">
          <li>Sale</li>
        </ul>
        <a href="#" class="product-grid-img-thumbnail">
          
          
        </a>
        <!-- <div class="product-grid-atb text-center"><a class="" href="#">Quick View</a></div> -->
      </div>

      <div class="product-grid-caption text-center">
        <h6><a class="p-grid-title" href="#">Printed Summer
Dress</a></h6>

        <!-- <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
        <div class="pro-size"><span>XL / XXL / S </span></div>
        <span class="price"><span class="old-price">$ 150,00</span> $
<span class="special-price">99,99</span></span>
        <a class="btn btn-product cd-cart-btn" href="#"><i class="fa fa-shopping-cart"></i> Add to Cart</a> -->
      </div>
    </div>
  </div>
</div>
<div class="col-md-3 col-sm-6 col-xs-12">
  <div class="products grid-product mb30">
    <div class="product-grid-inner">
      <div class="product-grid-img">
        <ul class="product-labels">
          <li>Sale</li>
        </ul>
        <a href="#" class="product-grid-img-thumbnail">
          
          
        </a>
        <!-- <div class="product-grid-atb text-center"><a class="" href="#">Quick View</a></div> -->
      </div>

      <div class="product-grid-caption text-center">
        <h6><a class="p-grid-title" href="#">Printed Summer
Dress</a></h6>

        <!-- <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
        <div class="pro-size"><span>XL / XXL / S </span></div>
        <span class="price"><span class="old-price">$ 150,00</span> $
<span class="special-price">99,99</span></span>

```

```

        <a class="btn btn-product cd-cart-btn" href="#"><i class="fa fa-
shopping-cart"></i> Add to Cart</a> -->
    </div>
</div>
</div>
</div>
<div class="col-md-3 col-sm-6 col-xs-12">
    <div class="products grid-product mb30">
        <div class="product-grid-inner">
            <div class="product-grid-img">
                <ul class="product-labels">
                    <li>Sale</li>
                </ul>
                <a href="#" class="product-grid-img-thumbnail">
                    
                    
                </a>
                <!-- <div class="product-grid-atb text-center"><a class=""
href="#">Quick View</a></div> -->
            </div>

            <div class="product-grid-caption text-center">
                <h6><a class="p-grid-title" href="#">Printed Summer
Dress</a></h6>
                <!-- <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
                <div class="pro-size"><span>XL / XXL / S </span></div>
                <span class="price"><span class="old-price">$ 150,00</span> $
<span class="special-price">99,99</span></span>
                <a class="btn btn-product cd-cart-btn" href="#"><i class="fa fa-
shopping-cart"></i> Add to Cart</a> -->
            </div>
        </div>
    </div>
</div>
</div>
</div>
<div class="col-md-3 col-sm-6 col-xs-12">
    <div class="products grid-product mb30">
        <div class="product-grid-inner">
            <div class="product-grid-img">
                <ul class="product-labels">
                    <li>Sale</li>
                </ul>
                <a href="#" class="product-grid-img-thumbnail">
                    
                    
                </a>
                <!-- <div class="product-grid-atb text-center"><a class=""
href="#">Quick View</a></div> -->
            </div>

            <div class="product-grid-caption text-center">

```



```

        <h6><a class="p-grid-title" href="#">Printed Summer
Dress</a></h6>
        <!-- <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
        <div class="pro-size"><span>XL / XXL / S </span></div>
        <span class="price"><span class="old-price">$ 150,00</span> $
<span class="special-price">99,99</span></span>
        <a class="btn btn-product cd-cart-btn" href="#"><i class="fa fa-
shopping-cart"></i> Add to Cart</a> -->
        </div>
        </div>
        </div>
        </div>
        <div class="col-md-3 col-sm-6 col-xs-12">
        <div class="products grid-product">
        <div class="product-grid-inner">
        <div class="product-grid-img">
        <ul class="product-labels">
        <li>Sale</li>
        </ul>
        <a href="#" class="product-grid-img-thumbnail">
        
        
        </a>
        <!-- <div class="product-grid-atb text-center"><a class=""
href="#">Quick View</a></div> -->
        </div>

        <div class="product-grid-caption text-center">
        <h6><a class="p-grid-title" href="#">Printed Summer
Dress</a></h6>
        <!-- <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
        <div class="pro-size"><span>XL / XXL / S </span></div>
        <span class="price"><span class="old-price">$ 150,00</span> $
<span class="special-price">99,99</span></span>
        <a class="btn btn-product cd-cart-btn" href="#"><i class="fa fa-
shopping-cart"></i> Add to Cart</a> -->
        </div>
        </div>
        </div>
        </div>
        <div class="col-md-3 col-sm-6 col-xs-12">
        <div class="products grid-product">
        <div class="product-grid-inner">
        <div class="product-grid-img">
        <ul class="product-labels">
        <li>Sale</li>
        </ul>
        <a href="#" class="product-grid-img-thumbnail">
        
        
        </a>

```

```

        <!-- <div class="product-grid-atb text-center"><a class=""
href="#">Quick View</a></div> -->
    </div>

    <div class="product-grid-caption text-center">
        <h6><a class="p-grid-title" href="#">Printed Summer
Dress</a></h6>

        <!-- <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
        <div class="pro-size"><span>XL / XXL / S </span></div>
        <span class="price"><span class="old-price">$ 150,00</span> $
<span class="special-price">99,99</span></span>
        <a class="btn btn-product cd-cart-btn" href="#"><i class="fa fa-
shopping-cart"></i> Add to Cart</a> -->
    </div>
</div>
</div>
</div>
<div class="col-md-3 col-sm-6 col-xs-12">
    <div class="products grid-product">
        <div class="product-grid-inner">
            <div class="product-grid-img">
                <ul class="product-labels">
                    <li>Sale</li>
                </ul>
                <a href="#" class="product-grid-img-thumbnail">
                    
                    
                </a>
                <!-- <div class="product-grid-atb text-center"><a class=""
href="#">Quick View</a></div> -->
            </div>

            <div class="product-grid-caption text-center">
                <h6><a class="p-grid-title" href="#">Printed Summer
Dress</a></h6>

                <!-- <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
                <div class="pro-size"><span>XL / XXL / S </span></div>
                <span class="price"><span class="old-price">$ 150,00</span> $
<span class="special-price">99,99</span></span>
                <a class="btn btn-product cd-cart-btn" href="#"><i class="fa fa-
shopping-cart"></i> Add to Cart</a> -->
            </div>
        </div>
    </div>
</div>
<div class="col-md-3 col-sm-6 col-xs-12">
    <div class="products grid-product">
        <div class="product-grid-inner">
            <div class="product-grid-img">
                <ul class="product-labels">
                    <li>Sale</li>
                </ul>
                <a href="#" class="product-grid-img-thumbnail">

```

```

        
        
        </a>
        <!-- <div class="product-grid-atb text-center"><a class=""
href="#">Quick View</a></div> -->
        </div>

        <div class="product-grid-caption text-center">
        <h6><a class="p-grid-title" href="#">Printed Summer
Dress</a></h6>
        <!-- <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
        <div class="pro-size"><span>XL / XXL / S </span></div>
        <span class="price"><span class="old-price">$ 150,00</span> $
<span class="special-price">99,99</span></span>
        <a class="btn btn-product cd-cart-btn" href="#"><i class="fa fa-
shopping-cart"></i> Add to Cart</a> -->
        </div>
        </div>
        </div>
        </div>
        </div>
        </div>
        </section>
        <!-- END Featured -->

</div>

<%-- app footer here--%>
</asp:Content>

```

```

***** BuyerHome.aspx.cs*****
using DowBAL;
using DowBEntity;
using System;
using System.Collections.Generic;
using System.Data;
using System.Linq;
using System.Text;
using System.Text.RegularExpressions;
using System.Web;
using System.Web.UI;

```

```

using System.Web.UI.WebControls;
using System.Web.Security;
//using LogTracer;

public partial class BuyerHome : System.Web.UI.Page
{
    public string UserName = "";
    DDLogTracer objlog = new DDLogTracer();
    protected void Page_Load(object sender, EventArgs e)
    {
        try
        {
            bool IsMobile = false;
            if (Request.Browser.IsMobileDevice) { IsMobile = true; } else { IsMobile = false; }
            if (IsMobile)
            {
                megaslider.Attributes.CssStyle.Add("display", "none;");
            }

            ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
            if (Clsusr != null)
            {
                UserName = Clsusr.UserName;
                if (Clsusr.GroupId != "2" || Clsusr.GroupId != "5")
                {
                    //Response.Redirect("Home.aspx");
                }
            }
            else
            {
                string user;
                HttpCookie myCookie = Request.Cookies["DateOrWedCookie"];
                if (myCookie != null)
                {
                    user = myCookie.Value.ToString();
                }
                else
                {
                    user = "Guest";
                }
                guest_login(user);

                // Response.Redirect("MLB_Login.aspx");
            }
            string pageid = Request.QueryString["pageid"];
            if (pageid == null) { pageid = "1"; }
            Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
            if (Clsusr != null)
            {
            }
            string msg = Request.QueryString["msg"];
            if (msg != null)
            {

```

```

        Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
    }
    Bindetails(Convert.ToInt16(Clsusr.GroupId));
}
catch (Exception ex)
{
    Common.ShowMessageHtml(ex.Message, 1, true, dvMsgBox);
    ExceptionLogging.SendExcepToDB(ex);

    objlog.Trace_Error(5, "DateOrWed", "Shopping", "Page_load", ex.Message);

}
}

private void ManageNavigations(int pageid, int userid, int groupid)
{
    bool IsLeftBar = false;
    //IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
myBtnClickHandler);

    if (IsLeftBar == true)
    {
    }
    else
    {
    }
}

protected void guest_login(string user)
{

    string City = "Hyderabad";// _City;
    string Country = "India";// _Country;

    BALCreateUser objlogin = new BALCreateUser();
    DataTable status_login = objlogin.LoginGuest(user, user,
Request.ServerVariables["REMOTE_ADDR"].ToString(), "", City, Country);
    string login_status = status_login.Rows[0]["UserStatus"].ToString();

    ClsLoggedInUser objLogedInUser;
    if (login_status == "True")
    {
        objLogedInUser = new ClsLoggedInUser();

        objLogedInUser.UserId = status_login.Rows[0]["UserId"].ToString();
        objLogedInUser.UserName = status_login.Rows[0]["UserName"].ToString();
        objLogedInUser.GroupId = status_login.Rows[0]["GroupId"].ToString();
        objLogedInUser.LoginStatus = "True";
        objLogedInUser.LoginID = Convert.ToInt32(status_login.Rows[0]["LoginId"]);
        objLogedInUser.DealerId = Convert.ToInt32(status_login.Rows[0]["DealerId"]);
        objLogedInUser.RoleId = Convert.ToInt32(status_login.Rows[0]["RoleId"]);
        objLogedInUser.BrandId = 1; //Convert.ToInt32(status_login.Rows[0]["BrandId"]);
    }
}

```

```

Session["LoggedUser_Specs"] = objLogedInUser;

FormsAuthentication.SetAuthCookie(objLogedInUser.UserName, false);
HttpCookie myCookie = new HttpCookie("DateOrWedCookie");
DateTime now = DateTime.Now;
myCookie.Value = objLogedInUser.UserName;
myCookie.Expires = now.AddYears(1);

Response.Cookies.Add(myCookie);

ClsPermissioGroup clsperGrp = new ClsPermissioGroup();
UserPermissions Entuserper =
clsperGrp.GetUserLandingPage(objLogedInUser.UserName);

if (Entuserper.Pageld != 0)
{
    Entuserper.Add = true;
    Entuserper.Edit = true;
    Entuserper.Delete = true;
    Entuserper.View = true;
    //BalRecentPlacess objRemoveRcntPlc = new BalRecentPlacess();
    //bool rst =
objRemoveRcntPlc.DeleteRecentPlaces(Convert.ToInt32(status_login.Rows[0]["UserId"].ToString()));
    Response.Redirect("~/ " + Entuserper.PageUrl);
}

}
}
// This is Event Handler
private void Bindetails(int Groupid)
{
    if (Groupid == 0)
    {

    }
    else if (Groupid == 3 || Groupid == 1)
    {
        List<Eteproductbuyers> Lstx = (new Products()).GetProductBuyers(1, Groupid, "");

        if (Lstx != null && Lstx.Count > 0)
        {
        }
        else
        {
        }
        List<Eteproductbuyers> Lst1 = (new Products()).GetProductBuyers(2, Groupid, "");

        if (Lst1 != null && Lst1.Count > 0)
        {
        }
        else
        {

```

```

    }
}
else
{

}
//try
//{{
//CategoriesDS.DataBind();
//ProductsGrDS.DataBind();
//ProductsDS.DataBind();
//whds.DataBind();
//WebDataTree1.DataSourceID = "whds";

//List<Eteproductcat> Lst = (new Products()).GetProductCat("Managedealer.aspx");

//if (Lst != null && Lst.Count > 0)
//{{

//WebDataTree1.DataSourceID DataSource = Lst;
//WebDataTree1.DataBind();
//WebDataTree1.Visible = true;
///}
//else
//{{
// WebDataTree1.Visible = false;
//}
}
protected void myBtnClickHandler(object sender, EventArgs e)
{
    LinkButton lnk = sender as LinkButton;

    // LinkButton lnk = (LinkButton)sender;
    if (lnk != null)
    {

        string ID = lnk.ID;

        DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
        if (ds != null)
        {
            if (ds.Tables[0].Rows.Count > 0)
            {
                DataTable dtdll = ds.Tables[0];
                foreach (DataRow drnew in dtdll.Rows)
                {
                    switch (drnew["lbtncode"].ToString())
                    {
                        case "SAVE":
                            // Call SAVE Method
                            break;
                        case "URL":
                            Response.Redirect(drnew["pageurl"].ToString() + "&pid=" + ID);
                    }
                }
            }
        }
    }
}

```



```

    {
        sb.Append(" <div class='item" + tm + "'>\n");
        sb.Append("<img src='" + row.fpath + "' alt='" + row.aname + "' style='width:auto;
onclick=~/ViewSearchResult.aspx?id='" + row.id.ToString() + "'/>\n");
        sb.Append(" </div>\n");

        tm = "";
    }
    sb.Append(" <a class='left carousel-control' href='#myCarousel' data-slide='prev'>\n");
    sb.Append("     <span class='glyphicon glyphicon-chevron-left'></span>\n");
    sb.Append("     <span class='sr-only'><</span>\n");
    sb.Append(" </a>\n");
    sb.Append(" <a class='right carousel-control' href='#myCarousel' data-
slide='next'>\n");
    sb.Append("     <span class='glyphicon glyphicon-chevron-right'></span>\n");
    sb.Append("     <span class='sr-only'>></span>\n");
    sb.Append(" </a>\n");
    //sb.Append("     <a class='igte_Button' href='ViewSearchResult.aspx'>Search</a>\n");

    PreHtml = sb.ToString();
    //objlog.Trace_Log(4, "home", "1", "corousal", PreHtml);
    if (PreHtml.Length > 0)
    {
        //myCarousel.InnerHtml = PreHtml;
    }
}

private void FillRecentNews()
{
    string PreHtml = ""; // "<h5><strong>Recent News</strong></h5><table class='table'>";
    StringBuilder sb = new StringBuilder();
    //sb.Append("<h5 class='customer'><strong>Recent News</h5><table class='table'>");
    sb.Append("<table class='table'>");
    //THU 28 OCT 2015 By Sudheer
    List<EteNewsItem> RecentNew = (new NewsItems()).GetNewsItems();
    foreach (var row in RecentNew)
    {
        string PubDate = row.PublishDate.ToString("ddd dd MMM yyyy").ToUpper(); //ddd-
mmm-yyyy hh:mm:ss
        sb.Append("<tr><th width=250px>" + row.Summary + "</th><td style='font-size:
12px; text-align: right;'>" + PubDate + " By " + row.UserName + "</td></tr>");
        sb.Append("<tr><td style='font-size: 12px; text-align: left; padding-left: 15px; border:
0px; padding-top: 5px; padding-bottom: 5px; color: blue; font-weight: 600;'>" + row.Details +
"</td></tr>");
    }
    PreHtml = sb.ToString();
    if (PreHtml.Length > 0)
    {
        PreHtml += "</table>";
        //dv_RecentNews.InnerHtml = PreHtml;
    }
}
}

```

```

//private void FillCommonCount()
//{{

// ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
// if (Clsusr != null)
// {
//     EteCommonCount ComnCount = new
NewsItems().GetCommonCount(Clsusr.RoleId, Clsusr.DealerId,
Convert.ToInt32(Clsusr.UserId));
//     if (ComnCount != null)
//     {
//         ;

//         decimal SevenDaysCanPercent =
Convert.ToDecimal((ComnCount.CancellationPerLast7Days ?? 0) /
(Convert.ToDecimal(ComnCount.Last7Days == 0 ? 1 : ComnCount.Last7Days)) * 100);
//         decimal TwentyEightDaysCanPercent =
Convert.ToDecimal((ComnCount.CancellationPerLast28Days ?? 0) /
(Convert.ToDecimal(ComnCount.Last28Days == 0 ? 1 : ComnCount.Last28Days)) * 100);

//     }

// }
//}}

private void FillMostRecentOrder()
{
    string PreHtml = ""; // "<h5><strong>Recent News</strong></h5><table class='table'>";
    StringBuilder sb = new StringBuilder();
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    if (Clsusr != null)
    {
        //List<EteMostRecentOrders> MostRecentOrders = new
NewsItems().GetMostRecentOrders(Clsusr.RoleId, Clsusr.DealerId,
Convert.ToInt32(Clsusr.UserId));
        //foreach (var row in MostRecentOrders)
        //{
        //    sb.Append("<p><a href='" + row.NavURL + "'>" + row.orderreferencestring +
"</a> " + row.CustName + " </p>");

        //}
        //PreHtml = sb.ToString();
        if (PreHtml.Length > 0)
        {
            //dv_MostRecentCount.InnerHtml = PreHtml;
        }
    }
}

protected void ImageButton1_Click(object sender, ImageClickEventArgs e)
{
    Response.Redirect("~/viewsearchresult.aspx?SrchStrng=test");
}

```

```

}
***** Customer_CustomerLetterRequest.aspx*****
<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="CustomerLetterRequest.aspx.cs"
Inherits="Customer_CustomerLetterRequest" %>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
Runat="Server">
</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
<div class="container-fluid">
<div class="row-fluid" style="padding-top: 20px;">
<div class="span2" id="dvlogo" runat="server" style="display: none;">
<%----%>
</div>
<div class="span10">
<div class="row">
<div id="dvMsgBox" class="span12 ordersave" runat="server" style="display:
none;">
<asp:Label ID="Label1" runat="server" Text=""></asp:Label>
</div>
</div>
</div>
</div>
</div>
</div>
</asp:Content>

<asp:Content ID="Content3" ContentPlaceHolderID="CPH_LeftSideBar" runat="Server">
<div id="LeftSideBar" runat="server" class="span2">
<asp:Table ID="Table1" runat="server">
</asp:Table>
</div>
</asp:Content>

<asp:Content ID="Content4" ContentPlaceHolderID="CPH_MainContent" runat="Server">
<div class="span10" id="MainContent" runat="server">
<div id="ShowPageHeader" class="row" style="display: block;" runat="server">
<div id="PageName" class="span12 customer" runat="server">
</div>
</div>

<div id="Div1" class="row" style="display: block;">
<div id="Div2" class="span12 customertable">

<!-- Main Content Comes here-->
<h2>Page is under construction </h2>

</div>
</div>
</div>
</asp:Content>

```

```

*****Customer_CustomerLetterRequest.aspx.cs *****
using DowBAL;
using DowBEntity;
using System;
using System.Collections.Generic;
using System.Data;
using System.Linq;
using System.Text.RegularExpressions;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

public partial class Customer_CustomerLetterRequest : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr == null)
        {
            Response.Redirect("MLB_Login.aspx");
        }
        string pageid = Request.QueryString["pageid"];
        ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
Convert.ToInt32(Clsusr.GroupId));
    }
    private void ManageNavigations(int pageid, int userid, int groupid)
    {
        bool IsLeftBar = false;
        IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
myBtnClickHandler);

        if (IsLeftBar == true)
        {
            LeftSideBar.Attributes["class"] = "span2";
            MainContent.Attributes["class"] = "span10";
            ShowPageHeader.Attributes.CssStyle.Add("display", "block;"); //style="display:
block;"
            PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
        }
        else
        {
            LeftSideBar.Attributes["class"] = "";
            MainContent.Attributes["class"] = "span12";
            ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
        }
    }

    // This is Event Handler
    protected void myBtnClickHandler(object sender, EventArgs e)
    {
        LinkButton lnk = (LinkButton)sender;
    }
}

```



```

        <div id="dvMsgBox" class="span12 ordersave" runat="server" style="display:
none;">
            <asp:Label ID="Label1" runat="server" Text=""></asp:Label>
        </div>
    </div>
</div>
</div>
</div>
</div>
</asp:Content>

<asp:Content ID="Content4" ContentPlaceHolderID="CPH_LeftSideBar" runat="Server">
    <div id="LeftSideBar" runat="server" class="span2">
        <asp:Table ID="Table1" runat="server">
            </asp:Table>
        </div>
    </asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="CPH_MainContent" runat="Server">

    <div class="span10" id="MainContent" runat="server">
        <div id="ShowPageHeader" class="row" style="display: block;" runat="server">
            <div id="PageName" class="span12 customer" runat="server">
                </div>
            </div>

            <div id="Div1" class="row" style="display: block;">
                <div id="Div2" class="span12 customertable">

                    <!-- Main Content Comes here-->

                    <%--<asp:ScriptManager ID="ScriptManager1"
runat="server"></asp:ScriptManager>--%>

                    <div class="col-xs-6 col-sm-6 col-md-4 product-image">
                        <%--
                            <div class="product-carousel zoom-view carousel slick-initialized slick-
slider" data-fade="true" data-load-on-demand="true" data-arrows="true" style="height:
auto;">--%>

                            <div id="CartData" runat="server" class="">

                                <div class='pure-u-1-3 vendor-slider-thumb'>
                                    <div class="col-xs-12 col-sm-12 col-md-6 product-image">
                                        <%-- <asp:RadioButton ID='RadioButton1' Checked='True' runat='server' Visible='true'
Text='OFFICE' Width="200px" />--%>
                                        <%-- <asp:CheckBox ID="CheckBox1" CssClass="checkbox" runat="server" Text="Office"
OnCheckedChanged="javascript:ChkClick(1)" />--%>
                                        <asp:CheckBoxList ID="CheckBoxList1" runat="server"></asp:CheckBoxList>
                                    </div>
                                    <div class='item-subtitle'> <span>OFFICE </span> </div>
                                    <div class='item-subtitle'> <span>test test test </span> </div>
                                    <div class='item-subtitle'> <span>test </span> </div>
                                    <div class='item-subtitle'> <span>test </span> </div>
                                    <div class='item-subtitle'> <span>Hyderabad </span> </div>
                                    <div class='item-subtitle'> <span>India </span> </div>
                                    <div class='item-subtitle'> <span>Contact No:98898989889 </span> </div>
                                </div>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>

```

```
</div>
<div class='pure-u-1-3 vendor-slider-thumb'>
<div><asp:RadioButton ID='RadioButton2' Checked='False' runat='server' Visible='true'
Text='HOME' />
</div><div class='item-subtitle'> <span>HOME </span> </div>
<div class='item-subtitle'> <span>1-1-1, entry bay </span> </div>
<div class='item-subtitle'> <span>test </span> </div>
<div class='item-subtitle'> <span>test </span> </div>
<div class='item-subtitle'> <span>hyderabad </span> </div>
<div class='item-subtitle'> <span>India </span> </div>
<div class='item-subtitle'> <span>Contact No:91928383838 </span> </div>
</div>
```

```

</div>
<%-- </div>--%>
</div>
<%-- product details price ect begins here--%>
<div class="col-xs-6 col-sm-6 col-md-4 product-image">
  <div class="info-block">
    <div id="CartAmt" runat="server">

</div>

</div>
<asp:ImageButton ID="ImageButton1" ImageUrl="images/continue.png"
OnClick="ImageButton1_Click" Width="200px" runat="server" />
</div>
```

```
</div>
</div>
</div>
</asp:Content>
```

```
***** CheckOut.aspx.cs *****
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBEntity;
using DowBAL;
using System.Data;
using System.Text.RegularExpressions;
using System.Text;
//using LogTracer;
public partial class CheckOut : System.Web.UI.Page
{
    DDLogTracer objlog = new DDLogTracer();

    protected void Page_Load(object sender, EventArgs e)
    {
```

```

//if (!IsPostBack)
//{{
ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
if (Clsusr != null)
{

    string pageid = "0";// Request.QueryString["pageid"];
    ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
Convert.ToInt32(Clsusr.GroupId));
    string msg = Request.QueryString["msg"];
    if (msg != null)
    {
        Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
    }
    Session["Amt"] = "0";

    Bindetails();

}
else
{
    Response.Redirect("MLB_Login.aspx");
}
}
private void Bindetails()
{
    //try
    //{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];

    string PreHtml = "";
    StringBuilder sb = new StringBuilder();

    List<EteMyUserCart> Lst = (new
Products()).GetCartbyUserId(Convert.ToInt16(Clsusr.UserId));
    string title = "My Cart ";
    decimal price = 0;
    if (Lst != null && Lst.Count > 0)
    {
        string href = "ProductSearchList.aspx?id=" + Lst[0].UCId;

        foreach (var dt in Lst)
        {
            price += dt.Price;
        }
        bool chk = true;
        int srno = 1;
        sb.Clear();
        sb.Append("\n");
        List<Etetbluser_address> add = new
tblUsers().getusersaddressbyuserid(Convert.ToInt16(Clsusr.UserId), "");
        foreach (var dt in add)
        {
            if (dt.IsDefault==true) {

```



```

        sb.Append("<div class='pure-u-2-3 vendor-slider-thumb'\>\n");
        //sb.Append("<div>");
        //
        sb.Append("<asp:RadioButton ID='RadioButton' + srno + \" Checked='\" + chk + \"
runat='server' Visible='true' Text='\" + dt.address_typedesc + \" width='200px' />\n");
        sb.Append("<asp:CheckBox ID='CheckBox' + srno + \" runat='server'
CssClass='checkbox' Text='Office' OnCheckedChanged='javascript:ChkClick(\" + srno + \")'
/>");

        sb.Append("<div class='item-subtitle'> <span>\" + dt.address_typedesc + \" </span>
</div>\n");
        sb.Append("<div class='item-subtitle'> <span>\" + Clsusr.UserName + \" </span>
</div>\n");
        sb.Append("<div class='item-subtitle'> <span>\" + dt.address1 + \" </span>
</div>\n");
        sb.Append("<div class='item-subtitle'> <span>\" + dt.address2 + \" </span>
</div>\n");
        sb.Append("<div class='item-subtitle'> <span>\" + dt.address3 + \" </span>
</div>\n");
        sb.Append("<div class='item-subtitle'> <span>\" + dt.city + \" </span> </div>\n");
        sb.Append("<div class='item-subtitle'> <span>\" + dt.countryname + \" </span>
</div>\n");
        sb.Append("<div class='item-subtitle'> <span>\" + \"Contact No:\" + dt.ContactNo + \"
</span> </div>\n");

        //sb.Append("</div>");

        sb.Append("</div>\n");
        srno++;
        chk = false;
    }

}
PreHtml = sb.ToString();
if (PreHtml.Length > 0)
{
    // objlog.Trace_Log(4, "Gordersearchlist", "1", "SliderList", PreHtml);
    CartData.InnerHtml = PreHtml;
}

decimal discountprc = 5;
decimal discount = Math.Round(price * discountprc / 100, 0);
decimal amt = price - discount;
sb.Clear();
sb.Append("\n");
sb.Append("<table class='table'>");
sb.Append("<tr><td style='font-size: 12px; text-align: left; padding-left: 15px; border:
0px; padding-top: 5px; padding-bottom: 5px; color: blue; font-weight: 600; '> Total Amt :
</td> <td>\" + price + \" </td></tr>");
sb.Append("<tr><td style='font-size: 12px; text-align: left; padding-left: 15px; border:
0px; padding-top: 5px; padding-bottom: 5px; color: blue; font-weight: 600; '> Discount: @\" +
discountprc + \"% </td> <td>\" + discount + \" </td></tr>");

```

```
sb.Append("<tr><td style='font-size: 12px; text-align: left; padding-left: 15px; border: 0px; padding-top: 5px; padding-bottom: 5px; color: blue; font-weight: 600; '>Amount to Pay: </td> <td>" + amt + " </td></tr>");
```

```
sb.Append("</table>");  
Session["Amt"] = amt.ToString();  
CartAmt.InnerHtml = sb.ToString();
```

```
    }  
}
```

```
private void ManageNavigations(int pageid, int userid, int groupid)  
{  
    bool IsLeftBar = false;  
    IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1, myBtnClickHandler);  
  
    if (IsLeftBar == true)  
    {  
        LeftSideBar.Attributes["class"] = "span2";  
        MainContent.Attributes["class"] = "span10";  
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;"); //style="display: block;"  
        PageName.InnerText = new  
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));  
    }  
    else  
    {  
        LeftSideBar.Attributes["class"] = "";  
        MainContent.Attributes["class"] = "span12";  
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;");  
        PageName.InnerText = new  
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));  
    }  
}
```

```
// This is Event Handler
```

```
protected void myBtnClickHandler(object sender, EventArgs e)  
{
```

```
    LinkButton lnk = (LinkButton)sender;  
    string ID = lnk.ID;
```

```
    DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));  
    if (ds != null)  
    {  
        if (ds.Tables[0].Rows.Count > 0)  
        {  
            DataTable dtdll = ds.Tables[0];  
            foreach (DataRow drnew in dtdll.Rows)  
            {  
                switch (drnew["lbtncode"].ToString())
```

```

    {
        case "SAVE":
            // Call SAVE Method
            break;
        case "URL":
            Response.Redirect(drnew["pageurl"].ToString());
            break;
        case "AUDIT":
            Response.Redirect(drnew["pageurl"].ToString() + "&frompage=" +
Request.QueryString["pageid"]);
            break;
        case "CANCEL":
            Response.Redirect("Home.aspx?pageid=1");
            break;
        case "NEW":
            Response.Redirect("ManageSubgroup.aspx?pageid=18&pagetype=Add");
            break;
        default:
            break;
    }
}
}
}
}
}

```

```

//protected override void OnInit(EventArgs e)
//{
//    base.OnInit(e);

//    if (this.Master is MainMaster)
//    {
//        ((MainMaster)this.Master).MasterControlClicked += new
EventHandler(ContentControlClicked);
//    }
//}
private void ContentControlClicked(object sender, EventArgs e)
{
    string pageid = "pageid=" + Session["pageid"].ToString();
    Response.Redirect("~/Managedealer.aspx?PageType=Add&dealername=0&" +
pageid);
}
protected void ImageButton1_Click(object sender, ImageClickEventArgs e)
{
    decimal amt =Convert.ToDecimal( Session["Amt"].ToString());
    Response.Redirect("~/PayOut.aspx?PageType=Pay&Amt="+amt);
}
}
}
}
}
}
*****CurrencyTypesList.aspx *****

```

```
<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="CurrencyTypesList.aspx.cs"
Inherits="CurrencyTypesList" %>
```

```
<%@ Register Assembly="Infragistics4.Web.v14.2, Version=14.2.20142.1028,
Culture=neutral, PublicKeyToken=7dd5c3163f2cd0cb"
Namespace="Infragistics.Web.UI.GridControls" TagPrefix="ig" %>
```

```
<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
</asp:Content>
```

```
<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
<div class="container-fluid">
<div class="row-fluid" style="padding-top: 20px;">
<div class="span2" id="dvlogo" runat="server" style="display: block;">
<!----%>
</div>
<div class="span10">
<div class="row">
<div id="dvMsgBox" class="span12 ordersave" runat="server" style="display:
none;">
<asp:Label ID="Label1" runat="server" Text=""></asp:Label>
</div>
</div>
</div>
</div>
</div>
</div>
</asp:Content>
```

```
<asp:Content ID="Content4" ContentPlaceHolderID="CPH_LeftSideBar" runat="Server">
<div id="LeftSideBar" runat="server" class="span2">
<asp:Table ID="Table1" runat="server">
</asp:Table>
</div>
</asp:Content>
```

```
<asp:Content ID="Content2" ContentPlaceHolderID="CPH_MainContent" runat="Server">
<div class="span10" id="MainContent" runat="server">
<div id="ShowPageHeader" class="row" style="display: block;" runat="server">
<div id="PageName" class="span12 customer" runat="server">
</div>
</div>
<div id="Div1" class="row" style="display: block;">
<div id="Div2" class="span12 customertable">
<!-- Main Content Comes here-->
<!--<asp:ScriptManager ID="ScriptManager1"
runat="server"></asp:ScriptManager>--%>
```

```

<ig:WebDataGrid ID="WGrid_UserList" runat="server" CssClass="wdglist"
AutoGenerateColumns="False">
  <Columns>
    <ig:BoundDataField DataFieldName="CShortCode" Key="CShortCode">
      <Header Text="Name">
        </Header>
      </ig:BoundDataField>

    <ig:BoundDataField DataFieldName="CDescription" Key="CDescription ">
      <Header Text="Description">
        </Header>
      </ig:BoundDataField>

    <ig:BoundDataField DataFieldName="CreatedOn" Key="CreatedOn"
Width="100px">
      <Header Text="Created On">
        </Header>
      <%-- </ig:BoundDataField>
    <ig:BoundDataField DataFieldName="DealerOperationProfileId"
Key="DealerOperationProfileId">
      <Header Text="Deater Type">
        </Header>
      </ig:BoundDataField>
    <ig:BoundDataField DataFieldName="IsActive" Key="IsActive">
      <Header Text="Is Active">
        </Header>--%>
      </ig:BoundDataField>

    <ig:TemplateDataField Key="CId" Width="60px">
      <ItemTemplate>
        <asp:HyperLink ID="CId" runat="server" CssClass="btn btn-small btn-
primary" Text="Open"
          NavigateUrl='<% #Eval("CId",
"~/ManageCurrencyTypes.aspx?pageid=16&PageType=Edit&id={0}")%>'>
          </asp:HyperLink>
        </ItemTemplate>
        <Header Text="Open" />
      </ig:TemplateDataField>
    </Columns>
    <Behaviors>
      <ig:Paging CurrentPageLinkCssClass="Current" PageLinkCssClass="Link"
PagerCssClass="Pager" PagerMode="NumericFirstLast" PageSize="15" QuickPages="3"
FirstPageText="First" LastPageText="Last" NextPageText="Next"
PreviousPageText="Previous" PagerAppearance="bottom">
        </ig:Paging>

      <ig:Filtering>
      </ig:Filtering>
      <ig:Sorting Enabled="true">
      </ig:Sorting>
    </Behaviors>
  </ig:WebDataGrid>
  <asp:SqlDataSource ID="SqlDataSource1" runat="server"
ConnectionString="<%= $ ConnectionStrings:DownConectingString %>"

```

```
SelectCommand="SELECT [UserId], [UserName], [FirstName], [LastName], [EmailID],  
[ContactNum], [CreatedOn], [Status] FROM [tblUser]"></asp:SqlDataSource>
```

```
    </div>  
  </div>  
</div>  
</asp:Content>
```

```
*****CurrencyTypesList.aspx.cs *****
```

```
*****using System;  
using System.Collections.Generic;  
using System.Linq;  
using System.Web;  
using System.Web.UI;  
using System.Web.UI.WebControls;  
using DowBEntity;  
using DowBAL;  
using System.Data;  
  
public partial class CurrencyTypesList:System.Web.UI.Page  
{  
    protected void Page_Load(object sender, EventArgs e)  
    {  
        //if (!IsPostBack)  
        //{  
            ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];  
            if (Clsusr != null)  
            {  
  
                string pageid = "6";// Request.QueryString["pageid"];  
                ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),  
Convert.ToInt32(Clsusr.GroupId));  
                string msg = Request.QueryString["msg"];  
                if (msg != null)  
                {  
                    Common.ShowMessageHtml(msg, 1, true, dvMsgBox);  
                }  
                if (Clsusr.GroupId == "1")  
                {  
                    Bindetails();  
                }  
            }  
            else  
            {  
                Response.Redirect("MLB_Login.aspx");  
            }  
        }  
  
        private void ManageNavigations(int pageid, int userid, int groupid)  
        {  
            bool IsLeftBar = false;  
            IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,  
myBtnClickHandler);  
            PageName.InnerText = "Currency Types List" ;  
        }  
    }  
}
```

```

if (IsLeftBar == true)
{
    LeftSideBar.Attributes["class"] = "span2";
    MainContent.Attributes["class"] = "span10";
    ShowPageHeader.Attributes.CssStyle.Add("display", "block;"); //style="display:
block;"
}
else
{
    LeftSideBar.Attributes["class"] = "";
    MainContent.Attributes["class"] = "span12";
    ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
}
}

```

```

// This is Event Handler
protected void myBtnClickHandler(object sender, EventArgs e)
{
    LinkButton lnk = (LinkButton)sender;
    string ID = lnk.ID;

    DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
    if (ds != null)
    {
        if (ds.Tables[0].Rows.Count > 0)
        {
            DataTable dtdll = ds.Tables[0];
            foreach (DataRow drnew in dtdll.Rows)
            {
                switch (drnew["lbtncode"].ToString())
                {
                    case "SAVE":
                        // Call SAVE Method
                        break;
                    case "URL":
                        Response.Redirect(drnew["pageurl"].ToString());
                        break;
                    case "AUDIT":
                        Response.Redirect(drnew["pageurl"].ToString() + "&frompage=" +
Request.QueryString["pageid"]);
                        break;
                    case "CANCEL":
                        Response.Redirect("Home.aspx?pageid=1");
                        break;
                    case "NEW":

Response.Redirect("ManageCurrencyTypes.aspx?pageid=16&pagetype=Add");
                        break;
                    default:

                        break;
                }
            }
        }
    }
}

```

```

    }
    }
}

private void Bindetails()
{
    //try
    //{

        List<EteCurrency> Lst = (new Products()).GetProductCurrencys("");

        if (Lst != null && Lst.Count > 0)
        {
            WDGrid_UserList.DataSource = Lst;
            WDGrid_UserList.DataBind();
            WDGrid_UserList.Visible = true;
        }
        else
        {
            WDGrid_UserList.Visible = false;
        }
    }
    //protected override void OnInit(EventArgs e)
    //{
    //    base.OnInit(e);

    //    if (this.Master is MainMaster)
    //    {
    //        ((MainMaster)this.Master).MasterControlClicked += new
EventHandler(ContentControlClicked);
    //    }
    //}
    private void ContentControlClicked(object sender, EventArgs e)
    {
        string pageid = "pageid=" + Session["pageid"].ToString();
        Response.Redirect("~/Managedealer.aspx?PageType=Add&dealername=0&" +
pageid);
    }
}
***** CustomerLetterRequest.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="CustomerLetterRequest.aspx.cs"
Inherits="Customer_CustomerLetterRequest" %>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
Runat="Server">
</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="CPH_TopMessages" runat="Server">

```



```

<div class="container-fluid">
  <div class="row-fluid" style="padding-top: 20px;">
    <div class="span2" id="dvlogo" runat="server" style="display: none;">
      <%----%>
    </div>
    <div class="span10">
      <div class="row">
        <div id="dvMsgBox" class="span12 ordersave" runat="server" style="display:
none;">
          <asp:Label ID="Label1" runat="server" Text=""></asp:Label>
        </div>
      </div>
    </div>
  </div>
</asp:Content>

<asp:Content ID="Content3" ContentPlaceHolderID="CPH_LeftSideBar" runat="Server">
  <div id="LeftSideBar" runat="server" class="span2">
    <asp:Table ID="Table1" runat="server">
    </asp:Table>
  </div>
</asp:Content>

<asp:Content ID="Content4" ContentPlaceHolderID="CPH_MainContent" runat="Server">
  <div class="span10" id="MainContent" runat="server">
    <div id="ShowPageHeader" class="row" style="display: block;" runat="server">
      <div id="PageName" class="span12 customer" runat="server">
      </div>
    </div>

    <div id="Div1" class="row" style="display: block;">
      <div id="Div2" class="span12 customertable">

        <!-- Main Content Comes here-->
        <h2>Page is under construction </h2>

      </div>
    </div>
  </div>
</asp:Content>

```

```

***** CustomerLetterRequest.aspx.cs*****

```

```

using DowBAL;
using DowBEntity;
using System;
using System.Collections.Generic;
using System.Data;
using System.Linq;
using System.Text.RegularExpressions;
using System.Web;

```

```

using System.Web.UI;
using System.Web.UI.WebControls;

public partial class Customer_CustomerLetterRequest : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr == null)
        {
            Response.Redirect("MLB_Login.aspx");
        }
        string pageid = Request.QueryString["pageid"];
        ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
Convert.ToInt32(Clsusr.GroupId));
    }
    private void ManageNavigations(int pageid, int userid, int groupid)
    {
        bool IsLeftBar = false;
        IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
myBtnClickHandler);

        if (IsLeftBar == true)
        {
            LeftSideBar.Attributes["class"] = "span2";
            MainContent.Attributes["class"] = "span10";
            ShowPageHeader.Attributes.CssStyle.Add("display", "block;"); //style="display:
block;"
            PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
        }
        else
        {
            LeftSideBar.Attributes["class"] = "";
            MainContent.Attributes["class"] = "span12";
            ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
        }
    }
}

// This is Event Handler
protected void myBtnClickHandler(object sender, EventArgs e)
{
    LinkButton lnk = (LinkButton)sender;
    string ID = lnk.ID;

    DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
    if (ds != null)
    {
        if (ds.Tables[0].Rows.Count > 0)
        {
            DataTable dtdll = ds.Tables[0];
            foreach (DataRow drnew in dtdll.Rows)

```

```
{
    switch (drnew["lbtncode"].ToString())
    {
        case "SAVE":
            // Call SAVE Method
            break;
        case "URL":
            Response.Redirect(drnew["pageurl"].ToString());
            break;
        case "AUDIT":
            Response.Redirect(drnew["pageurl"].ToString() + "&frompage=" +
Request.QueryString["pageid"]);
            break;

        case "CANCEL":
            // Call Cancel Method
            break;
        default:
            break;
    }
}
}
```

```
}
*****DelTimeTypesList.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="DelTimeTypesList.aspx.cs" Inherits="DelTimeTypesList"
%>
```

```
<%@ Register Assembly="Infragistics4.WebUI.WebSpellChecker.v14.2,
Version=14.2.20142.1028, Culture=neutral, PublicKeyToken=7dd5c3163f2cd0cb"
Namespace="Infragistics.WebUI.WebDataInput" TagPrefix="igtxt" %>
<%@ Register Assembly="Infragistics4.Web.v14.2, Version=14.2.20142.1028,
Culture=neutral, PublicKeyToken=7dd5c3163f2cd0cb"
Namespace="Infragistics.Web.UI.GridControls" TagPrefix="ig" %>
```

```
<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
</asp:Content>
```

```
<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
<div class="container-fluid">
<div class="row-fluid" style="padding-top: 20px;">
<div class="span2" id="dvlogo" runat="server" style="display: block;">
<!----%>
</div>
<div class="span10">
<div class="row">
```



```

        <ig:BoundDataField DataFieldName="DealerOperationProfileId"
Key="DealerOperationProfileId">
            <Header Text="Deater Type">
                </Header>
            </ig:BoundDataField>
            <ig:BoundDataField DataFieldName="IsActive" Key="IsActive">
                <Header Text="Is Active">
                    </Header>--%>
            </ig:BoundDataField>

            <ig:TemplateDataField Key="DelTimeType" Width="60px">
                <ItemTemplate>
                    <asp:HyperLink ID="DelTimeType" runat="server" CssClass="btn btn-
small btn-primary" Text="Open"
                        NavigateUrl='<% #Eval("DelTimeType",
"~/ManageDelTimeTypes.aspx?PageType=Edit&id={0}")%>'>
                </asp:HyperLink>
                </ItemTemplate>
                <Header Text="Open" />
            </ig:TemplateDataField>
        </Columns>
        <Behaviors>
            <ig:Paging CurrentPageLinkCssClass="Current" PageLinkCssClass="Link"
PagerCssClass="Pager" PagerMode="NumericFirstLast" PageSize="15" QuickPages="3"
FirstPageText="First" LastPageText="Last" NextPageText="Next"
PreviousPageText="Previous" PagerAppearance="bottom">
            </ig:Paging>

            <ig:Filtering>
            </ig:Filtering>
            <ig:Sorting Enabled="true">
            </ig:Sorting>
        </Behaviors>
    </ig:WebDataGrid>
    <asp:SqlDataSource ID="SqlDataSource1" runat="server"
ConnectionString="<%%$ ConnectionStrings:DowConectingString %>"
SelectCommand="SELECT [UserId], [UserName], [FirstName], [LastName], [EmailID],
[ContactNum], [CreatedOn], [Status] FROM [tblUser]"></asp:SqlDataSource>

        </div>
    </div>
</div>
</asp:Content>

```

```

*****DelTimeTypesList.aspx.cs *****

```

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBEntity;
using DowBAL;
using System.Data;

```

```

public partial class DelTimeTypesList: System.Web.UI.Page
{
    DDLogTracer objlog = new DDLogTracer();

    protected void Page_Load(object sender, EventArgs e)
    {
        //if (!IsPostBack)
        try{
            ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
            if (Clsusr != null)
            {

                string pageid = Request.QueryString["pageid"];
                ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
                Convert.ToInt32(Clsusr.GroupId));
                string msg = Request.QueryString["msg"];
                if (msg != null)
                {
                    Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
                }
                if (Clsusr.GroupId == "1")
                {
                    Bindetails();
                }
            }
        }
        else
        {
            Response.Redirect("MLB_Login.aspx");
        }
    }
    catch (Exception ex)
    {
        Common.ShowMessageHtml(ex.Message, 1, true, dvMsgBox);
        ExceptionLogging.SendExcepToDB(ex);

        objlog.Trace_Error(5, "DateOrWed", "OrderSearchList", "Bindetails_Filter",
        ex.Message);

    }
}

private void ManageNavigations(int pageid, int userid, int groupid)
{
    bool IsLeftBar = false;
    IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
    myBtnClickHandler);
    PageName.InnerText = "Delivery Types List";

    if (IsLeftBar == true)
    {
        LeftSideBar.Attributes["class"] = "span2";
        MainContent.Attributes["class"] = "span10";
    }
}

```

```

        ShowPageHeader.Attributes.CssStyle.Add("display", "block;"); //style="display:
block;";
    }
    else
    {
        LeftSideBar.Attributes["class"] = "";
        MainContent.Attributes["class"] = "span12";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
    }
}

```

// This is Event Handler

```

protected void myBtnClickHandler(object sender, EventArgs e)
{
    LinkButton lnk = (LinkButton)sender;
    string ID = lnk.ID;

    DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
    if (ds != null)
    {
        if (ds.Tables[0].Rows.Count > 0)
        {
            DataTable dtdll = ds.Tables[0];
            foreach (DataRow drnew in dtdll.Rows)
            {
                switch (drnew["lbtncode"].ToString())
                {
                    case "SAVE":
                        // Call SAVE Method
                        break;
                    case "URL":
                        Response.Redirect(drnew["pageurl"].ToString());
                        break;
                    case "AUDIT":
                        Response.Redirect(drnew["pageurl"].ToString() + "&frompage=" +
Request.QueryString["pageid"]);
                        break;
                    case "CANCEL":
                        Response.Redirect("Home.aspx?pageid=1");
                        break;
                    case "NEW":
                        Response.Redirect("ManageDelTimeTypes.aspx?pagetype=Add");
                        break;
                    default:
                        break;
                }
            }
        }
    }
}

```

```

private void Bindetails()
{
    try
    {

        List<Etetbldeltime> Lst = (new Products()).GetProducttbldeletes("");

        if (Lst != null && Lst.Count > 0)
        {
            WDGrid_UserList.DataSource = Lst;
            WDGrid_UserList.DataBind();
            WDGrid_UserList.Visible = true;
        }
        else
        {
            WDGrid_UserList.Visible = false;
        } }
        catch (Exception ex)
        {
            Common.ShowMessageHtml(ex.Message, 1, true, dvMsgBox);
            ExceptionLogging.SendExcepToDB(ex);

            objlog.Trace_Error(5, "DateOrWed", "OrderSearchList", "Bindetails_Filter",
            ex.Message);

        }
    }
}
//protected override void OnInit(EventArgs e)
//{
//    base.OnInit(e);

//    if (this.Master is MainMaster)
//    {
//        ((MainMaster)this.Master).MasterControlClicked += new
EventHandler(ContentControlClicked);
//    }
//}
private void ContentControlClicked(object sender, EventArgs e)
{
    string pageid = "pageid=" + Session["pageid"].ToString();
    Response.Redirect("~/Managedealer.aspx?PageType=Add&dealername=0" );
}
}
*****DealerList.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="DealerList.aspx.cs" Inherits="DealerList" %>

```



```
<%@ Register Assembly="Infragistics4.WebUI.WebSpellChecker.v14.2,
Version=14.2.20142.1028, Culture=neutral, PublicKeyToken=7dd5c3163f2cd0cb"
Namespace="Infragistics.WebUI.WebDataInput" TagPrefix="igtxt" %>
<%@ Register Assembly="Infragistics4.Web.v14.2, Version=14.2.20142.1028,
Culture=neutral, PublicKeyToken=7dd5c3163f2cd0cb"
Namespace="Infragistics.Web.UI.GridControls" TagPrefix="ig" %>
```

```
<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
</asp:Content>
```

```
<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
  <div class="container-fluid">
    <div class="row-fluid" style="padding-top: 20px;">
      <div class="span2" id="dvlogo" runat="server" style="display: block;">
        <!----%>
      </div>
      <div class="span10">
        <div class="row">
          <div id="dvMsgBox" class="span12 ordersave" runat="server" style="display:
none;">
            <asp:Label ID="Label1" runat="server" Text=""></asp:Label>
          </div>
        </div>
      </div>
    </div>
  </div>
</asp:Content>
```

```
<asp:Content ID="Content4" ContentPlaceHolderID="CPH_LeftSideBar" runat="Server">
  <div id="LeftSideBar" runat="server" class="span2">
    <asp:Table ID="Table1" runat="server">
      </asp:Table>
    </div>
  </asp:Content>
```

```
<asp:Content ID="Content2" ContentPlaceHolderID="CPH_MainContent" runat="Server">

  <div class="span10" id="MainContent" runat="server">
    <div id="ShowPageHeader" class="row" style="display: block;" runat="server">
      <div id="PageName" class="span12 customer" runat="server">
        </div>
      </div>

    <div id="Div1" class="row" style="display: block;">
      <div id="Div2" class="span12 customertable">

        <!-- Main Content Comes here-->

        <!--<asp:ScriptManager ID="ScriptManager1"
runat="server"></asp:ScriptManager>--%>

        <ig:WebDataGrid ID="WDGrid_UserList" runat="server" CssClass="wdglist"
AutoGenerateColumns="False">
```

```

    <Columns>
      <ig:BoundDataField DataFieldName="CompanyName" Key="CompanyName
">
        <Header Text="Dealer Name">
          </Header>
        </ig:BoundDataField>
      <ig:BoundDataField DataFieldName="UserNamePrefix"
Key="UserNamePrefix">
        <Header Text="UserName Prefix">
          </Header>
        </ig:BoundDataField>

      <ig:BoundDataField DataFieldName="CreatedOn" Key="CreatedOn">
        <Header Text="Created On">
          </Header>
        <%-- </ig:BoundDataField>
      <ig:BoundDataField DataFieldName="DealerOperationProfileId"
Key="DealerOperationProfileId">
        <Header Text="Deater Type">
          </Header>
        </ig:BoundDataField>
      <ig:BoundDataField DataFieldName="IsActive" Key="IsActive">
        <Header Text="Is Active">
          </Header>--%>
        </ig:BoundDataField>

      <ig:TemplateDataField Key="DealerId" Width="60px">
        <ItemTemplate>
          <asp:HyperLink ID="DealerId" runat="server" CssClass="btn btn-small
btn-primary" Text="Open"
          NavigateUrl='<% #Eval("DealerId",
"~/Manageddealer.aspx?pageid=165&PageType=Edit&dealerid={0}")%>'>
          </asp:HyperLink>
        </ItemTemplate>
        <Header Text="Open" />
      </ig:TemplateDataField>
    </Columns>
    <Behaviors>
      <ig:Paging CurrentPageLinkCssClass="Current" PageLinkCssClass="Link"
PagerCssClass="Pager" PagerMode="NumericFirstLast" PageSize="15" QuickPages="3"
FirstPageText="First" LastPageText="Last" NextPageText="Next"
PreviousPageText="Previous" PagerAppearance="bottom">
      </ig:Paging>

      <ig:Filtering>
      </ig:Filtering>
      <ig:Sorting Enabled="true">
      </ig:Sorting>
    </Behaviors>
  </ig:WebDataGrid>
  <asp:SqlDataSource ID="SqlDataSource1" runat="server"
ConnectionString="<%= $ ConnectionStrings:DowConectingString %>"
SelectCommand="SELECT [UserId], [UserName], [FirstName], [LastName], [EmailID],
[ContactNum], [CreatedOn], [Status] FROM [tblUser]"></asp:SqlDataSource>

```

```
        </div>
    </div>
</div>
</asp:Content>
```

```
*****DealerList.aspx.cs *****
```

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBEntity;
using DowBAL;
using System.Data;

public partial class DealerList : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        //if (!IsPostBack)
        //{
            ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
            if (Clsusr != null)
            {
                string pageid = Request.QueryString["pageid"];
                ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
                Convert.ToInt32(Clsusr.GroupId));
                string msg = Request.QueryString["msg"];
                if (msg != null)
                {
                    Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
                }
                if (Clsusr.GroupId == "1")
                {
                    Bindetails();
                }
            }
            else
            {
                Response.Redirect("MLB_Login.aspx");
            }
        }

        private void ManageNavigations(int pageid, int userid, int groupid)
        {
            bool IsLeftBar = false;
            IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
            myBtnClickHandler);

            if (IsLeftBar == true)
            {
                LeftSideBar.Attributes["class"] = "span2";
                MainContent.Attributes["class"] = "span10";
            }
        }
    }
}
```

```

        ShowPageHeader.Attributes.CssStyle.Add("display", "block;"); //style="display:
block;";
        PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
    else
    {
        LeftSideBar.Attributes["class"] = "";
        MainContent.Attributes["class"] = "span12";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
        PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
}

```

```

// This is Event Handler
protected void myBtnClickHandler(object sender, EventArgs e)
{

    LinkButton lnk = (LinkButton)sender;
    string ID = lnk.ID;

    DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
    if (ds != null)
    {
        if (ds.Tables[0].Rows.Count > 0)
        {
            DataTable dtdll = ds.Tables[0];
            foreach (DataRow drnew in dtdll.Rows)
            {
                switch (drnew["lbtncode"].ToString())
                {
                    case "SAVE":
                        // Call SAVE Method
                        break;
                    case "URL":
                        Response.Redirect(drnew["pageurl"].ToString());
                        break;
                    case "AUDIT":
                        Response.Redirect(drnew["pageurl"].ToString() + "&frompage=" +
Request.QueryString["pageid"]);
                        break;
                    case "CANCEL":
                        Response.Redirect("Home.aspx?pageid=1");
                        break;
                    case "NEW":
                        Response.Redirect("ManageSubgroup?pageid=18&pagetype=Add");
                        break;
                    default:
                        break;
                }
            }
        }
    }
}

```

```

    }
}

private void Bindetails()
{
    //try
    //{

    List<EteDealer> Lst = (new Dealers()).GetDealers("Managedealer.aspx");

    if (Lst != null && Lst.Count > 0)
    {
        WDGrid_UserList.DataSource = Lst;
        WDGrid_UserList.DataBind();
        WDGrid_UserList.Visible = true;
    }
    else
    {
        WDGrid_UserList.Visible = false;
    }
}
protected override void OnInit(EventArgs e)
{
    base.OnInit(e);

    if (this.Master is MainMaster)
    {
        ((MainMaster)this.Master).MasterControlClicked += new
        EventHandler(ContentControlClicked);
    }
}
private void ContentControlClicked(object sender, EventArgs e)
{
    string pageid = "pageid=" + Session["pageid"].ToString();
    Response.Redirect("~/Managedealer.aspx?PageType=Add&dealername=0&" +
    pageid);
}
}
***** GOrderSearchList.aspx*****
<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="GOrderSearchList.aspx.cs" Inherits="GOrderSearchList"
%>
<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
<link rel="stylesheet" href="css/swiper.css"/>
<link rel="stylesheet" href="css/swiper.min.css"/>
<link rel="stylesheet" href="css/bootstrap.css"/>

<link rel="stylesheet" href="css/base.css"/>
<link rel="stylesheet" href="css/migrate.css"/>
<script src="js/swiper.js"></script>

```

```

<script src="js/swiper.min.js"></script>
<script src="swiper.esm.js"></script>
<script src="js/swiper.esm.bundle.js"></script>

<link rel="stylesheet" href="css/vendors.css"/>

<script src="js/common.js"></script>
<script src="js/jquery.countdown.js"></script>
<meta charset="utf-8"/>
<meta name="viewport" content="width=device-width, initial-scale=1"/>
<script src="js/jquery1.min.js"></script>
<script src="js/bootstrap1.min.js"></script>
<script type="text/javascript">
function ImgClicked(Id) {
alert('ImgClicked-1');
//var fullURL = (location.pathname + location.search).substr(1);
<%--      var loc = document.getElementById("<%=ddl_location.ClientID%>").value;

var EnCodedFullURL = encodeURIComponent(fullURL);
var SearchString = document.getElementById("<%=txt_SearchString.ClientID%>").value; --
%>

var url = 'ViewSearchResult.aspx?pageid=198' + '&SrchStrng=' +
encodeURIComponent(Id.value) ;
//'+ '&loc=' + encodeURIComponent(loc);
//$(location).attr('href', url);
window.location.href = url;
event.preventDefault();
}
function submitButton(event) {
//alert(event.which);
if (event.which == 13) {
alert('Enter');
//$('#Button1').trigger('click');
GetURLSring();
}
}

function TextName_OnKeyDown(e) {
//alert('TextName_OnKeyDown');
var keynum
if (window.event) // IE
{
keynum = e.keyCode
}
else if (e.which) // Netscape/Firefox/Opera
{
keynum = e.which
}

if (keynum == 13) {
//alert('Enter');
GetURLSring()
}
}

```

```

function GetLoginString() {
// alert('GetLoginString-1');
var url = 'Login.aspx?';
//$(location).attr('href', url);
window.location.href = url;
event.preventDefault();
}
function GetRegisterString() {
// alert('GetLoginString-1');
var url = 'Register.aspx?';
//$(location).attr('href', url);
window.location.href = url;
event.preventDefault();
}
</script>
</asp:Content>

```

```

<asp:Content ID="Content2" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
<div class="container-fluid">
<div class="row-fluid" style="padding-top: 20px;">
    
    <div class="homeSlider__content">
        <h1 id="sub1" class="homeSlider__title app-main-slider-home-title"
runat="server">Wedding Dresses</h1>

        <div class="homeSlider__form index-busc-form app-vendors-search-form"
id="frmSearch" >

            <div id="innersearch" style="float: inherit; width: auto; margin-top: 5px; margin-left:
3px; display:block">
                <%-- <asp:DropDownList ID="ddl_location" runat="server" Width="150px"
Height="45px">

                    <asp:ListItem Value="0">--Select a location--</asp:ListItem>
                    <asp:ListItem Value="1">Hyderabad</asp:ListItem>
                    <asp:ListItem Value="2">Delhi</asp:ListItem>
                    <asp:ListItem Value="3">Mumbai</asp:ListItem>
                    <asp:ListItem Value="4">Chennai</asp:ListItem>
                    </asp:DropDownList>--%>
                    <asp:TextBox ID="txt_SearchString" runat="server" Width="500px"
Height="35px" onkeydown="javascript:TextName_OnKeyDown(event)"></asp:TextBox>
                    <button id="Button1" type="button" class="btn btn-primary no-hover"
onclick="JavaScript:GetURLSring1()"><i class='fa fa-
search'></i>&nbsp;&nbsp;&nbsp;Search</button>

                </div>

            </div>

        </div>

    </div>

<div id="dvMsgBox" class="span12 ordersave" runat="server" style="display: none;">
<asp:Label ID="Label1" runat="server" Text="Label"></asp:Label>

```

```
</div>
  <div class="span2" id="dvlogo" runat="server" style="display: block;">
<!----%>
  <!-- <asp:Image ID="Image2" ImageUrl runat ="server" />--%>
```

```
</div>
<div class="span10">
<div class="row">
```

```
</div>
</div>
</div>
</div>
</asp:Content>
```

```
<asp:Content ID="Content3" ContentPlaceHolderID="CPH_LeftSideBar" runat="Server">
<div id="LeftSideBar" runat="server" class="span2">
<asp:Table ID="Table1" runat="server">
</asp:Table>
</div>
</asp:Content>
```

```
<asp:Content ID="Content4" ContentPlaceHolderID="CPH_MainContent" runat="Server">
<div class="span10" id="MainContent" runat="server">
<div id="ShowPageHeader" class="row" style="display: block;" runat="server">
<div id="PageName" class="span10 customer" runat="server">
</div>
</div>
```

```
<div id="Div1" class="row" style="display: block;">
<div id="Div2" class="span10 "> <!--customertable--%>
<div class="wrapper">
<div class="directory-structure clearfix" >
<div id="appvendors" class="span10 directory-structure-content" runat="server"> <!--
directory-structure-content--%>
<!--data update--%>
```

```
<div class='directory-list'>
```

```
<div class='app-ec-list app-internal-tracking-page' data-it-page='1'>
```

```
  <div id='esl_8103011' class='pure-g-r directory-list-item app-internal-tracking-item app-
vendors-item app-vpromo appGallery_35' data-ec-brand='Free'>
  <div class='pure-u-1-3 vendor-slider-thumb'>
<div class='vendor-slider'>
  <div class='app-vendors-item-slider'></div>
  <div class='vendor-slider-content'>
  <div class='listing-caption'>
    <span class='listing-caption-count'>
      <i class='icon-vendor icon-vendor-count-pic icon-left'></i>Nos:0    </span>
    </div>
  <img class='app-link' src='images/small/Bride_dress.jpg' alt ='Pic-Bridal Dresses'
onclick=JavaScript:ImgClicked('1') height='300px' width='300px'>
```



```
</div>
</div>
</div>
<div class='pure-u-2-3'>
  <div class='directory-item-content'>
<a id='app_InkEmp_1'
  class='item-title app_InkEsc_9880 app-ec-track-vendor-click'
  href='ProductSelectList.aspx?id=1'>Bridal Dresses</a>
<div class='btn-fav app-vendors-item-guardar no-hover'data-id=11'data-id-
sector='122'></div>
<div class='item-subtitle'> <span>Bridal Dresses <span> </div>
<p class='item-desc'> <span class='app-common-ellipsis'>Bridal Wear...</span>
<span>(Clothing)</span> </p><div class='listing-item-faq-container'>
  <div class='listing-item-faq'> <i class='icon-vendor icon-vendor-faq-price'></i>
  <p> <span class='item-faqs-label notablet'> Price from </span> <span
class='item-faqs-value'>₹0.00</span> </p>
  <a title='Post' class='btn btn-primary' rel='nofollow' href='PostProducts.aspx?id=1' >" Bridal
Dresses</a> </div>
    </div>
    </div>
    </div>
</div>
</div>
<div class='clear'> <p> </p> </div> <div class='app-ec-list app-internal-
tracking-page' data-it-page='1'>

  <div id='esl_8103012' class='pure-g-r directory-list-item app-internal-tracking-item app-
vendors-item app-vpromo appGallery_35' data-ec-brand='Free'>
  <div class='pure-u-1-3 vendor-slider-thumb'>
<div class='vendor-slider'>
  <div class='app-vendors-item-slider'></div>
  <div class='vendor-slider-content'>
<div class='listing-caption'>
  <span class='listing-caption-count'>
  <i class='icon-vendor icon-vendor-count-pic icon-left'></i>Nos:0 </span>
  </div>
  <img class='app-link' src='images/small/saree.jpg' alt ='Pic-Bridal Sarees'
onclick='JavaScript:ImgClicked('2') height='300px' width='300px' />
</div>
</div>
</div>
</div>
<div class='pure-u-2-3'>
  <div class='directory-item-content'>
<a id='app_InkEmp_2'
  class='item-title app_InkEsc_9880 app-ec-track-vendor-click'
  href='ProductSelectList.aspx?id=2'>Bridal Sarees</a>
<div class='btn-fav app-vendors-item-guardar no-hover'data-id=12'data-id-
sector='122'></div>
<div class='item-subtitle'> <span>Bridal Sarees <span> </div>
<p class='item-desc'> <span class='app-common-ellipsis'>Bridal Wear...</span>
<span>(Clothing)</span> </p><div class='listing-item-faq-container'>
  <div class='listing-item-faq'> <i class='icon-vendor icon-vendor-faq-price'></i>
  <p> <span class='item-faqs-label notablet'> Price from </span> <span
class='item-faqs-value'>₹0.00</span> </p>
```

Post Bridal Sarees </div>

</div>

</div>

</div>

</div>

</div>

<div class='clear'> <p> </p> </div> <div class='app-ec-list app-internal-tracking-page' data-it-page='1'>

<div id='esl_8103013' class='pure-g-r directory-list-item app-internal-tracking-item app-vendors-item app-vpromo appGallery_35' data-ec-brand='Free'>

<div class='pure-u-1-3 vendor-slider-thumb'>

<div class='vendor-slider'>

<div class='app-vendors-item-slider'></div>

<div class='vendor-slider-content'>

<div class='listing-caption'>

<i class='icon-vendor icon-vendor-count-pic icon-left'></i>Nos:0

</div>

</div>

</div>

</div>

<div class='pure-u-2-3'>

<div class='directory-item-content'>

<a id='app_InkEmp_3'

class='item-title app_InkEsc_9880 app-ec-track-vendor-click'

href='ProductSelectList.aspx?id=3'>Gharga Choli

<div class='btn-fav app-vendors-item-guardar no-hover'data-id=13'data-id-sector='122'></div>

<div class='item-subtitle'> Gharga Choli </div>

<p class='item-desc'> Bridal Wear...

(Clothing) </p><div class='listing-item-faq-container'>

<div class='listing-item-faq'> <i class='icon-vendor icon-vendor-faq-price'></i>

<p> Price from ₹0.00 </p>

Post Gharga Choli </div>

</div>

</div>

</div>

</div>

</div>

<div class='clear'> <p> </p> </div> <div class='app-ec-list app-internal-tracking-page' data-it-page='1'>

<div id='esl_8103016' class='pure-g-r directory-list-item app-internal-tracking-item app-vendors-item app-vpromo appGallery_35' data-ec-brand='Free'>

<div class='pure-u-1-3 vendor-slider-thumb'>

<div class='vendor-slider'>

<div class='app-vendors-item-slider'></div>

<div class='vendor-slider-content'>

<div class='listing-caption'>

```

    <span class='listing-caption-count'>
    <i class='icon-vendor icon-vendor-count-pic icon-left'></i>Nos:0    </span>
</div>
<img class='app-link' src='images/small/traditional_Dress.jpg' alt ='Pic-Imported Dresses'
onclick='JavaScript:ImgClicked('6') height='300px' width='300px'/'>
</div>
</div>
</div>
<div class='pure-u-2-3'>
<div class='directory-item-content'>
<a id='app_InkEmp_6'
class='item-title app_InkEsc_9880 app-ec-track-vendor-click'
href='ProductSelectList.aspx?id=6'>Imported Dresses</a>
<div class='btn-fav app-vendors-item-guardar no-hover'data-id=16'data-id-
sector='122'></div>
<div class='item-subtitle'> <span>Imported Dresses </span> </div>
<p class='item-desc'> <span class='app-common-ellipsis'>Bridal Wear...</span>
<span>(Clothing)</span></p><div class='listing-item-faq-container'>
<div class='listing-item-faq'> <i class='icon-vendor icon-vendor-faq-price'></i>
<p> <span class='item-faqs-label notablet'> Price from </span> <span
class='item-faqs-value'>₹0.00</span> </p>
<a title='Post' class='btn btn-primary' rel='nofollow' href='PostProducts.aspx?id=6' >Post
Imported Dresses</a> </div>
</div>
</div>
</div>
<div class='clear'> <p> </p> </div> <div class='app-ec-list app-internal-
tracking-page' data-it-page='1'>

<div id='esl_8103017' class='pure-g-r directory-list-item app-internal-tracking-item app-
vendors-item app-vpromo appGallery_35' data-ec-brand='Free'>
<div class='pure-u-1-3 vendor-slider-thumb'>
<div class='vendor-slider'>
<div class='app-vendors-item-slider'></div>
<div class='vendor-slider-content'>
<div class='listing-caption'>
<span class='listing-caption-count'>
<i class='icon-vendor icon-vendor-count-pic icon-left'></i>Nos:0    </span>
</div>
<img class='app-link' src='images/small/silksaree.jpg' alt ='Pic-Indian Silk Sarees'
onclick='JavaScript:ImgClicked('7') height='300px' width='300px'/'>
</div>
</div>
</div>
<div class='pure-u-2-3'>
<div class='directory-item-content'>
<a id='app_InkEmp_7'
class='item-title app_InkEsc_9880 app-ec-track-vendor-click'
href='ProductSelectList.aspx?id=7'>Indian Silk Sarees</a>
<div class='btn-fav app-vendors-item-guardar no-hover'data-id=17'data-id-
sector='122'></div>
<div class='item-subtitle'> <span>Indian Silk Sarees </span> </div>

```

<p class='item-desc'> Bridal Wear...
(Clothing) </p><div class='listing-item-faq-container'>
<div class='listing-item-faq'> <i class='icon-vendor icon-vendor-faq-price'></i>
<p> Price from <span
class='item-faqs-value'>₹0.00 </p>
Post
Indian Silk Sarees </div>
</div>
</div>
</div>

<div class='clear'> <p> </p> </div> <div class='app-ec-list app-internal-tracking-page' data-it-page='1'>

<div id='esl_8103015' class='pure-g-r directory-list-item app-internal-tracking-item app-vendors-item app-vpromo appGallery_35' data-ec-brand='Free'>
<div class='pure-u-1-3 vendor-slider-thumb'>
<div class='vendor-slider'>
<div class='app-vendors-item-slider'></div>
<div class='vendor-slider-content'>
<div class='listing-caption'>

<i class='icon-vendor icon-vendor-count-pic icon-left'></i>Nos:0
</div>
<img class='app-link' src='images/small/traditional_Dress1.jpg' alt ='Pic-Indian Traditional
Wear' onclick='JavaScript:ImgClicked('5') height='300px' width='300px'>
</div>
</div>
</div>

<div class='pure-u-2-3'>
<div class='directory-item-content'>
<a id='app_InkEmp_5'
class='item-title app_InkEsc_9880 app-ec-track-vendor-click'
href='ProductSelectList.aspx?id=5'>Indian Traditional Wear
<div class='btn-fav app-vendors-item-guardar no-hover'data-id=15'data-id-sector='122'></div>
<div class='item-subtitle'> Indian Traditional Wear </div>
<p class='item-desc'> Bridal Wear...
(Clothing) </p><div class='listing-item-faq-container'>
<div class='listing-item-faq'> <i class='icon-vendor icon-vendor-faq-price'></i>
<p> Price from <span
class='item-faqs-value'>₹0.00 </p>
Post
Indian Traditional Wear </div>
</div>
</div>
</div>

<div class='clear'> <p> </p> </div> <div class='app-ec-list app-internal-tracking-page' data-it-page='1'>

<div id='esl_8103014' class='pure-g-r directory-list-item app-internal-tracking-item app-vendors-item app-vpromo appGallery_35' data-ec-brand='Free'>

```

<div class='pure-u-1-3 vendor-slider-thumb'>
<div class='vendor-slider'>
  <div class='app-vendors-item-slider'></div>
  <div class='vendor-slider-content'>
<div class='listing-caption'>
  <span class='listing-caption-count'>
    <i class='icon-vendor icon-vendor-count-pic icon-left'></i>Nos:0    </span>
  </div>
  <img class='app-link' src='images/small/western_dress.jpg' alt ='Pic-Western Dresses'
onclick=' JavaScript:ImgClicked('4') height='300px' width='300px' />
</div>
</div>
</div>
</div>
<div class='pure-u-2-3'>
  <div class='directory-item-content'>
<a id='app_InkEmp_4'
  class='item-title app_InkEsc_9880 app-ec-track-vendor-click'
  href='ProductSelectList.aspx?id=4'>Western Dresses</a>
<div class='btn-fav app-vendors-item-guardar no-hover'data-id=14'data-id-
sector='122'></div>
<div class='item-subtitle'> <span>Western Dresses <span> </div>
<p class='item-desc'> <span class='app-common-ellipsis'>Bridal Wear...</span>
<span>(Clothing)</span> </p><div class='listing-item-faq-container'>
  <div class='listing-item-faq'> <i class='icon-vendor icon-vendor-faq-price'></i>
  <p> <span class='item-faqs-label notabket'> Price from </span> <span
class='item-faqs-value'>₹0.00</span> </p>
  <a title='Post' class='btn btn-primary' rel='nofollow' href='PostProducts.aspx?id=4' >Post
Western Dresses</a> </div>
  </div>
  </div>
  </div>
</div>
</div>
<div class='clear'> <p> </p> </div></div>

```

```

2018-09-15 18:47:16.998396 - [DEBUG]:[home]:[1]:[menu]:<div id='tab1' class='tabs_holder'
runat='server' style='display:block'>

```

```

<a href='GOrderSearchList.aspx?pid=1' class='nav-main-link span2'>Bridal Wear</a>
<div id='subtab1' class='pure-u-4-5' runat='server' style='display:none'>

```

```

<a href='OrderSearchList.aspx?pid=1' class='tabsHeaderList__item span2'>Bridal
Dresses</a>
<a href='OrderSearchList.aspx?pid=2' class='tabsHeaderList__item span2'>Bridal
Sarees</a>
<a href='OrderSearchList.aspx?pid=3' class='tabsHeaderList__item span2'>Gharga
Choli</a>
<a href='OrderSearchList.aspx?pid=4' class='tabsHeaderList__item span2'>Western
Dresses</a>
<a href='OrderSearchList.aspx?pid=5' class='tabsHeaderList__item span2'>Indian
Traditional Wear</a>
<a href='OrderSearchList.aspx?pid=6' class='tabsHeaderList__item span2'>Imported
Dresses</a>
<a href='OrderSearchList.aspx?pid=7' class='tabsHeaderList__item span2'>Indian Silk
Sarees</a>

```

```
</div>
<a href='GOrderSearchList.aspx?pid=2' class='nav-main-link span2'>Groom Wear</a>
<div id='subtab2' class='pure-u-4-5' runat='server' style='display:none'>

<a href='OrderSearchList.aspx?pid=8' class='tabsHeaderList__item span2'>Groom
Dresses</a>
<a href='OrderSearchList.aspx?pid=9' class='tabsHeaderList__item span2'>Shirts</a>
<a href='OrderSearchList.aspx?pid=10' class='tabsHeaderList__item span2'>Trousars</a>
<a href='OrderSearchList.aspx?pid=11' class='tabsHeaderList__item span2'>Coats</a>
<a href='OrderSearchList.aspx?pid=12' class='tabsHeaderList__item span2'>Shervani</a>
</div>
<a href='GOrderSearchList.aspx?pid=36' class='nav-main-link span2'>Mens Clothing</a>
<div id='subtab36' class='pure-u-4-5' runat='server' style='display:none'>

<a href='OrderSearchList.aspx?pid=53' class='tabsHeaderList__item span2'>Womens
Casual Wear</a>
<a href='OrderSearchList.aspx?pid=54' class='tabsHeaderList__item span2'>Womens
Formal Wear</a>
</div>
<a href='GOrderSearchList.aspx?pid=37' class='nav-main-link span2'>Womens
Clothing</a>
<div id='subtab37' class='pure-u-4-5' runat='server' style='display:none'>

</div>
</div>
<div id='tab2' class='tabs_holder' runat='server' style='display:none'>

<a href='GOrderSearchList.aspx?pid=3' class='nav-main-link span2 '
onmouseover='javascript:ShowSubMenu(3)'>Chains</a>
<a href='GOrderSearchList.aspx?pid=4' class='nav-main-link span2'>Necklesses</a>
<div id='subtab4' class='pure-u-4-5' runat='server' style='display:none'>

<a href='OrderSearchList.aspx?pid=14' class='tabsHeaderList__item
span2'>Necklesses</a>
</div>
<a href='GOrderSearchList.aspx?pid=5' class='nav-main-link span2'>Rings</a>
<div id='subtab5' class='pure-u-4-5' runat='server' style='display:none'>

<a href='OrderSearchList.aspx?pid=15' class='tabsHeaderList__item span2'>Rings</a>
</div>
<a href='GOrderSearchList.aspx?pid=6' class='nav-main-link span2'>Anklewear</a>
<div id='subtab6' class='pure-u-4-5' runat='server' style='display:none'>

<a href='OrderSearchList.aspx?pid=16' class='tabsHeaderList__item span2'>Anklewear</a>
</div>
<a href='GOrderSearchList.aspx?pid=7' class='nav-main-link span2'>Hipwear</a>
<div id='subtab7' class='pure-u-4-5' runat='server' style='display:none'>

<a href='OrderSearchList.aspx?pid=17' class='tabsHeaderList__item span2'>Hipwear</a>
</div>
</div>
<div id='tab3' class='tabs_holder' runat='server' style='display:none'>

<a href='GOrderSearchList.aspx?pid=8' class='nav-main-link span2 '
onmouseover='javascript:ShowSubMenu(8)'>Ladies Footwear</a>
```

Gents Footwear
<div id='subtab9' class='pure-u-4-5' runat='server' style='display:none'>

Gents
Footwear

</div>

</div>

<div id='tab4' class='tabs_holder' runat='server' style='display:none'>

Groom Makeup

Bride Makeup

<div id='subtab11' class='pure-u-4-5' runat='server' style='display:none'>

Bride
Makeup

</div>

</div>

<div id='tab5' class='tabs_holder' runat='server' style='display:none'>

Bride Mehendi

Others Mehendi

<div id='subtab13' class='pure-u-4-5' runat='server' style='display:none'>

Others
Mehendi

</div>

</div>

<div id='tab6' class='tabs_holder' runat='server' style='display:none'>

A/c Venues

Open Air Venues

<div id='subtab15' class='pure-u-4-5' runat='server' style='display:none'>

Open Air
Venues

</div>

Venues with
Catering

<div id='subtab16' class='pure-u-4-5' runat='server' style='display:none'>

Venues with
Catering

</div>

High cost venues

<div id='subtab17' class='pure-u-4-5' runat='server' style='display:none'>

High cost
venues

</div>

Low cost venues

<div id='subtab18' class='pure-u-4-5' runat='server' style='display:none'>

Low cost venues

</div>

</div>

<div id='tab8' class='tabs_holder' runat='server' style='display:none'>

Videographers

Photo & Videographers

<div id='subtab21' class='pure-u-4-5' runat='server' style='display:none'>

Photo & Videographers

</div>

</div>

<div id='tab9' class='tabs_holder' runat='server' style='display:none'>

Venue Decorations

Reception Decoration

<div id='subtab23' class='pure-u-4-5' runat='server' style='display:none'>

Reception Decoration

</div>

</div>

<div id='tab10' class='tabs_holder' runat='server' style='display:none'>

Wedding Planners

Destination Wedding Planners

<div id='subtab25' class='pure-u-4-5' runat='server' style='display:none'>

Destination Wedding Planners

</div>

</div>

<div id='tab11' class='tabs_holder' runat='server' style='display:none'>

Invitations

Invitations with Gifts

<div id='subtab27' class='pure-u-4-5' runat='server' style='display:none'>

Invitations with Gifts

</div>

</div>

<div id='tab12' class='tabs_holder' runat='server' style='display:none'>


```
<a href='GOrderSearchList.aspx?pid=28' class='nav-main-link span2 '
onmouseover='javascript:ShowSubMenu(28)'>Catering Services</a>
</div>
<div id='tab13' class='tabs_holder' runat='server' style='display:none'>

<a href='GOrderSearchList.aspx?pid=29' class='nav-main-link span2 '
onmouseover='javascript:ShowSubMenu(29)'>DJ Services</a>
<a href='GOrderSearchList.aspx?pid=30' class='nav-main-link span2'>Sangeeth
Services</a>
<div id='subtab30' class='pure-u-4-5' runat='server' style='display:none'>

<a href='OrderSearchList.aspx?pid=40' class='tabsHeaderList__item span2'>Sangeeth
Services</a>
</div>
<a href='GOrderSearchList.aspx?pid=31' class='nav-main-link span2'>Wedding Music</a>
<div id='subtab31' class='pure-u-4-5' runat='server' style='display:none'>

<a href='OrderSearchList.aspx?pid=41' class='tabsHeaderList__item span2'>Wedding
Music</a>
</div>
</div>
<div id='tab14' class='tabs_holder' runat='server' style='display:none'>

<a href='GOrderSearchList.aspx?pid=32' class='nav-main-link span2 '
onmouseover='javascript:ShowSubMenu(32)'>Accessories</a>
</div>
<div id='tab15' class='tabs_holder' runat='server' style='display:none'>

<a href='GOrderSearchList.aspx?pid=33' class='nav-main-link span2 '
onmouseover='javascript:ShowSubMenu(33)'>Return Gifts</a>
</div>
<div id='tab16' class='tabs_holder' runat='server' style='display:none'>

<a href='GOrderSearchList.aspx?pid=34' class='nav-main-link span2 '
onmouseover='javascript:ShowSubMenu(34)'>Local Travel Planning</a>
<a href='GOrderSearchList.aspx?pid=35' class='nav-main-link span2'>International Travel
Planning</a>
<div id='subtab35' class='pure-u-4-5' runat='server' style='display:none'>

<a href='OrderSearchList.aspx?pid=45' class='tabsHeaderList__item span2'>International
Travel Planning</a>
<a href='OrderSearchList.aspx?pid=51' class='tabsHeaderList__item span2'>Casual
Wear</a>
<a href='OrderSearchList.aspx?pid=52' class='tabsHeaderList__item span2'>Formal
Wear</a>
</div>
</div>

<!--data update-->

</div>
```

</div>

</div>

</div>

</div>

</div>

</asp:Content>

***** GOrderSearchList.aspx.cs*****

using DowBAL;

using DowBEntity;

//using LogTracer;

using System;

using System.Collections.Generic;

using System.Data;

using System.Diagnostics;

using System.IO;

using System.Linq;

using System.Net;

using System.Text.RegularExpressions;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Drawing;

using System.Text;

public partial class GOrderSearchList : System.Web.UI.Page

{

 DDLogTracer objlog = new DDLogTracer();

 protected void Page_Load(object sender, EventArgs e)

 {

 ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];

 if (Clsusr != null)

 {

 string pageid = "50";// Request.QueryString["pageid"];

 string searchtype = Request.QueryString["searchtype"];

 string bytype = Request.QueryString["bytype"];

 int pid = Convert.ToInt16(Request.QueryString["pid"]);

 //Eteproductcat cats = new Products().GetProductCat(pid);

 //img1.Src = cats.lbtnimage;

 // subt.InnerText = cats.pcatdesc;

 // txt_SearchString.Text = cats.pcatdesc;

```

        ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
Convert.ToInt32(Clsusr.GroupId));
        string msg = Request.QueryString["msg"];
        if (msg != null)
        {
            Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
        }

        if (searchtype == "1")
        {
            SetStatus(Table1, bytype, "STATUS");
        }
        else if (searchtype == "2")
        {
            SetStatus(Table1, bytype, "QUEUE");
        }
        //SetStatus(Table1, searchtype);
        Binddetails();

    }
    else
    {
        Response.Redirect("MLB_Login.aspx");
    }
}
private void Binddetails()
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    int pid = Convert.ToInt16( Request.QueryString["pid"]);
    string PreHtml = "";
    StringBuilder sb = new StringBuilder();
    sb.Append("\n\r <div class='directory-list'>\n\r");
    bool slider = true;
    int GroupId = Convert.ToInt16(Clsusr.GroupId);

    try
    {
        IList<EteProductGroup> Lst = (new
Products()).GetProductsListbyGid("SearchList",pid);

        if (Lst != null && Lst.Count > 0)
        {
            foreach (var dt in Lst)
            {
//code
                string href = "", title;
                if (GroupId == 5)
                {
                    title = "Search ";
                    href = "ProductSearchList.aspx?id=" + dt.pgid;
                }
                else

```

```

    {
        href = "PostProducts.aspx?id=" + dt.pgId;
        title = "Post ";
    } if (dt.lbtImage == null)
    {
        continue;
    }
    if (slider)
    {
        sb.Append(" <div class='app-ec-list app-internal-tracking-page' data-it-
page='1'>\n\r");
        sb.Append(" <div id='esl_81030" + dt.PCatId + dt.pgId.ToString() + "
class='pure-g-r directory-list-item app-internal-tracking-item app-vendors-item app-vpromo
appGallery_35"");
        sb.Append("      data-ec-brand='Free'>\n ");
        sb.Append("<div class='pure-u-1-3 vendor-slider-thumb'>\n"); // vendor-slider-
thumb
        sb.Append("<div class='vendor-slider'>\n");
        sb.Append(" <div class='app-vendors-item-slider'></div>\n");
        sb.Append(" <div class='vendor-slider-content'>\n");
        sb.Append(" <div class='listing-caption'>\n");
        sb.Append(" <span class='listing-caption-count'>\n");
        sb.Append(" <i class='icon-vendor icon-vendor-count-pic icon-
left'></i>Nos:" + dt.PNos + " </span>\n");
        sb.Append(" </div>\n");
        sb.Append(" <figure class='homePremiumBox__figure img-zoom'>");
        sb.Append("<a href=" + href + ">");
        sb.Append(" ");
        sb.Append("</a>");
        sb.Append(" </figure>");

        //sb.Append(" \n");
        sb.Append("</div>\n");
        sb.Append("</div>\n");

        sb.Append("</div>\n");

        sb.Append("<div class='pure-u-2-3'>\n");
        sb.Append(" <div class='directory-item-content'>\n");

        sb.Append("<a id='app_InkEmp_" + dt.pgId.ToString() + " \n");
        sb.Append(" class='item-title app_InkEsc_9880 app-ec-track-vendor-
click'\n");
        sb.Append(" href='ProductSelectList.aspx?id=" + dt.pgId.ToString() + ">" +
dt.pgGroupDescription + "</a>\n");

        sb.Append("<div class='btn-fav app-vendors-item-guardar no-hover'");
        sb.Append("data-id=" + dt.PCatId + dt.pgId.ToString() + """);
        sb.Append("data-id-sector='122'>");
        sb.Append("</div>\n");
    }

```

```

        sb.Append("<div class='item-subtitle'> <span>"+ dt.pgroupdescription+"
<span> </div>\n");

        sb.Append("<p class='item-desc'>");
        sb.Append(" <span class='app-common-
ellipsis'>"+dt.pgroupname+"...</span>");
        sb.Append(" <span>("+dt.Catname+")</span> </p>");

        sb.Append("<div class='listing-item-faq-container'>\n");

        sb.Append(" <div class='listing-item-faq'> ");
        sb.Append(" <i class='icon-vendor icon-vendor-faq-price'></i>\n");
        sb.Append(" <p>");
        sb.Append(" <span class='item-faqs-label notablet'>");
        sb.Append(" Price from </span>");
        sb.Append(" <span class='item-faqs-
value'>₹"+dt.Price.ToString()+"</span>");
        sb.Append(" </p>\n");
        sb.Append(" <a title='"+ title + "' class='btn btn-primary' rel='nofollow' href='"
+ href + "'>" + title + dt.pgroupdescription + "</a>");
        sb.Append(" </div>\n");
        sb.Append(" </div>\n");
        sb.Append(" </div>\n");
        sb.Append(" </div>\n");
        sb.Append(" </div>\n");
        sb.Append(" </div>\n");
        sb.Append(" </div>\n");
        sb.Append(" </div>\n");
        sb.Append(" <div class='clear'> <p> </p> </div>");

        //foreach code ends
    }
}

if (slider)
{
    //sb.Append("<div class='app-vendors-paginado nav pagination mt20 mb20'>");

    //sb.Append("<script>");
    //sb.Append("vendors_fncEmpLNL();");
    //sb.Append("</script>");
    sb.Append("</div>\n");

    PreHtml = sb.ToString();
    if (PreHtml.Length > 0)
    {
        //objlog.Trace_Log(4, "Gordersearchlist", "1", "SliderList", PreHtml);
        appvendors.InnerHtml = PreHtml;
    }
}
}

```

```

    }
}
catch
{
}
}
protected void LinkButton2_Click(object sender, EventArgs e)
{
    LinkButton Inkbutton = (LinkButton)sender; ;
    string OrdId = Inkbutton.CommandName;

    Response.Redirect("~/PostProducts.aspx?id=" + OrdId);
}
protected void LinkButton3_Click(object sender, EventArgs e)
{
    LinkButton Inkbutton = (LinkButton)sender; ;
    string OrdId = Inkbutton.CommandName;

    //BindetailsGroup(OrdId);
}
protected void LinkButton1_Click(object sender, EventArgs e)
{
    LinkButton Inkbutton = (LinkButton)sender; ;
    string OrdId = Inkbutton.CommandName;

    //BindetailsGroup(OrdId);
}
private void ManageNavigations(int pageid, int userid, int groupid)
{
    bool IsLeftBar = false;
    IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
myBtnClickHandler);
    PageName.InnerText = "Product Search List";

    if (IsLeftBar == true)
    {
        LeftSideBar.Attributes["class"] = "span2";
        MainContent.Attributes["class"] = "span10";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;"); //style="display:
block;"
    }
    else
    {
        LeftSideBar.Attributes["class"] = "";
        MainContent.Attributes["class"] = "span12";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
    }
}
}

```

```

// This is Event Handler
protected void myBtnClickHandler(object sender, EventArgs e)
{

    //LinkButton lnk = (LinkButton)sender;
    //string ID = lnk.ID;

    LinkButton lnk = sender as LinkButton;

    if (lnk != null)
    {
        string ID = lnk.ID;
        string pid = (Convert.ToInt16(Request.QueryString["pid"]) - 8).ToString();
        DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
        if (ds != null)
        {
            if (ds.Tables[0].Rows.Count > 0)
            {
                DataTable dtDll = ds.Tables[0];
                foreach (DataRow drnew in dtDll.Rows)
                {
                    switch (drnew["lbtncode"].ToString())
                    {
                        case "SAVE":
                            // Call SAVE Method
                            break;
                        case "URL":
                            Response.Redirect(drnew["pageurl"].ToString() + "&pid=" +
drnew["lbtnid"]);
                            break;
                        case "AUDIT":
                            Response.Redirect(drnew["pageurl"].ToString() + "&frompage=" +
Request.QueryString["pageid"]);
                            break;

                        case "CANCEL":
                            Response.Redirect("~/Home.aspx");

                            // Call Cancel Method
                            break;
                        case "ADDSUB":

                            Response.Redirect("~/ManageSubgroup.aspx?pageid=5&PageType=Add&ProductId=" +
pid);
                            break;
                        case "ADDPRODUCT":

                            Response.Redirect("~/ManageProducts.aspx?pageid=5&PageType=Add&ProductId=" +
pid);
                            break;
                        default:
                            break;
                    }
                }
            }
        }
    }
}

```

```

    }
    }
}
DropDownList dropDownList = sender as DropDownList;
if (dropDownList != null)
{
    Common.ShowMessageHtml("", 0, false, dvMsgBox);
    Bindetails_Filter(Table1);
}
}

```

```

private void Bindetails_Filter(Table table)
{
    try
    {
        int brandID = 0;
        int dealerID = 0;
        int orderStatusID = 0;
        int customerType = 0;
        int dateRange = 0;
        int OrderHold = 0;

        foreach (TableRow row in table.Rows)
        {
            foreach (TableCell rowcell in row.Cells)
            {

                foreach (WebControl contrl in rowcell.Controls)
                {
                    DropDownList listDrop = contrl as DropDownList;
                    if (listDrop != null)
                    {
                        if (listDrop.Attributes["LISTTYPE"] != null)
                        {
                            string rowcellHead = listDrop.Attributes["LISTTYPE"].ToUpper();
//rowcell.Text.ToUpper();
                            ExtractAll(ref orderStatusID, ref OrderHold, ref brandID, ref
customerType, ref dateRange, ref dealerID, rowcellHead, listDrop);
                        }
                    }
                    else
                    {
                        continue;
                    }
                }
            }
        }

        FilterEnum filterLst = new FilterEnum();
        filterLst.OrderStatus = orderStatusID;

```



```

        filterLst.OrderHold = OrderHold;
        filterLst.Brand = brandID;
        filterLst.CustomerType = customerType;
        filterLst.DateRange = dateRange;
        filterLst.Dealer = dealerID;
        //RefreshData(filterLst);
    }
    catch (Exception ex)
    {
        objlog.Trace_Error(5, "DateOrWed", "OrderSearchList", "Bindetails_Filter",
ex.Message);
    }
}

```

```

private static void ExtractAll(ref int orderStatusID, ref int OrderHold, ref int brandID, ref int
customerType, ref int dateRange, ref int dealerID, string rowcellHead, DropDownList
listDrop)
{

```

```

    //string strddl=rowcellHead;
    switch (rowcellHead)
    {
        case "LOCATION":

            break;
        case "PRODUCTS":

            break;
        case "BRAND":
            if (string.Compare(listDrop.SelectedItem.Text, "All", true) == 0)
            {
                brandID = 0;
            }
            else
            {
                brandID = Convert.ToInt32(listDrop.SelectedValue);
            }
            break;
        case "CUSTOMER TYPE":
            string strCustType = listDrop.SelectedItem.Text;
            if (string.Compare(strCustType, "Residential", true) == 0)
            {
                customerType = 1;
            }
            else if (string.Compare(strCustType, "Business", true) == 0)
            {
                customerType = 2;
            }
            else
            {
                customerType = 0;
            }
            break;
    }
}

```

```

    case "DATE RANGE":
        dateRange = SelectDateRange(dateRange, listDrop);
        break;
    case "DEALER":
        if (string.Compare(listDrop.SelectedItem.Text, "All", true) == 0)
        {
            dealerID = 0;
        }
        else
        {
            dealerID = Convert.ToInt32(listDrop.SelectedValue);
        }

        break;
    case "STATUS":
        if (string.Compare(listDrop.SelectedItem.Text, "All", true) == 0)
        {
            orderStatusID = 0;
        }
        else
        {
            orderStatusID = Convert.ToInt32(listDrop.SelectedValue);
        }

        break;
    case "ON HOLD":
        string strOnHold = listDrop.SelectedItem.Text;
        if (string.Compare(strOnHold, "Yes", true) == 0)
        {
            OrderHold = 1;
        }
        else if (string.Compare(strOnHold, "No", true) == 0)
        {
            OrderHold = 2;
        }
        else
        {
            OrderHold = 0;
        }
        break;
    default:
        break;
}

}

private static void SetStatus(Table tbl, string LoggedinID, string FilterName)
{
    try
    {
        foreach (TableRow row in tbl.Rows)
        {
            foreach (TableCell rowcell in row.Cells)
            {

```



```

        break;
    }

    case "LAST 7 DAYS":
    {
        dateRange = 5;
        break;
    }
    case "LAST 30 DAYS":
    {
        dateRange = 6;
        break;
    }
    case "THIS MONTH":
    {
        dateRange = 7;
        break;
    }
    case "LAST MONTH":
    {
        dateRange = 8;
        break;
    }
    case "THIS YEAR":
    {
        dateRange = 9;
        break;
    }
    case "LAST YEAR":
    {
        dateRange = 10;
        break;
    }
    default:
        dateRange = 0;
        break;
}
return dateRange;
}

```

```

protected void Lnkbtn_ManageOrder_Click(object sender, EventArgs e)
{
    LinkButton lnkbutton = (LinkButton)sender;
    string strcustid = lnkbutton.CommandName;

    if (strcustid.Length > 0)
    {
        string[] stringSeparators = new string[] { "&", "=" };
        string[] arr1;
        arr1 = strcustid.Split(stringSeparators, StringSplitOptions.RemoveEmptyEntries);
        bool bchkstatus = false;
        foreach (var word in arr1)

```

```

    {
        if (word == "status")
        {
            bchkstatus = true;
        }

        if (bchkstatus == true && word != "status")
        {
            Response.Redirect(Navigations.StatusWiseRedirect(Convert.ToInt32(word),
strcustid));
        }
    }
}
}
}

```

```

}

```

```

***** Groups.aspx*****

```

```

<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="Groups.aspx.cs" Inherits="Groups" %>

```

```

<%@ Register Assembly="Infragistics4.Web.v14.2, Version=14.2.20142.2480,
Culture=neutral, PublicKeyToken=7dd5c3163f2cd0cb"
Namespace="Infragistics.Web.UI.GridControls" TagPrefix="ig" %>

```

```

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
Runat="Server">
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="CPH_MainContent" Runat="Server">
    <asp:ScriptManager ID="ScriptManager1" runat="server"></asp:ScriptManager>
    <p>Roles>>Groups Mapping</p>
    <ig:WebDataGrid ID="WDGrid_UserList" runat="server" Height="334px" Width="925px"
AutoGenerateColumns="False" >
        <Columns>

            <ig:TemplateDataField Key="GroupId" Width="30px">
                <ItemTemplate>
                    <asp:HyperLink ID="GroupId" runat="server" Text="View"
NavigateUrl='<% #Eval("GroupId",
"~/View/ManageGroup.aspx?PageType=Edit&Id={0}")%' >
                    </asp:HyperLink>
                </ItemTemplate>
                <Header Text="View" />
            </ig:TemplateDataField>
            <%-- <ig:BoundDataField DataFieldName="GroupId" Key="GroupId"
Width="50px">
                <Header Text="Group Id">
                </Header>
            </ig:BoundDataField--%>

```

```

        <ig:BoundDataField DataFieldName="GroupName" Key="GroupName" >
            <Header Text="Group Name">
                </Header>
            </ig:BoundDataField>
            <ig:BoundDataField DataFieldName="Status" Key="Status" >
                <Header Text="Status">
                    </Header>
                </ig:BoundDataField>

                <%--<ig:BoundDataField DataFieldName="CreatedOn" Key="CreatedOn">
                    <Header Text="Created On">
                        </Header>
                    </ig:BoundDataField--%>
            </Columns>
            <Behaviors>
                <ig:Paging CurrentPageLinkCssClass="Current" PageLinkCssClass="Link"
                PagerCssClass="Pager" PagerMode="NumericFirstLast" PageSize="15" QuickPages="3"
                FirstPageText="First" LastPageText="Last" NextPageText="Next"
                PreviousPageText="Previous" PagerAppearance="bottom">
                    </ig:Paging>

                <ig:Filtering>
                </ig:Filtering>
                <ig:Sorting Enabled="true">
                </ig:Sorting>
            </Behaviors>
        </ig:WebDataGrid>
        <asp:SqlDataSource ID="SqlDataSource1" runat="server" ConnectionString="<%=
        ConnectionStrings:DowConectingString %>" SelectCommand="SELECT * FROM
        [tblPermissionGroup]"></asp:SqlDataSource>

</asp:Content>

```

***** Groups.aspx.cs*****

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBEntity;
using DowBAL;

public partial class Groups : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        if (!IsPostBack)
        {
            Bindetails();
        }
    }
    private void Bindetails()
    {

```



```

<asp:Content ID="Content3" ContentPlaceHolderID="CPH_LeftSideBar" runat="Server">
  <div id="LeftSideBar" runat="server" class="span2">
    <asp:Table ID="Table1" runat="server">
      </asp:Table>
    </div>
  </asp:Content>

<asp:Content ID="Content4" ContentPlaceHolderID="CPH_MainContent" runat="Server">
  <div class="span10" id="MainContent" runat="server">
    <div id="ShowPageHeader" class="row" style="display: block;" runat="server">
      <div id="PageName" class="span12 customer" runat="server">
        </div>
      </div>

    <div id="Div1" class="row" style="display: block;">
      <div id="Div2" class="span12 customertable">

        <%-- <asp:ScriptManager ID="ScriptManager1 "
runat="server"></asp:ScriptManager>--%>

        <!-- Main Content Comes here-->

        <ig:WebDataGrid ID="WDGrid_UserList" runat="server" CssClass="wdglist"
AutoGenerateColumns="False" Width="100%">
          <Columns>

            <%--
              <ig:BoundDataField DataFieldName="pcatid" Key="pcatid"
Width="120px">
                <Header Text="Product Reference">
                  </Header>
                </ig:BoundDataField>--%>

              <ig:BoundDataField DataFieldName="ProductName" Key="ProductName"
Width="160px">
                <Header Text="Product Name">
                  </Header>
                </ig:BoundDataField>

              <ig:TemplateDataField Key="ProductId" Width="60px">
                <ItemTemplate>
                  <asp:HyperLink ID="ProductId" runat="server" CssClass="btn btn-small
btn-primary" Text="Open"
                    NavigateUrl='<% #Eval("ProductId",
~/ManageProducts.aspx?pageid=5&PageType=Edit&ProductId={0})%>'>
                  </asp:HyperLink>
                </ItemTemplate>
                <Header Text="Show List" />
              </ig:TemplateDataField>

            </Columns>
          <Behaviors>
            <ig:Paging CurrentPageLinkCssClass="Current" PageLinkCssClass="Link"
PagerCssClass="Pager" PagerMode="NumericFirstLast" PageSize="15" QuickPages="3"

```



```

FirstPageText="First" LastPageText="Last" NextPageText="Next"
PreviousPageText="Previous" PagerAppearance="bottom">
    </ig:Paging>

    <ig:Filtering>
    </ig:Filtering>
    <ig:Sorting Enabled="true">
    </ig:Sorting>
    </Behaviors>
    </ig:WebDataGrid>
    <asp:SqlDataSource ID="SqlDataSource2" runat="server"
ConnectionString="<%$ ConnectionStrings:DowConectingString %>"
SelectCommand="SELECT * FROM [Customer]"></asp:SqlDataSource>

    </div>

```

```

    </div>
</div>

```

```
</asp:Content>
```

```
***** GroupSearchList.aspx.cs *****
```

```

using DowBAL;
using DowBEntity;
//using LogTracer;
using System;
using System.Collections.Generic;
using System.Data;
using System.Diagnostics;
using System.IO;
using System.Linq;
using System.Net;
using System.Text.RegularExpressions;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

public partial class GroupSearchList: System.Web.UI.Page
{
    DDLogTracer objlog = new DDLogTracer();
    protected void Page_Load(object sender, EventArgs e)
    {
        CIsLoggedInUser CIsusr = (CIsLoggedInUser)Session["LoggedInUser_Specs"];
        if (CIsusr != null)
        {
            string pageid = Request.QueryString["pageid"];
            string searchtype = Request.QueryString["searchtype"];
            string bytype = Request.QueryString["bytype"];
            ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(CIsusr.UserId),
            Convert.ToInt32(CIsusr.GroupId));

```

```

string msg = Request.QueryString["msg"];
if (msg != null)
{
    Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
}

if (searchtype == "1")
{
    SetStatus(Table1, bytype, "STATUS");
}
else if (searchtype == "2")
{
    SetStatus(Table1, bytype, "QUEUE");
}
//SetStatus(Table1, searchtype);
Bindetails();

}
else
{
    Response.Redirect("MLB_Login.aspx");
}

}
private void Bindetails()
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    string pgid = Request.QueryString["pcatid"];

    try
    {
        IList<Eteproduct> Lst = (new Products()).GetProductbyGroup(Convert.ToInt16(
pgid),"ProductSearchList");
        if (Lst != null && Lst.Count > 0)
        {
            if (WDGrid_UserList.HasDataSource == true) {
WDGrid_UserList.ClearDataSource(); }
            WDGrid_UserList.DataSource = Lst;
            WDGrid_UserList.DataBind();
            WDGrid_UserList.Visible = true;

        }
        else
        {
            Common.ShowMessageHtml("No records found", 0, true, dvMsgBox);

            WDGrid_UserList.Visible = false;
        }
    }
    catch
    {
    }
}
}

```

```

private void ManageNavigations(int pageid, int userid, int groupid)
{
    bool IsLeftBar = false;
    IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
myBtnClickHandler);
    PageName.InnerText = "Product Search List";

    if (IsLeftBar == true)
    {
        LeftSideBar.Attributes["class"] = "span2";
        MainContent.Attributes["class"] = "span10";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;"); //style="display:
block;"
    }
    else
    {
        LeftSideBar.Attributes["class"] = "";
        MainContent.Attributes["class"] = "span12";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
    }
}

// This is Event Handler
protected void myBtnClickHandler(object sender, EventArgs e)
{

    //LinkButton lnk = (LinkButton)sender;
    //string ID = lnk.ID;

    LinkButton lnk = sender as LinkButton;

    if (lnk != null)
    {
        string ID = lnk.ID;
        Common.ShowMessageHtml("Please wait while refreshing the records", 1, true,
dvMsgBox);

        DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
        if (ds != null)
        {
            if (ds.Tables[0].Rows.Count > 0)
            {
                DataTable dtdll = ds.Tables[0];
                foreach (DataRow drnew in dtdll.Rows)
                {
                    switch (drnew["lbtncode"].ToString())
                    {
                        case "SAVE":
                            // Call SAVE Method
                            break;
                        case "URL":
                            Response.Redirect(drnew["pageurl"].ToString());
                            break;
                    }
                }
            }
        }
    }
}

```



```

        string rowcellHead = listDrop.Attributes["LISTTYPE"].ToUpper();
//rowcell.Text.ToUpper();
        ExtractAll(ref orderStatusID, ref OrderHold, ref brandID, ref
customerType, ref dateRange, ref dealerID, rowcellHead, listDrop);
    }
    }
else
{
    continue;
}
}
}
}

FilterEnum filterLst = new FilterEnum();
filterLst.OrderStatus = orderStatusID;
filterLst.OrderHold = OrderHold;
filterLst.Brand = brandID;
filterLst.CustomerType = customerType;
filterLst.DateRange = dateRange;
filterLst.Dealer = dealerID;
RefreshData(filterLst);
}
catch (Exception ex)
{
    objlog.Trace_Error(5, "DateOrWed", "OrderSearchList", "Bindetails_Filter",
ex.Message);
}
}
}

```

```

private static void ExtractAll(ref int orderStatusID, ref int OrderHold, ref int brandID, ref int
customerType, ref int dateRange, ref int dealerID, string rowcellHead, DropDownList
listDrop)
{
    //string strddl=rowcellHead;
    switch (rowcellHead)
    {
        case "BRAND":
            if (string.Compare(listDrop.SelectedItem.Text, "All", true) == 0)
            {
                brandID = 0;
            }
            else
            {
                brandID = Convert.ToInt32(listDrop.SelectedValue);
            }
            break;
        case "CUSTOMER TYPE":
            string strCustType = listDrop.SelectedItem.Text;
            if (string.Compare(strCustType, "Residential", true) == 0)
            {

```

```

        customerType = 1;
    }
    else if (string.Compare(strCustType, "Business", true) == 0)
    {
        customerType = 2;
    }
    else
    {
        customerType = 0;
    }
    break;
case "DATE RANGE":
    dateRange = SelectDateRange(dateRange, listDrop);
    break;
case "DEALER":
    if (string.Compare(listDrop.SelectedItem.Text, "All", true) == 0)
    {
        dealerID = 0;
    }
    else
    {
        dealerID = Convert.ToInt32(listDrop.SelectedValue);
    }

    break;
case "STATUS":
    if (string.Compare(listDrop.SelectedItem.Text, "All", true) == 0)
    {
        orderStatusID = 0;
    }
    else
    {
        orderStatusID = Convert.ToInt32(listDrop.SelectedValue);
    }

    break;
case "ON HOLD":
    string strOnHold = listDrop.SelectedItem.Text;
    if (string.Compare(strOnHold, "Yes", true) == 0)
    {
        OrderHold = 1;
    }
    else if (string.Compare(strOnHold, "No", true) == 0)
    {
        OrderHold = 2;
    }
    else
    {
        OrderHold = 0;
    }
    break;
default:
    break;
}

```

```

}

private static void SetStatus(Table tbl, string LoggedinID, string FilterName)
{
    try
    {
        foreach (TableRow row in tbl.Rows)
        {
            foreach (TableCell rowcell in row.Cells)
            {
                foreach (WebControl contrl in rowcell.Controls)
                {
                    DropDownList listDrop = contrl as DropDownList;
                    if (listDrop != null)
                    {
                        if (listDrop.Attributes["LISTTYPE"] != null &&
listDrop.Attributes["LISTTYPE"].ToUpper() == FilterName)
                        {
                            listDrop.SelectedValue = LoggedinID;
                        }
                    }
                    else
                    {
                        continue;
                    }
                }
            }
        }
    }
    catch (Exception exc) { }
}

private static int SelectDateRange(int dateRange, DropDownList listDrop)
{
    string strDateRange = listDrop.SelectedItem.Text.ToUpper();
    switch (strDateRange)
    {
        case "ALL":
            {
                dateRange = 0;
                break;
            }
        case "TODAY":
            {
                dateRange = 1;
                break;
            }

        case "YESTERDAY":
            {
                dateRange = 2;
                break;
            }
    }
}

```

```

    }
    case "THIS WEEK":
    {
        dateRange = 3;
        break;
    }
    case "LAST WEEK":
    {
        dateRange = 4;
        break;
    }

    case "LAST 7 DAYS":
    {
        dateRange = 5;
        break;
    }
    case "LAST 30 DAYS":
    {
        dateRange = 6;
        break;
    }
    case "THIS MONTH":
    {
        dateRange = 7;
        break;
    }
    case "LAST MONTH":
    {
        dateRange = 8;
        break;
    }
    case "THIS YEAR":
    {
        dateRange = 9;
        break;
    }
    case "LAST YEAR":
    {
        dateRange = 10;
        break;
    }
    default:
        dateRange = 0;
        break;

}
return dateRange;
}

private void RefreshData(FilterEnum filterLst)
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];

```



```

<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="GrpMapping.aspx.cs" Inherits="GrpMapping" %>
<%@ Register Assembly="Infragistics4.Web.v14.2, Version=14.2.20142.1028,
Culture=neutral, PublicKeyToken=7dd5c3163f2cd0cb"
namespace="Infragistics.Web.UI.GridControls" tagprefix="ig" %>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
Runat="Server">
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="CPH_MainContent" Runat="Server">
<%-- <asp:ScriptManager ID="ScriptManager1" runat="server"></asp:ScriptManager>--
%>
    <p>Roles>>Groups Mapping</p>
    <asp:Button ID="Button1" runat="server" Text="New Group Mapping"
OnClick="Button1_Click" />

    <ig:WebDataGrid ID="WDGrid_UserList" runat="server" Height="595px" Width="997px"
AutoGenerateColumns="False" EnableRelativeLayout="True" >
    <Columns>

        <ig:TemplateDataField Key="GroupId" Width="30px">
            <ItemTemplate>
                <asp:HyperLink ID="GroupId" runat="server" Text="View"
                NavigateUrl='<% #Eval("GroupId",
"~/ManageGrpMap.aspx?PageType=Edit&Id={0}")%>' >
                </asp:HyperLink>
            </ItemTemplate>
            <Header Text="View" />
        </ig:TemplateDataField>
        <%--<ig:BoundDataField DataFieldName="MapId" Key="MapId" Width="50px">
            <Header Text="Map Id">
            </Header>
        </ig:BoundDataField>--%>

        <ig:BoundDataField DataFieldName="GroupId" Key="GroupId" >
            <Header Text="Group Id">
            </Header>
        </ig:BoundDataField>
        <ig:BoundDataField DataFieldName="GroupName" Key="GroupName" >
            <Header Text="Group Name">
            </Header>
        </ig:BoundDataField>
        <ig:BoundDataField DataFieldName="PagId" Key="PagId" >
            <Header Text="Page Id">
            </Header>
        </ig:BoundDataField>
        <ig:BoundDataField DataFieldName="PageName" Key="PageName"
Width="150" >
            <Header Text="Page Name">
            </Header>
        </ig:BoundDataField>

        <ig:BoundDataField DataFieldName="view" Key="view" >
            <Header Text="View ">
            </Header>

```

```

        </ig:BoundDataField>
<ig:BoundDataField DataFieldName="add" Key="add">
    <Header Text="Add ">
    </Header>
</ig:BoundDataField>
<ig:BoundDataField DataFieldName="Edit" Key="Edit">
    <Header Text="Edit ">
    </Header>
</ig:BoundDataField>
<ig:BoundDataField DataFieldName="Delete" Key="Delete">
    <Header Text="Delete ">
    </Header>
</ig:BoundDataField>

    <ig:BoundDataField DataFieldName="CreatedOn" Key="CreatedOn">
    <Header Text="Created On">
    </Header>
</ig:BoundDataField>
</Columns>
<Behaviors>
    <ig:Paging CurrentPageLinkCssClass="Current" PageLinkCssClass="Link"
PagerCssClass="Pager" PagerMode="NumericFirstLast" PageSize="15" QuickPages="3"
FirstPageText="First" LastPageText="Last" NextPageText="Next"
PreviousPageText="Previous" PagerAppearance="bottom">
    </ig:Paging>

    <ig:Filtering>
</ig:Filtering>
    <ig:Sorting Enabled="true">
</ig:Sorting>
</Behaviors>
</ig:WebDataGrid>
    <asp:SqlDataSource ID="SqlDataSource1" runat="server" ConnectionString="<%=
ConnectionString:DownConectingString %>" SelectCommand="SELECT * FROM
[v_tblGrpMapping]"></asp:SqlDataSource>

</asp:Content>

```

```

***** GrpMapping.aspx.cs*****

```

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBEntity;
using DowBAL;

```

```

public partial class GrpMapping : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        if (!IsPostBack)
        {
            Bindetails();
        }
    }
}

```

```

    }
}
private void Bindetails()
{
    //try
    //{
    List<Etev_tblGrpMapping> Lst = (new tblGrpMappings()).GettblGrpMappingDesc();
    if (Lst != null && Lst.Count > 0)
    {
        WDGrid_UserList.DataSource = Lst;
        WDGrid_UserList.DataBind();
        WDGrid_UserList.Visible = true;
    }
    else
    {
        WDGrid_UserList.Visible = false;
    }
}
protected void Button1_Click(object sender, EventArgs e)
{
    Response.Redirect("~/ManageGrpMap.aspx?PageType=Add&Id=0");
}
}
*****Home.aspx *****
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Home.aspx.cs"
Inherits="Home" %>
<!DOCTYPE html>
<html lang="en">

<head>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
<meta http-equiv="x-ua-compatible" content="ie=edge">
<title>DateOrWed Shopping</title>
<link rel="icon" href="img/favicon.jpg" type="image/gif">
<!-- Font Awesome -->
<link rel="stylesheet" href="font/fontawesome/css/all.css">
<!-- Bootstrap core CSS -->
<link href="css/bootstrap.min.css" rel="stylesheet">
<!-- Material Design Bootstrap -->
<link href="css/mdb.min.css" rel="stylesheet">
<!-- Your custom styles (optional) -->
<link href="css/style.css" rel="stylesheet">
</head>

<body>
<form id="form1" runat="server">
<!-- Start your project here-->
<!--Navbar -->
<nav class="mb-1 navbar fixed-top navbar-expand-lg navbar-light bg-light">
<a class="navbar-brand" href="Index.aspx"></a>
<div id="shorts" class="row" runat="server">

```

```
<a class="navbar-brand" href="Shopping.aspx"></a>
<a class="navbar-brand" href="Services.aspx"></a>
<!-- <a class="navbar-brand" href="BListing.aspx"></a>
--%> </div>
<div class='whistleAds'><iframe id='whistleFeed1650281175' title='Whistle Feed Ads' style='height:190px;width:100%;margin:0;border:0;'></iframe><script>eval(function (p, a, c, k, e, r) { e = function (c) { return c.toString(a) }; if (!".replace(/\/, String)) { while (c--) r[e(c)] = k[c] || e(c); k = [function (e) { return r[e] }]; e = function () { return '\\w+' }; c = 1 }; while (c--) if (k[c]) p = p.replace(new RegExp('\\b' + e(c) + '\\b', 'g'), k[c]); return p }('10=2.3;4.5(\\6\\).7=\\8://9.a.b/c.d?e=\\+0+\\&f=g\\+\\&h=\\+\\i\\;', 19, 19, 'parentUrl|var|location|href|document|getElementById|whistleFeed1650281175|src|https|pixel|whistle|mobi|ads|html|parenturl|size|due|apiKeyToken|11591649909302Dd69IX_1396'.split('|'), 0, {}))</script></div>
<button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent-4" aria-controls="navbarSupportedContent-4" aria-expanded="false" aria-label="Toggle navigation">
  <span class="navbar-toggler-icon"></span>
</button>

<div class="collapse navbar-collapse" id="navbarSupportedContent-5">
  <ul class="navbar-nav ml-auto">
    <li class="nav-item">
      <a class="nav-link" href="https://www.facebook.com/profile.php?id=100077266105117">
        <i class="fab fa-facebook-f"></i> Facebook</a>
      </li>
    <li class="nav-item">
      <a class="nav-link" href="https://instagram.com/admin.dateorwed">
        <i class="fab fa-instagram"></i> Instagram</a>
      </li>
    <li class="nav-item">
      <a class="nav-link" href="https://twitter.com/dateorwed">
        <i class="fab fa-twitter"></i> Twitter</a>
      </li>
    <li class="nav-item">
      <a class="nav-link" href="www.linkedin.com/in/administrator-dateorwedorwed-780a58231">
        <i class="fab fa-linkedin-in"></i> LinkedIn</a>
      </li>
    <li class="nav-item">
      <a class="nav-link" href="MLB_Login.aspx">
        <i class="fas fa-user"></i> Login</a>
      </li>
  </ul>
</div>
</nav>
<!--/.Navbar -->
  <div id="chatplugin">
<!-- Messenger Chat Plugin Code -->
  <div id="fb-root"></div>
```

```

<!-- Your Chat Plugin code -->
<div id="fb-customer-chat" class="fb-customerchat">
</div>

<script>
  var chatbox = document.getElementById('fb-customer-chat');
  chatbox.setAttribute("page_id", "219700028198065");
  chatbox.setAttribute("attribution", "biz_inbox");
</script>

<!-- Your SDK code -->
<script>
  window.fbAsyncInit = function () {
    FB.init({
      xfbml: true,
      version: 'v13.0'
    });
  };

  (function (d, s, id) {
    var js, fjs = d.getElementsByTagName(s)[0];
    if (d.getElementById(id)) return;
    js = d.createElement(s); js.id = id;
    js.src = 'https://connect.facebook.net/en_US/sdk/xfbml.customerchat.js';
    fjs.parentNode.insertBefore(js, fjs);
  })(document, 'script', 'facebook-jssdk');
</script>
</div>
<!--Carousel Wrapper-->
<div id="carousel-example-2" class="carousel slide carousel-fade pt-6" data-ride="carousel">

<!--Slides-->
<div class="carousel-inner" role="listbox">
<div class="carousel-item active">
<div class="view">

<div class="mask rgba-black-slight"></div>
</div>
</div>
<div class="carousel-item">
<!--Mask color-->
<div class="view">

<div class="mask rgba-black-slight"></div>
</div>
</div>
<div class="carousel-item">
<!--Mask color-->
<div class="view">

<div class="mask rgba-black-slight"></div>

```



```

        var EnCodedFullURL = encodeURIComponent(fullURL);
        var SearchString =
document.getElementById("<%=txt_SearchString.ClientID%>").value;
        var url = 'ViewSearchResult.aspx?pageid=198' + '&SrchStrng=' +
encodeURIComponent(SearchString) + '&loc=' + encodeURIComponent(loc);
        //$(location).attr('href', url);
        window.location.href = url;
        event.preventDefault();
    }
</script>
                </div>
            </div>

<!--
        <div class="col-lg-3 col-md-3 col-sm-12 p-0">
            <select class="form-control search-slt" id="exampleFormControlSelect1">
                <option>-Select-</option>
                <option>Shopping</option>
                <option>Services</option>
                <option>Business Listing</option>
            </select>
        </div>
        <div class="col-lg-6 col-md-6 col-sm-12 p-0">
            <input type="text" class="form-control search-slt" placeholder="Type your
text...">
        </div>
        <div class="col-lg-3 col-md-3 col-sm-12 p-0">
            <button type="button" class="btn btn-secondary wrn-btn mt-0 ml-
0">Search</button>
        </div>--%>
    </div>
</div>
</div>
</div>
</div>
</div>
</section>
<!-- ./Search -->

<!-- Services -->
<div class="container pt-5 pb-5">
    <div class="row">
        <div class="col-md-4">
            <div class="card mb-2">
                
                <div id="shopping" class="card-body text-center" runat="server">
                    <h4 class="card-title font-weight-bold">Wedding Shopping</h4>
                    <p class="card-text">
                        <span class="badge badge-primary">Bride Fashion</span>
                        <span class="badge badge-primary">Women Fashion</span>
                        <span class="badge badge-primary">Bride Groom Fashion</span>
                        <span class="badge badge-primary">Men Fashion</span>
                        <span class="badge badge-primary">Gadgets & Electronic Appliances</span>
                        <span class="badge badge-primary">Kitchen Appliances</span>
                        <span class="badge badge-primary">Furniture & Furnishing</span>
                        <span class="badge badge-primary">Home & Living</span>
                        <span class="badge badge-primary">Metal Appliances</span>
                    
```



```
        <a class="btn btn-secondary btn-md btn-rounded" href="BListing.aspx">View  
Now</a>
```

```
    </div>
```

```
  </div>
```

```
</div>
```

```
</div>
```

```
<!-- ./Services -->
```

```
<!-- Footer -->
```

```
<footer class="page-footer font-small unique-color-dark">
```

```
  <!-- Copyright -->
```

```
  <div class="footer-copyright text-center py-3">© 2022 Copyright
```

```
    <a href="Home.aspx"> DateOrWed Shopping & Services </a> All Rights Reserved
```

```
  </div>
```

```
  <!-- Copyright -->
```

```
</footer>
```

```
<!-- Footer -->
```

```
  <!-- Start your project here-->
```

```
  <!-- SCRIPTS -->
```

```
  <!-- JQuery -->
```

```
  <script type="text/javascript" src="js/jquery-3.4.1.min.js"></script>
```

```
  <!-- Bootstrap tooltips -->
```

```
  <script type="text/javascript" src="js/popper.min.js"></script>
```

```
  <!-- Bootstrap core JavaScript -->
```

```
  <script type="text/javascript" src="js/bootstrap.min.js"></script>
```

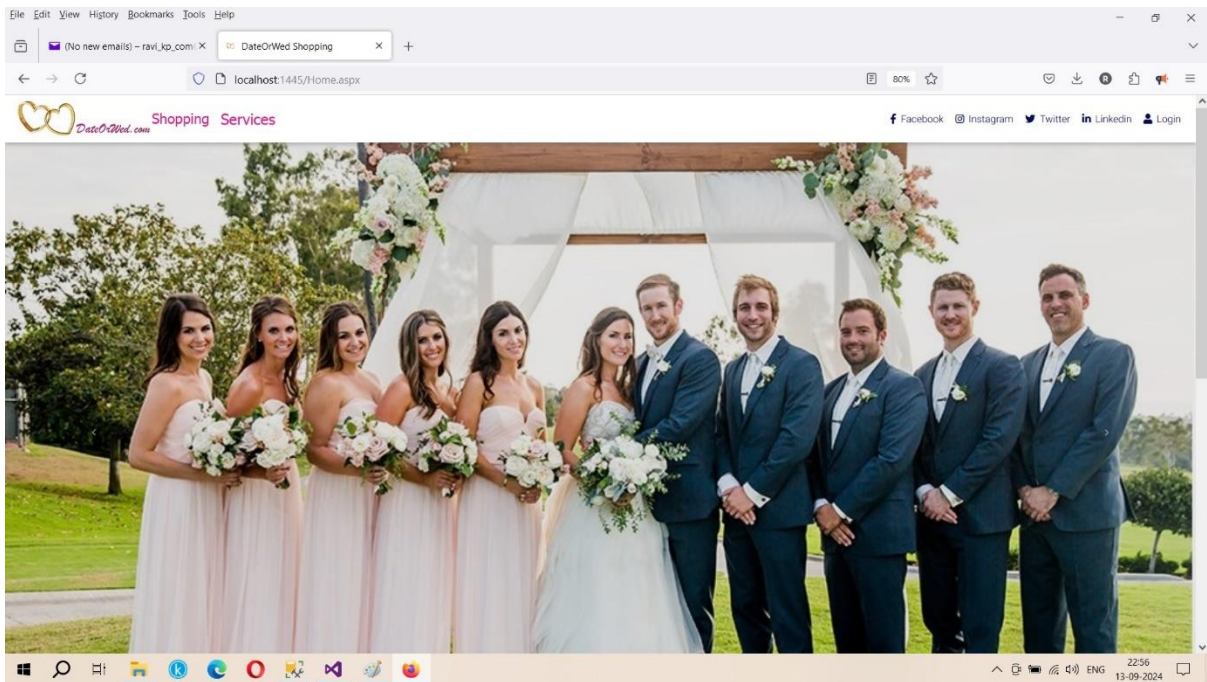
```
  <!-- MDB core JavaScript -->
```

```
  <script type="text/javascript" src="js/mdb.min.js"></script>
```

```
</form>
```

```
</body>
```

```
</html>
```



```
*****Home.aspx.cs *****
```

```
using DowBAL;
using DowBEntity;
using DowDAL;
using System;
using System.Collections.Generic;
using System.Data;
using System.Linq;
using System.Text;
using System.Text.RegularExpressions;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.Security;
//using LogTracer;
public partial class Home1 : System.Web.UI.Page
{
    public string UserName = "";
    DDLogTracer objlog = new DDLogTracer();
    protected void Page_Load(object sender, EventArgs e)
    {
        try{
            bool IsMobile = false;
            if (Request.Browser.IsMobileDevice) { IsMobile = true; } else { IsMobile = false; }
            if (IsMobile)
            {
                hometools.Visible = false;
                slider.Visible = false;
            }

            CIsLoggedInUser CIsusr = (CIsLoggedInUser)Session["LoggedInUser_Specs"];
            if (CIsusr != null)
            {
                UserName = CIsusr.UserName;
            }
        }
    }
}
```

```

}
else
{
    string user;
    HttpCookie myCookie = Request.Cookies["DateOrWedCookie"];
    if (myCookie != null)
    {
        user = myCookie.Value.ToString();
    }
    else
    {
        user = "Guest";
    }
    guest_login(user);

//    Response.Redirect("MLB_Login.aspx");
}
string pageid = Request.QueryString["pageid"];
if (pageid == null) { pageid = "1"; }
Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
if (Clsusr != null)
{
    welcome.InnerText = "Welcome " + Clsusr.UserName;
    if (Clsusr.UserName != "Guest")
    {
        Lin.Visible = false;
        LReg.Visible = false;
        Lout.Visible = true;
        Profiles.Visible = true;
    }
    else
    {
        Lin.Visible = true;
        LReg.Visible = true;
        Lout.Visible = false;
        Profiles.Visible = false;
    }
    ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
Convert.ToInt32(Clsusr.GroupId));
}
string msg = Request.QueryString["msg"];
if (msg != null)
{
    Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
}
//FillCourosal();
Bindetails(Convert.ToInt16( Clsusr.GroupId));
//FillRecentNews();

//    FillCommonCount();
//    FillMostRecentOrder();
}
catch (Exception ex)
{

```

```

Common.ShowMessageHtml(ex.Message, 1, true, dvMsgBox);
ExceptionLogging.SendExcepToDB(ex);

objlog.Trace_Error(5, "DateOrWed", "OrderSearchList", "Bindetails_Filter",
ex.Message);

}
}
private void ManageNavigations(int pageid, int userid, int groupid)
{
    bool IsLeftBar = false;
    //IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
myBtnClickHandler);

    if (IsLeftBar == true)
    {
        LeftSideBar.Attributes["class"] = "span2";
        MainContent.Attributes["class"] = "span10";
        ShowPageHeader.Attributes.CssStyle.Add("display", "none;");//style="display:
block;"
        PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
    else
    {
        LeftSideBar.Attributes["class"] = "span1";
        MainContent.Attributes["class"] = "span12";
        ShowPageHeader.Attributes.CssStyle.Add("display", "none;");
    }
}

protected void guest_login(string user)
{

    string City = "Hyderabad";// _City;
    string Country = "India";// _Country;

    BALCreateUser objlogin = new BALCreateUser();
    DataTable status_login = objlogin.LoginGuest(user, user,
Request.ServerVariables["REMOTE_ADDR"].ToString(), "", City, Country);
    string login_status = status_login.Rows[0]["UserStatus"].ToString();

    ClsLoggedInUser objLogedInUser;
    if (login_status == "True")
    {
        objLogedInUser = new ClsLoggedInUser();

        objLogedInUser.UserId = status_login.Rows[0]["UserId"].ToString();
        objLogedInUser.UserName = status_login.Rows[0]["UserName"].ToString();
        objLogedInUser.GroupId = status_login.Rows[0]["GroupId"].ToString();
        objLogedInUser.LoginStatus = "True";
        objLogedInUser.LoginID = Convert.ToInt32(status_login.Rows[0]["LoginId"]);
    }
}

```

```

objLoggedInUser.DealerId = Convert.ToInt32(status_login.Rows[0]["DealerId"]);
objLoggedInUser.RoleId = Convert.ToInt32(status_login.Rows[0]["RoleId"]);
objLoggedInUser.BrandId = 1; //Convert.ToInt32(status_login.Rows[0]["BrandId"]);

Session["LoggedInUser_Specs"] = objLoggedInUser;

FormsAuthentication.SetAuthCookie(objLoggedInUser.UserName, false);
HttpCookie myCookie = new HttpCookie("DateOrWedCookie");
DateTime now = DateTime.Now;
myCookie.Value = objLoggedInUser.UserName;
myCookie.Expires = now.AddYears(1);

Response.Cookies.Add(myCookie);

ClsPermissioGroup clsperGrp = new ClsPermissioGroup();
UserPermissions Entuserper =
clsperGrp.GetUserLandingPage(objLoggedInUser.UserName);

if (Entuserper.PagId != 0)
{
    Entuserper.Add = true;
    Entuserper.Edit = true;
    Entuserper.Delete = true;
    Entuserper.View = true;
    //BalRecentPlacess objRemoveRcntPlc = new BalRecentPlacess();
    //bool rst =
objRemoveRcntPlc.DeleteRecentPlaces(Convert.ToInt32(status_login.Rows[0]["UserId"].ToString()));
    Response.Redirect("~/Home1.aspx");// + Entuserper.PageUrl;
}
}
}
// This is Event Handler
private void Bindetails(int Groupid)
{
    if (Groupid == 0)
    {
    }
    else if ( Groupid == 3 || Groupid==1)
    {
        List<Eteproductbuyers> Lstx = (new Products()).GetProductBuyers(1,Groupid, "");

        if (Lstx != null && Lstx.Count > 0)
        {
        }
        else
        {
        }
        List<Eteproductbuyers> Lst1 = (new Products()).GetProductBuyers(2, Groupid, "");

        if (Lst1 != null && Lst1.Count > 0)

```

```

        {
        }
        else
        {
        }
    }
    else
    {

    }

    //try
    //{
    //CategoriesDS.DataBind();
    //ProductsGrDS.DataBind();
    //ProductsDS.DataBind();
    //whds.DataBind();
    //WebDataTree1.DataSourceID = "whds";

    //List<Eteproductcat> Lst = (new Products()).GetProductCat("Managedealer.aspx");

    //if (Lst != null && Lst.Count > 0)
    //{

        //WebDataTree1.DataSourceID DataSource = Lst;
        //WebDataTree1.DataBind();
        //WebDataTree1.Visible = true;
        ///}
        //else
        //{
        // WebDataTree1.Visible = false;
        //}
    }
    protected void myBtnClickHandler(object sender, EventArgs e)
    {
        LinkButton lnk = sender as LinkButton;

//    LinkButton lnk = (LinkButton)sender;
        if (lnk != null)
        {

            string ID = lnk.ID;

            DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
            if (ds != null)
            {
                if (ds.Tables[0].Rows.Count > 0)
                {
                    DataTable dtdll = ds.Tables[0];
                    foreach (DataRow drnew in dtdll.Rows)
                    {
                        switch (drnew["lbtncode"].ToString())
                        {
                            case "SAVE":

```

```

        // Call SAVE Method
        break;
    case "URL":
        Response.Redirect(drnew["pageurl"].ToString() + "&pid=" + ID);
        break;
    case "PRD":
        Response.Redirect(drnew["lbtnurl"].ToString() + "&pid=" +
drnew["lbtnid"].ToString());
        break;
    case "AUDIT":
        Response.Redirect(drnew["pageurl"].ToString() + "&frompage=" +
Request.QueryString["pageid"]);
        break;

    case "CANCEL":
        // Call Cancel Method
        break;
    case "MENU":
        // Call MENUS Method
        Response.Redirect(drnew["lbtnurl"].ToString() );
        break;
    default:

        Response.Redirect(drnew["lbtnurl"].ToString() );

        break;
    }
}
}
}
}
}
DropDownList dropDownList = sender as DropDownList;
if (dropDownList != null)
{
    //Common.ShowMessageHtml("", 0, false, dvMsgBox);
    //Bindetails_Filter(Table1);
}
}

private void FillCourosal()
{
    string PreHtml = "";
    StringBuilder sb = new StringBuilder();
    sb.Append(" <ol class='carousel-indicators'>");
    List<Etetblcourosalitems> RecentNew = (new NewsItems()).GetcourosalItems();
    string tm = "";
    for (int ctr = 0; ctr < RecentNew.Count; ctr++)
    {
        tm = "";

        if (ctr == 0) { tm = " class='active' "; }
        sb.Append("<li data-target='#myCarousel' data-slide-to='"+ctr+"'"+tm+" ></li>\n");
    }
    sb.Append("</ol>");
    sb.Append("<div class='carousel-inner'> ");

```



```

        tm = " active ";
        foreach (var row in RecentNew)
        {
            sb.Append(" <div class='item'+tm+'>\n");
            sb.Append("<img src='"+row.fpath+"' alt='"+row.aname+"' style='width:auto;
onclick='~/ViewSearchResult.aspx?id='"+row.id.ToString()+"'/>\n");
            sb.Append(" </div>\n");

            tm = "";
        }
        sb.Append(" <a class='left carousel-control' href='#myCarousel' data-slide='prev'>\n");
        sb.Append(" <span class='glyphicon glyphicon-chevron-left'></span>\n");
        sb.Append(" <span class='sr-only'><</span>\n");
        sb.Append(" </a>\n");
        sb.Append(" <a class='right carousel-control' href='#myCarousel' data-
slide='next'>\n");
        sb.Append(" <span class='glyphicon glyphicon-chevron-right'></span>\n");
        sb.Append(" <span class='sr-only'>></span>\n");
        sb.Append(" </a>\n");
        //sb.Append(" <a class='igte_Button' href='ViewSearchResult.aspx'>Search</a>\n");

        PreHtml = sb.ToString();
        //objlog.Trace_Log(4, "home", "1", "corousal", PreHtml);
        if (PreHtml.Length > 0)
        {
            //myCarousel.InnerHtml = PreHtml;
        }
    }

    private void FillRecentNews()
    {
        string PreHtml = ""; // "<h5><strong>Recent News</strong></h5><table class='table'>";
        StringBuilder sb = new StringBuilder();
        //sb.Append("<h5 class='customer'><strong>Recent News</h5><table class='table'>");
        sb.Append("<table class='table'>");
        //THU 28 OCT 2015 By Sudheer
        List<EteNewsItem> RecentNew = (new NewsItems()).GetNewsItems();
        foreach (var row in RecentNew)
        {
            string PubDate = row.PublishDate.ToString("ddd dd MMM yyyy").ToUpper(); //ddd-
            mmm-yyyy hh:mm:ss
            sb.Append("<tr><th width=250px>" + row.Summary + "</th><td style='font-size:
12px; text-align: right;'>" + PubDate + " By " + row.UserName + "</td></tr>");
            sb.Append("<tr><td style='font-size: 12px; text-align: left; padding-left: 15px; border:
0px; padding-top: 5px; padding-bottom: 5px; color: blue; font-weight: 600; '>" + row.Details +
"</td></tr>");

        }
        PreHtml = sb.ToString();
        if (PreHtml.Length > 0)
        {
            PreHtml += "</table>";
            //dv_RecentNews.InnerHtml = PreHtml;
        }
    }

```

```

}
//private void FillCommonCount()
//{

// ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
// if (Clsusr != null)
// {
//     EteCommonCount ComnCount = new
NewsItems().GetCommonCount(Clsusr.RoleId, Clsusr.DealerId,
Convert.ToInt32(Clsusr.UserId));
//     if (ComnCount != null)
//     {
//         ;

//         decimal SevenDaysCanPercent =
Convert.ToDecimal((ComnCount.CancellationPerLast7Days ?? 0) /
(Convert.ToDecimal(ComnCount.Last7Days == 0 ? 1 : ComnCount.Last7Days)) * 100);
//         decimal TwentyEightDaysCanPercent =
Convert.ToDecimal((ComnCount.CancellationPerLast28Days ?? 0) /
(Convert.ToDecimal(ComnCount.Last28Days == 0 ? 1 : ComnCount.Last28Days)) * 100);

//     }

// }
//}

private void FillMostRecentOrder()
{
    string PreHtml = ""; // "<h5><strong>Recent News</strong></h5><table class='table'>";
    StringBuilder sb = new StringBuilder();
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    if (Clsusr != null)
    {
        //List<EteMostRecentOrders> MostRecentOrders = new
NewsItems().GetMostRecentOrders(Clsusr.RoleId, Clsusr.DealerId,
Convert.ToInt32(Clsusr.UserId));
        //foreach (var row in MostRecentOrders)
        //{
        //    sb.Append("<p><a href='" + row.NavURL + "'>" + row.orderreferencestring +
"</a> " + row.CustName + " </p>");

        //}
        //PreHtml = sb.ToString();
        if (PreHtml.Length > 0)
        {
            //dv_MostRecentCount.InnerHtml = PreHtml;
        }
    }
}

protected void ImageButton1_Click(object sender, ImageClickEventArgs e)
{
    Response.Redirect("~/viewsearchresult.aspx?SrchStrng=test");
}

```

```

}
***** Home1.aspx*****
<%@ Page Title="DateOrWed-Vendor-Home" Language="C#"
MasterPageFile="~/MainMaster.master" AutoEventWireup="true"
CodeFile="Home1.aspx.cs" Inherits="Home1" %>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
    <%-- <link rel="stylesheet" href="css/swiper.css"/>
<link rel="stylesheet" href="css/swiper.min.css"/>

<script src="js/swiper.js"></script>
<script src="js/swiper.min.js"></script>
<script src="swiper.esm.js"></script>
<script src="js/swiper.esm.bundle.js"></script>--%>

<style type="text/css">
    .padding {
        padding: 0px;
    }

    .padding th {
        padding: 0px !important;
    }

    .span3 td {
        padding: 0px;
    }

    .span5 td {
        padding: 0px;
    }

    .span5 th {
        padding: 0px;
    }
</style>
<script type="text/javascript">
    function GetURLSring1() {
        // alert('GetURLSring-1');
        //var fullURL = (location.pathname + location.search).substr(1);

        var loc="";
        var EnCodedFullURL = encodeURIComponent(fullURL);
        var SearchString =
document.getElementById("<%=txt_SearchString.ClientID%>").value;
        var url = 'ViewSearchResult.aspx?pageid=198' + '&SrchStrng=' +
encodeURIComponent(SearchString) + '&loc=' + encodeURIComponent(loc);
        //$(location).attr('href', url);
        window.location.href = url;
        event.preventDefault();
    }
</script>

```

</asp:Content>

```
<asp:Content ID="Content2" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
  <div id="hometools" class="homeTools" runat="server" style="height:150px" >
    <div class="wrapper">
      <div class="pure-g">

        <div class="pure-u-3-4">
          <div class="pure-g">
            <div class="homeTools__item pure-u-1-5 app-link" data-href="Home.aspx">
<%--      <span class="homeTools__itemIcon homeTools__itemIcon--website"></span>--
%>
          <i class='fa fa-home '></i><p class="homeTools__itemTitle">Shopping</p>
          </div>
          <div class="homeTools__item pure-u-1-5 app-link" data-
href="MyCart.aspx?page=Orders">
<%--      <span class="homeTools__itemIcon homeTools__itemIcon--myvendors"></span>--
-%>
          <i class='fa fa-cutlery'></i><p class="homeTools__itemTitle">Services</p>
          </div>
          <div class="homeTools__item pure-u-1-5 app-link" data-
href="MyCart.aspx?page=Orders">
<%--      <span class="homeTools__itemIcon homeTools__itemIcon--myvendors"></span>--
-%>
          <i class='fa fa-building icon-vendor-organizacion'></i><p
class="homeTools__itemTitle">Business Listings</p>
          </div>
          <div class="homeTools__item pure-u-1-5 app-link" data-
href="MyCart.aspx?page=Orders">
<%--      <span class="homeTools__itemIcon homeTools__itemIcon--myvendors"></span>--
-%>
          <i class='fa fa-shopping-cart fa-lg'></i><p class="homeTools__itemTitle">Your
Kart</p>
          </div>
          <div class="homeTools__item pure-u-1-5 app-link" data-href="MyProductsList.aspx">
<%--      <span class="homeTools__itemIcon homeTools__itemIcon--guests"></span>--%>
          <i class='fa fa-laptop fa-lg'></i> <p class="homeTools__itemTitle">Your wishlist</p>
          </div>

        <%--      <div class="homeTools__item pure-u-1-5 app-link" data-
href="MyCart.aspx?page=Pending">
          <span class="homeTools__itemIcon homeTools__itemIcon--checklist"></span>
          <i class='fa fa-home'></i><p class="homeTools__itemTitle">Your Pending Orders</p>
          </div>
          --%>
        <%--      <div class="homeTools__item pure-u-1-5 app-link" data-
href="BSOffers.aspx?page=Budget">
          <i class='fa fa-dollar fa-lg'></i> <p class="homeTools__itemTitle">Check your
Account</p>
          </div>--%>
        </div>
        </div>
        </div>
        <div class="pure-u-1-4">
          <div class="homeTools__more">
            <p id="welcome" class="homeTools__moreText" runat="server"> Welcome Vendor</p>
```

```
        <button id="Profiles" class="btn-outline outline-red app-link" role="button"
runat="server" data-href="UserDetail.aspx">
        Profile          </button>
        <a id="Lin" class="header-join-link" rel="nofollow" href="MLB_login.aspx"
runat="server">Log In</a>
        <a id="LReg" class="header-join-link app-ua-track-event" rel="nofollow"
href="Manageuser.aspx" runat="server">
        Sign up          </a>
        <a id="Lout" class="header-join-link" rel="nofollow" href="logout.aspx"
runat="server">Log Out</a>
    </div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<div class="container-fluid">
    <div class="row-fluid"> <!-- style="padding-top: 20px;--%>
        <div class="span2" id="dvlogo" runat="server" style="display: none;">
<!--          --%>
        </div>
        <div class="span10">
            <div class="row">
                <!--<h3>Ravikiran Techno Soft</h3>--%>
                <div id="dvMsgBox" class="span10 ordersave" runat="server" style="display:
block;">
                    <asp:Label ID="Label2" runat="server" Text=""></asp:Label>
                </div>
            </div>
        </div>
    </div>
</div>
</asp:Content>
```

```
<asp:Content ID="Content3" ContentPlaceHolderID="CPH_LeftSideBar" runat="Server">
    <div id="LeftSideBar" runat="server" class="span2">
        <asp:Table ID="Table1" runat="server" Width="250px">
        </asp:Table>
    </div>
</asp:Content>
```

```
<asp:Content ID="Content4" ContentPlaceHolderID="CPH_MainContent" runat="Server"
Visible="false" >
    <div class="span10" id="MainContent" runat="server">
        <div id="ShowPageHeader" class="row" style="display: none;" runat="server">
            <div id="PageName" class="span10 customer" runat="server">
            </div>
        </div>
    </div>
```

```
<div id="Div1" class="row" style="display: block;">
    <div id="Div2" class="span14">
```

```

<div class="content_holder">
  <div id="slider" class="homeSlider" runat="server">
    <div class="homeSlider__container app-main-slider-home swiper-container"
style="height:340px"> <%--style="height:340px"--%>
      <div class="swiper-wrapper">
        <div class="homeSlider__item swiper-slide">
          <div class="homeSlider__droplayer"></div>
          
          <div class="homeSlider__boxTags app-main-slider-home-caption">
            <div class="wrapper">
              <div class="homeSlider__tags app-main-slider-home-related"></div>
            </div>
          </div>
        </div>
      </div>
    </div>
    <div class="swiper-pagination homeSlider__pagination app-main-slider-home-
bullet"></div>
  </div>
  <div class="homeSlider__content">
    <p class="homeSlider__subtitle"> Shopping Destination in Hyderabad is Laad
Bazar</p>
    <h1 class="homeSlider__title app-main-slider-home-title">Shopping Destination Online
is DateOrWed.com!</h1>

    <div class="homeSlider__form index-busc-form app-vendors-search-form"
id="frmSearch" >

      <div id="innersearch" style="float: inherit; width: auto; margin-top: 5px; margin-left:
3px; display:block">
        <asp:DropDownList ID="ddl_location" runat="server" Width="150px"
Height="45px">

          <asp:ListItem Value="0">--Select a location--</asp:ListItem>
          <asp:ListItem Value="1">Hyderabad</asp:ListItem>
          <asp:ListItem Value="2">Delhi</asp:ListItem>
          <asp:ListItem Value="3">Mumbai</asp:ListItem>
          <asp:ListItem Value="4">Chennai</asp:ListItem>
        </asp:DropDownList>
        <asp:TextBox ID="txt_SearchString" runat="server" Width="500px"
Height="35px" onkeydown="javascript:TextName_OnKeyDown(event)"></asp:TextBox>
        <button id="Button1" type="button" class="btn btn-primary no-hover"
onclick="JavaScript:GetURLSring1()"><i class='fa fa-
search'></i>&nbsp;&nbsp;&nbsp;Search</button>

      </div>

    </div>

    <p class="homeSlider__boxLinks">
      <a class="homeSlider__boxLinksItem" data-group="1"
href="GOrderSearchList.aspx?pid=1">Wedding Dress</a>,

```

```

        <a class="homeSlider__boxLinksItem" data-group="2"
href="GOrderSearchList.aspx?pid=2"> Wedding Suits</a>.
        <a class="homeSlider__boxLinksItem" data-group="5"
href='OrderSearchList.aspx?pid=5'>Wedding Venues</a>,
        <a class="homeSlider__boxLinksItem" data-group="8"
href="OrderSearchList.aspx?pid=8">Wedding Photographer</a>,
        <a class="homeSlider__boxLinksItem" data-group="13"
href="OrderSearchList.aspx?pid=13">Wedding Music</a>,
        <%--<a class="homeSlider__boxLinksItem" data-group="10" href="wedding-
cars">Wedding Cars</a>, --%>
        <a class="homeSlider__boxLinksItem" data-group="11"
href="OrderSearchList.aspx?pid=11"> Wedding Invitations</a>,
        <a class="homeSlider__boxLinksItem" data-group="9"
href="OrderSearchList.aspx?pid=9">Wedding Florist</a>        </p>
    </div>

```

```
</div>
```

```
<script>
```

```

$(document).ready(function () {
    $('input[name=id_poblacion]').val("");
    $('input[name=id_sector]').val("");

```

```

    homeSliderInit({
        "1": {
            "imgs": [{ "src": "images\wide\weddingdress.png", "rw": "10", "NOVIO": "Wedding
Dresses" },
            { "src": "images\wide\weddingjewellery.png", "rw": "10", "NOVIO": "Wedding
Jewellery" }],
            "text": "Post your products for Sale"
        },
        "2": {
            "imgs": [{ "src": "images\wide\djsangeeth.png", "rw": "10", "NOVIO": "DJs,
Sangeeth & Music" },
            { "src": "images\wide\travel.png", "rw": "10", "NOVIO": "Travel Arrangements"}],
            "text": "Shopping Destination Online is DateOrWed.com!"
        },
        "3": {
            "imgs": [{ "src": "images\wide\weddingjewellery.png", "rw": "10", "NOVIO":
"Wedding Jewellery" },
            { "src": "images\wide\WEDDINGMEHENDI.png", "rw": "10", "NOVIO": "Wedding
Mehendi & Makeup"}],
            "text": "Post your products for Sale"
        },
        "4": {
            "imgs": [{ "src": "images\wide\weddingfootwear.png", "rw": "100", "NOVIO":
"Footwear" },
            { "src": "images\wide\djsangeeth.png", "rw": "100", "NOVIO": "DJs, Sangeeth &
Music" }],
            "text": "Shopping Destination Online is DateOrWed.com!"
        },
        "5": {
            "imgs": [{ "src": "images\wide\weddingjewellery.png", "rw": "100", "NOVIO":
"Wedding Jewellery" },
            { "src": "images\wide\weddingfootwear.png", "rw": "100", "NOVIO": "Footwear" }],

```

```

        "text": "Post your products for Sale"
    },
    "6": {
        "imgs": [{"src": "images\\wide\\WEDDINGMAKEUP.png", "rw": "100", "NOVIO":
"Wedding Mehendi & Makeup" },
        {"src": "images\\wide\\weddinginvites.png", "rw": "100", "NOVIO": "Wedding
Invitations" }],
        "text": "Shopping Destination Online is DateOrWed.com!"
    },
    "7": {
        "imgs": [{"src": "images\\wide\\djsangeeth.png", "rw": "100", "NOVIO": "DJs,
Sangeeth & Music" },
        {"src": "images\\wide\\weddingfootwear.png", "rw": "100", "NOVIO": "Footwear" }],
        "text": "Post your products for Sale"
    },
    "firstPictureIndex": 1
});
});
</script>

```

```
<div class="clear"> <p><br /> </p> </div>
```

```

<div class="container" runat="server" dir="auto">
<%--first container starts--%>
<div class="homePremiumSlider" >
    <div class="wrapper">
        <p class="homeSection__title homeSection__title--md homeSection__title--
white">Wedding Shopping Vendors</p>
        <p>&nbsp;</p>
        <div class="homePremiumSlider__container">
            <div class="app-slideshow-vendors-home swiper-container">

                <div class="homePremiumBox swiper-slide app-link" data-
href="OrderSearchList.aspx?id=1">
                    <div class="homePremiumBox__item">
                        <figure class="homePremiumBox__figure img-zoom">
                            <a href="OrderSearchList.aspx?id=1">
                                
                            </a>
                        </figure>
                        <div class="homePremiumBox__headerBox">
                            <a class="homePremiumBox__name" href="OrderSearchList.aspx?id=1">
                                Wedding Dresses </a>
                            <span class="homePremiumBox__category">Stylish Dresses</span>
                            <span class="homePremiumBox__location">From All over India</span>
                        </div>
                        <div class="homePremiumBox__pricesBox">
                            <div class="homePremiumBox__section">

```



```

                <p class="homePremiumBox__priceTitle homePremiumBox__priceTitle--
empty">
                    for all Budgets
</span>--</span>
                </p>
            </div>
        </div>
        <p class="homePremiumBox__description">
            All kinds of Wedding Dresses for Brides and Grooms are available. Which includes
            Tradional wear, Silk Sarees, Lehengha, Gagra Cholies for brides. </p>
        </div>
    </div>
    <div class="homePremiumBox swiper-slide app-link" data-
href="OrderSearchList.aspx?id=2">
        <div class="homePremiumBox__item">
            <figure class="homePremiumBox__figure img-zoom">
                <a href="OrderSearchList.aspx?id=2">
                    
                </a>
            </figure>
            <div class="homePremiumBox__headerBox">
                <a class="homePremiumBox__name" href="OrderSearchList.aspx?id=2">
                    Jewellery </a>
                <span class="homePremiumBox__category">Speciality Jewellery</span>
                <span class="homePremiumBox__location">All over India</span>
            </div>
            <div class="homePremiumBox__pricesBox">
                <div class="homePremiumBox__section">
                    <p class="homePremiumBox__priceTitle homePremiumBox__priceTitle--
empty">
                        Wide range of prices <span>--</span>
                    </p>
                </div>
            </div>
            <p class="homePremiumBox__description">
                Jewellery is available in Rings, Chains, Necklaces, Ear Rings, Headwear, Nose Ring,
                Hipwear and Anklets <b>...</b> </p>
            </div>
        </div>
    </div>
    <div class="homePremiumBox swiper-slide app-link" data-
href="OrderSearchList.aspx?id=3">
        <div class="homePremiumBox__item">
            <figure class="homePremiumBox__figure img-zoom">
                <a href="OrderSearchList.aspx?id=2">
                    
                </a>
            </figure>
            <div class="homePremiumBox__headerBox">
                <a class="homePremiumBox__name" href="OrderSearchList.aspx?id=3">
                    Footwear </a>
                <span class="homePremiumBox__category">Sepciality Footwear</span>
                <span class="homePremiumBox__location">All over India</span>
            </div>
        </div>
    </div>

```

```
</div>
  <div class="homePremiumBox__pricesBox">
    <div class="homePremiumBox__section">
      <p class="homePremiumBox__priceTitle homePremiumBox__priceTitle--
empty">
          Wide range of prices          <span>--</span>
    </p>
  </div>
</div>
  <p class="homePremiumBox__description">
    All kinds of footwear available from across country<b>...</b>  </p>
</div>
</div>
```

```
<div class="homePremiumBox swiper-slide app-link" data-
href="OrderSearchList.aspx?id=4">
  <div class="homePremiumBox__item">
    <figure class="homePremiumBox__figure img-zoom">
      <a href="OrderSearchList.aspx?id=20">
        
      </a>
    </figure>
    <div class="homePremiumBox__headerBox">
      <a class="homePremiumBox__name" href="OrderSearchList.aspx?id=4">
        Makeup      </a>
      <span class="homePremiumBox__category">Unique Wedding Makeup</span>
      <span class="homePremiumBox__location">All over India</span>
    </div>
    <div class="homePremiumBox__pricesBox">
      <div class="homePremiumBox__section">
        <p class="homePremiumBox__priceTitle homePremiumBox__priceTitle--
empty">
            Prices for all Budgets          <span>--</span>
        </p>
      </div>
    </div>
    <p class="homePremiumBox__description">
      Makeup for Brides and family members <b>...</b>  </p>
    </div>
  </div>
  <p class="homeSection__extraTitle">&nbsp;</p>
  <p>&nbsp;</p>
  <p class="homeSection__title homeSection__title--md homeSection__title--
white">Wedding Services Vendors</p>
  <p>&nbsp;</p>
```

```
<div class="homePremiumBox swiper-slide app-link" data-
href="OrderSearchList.aspx?id=9">
  <div class="homePremiumBox__item">
    <figure class="homePremiumBox__figure img-zoom">
```

```

<a href="OrderSearchList.aspx?id=2">
  
</a>
</figure>
<div class="homePremiumBox__headerBox">
  <a class="homePremiumBox__name" href="OrderSearchList.aspx?id=9">
    Venues & Decoration </a>
  <span class="homePremiumBox__category">Wedding Venues</span>
  <span class="homePremiumBox__location">All over India</span>
</div>
  <div class="homePremiumBox__pricesBox">
    <div class="homePremiumBox__section">
      <p class="homePremiumBox__priceTitle homePremiumBox__priceTitle--
empty">
        Wide range of prices <span>--</span>
      </p>
    </div>
  </div>
  <div class="homePremiumBox__description">
    Venues are availabe in Banquet Halls, Function Halls, Party Lounge, Resorts and
    Farmhouses <b>...</b> </p>
  </div>
</div>
  <%-- added for container--%>
  <div class="homePremiumBox swiper-slide app-link" data-
href="ProductSearchList.aspx?id=5">
    <div class="homePremiumBox__item">
      <figure class="homePremiumBox__figure img-zoom">
        <a href="ProductSearchList.aspx?id=1">
          
        </a>
      </figure>
      <div class="homePremiumBox__headerBox">
        <a class="homePremiumBox__name" href="OrderSearchList.aspx?id=5">
          Mehendi </a>
        <span class="homePremiumBox__category">Mehendi</span>
        <span class="homePremiumBox__location">for Bride and others</span>
      </div>
      <div class="homePremiumBox__pricesBox">
        <div class="homePremiumBox__section">
          <p class="homePremiumBox__priceTitle homePremiumBox__priceTitle--
empty">
            for all Budgets
          </p>
        </div>
      </div>
      <div class="homePremiumBox__description">
        Various kinds of Mehendi designs are available for Bride and others <b>...</b>
      </p>
    </div>
  </div>
</div>

```

```
<div class="homePremiumBox swiper-slide app-link" data-
href="OrderSearchList.aspx?id=13">
  <div class="homePremiumBox__item">
    <figure class="homePremiumBox__figure img-zoom">
      <a href="OrderSearchList.aspx?id=20">
        
      </a>
    </figure>
    <div class="homePremiumBox__headerBox">
      <a class="homePremiumBox__name" href="OrderSearchList.aspx?id=13">
        DJ & Sangeeth      </a>
      <span class="homePremiumBox__category">DJ & Sangeeth</span>
      <span class="homePremiumBox__location">All over India</span>
    </div>
    <div class="homePremiumBox__pricesBox">
      <div class="homePremiumBox__section">
        <p class="homePremiumBox__priceTitle homePremiumBox__priceTitle--
empty">
          Prices range based on budgets                                <span>--
</span>
        </p>
      </div>
    </div>
    <p class="homePremiumBox__description">
      Available services include DJs, Music, Sangeeth and Singers for various functions
<b>...</b>      </p>
    </div>
  </div>
```

```
<div class="homePremiumBox swiper-slide app-link" data-
href="OrderSearchList.aspx?id=16">
  <div class="homePremiumBox__item">
    <figure class="homePremiumBox__figure img-zoom">
      <a href="OrderSearchList.aspx?id=20">
        
      </a>
    </figure>
    <div class="homePremiumBox__headerBox">
      <a class="homePremiumBox__name" href="OrderSearchList.aspx?id=16">
        Travelling      </a>
      <span class="homePremiumBox__category">Travel Planning</span>
      <span class="homePremiumBox__location">Local & Internatinal</span>
    </div>
    <div class="homePremiumBox__pricesBox">
      <div class="homePremiumBox__section">
        <p class="homePremiumBox__priceTitle homePremiumBox__priceTitle--
empty">
          Prices for your budget                                <span>--</span>
        </p>
      </div>
    </div>
    <p class="homePremiumBox__description">
      India and International Travelling arrangement can be arranged for
everyone.<b>...</b>      </p>
```

```
</div>
</div>
```

```
    <%--container ends--%>
    <%-- </div>--%>
    </div>
  </div>
  <%--
    <span class="homePremiumSlider__nav homePremiumSlider__nav--left app-
    slideshow-vendors-prev">
      <svg version="1.1" viewBox="0 0 24 24" xmlns="http://www.w3.org/2000/svg">
        <g fill="none" fill-rule="evenodd" stroke-linecap="round">
          <g stroke="#fff" stroke-width="2">
            <path d="m12.012 1.9875l-10.026 10.026"/>
            <path transform="translate(7 17) scale(1 -1) translate(-7 -17)" d="m12.012
            11.988l-10.026 10.026"/>
          </g>
        </g>
      </svg>
    </span>
```

```
    <span class="homePremiumSlider__nav homePremiumSlider__nav--right app-
    slideshow-vendors-next">
      <svg version="1.1" viewBox="0 0 24 24" xmlns="http://www.w3.org/2000/svg">
        <g fill="none" fill-rule="evenodd" stroke-linecap="round">
          <g stroke="#fff" stroke-width="2">
            <path transform="translate(17 7) scale(-1 1) translate(-17 -7)" d="m22.012
            1.9875l-10.026 10.026"/>
            <path transform="translate(17 17) scale(-1) translate(-17 -17)" d="m22.012
            11.988l-10.026 10.026"/>
          </g>
        </g>
      </svg>
    </span>
```

```
<script>
  $(document).ready(function () {
    $('input[name=id_poblacion]').val("");
    $('input[name=id_sector]').val("");
  });
</script>--%>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<h1></h1>
```

```
<%--second cotainer starts--%>
```

```
<%-- second container ends--%>
```

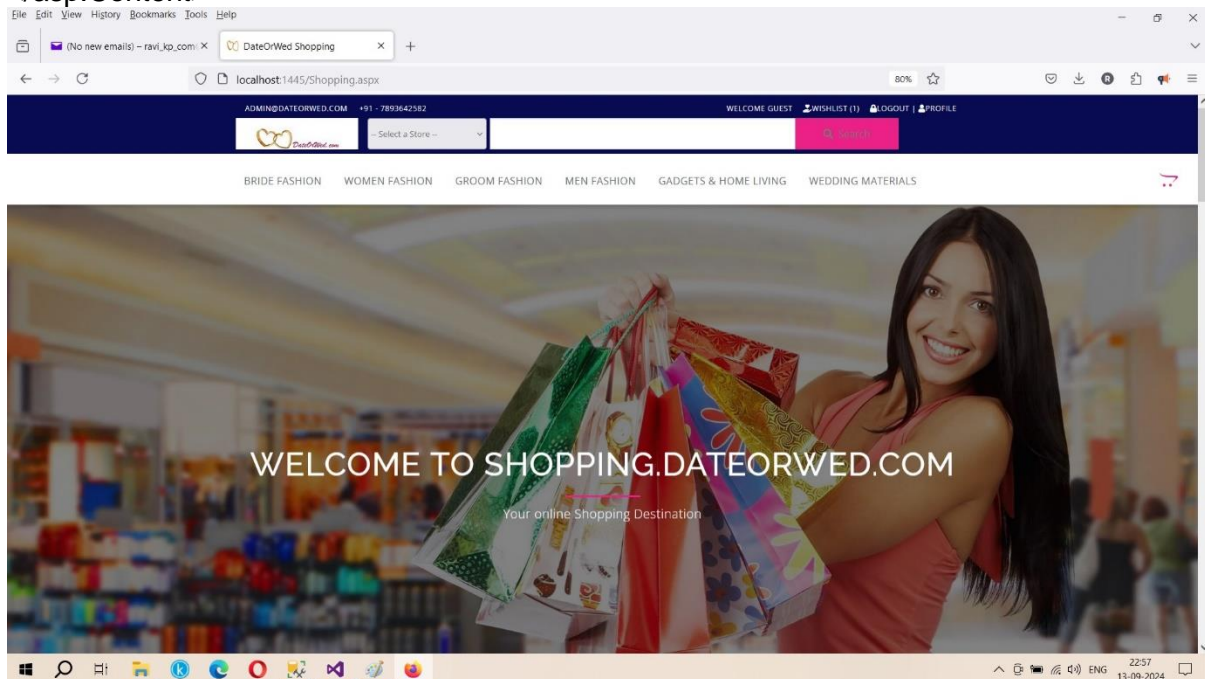
```
</div>
```

```
</div>
```

```

</div>
    </div>
    <%-- app footer here--%>
</asp:Content>

```



```

*****Home1.aspx.cs *****
using DowBAL;
using DowBEntity;
using DowDAL;
using System;
using System.Collections.Generic;
using System.Data;
using System.Linq;
using System.Text;
using System.Text.RegularExpressions;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.Security;
//using LogTracer;
public partial class Home1 : System.Web.UI.Page
{
    public string UserName = "";
    DDLogTracer objlog = new DDLogTracer();
    protected void Page_Load(object sender, EventArgs e)
    {
        try{
            bool IsMobile = false;
            if (Request.Browser.IsMobileDevice) { IsMobile = true; } else { IsMobile = false; }
            if (IsMobile)
            {
                hometools.Visible = false;
                slider.Visible = false;
            }
        }
    }
}

```

```

ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
if (Clsusr != null)
{
    UserName = Clsusr.UserName;
}
else
{
    string user;
    HttpCookie myCookie = Request.Cookies["DateOrWedCookie"];
    if (myCookie != null)
    {
        user = myCookie.Value.ToString();
    }
    else
    {
        user = "Guest";
    }
    guest_login(user);

//    Response.Redirect("MLB_Login.aspx");
}
string pageid = Request.QueryString["pageid"];
if (pageid == null) { pageid = "1"; }
Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
if (Clsusr != null)
{
    welcome.InnerText = "Welcome " + Clsusr.UserName;
    if (Clsusr.UserName != "Guest")
    {
        Lin.Visible = false;
        LReg.Visible = false;
        Lout.Visible = true;
        Profiles.Visible = true;
    }
    else
    {
        Lin.Visible = true;
        LReg.Visible = true;
        Lout.Visible = false;
        Profiles.Visible = false;
    }
    ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
Convert.ToInt32(Clsusr.GroupId));
}
string msg = Request.QueryString["msg"];
if (msg != null)
{
    Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
}
//FillCourosal();
Bindetails(Convert.ToInt16( Clsusr.GroupId));
//FillRecentNews();

```

```

//    FillCommonCount();
//    FillMostRecentOrder();
}
catch (Exception ex)
{
    Common.ShowMessageHtml(ex.Message, 1, true, dvMsgBox);
    ExceptionLogging.SendExcepToDB(ex);

    objlog.Trace_Error(5, "DateOrWed", "OrderSearchList", "Bindetails_Filter",
ex.Message);

}
}
private void ManageNavigations(int pageid, int userid, int groupid)
{
    bool IsLeftBar = false;
    //IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
myBtnClickHandler);

    if (IsLeftBar == true)
    {
        LeftSideBar.Attributes["class"] = "span2";
        MainContent.Attributes["class"] = "span10";
        ShowPageHeader.Attributes.CssStyle.Add("display", "none;"); //style="display:
block;"
        PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
    else
    {
        LeftSideBar.Attributes["class"] = "span1";
        MainContent.Attributes["class"] = "span12";
        ShowPageHeader.Attributes.CssStyle.Add("display", "none;");
    }
}

protected void guest_login(string user)
{
    string City = "Hyderabad";// _City;
    string Country = "India";// _Country;

    BALCreateUser objlogin = new BALCreateUser();
    DataTable status_login = objlogin.LoginGuest(user, user,
Request.ServerVariables["REMOTE_ADDR"].ToString(), "", City, Country);
    string login_status = status_login.Rows[0]["UserStatus"].ToString();

    ClsLoggedInUser objLoggedInUser;
    if (login_status == "True")
    {
        objLoggedInUser = new ClsLoggedInUser();
    }
}

```



```

objLogedInUser.UserId = status_login.Rows[0]["UserId"].ToString();
objLogedInUser.UserName = status_login.Rows[0]["UserName"].ToString();
objLogedInUser.GroupId = status_login.Rows[0]["GroupId"].ToString();
objLogedInUser.LoginStatus = "True";
objLogedInUser.LoginID = Convert.ToInt32(status_login.Rows[0]["LoginId"]);
objLogedInUser.DealerId = Convert.ToInt32(status_login.Rows[0]["DealerId"]);
objLogedInUser.RoleId = Convert.ToInt32(status_login.Rows[0]["RoleId"]);
objLogedInUser.BrandId = 1; //Convert.ToInt32(status_login.Rows[0]["BrandId"]);

Session["LoggedUser_Specs"] = objLogedInUser;

FormsAuthentication.SetAuthCookie(objLogedInUser.UserName, false);
HttpCookie myCookie = new HttpCookie("DateOrWedCookie");
DateTime now = DateTime.Now;
myCookie.Value = objLogedInUser.UserName;
myCookie.Expires = now.AddYears(1);

Response.Cookies.Add(myCookie);

ClsPermissioGroup clsperGrp = new ClsPermissioGroup();
UserPermissions Entuserper =
clsperGrp.GetUserLandingPage(objLogedInUser.UserName);

if (Entuserper.Pageld != 0)
{
    Entuserper.Add = true;
    Entuserper.Edit = true;
    Entuserper.Delete = true;
    Entuserper.View = true;
    //BalRecentPlacess objRemoveRcntPlc = new BalRecentPlacess();
    //bool rst =
objRemoveRcntPlc.DeleteRecentPlaces(Convert.ToInt32(status_login.Rows[0]["UserId"].To
String()));
    Response.Redirect("~/Home1.aspx");// + Entuserper.PageUrl);
}

}
}
// This is Event Handler
private void Bindetails(int Groupid)
{
    if (Groupid == 0)
    {
    }
    else if ( Groupid == 3 || Groupid==1)
    {
        List<Eteproductbuyers> Lstx = (new Products()).GetProductBuyers(1,Groupid, "");

        if (Lstx != null && Lstx.Count > 0)
        {
        }
        else

```

```

    {
    }
    List<Eteproductbuyers> Lst1 = (new Products()).GetProductBuyers(2, Groupid, "");

    if (Lst1 != null && Lst1.Count > 0)
    {
    }
    else
    {
    }
}
else
{

}

//try
//{
//CategoriesDS.DataBind();
//ProductsGrDS.DataBind();
//ProductsDS.DataBind();
//whds.DataBind();
//WebDataTree1.DataSourceID = "whds";

//List<Eteproductcat> Lst = (new Products()).GetProductCat("Managedealer.aspx");

//if (Lst != null && Lst.Count > 0)
//{

//WebDataTree1.DataSourceID DataSource = Lst;
//WebDataTree1.DataBind();
//WebDataTree1.Visible = true;
//}
//else
//{
// WebDataTree1.Visible = false;
//}
}
protected void myBtnClickHandler(object sender, EventArgs e)
{
    LinkButton lnk = sender as LinkButton;

//    LinkButton lnk = (LinkButton)sender;
    if (lnk != null)
    {

        string ID = lnk.ID;

        DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
        if (ds != null)
        {
            if (ds.Tables[0].Rows.Count > 0)
            {
                DataTable dtdll = ds.Tables[0];
            }
        }
    }
}

```



```

        if (ctr == 0) { tm = " class='active' "; }
        sb.Append("<li data-target='#myCarousel' data-slide-to='"+ctr+"'"+tm+" ></li>\n");
    }
    sb.Append("</ol>");
    sb.Append("<div class='carousel-inner' > ");
    tm = " active ";
    foreach (var row in RecentNew)
    {
        sb.Append(" <div class='item'+tm+'>\n");
        sb.Append("<img src='"+row.fpath+"' alt='"+row.aname+"' style='width:auto;
onclick='~/ViewSearchResult.aspx?id='"+row.id.ToString()+"'/>\n");
        sb.Append(" </div>\n");

        tm = "";
    }
    sb.Append(" <a class='left carousel-control' href='#myCarousel' data-slide='prev'>\n");
    sb.Append(" <span class='glyphicon glyphicon-chevron-left'></span>\n");
    sb.Append(" <span class='sr-only'><</span>\n");
    sb.Append(" </a>\n");
    sb.Append(" <a class='right carousel-control' href='#myCarousel' data-
slide='next'>\n");
    sb.Append(" <span class='glyphicon glyphicon-chevron-right'></span>\n");
    sb.Append(" <span class='sr-only'>></span>\n");
    sb.Append(" </a>\n");
    //sb.Append(" <a class='igte_Button' href='ViewSearchResult.aspx'>Search</a>\n");

    PreHtml = sb.ToString();
    //objlog.Trace_Log(4, "home", "1", "corousal", PreHtml);
    if (PreHtml.Length > 0)
    {
        //myCarousel.InnerHtml = PreHtml;
    }
}

private void FillRecentNews()
{
    string PreHtml = ""; // "<h5><strong>Recent News</strong></h5><table class='table'>";
    StringBuilder sb = new StringBuilder();
    //sb.Append("<h5 class='customer'><strong>Recent News</strong></h5><table class='table'>");
    sb.Append("<table class='table'>");
    //THU 28 OCT 2015 By Sudheer
    List<EteNewsItem> RecentNew = (new NewsItems()).GetNewsItems();
    foreach (var row in RecentNew)
    {
        string PubDate = row.PublishDate.ToString("ddd dd MMM yyyy").ToUpper(); //ddd-
mmm-yyyy hh:mm:ss
        sb.Append("<tr><th width=250px>" + row.Summary + "</th><td style='font-size:
12px; text-align: right;'>" + PubDate + " By " + row.UserName + "</td></tr>");
        sb.Append("<tr><td style='font-size: 12px; text-align: left; padding-left: 15px; border:
0px; padding-top: 5px; padding-bottom: 5px; color: blue; font-weight: 600;'>" + row.Details +
"</td></tr>");
    }
    PreHtml = sb.ToString();
}

```

```

        if (PreHtml.Length > 0)
        {
            PreHtml += "</table>";
            //dv_RecentNews.InnerHtml = PreHtml;
        }
    }
    //private void FillCommonCount()
    //{
        // ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        // if (Clsusr != null)
        // {
            // EteCommonCount ComnCount = new
NewsItems().GetCommonCount(Clsusr.RoleId, Clsusr.DealerId,
Convert.ToInt32(Clsusr.UserId));
            // if (ComnCount != null)
            // {
                // ;

                // decimal SevenDaysCanPercent =
Convert.ToDecimal((ComnCount.CancellationPerLast7Days ?? 0) /
(Convert.ToDecimal(ComnCount.Last7Days == 0 ? 1 : ComnCount.Last7Days)) * 100);
                // decimal TwentyEightDaysCanPercent =
Convert.ToDecimal((ComnCount.CancellationPerLast28Days ?? 0) /
(Convert.ToDecimal(ComnCount.Last28Days == 0 ? 1 : ComnCount.Last28Days)) * 100);

                // }

            // }
        //}

    private void FillMostRecentOrder()
    {
        string PreHtml = ""; // "<h5><strong>Recent News</strong></h5><table class='table'>";
        StringBuilder sb = new StringBuilder();
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr != null)
        {
            //List<EteMostRecentOrders> MostRecentOrders = new
NewsItems().GetMostRecentOrders(Clsusr.RoleId, Clsusr.DealerId,
Convert.ToInt32(Clsusr.UserId));
            //foreach (var row in MostRecentOrders)
            //{
                // sb.Append("<p><a href=" + row.NavURL + ">" + row.orderreferencestring +
"</a> " + row.CustName + " </p>");

                //}
            //PreHtml = sb.ToString();
            if (PreHtml.Length > 0)
            {
                //dv_MostRecentCount.InnerHtml = PreHtml;
            }
        }
    }
}

```

```

    }
    protected void ImageButton1_Click(object sender, ImageClickEventArgs e)
    {
        Response.Redirect("~/viewsearchresult.aspx?SrchStrng=test");
    }
}
***** Image.ashx *****
<%@ WebHandler Language="C#" Class="Image" %>

using System;
using System.Web;
using System.IO;
public class Image : IHttpHandler {

    public void ProcessRequest (HttpContext context) {
        context.Response.ContentType = "image/jpeg";
        if (!string.IsNullOrEmpty(context.Request["id"]))
        {
            int id = int.Parse(context.Request["id"]);
            string path = context.Server.MapPath(string.Format("~/images/{0}.jpg", id));
            using (FileStream fs = File.OpenRead(path))
            {

                using (BinaryReader br = new BinaryReader(fs))
                {

                    context.Response.OutputStream.Write(br.ReadBytes((int)fs.Length), 0,
(int)fs.Length);
                }
            }
        }
    }

    public bool IsReusable {
        get {
            return false;
        }
    }

}
***** ImageDownloader.ashx *****
<%@ WebHandler Language="C#" Class="ImageDownloader" %>

using System;
using System.Web;

public class ImageDownloader : IHttpHandler
{

    public void ProcessRequest (HttpContext context) {
        context.Response.ContentType = "IMAGE/JPEG";
        //context.Response.Write("Hello World");

        if (context.User.Identity.IsAuthenticated)
        {

```

```

        var ImageFilePath
=System.Configuration.ConfigurationSettings.AppSettings["ImageOutPut"];
        string file = context.Request.QueryString["file"];
        string FileFullPath = ImageFilePath + file;
        if (!string.IsNullOrEmpty(file) && System.IO.File.Exists(FileFullPath))
        {
            context.Response.Clear();
            context.Response.ContentType = "IMAGE/JPEG";
            //I have set the ContentType to "application/octet-stream" which cover any type of
file
            context.Response.AddHeader("content-disposition", "attachment;filename=" +
System.IO.Path.GetFileName(file));
            context.Response.WriteFile(FileFullPath);
            //here you can do some statistic or tracking
            //you can also implement other business request such as delete the file after
download
            context.Response.End();
        }
        else
        {
            context.Response.ContentType = "IMAGE/JPEG";
            context.Response.Write("File cannot be found!");
        }
    }
}

```

```

public bool IsReusable {
    get {
        return false;
    }
}

```

```

}
***** ImageCSharp.aspx *****
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="ImageCSharp.aspx.cs"
Inherits="_Default" %>

```

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

```

```

<html xmlns="http://www.w3.org/1999/xhtml" >
<head runat="server">
    <title>Untitled Page</title>
</head>
<body>
    <form id="form1" runat="server">
        <div>

            </div>
        </form>
</body>
</html>

```

```

***** ImageCSharp.aspx.cs *****
using System;
using System.Data;

```

```

using System.Data.SqlClient ;
using System.Configuration;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Web.UI.HtmlControls;

public partial class _Default : System.Web.UI.Page
{
protected void Page_Load(object sender, EventArgs e)
{
    if (Request.QueryString["ImageID"] != null)
    {
        string strQuery = "";
        string Type = Request.QueryString["Type"];
        if (Type == "1")
        {
            strQuery = "select Name='File'+convert(varchar(10),productid)+'jpg',
Contentype='image/jpeg', Data=images from productSellerImages where productid=@id";
        }
        else if (Type == "3")
        {
            strQuery = "select Name='File'+convert(varchar(10),productid)+'jpg',
Contentype='image/jpeg', Data=company_logo from tbluser_profile where userid=@id";
        }
        String strConnString =
System.Configuration.ConfigurationManager.ConnectionStrings["DowConectingString"].Con
nectionString;
        SqlCommand cmd = new SqlCommand(strQuery);
        cmd.Parameters.Add("@id", SqlDbType.Int).Value =
Convert.ToInt32(Request.QueryString["ImageID"]);
        SqlConnection con = new SqlConnection(strConnString);
        SqlDataAdapter sda = new SqlDataAdapter();
        cmd.CommandType = CommandType.Text;
        DataTable dt = new DataTable();
        cmd.Connection = con;
        try
        {
            con.Open();
            sda.SelectCommand = cmd;
            sda.Fill(dt);
        }
        catch
        {
            dt = null;
        }
        finally
        {
            con.Close();
            sda.Dispose();
            con.Dispose();
        }
    }
}

```



```

        if (dt != null)
        {
            Byte[] bytes = (Byte[])dt.Rows[0]["Data"];
            Response.Buffer = true;
            Response.Charset = "";
            Response.Cache.SetCacheability(HttpCacheability.NoCache);
            Response.ContentType = dt.Rows[0]["ContentType"].ToString();
            Response.AddHeader("content-disposition", "attachment;filename=" +
dt.Rows[0]["Name"].ToString());
            Response.BinaryWrite(bytes);
            Response.Flush();
            Response.End();
        }
    }
}

private DataTable GetData(SqlCommand cmd)
{
    DataTable dt = new DataTable();
    String strConnString =
System.Configuration.ConfigurationManager.ConnectionStrings["conString"].ConnectionStri
ng;
    SqlConnection con = new SqlConnection(strConnString);
    SqlDataAdapter sda = new SqlDataAdapter();
    cmd.CommandType = CommandType.Text;
    cmd.Connection = con;
    try
    {
        con.Open();
        sda.SelectCommand = cmd;
        sda.Fill(dt);
        return dt;
    }
    catch
    {
        return null;
    }
    finally
    {
        con.Close();
        sda.Dispose();
        con.Dispose();
    }
}
}

```

***** Index.aspx *****

```

<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Index.aspx.cs"
Inherits="Index" %>

```

```

<!DOCTYPE html>
<html lang="en">

```

```

<head>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
<meta http-equiv="x-ua-compatible" content="ie=edge">

```

```

<title>DateOrWed</title>
<link rel="icon" href="img/favicon.png" type="image/gif">
<!-- Font Awesome -->
<link rel="stylesheet" href="font/fontawesome/css/all.css">
<!-- Bootstrap core CSS -->
<link href="css/bootstrap.min.css" rel="stylesheet">
<!-- Material Design Bootstrap -->
<link href="css/mdb.min.css" rel="stylesheet">
<!-- Your custom styles (optional) -->
<link href="css/style.css" rel="stylesheet">
</head>

<body>
  <form runat="server">
    <asp:ScriptManager ID="ScriptManager1" runat="server">
      </asp:ScriptManager>
    <!-- Start your project here-->
    <!--Navbar -->
<nav class="mb-1 navbar fixed-top navbar-expand-lg navbar-light bg-light">
  <a class="navbar-brand" href="Home.aspx"></a>
  <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent-4"
    aria-controls="navbarSupportedContent-4" aria-expanded="false" aria-label="Toggle navigation">
    <span class="navbar-toggler-icon"></span>
  </button>
  <div class="collapse navbar-collapse" id="navbarSupportedContent-4">
    <ul class="navbar-nav ml-auto">
      <li class="nav-item">
        <a class="nav-link text-center" href="#">
          <i class="fas fa-store-alt"></i><br> Sell</a>
        </li>
      <li class="nav-item">
        <a class="nav-link text-center" href="#">
          <i class="far fa-question-circle"></i><br> Help</a>
        </li>
      <li class="nav-item">
        <a class="nav-link text-center" href="#">
          <i class="fas fa-envelope-open-text"></i><br> Messages</a>
        </li>
      <li class="nav-item dropdown">
        <a class="nav-link dropdown-toggle text-center" id="navbarDropdownMenuLink-4"
          data-toggle="dropdown"
          aria-haspopup="true" aria-expanded="false">
          <i class="far fa-user-circle"></i><br> Sign In</a>
        <div class="dropdown-menu dropdown-menu-right dropdown-info dropdown-primary"
          aria-labelledby="navbarDropdownMenuLink-4">
          <a class="dropdown-item" href="#">My account</a>
          <a class="dropdown-item" href="#">Log out</a>
        </div>
        </li>
    </ul>
  </div>
</nav>

```

```

<!--/.Navbar -->

<!--Carousel Wrapper-->
<div id="carousel-example-2" class="carousel slide carousel-fade" data-ride="carousel">

  <!--Slides-->
  <div class="carousel-inner" role="listbox">
    <div class="carousel-item active">
      <div class="view">
        
        <div class="mask rgba-black-slight"></div>
      </div>
    </div>
  </div>
  <!--/.Slides-->
  <!--Controls-->
  <!-- <a class="carousel-control-prev" href="#carousel-example-2" role="button" data-
slide="prev">
    <span class="carousel-control-prev-icon" aria-hidden="true"></span>
    <span class="sr-only">Previous</span>
  </a>
  <a class="carousel-control-next" href="#carousel-example-2" role="button" data-
slide="next">
    <span class="carousel-control-next-icon" aria-hidden="true"></span>
    <span class="sr-only">Next</span>
  </a> -->
  <!--/.Controls-->
</div>
<!--/.Carousel Wrapper-->

<!-- Search -->

<section class="search-sec">
  <div class="container">
    <div class="row">
      <div class="col-lg-12">
        <div class="row">
          <div class="col-lg-3 col-md-3 col-sm-12 p-0">
            <%--
              <select class="form-control search-slt"
id="exampleFormControlSelect1">
                <option>-Select-</option>
                <option>Shopping</option>
                <option>Services</option>
                <option>Business Listing</option>
              </select>
            --%>
          </div>
          <div class="col-lg-6 col-md-6 col-sm-12 p-0">

            <%--
              <input type="text" class="form-control search-slt" placeholder="Enter
Product/Service name">
            --%>
          </div>
          <div class="col-lg-3 col-md-3 col-sm-12 p-0">
            <%--
              <button type="button" class="btn btn-secondary wrn-btn mt-0 ml-
0">Search</button>--%>

```


[Fashion Shopping](#) *fas fa-chevron-right* style="font-size: 12px;">

>

[>*fas fa-caret-right*](ShoppingList.aspx?cid=1) Bride

[>*fas fa-caret-right*](ShoppingList.aspx?cid=3) BrideGroom

[>*fas fa-caret-right*](ShoppingList.aspx?cid=4) Men

[>*fas fa-caret-right*](ShoppingList.aspx?cid=2) Women

[>*fas fa-caret-right*](ShoppingList.aspx?cid=5) Kids-Girls

[>*fas fa-caret-right*](ShoppingList.aspx?cid=6) Kids-Boys

</div>

[> >*fas fa-cart-plus* Gadget Shopping *fas fa-chevron-right* style="font-size: 12px;">](#menu2)

>

[>*fas fa-caret-right*](ShoppingList.aspx?cid=7) TV & Electronics

[>*fas fa-caret-right*](ShoppingList.aspx?cid=8) Furniture & Furnishing

[>*fas fa-caret-right*](ShoppingList.aspx?cid=9) Metal Appliances

[>*fas fa-caret-right*](ShoppingList.aspx?cid=8) Home & Living

[>*fas fa-caret-right*](ShoppingList.aspx?cid=10) Wedding Material

</div>

[> >*fas fa-baby-carriage* Wedding Services *fas fa-chevron-right* style="font-size: 12px;">](#menu3)

>

[>*fas fa-caret-right*](ServicesList.aspx?cid=13) Venues

[>*fas fa-caret-right*](ServicesList.aspx?cid=11) Catering

[>*fas fa-caret-right*](ServicesList.aspx?cid=12) Decoration

[>*fas fa-caret-right*](ServicesList.aspx?cid=14) DJ & Music

[>*fas fa-caret-right*](ServicesList.aspx?cid=15) Transport

</div>

[> >*fas fa-building* Businesses *fas fa-chevron-right* style="font-size: 12px;">](#menu4)

>

```

        <a href="#" class="list-group-item" data-parent="#menu4"><i class="fas fa-
        caret-right"></i> Wedding Fashion Shopping</a>
        <a href="#" class="list-group-item" data-parent="#menu4"><i class="fas fa-
        caret-right"></i> Fashion Shopping</a>
        <a href="#" class="list-group-item" data-parent="#menu4"><i class="fas fa-
        caret-right"></i> Kids Shopping</a>
        <a href="#" class="list-group-item" data-parent="#menu4"><i class="fas fa-
        caret-right"></i> Electronics </a>
        <a href="#" class="list-group-item" data-parent="#menu4"><i class="fas fa-
        caret-right"></i> Furniture</a>
        <a href="#" class="list-group-item" data-parent="#menu4"><i class="fas fa-
        caret-right"></i> Metal Appliances</a>
        <a href="#" class="list-group-item" data-parent="#menu4"><i class="fas fa-
        caret-right"></i> Wedding Materials</a>
        <a href="#" class="list-group-item" data-parent="#menu4"><i class="fas fa-
        caret-right"></i> Wedding Services</a>
    </div>--%>

```

```

</div>

```

```

</div>

```

```

<div class="col-md-9">

```

```

    <!--Carousel Wrapper-->

```

```

<div id="carousel-example-1z" class="carousel slide carousel-fade" data-ride="carousel">

```

```

    <!--Indicators-->

```

```

    <ol class="carousel-indicators">

```

```

        <li data-target="#carousel-example-1z" data-slide-to="0" class="active"></li>

```

```

        <li data-target="#carousel-example-1z" data-slide-to="1"></li>

```

```

        <li data-target="#carousel-example-1z" data-slide-to="2"></li>
    </ol>

```

```

</ol>

```

```

    <!--/.Indicators-->

```

```

    <!--Slides-->

```

```

    <div class="carousel-inner" role="listbox">

```

```

        <!--First slide-->

```

```

        <div class="carousel-item active">

```

```

            
        </div>

```

```

    </div>

```

```

    <!--/First slide-->

```

```

    <!--Second slide-->

```

```

    <div class="carousel-item">

```

```

            
        </div>

```

```

    </div>

```

```

    <!--/Second slide-->

```

```

    <div class="carousel-item">

```

```

            
        </div>

```

```

    </div>

```

```

</div>

```

```

    <!--/.Slides-->

```

```

    <!--Controls-->

```

```

    <a class="carousel-control-prev" href="#carousel-example-1z" role="button" data-
    slide="prev">

```

```

    <span class="carousel-control-prev-icon" aria-hidden="true"></span>
    <span class="sr-only">Previous</span>
  </a>
  <a class="carousel-control-next" href="#carousel-example-1z" role="button" data-
slide="next">
    <span class="carousel-control-next-icon" aria-hidden="true"></span>
    <span class="sr-only">Next</span>
  </a>
<!--/.Controls-->
</div>
<!--/.Carousel Wrapper-->
  </div>
</div>
</div>
</div>
</section>

<section class="furec pb-3 pt-3">
  <div class="container-fluid">
    <div class="card">
      <h3 class="h3-responsive font-weight-bold pb-0 mb-0 pl-3">Featured
Recommendations</h3>
      <div class="p-3">
        <div class="row">
          <div class="col-md-2">
            <div class="card1 p-2 text-center">
              
              <div class="card-body">
                <h6 class="card-title font-weight-bold">Bride Fashion Store</h6>
                <div class="mb-4"></div>
                <a href="#">By: ABC</a>
                <div><a class="btn btn-secondary btn-sm btn-rounded">Get Best Price</a></div>
              </div>
            </div>
          </div>
          <div class="col-md-2">
            <div class="card1 p-2 text-center">
              
              <div class="card-body">
                <h6 class="card-title font-weight-bold">Bride Fashion Store</h6>
                <div class="mb-4"></div>
                <a href="#">By: ABC</a>
                <div><a class="btn btn-secondary btn-sm btn-rounded">Get Best Price</a></div>
              </div>
            </div>
          </div>
          <div class="col-md-2">
            <div class="card1 p-2 text-center">
              
              <div class="card-body">
                <h6 class="card-title font-weight-bold">Bride Fashion Store</h6>
                <div class="mb-4"></div>
                <a href="#">By: ABC</a>
                <div><a class="btn btn-secondary btn-sm btn-rounded">Get Best Price</a></div>
              </div>
            </div>
          </div>
        </div>
      </div>
    </div>
  </div>

```



```
<li><a href="#">Bride Fashion Store</a></li>
<li><a href="#">Groom Fashion</a></li>
<li><a href="#">Men Fashion Store</a></li>
<li><a href="#">Kids Fashion</a></li><br>
<li><a class="btn btn-secondary btn-sm" style="margin-left:auto;">View All</a></li>
</ul>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<div class="col-md-9">
```

```
<div class="row pb-4">
```

```
<div class="col-md-4">
```

```
<div class="card1 p-2">
```

```
<div class="row">
```

```
<div class="col-md-4">
```

```

```

```
</div>
```

```
<div class="col-md-8">
```

```
<p class="card-title font-weight-bold">Bride Fashion Store</p>
```

```
<p><a href="#">Bridal Dresses</a></p>
```

```
<p><a href="#">Best Quality Dresses</a></p>
```

```
<p><a href="#">Best Quality Bridal Dresses</a></p>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<div class="col-md-4">
```

```
<div class="card1 p-2">
```

```
<div class="row">
```

```
<div class="col-md-4">
```

```

```

```
</div>
```

```
<div class="col-md-8">
```

```
<p class="card-title font-weight-bold">Women Fashion Store</p>
```

```
<p><a href="#">Best Bridal Dresses</a></p>
```

```
<p><a href="#">Best Quality Bridal Dresses</a></p>
```

```
<p><a href="#">Best Quality Bridal</a></p>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<div class="col-md-4">
```

```
<div class="card1 p-2">
```

```
<div class="row">
```

```
<div class="col-md-4">
```

```

```

```
</div>
```

```
<div class="col-md-8">
```

```
<p class="card-title font-weight-bold">Groom Fashion Store</p>
```

```
<p><a href="#">Best Quality Bridal</a></p>
```

```
<p><a href="#">Best Quality Bridal Dresses</a></p>
```

```

    <p><a href="#">Best QualityDresses</a></p>
  </div>
</div>
</div>
  </div>
</div>

<div class="row pb-4">
<div class="col-md-4">
  <div class="card1 p-2">
  <div class="row">
    <div class="col-md-4">
      
    </div>
    <div class="col-md-8">
      <p class="card-title font-weight-bold">Women Fashion Store</p>
      <p><a href="#">Best Quality Dresses</a></p>
      <p><a href="#">Best Quality Bridal Dresses</a></p>
      <p><a href="#">Best Bridal Dresses</a></p>
    </div>
  </div>
</div>
</div>
  </div>

  <div class="col-md-4">
  <div class="card1 p-2">
  <div class="row">
    <div class="col-md-4">
      
    </div>
    <div class="col-md-8">
      <p class="card-title font-weight-bold">Men Fashion Store</p>
      <p><a href="#">Best Quality Bridal Dresses</a></p>
      <p><a href="#">Quality Bridal Dresses</a></p>
      <p><a href="#">Best Quality Bridal</a></p>
    </div>
  </div>
</div>
</div>
  </div>

  <div class="col-md-4">
  <div class="card1 p-2">
  <div class="row">
    <div class="col-md-4">
      
    </div>
    <div class="col-md-8">
      <p class="card-title font-weight-bold">Kids Fashion Store</p>
      <p><a href="#">Best Bridal Dresses</a></p>
      <p><a href="#">Best Quality Bridal Dresses</a></p>
      <p><a href="#">Best Quality Bridal</a></p>
    </div>
  </div>
</div>
</div>
  </div>

```



```
</div>

</section>

<!-- Footer -->
<footer class="page-footer font-small unique-color-dark">

<div style="background-color: #080A52;">
  <div class="container">

    <!-- Grid row-->
    <div class="row py-4 d-flex align-items-center">

      <!-- Grid column -->
      <div class="col-md-6 col-lg-5 text-center text-md-left mb-4 mb-md-0">
        <h6 class="mb-0">Get connected with us on social networks!</h6>
      </div>
      <!-- Grid column -->

      <!-- Grid column -->
      <div class="col-md-6 col-lg-7 text-center text-md-right">

        <!-- Facebook -->
        <a class="fb-ic">
          <i class="fab fa-facebook-f white-text mr-4"> </i>
        </a>
        <!-- Twitter -->
        <a class="tw-ic">
          <i class="fab fa-twitter white-text mr-4"> </i>
        </a>
        <!--Linkedin -->
        <a class="li-ic">
          <i class="fab fa-linkedin-in white-text mr-4"> </i>
        </a>
        <!--Instagram-->
        <a class="ins-ic">
          <i class="fab fa-instagram white-text"> </i>
        </a>

      </div>
      <!-- Grid column -->

    </div>
    <!-- Grid row-->

  </div>
</div>

<!-- Footer Links -->
<div class="container text-center text-md-left mt-5">

  <!-- Grid row -->
  <div class="row mt-3">

    <!-- Grid column -->
```

```

<div class="col-md-2 col-lg-2 col-xl-2 mx-auto mb-4">
    
</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-md-3 col-lg-3 col-xl-3 mx-auto mb-4">

    <!-- Links -->
    <h6 class="text-uppercase font-weight-bold">Contact Us</h6>
    <hr class="deep-purple accent-2 mb-4 mt-0 d-inline-block mx-auto" style="width:
60px;">
    <p>
        <i class="fas fa-envelope mr-3"></i> admin@dateorwed.com</p>
    <p>
        <i class="fas fa-phone mr-3"></i> +91 - 7893642582</p>

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-md-2 col-lg-2 col-xl-2 mx-auto mb-4">

    <!-- Links -->
    <h6 class="text-uppercase font-weight-bold">Users</h6>
    <hr class="deep-purple accent-2 mb-4 mt-0 d-inline-block mx-auto" style="width:
40px;">
    <p>
        <a href="#">My Profile</a>
    </p>

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-md-2 col-lg-2 col-xl-2 mx-auto mb-md-0 mb-4">

    <!-- Links -->
    <h6 class="text-uppercase font-weight-bold">About Us</h6>
    <hr class="deep-purple accent-2 mb-4 mt-0 d-inline-block mx-auto" style="width:
60px;">
    <p>
        <a href="#">About Us</a></p>
    <p>
        <a href="#">FAQs</a></p>
    <p>
        <a href="#">Feedback</a></p>

</div>
<!-- Grid column -->
<!-- Grid column -->
<div class="col-md-3 col-lg-3 col-xl-3 mx-auto mb-md-0 mb-4">

```

```

<!-- Links -->
<h6 class="text-uppercase font-weight-bold">SERVICE PROVIDERS</h6>
<hr class="deep-purple accent-2 mb-4 mt-0 d-inline-block mx-auto" style="width:
60px;">
<p>
  <a href="#">Become a Service Provider</a></p>
<p>
  <a href="#">Dashboard</a></p>
<p>
  <a href="#">Emergency Contacts</a></p>

</div>
<!-- Grid column -->

</div>
<!-- Grid row -->
<div class="border-bottom border-light"></div>
<!-- Grid row -->
<div class="row mt-3">

  <!-- Grid column -->
  <div class="col-md-3 col-lg-4 col-xl-3 mx-auto mb-4">

    <!-- Content -->
    <h6 class="text-uppercase font-weight-bold">Top Business Listings</h6>
    <hr class="deep-purple accent-2 mb-4 mt-0 d-inline-block mx-auto" style="width:
100px;">
    <p>
      <a href="#">Shopping Vendors</a></p>
    <p>
      <a href="#">Bride </a></p>
    <p>
      <a href="#">Men </a></p>

  </div>
  <!-- Grid column -->

  <!-- Grid column -->
  <div class="col-md-2 col-lg-2 col-xl-2 mx-auto mb-4">

    <!-- Links -->
    <h6 class="text-uppercase font-weight-bold">&nbsp;</h6>
    <hr class="deep-purple accent-2 mb-4 mt-0 d-inline-block mx-auto border-0"
style="width: 60px;">
    <p>
      <a href="#">Women</a>
    </p>
    <p>
      <a href="#">Kids</a>
    </p>
    <p>
      <a href="#">TV & Appliances</a>
    </p>

  </div>

```

```

<!-- Grid column -->

<!-- Grid column -->
<div class="col-md-3 col-lg-2 col-xl-2 mx-auto mb-4">

  <!-- Links -->
  <h6 class="text-uppercase font-weight-bold">&nbsp;</h6>
  <hr class="deep-purple accent-2 mb-4 mt-0 d-inline-block mx-auto border-0"
style="width: 60px;">
  <p>
    <a href="#">Electronics</a>
  </p>
  <p>
    <a href="#">Furniture & Furnishing</a>
  </p>
  <p>
    <a href="#">Home & Living</a>
  </p>

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-md-4 col-lg-3 col-xl-3 mx-auto mb-md-0 mb-4">

  <!-- Links -->
  <h6 class="text-uppercase font-weight-bold">&nbsp;</h6>
  <hr class="deep-purple accent-2 mb-4 mt-0 d-inline-block mx-auto border-0"
style="width: 60px;">
  <p>
    <a href="#">Metal Appliances</a>
  </p>
  <p>
    <a href="#">Wedding Material</a>
  </p>
  <p>
    <a href="#">Jewelry</a>
  </p>

</div>
<!-- Grid column -->

</div>
<!-- Grid row -->

</div>
<!-- Footer Links -->

<!-- Copyright -->
<div class="footer-copyright text-center py-3">© 2019 Copyright
  <a href="Home.aspx"> Date Or Wed</a> All Rights Reserved
</div>
<!-- Copyright -->

</footer>

```

```
<!-- Footer -->
```

```
<div id="btnTop" class="fixed-action-btn smooth-scroll" style="bottom: 45px; right: 24px; display: none;">  
  <a href="#topSection" class="btn-floating btn-large red">  
    <i class="fa fa-arrow-up"></i>  
  </a>  
</div>
```

```
<!-- Start your project here-->
```

```
<!-- SCRIPTS -->
```

```
<!-- JQuery -->
```

```
<script type="text/javascript" src="js/jquery-3.4.1.min.js"></script>
```

```
<!-- Bootstrap tooltips -->
```

```
<script type="text/javascript" src="js/popper.min.js"></script>
```

```
<!-- Bootstrap core JavaScript -->
```

```
<script type="text/javascript" src="js/bootstrap.min.js"></script>
```

```
<!-- MDB core JavaScript -->
```

```
<script type="text/javascript" src="js/mdb.min.js"></script>
```

```
<script type="text/javascript">
```

```
import { mdbContainer, mdbRow, mdbIcon, mdbNavbar, mdbNavbarBrand, mdbNavbarToggler, mdbNavbarNav, mdbNavItem, mdbInput, mdbDropdown, mdbDropdownToggle, mdbDropdownMenu, mdbDropdownItem } from 'mdbvue';
```

```
export default {
```

```
  name: 'MegaMenuPage',
```

```
  components: {
```

```
    mdbContainer,
```

```
    mdbRow,
```

```
    mdbIcon,
```

```
    mdbNavbar,
```

```
    mdbNavbarBrand,
```

```
    mdbNavbarToggler,
```

```
    mdbNavbarNav,
```

```
    mdbNavItem,
```

```
    mdbInput,
```

```
    mdbDropdown,
```

```
    mdbDropdownToggle,
```

```
    mdbDropdownMenu,
```

```
    mdbDropdownItem
```

```
  }
```

```
};
```

```
</script>
```

```
<script type="text/javascript">
```

```
$(function() {
```

```
  var $btn = $('#btnTop');
```

```
  var $home = $('#topSection');
```

```
  var startpoint = $home.scrollTop() + $home.height();
```

```
$(window).on('scroll', function() {
```

```
  if($(window).scrollTop() > startpoint) {
```

```
    $btn.show();
```

```
  } else {
```

```
    $btn.hide();
```



```

    }
  });
});
</script>
</form>
</body>

```

```
</html>
```

```
***** Index.aspx.cs *****
```

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBAL;
using DowBEntity;
using System.Text;

```

```
public partial class Index : System.Web.UI.Page
```

```

{
    DDLogTracer objlog = new DDLogTracer();

    protected void Page_Load(object sender, EventArgs e)
    {
        var sb = new StringBuilder();
        int srno = 0;
        List<EteMenuList> Lst = new tblPagess().GetMenuListbystype(1);//new
List<EteMenuList>();//
        var lst = (from c in Lst where c.Mlevel == 1 orderby c.priority select c);
        foreach (var dt in lst)
        {
            srno++;
            sb.Append("  <a href='#menu" + dt.pageid + "' class='list-group-item collapsed'
data-toggle='collapse' data-parent='#sidebar' aria-expanded='false'> <span class='hidden-
sm-down'><i class='" + dt.Btnlmg + "'></i> " + dt.pagename + " <i class='fas fa-chevron-right'
style='font-size: 12px;'></i></span></a>\n");
            sb.Append("<div class='collapse' id='menu" + dt.pageid + "'>\n");
            var data = (from d in Lst where d.parentid == dt.pageid orderby d.priority select d);
            foreach (var data1 in data)
            {
                sb.Append("  <a href='#menu" + data1.pageid + "' class='list-group-item
collapsed' data-toggle='collapse' data-parent='#sidebar' aria-expanded='false'> <span
class='hidden-sm-down'><i class='" + data1.Btnlmg + "'></i> " + data1.pagename + " <i
class='fas fa-chevron-right' style='font-size: 12px;'></i></span></a>\n");
                sb.Append("<div class='collapse' id='menu" + data1.pageid + "'>\n");
                var prod = (from d in Lst where d.parentid == data1.pageid orderby d.priority select
d);
                foreach (var prd in prod)
                {
                    sb.Append("<a href=' ShoppingList.aspx?pgid=" + prd.pageid + "' class='list-
group-item' data-parent='#menu" + prd.pageid + "'><i class='fas " + prd.Btnlmg + "'></i> " +
prd.pagename + " </a>\n");
                }
                sb.Append("</div>\n");
            }
        }
    }
}

```

```

    }
    sb.Append("</div>\n");

}
menu.InnerHtml = sb.ToString();
objlog.Trace_Error(5, "DateOrWed", "services", "pageLoad", sb.ToString());

}

protected void Button1_Click(object sender, EventArgs e)
{
    //string Soption = ddl_location.SelectedValue;
    //string svalue = txt_SearchString.Text;

}
}
}
***** Login.aspx *****
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Login.aspx.cs"
Inherits="Login" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head>

    <link href="css/font-awesome.css" type="text/css" rel="stylesheet"/>
    <link href="css/font-awesome.min.css" type="text/css" rel="stylesheet"/>
    <link rel="stylesheet" type="text/css" href="css/bootstrap.css"/>
    <link rel="stylesheet" type="text/css" href="css/bootstrap.min.css"/>
    <link rel="stylesheet" type="text/css" href="css/style.css"/>

    <link href="css/login-style.css" type="text/css" rel="stylesheet"/>

    <link rel="stylesheet" type="text/css" href="css/loginstyle.css"/>

    <script type="text/javascript" src="js/jquery-1.6.1.min.js"></script>

    <script type="text/javascript" src="js/ddsmoothmenu.js"></script>
        <script type="text/javascript" src="js/bootstrap.js"></script>
    <script type="text/javascript" src="js/bootstrap.min.js"></script>
    <link href="css/menu.css" type="text/css" rel="stylesheet"/>

<meta name="viewport" content="width=device-width, initial-scale=1.0" />
</head>

<body>
    <form id="form1" class="box login" runat="server">
        <div>
            <div>

```

```
<div class="Ecgl" style="text-align:center; padding:20px 0px 20px 0px;">
```

```

```

```
<div style="text-align: left;width:30em;margin: 2em auto 3em auto;">
```

```
    <fieldset class="boxBody">
<table class="fields">
  <tbody><tr>
    <td class="editor-label" style="width:8em;">
      User Name:
    </td>
  </tr>
  <tr>
    <td>
      <asp:TextBox ID="txtuname" runat="server"></asp:TextBox>
    </td>
  </tr>
  <tr>
    <td class="editor-label" style="width:8em; padding-top:10px;">
      Password:
    </td>
  </tr>
  <tr>
    <td>
      <asp:TextBox ID="txtpassword" runat="server" MaxLength="50"
      TextMode="Password">admin</asp:TextBox>

      <asp:ImageButton ID="ImageButton1" CssClass="pull-right" runat="server"
      style="margin-top:20px;" ImageUrl="~/images/LogIn.png" OnClick="ImageButton1_Click" />
      <br />
      <asp:Label ID="lblStatus" runat="server" CssClass="login_message"
      Visible="false" Text=""></asp:Label>
    </td>
  </tr>
</tbody></table>

    <div class="validation-summary-valid" id="validationSummary"><ul><li
    style="display:none"></li>
</ul></div>
  </fieldset>

<script type="text/javascript">

    <![CDATA[
      if (!window.mvcClientValidationMetadata) {
        window.mvcClientValidationMetadata = []; }
        window.mvcClientValidationMetadata.push({ "Fields": [], "FormId": "form0",
"ReplaceValidationSummary": true, "ValidationSummaryId": "validationSummary" });
```

```

//]]>
</script>

</div>
</div>

<script type="text/javascript">
    document.getElementById("userName").focus();
</script>

</div>

</div>
</form>

<div id="footer" style="text-align:center; position:relative; top:750px; margin-left:15px;">
    BMS Version 1.39.10. <a
href="http://portaltest.DateOrWed.com:699/Heartbeat">Heartbeat</a>. Copyright © 2013
ECCGPLC.
</div>

</body>
</html>
***** Login.aspx.cs *****
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBAL;
using System.Data;
using DowDAL;
using DowBEntity;
using System.Web.Security;
public partial class Login : System.Web.UI.Page
{

    public static string _Country;
    public static string _City;
    protected void Page_Load(object sender, EventArgs e)
    {

        Session["PageType"] = "";

    }

    //[System.Web.Services.WebMethod]
    public static string GetCountrynCity(string City, string Country)

```

```

{
    _City = City;
    _Country = Country;
    return "";
}

protected void Login_Click(object sender, ImageClickEventArgs e)
{
    string City = "London";// _City;
    string Country = "U.K.");// _Country;
    BALCreateUser objlogin = new BALCreateUser();
    DataTable status_login = objlogin.Login(txtuname.Text.Trim(), txtpassword.Text.Trim(),
Request.ServerVariables["REMOTE_ADDR"].ToString(), "", City, Country);
    string login_status = status_login.Rows[0]["UserStatus"].ToString();
    Session["page"] = "1";
    ClsLoggedInUser objLoggedInUser;
    if (login_status == "True")
    {
        objLoggedInUser = new ClsLoggedInUser();

        objLoggedInUser.UserId = status_login.Rows[0]["UserId"].ToString();
        objLoggedInUser.UserName = status_login.Rows[0]["UserName"].ToString();
        objLoggedInUser.GroupId = status_login.Rows[0]["GroupId"].ToString();

        objLoggedInUser.LoginID = Convert.ToInt32(status_login.Rows[0]["LoginId"]);
        objLoggedInUser.DealerId = Convert.ToInt32(status_login.Rows[0]["DealerId"]);
        objLoggedInUser.RoleId = Convert.ToInt32(status_login.Rows[0]["RoleId"]);
        objLoggedInUser.BrandId = Convert.ToInt32(status_login.Rows[0]["BrandId"]);

        Session["LoggedInUser_Specs"] = objLoggedInUser;
        Session["AdminCode"] = "0";

        FormsAuthentication.SetAuthCookie(objLoggedInUser.UserName, false);
        ClsPermissioGroup clsperGrp = new ClsPermissioGroup();
        UserPermissions Entuserper =
clsperGrp.GetUserLandingPage(objLoggedInUser.UserName);

        if (Entuserper.PagId != 0)
        {
            BalRecentPlacess objRemoveRcntPlc = new BalRecentPlacess();
            bool rst =
objRemoveRcntPlc.DeleteRecentPlaces(Convert.ToInt32(status_login.Rows[0]["UserId"].To
String()));
            Response.Redirect("~/ " + Entuserper.PageUrl);
        }
        else
        {
            lblStatus.Visible = true;
            lblStatus.Text = "No User Permissions for the user. Please contact Admin.";
        }
    }
    else
    {
        lblStatus.Visible = true;
        lblStatus.Text = status_login.Rows[0]["ResultStatus"].ToString();
    }
}

```

```

    }
}
protected void ImageButton2_Click(object sender, ImageClickEventArgs e)
{
    Response.Redirect("~/ManageUser.aspx");
}
protected void ImageButton1_Click(object sender, ImageClickEventArgs e)
{
    string City = "Hyderabad "; // _City;
    string Country = "India"; // _Country;
    string password = "";
    BALCreateUser objlogin = new BALCreateUser();
    password = txtpassword.Text.Trim();
    //password = new BalCommon().EncryptPwd(txtpassword.Text.Trim());

    DataTable status_login = objlogin.Login(txtuname.Text.Trim(), password,
Request.ServerVariables["REMOTE_ADDR"].ToString(), "", City, Country);
    string login_status = status_login.Rows[0]["UserStatus"].ToString();

    ClsLoggedInUser objLoggedInUser;
    if (login_status == "True")
    {
        objLoggedInUser = new ClsLoggedInUser();

        objLoggedInUser.UserId = status_login.Rows[0]["UserId"].ToString();
        objLoggedInUser.UserName = status_login.Rows[0]["UserName"].ToString();
        objLoggedInUser.GroupId = status_login.Rows[0]["GroupId"].ToString();

        objLoggedInUser.LoginID = Convert.ToInt32(status_login.Rows[0]["LoginId"]);
        objLoggedInUser.DealerId = Convert.ToInt32(status_login.Rows[0]["DealerId"]);
        objLoggedInUser.RoleId = Convert.ToInt32(status_login.Rows[0]["RoleId"]);
        objLoggedInUser.BrandId = 1; //Convert.ToInt32(status_login.Rows[0]["BrandId"]);

        Session["LoggedUser_Specs"] = objLoggedInUser;

        FormsAuthentication.SetAuthCookie(objLoggedInUser.UserName, false);
        HttpCookie myCookie = new HttpCookie("DateOrWedCookie");
        DateTime now = DateTime.Now;
        myCookie.Value = objLoggedInUser.UserName;
        myCookie.Expires = now.AddYears(1);

        Response.Cookies.Add(myCookie);

        ClsPermissioGroup clsperGrp = new ClsPermissioGroup();
        UserPermissions Entuserper =
clsperGrp.GetUserLandingPage(objLoggedInUser.UserName);

        if (Entuserper.PagId != 0)
        {
            Entuserper.Add = true;
            Entuserper.Edit = true;
            Entuserper.Delete = true;
            Entuserper.View = true;
        }
    }
}

```

```

        //BalRecentPlacess objRemoveRcntPlc = new BalRecentPlacess();
        //bool rst =
objRemoveRcntPlc.DeleteRecentPlaces(Convert.ToInt32(status_login.Rows[0]["UserId"].ToString()));
        string page =new tblUsers().getuserpage(objLoggedInUser.UserName);
        //Response.Redirect("~/ " + Entuserper.PageUrl);
        if (page == null)
        {
            Response.Redirect("~/ " + Entuserper.PageUrl);
        }
        else
        {
            Response.Redirect("~/ " + page);
        }

    }
    else
    {
        lblStatus.Visible = true;
        lblStatus.Text = "No User Permissions for the user. Please contact Admin.";
    }
}
else
{
    lblStatus.Visible = true;
    lblStatus.Text = status_login.Rows[0]["ResultStatus"].ToString();
}
}
protected void Button1_Click(object sender, EventArgs e)
{
    Response.Redirect("~/ManageUser.aspx");
}
protected void Button2_Click(object sender, EventArgs e)
{
    string City = "Hyderabad";// _City;
    string Country = "India";// _Country;
    string password = "";
    BALCreateUser objlogin = new BALCreateUser();
    password = txtpassword.Text.Trim();
    //password = new BalCommon().EncryptPwd(txtpassword.Text.Trim());

    DataTable status_login = objlogin.Login(txtuname.Text.Trim(), password,
Request.ServerVariables["REMOTE_ADDR"].ToString(), "", City, Country);
    string login_status = status_login.Rows[0]["UserStatus"].ToString();

    ClsLoggedInUser objLoggedInUser;
    if (login_status == "True")
    {
        objLoggedInUser = new ClsLoggedInUser();

        objLoggedInUser.UserId = status_login.Rows[0]["UserId"].ToString();
        objLoggedInUser.UserName = status_login.Rows[0]["UserName"].ToString();
        objLoggedInUser.GroupId = status_login.Rows[0]["GroupId"].ToString();
    }
}

```

```

objLoggedInUser.LoginID = Convert.ToInt32(status_login.Rows[0]["LoginId"]);
objLoggedInUser.DealerId = Convert.ToInt32(status_login.Rows[0]["DealerId"]);
objLoggedInUser.RoleId = Convert.ToInt32(status_login.Rows[0]["RoleId"]);
objLoggedInUser.BrandId = 1; //Convert.ToInt32(status_login.Rows[0]["BrandId"]);

Session["LoggedUser_Specs"] = objLoggedInUser;

FormsAuthentication.SetAuthCookie(objLoggedInUser.UserName, false);
HttpCookie myCookie = new HttpCookie("DateOrWedCookie");
DateTime now = DateTime.Now;
myCookie.Value = objLoggedInUser.UserName;
myCookie.Expires = now.AddYears(1);

Response.Cookies.Add(myCookie);
Session["AdminCode"] = "0";
Session["VendorCode"] = "0";
Session["ServiceCode"] = "0";
Session["BLCode"] = "0";

ClsPermissioGroup clsperGrp = new ClsPermissioGroup();
UserPermissions Entuserper =
clsperGrp.GetUserLandingPage(objLoggedInUser.UserName);

if (Entuserper.Pageld != 0)
{
    Entuserper.Add = true;
    Entuserper.Edit = true;
    Entuserper.Delete = true;
    Entuserper.View = true;
    //BalRecentPlacess objRemoveRcntPlc = new BalRecentPlacess();
    //bool rst =
objRemoveRcntPlc.DeleteRecentPlaces(Convert.ToInt32(status_login.Rows[0]["UserId"].ToString()));
    string page = new tblUsers().getuserpage(objLoggedInUser.UserName);
    //Response.Redirect("~/ " + Entuserper.PageUrl);
    if (page == null)
    {
        Response.Redirect("~/ " + Entuserper.PageUrl);
    }
    else
    {
        Response.Redirect("~/ " + page);
    }
}
else
{
    lblStatus.Visible = true;
    lblStatus.Text = "No User Permissions for the user. Please contact Admin.";
}
}
else
{
    lblStatus.Visible = true;
}

```



```

        lblStatus.Text = status_login.Rows[0]["ResultStatus"].ToString();
    }

}
}
}
***** logout.aspx *****
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="logout.aspx.cs"
Inherits="logout" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
</head>
<body>
    <form id="form1" runat="server">
    <div>

        </div>
    </form>
</body>
</html>
***** logout.aspx.cs *****
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

using System.IO;
using DowBEntity;
using DowBAL;
using System.Web.Security;

public partial class logout : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        if (HttpContext.Current.Session["LoggedUser_Specs"] != null)
        {
            ClsLoggedInUser Clsusr =
(ClsLoggedInUser)HttpContext.Current.Session["LoggedUser_Specs"];
            BALCreateUser objbal = new BALCreateUser();
            objbal.LogOut(Convert.ToInt32(Clsusr.UserId), Convert.ToInt32(Clsusr.LoginID));
            HttpContext.Current.Session.Abandon();
            Session.Clear();
            Session.RemoveAll();
            Session.Abandon();

            HttpCookie myCookie = new HttpCookie("DateOrWedCookie");

```

```

DateTime now = DateTime.Now;
myCookie.Value = "";
myCookie.Expires = now.AddYears(-1);
Response.Cookies.Clear();
Response.Cookies.Add(myCookie);

Response.Redirect("~/MLB_Login.aspx");
FormsAuthentication.SignOut();
}
else
{
    HttpContext.Current.Session.Abandon();
    Response.Redirect("~/MLB_Login.aspx");
    FormsAuthentication.SignOut();
}
}
}
}
}
***** ManageVatCodes.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="ManageVatCodes.aspx.cs" Inherits="ManageVatCodes"
%>

<%--<%@ Register Assembly="Infragistics4.Web.v14.2, Version=14.2.20142.1028,
Culture=neutral, PublicKeyToken=7dd5c3163f2cd0cb"
Namespace="Infragistics.Web.UI.EditorControls" TagPrefix="ig" %>--%>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
    <script type="text/javascript">
        function checkTextAreaMaxLength(textBox, e, length) {

            var mLen = textBox["MaxLength"];
            if (null == mLen)
                mLen = length;

            var maxLength = parseInt(mLen);
            if (!checkSpecialKeys(e)) {
                if (textBox.value.length > maxLength - 1) {
                    if (window.event)//IE
                        e.returnValue = false;
                    else
                        e.preventDefault();
                }
            }
        }

    }

function isNumberKey(evt) {
    var charCode = (evt.which) ? evt.which : event.keyCode;
    if (charCode != 46 && charCode > 31 && (charCode < 48 || charCode > 57)) {
        return false;
    } else {
        // If the number field already has . then don't allow to enter . again.

```



```

customertable">
    <div style="background-color: #f8f9fa;" class="span12
    <div class="row-fluid">
        <div style="background-color: #ffffff; border: 1px solid #eefef;
border-top: none; padding: 15px 5px;" class="span12 detailview">
            <div class="control-group">
                <table id="Table5" class="table table-striped table-detail
span12" runat="server">
                    <tbody>
                        <tr>
                            <td width="14%" align="right" class="label_txt"
dir="auto">Vat Code:</td>
                            <td>
                                <asp:TextBox ID="txt_vatcode" runat="server"
MaxLength="40" Width="600px"></asp:TextBox>
                            </td>
                        </tr>
                        <tr>
                            <td width="14%" align="right" class="label_txt"
dir="auto">Description :</td>
                            <td>
                                <asp:TextBox ID="txt_descr" runat="server"
></asp:TextBox>
                            </td>
                        </tr>
                        <tr>
                            <td width="14%" align="right" class="label_txt"
dir="auto">Rate :</td>
                            <td>
                                <asp:TextBox ID="txt_rate" runat="server"
></asp:TextBox>
                            </td>
                        </tr>
                        <tr>
                            <td width="14%" align="right" class="label_txt"
dir="auto">Default :</td>
                            <td>
                                <asp:CheckBox ID="chk_default" runat="server" />
                            </td>
                        </tr>
                    </tbody>
                </table>
            </div>
        </div>
    </div>
</div>
<!-- /.content_holder -->

```



```

    {
        Id = Convert.ToInt32(Request.QueryString["Id"]);
        EteVatCode Hold = new EteVatCode();
        Hold = new BalVatCodes().getVatCodeById(Id);
        if (Hold != null)
        {
            txt_vatcode.Text = Hold.Code;
            txt_descr.Text = Hold.Description;
            txt_rate.Text = Hold.Rate.ToString();
            chk_default.Checked = Hold.IsDefault;
        }
        else
        {
            Common.ShowMessageHtml("Missing Vat Code Details ", 0, true, dvMsgBox);
        }
    }
}

private void ManageNavigations(int pageid, int userid, int groupid, string PageType)
{
    bool IsLeftBar = false;
    IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
myBtnClickHandler);

    if (IsLeftBar == true)
    {
        LeftSideBar.Attributes["class"] = "span2";
        MainContent.Attributes["class"] = "span10";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;"); //style="display:
block;"
        PageName.InnerText = new
tblPagess().GettblPageNameById(Convert.ToInt32(pageid));
    }
    else
    {
        LeftSideBar.Attributes["class"] = "";
        MainContent.Attributes["class"] = "span12";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
    }
    if (PageType == "Add")
    {
        PageName.InnerText = "Add VAT Code";
    }
    else
    {
        PageName.InnerText = "Manage VAT Code";
    }
}

// This is Event Handler
protected void myBtnClickHandler(object sender, EventArgs e)

```

```

{
    LinkButton Ink = (LinkButton)sender;
    string ID = Ink.ID;

    DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
    if (ds != null)
    {
        if (ds.Tables[0].Rows.Count > 0)
        {
            DataTable dtdll = ds.Tables[0];
            foreach (DataRow drnew in dtdll.Rows)
            {
                switch (drnew["lbtncode"].ToString())
                {
                    case "SAVE":
                        // Call SAVE Method
                        ContentSaveControlClicked();
                        break;
                    case "AUDIT":
                        Response.Redirect(drnew["pageurl"].ToString() + "&frompage=" +
Request.QueryString["pageid"]);
                        break;
                    case "URL":
                        Response.Redirect(drnew["pageurl"].ToString());
                        break;
                    case "CANCEL":
                        ContentCancelControlClicked();
                        break;
                    default:
                        break;
                }
            }
        }
    }
}

```

```

private void ContentCancelControlClicked()
{
    Common.ShowMessageHtml("", 0, false, dvMsgBox);

    PageType = Session["PageType"].ToString();
    if (PageType == "Add")
    {
    }
    else if (PageType == "Edit")
    {
        Response.Redirect("~/NoteTypesList.aspx?pageid=149");
    }
}

```

```

}
private void ContentSaveControlClicked()
{
    //objlog.Trace_Log(0, "DateOrWed", "MangeDealer", "SaveClick", "Start Here..");

    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    try
    {
        if (txt_vatcode.Text == "")
        {
            Common.ShowMessageHtml("Code cannot be blank", 0, true, dvMsgBox);

        }
        else if (txt_descr.Text == "")
        {
            Common.ShowMessageHtml("Description cannot be blank", 0, true, dvMsgBox);

        }
        else if (txt_rate.Text == "" || Convert.ToDecimal(txt_rate.Text) == 0)
        {
            Common.ShowMessageHtml("Rate cannot be blank", 0, true, dvMsgBox);

        }
        else
        {
            EteVatCode VatCode = new EteVatCode();
            VatCode.Id = Convert.ToInt32(Request.QueryString["Id"]);
            VatCode.Code = txt_vatcode.Text;
            VatCode.Description = txt_descr.Text;
            VatCode.CreatedOn = DateTime.Now;
            VatCode.IsDefault = chk_default.Checked;
            VatCode.UId = Convert.ToInt32(Clsusr.UserId);
            VatCode.IsActive = true;
            VatCode.IsActive = true;
            VatCode.Rate = Convert.ToDecimal(txt_rate.Text);
            bool Result = false;
            PageType = Session["PageType"].ToString();

            if (PageType == "Add")
            {
                Result = new BalVatCodes().Insert(VatCode);
            }
            else
            {
                Result = new BalVatCodes().Update(VatCode);
            }
            if (Result == true)
            {
                if (PageType == "Add")
                {

```



```

        Common.ShowMessageHtml("Adding of Vat Code Sucessful ", 1, true,
dvMsgBox);
    }
    else
    {
        Common.ShowMessageHtml("Vat Code Edited Sucessfully ", 1, true,
dvMsgBox);
    }
    Response.Redirect("~/VatCodeList.aspx?pageid=154&msg=Vat Code Saved
Sucessfully");
}
}
}

catch (Exception exc)
{
    //objlog.Trace_Error(4, "DateOrWed", "MangeDealer", "Load", "Exception:" +
exc.ToString());
    Common.ShowMessageHtml(exc.ToString(), 0, true, dvMsgBox);
}
//objlog.Trace_Log(0, "DateOrWed", "MangeDealer", "SaveClick", "Start Here..");
}
}

```

```

}
***** ManageDealer.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="ManageDealer.aspx.cs" Inherits="View_ManageDealer"
%>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
    <style type="text/css">
        .auto-style1 {
            height: 22px;
        }
    </style>
</asp:Content>

<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
    <div class="container-fluid">
        <div class="row-fluid" style="padding-top: 20px;">
            <div class="span2" id="dvlogo" runat="server" style="display: block;">
                <%----%>
            </div>
            <div class="span10">
                <div class="row">

```

```

        <div id="dvMsgBox" class="span12 ordersave" runat="server" style="display:
none;">
            <asp:Label ID="Label2" runat="server" Text=""></asp:Label>
        </div>
    </div>
</div>
</div>
</div>
</div>
</asp:Content>

<asp:Content ID="Content4" ContentPlaceHolderID="CPH_LeftSideBar" runat="Server">
    <div id="LeftSideBar" runat="server" class="span2">
        <asp:Table ID="Table1" runat="server">
            </asp:Table>
        </div>
    </asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="CPH_MainContent" runat="Server">

    <div class="span10" id="MainContent" runat="server">
        <div id="ShowPageHeader" class="row" style="display: block;" runat="server">
            <div id="PageName" class="span12 customer" runat="server">
                </div>
            </div>

            <div id="Div1" class="row" style="display: block;">
                <div id="Div2" class="span12 customertable">
                    <div class="tabs_holder tabs_wrapper skin7 top" style="display: block; overflow:
hidden; position: relative;">

                        <div class="content_holder">
                            <div style="overflow: hidden; position: relative; z-index: 1; padding: 10px
0px;" class="content_holder_inner">
                                <div id="your-tab-id-1" class="tab_content" style="display: block;">
                                    <div class="inner_content">
                                        <div style="background-color: #f8f9fa;" class="span12
customertable">

                                            <div class="row-fluid">
                                                <div style="background-color: #ffffff; border: 1px solid #eefef;
border-top: none; padding: 15px 5px;" class="span12 detailview">

                                                    <div class="control-group">
                                                        <table id="DealerDet" class="table table-striped table-detail
span12" runat="server">

                                                            <tbody>
                                                                <tr>
                                                                    <td width="14%" align="right"
class="label_txt">Dealer Name : * </td>
                                                                    <td width="25%" align="left">
                                                                        <asp:TextBox ID="txt_dealername"
runat="server" MaxLength="40"></asp:TextBox>
                                                                    </td>
                                                                </tr>
                                                            </tbody>
                                                        </table>
                                                    </div>
                                                </div>
                                            </div>
                                        </div>
                                    </div>
                                </div>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>

```

```

        <tr>
            <td align="right" class="label_txt">User Prefix :</td>
            <td width="25%" align="left">
                <asp:TextBox ID="txt_UserNamePrefix"
runat="server" MaxLength="40"></asp:TextBox>
            </td>
        </tr>
        <tr>
            <td align="right" class="label_txt">EMail String
: </td>
            <td width="25%" align="left">
                <asp:TextBox ID="txt_EMailString" runat="server"
MaxLength="40">TechSupport@eze-talk.com</asp:TextBox>
            </td>
        </tr>
        <tr>
            <td align="right" class="label_txt">Dealer IsActive
: </td>
            <td width="25%" align="left">
                <asp:CheckBox ID="chk_IsActive" runat="server"
Text="Active" Visible="true" />
            </td>
        </tr>
        <tr>
            <td align="right" class="auto-style1">Operation
Mode: * </td>
            <td class="auto-style1">
                <asp:DropDownList ID="ddl_OperationMode"
runat="server">
                    <asp:ListItem
Value="1">Standard</asp:ListItem>
                    <asp:ListItem
Value="2">SmoothTelecom</asp:ListItem>
                </asp:DropDownList>
            </td>
        </tr>
        <tr>
            <td height="5">Select Brands : *
            </td>
            <td>
                <asp:CheckBoxList ID="chkBoxList"
runat="server" Width="40%" AutoPostBack="True"
ViewStateMode="Enabled"></asp:CheckBoxList>
                <asp:DropDownList ID="ddl_brand"
runat="server" Visible="False"></asp:DropDownList>
                <asp:ListBox ID="ListBrand" runat="server"
Height="162px" Rows="5" OnSelectedIndexChanged="ListBrand_SelectedIndexChanged"
Width="122px" Visible="False" AutoPostBack="True"></asp:ListBox>
                <asp:TextBox ID="txt_id" runat="server"
Visible="False"></asp:TextBox>
            </td>
        </tr>
    </td>

```



```

ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
EteDealer lst = new EteDealer();
if (Clsusr != null)
{
    PageType = Page.Request.QueryString["PageType"];
    if (PageType == null)
    {
        PageType = "Add";
    }
    Session["PageType"] = PageType;

    pageid = "4";// Request.QueryString["pageid"];
    ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
Convert.ToInt32(Clsusr.GroupId), PageType);
    //if (!IsPostBack)
    //{
    if (Convert.ToInt32(Clsusr.GroupId) != 1)
    {
        //Disabel the screen entry
    }
}

// objlog.Trace_Log(0, "DateOrWed", "MangeDealer", "Load", "Start Here..");
try
{
    //ListBrand.SelectionMode = ListSelectionMode.Single;
    DealerDet.Disabled = false;
    dealerid = Convert.ToInt32(Page.Request.QueryString["dealerid"]);
    //ListBrand.SelectionMode = ListSelectionMode.Single;
    PopDlIs ObjPD = new PopDlIs();
    //ObjPD.BindDropDowns(ddl_group, (new PopDlIs()).GetRoles(Clsusr.RoleId));
    ObjPD.BindDropDowns(ddl_brand, (new PopDlIs()).GetBrands());

    //ListBrand.Items.Clear();
    // List<EteBrand> lbrand = (new Branding()).getbrands();

//
    int cnt = 0;
    //if (lbrand.Count > 1)
    //{
    // ListBrand.SelectionMode = ListSelectionMode.Multiple;
    //}
    //else
    //{
    // ListBrand.SelectionMode = ListSelectionMode.Single;

    //}
    //foreach (var row in lbrand)
    //{
    // ListBrand.Items.Add(row.BrandName);
    // bid[cnt] = row.Id;
    // cnt++;
    //}

    if (!IsPostBack)
    {

```

```

Fill_CheckBox_List();
chk_IsActive.Checked = true;

if (PageType == "Edit")
{
    lst = (new Dealers()).getdealer(dealerid);
    Id = lst.DealerId;
    txt_id.Text = lst.DealerId.ToString();
    txt_dealname.Text = lst.CompanyName;
    txt_UserNamePrefix.Text = lst.UserNamePrefix;
    txt_EMailString.Text = lst.EMailString;
    Manage_CheckBox_List(Id);
    ddl_OperationMode.Text =
System.Convert.ToString(lst.DealerOperationProfileId);
    if (lst.IsActive == true)
    {
        chk_IsActive.Checked = true;

    }
    else
    {
        chk_IsActive.Checked = false;
    }

    //if (PageType != "Edit")
    //{
    //    List<EteDealerBrand> blst = (new Branding()).getdealerbrand(Id);
    //    foreach (var frow in blst)
    //    {
    //        ddl_brand.SelectedValue = frow.BrandId.ToString();
    //        lst.BrandId = frow.BrandId;
    //        break;
    //    }
    //    foreach (var row in blst)
    //    {
    //        DealerId = row.DealerId;
    //        BrandId = row.BrandId;
    //        ShortCode = row.ShortCode;
    //        BrandName = row.BrandName;
    //        FilePath = row.FilePath;
    //        int Ctr = 0;
    //        do
    //        {
    //            if (BrandId == bid[Ctr])
    //            {
    //                ListBrand.Items[Ctr].Selected = true;
    //            }
    //            Ctr++;
    //        } while (Ctr < cnt);
    //    }
    //}
}
}

```

```

        //else
        //{
        // //txt_dealername.Text = "";
        // //txt_UserNamePrefix.Text = "";
        // //ddl_OperationMode.SelectedIndex = 1;
        //}
        Session["DEALERDATA"] = lst;
    }

    catch (Exception exc)
    {
        objlog.Trace_Error(4, "DateOrWed", "MangeDealer", "Load", "Exception:" +
exc.ToString());
        Common.ShowMessageHtml(exc.ToString(), 0, true, dvMsgBox);

    }
    // objlog.Trace_Log(0, "DateOrWed", "MangeDealer", "Load", "Ends Here..");

    //}
}
else
{
    Response.Redirect("MLB_Login.aspx");
}
}

private void ManageNavigations(int pageid, int userid, int groupid, string PageType)
{
    bool IsLeftBar = false;
    IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
myBtnClickHandler);

    if (IsLeftBar == true)
    {
        LeftSideBar.Attributes["class"] = "span2";
        MainContent.Attributes["class"] = "span10";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;"); //style="display:
block;"
        PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
    else
    {
        LeftSideBar.Attributes["class"] = "";
        MainContent.Attributes["class"] = "span12";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
    }
    if (PageType == "Add")
    {
        PageName.InnerText = "Add Dealer";
    }
    else
    {
        PageName.InnerText = "Edit Dealer";
    }
}

```

```

    }
}

// This is Event Handler
protected void myBtnClickHandler(object sender, EventArgs e)
{
    LinkButton lnk = (LinkButton)sender;
    string ID = lnk.ID;

    DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
    if (ds != null)
    {
        if (ds.Tables[0].Rows.Count > 0)
        {
            DataTable dtdll = ds.Tables[0];
            foreach (DataRow drnew in dtdll.Rows)
            {
                switch (drnew["lbtncode"].ToString())
                {
                    case "SAVE":
                        // Call SAVE Method
                        ContentSaveControlClicked();
                        break;
                    case "AUDIT":
                        Response.Redirect(drnew["pageurl"].ToString() + "&frompage=" +
Request.QueryString["pageid"] + "&Id=" + Request.QueryString["dealerid"]);
                        break;
                    case "URL":
                        Response.Redirect(drnew["pageurl"].ToString());
                        break;
                    case "CANCEL":
                        ContentCancelControlClicked();
                        break;
                    default:
                        break;
                }
            }
        }
    }
}

private void Fill_CheckBox_List()
{
    List<EteBrand> lbrand = (new Branding()).getbrands();
    // ds = ObjDB.Qry_GetDataSet("select * from dbo.Educational_Name");
    chkBoxList.DataSource = lbrand;
    chkBoxList.DataTextField = "BrandName";
    chkBoxList.DataValueField = "id";
}

```



```

        chkBoxList.DataBind();
    }

private void Manage_CheckBox_List(int ID)
{
    //List<EteDealerBrand> blst = (new Branding()).getdealerbrand(ID);
    //foreach (var row in blst)
    //{
    //    //ddl_brand.SelectedValue = row.BrandId.ToString();
    //    foreach (ListItem s in chkBoxList.Items)
    //    {
    //        if (s.Value == row.BrandId.ToString())
    //        {
    //            s.Selected = true;
    //            break;
    //        }
    //    }
    //}

    //List<EteBrand> lbrand = (new Branding()).getbrands();
    /// ds = ObjDB.Qry_GetDataSet("select * from dbo.Educational_Name");
    //chkBoxList.DataSource = lbrand;
    //chkBoxList.DataTextField = "BrandName";
    //chkBoxList.DataValueField = "id";
    //chkBoxList.DataBind();
}

private void ContentCancelControlClicked()
{
    Common.ShowMessageHtml("", 0, false, dvMsgBox);

    PageType = Session["PageType"].ToString();
    if (PageType == "Add")
    {
        txt_dealername.Text = "";
        txt_UserNamePrefix.Text = "";
    }
    else if (PageType == "Edit")
    {
        Response.Redirect("~/DealerList.aspx?pageid=166");
    }
}

private void ContentSaveControlClicked()
{
    // objlog.Trace_Log(0, "DateOrWed", "MangeDealer", "SaveClick", "Start Here..");
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    try
    {
        if (txt_dealername.Text == "")
        {
            Common.ShowMessageHtml("Dealer name cannot be blank", 0, true, dvMsgBox);
        }
    }
}

```

```

else if (txt_UserNamePrefix.Text == "")
{
    txt_UserNamePrefix.Text = txt_dealername.Text;
    //Common.ShowMessageHtml("User Prefix cannot be blank", 0, true, dvMsgBox);

}
else
{
    EteDealer _bmsuser = new EteDealer();
    _bmsuser.CompanyName = txt_dealername.Text;
    _bmsuser.UserNamePrefix = txt_UserNamePrefix.Text;
    int brandid;
    brandid = ListBrand.SelectedIndex;
    _bmsuser.AutoProcessCommissions = true;
    _bmsuser.DealerOperationProfileId =
Convert.ToInt32(ddl_OperationMode.SelectedValue);
    _bmsuser.PagelId = Convert.ToInt32(Request.QueryString["pageid"]);
    _bmsuser.UId = Convert.ToInt32(Clsusr.UserId);
    _bmsuser.EMailString = txt_EmailString.Text;
    _bmsuser.IsActive = chk_IsActive.Checked;
    PageType = Session["PageType"].ToString();
    if (PageType == "Add")
    {
        DealerId = System.Convert.ToInt32((new Dealers()).GetNextDealerId());
        txt_id.Text = DealerId.ToString();
        _bmsuser.DealerId = DealerId;
        _bmsuser.CreatedOn = DateTime.Now;
        bool exists = (new Dealers()).DealerNameExists(_bmsuser.CompanyName);
        if (exists == false)
        {
            sresult = (new Dealers()).Insert(_bmsuser);

        }
    }
    else
    {
        _bmsuser.DealerId = System.Convert.ToInt32(txt_id.Text);
        _bmsuser.UpdatedOn = DateTime.Now;
        sresult = (new Dealers()).Update(_bmsuser);
    }

    //EteDealerBrand _dbrand = new EteDealerBrand();
    //_dbrand.DealerId = System.Convert.ToInt32(txt_id.Text); ;
    //_dbrand.BrandId = Convert.ToInt32(ddl_brand.SelectedValue);

    //bool DelResult = (new
Branding()).DeleteDealerBrand(System.Convert.ToInt32(txt_id.Text));
    List<EteDealerBrand> LstDirBrnad = new List<EteDealerBrand>();
    foreach (ListItem s in chkBoxList.Items)
    {
        if (s.Selected == true)
        {
            EteDealerBrand _dbrand = new EteDealerBrand();
            _dbrand.DealerId = System.Convert.ToInt32(txt_id.Text);

```

```

        _dbrand.BrandId = Convert.ToInt32(s.Value);
        //if (PageType == "Add" && sresult == true)
        //{
        //sresult = (new Branding()).InsertDealerBrand(_dbrand);
        //}
        //else
        //{
        // if (_dbrand.DealerId != 1)
        // {
        //     sresult = (new Branding()).Update(_dbrand);
        // }
        //}

        LstDirBrnad.Add(new EteDealerBrand
        {
            DealerId = System.Convert.ToInt32(txt_id.Text),
            BrandId = Convert.ToInt32(s.Value),
        });
    }
}

if (LstDirBrnad.Count > 0)
{
    sresult = (new Branding()).InsertDealerBrand(LstDirBrnad);
}

//for updating multiple brands for dealers
//List<EteBrand> lbrand = (new Branding()).getbrands();
//int cnt = 0;
//foreach (var row in lbrand)
//{
// ListBrand.Items.Add(row.BrandName);
// bid[cnt] = row.Id;
// cnt++;
//}
// cnt = 0;
//do
//{
// if (ListBrand.Items[cnt].Selected == true)
// {
//     BrandName = ListBrand.Items[cnt].Value;

//     EteDealerBrand _dbrand = new EteDealerBrand();
//     _dbrand.DealerId = DealerId;
//     _dbrand.BrandId = bid[cnt];
//     if (PageType == "Add")
//     {
//         sresult = (new Branding()).InsertDealerBrand(_dbrand);
//     }
//     cnt++;
// }
//} while (cnt < 5);
if (sresult == true)

```

```

        {
            Common.ShowMessageHtml("Record Save Sucessful", 0, true, dvMsgBox);
            Response.Redirect("~/DealerList.aspx?pageid=166&msg=Dealer Saved
Sucessfully");
        }
        else
        {
            Common.ShowMessageHtml("Record Save Un-Sucessful", 0, true, dvMsgBox);
        }
    }
}
}
catch (Exception exc)
{
    objlog.Trace_Error(4, "DateOrWed", "MangeDealer", "Load", "Exception:" +
exc.ToString());
    Common.ShowMessageHtml(exc.ToString(), 0, true, dvMsgBox);
}
// objlog.Trace_Log(0, "DateOrWed", "MangeDealer", "SaveClick", "Start Here..");
}

```

```

protected void ListBrand_SelectedIndexChanged(object sender, EventArgs e)
{
    int brandid;
    brandid = ListBrand.SelectedIndex;
}

```

```

}
***** ManageDelTimeTypes.aspx*****
<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="ManageDelTimeTypes.aspx.cs"
Inherits="ManageDelTimeTypes" %>

<%@ Register Assembly="Infragistics4.Web.v14.2, Version=14.2.20142.1028,
Culture=neutral, PublicKeyToken=7dd5c3163f2cd0cb"
Namespace="Infragistics.Web.Ul.EditorControls" TagPrefix="ig" %>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
    <script type="text/javascript">
        function checkTextAreaMaxLength(textBox, e, length) {

            var mLen = textBox["MaxLength"];
            if (null == mLen)
                mLen = length;

            var maxLength = parseInt(mLen);
            if (!checkSpecialKeys(e)) {
                if (textBox.value.length > maxLength - 1) {

```

```

        if (window.event)//IE
            e.returnValue = false;
        else
            e.preventDefault();
    }
}
}

```

```

function isNumberKey(evt) {
    var charCode = (evt.which) ? evt.which : event.keyCode;
    if (charCode != 46 && charCode > 31 && (charCode < 48 || charCode > 57)) {
        return false;
    } else {
        // If the number field already has . then don't allow to enter . again.
        if (evt.target.value.search(/\./) > -1 && charCode == 46) {
            return false;
        }
        return true;
    }
}

```

```

</script>
</asp:Content>

```

```

<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
    <div class="container-fluid">
        <div class="row-fluid" style="padding-top: 20px;">
            <div class="span2" id="dvlogo" runat="server" style="display: block;">
                <!---->
            </div>
            <div class="span10">
                <div class="row">
                    <div id="dvMsgBox" class="span12 ordersave" runat="server" style="display:
none;">
                        <asp:Label ID="Label1" runat="server" Text=""></asp:Label>
                    </div>
                </div>
            </div>
        </div>
    </div>
</asp:Content>

```

```

<asp:Content ID="Content4" ContentPlaceHolderID="CPH_LeftSideBar" runat="Server">
    <div id="LeftSideBar" runat="server" class="span2">
        <asp:Table ID="Table1" runat="server">
            </asp:Table>
        </div>
    </asp:Content>

```

```

<asp:Content ID="Content2" ContentPlaceHolderID="CPH_MainContent" runat="Server">
    <div class="span10" id="MainContent" runat="server">
        <div id="ShowPageHeader" class="row" style="display: block;" runat="server">
            <div id="PageName" class="span12 customer" runat="server">

```

```

    </div>
</div>

<div id="Div1" class="row" style="display: block;" >
  <div id="Div2" class="span12 customertable">

    <div class="tabs_holder tabs_wrapper skin7 top" style="display: block; overflow:
hidden; position: relative;">

      <div class="content_holder">
        <div style="overflow: hidden; position: relative; z-index: 1; padding: 10px
0px;" class="content_holder_inner">
          <div id="your-tab-id-1" class="tab_content" style="display: block;">
            <div class="inner_content">
              <div style="background-color: #f8f9fa;" class="span12
customertable">
                <div class="row-fluid">
                  <div style="background-color: #ffffff; border: 1px solid #eefef;
border-top: none; padding: 15px 5px;" class="span12 detailview">
                    <div class="control-group">
                      <div class="row-fluid">
                        <div class="span12 customer" style="text-align: left;">Delivery
Time Type Details</div>
                      </div>

                      <table id="Table5" class="table table-striped table-detail
span12" runat="server">

                        <tbody>

                          <tr>

                            <td width="14%" align="right" class="label_txt"
dir="auto">Note Type :</td>
                            <td>
                              <asp:TextBox ID="txt_holdname" runat="server"
MaxLength="40" Width="600px"></asp:TextBox>
                            </td>
                          </tr>
                          <tr>
                            <td width="14%" align="right" class="label_txt"
dir="auto">Delivery Time Type Description :</td>
                            <td>
                              <asp:TextBox ID="Details" runat="server"
></asp:TextBox>
                            </td>
                          </tr>
                        </tbody>
                      </table>

                    </div>
                  </div>
                </div>
              </div>
            </div>
          </div>
        </div>
      </div>
    </div>
  </div>

```



```

        ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
Convert.ToInt32(Clsusr.GroupId), PageType);
    }
    if (!IsPostBack)
    {
        if (PageType=="Edit")
        {
            Id =Convert.ToInt32( Request.QueryString["Id"]);
            Etetbldeltime Hold = new Etetbldeltime();
            Hold = new Products().GetProducttbldeltimebyId(Id);
            if (Hold != null)
            {
                txt_holdname.Text = Hold.DelTimeDescription;
                Details.Text = Hold.DelTimeDescription;
            }
        }
        else
        {
            Common.ShowMessageHtml("Missing Tasks Details ", 0, true, dvMsgBox);
        }
    }
}

private void ManageNavigations(int pageid, int userid, int groupid, string PageType)
{
    bool IsLeftBar = false;
    IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
myBtnClickHandler);
    PageName.InnerText = "Manage Delivery Times";

    if (IsLeftBar == true)
    {
        LeftSideBar.Attributes["class"] = "span2";
        MainContent.Attributes["class"] = "span10";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;"); //style="display:
block;"
    }
    else
    {
        LeftSideBar.Attributes["class"] = "";
        MainContent.Attributes["class"] = "span12";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
    }
}

// This is Event Handler
protected void myBtnClickHandler(object sender, EventArgs e)
{
    LinkButton Ink = (LinkButton)sender;
    string ID = Ink.ID;
}

```



```

ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
try
{
    if (txt_holdname.Text == "")
    {
        Common.ShowMessageHtml("Name cannot be blank", 0, true, dvMsgBox);

    }
    else if (Details.Text == "")
    {

        Common.ShowMessageHtml("Description cannot be blank", 0, true, dvMsgBox);

    }
    else
    {
        Etetbldeltime ETask = new Etetbldeltime();
        ETask.DelTimeType = Convert.ToInt32(Request.QueryString["Id"]);
        ETask.CreatedOn = DateTime.Now;
        ETask.UserId = Convert.ToInt32(Clsusr.UserId);
        ETask.DelTimeDescription = Details.Text;

        bool Result = false;
        PageType = Session["PageType"].ToString();

        if (PageType == "Add")
        {
            Result = new Products().InsertDelTimes(ETask);
        }
        else
        {
            Result = new Products().UpdateDelTimes(ETask);
        }
        if (Result == true)
        {
            if (PageType == "Add")
            {

                Common.ShowMessageHtml("Adding of Note Type Sucessful ", 1, true,
dvMsgBox);
            }
            else
            {
                Common.ShowMessageHtml("Note Type Edited Sucessfully ", 1, true,
dvMsgBox);
            }
            Response.Redirect("~/NoteTypesList.aspx?pageid=149&msg=Note Type
Saved Sucessfully");
        }
    }
}

```

```

    }
}

catch (Exception ex)
{
    Common.ShowMessageHtml(ex.Message, 1, true, dvMsgBox);
    ExceptionLogging.SendExcepToDB(ex);

    objlog.Trace_Error(5, "DateOrWed", "OrderSearchList", "Bindetails_Filter",
ex.Message);
}
//objlog.Trace_Log(0, "DateOrWed", "MangeDealer", "SaveClick", "Start Here..");
}

```

```

}
***** MapsAdd.aspx *****
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="MapsAdd.aspx.cs"
Inherits="MapsAdd" %>

<%@ Register Assembly="System.Web.DataVisualization, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35"
    Namespace="System.Web.UI.DataVisualization.Charting" TagPrefix="asp" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html>
<head>
<meta name="viewport" content="initial-scale=1.0, user-scalable=no" />
<title>DateOrWed</title>
<script type="text/javascript"
src="https://maps.googleapis.com/maps/api/js?v=3.exp&libraries=places"></script>
<script src="js/maps.js" type="text/javascript"></script>
<link href="css/maps.css" rel="stylesheet" type="text/css" />

</head>
<body bgcolor="Blue">
<input id="txtsearch" class="apply" type="text" />
<div id="divloadMap">

</div>

</body>
</html>
***** MapsAdd.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="ManageRoles.aspx.cs" Inherits="ManageRoles" %>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
Runat="Server">
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="CPH_MainContent" Runat="Server">

```

```

<script type="text/javascript" src="js/jquery-1.5.2.min.js"></script>
<script type="text/javascript" src="js/scriptbreaker-multiple-accordion-1.js"></script>
<script type="text/javascript">
    var jq = $.noConflict();
    jq(document).ready(function () {
        BindPersets();
        jq(".topnav").accordion({
        });
        $("#btnUpdate").click(sendRegistration);
    });

    function changePer() {
        BindPersets();
        jq(".topnav").accordion({
        });
    }

    function BindPersets() {

        jQuery.ajax({

            type: "POST",
            // data: "{}",
            data: "{GroupId:" + $("#CPH_MainContent_ddlper :selected").val() + ""}",
            contentType: "application/json; charset=utf-8",
            dataType: "json",
            async: false,

            url: "ManageRoles.aspx/GetPerSetsForGroup",

            success: function (msg) {
                $("#ultopnav").html(msg.d);
            },
            error: function (msg) {
                $(document.body).html(msg.responseText);
            }
        });
        $("#div_msg").html("");
    }

    function GetPerSets() {

        jQuery.ajax({
            type: "POST",
            data: "{}",

            contentType: "application/json; charset=utf-8",
            dataType: "json",
            async: false,
            url: "ManageRoles.aspx/ChkPermissions",

            success: function (msg) {
                $('#DivPer').html(msg.d);
            }
        });
    }

```

```

    },
    error: function (msg) {
        $(document.body).html(msg.responseText);
    }
});
}

function chkPriv(txt) {

    if (txt == "AccessTosite") {

        if ($("#hdenSiteacc").val() == "True") {

            return true;
        }
        else {
            return false;
        }
    }

    if (txt == "View") {

        if ($("#hdenPerSetView").val() == txt) {

            return true;
        }
        else {
            return false;
        }
    }
    if (txt == "Edit") {
        if ($("#hdenPerSetEdit").val() == txt) {
            return true;
        }
        else {
            return false;
        }
    }
    if (txt == "Add") {

        if ($("#hdenPerSetADD").val() == txt) {
            return true;
        }
        else {
            return false;
        }
    }
}

```

```

if (txt == "Delete") {
    if ($("#hdenPerSetDelete").val() == txt) {
        return true;
    }
    else {
        return false;
    }
}
}

```

```

function NavtoNoAccessPage() {
    GetPerSets();
    if (chkPriv('AccessTosite')) {
        if (!chkPriv('View')) {
            window.location.href = "NoPermission.aspx?PageName=" + "Edit
Roles&RedirectFlag=2 ";
            return false;
        }
        else {
            return true;
        }
    }
    else {
        window.location.href = "NoPermission.aspx?PageName=" + "Access Denied for the
Site. Plaese Contact Admin&RedirectFlag=1 ";
        return false;
    }
}

```

```

function sendRegistration() {
    if (NavtoNoAccessPage()) {
        if (!chkPriv('Edit')) {
            alert('You donot have permissions to Edit Roles. Please contact Admin');
            return false;
        }
    }
    else {
        return false;
    }
    var rows = [];
    var ulld;
    var _gid, _add, _edit, _del, _view;
    var linenumber;
    $('[id^="ul_"]').each(function (i, n) {
        ulld = $(this).attr("id");
    });
}

```

```

        ulld = ulld.substr(ulld.indexOf('_') + 1);

        _gid = ulld

        _view = $('#ChkView_' + _gid).attr("checked");
        _add = $('#ChkAdd_' + _gid).attr("checked");
        _edit = $('#ChkEdit_' + _gid).attr("checked");
        _del = $('#ChkDelete_' + _gid).attr("checked");
        rows.push({
            GroupID: _gid,
            Add: _add,
            View: _view,
            Edit: _edit,
            Delete: _del
        });
    });
    //alert(JSON.stringify(rows));
    $.ajax({
        type: "POST",
        url: "ManageRoles.aspx/UpdateXml",
        data: '{"name': "" + JSON.stringify(rows) + "",grpld:"" + $("#CPH_MainContent_ddlper
:selected").val() + ""}',
        contentType: "application/json; charset=utf-8",
        dataType: "json",
        success: function (html, status) {

            if (status) {
                //window.location.href = 'assignRoles.aspx';
                //$("#div_msg").html("Successfully Updated");
                alert("Successfully Updated");
            }
        }
    });

}
</script>
<div class="mid_block">
    <h1><span>Assign ROLES</span></h1>
    <div id="mid_panel">

        <div class="mid_main_container">

            <h2>Roles / permission groups</h2>
            <hr style="background-color:#999">
            <div class="lft_pad_block">
                <table width="100%" border="0" cellspacing="0" cellpadding="0"
class="form_tab">
                    <tr>
                        <td width="20%" align="right" >Permission Group</td>
                        <td width="2%" align="center" >:</td>
                        <td width="78%"> <asp:DropDownList CssClass="drop_small width1"
ID="ddlper" runat="server" onchange="changePer();">

```

```

        </asp:DropDownList></td>
    </tr>
</table>
<ul id="ultopnav" class="topnav">

    </ul>
    <div id="div_msg"> </div>
    <div class="actions mar_top">
    <ul>
        <li class="flt"><a href="ViewUsers.aspx"></a></li>
        <li class="flt"><a href="#"></a></li>

    </ul>
        <input id="hdnpriv" runat="server" type="hidden" value="1" />
    </div>
    </div>
    </div>
    </div>
    </div>
</div>

    <div id="DivPer"></div>
</asp:Content>

***** MapsAdd.aspx.cs *****
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

public partial class MapsAdd : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {

    }
}
***** ManageRoles.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="ManageRoles.aspx.cs" Inherits="ManageRoles" %>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
Runat="Server">
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="CPH_MainContent" Runat="Server">
    <script type="text/javascript" src="js/jquery-1.5.2.min.js"></script>
<script type="text/javascript" src="js/scriptbreaker-multiple-accordion-1.js"></script>

```



```

<script type="text/javascript">
  var jq = $.noConflict();
  jq(document).ready(function () {
    BindPersets();
    jq(".topnav").accordion({
    });
    $("#btnUpdate").click(sendRegistration);
  });

function changePer() {
  BindPersets();
  jq(".topnav").accordion({
  });
}

function BindPersets() {

  jQuery.ajax({

    type: "POST",
    // data: "{}",
    data: "{GroupId:" + $("#CPH_MainContent_ddlper :selected").val() + "}",
    contentType: "application/json; charset=utf-8",
    dataType: "json",
    async: false,

    url: "ManageRoles.aspx/GetPerSetsForGroup",

    success: function (msg) {
      $("#ultopnav").html(msg.d);
    },
    error: function (msg) {
      $(document.body).html(msg.responseText);
    }
  });
  $("#div_msg").html("");
}

function GetPerSets() {

  jQuery.ajax({
    type: "POST",
    data: "{}",

    contentType: "application/json; charset=utf-8",
    dataType: "json",
    async: false,
    url: "ManageRoles.aspx/ChkPermissions",

    success: function (msg) {
      $('#DivPer').html(msg.d);
    }
  });
}

```

```

    },
    error: function (msg) {
        $(document.body).html(msg.responseText);
    }
});
}

function chkPriv(txt) {

    if (txt == "AccessTosite") {

        if ($("#hdenSiteacc").val() == "True") {

            return true;
        }
        else {
            return false;
        }
    }

    if (txt == "View") {

        if ($("#hdenPerSetView").val() == txt) {

            return true;
        }
        else {
            return false;
        }
    }
    if (txt == "Edit") {
        if ($("#hdenPerSetEdit").val() == txt) {
            return true;
        }
        else {
            return false;
        }
    }
    if (txt == "Add") {

        if ($("#hdenPerSetADD").val() == txt) {
            return true;
        }
        else {
            return false;
        }
    }
    if (txt == "Delete") {

```

```

    if ($("#hdenPerSetDelete").val() == txt) {
        return true;
    }
    else {
        return false;
    }
}
}

```

```

function NavtoNoAccessPage() {

```

```

    GetPerSets();
    if (chkPriv('AccessTosite')) {

        if (!chkPriv('View')) {
            window.location.href = "NoPermission.aspx?PageName=" + "Edit
Roles&RedirectFlag=2 ";
            return false;
        }
        else {
            return true;
        }
    }
    else {
        window.location.href = "NoPermission.aspx?PageName=" + "Access Denied for the
Site. Plaese Contact Admin&RedirectFlag=1 ";
        return false;
    }
}

```

```

function sendRegistration() {
    if (NavtoNoAccessPage()) {

```

```

        if (!chkPriv('Edit')) {
            alert('You donot have permissions to Edit Roles. Please contact Admin');
            return false;
        }
    }
    else {

        return false;
    }
    var rows = [];
    var ulld;
    var _gid, _add, _edit, _del, _view;
    var linenumber;
    $('[id^="ul_"]').each(function (i, n) {
        ulld = $(this).attr("id");
        ulld = ulld.substr(ulld.indexOf('_') + 1);

```



```

        </tr>
    </table>
    <ul id="ultopnav" class="topnav">

        </ul>
        <div id="div_msg"> </div>
        <div class="actions mar_top">
        <ul>
            <li class="flt"><a href="ViewUsers.aspx"></a></li>
            <li class="flt"><a href="#"></a></li>

        </ul>
        <input id="hdnpriv" runat="server" type="hidden" value="1" />
    </div>
    </div>
    </div>
    </div>
    </div>
</div>

```

```

    <div id ="DivPer"></div>
</asp:Content>

```

```

***** ManageRoles.aspx.cs *****

```

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.IO;
using System.Data;
using System.Text;
using DowBAL;
using System.Xml;
using DowBEntity;
public partial class ManageRoles : System.Web.UI.Page
{
    public static string Ids = ",";
    public static string ParentIds = ",";
    public static bool HasEditPriv = true;
    public static string FileName = string.Empty;
    protected void Page_Load(object sender, EventArgs e)
    {
        if (!IsPostBack)
        {
            DataSet dsPer = null;
            DowBAL.BALCreateUser Buser = new DowBAL.BALCreateUser();
            dsPer = Buser.GetAllPersets();
            ddlper.DataSource = dsPer;

```

```

ddlper.DataTextField = "GroupName";
ddlper.DataValueField = "GroupId";
ddlper.DataBind();

GetPerSetsForGroup(ddlper.SelectedValue.ToString());

string AbsPath = Request.Url.AbsolutePath;
FileName = Path.GetFileName(AbsPath);
}
if (!User.IsInRole("Edit"))
{
    HasEditPriv = false;
    hdnpriv.Value = "0";

}
else
{
    HasEditPriv = true;
    hdnpriv.Value = "1";

}

}

[System.Web.Services.WebMethod]
public static string ChkPermissions()
{
    return
Common.GetPermissionSets(FileName.TrimEnd(".").ToCharArray()).TrimEnd(".").ToCharArray());
}

[System.Web.Services.WebMethod]
public static string GetPerSetsForGroup(string GroupId)
{
    if (GroupId != null && GroupId == "" && GroupId=="undefined")
    {
        ClsPermissioGroup objclsper = new ClsPermissioGroup();
        DataTable dtGetPerSetsForGroup =
objclsper.PopulateDynamicTree(Convert.ToInt32(GroupId));

        StringBuilder sbuilder = new StringBuilder();
        DataRow[] parentMenus = dtGetPerSetsForGroup.Select("ParentId = 0");

        var sb = new StringBuilder();

        return GenerateUL(parentMenus, dtGetPerSetsForGroup, sb, 0);
    }
    else { return null; }
}

private static string GenerateUL(DataRow[] menu, DataTable table, StringBuilder sb, int
flag)
{

```

```

string isdisabled = string.Empty;
if (!HasEditPriv)
{
    isdisabled = "disabled ='disabled'";
}

if (menu.Length > 0)
{
    foreach (DataRow dr in menu)
    {
        string handler = dr["PageUrl"].ToString();
        string menuText = dr["PageName"].ToString();
        string level = dr["level"].ToString();
        string clschkbox = "form_acording";
        string pid = dr["PagelId"].ToString();
        string line = string.Empty;
        if (menuText == "User Manager")
        {
            handler = handler.ToString();
        }

        if (dr["level"].ToString() == "0")
        {
            line = String.Format(@" <li class='bg1'><a href='#' class='bg'>{1}</a>", handler,
menuText);
        }
        if (dr["level"].ToString() == "1")
        {
            line = String.Format(@" <li class='bg2'><a href='#' class='bg2'>{1}</a>",
handler, menuText);
            clschkbox = "form_acording sub";
        }

        if (Convert.ToInt32(dr["level"].ToString()) >= 2)
        {
            line = String.Format(@" <li class='bg3'><a href='#' class='bg3'>{1}</a>",
handler, menuText);
            clschkbox = "form_acording sub_2";
        }
        sb.Append(line);
        DataRow[] subMenu = table.Select(String.Format("ParentId = {0}", pid));
        if (subMenu.Length > 0)
        {
            if (handler.ToLower() != "no-url")
            {
                string clsvchked = string.Empty;
                if (dr["view"].ToString().ToLower() == "true")
                {
                    clsvchked = "checked='checked'";
                }
                else
                {
                    clsvchked = "";
                }
            }
        }
    }
}

```

```

string clsechked = string.Empty;
if (dr["edit"].ToString().ToLower() == "true")
{
    clsechked = "checked='checked'";
}
else
{
    clsechked = "";
}

string clsachked = string.Empty;
if (dr["add"].ToString().ToLower() == "true")
{
    clsachked = "checked='checked'";
}
else
{
    clsachked = "";
}

string clsdchked = string.Empty;
if (dr["delete"].ToString().ToLower() == "true")
{
    clsdchked = "checked='checked'";
}
else
{
    clsdchked = "";
}

sb.Append("<div id='ul_" + pid + "'>");
sb.Append("<div class='form_acording position'>");
sb.Append("<label><input type='checkbox' " + isdisabled + " " + clsvchked + "
name='ChkView_" + pid + "' runat='server' id='ChkView_" + pid + "' />View</label>");
sb.Append("<label><input type='checkbox' " + isdisabled + " " + clsachked + "
name='ChkAdd_" + pid + "' runat='server' id='ChkAdd_" + pid + "' />Add</label>");
sb.Append("<label><input type='checkbox' " + isdisabled + " " + clsechked + "
name='ChkEdit_" + pid + "' runat='server' id='ChkEdit_" + pid + "' />Edit</label>");
sb.Append("<label><input type='checkbox' " + isdisabled + " " + clsdchked + "
name='ChkDelete_" + pid + "' runat='server' id='ChkDelete_" + pid + "' />Delete</label>");
sb.Append("</div>");
sb.Append("</div>");

}

var subMenuBuilder = new StringBuilder();

if (level != "0")
{
    string concat = pid + " " + "-" + " " + level;
    DataRow[] drn = table.Select("concatenador LIKE '%" + concat + "%'");
    Ids = Ids + drn[drn.Length - 1]["Pageld"].ToString() + ",";
    sb.Append("<ul class='no_mar'>");
}
else

```



```

{
    string concat = pid + " " + "-" + " " + level;
    DataRow[] drn = table.Select("concatenador LIKE '%" + concat + "%'");
    ParentIds = ParentIds + drn[drn.Length - 1]["PagId"].ToString() + ",";
    sb.Append(" <ul class='no_mar'>");
}

if (ParentIds.Contains(", " + pid.ToString() + ","))
{
    sb.Append(GenerateUL(subMenu, table, subMenuBuilder, 1));
}
else
{
    sb.Append(GenerateUL(subMenu, table, subMenuBuilder, 0));
}
}
else
{
    if (handler.ToLower() != "no-url")
    {
        string clsvchked = string.Empty;
        if (dr["view"].ToString().ToLower() == "true")
        {
            clsvchked = "checked='checked'";
        }
        else
        {
            clsvchked = "";
        }

        string clsechked = string.Empty;
        if (dr["edit"].ToString().ToLower() == "true")
        {
            clsechked = "checked='checked'";
        }
        else
        {
            clsechked = "";
        }

        string clsachked = string.Empty;
        if (dr["add"].ToString().ToLower() == "true")
        {
            clsachked = "checked='checked'";
        }
        else
        {
            clsachked = "";
        }

        string clsdchked = string.Empty;
        if (dr["delete"].ToString().ToLower() == "true")
        {
            clsdchked = "checked='checked'";
        }
    }
}

```

```

    }
    else
    {
        clsdchked = "";
    }

    sb.Append("<ul id='ul_" + pid + ">");
    sb.Append("<li>");
    sb.Append("<div class='" + clscheckbox + ">");
    sb.Append("<label><input type='checkbox' " + isdisabled + " " + clsvchked + "
name='ChkView_" + pid + " runat='server' id='ChkView_" + pid + "' />View</label>");
    sb.Append("<label><input type='checkbox' " + isdisabled + " " + clsachked + "
name='ChkAdd_" + pid + " runat='server' id='ChkAdd_" + pid + "' />Add</label>");
    sb.Append("<label><input type='checkbox' " + isdisabled + " " + clsechked + "
" name='ChkEdit_" + pid + " runat='server' id='ChkEdit_" + pid + "' />Edit</label>");
    sb.Append("<label><input type='checkbox' " + isdisabled + " " + clsdchked + "
name='ChkDelete_" + pid + " runat='server' id='ChkDelete_" + pid + "' />Delete</label>");
    sb.Append(" </div>");
    sb.Append("</li>");
    sb.Append("</ul>");
}
}
sb.Append("</li>");

if (Ids.Contains(", " + pid.ToString() + ",") || flag == 1)
{
    sb.Append("</ul>");
}
}
}

return sb.ToString();
}

```

```

[System.Web.Services.WebMethod()]
public static bool UpdateXml(string name, string grpId)
{
    try
    {
        string Data = "{/";
        Data += "root/";
        Data += ":{/";
        Data += "/Data/";
        Data += ":";

        Data += name + "}}";

        Data = Data.Replace('/', "");

        XmlDocument xml = Newtonsoft.Json.JsonConvert.DeserializeXmlNode(Data);

        ClsPermissioGroup clsper = new ClsPermissioGroup();
    }
}

```

```

        ClsLoggedInUser Clsusr =
(ClsLoggedInUser)HttpContext.Current.Session["LoggedInUser_Specs"];

        bool status = clsper.UpdatePermissionGrp(xml, Convert.ToInt32(grpId),
Convert.ToInt32(Clsusr.UserId), Clsusr.LoginID);
        if (status)
        {
            return true;
        }
        else
        {
            return false;
        }
    }
    catch (Exception ex)
    {
        return false;
    }
}

}
}
***** ManageGrpMap.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="ManageGrpMap.aspx.cs" Inherits="ManageGrpMap"
%>
<%@ Register Assembly="Infragistics4.Web.v14.2, Version=14.2.20142.1028,
Culture=neutral, PublicKeyToken=7dd5c3163f2cd0cb"
namespace="Infragistics.Web.UI.GridControls" tagprefix="ig" %>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
Runat="Server">
    <style type="text/css">
        .auto-style2 {
            height: 16px;
        }
    </style>
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="CPH_MainContent" Runat="Server">
    <asp:ScriptManager ID="ScriptManager1" runat="server"></asp:ScriptManager>
    <p>Group Mapping<> Manage Group Mapping</p>

    <br />
    <table width="28%" border="0" cellspacing="0" cellpadding="0" class="f_lft
compact_mar">
        <tr>
            <td height="5">
                <asp:Button ID="Button1" runat="server" OnClick="Button1_Click"
Text="Save" />
            </td>
        </tr>
    </tbody >
    <tr>
        <td align="right" class="label_txt" >Group : </td>
    </tr>

```

```

        <asp:DropDownList ID="ddl_GroupId" runat="server" ></asp:DropDownList>
        </td>
        <td height="5">
        <asp:Button ID="Button2" runat="server" OnClick="Button2_Click" Text="Go"
/>
        </td>
</tr>
<tr>
<td height="5"></td>
</tr>
<tr>
<td width="10" align="centre" class="auto-style2"> Page Id </td>
<td width="34" align="centre" class="auto-style2"> Page Name</td>
<td width="14" align="centre" class="auto-style2"> View </td>
<td width="14" align="centre" class="auto-style2"> Add </td>
<td width="14" align="centre" class="auto-style2"> Edit </td>
<td width="14" align="centre" class="auto-style2"> Delete </td>
<td width="4" align="centre" class="auto-style2"> Click</td>
</tr>
<tr>
<td align="left">
<asp:TextBox ID="txt_PageID1" runat="server" ReadOnly="true"
></asp:TextBox>
</td>
<td align="left">
<asp:TextBox ID="txt_PageName1" runat="server" ReadOnly="true"
></asp:TextBox>
</td>
<td width="14%" align="left">
<asp:CheckBox ID="chk_View1" runat="server" Text="" />
</td>
<td width="14%" align="left">
<asp:CheckBox ID="chk_add1" runat="server" Text="" />
</td>
<td width="14%" align="left">
<asp:CheckBox ID="chk_edit1" runat="server" Text="" />
</td>
<td width="14%" align="left">
<asp:CheckBox ID="chk_delete1" runat="server" Text="" />
</td>
<td align="left">
<asp:TextBox ID="MapID1" runat="server" Visible="False" ReadOnly="True"
></asp:TextBox>
</td>
<td height="5">
<asp:Button ID="Button3" runat="server" OnClick="Button3_Click"
Text="Button Mapping" />
</td>
</tr>
<tr>
<td width="24%" align="left">
<asp:TextBox ID="txt_PageID2" runat="server" ReadOnly="true"
></asp:TextBox>
</td>

```

```

        <td width="34%" align="left">
        <asp:TextBox ID="txt_PageName2" runat="server" ReadOnly="true"
></asp:TextBox>
        </td>
        <td width="24%" align="left">
        <asp:CheckBox ID="chk_View2" runat="server" Text="" />
        </td>
        <td width="24%" align="left">
        <asp:CheckBox ID="chk_add2" runat="server" Text="" />
        </td>
        <td width="24%" align="left">
        <asp:CheckBox ID="chk_edit2" runat="server" Text="" />
        </td>
        <td width="24%" align="left">
        <asp:CheckBox ID="chk_delete2" runat="server" Text="" />
        </td>
        <td align="left">
        <asp:TextBox ID="MapId2" runat="server" Visible="False" ReadOnly="True"
></asp:TextBox>
        </td>
        <td height="5">
        <asp:Button ID="Button4" runat="server" OnClick="Button4_Click"
Text="Button Mapping" />
        </td>
        </tr>
        <tr>
        <td width="34%" align="left">
        <asp:TextBox ID="txt_PageID3" runat="server"
ReadOnly="true"></asp:TextBox>
        </td>
        <td width="34%" align="left">
        <asp:TextBox ID="txt_PageName3" runat="server" ReadOnly="true"
></asp:TextBox>
        </td>
        <td width="34%" align="left">
        <asp:CheckBox ID="chk_View3" runat="server" Text="" />
        </td>
        <td width="34%" align="left">
        <asp:CheckBox ID="chk_add3" runat="server" Text="" />
        </td>
        <td width="34%" align="left">
        <asp:CheckBox ID="chk_edit3" runat="server" Text="" />
        </td>
        <td width="34%" align="left">
        <asp:CheckBox ID="chk_delete3" runat="server" Text="" />
        </td>
        <td align="left">
        <asp:TextBox ID="MapId3" runat="server" Visible="False" ReadOnly="True"
></asp:TextBox>
        </td>
        <td height="5">
        <asp:Button ID="Button5" runat="server" OnClick="Button5_Click"
Text="Button Mapping" />
        </td>
        </tr>

```

```

        <tr>
        <td width="44%" align="left">
            <asp:TextBox ID="txt_PageID4" runat="server" ReadOnly="true"
></asp:TextBox>
        </td>
        <td width="44%" align="left">
            <asp:TextBox ID="txt_PageName4" runat="server" ReadOnly="true"
></asp:TextBox>
        </td>
        <td width="44%" align="left">
            <asp:CheckBox ID="chk_View4" runat="server" Text="" />
        </td>
        <td width="44%" align="left">
            <asp:CheckBox ID="chk_add4" runat="server" Text="" />
        </td>
        <td width="44%" align="left">
            <asp:CheckBox ID="chk_edit4" runat="server" Text="" />
        </td>
        <td width="44%" align="left">
            <asp:CheckBox ID="chk_delete4" runat="server" Text="" />
        </td>
        <td align="left">
            <asp:TextBox ID="MapId4" runat="server" Visible="False" ReadOnly="True"
></asp:TextBox>
        </td>
        <td height="5">
            <asp:Button ID="Button6" runat="server" OnClick="Button6_Click"
Text="Button Mapping" />
        </td>
        </tr>
        <tr>
        <td height="5"></td>
        </tr>
    </tbody>
</table>
<table width="38%" border="0" cellspacing="0" cellpadding="0" class="f_rt
compact_mar">
<tbody>
<tr>
    <td width="14%" align="centre" class="auto-style2"> Page Id </td>
    <td width="34%" align="centre" class="auto-style2"> Page Name :</td>
    <td width="14%" align="centre" class="auto-style2"> Button Id </td>
    <td width="14%" align="centre" class="auto-style2"> Button Name </td>
    <td width="14%" align="centre" class="auto-style2"> Enabled </td>
    <td width="14" align="centre" class="auto-style2"> </td>
    <td height="5">
        <asp:Button ID="Button7" runat="server" OnClick="Button7_Click"
Text="Save Button Mapping" />
    </td>
</tr>
<tr>
    <td align="left">
        <asp:TextBox ID="txt_PbID1" runat="server" ></asp:TextBox>
    </td>

```

```

        <td align="left">
        <asp:TextBox ID="txt_PbName1" runat="server" ></asp:TextBox>
        </td>
            <td align="left">
            <asp:TextBox ID="txt_btID1" runat="server" ></asp:TextBox>
            </td>
            <td align="left">
            <asp:TextBox ID="txt_btName1" runat="server" ></asp:TextBox>
            </td>

        <td width="14%" align="left">
        <asp:CheckBox ID="CheckBox1" runat="server" Text="" />
        </td>

        <td align="left">
        <asp:TextBox ID="Id1" runat="server" Enabled="False" Visible="False"
></asp:TextBox>
        </td>
        </tr>
            <tr>
                <td align="left">
                <asp:TextBox ID="txt_PbID2" runat="server" ></asp:TextBox>
                </td>
                <td align="left">
                <asp:TextBox ID="txt_PbName2" runat="server" ></asp:TextBox>
                </td>
                <td align="left">
                <asp:TextBox ID="txt_btID2" runat="server" ></asp:TextBox>
                </td>
                <td align="left">
                <asp:TextBox ID="txt_btName2" runat="server" ></asp:TextBox>
                </td>

                <td width="24%" align="left">
                <asp:CheckBox ID="CheckBox2" runat="server" Text="" />
                </td>

                <td align="left">
                <asp:TextBox ID="Id2" runat="server" Enabled="False" Visible="False"
></asp:TextBox>
                </td>
            </tr>
                <tr>
                    <td align="left">
                    <asp:TextBox ID="txt_PbID3" runat="server" ></asp:TextBox>
                    </td>
                    <td align="left">
                    <asp:TextBox ID="txt_PbName3" runat="server" ></asp:TextBox>
                    </td>
                    <td align="left">
                    <asp:TextBox ID="txt_btID3" runat="server" ></asp:TextBox>
                    </td>
                    <td align="left">

```

```

        <asp:TextBox ID="txt_btName3" runat="server" ></asp:TextBox>
    </td>

    <td width="34%" align="left">
        <asp:CheckBox ID="CheckBox3" runat="server" Text="" />
    </td>

    <td align="left">
        <asp:TextBox ID="Id3" runat="server" Enabled="False" Visible="False"
></asp:TextBox>
    </td>
</tr>

    <tr>
    <td align="left">
        <asp:TextBox ID="txt_PbID4" runat="server" ></asp:TextBox>
    </td>
    <td align="left">
        <asp:TextBox ID="txt_PbName4" runat="server" ></asp:TextBox>
    </td>
    <td align="left">
        <asp:TextBox ID="txt_btID4" runat="server" ></asp:TextBox>
    </td>
    <td align="left">
        <asp:TextBox ID="txt_btName4" runat="server" ></asp:TextBox>
    </td>

    <td width="44%" align="left">
        <asp:CheckBox ID="CheckBox4" runat="server" Text="" />
    </td>

    <td align="left">
        <asp:TextBox ID="Id4" runat="server" Enabled="False" Visible="False"
></asp:TextBox>
    </td>
</tr>
</tbody>
</table>
</asp:Content>

```

```

***** ManageGrpMap.aspx.cs *****

```

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBEntity;
using DowBAL;
using Infragistics;
public partial class ManageGrpMap : System.Web.UI.Page
{
    int ctr = 0;
    int GroupId = 0;

```



```

string PageType;
int Pageid;
protected void Page_Load(object sender, EventArgs e)
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    if (Clsusr == null)
    {
        Response.Redirect("MLB_Login.aspx");
    }
    if (!IsPostBack)
    {
        PageType = Page.Request.QueryString["PageType"];
        GroupId = Convert.ToInt32(Page.Request.QueryString["Id"]);
        Session["PageType"] = PageType;

        PopDlls ObjPD = new PopDlls();
        ObjPD.BindDropDowns(ddl_GroupId, (new PopDlls()).GetRoles(Clsusr.RoleId));
        if (PageType == "Edit")
        {
            SetData(GroupId);
            ddl_GroupId.Enabled = false;
        }
        else
        {
            ddl_GroupId.Enabled = true;
        }
    }
}

private void SetData(int GroupId)
{
    List<Etev_tblGrpMapping> Lst = (new
tblGrpMappings()).GettblGrpMappingDescByGroupId(GroupId);
    if (Lst != null && Lst.Count > 0)
    {
        foreach (var row in Lst)
        {
            //if (row.Pageld > 0)
            {
                ctr++;
                if (ctr == 1)
                {
                    txt_PageID1.Text = row.Pageld.ToString();
                    txt_PageName1.Text = row.PageName;
                    chk_View1.Checked = Convert.ToBoolean(row.view);
                    chk_add1.Checked = Convert.ToBoolean(row.add);
                    chk_edit1.Checked = Convert.ToBoolean(row.Edit);
                    chk_delete1.Checked = Convert.ToBoolean(row.Delete);
                    MapId1.Text = Convert.ToString(row.MapId);
                }
                if (ctr == 2)
                {
                    txt_PageID2.Text = row.Pageld.ToString();
                    txt_PageName2.Text = row.PageName;
                }
            }
        }
    }
}

```

```

        chk_View2.Checked = Convert.ToBoolean(row.view);
        chk_add2.Checked = Convert.ToBoolean(row.add);
        chk_edit2.Checked = Convert.ToBoolean(row.Edit);
        chk_delete2.Checked = Convert.ToBoolean(row.Delete);
        MapId2.Text = Convert.ToString(row.MapId);
    }
    if (ctr == 3)
    {
        txt_PageId3.Text = row.PagId.ToString();
        txt_PageName3.Text = row.PageName;
        chk_View3.Checked = Convert.ToBoolean(row.view);
        chk_add3.Checked = Convert.ToBoolean(row.add);
        chk_edit3.Checked = Convert.ToBoolean(row.Edit);
        chk_delete3.Checked = Convert.ToBoolean(row.Delete);
        MapId3.Text = Convert.ToString(row.MapId);
    }
    if (ctr == 4)
    {
        txt_PageId4.Text = row.PagId.ToString();
        txt_PageName4.Text = row.PageName;
        chk_View4.Checked = Convert.ToBoolean(row.view);
        chk_add4.Checked = Convert.ToBoolean(row.add);
        chk_edit4.Checked = Convert.ToBoolean(row.Edit);
        chk_delete4.Checked = Convert.ToBoolean(row.Delete);
        MapId4.Text = Convert.ToString(row.MapId);
    }
    }
    }
    //return Lst.Count;
}
//else
//{
//    return 0;
//}
}
protected void Button1_Click(object sender, EventArgs e)
{
    PageType = Session["PageType"].ToString();
    GroupId = Convert.ToInt32(ddl_GroupId.SelectedValue);
    ctr = 1;
    bool rtn=false;
    EtetblGrpMapping grpmap=new EtetblGrpMapping();
    do
    {
        grpmap.GroupId=GroupId;
        grpmap.CreatedOn = DateTime.Now;
        if (ctr==1)
        {
            grpmap.PagId=Convert.ToInt32( txt_PageId1.Text);
            grpmap.view=Convert.ToBoolean(chk_View1.Checked);
            grpmap.add=Convert.ToBoolean(chk_add1.Checked);
            grpmap.Edit=Convert.ToBoolean(chk_edit1.Checked);
            grpmap.Delete=Convert.ToBoolean(chk_delete1.Checked);
            if(PageType=="Edit")

```

```

    {
        grpmap.MapId = Convert.ToInt32(MapId1.Text);
        rtn = (new tblGrpMappings()).Update(grpmap);
    }
    else{
        rtn = (new tblGrpMappings()).Insert(grpmap);
    }
}
if (ctr==2)
{
    grpmap.PageId=Convert.ToInt32( txt_PageID2.Text);
    grpmap.view=Convert.ToBoolean(chk_View2.Checked);
    grpmap.add=Convert.ToBoolean(chk_add2.Checked);
    grpmap.Edit=Convert.ToBoolean(chk_edit2.Checked);
    grpmap.Delete=Convert.ToBoolean(chk_delete2.Checked);
    if (PageType == "Edit")
    {
        grpmap.MapId = Convert.ToInt32(MapId2.Text);
        rtn = (new tblGrpMappings()).Update(grpmap);
    }
    else
    {
        rtn = (new tblGrpMappings()).Insert(grpmap);
    }
}
if (ctr==3)
{
    grpmap.PageId=Convert.ToInt32( txt_PageID3.Text);
    grpmap.view=Convert.ToBoolean(chk_View3.Checked);
    grpmap.add=Convert.ToBoolean(chk_add3.Checked);
    grpmap.Edit=Convert.ToBoolean(chk_edit3.Checked);
    grpmap.Delete=Convert.ToBoolean(chk_delete3.Checked);
    if (PageType == "Edit")
    {
        grpmap.MapId = Convert.ToInt32(MapId3.Text);
        rtn = (new tblGrpMappings()).Update(grpmap);
    }
    else
    {
        rtn = (new tblGrpMappings()).Insert(grpmap);
    }
}
if (ctr==4)
{
    grpmap.PageId=Convert.ToInt32( txt_PageID4.Text);
    grpmap.view=Convert.ToBoolean(chk_View4.Checked);
    grpmap.add=Convert.ToBoolean(chk_add4.Checked);
    grpmap.Edit=Convert.ToBoolean(chk_edit4.Checked);
    grpmap.Delete=Convert.ToBoolean(chk_delete4.Checked);

```

```

        if (PageType == "Edit")
        {
            grpmap.MapId = Convert.ToInt32(MapId4.Text);
            rtn = (new tblGrpMappings()).Update(grpmap);
        }
        else
        {
            rtn = (new tblGrpMappings()).Insert(grpmap);
        }
    }
    ctr++;
} while (ctr < 5);
}
protected void Button2_Click(object sender, EventArgs e)
{
    ctr = 0;
    GroupId = Convert.ToInt32(ddl_GroupId.SelectedValue);
    List<Etev_tblGrpMapping> List = (new
tblGrpMappings()).GettblGrpMappingDescByGroupId(GroupId);
    if (List != null && List.Count > 0)
    {
        SetData(GroupId);
        Session["PageType"] = "Edit";
    }
    else
    {
        List<EteTblPages> Lst = (new tblPages()).GettblPages();
        foreach (var row in Lst)
        {
            ctr++;
            if (ctr == 1)
            {
                txt_PageID1.Text = row.PagId.ToString();
                txt_PageName1.Text = row.PageName;
            }
            if (ctr == 2)
            {
                txt_PageID2.Text = row.PagId.ToString();
                txt_PageName2.Text = row.PageName;
            }
            if (ctr == 3)
            {
                txt_PageID3.Text = row.PagId.ToString();
                txt_PageName3.Text = row.PageName;
            }
            if (ctr == 4)
            {
                txt_PageID4.Text = row.PagId.ToString();
                txt_PageName4.Text = row.PageName;
            }
        }
    }
}
}
}

```

```

protected void Button3_Click(object sender, EventArgs e)
{
    Pageid = Convert.ToInt32(txt_PageID1.Text);
    ctr = 0;
    string PageName = txt_PageName1.Text;
    List<EtetblPageButtonMapping> Lst = (new
tblPageButtonMappings()).GettblPageButtonMappingbyPageId(Pageid);
    foreach (var row in Lst)
    {
        ctr++;
        if (ctr == 1)
        {
            txt_PbID1.Text = row.pageid.ToString();
            txt_PbName1.Text = PageName;
            txt_btID1.Text = row.lbtnid.ToString();
            txt_btName1.Text = row.lbtnName;
            CheckBox1.Checked = Convert.ToBoolean(row.show);
            Id1.Text = row.lbtnid.ToString();
        }
        if (ctr == 2)
        {
            txt_PbID2.Text = row.pageid.ToString();
            txt_PbName2.Text = PageName;
            txt_btID2.Text = row.lbtnid.ToString();
            txt_btName2.Text = row.lbtnName;
            CheckBox2.Checked = Convert.ToBoolean(row.show);
            Id2.Text = row.lbtnid.ToString();
        }
        if (ctr == 3)
        {
            txt_PbID3.Text = row.pageid.ToString();
            txt_PbName3.Text = PageName;
            txt_btID3.Text = row.lbtnid.ToString();
            txt_btName3.Text = row.lbtnName;
            CheckBox3.Checked = Convert.ToBoolean(row.show);
            Id3.Text = row.lbtnid.ToString();
        }
        if (ctr == 4)
        {
            txt_PbID4.Text = row.pageid.ToString();
            txt_PbName4.Text = PageName;
            txt_btID4.Text = row.lbtnid.ToString();
            txt_btName4.Text = row.lbtnName;
            CheckBox4.Checked = Convert.ToBoolean(row.show);
            Id4.Text = row.lbtnid.ToString();
        }
    }
}

protected void Button4_Click(object sender, EventArgs e)
{
    Pageid = Convert.ToInt32(txt_PageID2.Text);
    ctr = 0;
    string PageName = txt_PageName2.Text;

```

```

List<EtetblPageButtonMapping> Lst = (new
tblPageButtonMappings()).GettblPageButtonMappingbyPageId(Pageid);
foreach (var row in Lst)
{
    ctr++;
    if (ctr == 1)
    {
        txt_PbID1.Text = row.pageid.ToString();
        txt_PbName1.Text = PageName;
        txt_btID1.Text = row.lbtnid.ToString();
        txt_btName1.Text = row.lbtnName;
        CheckBox1.Checked = Convert.ToBoolean(row.show);
        Id1.Text = row.lbtnid.ToString();
    }
    if (ctr == 2)
    {
        txt_PbID2.Text = row.pageid.ToString();
        txt_PbName2.Text = PageName;
        txt_btID2.Text = row.lbtnid.ToString();
        txt_btName2.Text = row.lbtnName;
        CheckBox2.Checked = Convert.ToBoolean(row.show);
        Id2.Text = row.lbtnid.ToString();
    }
    if (ctr == 3)
    {
        txt_PbID3.Text = row.pageid.ToString();
        txt_PbName3.Text = PageName;
        txt_btID3.Text = row.lbtnid.ToString();
        txt_btName3.Text = row.lbtnName;
        CheckBox3.Checked = Convert.ToBoolean(row.show);
        Id3.Text = row.lbtnid.ToString();
    }
    if (ctr == 4)
    {
        txt_PbID4.Text = row.pageid.ToString();
        txt_PbName4.Text = PageName;
        txt_btID4.Text = row.lbtnid.ToString();
        txt_btName4.Text = row.lbtnName;
        CheckBox4.Checked = Convert.ToBoolean(row.show);
        Id4.Text = row.lbtnid.ToString();
    }
}
}
protected void Button5_Click(object sender, EventArgs e)
{
    Pageid = Convert.ToInt32(txt_PageID3.Text);
    ctr = 0;
    string PageName = txt_PageName3.Text;
    List<EtetblPageButtonMapping> Lst = (new
tblPageButtonMappings()).GettblPageButtonMappingbyPageId(Pageid);
    foreach (var row in Lst)
    {
        ctr++;
        if (ctr == 1)
        {

```

```

        txt_PbID1.Text = row.pageid.ToString();
        txt_PbName1.Text = PageName;
        txt_btID1.Text = row.lbtnid.ToString();
        txt_btName1.Text = row.lbtnName;
        CheckBox1.Checked = Convert.ToBoolean(row.show);
        Id1.Text = row.lbtnid.ToString();
    }
    if (ctr == 2)
    {
        txt_PbID2.Text = row.pageid.ToString();
        txt_PbName2.Text = PageName;
        txt_btID2.Text = row.lbtnid.ToString();
        txt_btName2.Text = row.lbtnName;
        CheckBox2.Checked = Convert.ToBoolean(row.show);
        Id2.Text = row.lbtnid.ToString();
    }
    if (ctr == 3)
    {
        txt_PbID3.Text = row.pageid.ToString();
        txt_PbName3.Text = PageName;
        txt_btID3.Text = row.lbtnid.ToString();
        txt_btName3.Text = row.lbtnName;
        CheckBox3.Checked = Convert.ToBoolean(row.show);
        Id3.Text = row.lbtnid.ToString();
    }
    if (ctr == 4)
    {
        txt_PbID4.Text = row.pageid.ToString();
        txt_PbName4.Text = PageName;
        txt_btID4.Text = row.lbtnid.ToString();
        txt_btName4.Text = row.lbtnName;
        CheckBox4.Checked = Convert.ToBoolean(row.show);
        Id4.Text = row.lbtnid.ToString();
    }
}
}
protected void Button6_Click(object sender, EventArgs e)
{
    Pageid = Convert.ToInt32(txt_PageID4.Text);
    ctr = 0;
    string PageName = txt_PageName4.Text;
    List<EtetblPageButtonMapping> Lst = (new
tblPageButtonMappings()).GettblPageButtonMappingbyPageId(Pageid);
    foreach (var row in Lst)
    {
        ctr++;
        if (ctr == 1)
        {
            txt_PbID1.Text = row.pageid.ToString();
            txt_PbName1.Text = PageName;
            txt_btID1.Text = row.lbtnid.ToString();
            txt_btName1.Text = row.lbtnName;
            CheckBox1.Checked = Convert.ToBoolean(row.show);
            Id1.Text = row.lbtnid.ToString();
        }
    }
}

```

```

if (ctr == 2)
{
    txt_PbID2.Text = row.pageid.ToString();
    txt_PbName2.Text = PageName;
    txt_btID2.Text = row.lbtnid.ToString();
    txt_btName2.Text = row.lbtnName;
    CheckBox2.Checked = Convert.ToBoolean(row.show);
    Id2.Text = row.lbtnid.ToString();
}
if (ctr == 3)
{
    txt_PbID3.Text = row.pageid.ToString();
    txt_PbName3.Text = PageName;
    txt_btID3.Text = row.lbtnid.ToString();
    txt_btName3.Text = row.lbtnName;
    CheckBox3.Checked = Convert.ToBoolean(row.show);
    Id3.Text = row.lbtnid.ToString();
}
if (ctr == 4)
{
    txt_PbID4.Text = row.pageid.ToString();
    txt_PbName4.Text = PageName;
    txt_btID4.Text = row.lbtnid.ToString();
    txt_btName4.Text = row.lbtnName;
    CheckBox4.Checked = Convert.ToBoolean(row.show);
    Id4.Text = row.lbtnid.ToString();
}
}
}
protected void Button7_Click(object sender, EventArgs e)
{bool rtn;
    EtetblPageButtonMapping grp = new EtetblPageButtonMapping();
    ctr=1;
    do
    {
        if (ctr==1 && Id1.Text!="")
        {
            grp.lbtnid =Convert.ToInt32( Id1.Text);
            grp.pageid = Convert.ToInt32(txt_PbID1.Text);
            grp.pagebtnid = Convert.ToInt32(txt_btID1.Text);
            grp.show = Convert.ToInt32(CheckBox1.Checked);
            rtn = (new tblPageButtonMapings()).Update(grp);
        }
        if (ctr == 2 && Id2.Text != "")
        {
            grp.lbtnid = Convert.ToInt32(Id2.Text);
            grp.pageid = Convert.ToInt32(txt_PbID2.Text);
            grp.pagebtnid = Convert.ToInt32(txt_btID2.Text);
            grp.show = Convert.ToInt32(CheckBox2.Checked);
            rtn = (new tblPageButtonMapings()).Update(grp);
        }
        if (ctr == 3 && Id3.Text != "")
        {
            grp.lbtnid = Convert.ToInt32(Id3.Text);
            grp.pageid = Convert.ToInt32(txt_PbID3.Text);

```



```

        grp.pagebtnid = Convert.ToInt32(txt_btID3.Text);
        grp.show = Convert.ToInt32(CheckBox3.Checked);
        rtn = (new tblPageButtonMapings()).Update(grp);
    }
    if (ctr == 4 && Id4.Text!="")
    {
        grp.lbtnid = Convert.ToInt32(Id4.Text);
        grp.pageid = Convert.ToInt32(txt_PbID4.Text);
        grp.pagebtnid = Convert.ToInt32(txt_btID4.Text);
        grp.show = Convert.ToInt32(CheckBox4.Checked);
        rtn = (new tblPageButtonMapings()).Update(grp);
    }
    ctr++;
} while (ctr < 5);
}
protected void txt_PageID1_TextChanged(object sender, EventArgs e)
{
}
}
}
***** ManageNoteTypes.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="ManageNoteTypes.aspx.cs"
Inherits="ManageNoteTypes" %>

<%@ Register Assembly="Infragistics4.Web.v14.2, Version=14.2.20142.1028,
Culture=neutral, PublicKeyToken=7dd5c3163f2cd0cb"
Namespace="Infragistics.Web.UI.EditorControls" TagPrefix="ig" %>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
    <script type="text/javascript">
        function checkTextAreaMaxLength(textBox, e, length) {

            var mLen = textBox["MaxLength"];
            if (null == mLen)
                mLen = length;

            var maxLength = parseInt(mLen);
            if (!checkSpecialKeys(e)) {
                if (textBox.value.length > maxLength - 1) {
                    if (window.event)//IE
                        e.returnValue = false;
                    else
                        e.preventDefault();
                }
            }
        }
    }

function isNumberKey(evt) {
    var charCode = (evt.which) ? evt.which : event.keyCode;
    if (charCode != 46 && charCode > 31 && (charCode < 48 || charCode > 57)) {
        return false;
    } else {

```



```
</div>  
</asp:Content>
```

```
***** ManageNoteTypes.aspx.cs *****
```

```
using System;  
using System.Collections.Generic;  
using System.Linq;  
using System.Web;  
using System.Web.UI;  
using System.Web.UI.WebControls;  
using System.Web.Security;  
using DowBAL;  
using System.Data;  
using DowDAL;  
using DowBEntity;  
using System.Data.SqlClient;  
using System.Configuration;
```

```
public partial class ManageNoteTypes : System.Web.UI.Page  
{  
    public string PageType;  
    string pageid;  
    int Id;  
    protected void Page_Load(object sender, EventArgs e)  
    {  
        CIsLoggedInUser CIsusr = (CIsLoggedInUser)Session["LoggedInUser_Specs"];  
        if (CIsusr != null)  
        {  
            PageType = Page.Request.QueryString["PageType"];  
            if (PageType == null)  
            {  
                PageType = "Add";  
            }  
            Session["PageType"] = PageType;  
            pageid = "4";// Request.QueryString["pageid"];  
            ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(CIsusr.UserId),  
Convert.ToInt32(CIsusr.GroupId), PageType);  
        }  
        if (!IsPostBack)  
        {  
            if (PageType=="Edit")  
            {  
                Id =Convert.ToInt32( Request.QueryString["Id"]);  
                EteNoteType Hold = new EteNoteType();  
                Hold = new NoteTypes().getNoteTypeById(Id);  
                if (Hold != null)  
                {  
                    txt_holdname.Text = Hold.Name;  
                    Details.Text = Hold.Description;  
                }  
            }  
            else  
            {
```

```

        Common.ShowMessageHtml("Missing Tasks Details ", 0, true, dvMsgBox);
    }
}

}

private void ManageNavigations(int pageid, int userid, int groupid, string PageType)
{
    bool IsLeftBar = false;
    IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
myBtnClickHandler);

    if (IsLeftBar == true)
    {
        LeftSideBar.Attributes["class"] = "span2";
        MainContent.Attributes["class"] = "span10";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;"); //style="display:
block;"
        PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
    else
    {
        LeftSideBar.Attributes["class"] = "";
        MainContent.Attributes["class"] = "span12";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
    }
}

// This is Event Handler
protected void myBtnClickHandler(object sender, EventArgs e)
{
    LinkButton lnk = (LinkButton)sender;
    string ID = lnk.ID;

    DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
    if (ds != null)
    {
        if (ds.Tables[0].Rows.Count > 0)
        {
            DataTable dtdll = ds.Tables[0];
            foreach (DataRow drnew in dtdll.Rows)
            {
                switch (drnew["lbtncode"].ToString())
                {
                    case "SAVE":
                        // Call SAVE Method
                        ContentSaveControlClicked();
                        break;
                    case "AUDIT":

```

```

                Response.Redirect(drnew["pageurl"].ToString() + "&frompage=" +
Request.QueryString["pageid"]);
                break;
            case "URL":
                Response.Redirect(drnew["pageurl"].ToString());
                break;
            case "CANCEL":
                ContentCancelControlClicked();
                break;
            default:
                break;
        }
    }
}
}
}
}
}

```

```

private void ContentCancelControlClicked()
{
    Common.ShowMessageHtml("", 0, false, dvMsgBox);

    PageType = Session["PageType"].ToString();
    if (PageType == "Add")
    {
    }
    else if (PageType == "Edit")
    {
        Response.Redirect("~/NoteTypesList.aspx?pageid=149");
    }
}
private void ContentSaveControlClicked()
{
    //objlog.Trace_Log(0, "DateOrWed", "MangeDealer", "SaveClick", "Start Here..");

    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    try
    {
        if (txt_holdname.Text == "")
        {
            Common.ShowMessageHtml("Name cannot be blank", 0, true, dvMsgBox);

        }
        else if (Details.Text == "")
        {
            Common.ShowMessageHtml("Description cannot be blank", 0, true, dvMsgBox);

```

```

    }
    else
    {
        EteNoteType ETask = new EteNoteType();
        ETask.Id = Convert.ToInt32(Request.QueryString["Id"]);
        ETask.Name = txt_holdname.Text;
        ETask.CreatedOn = DateTime.Now;
        ETask.Uid = Convert.ToInt32(Clsusr.UserId);
        ETask.IsActive = true;
        ETask.Description = Details.Text;

        bool Result = false;
        PageType = Session["PageType"].ToString();

        if (PageType == "Add")
        {
            Result = new NoteTypes().Insert(ETask);
        }
        else
        {
            Result = new NoteTypes().Update(ETask);
        }
        if (Result == true)
        {
            if (PageType == "Add")
            {
                Common.ShowMessageHtml("Adding of Note Type Sucessful ", 1, true,
dvMsgBox);
            }
            else
            {
                Common.ShowMessageHtml("Note Type Edited Sucessfully ", 1, true,
dvMsgBox);
            }
            Response.Redirect("~/NoteTypesList.aspx?pageid=149&msg=Note Type
Saved Sucessfully");
        }
    }
}
}

catch (Exception exc)
{
    //objlog.Trace_Error(4, "DateOrWed", "MangeDealer", "Load", "Exception:" +
exc.ToString());
    Common.ShowMessageHtml(exc.ToString(), 0, true, dvMsgBox);
}

//objlog.Trace_Log(0, "DateOrWed", "MangeDealer", "SaveClick", "Start Here..");
}
}

```

```

}
***** ManagePackTypes.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="ManagePackTypes.aspx.cs"
Inherits="ManagePackTypes" %>

<%@ Register Assembly="Infragistics4.Web.v14.2, Version=14.2.20142.2480,
Culture=neutral, PublicKeyToken=7dd5c3163f2cd0cb"
Namespace="Infragistics.Web.UI.GridControls" TagPrefix="ig" %>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
    <script type="text/javascript">
        function checkTextAreaMaxLength(textBox, e, length) {

            var mLen = textBox["MaxLength"];
            if (null == mLen)
                mLen = length;

            var maxLength = parseInt(mLen);
            if (!checkSpecialKeys(e)) {
                if (textBox.value.length > maxLength - 1) {
                    if (window.event)//IE
                        e.returnValue = false;
                    else
                        e.preventDefault();
                }
            }
        }

        function isNumberKey(evt) {
            var charCode = (evt.which) ? evt.which : event.keyCode;
            if (charCode != 46 && charCode > 31 && (charCode < 48 || charCode > 57)) {
                return false;
            } else {
                // If the number field already has . then don't allow to enter . again.
                if (evt.target.value.search(/\./) > -1 && charCode == 46) {
                    return false;
                }
                return true;
            }
        }
    </script>
</asp:Content>

<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
    <div class="container-fluid">
        <div class="row-fluid" style="padding-top: 20px;">
            <div class="span2" id="dvlogo" runat="server" style="display: block;">
                <%----%>

```



```

using System.Data.SqlClient;
using System.Configuration;

public partial class ManagePackTypes : System.Web.UI.Page
{
    public string PageType;
    string pageid;
    int Id;
    DDLogTracer objlog = new DDLogTracer();

    protected void Page_Load(object sender, EventArgs e)
    {
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr != null)
        {
            PageType = Page.Request.QueryString["PageType"];
            if (PageType == null)
            {
                PageType = "Add";
            }
            Session["PageType"] = PageType;
            pageid = "4";// Request.QueryString["pageid"];
            ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
            Convert.ToInt32(Clsusr.GroupId), PageType);
        }
        if (!IsPostBack)
        {
            if (PageType=="Edit")
            {
                Id =Convert.ToInt32( Request.QueryString["Id"]);
                Etetblpacking Hold = new Etetblpacking();
                Hold = new Products().GetProducttblpackingbyId(Id);
                if (Hold != null)
                {
                    txt_holdname.Text = Hold.PackShort;
                    Details.Text = Hold.PackDescription;
                }
            }
            else
            {
                Common.ShowMessageHtml("Missing Tasks Details ", 0, true, dvMsgBox);
            }
        }
    }

    private void ManageNavigations(int pageid, int userid, int groupid, string PageType)
    {
        bool IsLeftBar = false;
        IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
        myBtnClickHandler);
    }
}

```

```

    if (IsLeftBar == true)
    {
        LeftSideBar.Attributes["class"] = "span2";
        MainContent.Attributes["class"] = "span10";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;"); //style="display:
block;"
        PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
    else
    {
        LeftSideBar.Attributes["class"] = "";
        MainContent.Attributes["class"] = "span12";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
    }
}

```

```

// This is Event Handler
protected void myBtnClickHandler(object sender, EventArgs e)
{
    LinkButton lnk = (LinkButton)sender;
    string ID = lnk.ID;

    DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
    if (ds != null)
    {
        if (ds.Tables[0].Rows.Count > 0)
        {
            DataTable dtdll = ds.Tables[0];
            foreach (DataRow drnew in dtdll.Rows)
            {
                switch (drnew["lbtncode"].ToString())
                {
                    case "SAVE":
                        // Call SAVE Method
                        ContentSaveControlClicked();
                        break;
                    case "AUDIT":
                        Response.Redirect(drnew["pageurl"].ToString() + "&frompage=" +
Request.QueryString["pageid"]);
                        break;
                    case "URL":
                        Response.Redirect(drnew["pageurl"].ToString());
                        break;
                    case "CANCEL":
                        ContentCancelControlClicked();
                        break;
                    default:
                        break;
                }
            }
        }
    }
}

```

```
    }  
  }  
}
```

```
private void ContentCancelControlClicked()  
{  
    Common.ShowMessageHtml("", 0, false, dvMsgBox);  
  
    PageType = Session["PageType"].ToString();  
    if (PageType == "Add")  
    {  
  
    }  
    else if (PageType == "Edit")  
    {  
        Response.Redirect("~/PackTypesList.aspx?pageid=6");  
    }  
}  
private void ContentSaveControlClicked()  
{  
    //objlog.Trace_Log(0, "DateOrWed", "MangeDealer", "SaveClick", "Start Here..");  
  
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];  
    try  
    {  
        if (txt_holdname.Text == "")  
        {  
            Common.ShowMessageHtml("Name cannot be blank", 0, true, dvMsgBox);  
  
        }  
        else if (Details.Text == "")  
        {  
  
            Common.ShowMessageHtml("Description cannot be blank", 0, true, dvMsgBox);  
  
        }  
    }  
    else  
    {  
        Etetblpacking ETask = new Etetblpacking();  
        ETask.PackID = Convert.ToInt32(Request.QueryString["Id"]);  
        ETask.PackShort = txt_holdname.Text;  
        ETask.CreatedOn = DateTime.Now;  
        ETask.UserId = Convert.ToInt32(Clsusr.UserId);  
  
        ETask.PackDescription = Details.Text;  
  
        bool Result = false;  
        PageType = Session["PageType"].ToString();  
    }  
}
```

```

        if (PageType == "Add")
        {
            Result = new Products().InsertPacking(ETask);
        }
        else
        {
            Result = new Products().UpdatePacking(ETask);
        }
        if (Result == true)
        {
            if (PageType == "Add")
            {
                Common.ShowMessageHtml("Adding of Pack Type Sucessful ", 1, true,
dvMsgBox);
            }
            else
            {
                Common.ShowMessageHtml("Pack Type Edited Sucessfully ", 1, true,
dvMsgBox);
            }
            Response.Redirect("~/PackTypesList.aspx?pageid=6&msg=Note Type Saved
Sucessfully");
        }
    }
}
}

catch (Exception ex)
{
    Common.ShowMessageHtml(ex.Message, 1, true, dvMsgBox);
    ExceptionLogging.SendExcepToDB(ex);

    objlog.Trace_Error(5, "DateOrWed", "OrderSearchList", "Bindetails_Filter",
ex.Message);
}
objlog.Trace_Log(0, "DateOrWed", "MangeDealer", "SaveClick", "Start Here..");
}

}

}
***** ManagePayment.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="ManagePayment.aspx.cs" Inherits="ManagePayment"
%>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">

```

```

<style type="text/css">
    .auto-style1 {
        width: 100%;
    }
</style>
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="CPH_MainContent" runat="Server">
    <div id="mainContent">
        <h2> Underdevelopment </h2>
        <table class="auto-style1">
            <tr>
                <td>&nbsp;</td>
                <td>&nbsp;</td>
                <td>&nbsp;</td>
                <td>&nbsp;</td>
                <td>&nbsp;</td>
            </tr>
            <tr>
                <td>&nbsp;</td>
                <td>&nbsp;</td>
                <td>&nbsp;</td>
                <td>&nbsp;</td>
                <td>&nbsp;</td>
            </tr>
            <tr>
                <td>&nbsp;</td>
                <td colspan="3">&nbsp;</td>
                <td>&nbsp;</td>
            </tr>
            <tr>
                <td>&nbsp;</td>
                <td>&nbsp;</td>
                <td>&nbsp;</td>
                <td>&nbsp;</td>
                <td>&nbsp;</td>
            </tr>
            <tr>
                <td>&nbsp;</td>
                <td>&nbsp;</td>
                <td>&nbsp;</td>
                <td>&nbsp;</td>
                <td>&nbsp;</td>
            </tr>
        </table>
    </div>
</asp:Content>

```

```

***** ManagePayment.aspx.cs *****
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

```

```

public partial class ManagePayment: System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {

    }

}
}
***** ManagePermission.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="ManagePermission.aspx.cs"
Inherits="ManagePermission" %>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
</asp:Content>

<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
    <div class="container-fluid">
        <div class="row-fluid" style="padding-top: 20px;">
            <div class="span2" id="dvlogo" runat="server" style="display: block;">
                <%----%>
            </div>
            <div class="span10">
                <div class="row">
                    <div id="dvMsgBox" class="span12 ordersave" runat="server" style="display:
none;">
                        <asp:Label ID="Label1" runat="server" Text="Label"></asp:Label>
                    </div>
                </div>
            </div>
        </div>
    </div>
</asp:Content>

<asp:Content ID="Content4" ContentPlaceHolderID="CPH_LeftSideBar" runat="Server">
    <div id="LeftSideBar" runat="server" class="span2">
        <asp:Table ID="Table1" runat="server">
        </asp:Table>
    </div>
</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="CPH_MainContent" runat="server">

    <div class="span10" id="MainContent" runat="server">
        <div id="ShowPageHeader" class="row" style="display: block;" runat="server">
            <div id="PageName" class="span12 customer" runat="server">
            </div>
        </div>

        <div id="Div1" class="row" style="display: block;">
            <div id="Div2" class="span12 customertable">

                <!-- Place your content here-->

```



```

        <asp:DropDownList ID="ddl_Groups" runat="server"
OnSelectedIndexChanged="ddl_Groups_SelectedIndexChanged" Style="float: left;"
AutoPostBack="True">
        </asp:DropDownList>

        <div id="ShowPermission_Enable" runat="server" style="display: none; margin:
3px 0px 20px 10px;">

                <div id="div" class="btn btn-primary" onclick="LoadxmlPermissions();"
style="margin: 0px 0px 0px 10px;">Show Permission</div>
                <div id="processimage">
                        
                </div>

                <%--                <div id="div_update" class="btn btn-primary"
onclick="UpdateGroupPermission();">
                        <span>Update Permission</span>
                </div>--%>
        </div>

        <div id="PermissionGroup" runat="server" style="display: none;">
        </div>

</div>
</div>
</div>
</asp:Content>

```

```

***** ManagePermission.aspx.cs *****

```

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBEntity;
using DowBAL;
using System.Web.UI.HtmlControls;
using System.Data;

```

```

public partial class ManagePermission : System.Web.UI.Page
{
    public string groupid { get; set; }
    public string JSString { get; set; }

```

```

protected void Page_Load(object sender, EventArgs e)
{

```

```

    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    if (Clsusr != null)
    {
        groupid = Clsusr.GroupId;
    }

```

```

string pageid = Request.QueryString["pageid"];
ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
Convert.ToInt32(Clsusr.GroupId));

if (IsPostBack == false)
{
    // Button1.Attributes.Add("onclick", "LoadxmlPermissions('groupname');");

    PopDlls ObjPD = new PopDlls();

    ObjPD.BindDropDowns(ddl_Groups, (new
PopDlls()).GetGroups(Convert.ToInt32(Clsusr.UserId), Convert.ToInt32(Clsusr.GroupId)));
    //ddl_Groups.SelectedItem.Value = groupid;
    ddl_Groups.Items.Insert(0, new ListItem("--Select Group--", "0"));
    ddl_Groups.SelectedIndex = 0;

}
}
else
{
    Response.Redirect("MLB_Login.aspx");
}

}

// This will manage the page headings and Leftbar
private void ManageNavigations(int pageid, int userid, int groupid)
{
    bool IsLeftBar = false;
    IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
myBtnClickHandler);

    if (IsLeftBar == true)
    {
        LeftSideBar.Attributes["class"] = "span2";
        MainContent.Attributes["class"] = "span10";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;"); //style="display:
block;";
        PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
    else
    {
        LeftSideBar.Attributes["class"] = "";
        MainContent.Attributes["class"] = "span12";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
    }
}

// This is Event Handler
protected void myBtnClickHandler(object sender, EventArgs e)
{

```

```

LinkButton lnk = (LinkButton)sender;
string ID = lnk.ID;

DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
if (ds != null)
{
    if (ds.Tables[0].Rows.Count > 0)
    {
        DataTable dtdll = ds.Tables[0];
        foreach (DataRow drnew in dtdll.Rows)
        {
            switch (drnew["lbtncode"].ToString())
            {
                case "SAVE":
                    // Call SAVE Method
                    break;
                case "URL":
                    Response.Redirect(drnew["pageurl"].ToString());
                    break;
                case "AUDIT":
                    Response.Redirect(drnew["pageurl"].ToString() + "&frompage=" +
Request.QueryString["pageid"]);
                    break;

                case "CANCEL":
                    // Call Cancel Method
                    break;
                default:
                    break;
            }
        }
    }
}

//protected void page_LoadComplete(object sender, EventArgs e)
//{
//    PermissionsGroup[] PerGrp = new PermissionsGroup[] { };
//    PerGrp =
GetAttributesList(Convert.ToInt32(Convert.ToInt32(ddl_Groups.SelectedItem.Value)));
//    string js = Newtonsoft.Json.JsonConvert.SerializeObject(PerGrp);
//    JSString = js;

//    string script = string.Format("<script type=\"text/javascript\"> LoadxmlPermissions();
</script>");
//    ClientScript.RegisterClientScriptBlock(this.GetType(), "myscript", script);
//}
//public static PermissionsGroup[] GetAttributes(int Groupid)
//{
//    try
//    {

```

```

// List<mapids> lst = new List<mapids>();
// int Pageid = 0, arr = 0;

// lst = new tblGrpMappings().GetPerSetsForGroup(Groupid).ToList();
// int lstindex = lst.Count();
// PermissionsGroup[] PerGrp = new PermissionsGroup[lstindex];
// foreach (var Pgrow in lst)
// {

//     PerGrp[arr].mpageid = Pgrow.mpageid;
//     PerGrp[arr].buttonid = Pgrow.buttonid;
//     PerGrp[arr].status = Pgrow.status;
//     Pageid = Pgrow.pageid;
//     List<mapids> blst = new List<mapids>();
//     blst = new tblGrpMappings().GetLeftButtonsPermission(Pageid, Groupid).ToList();
//     foreach (var Btrow in blst)
//     {
//         arr++;
//         PerGrp[arr].mpageid = Btrow.mpageid;
//         PerGrp[arr].buttonid = Btrow.buttonid;
//         PerGrp[arr].status = Btrow.status;
//     } arr++;
// }

// return PerGrp;
// }
// catch (Exception exe)
// {
//     return null;
// }
//}

```

```

public static PermissionsGroup[] GetAttributesList(int Groupid,int type)
{
    try
    {
        List<PermissionsGroup> Pergrp = new List<PermissionsGroup>();
        List<mapids> lst = new List<mapids>();
        int Pageid = 0;

        lst = new tblGrpMappings().GetPerSetsForGroup(Groupid,type).ToList();
        int lstindex = lst.Count();
        PermissionsGroup[] PerGroup;

        foreach (var Pgrow in lst)
        {

            Pergrp.Add(new PermissionsGroup
            {

                mpageid = Pgrow.mpageid,
                status = Pgrow.status

            });

```

```

        Pageid = Pgrow.pageid;
        List<mapids> blst = new List<mapids>();
        blst = new tblGrpMappings().GetLeftButtonsPermission(Pageid, Groupid).ToList();
        foreach (var Btrow in blst)
        {
            Pergrp.Add(new PermissionsGroup
            {
                buttonid = Btrow.buttonid,
                status = Btrow.status
            });
        }
        PerGroup = Pergrp.ToArray();
        return PerGroup;
    }
    catch (Exception exe)
    {
        return null;
    }
}

//protected override void OnInit(EventArgs e)
//{
//    base.OnInit(e);

//    if (this.Master is MainMaster)
//    {
//        ((MainMaster)this.Master).MasterControlClicked += new
EventHandler(ContentControlClicked);
//    }
//}

//private void ContentControlClicked(object sender, EventArgs e)
//{

//}

/*Recursive method that populates the Dynamic Menus*/

[System.Web.Services.WebMethod]
public static string UpdateAttributes(string XmlAttr)
{
    try
    {
        if (XmlAttr.Length <= 0) { return "Permissions string is empty."; }
        else
        {
            List<EtetblGrpMapping> Istgrpmap = new List<EtetblGrpMapping>();
            List<EtetblPageButtonMapping> Istbtn = new List<EtetblPageButtonMapping>();

            int i_pagebtnid, i_mapid, status;
            string failed = "";

```

```

        PermissionsGroup[] PerGrp =
Newtonsoft.Json.JsonConvert.DeserializeObject<PermissionsGroup[]>(XmlAttr);

        for (int i = 0; i < PerGrp.GetUpperBound(0); i++)
        {
            if (PerGrp[i].mpageid != null && PerGrp[i].mpageid != "")
            {
                i_mapid = Convert.ToInt32(PerGrp[i].mpageid);
                status = Convert.ToInt32(PerGrp[i].status);
                Istgrpmap.Add(new EtetblGrpMapping { MapId = i_mapid, view =
Convert.ToBoolean(status) });
            }
            else if (PerGrp[i].buttonid != null && PerGrp[i].buttonid != "")
            {
                i_pagebtnid = Convert.ToInt32(PerGrp[i].buttonid);
                status = Convert.ToInt32(PerGrp[i].status);
                Istbtn.Add(new EtetblPageButtonMapping
                {
                    pagebtnid = i_pagebtnid,
                    show = status
                });
            }
            else
            {
                failed = failed + System.Environment.NewLine + "pageid:" +
PerGrp[i].mpageid + "|buttonid:" + PerGrp[i].buttonid + "|status:" + PerGrp[i].status;
            }
        }

        bool pageresult = false, buttonresult = false;

        pageresult = new tblGrpMappings().UpdatePageList(Istgrpmap);
//UpdateButtonList
        buttonresult = new tblPageButtonMapings().UpdateButtonList(Istbtn);
    }
    return "Permission updated successfully";
}
catch (Exception exe)
{
    return "Error occured:" + exe.ToString();
}
}

protected void Button1_Click(object sender, EventArgs e)
{
    try
    {
        //PermissionsGroup[] PerGrp = new PermissionsGroup[] { };
        //PerGrp = GetAttributesList(Convert.ToInt32(ddl_Groups.SelectedItem.Value));

        // string js =
Newtonsoft.Json.JsonConvert.DeserializeObject<PermissionsGroup[]>(PerGrp);
        //string js = Newtonsoft.Json.JsonConvert.SerializeObject(PerGrp);
    }
}

```

```

        //string script = string.Format("<script type=\"text/javascript\">
LoadxmlPermissions('{0}'); </script>", js);

        string script = string.Format("<script type=\"text/javascript\"> LoadxmlPermissions();
</script>");
        ClientScript.RegisterClientScriptBlock(this.GetType(), "myscript", script);

        //string[] arr1 = new string[PerGrp.Length];
        // for (int i = 0; i < PerGrp.Length; i++)
        // {
        //     if(PerGrp[i].buttonid==null)
        //     {
        //         arr1[i] = "{mpageid"+":" + PerGrp[i].mpageid + ",status:" + PerGrp[i].status +
"}";
        //     }
        //     else if(PerGrp[i].mpageid==null)
        //     {
        //         arr1[i] = "{buttonid:" + PerGrp[i].buttonid + ",status:" + PerGrp[i].status + ""}";
        //     }

        // }

        ///string js =
Newtonsoft.Json.JsonConvert.DeserializeObject<PermissionsGroup[]>(PerGrp);
        //string js = Newtonsoft.Json.JsonConvert.SerializeObject(arr1);
        //btnDoRegister.Attributes.Add("onclick", "funForButton()");

    }
    catch (Exception exe)
    {
        throw (exe);
    }
}

[System.Web.Services.WebMethod]
public static string ResetPermissions(string groupid)
{
    try
    {
        if (groupid.Length <= 0)
        {
            return "Error: groupid not send.";
        }
        else
        {
            PermissionsGroup[] PerGrp = new PermissionsGroup[] { };
            PerGrp = GetAttributesList(Convert.ToInt32(groupid), 1);
            string js = Newtonsoft.Json.JsonConvert.SerializeObject(PerGrp);
            return js;
            //return PerGrp;
        }
    }
}
}

```

```

catch (Exception exe)
{
    return "Error: exception - " + exe.ToString();
}
}

protected void ddl_Groups_SelectedIndexChanged(object sender, EventArgs e)
{
    ShowPermission_Enable.Attributes.CssStyle.Add("display", "none;");

    int selectedGroupid = Convert.ToInt32(ddl_Groups.SelectedItem.Value);
    if (selectedGroupid > 0)
    {
        PermissionGroup.InnerHtml = Common.Permission_Group(selectedGroupid, 0);
        ShowPermission_Enable.Attributes.CssStyle.Add("display", "block;");
    }
}
}
}
***** ManageProductCat.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="ManageProductCat.aspx.cs"
Inherits="ManageProductCat" %>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
    <script type="text/javascript">
        function checkTextAreaMaxLength(textBox, e, length) {

            var mLen = textBox["MaxLength"];
            if (null == mLen)
                mLen = length;

            var maxLength = parseInt(mLen);
            if (!checkSpecialKeys(e)) {
                if (textBox.value.length > maxLength - 1) {
                    if (window.event)//IE
                        e.returnValue = false;
                    else
                        e.preventDefault();
                }
            }
        }

    }

function isNumberKey(evt) {
    var charCode = (evt.which) ? evt.which : event.keyCode;
    if (charCode != 46 && charCode > 31 && (charCode < 48 || charCode > 57)) {
        return false;
    } else {
        // If the number field already has . then don't allow to enter . again.
        if (evt.target.value.search(/\./) > -1 && charCode == 46) {
            return false;
        }
    }
}

```


</asp:Content>

***** ManageProductCat.aspx.cs *****

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.IO;
using System.Data;
using System.Text;
using DowBAL;
using System.Xml;
using DowBEntity;
```

```
public partial class ManageProductCat: System.Web.UI.Page
{
    DDLogTracer objlog = new DDLogTracer();

    public string PageType;
    string pageid;
    int Id;
    protected void Page_Load(object sender, EventArgs e)
    {
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr != null)
        {
            PageType = Page.Request.QueryString["PageType"];
            if (PageType == null)
            {
                PageType = "Add";
            }
            Session["PageType"] = PageType;
            pageid = "4";// Request.QueryString["pageid"];
            ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
            Convert.ToInt32(Clsusr.GroupId), PageType);
        }
        if (!IsPostBack)
        {
            if (PageType == "Edit")
            {
                Id = Convert.ToInt32(Request.QueryString["Id"]);
                Eteproductcat Hold = new Eteproductcat();
                Hold = new Products().GetProductCat(Id);
                if (Hold != null)
                {
                    txt_holdname.Text = Hold.pcatdesc;
                    Details.Text = Hold.pcatdesc;
                }
            }
            else
            {

```

```

        Common.ShowMessageHtml("Missing Tasks Details ", 0, true, dvMsgBox);
    }
}
}

private void ManageNavigations(int pageid, int userid, int groupid, string PageType)
{
    bool IsLeftBar = false;
    IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
myBtnClickHandler);

    PageName.InnerText = "Manage Product Categories";
    if (IsLeftBar == true)
    {
        LeftSideBar.Attributes["class"] = "span2";
        MainContent.Attributes["class"] = "span10";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;"); //style="display:
block;"
    }
    else
    {
        LeftSideBar.Attributes["class"] = "";
        MainContent.Attributes["class"] = "span12";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
    }
}

// This is Event Handler
protected void myBtnClickHandler(object sender, EventArgs e)
{
    LinkButton lnk = (LinkButton)sender;
    string ID = lnk.ID;

    DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
    if (ds != null)
    {
        if (ds.Tables[0].Rows.Count > 0)
        {
            DataTable dtdll = ds.Tables[0];
            foreach (DataRow drnew in dtdll.Rows)
            {
                switch (drnew["lbtncode"].ToString())
                {
                    case "SAVE":
                        // Call SAVE Method
                        ContentSaveControlClicked();
                        break;
                    case "AUDIT":
                        Response.Redirect(drnew["pageurl"].ToString() + "&frompage=" +
Request.QueryString["pageid"]);
                }
            }
        }
    }
}

```



```

<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="ManageProductList.aspx.cs"
Inherits="ManageProductList" %>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
</asp:Content>

<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
<div class="container-fluid">
<div class="row-fluid" style="padding-top: 20px;">
<div class="span2" id="dvlogo" runat="server" style="display: block;">
<%----%>
</div>
<div class="span10">
<div class="row">
<div id="dvMsgBox" class="span12 ordersave" runat="server" style="display:
none;">
<asp:Label ID="Label1" runat="server" Text="Label"></asp:Label>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
</asp:Content>

<asp:Content ID="Content4" ContentPlaceHolderID="CPH_LeftSideBar" runat="Server">
<div id="LeftSideBar" runat="server" class="span2">
<asp:Table ID="Table1" runat="server">
</asp:Table>
</div>
</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="CPH_MainContent" runat="server">

<div class="span10" id="MainContent" runat="server">
<div id="ShowPageHeader" class="row" style="display: block;" runat="server">
<div id="PageName" class="span12 customer" runat="server">
</div>
</div>

<div id="Div1" class="row" style="display: block;">
<div id="Div2" class="span12 customertable">

<!-- Place your content here-->

<asp:DropDownList ID="ddl_Groups" runat="server"
OnSelectedIndexChanged="ddl_Groups_SelectedIndexChanged" Style="float: left;"
AutoPostBack="True">
</asp:DropDownList>

<div id="ShowPermission_Enable" runat="server" style="display: none; margin:
3px 0px 20px 10px;">

```

```

        <div id="div" class="btn btn-primary" onclick="LoadxmlProducts();"
style="margin: 0px 0px 0px 10px;">Show Products</div>
        <div id="processimage">
            
        </div>

        <%--          <div id="div_update" class="btn btn-primary"
onclick="UpdateGroupPermission();">
            <span>Update Permission</span>
        </div>--%>
    </div>

    <div id="PermissionGroup" runat="server" style="display: none;">
    </div>

</div>
</div>
</div>

```

```
</asp:Content>
```

```
***** ManageProductList.aspx.cs *****
```

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBEntity;
using DowBAL;
using System.Web.UI.HtmlControls;
using System.Data;

public partial class ManageProductList: System.Web.UI.Page
{
    public string groupid { get; set; }
    public string JSString { get; set; }

    protected void Page_Load(object sender, EventArgs e)
    {
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr != null)
        {
            groupid = Clsusr.GroupId;
            string pageid = Request.QueryString["pageid"];
            ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
            Convert.ToInt32(Clsusr.GroupId));

            if (IsPostBack == false)
            {
                // Button1.Attributes.Add("onclick", "LoadxmlProducts('groupname');");

                PopDlls ObjPD = new PopDlls();
            }
        }
    }
}

```



```

        ObjPD.BindDropDowns(ddl_Groups, (new PopDlls()).GetProductCats());
        //ddl_Groups.SelectedItem.Value = groupid;
//        ddl_Groups.Items.Insert(0, new ListItem("--Select Category--", "0"));
        ddl_Groups.SelectedIndex = 0;

    }
}
else
{
    Response.Redirect("MLB_Login.aspx");
}

}

// This will manage the page headings and Leftbar
private void ManageNavigations(int pageid, int userid, int groupid)
{
    bool IsLeftBar = false;
    IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
myBtnClickHandler);

    if (IsLeftBar == true)
    {
        LeftSideBar.Attributes["class"] = "span2";
        MainContent.Attributes["class"] = "span10";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;"); //style="display:
block;"
        PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
    else
    {
        LeftSideBar.Attributes["class"] = "";
        MainContent.Attributes["class"] = "span12";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
    }
}

// This is Event Handler
protected void myBtnClickHandler(object sender, EventArgs e)
{
    LinkButton lnk = (LinkButton)sender;
    string ID = lnk.ID;

    DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
    if (ds != null)
    {
        if (ds.Tables[0].Rows.Count > 0)
        {
            DataTable dtdll = ds.Tables[0];
            foreach (DataRow drnew in dtdll.Rows)

```



```

Pageid = Pgrow.pageid;
List<mapids> blst = new List<mapids>();
blst = new tblGrpMappings().GetLeftButtonsPermission(Pageid, Groupid).ToList();
foreach (var Bthrow in blst)
{
    Pergrp.Add(new PermissionsGroup
    {
        buttonid = Bthrow.buttonid,
        status = Bthrow.status
    });
}
}
PerGroup = Pergrp.ToArray();
return PerGroup;
}
catch (Exception exe)
{
    return null;
}
}

```

/*Recursive method that populates the Dynamic Menus*/

```

[System.Web.Services.WebMethod]
public static string UpdateAttributes(string XmlAttr)
{
    try
    {
        if (XmlAttr.Length <= 0) { return "Permissions string is empty."; }
        else
        {
            List<EtetblGrpMapping> lstgrpmap = new List<EtetblGrpMapping>();
            List<EtetblPageButtonMapping> lstbtn = new List<EtetblPageButtonMapping>();

            int i_pagebtnid, i_mapid, status;
            string failed = "";

            PermissionsGroup[] PerGrp =
Newtonsoft.Json.JsonConvert.DeserializeObject<PermissionsGroup[]>(XmlAttr);

            for (int i = 0; i < PerGrp.GetUpperBound(0); i++)
            {
                if (PerGrp[i].mpageid != null && PerGrp[i].mpageid != "")
                {
                    i_mapid = Convert.ToInt32(PerGrp[i].mpageid);
                    status = Convert.ToInt32(PerGrp[i].status);
                    lstgrpmap.Add(new EtetblGrpMapping { MapId = i_mapid, view =
Convert.ToBoolean(status) });
                }
                else if (PerGrp[i].buttonid != null && PerGrp[i].buttonid != "")
                {
                    i_pagebtnid = Convert.ToInt32(PerGrp[i].buttonid);
                    status = Convert.ToInt32(PerGrp[i].status);

```

```

        Istbtn.Add(new EtetblPageButtonMapping
        {
            pagebtnid = i_pagebtnid,
            show = status
        });
    }
    else
    {
        failed = failed + System.Environment.NewLine + "pageid:" +
PerGrp[i].mpageid + "|buttonid:" + PerGrp[i].buttonid + "|status:" + PerGrp[i].status;
    }
}

bool pageresult = false, buttonresult = false;

pageresult = new tblGrpMappings().UpdatePageList(Istgrpmap);
//UpdateButtonList
buttonresult = new tblPageButtonMapings().UpdateButtonList(Istbtn);
}
return "Permission updated successfully";
}
catch (Exception exe)
{
    return "Error occured:" + exe.ToString();
}
}

protected void Button1_Click(object sender, EventArgs e)
{
    try
    {

        string script = string.Format("<script type=\"text/javascript\"> LoadxmlProducts();
</script>");
        ClientScript.RegisterClientScriptBlock(this.GetType(), "myscript", script);

    }
    catch (Exception exe)
    {
        throw (exe);
    }
}

[System.Web.Services.WebMethod]
public static string ResetPermissions(string groupid)
{
    try
    {
        if (groupid.Length <= 0)
        {
            return "Error: groupid not send.";
        }
    }
}

```

```

    }
    else
    {
        PermissionsGroup[] PerGrp = new PermissionsGroup[] { };
        PerGrp = GetAttributesList(Convert.ToInt32(groupid),1);
        string js = Newtonsoft.Json.JsonConvert.SerializeObject(PerGrp);
        return js;
        //return PerGrp;
    }

}
catch (Exception exe)
{
    return "Error: exception - " + exe.ToString();
}
}

protected void ddl_Groups_SelectedIndexChanged(object sender, EventArgs e)
{
    ShowPermission_Enable.Attributes.CssStyle.Add("display", "none;");

    int selectedGroupid = Convert.ToInt32(ddl_Groups.SelectedItem.Value);
    if (selectedGroupid > 0)
    {
        PermissionGroup.InnerHtml = Common.Products_Group(selectedGroupid, 0);
        ShowPermission_Enable.Attributes.CssStyle.Add("display", "block;");
    }

}
}
}
***** ManageProfile.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="ManageProfile.aspx.cs" Inherits="ManageProfile" %>

<%@ Register Assembly="Infragistics4.Web.v14.2, Version=14.2.20142.2480,
Culture=neutral, PublicKeyToken=7dd5c3163f2cd0cb"
Namespace="Infragistics.Web.UI.ListControls" TagPrefix="ig" %>

<%@ Register Assembly="Infragistics4.Web.v14.2, Version=14.2.20142.2480,
Culture=neutral, PublicKeyToken=7dd5c3163f2cd0cb"
Namespace="Infragistics.Web.UI.GridControls" TagPrefix="ig" %>
<%--<%@ Register Assembly="AjaxControlToolkit" Namespace="AjaxControlToolkit"
TagPrefix="cc1" %>--%>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
<link rel="stylesheet" href="oldcss/css/bootstrap.css"/>
<link href="oldcss/CSS/PopupModal.css" rel="stylesheet" type="text/css" />
<script type="text/javascript" src="oldcss/js/jquery-1.6.1.min.js"></script>
<script type="text/javascript" src="oldcss/js/ddsmoothmenu.js"></script>
<script type="text/javascript" src="oldcss/js/bootstrap.js"></script>
<script type="text/javascript" src="oldcss/js/bootstrap.min.js"></script>

```

```

<script type="text/javascript" src="oldcss/js/jquery.js"></script>
<script type="text/javascript" src="oldcss/js/bootstrap-collapse.js"></script>

<link href="oldcss/css/tabstyle.css" type="text/css" rel="stylesheet" />
<style type="text/css">
    .auto-style1 {
        width: 104px;
    }

</style>
<script type="text/javascript" src="oldcss/js/skinable_tabs.min.js"></script>
<script type="text/javascript" src="oldcssjs/jquery-ui.min.js"></script>
<script type="text/javascript">
    $(':tabs_holder').skinableTabs({
        effect: 'basic_display',
        skin: 'skin7',
        position: 'top',
        persist_tab: false
    });
</script>

<script type="text/javascript">
    function Managetabs() {
        document.getElementById('tab3').style.display = "none";
        document.getElementById('tab4').style.display = "none";
        document.getElementById('tab5').style.display = "none";
        document.getElementById('tab6').style.display = "none";
    };
</script>

<script type="text/javascript" src="oldcss/js/CommonScript.js"></script>

<script type="text/javascript">

    $(window).load(function () {
        activeTab = getCookie('seltab');
        //alert("." + activeTab);
        $("." + activeTab).trigger('click');
        //setCookie('seltab', "", 2);
    })

    function saveCookie(ckname) {
        //alert('Tab-' + ckname);
        setCookie('seltab', ckname, 2);
        //alert(ckname);
    }

    function setCookie(cname, cvalue, exdays) {
        var d = new Date();
        d.setTime(d.getTime() + (exdays * 24 * 60 * 60 * 1000));
        var expires = "expires=" + d.toUTCString();
        document.cookie = cname + "=" + cvalue + "; " + expires;
    }

```

```

function getCookie(cname) {
    var name = cname + "=";
    var ca = document.cookie.split(';');
    for (var i = 0; i < ca.length; i++) {
        var c = ca[i];
        while (c.charAt(0) == ' ') c = c.substring(1);
        if (c.indexOf(name) == 0) return c.substring(name.length, c.length);
    }
    return "";
}

```

</script>

</asp:Content>

<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" runat="Server">

<div class="container-fluid">

<div class="row-fluid" style="padding-top: 20px;">

<div class="span2" id="dvlogo" runat="server" style="display: block;">

<%----%>

</div>

<div class="span10">

<div class="row">

<div id="dvMsgBox" class="span12 ordersave" runat="server" style="display: none;">

<asp:Label ID="Label1" runat="server" Text=""></asp:Label>

</div>

</div>

</div>

</div>

</div>

</asp:Content>

<asp:Content ID="Content4" ContentPlaceHolderID="CPH_LeftSideBar" runat="Server">

<div id="LeftSideBar" runat="server" class="span2">

<asp:Table ID="Table1" runat="server">

</asp:Table>

</div>

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="CPH_MainContent" runat="Server">

<div class="span10" id="MainContent" runat="server">

<div id="ShowPageHeader" class="row" style="display: block;" runat="server">

<div id="PageName" class="span12 customer" runat="server">

</div>

</div>

<asp:HiddenField ID="HF_HasStoppedService" runat="server" />

<div id="Div1" class="row" style="display: block;">

<div id="Div2" class="span12 customertable">

<div class="tabs_holder">

```

        <ul id="tabs">
            <li id="tab1" class="tab_selected customer span2"><a href="#your-tab-id-1"
onclick="saveCookie('mctb1')" class="mctb1 customer "><i class="fa fa-th-
large"></i>&nbsp;</i>&nbsp;</a></li>
            <li id="tab2" class="customer span2"><a href="#your-tab-id-2"
onclick="saveCookie('mctb2')" class="mctb2"><i class="fa fa-bank"></i>&nbsp;</i>&nbsp;</a></li>
            <li id="tab3" class="customer span2"><a href="#yourtabid3"
onclick="saveCookie('mctb3')" class="mctb3"><i class="fa fa-
bars"></i>&nbsp;</i>&nbsp;</a></li>
            <li id="tab4" class="customer span2"><a href="#yourtabid4"
onclick="saveCookie('mctb4')" class="mctb4"><i class="fa fa-check-square-
o"></i>&nbsp;</i>&nbsp;</a></li>
            <li id="tab5" class="customer span2" style="display:none"><a
href="#yourtabid5" onclick="saveCookie('mctb5')" class="mctb5"><i class="fa fa-
tasks"></i>&nbsp;</i>&nbsp;</a></li>
            <li id="tab6" class="customer span2" style="display:none"><a
href="#yourtabid6" onclick="saveCookie('mctb6')" class="mctb6" style="display:none"><i
class="fa fa-envelope-o"></i>&nbsp;</i>&nbsp;</a></li>
        </ul>
        <div class="content_holder">

        <div id="your-tab-id-1">

            <div class="span12 customertable" style="background-color: #f8f9fa;">
                <div class="row-fluid">
                    <div class="span12 customer" style="text-align: left;">BASIC
INFORMATION</div>
                </div>

                <div class="row-fluid">
                    <div class="span12 detailview" style="background-color: #ffffff;
border: 1px solid #eefef; border-top: none; padding: 15px 5px;">

                        <!--<form class="form-horizontal">-->
                        <div class="control-group">
                            <table class="table table-striped table-detail span12">
                                <tbody>

                                    <tr>
                                        <td><strong>Email/User Id :*</strong></td>
                                        <td align="left" class="auto-style1">
                                            <asp:TextBox ID="txt_username" runat="server"
MaxLength="50" TextMode="Email"></asp:TextBox>
                                        </td>
                                    </tr>
                                    <tr>
                                        <td align="right" class="label_txt"><strong>Name
:*</strong></td>
                                        <td align="left">
                                            <asp:TextBox ID="txt_FirstName" runat="server"
MaxLength="100" onkeypress="return validatealpha(event)"></asp:TextBox>
                                        </td>
                                </tbody>
                            </table>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    
```



```

</tr>
<tr>
    <td align="right" class="label_txt"><strong>Company
Name :</strong></td>
    <td width="25%" align="left">
        <asp:TextBox ID="txt_LastName" runat="server"
MaxLength="40" onkeypress="return validatealpha(event)"></asp:TextBox>
    </td>
</tr>

```

```

<tr>
    <td align="right"
class="label_txt"><strong>Country:</strong></td>
    <td width="25%" align="left">
        <asp:DropDownList ID="ddl_country"
runat="server"></asp:DropDownList></td>

```

```
</tr>
```

```

<tr>
    <td align="right" class="label_txt"><strong>Mobile
:*</strong></td>
    <td width="25%" align="left">
        <asp:TextBox ID="txt_mobile" runat="server"
onkeypress="return isNumberKey(event)" MaxLength="12"
TextMode="Phone"></asp:TextBox>
    </td>

```

```
</tr>
```

```

<tr>
    <td align="right" class="label_txt"><strong>LandLine
:*</strong></td>
    <td width="25%" align="left">
        <asp:TextBox ID="txt_landline" runat="server"
onkeypress="return isNumberKey(event)" MaxLength="12"
TextMode="Phone"></asp:TextBox>
    </td>
</tr>

```

```
<tr>
```

```

    <td align="right" class="label_txt"><strong>Role
:*</strong></td>
    <td width="25%" align="left">
        <asp:DropDownList ID="ddl_GroupId"
runat="server"></asp:DropDownList>
    </td>

```

```
</tr>
```

```

<tr>
    <td align="right" class="label_txt"><strong>Password:*</strong></td>
    <td width="25%" align="left">

```

```
                <asp:TextBox ID="txt_Password" runat="server"
MaxLength="40" TextMode="Password"></asp:TextBox>
            </td>
        </tr>
```

```
        <tr>
            <td align="right"
class="label_txt">&nbsp;<strong>Password:*</strong></td>

                <td width="25%" align="left">
                    <asp:TextBox ID="txt_Password1" runat="server"
MaxLength="40" TextMode="Password"></asp:TextBox>
                    <asp:TextBox ID="email_name" runat="server" MaxLength="50"
TextMode="Email" Visible="false"></asp:TextBox>
                </td>
        </tr>
    </tbody>
```

```
</table>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<div id="your-tab-id-2">
```

```
    <div class="span12 customertable" style="background-color: #f8f9fa;">
```

```
        <div class="row-fluid">
```

```
            <!--Profile Account----->
```

```
            <div class="row-fluid">
```

```
                <div class="span12 customer" style="text-align: left;">Profile</div>
```

```
            </div>
```

```
            <div class="row-fluid">
```

```
                <div class="span12" style="background-color: #ffffff; border: 1px
solid #eefef; border-top: none; padding: 5px;">
```

```
                    <table class="table table-striped table-detail span12">
```

```
                        <tbody>
```

```
                            <tr>
```

```
                                <td class="span3" dir="auto">User Name :</td>
```

```
                                <td>
```

```
                                    <asp:DropDownList ID="ddl_users"
runat="server"></asp:DropDownList>
```

```
                                </td>
```

```
                            </tr>
```

```
                            <tr>
```

```
                                <td class="span3" dir="auto">Company Description :</td>
```

```
                                <td>
```

```

        <asp:TextBox ID="Details" runat="server" Width="600px"
Height="100px" Rows="7" style="resize:none" TextMode="MultiLine"
MaxLength="250"></asp:TextBox>
        </td>
    </tr>
    <tr>
        <td class="auto-style1" dir="auto">Business Type</td>
        <td class="auto-style2">
            <asp:TextBox ID="txt_BusinessType" runat="server"
Width="600px" MaxLength="40" ></asp:TextBox>
        </td>
    </tr>
    <tr>
        <td class="auto-style1" dir="auto">Legal Structure:</td>
        <td>
            <asp:DropDownList ID="ddl_legalstr"
runat="server"></asp:DropDownList>
        </td>
    </tr>
    <tr>
        <td class="span3" dir="auto">Year Established :</td>
        <td>
            <asp:TextBox ID="txt_yrest" runat="server"
Width="600px" MaxLength="40" ></asp:TextBox>
        </td>
    </tr>
    <tr>
        <td class="span3" dir="auto">No of Employees</td>
        <td>
            <asp:TextBox ID="txt_noempl" runat="server"
Width="600px" MaxLength="40" ></asp:TextBox>
        </td>
    </tr>
    <tr>
        <td class="span3" dir="auto">Web-URL</td>
        <td>
            <asp:TextBox ID="txt_website" runat="server"
Width="600px" MaxLength="40" ></asp:TextBox>
        </td>
    </tr>
    <tr>
        <td class="span3" dir="auto">Turnover Per Year:</td>
        <td>
            <asp:TextBox ID="txt_turnover" runat="server" MaxLength="40"
Width="250px"></asp:TextBox>
        </td>
    </tr>

```

```

                                <asp:Label ID="curr" runat="server" class="auto-
style1" dir="auto">In Currency:</asp:Label>

                                <asp:DropDownList ID="ddl_currency"
runat="server"></asp:DropDownList>

                                </td>

                                </tr>
                                <tr>
                                <td class="span3" dir="auto">Quality Assurance Accredition
:</td>

                                <td>
                                <asp:TextBox ID="txt_qa" runat="server" Width="600px"
MaxLength="40" ></asp:TextBox>
                                </td>
                                </tr>
                                <tr>
                                <td class="span3" dir="auto">Registration Number :</td>
                                <td>
                                <asp:TextBox ID="txt_regno" runat="server"
Width="600px" MaxLength="40" ></asp:TextBox>
                                </td>
                                </tr>
                                <tr>
                                <td class="span3" dir="auto">D U N S Number :</td>
                                <td>
                                <asp:TextBox ID="txt_dunsno" runat="server"
Width="600px" MaxLength="40" ></asp:TextBox>
                                </td>
                                </tr>
                                <tr>
                                <td class="span3" dir="auto">Specializing in :</td>
                                <td>
                                <asp:TextBox ID="txt_special" runat="server"
Width="600px" MaxLength="40" ></asp:TextBox>
                                </td>
                                </tr>
                                <tr>
                                <td class="span3" dir="auto">Monthly Production Capacity
:</td>

                                <td>
                                <asp:TextBox ID="txt_prodcap" runat="server" MaxLength="40"
Width="250px"></asp:TextBox>

                                <asp:Label ID="tet" runat="server" class="auto-
style1" dir="auto">In Unit of Measure:</asp:Label>

                                <asp:DropDownList ID="ddl_uom"
runat="server"></asp:DropDownList>

```

```

        </td>
      </tr>

      <tr>
        <td>
          <asp:image ToolTip = "ASP Image Control" ID="Image1" runat="server"
            ImageUrl = "ManageProfile.aspx?ImageID=1" Height="156px" Width="174px"></asp:image>
        </td>
      </tr>
    </tbody>
  </table>

  </div>
  <!--Bank Account END-->
</div>

  </div>
</div>
</div>
<div id="yourtabid3">
  <div class="row-fluid">
    <div class="span12 customer" style="text-align: left;">Addresses</div>
  </div>
  <div class="row-fluid">
    <div class="span12 detailview" style="background-color: #ffffff;
border: 1px solid #eefef; border-top: none; padding: 15px 5px;">

      <!--<form class="form-horizontal">-->
      <div class="control-group">
        <ig:WebDataGrid ID="WDGrid_UserAddress" runat="server" CssClass="wdglist"
          AutoGenerateColumns="False">
          <Columns>

<%--      <ig:BoundDataField DataFieldName="UserName" Key="UserName " >
        <Header Text="Contact Name">
        </Header>
--%>      </ig:BoundDataField>
        <ig:BoundDataField DataFieldName="address1" Key="address1">
        <Header Text="Address1">
        </Header>
        </ig:BoundDataField>
        <ig:BoundDataField DataFieldName="address2" Key="address2">
        <Header Text="Address2">
        </Header>
        </ig:BoundDataField>
        <ig:BoundDataField DataFieldName="address3" Key="address3">

```

```

        <Header Text="Address3">
        </Header>
    </ig:BoundDataField>
    <ig:BoundDataField DataFieldName="City" Key="City">
        <Header Text="City">
        </Header>
    </ig:BoundDataField>
    <ig:BoundDataField DataFieldName="pincode" Key="pincode"
Width="50px">
        <Header Text="ZipCode">
        </Header>
    </ig:BoundDataField>

    <ig:BoundDataField DataFieldName="country" Key="country" >
        <Header Text="Country">
        </Header>
    </ig:BoundDataField>
    <ig:BoundDataField DataFieldName="ContactNo" Key="ContactNo">
        <Header Text="Contact No.">
        </Header>
    </ig:BoundDataField>
    <ig:BoundDataField DataFieldName="address_typedesc"
Key="address_typedesc">
        <Header Text="Address Type">
        </Header>
    </ig:BoundDataField>

    </Columns>
    <Behaviors>
        <ig:Paging CurrentPageLinkCssClass="Current" PageLinkCssClass="Link"
PagerCssClass="Pager" PagerMode="NumericFirstLast" PageSize="15" QuickPages="3"
FirstPageText="First" LastPageText="Last" NextPageText="Next"
PreviousPageText="Previous" PagerAppearance="bottom">
        </ig:Paging>

        <ig:Filtering>
        </ig:Filtering>
        <ig:Sorting Enabled="true">
        </ig:Sorting>
    </Behaviors>
</ig:WebDataGrid>

        <table id="Address" class="table table-striped table-detail span12"
runat="server" visible="true">
            <tbody>
                <tr>
                    <td class="span3" dir="auto">Address Type :</td>
                    <td>
                        <asp:DropDownList ID="ddl_addtype"
runat="server"></asp:DropDownList>
                    </td>
                </tr>
                <tr>
                    <td class="span3" dir="auto">Address1 :</td>

```

```

        <td>
            <asp:TextBox ID="txt_add1" runat="server" MaxLength="100"
Width="250px"></asp:TextBox>

        </td>
    </tr>
    <tr>
        <td class="span3" dir="auto">Address2 :</td>

        <td>
            <asp:TextBox ID="txt_add2" runat="server" MaxLength="100"
Width="250px"></asp:TextBox>

        </td>
    </tr>
    <tr>
        <td class="span3" dir="auto">Address3 :</td>

        <td>
            <asp:TextBox ID="txt_add3" runat="server" MaxLength="100"
Width="250px"></asp:TextBox>

        </td>
    </tr>
    <tr>
        <td class="span3" dir="auto">City :</td>

        <td>
            <asp:TextBox ID="txt_city" runat="server" MaxLength="100"
Width="250px"></asp:TextBox>

        </td>
    </tr>
    <tr>
        <td class="span3" dir="auto">Pincode :</td>

        <td>
            <asp:TextBox ID="txt_pincode" runat="server" MaxLength="100"
Width="250px"></asp:TextBox>

        </td>
    </tr>
    <tr>
        <td class="span3" dir="auto">Contact No :</td>

        <td>
            <asp:TextBox ID="txt_contactno" runat="server" MaxLength="100"
Width="250px"></asp:TextBox>

        </td>
    </tr>

```

```

        </tbody>

                </table>

        </div>

</div>

</div>

</div>

<div id="yourtabid4">
    <div class="span12 customertable" style="background-color: #f8f9fa;">
        <div class="row-fluid">

                <!--Notes-->
                <div class="row-fluid">
                    <div class="span12 customer" style="text-align: left;">Pictures /
Images</div>

                </div>
                <div class="row-fluid">
                    <div class="span12" style="background-color: #ffffff; border: 1px
solid #eefef; border-top: none; padding: 5px;">

                        <ig:WebDataGrid ID="WebDataPictures" runat="server" CssClass="wdglist"
AutoGenerateColumns="False">
                            <Columns>
                                <ig:TemplateDataField Key="ImageUrl" Width="60px" >
                                    <ItemTemplate >
                                        <asp:Image ID="Image2" Height = "50" Width = "100"
runat="server" ImageUrl = '<% #Eval("ImageUrl")%>' />
                                        </ItemTemplate>
                                    </ig:TemplateDataField>

                                </Columns>
                                <Behaviors>
                                    <ig:Paging CurrentPageLinkCssClass="Current" PageLinkCssClass="Link"
PagerCssClass="Pager" PagerMode="NumericFirstLast" PageSize="15" QuickPages="3"
FirstPageText="First" LastPageText="Last" NextPageText="Next"
PreviousPageText="Previous" PagerAppearance="bottom">
                                        </ig:Paging>

                                    <ig:Filtering>
                                        </ig:Filtering>
                                    <ig:Sorting Enabled="true">
                                        </ig:Sorting>
                                </Behaviors>
                            </ig:WebDataGrid>

```



```
<ig:WebImageViewer ID="WebImageViewer1" runat="server" Height="150px"
Width="500px">
```

```
</ig:WebImageViewer>
```

```
<table class="table table-striped table-detail span12">
```

```
<tbody><tr>
```

```
<td>
```

```
<asp:image ToolTip = "ASP Image Control" ID="Image2" runat="server"
ImageUrl = "ManageProducts.aspx?ImageID=1" Height="156px" Width="174px"
Visible="false"></asp:image> </td>
```

```
<td> <asp:image ToolTip = "ASP Image Control" ID="Image3" runat="server"
ImageUrl = "ManageProducts.aspx?ImageID=2" Height="156px" Width="174px"
Visible="false"></asp:image>
```

```
<asp:image ToolTip = "ASP Image Control" ID="Image4" runat="server"
ImageUrl = "ManageProducts.aspx?ImageID=3" Height="156px" Width="174px"
Visible="false"> </asp:image>
```

```
<asp:image ToolTip = "ASP Image Control" ID="Image5" runat="server"
ImageUrl = "ManageProducts.aspx?ImageID=4" Height="156px" Width="174px"
Visible="false"> </asp:image>
```

```
</td>
```

```
</tr>
```

```
<tr>
```

```
<td class="span3" dir="auto">Add Images</td>
```

```
<td>
```

```
<asp:FileUpload ID="FileUpload1" runat="server" />
```

```
</td><td>
```

```
<asp:Button ID="btn_upload" runat="server" Text="Upload"
```

```
OnClick="btn_saveimage_Click" CausesValidation="false" />
```

```
</td>
```

```
</tr>
```

```
</tbody>
```

```
</table>
```

```
<table id="Table7" class="table table-striped table-detail span12"
```

```
runat="server">
```

```
</table>
```

```
<!--eENTER Notes BODY HERE----->
```

```
<!--Notes----->
```

```
</div>
```

```
</div>
```

```
<!--Notes Ends----->
```

```

        </div>
    </div>
</div>

<div id="yourtabid5" style="display:none">
    <div class="span12 customertable" style="background-color: #f8f9fa;">
        <div class="row-fluid">

            <!--Tasks----->
            <div class="row-fluid">
                <div class="span12 customer" style="text-align: left;">Emails
Received</div>

            </div>
            <div class="row-fluid">
                <ig:WebDataGrid ID="WebDataEmailRcvd" runat="server" CssClass="wdglist"
AutoGenerateColumns="False">
                    <Columns>

                        <ig:BoundDataField DataFieldName="EMSentUserName"
Key="EMSentUserName ">
                            <Header Text="Sent By Name">
                                </Header>
                            </ig:BoundDataField>
                        <ig:BoundDataField DataFieldName="CreatedOn" Key="CreatedOn">
                            <Header Text="Sent On">
                                </Header>
                            </ig:BoundDataField>
                        <ig:BoundDataField DataFieldName="EmSubject" Key="EmSubject">
                            <Header Text="Subject">
                                </Header>
                            </ig:BoundDataField>
                        <ig:BoundDataField DataFieldName="EmBody" Key="EmBody">
                            <Header Text="Message">
                                </Header>
                            </ig:BoundDataField>

                    </Columns>
                    <Behaviors>
                        <ig:Paging CurrentPageLinkCssClass="Current" PageLinkCssClass="Link"
PagerCssClass="Pager" PagerMode="NumericFirstLast" PageSize="15" QuickPages="3"
FirstPageText="First" LastPageText="Last" NextPageText="Next"
PreviousPageText="Previous" PagerAppearance="bottom">
                            </ig:Paging>

                        <ig:Filtering>
                            </ig:Filtering>
                        <ig:Sorting Enabled="true">
                            </ig:Sorting>
                        </Behaviors>
                    </ig:WebDataGrid>
                    <table class="table table-striped table-detail span12">

                        </table>
                    <%-- Add Tasks starts--%>

```

```

<%--
        <div class="row-fluid">
            <div class="span12 customer" style="text-align: left;"></div>
        </div>
        <table id="Table2" class="table table-striped table-detail span12"
runat="server">

        </table>--%>

        <%--Add Tasks end--%>

        <!--TASKS END-->
    </div>

    </div>
    </div>
    </div>
    <div id="yourtabid6" style="display:none">
        <div class="span12 customertable" style="background-color: #f8f9fa;">
            <div class="row-fluid">

                <!--Tasks----->
                <div class="row-fluid">
                    <div class="span12 customer" style="text-align: left;">Emails
Sent</div>

                </div>
                <div class="row-fluid">
                    <ig:WebDataGrid ID="WebDataEmailSend" runat="server"
CssClass="wdglist" AutoGenerateColumns="False">
                        <Columns>

                            <ig:BoundDataField DataFieldName="EMRcvdUserName"
Key="EMRcvdUserName ">
                                <Header Text="Sent to Name">
                                    </Header>
                                </ig:BoundDataField>
                                <ig:BoundDataField DataFieldName="CreatedOn" Key="CreatedOn">
                                    <Header Text="Sent On">
                                        </Header>
                                    </ig:BoundDataField>
                                    <ig:BoundDataField DataFieldName="EmSubject" Key="EmSubject">
                                        <Header Text="Subject">
                                            </Header>
                                        </ig:BoundDataField>
                                        <ig:BoundDataField DataFieldName="EmBody" Key="EmBody">
                                            <Header Text="Message">
                                                </Header>
                                            </ig:BoundDataField>

                                </Columns>
                                <Behaviors>
                                    <ig:Paging CurrentPageLinkCssClass="Current" PageLinkCssClass="Link"
PagerCssClass="Pager" PagerMode="NumericFirstLast" PageSize="15" QuickPages="3"

```



```

using System.Xml;
using System.Xml.Serialization;
using System.IO;
a
using System.Configuration;

using System.Drawing;
using Infragistics.Web.UI.ListControls;
public partial class ManageProfile : System.Web.UI.Page
{
    public string PageType;
    string pageid;
    int UserId;
    int PrId=0;
    DDLogTracer objlog = new DDLogTracer();

    protected void Page_Load(object sender, EventArgs e)
    {
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr != null)
        {
            PageType = Page.Request.QueryString["PageType"];
            if (PageType == null)
            {
                PageType = "Edit";
            }
            Session["PageType"] = PageType;
            pageid = "4";// Request.QueryString["pageid"];
            ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
            Convert.ToInt32(Clsusr.GroupId), PageType);
        }
        if (!this.IsPostBack)
        {
            PopDlIs ObjPD = new PopDlIs();
            // ObjPD.BindDropDowns(ddl_businessstype, (new PopDlIs()).GetBusinessType());
            ObjPD.BindDropDowns(ddl_legalstr, (new PopDlIs()).GetLegalstr());
            ObjPD.BindDropDowns(ddl_currency, (new PopDlIs()).GetCurrency());
            ObjPD.BindDropDowns(ddl_uom, (new PopDlIs()).Gettbluom());
            ObjPD.BindDropDowns(ddl_users, (new PopDlIs()).GetUserCompany());
            ObjPD.BindDropDowns(ddl_addtype, (new PopDlIs()).GetAddType());
            ObjPD.BindDropDowns(ddl_country, (new PopDlIs()).GetCountries());
            ObjPD.BindDropDowns(ddl_GroupId, (new
            PopDlIs()).GetGroups(Convert.ToInt32(Clsusr.UserId), Convert.ToInt32(Clsusr.GroupId)));

            Address.Visible = false;

            string ProductId = Page.Request.QueryString["Id"];
            UserId =Convert.ToInt16( Clsusr.UserId);
            if (UserId > 0)
            {
                //-----user infor
                EtblUser lst = (new tblUsers()).getuserbyid(UserId);
                txt_username.Text = lst.UserId.ToString();
                txt_username.Text = lst.UserName;
            }
        }
    }
}

```

```

txt_FirstName.Text = lst.FirstName;
txt_LastName.Text = lst.LastName;
email_name.Text = lst.EmailID;
txt_mobile.Text = lst.ContactNum;
ddl_country.SelectedIndex = Convert.ToInt16( lst.Region);
txt_mobile.Text = lst.ContactNum;
ddl_GroupId.SelectedIndex = lst.GroupId.ToString();
txt_Password.Text = new BalCommon().EncryptPwd(lst.Password.ToString());
txt_Password1.Text = new BalCommon().EncryptPwd(lst.Password.ToString());

//-----user infor ends
ddl_users.SelectedValue = UserId.ToString();
ddl_users.Enabled = false;
Etetbluser_profile Prd = new tblUsers().getuserprofilebyid(UserId);
if (Prd != null && Prd.userid>0)
{
    Details.Text = Prd.Company_Description;
    txt_BusinessType.Text = Prd.Business_Type;
    ddl_uom.SelectedIndex = (int)Prd.UOMId;
    ddl_legalstr.SelectedValue = Prd.Legal_Structure;
    ddl_currency.SelectedIndex = (int)Prd.CID;
    txt_yrest.Text = Prd.Year_Established ;
    txt_noempl.Text = Prd.Number_of_Employees;
    txt_turnover.Text = Prd.Turnover_per_Year;
    ddl_currency.SelectedIndex = Prd.CID;
    txt_qa.Text = Prd.Quality_Assurance_Accredition;
    txt_regno.Text = Prd.Registration_Number;
    txt_dunsno.Text = Prd.D_U_N_S_Number;
    txt_special.Text = Prd.Specializing_in;
    txt_prodcap.Text = Prd.Monthly_Production_Capacity;
    txt_website.Text = Prd.Company_Website_URL;
    if (Prd.Company_logo != null)
    {
        string conn =
ConfigurationManager.ConnectionStrings["DowConectingString"].ConnectionString;
        SqlConnection connection = new SqlConnection(conn);
        string sql = "SELECT
name='~\\images\\FILE;' + convert(varchar(5),userid) + '.jpg', ContentType='IMAGE/JPEG',
Images=company_logo FROM tbluser_profile WHERE UserId = @PID";
        SqlCommand cmd = new SqlCommand(sql, connection);
        cmd.CommandType = CommandType.Text;
        cmd.Parameters.AddWithValue("@PID", UserId);

        connection.Open();
        SqlDataReader sqlDataReader = cmd.ExecuteReader();
        int sno = 0;
        Image1.Visible = false;
        List<ImageItem> il = new List<ImageItem>();
        while (sqlDataReader.Read())
        {
            sno++;
            byte[] bytes = (byte[])sqlDataReader["Images"];
            string base64String = Convert.ToBase64String(bytes, 0, bytes.Length);
            ImageItem wimg = new ImageItem();
            Image1.Visible = true;

```

```

        wimg.ImageUrl = "data:image/png;base64," + base64String;
        il.Add(new ImageItem { Key = sno.ToString(), ImageUrl =
"data:image/png;base64," + base64String, ToolTip = "test image" });

        Image1.ImageUrl = "data:image/png;base64," + base64String;
        Image1.Visible = true;

    }

    connection.Close();
    // sda.Dispose();
    connection.Dispose();
}
}
//
}
//
List<Etetbluser_address> Lst = (new tblUsers()).getusersaddressbyuserid(UserId, "");

if (Lst != null && Lst.Count > 0)
{
    WDGrid_UserAddress.DataSource = Lst;
    WDGrid_UserAddress.DataBind();
    WDGrid_UserAddress.Visible = true;
}
else
{
    WDGrid_UserAddress.Visible = false;
}
DataTable dt = new tblUsers().GetUserImagesDTbyUid(UserId, "");
//download(dt);
List<EtetblUserImages> Pics = (new tblUsers()).getuserImagesbyuserid(UserId, "");
foreach (var pic in Pics)
{
    ImageItem img = new ImageItem(pic.ImageUrl, "", "");
    WebImageViewer1.Items.Add(img);

    switch (pic.srno){
        case 1:
            Image2.ImageUrl=pic.ImageUrl;
            Image2.Visible = true;

            break;
        case 2:
            Image3.ImageUrl = pic.ImageUrl;
            Image3.Visible = true;
            break;
        case 3:
            Image4.ImageUrl = pic.ImageUrl;
            Image4.Visible = true;
            break;
        case 4:

```

```

        Image5.ImageUrl = pic.ImageUrl;
        Image5.Visible = true;
    break;
    default:
        break;
    }
}

WebImageViewer1.Visible = true;
if (Pics != null && Pics.Count > 0)
{
    WebDataPictures.DataSource = Pics;
    WebDataPictures.DataBind();
    WebDataPictures.Visible = true;
}
else
{
    WebDataPictures.Visible = false;
}
List<EteTblUser_emails> Lst1 = (new tblUsers()).getuserEmailsbyuserid(UserId, "");
List<EteTblUser_emails> Rcvd = (from a in Lst1 where a.EMRcvdUserId == UserId
select a).ToList();
List<EteTblUser_emails> Sent = (from a in Lst1 where a.EMSentUserId == UserId
select a).ToList();
if (Rcvd != null && Rcvd.Count > 0)
{
    WebDataEmailRcvd.DataSource = Rcvd;
    WebDataEmailRcvd.DataBind();
    WebDataEmailRcvd.Visible = true;
}
else
{
    WebDataEmailRcvd.Visible = false;
}

if (Sent != null && Sent.Count > 0)
{
    WebDataEmailSend.DataSource = Sent;
    WebDataEmailSend.DataBind();
    WebDataEmailSend.Visible = true;
}
else
{
    WebDataEmailSend.Visible = false;
}
}
}
private DataTable GetData(SqlCommand cmd)
{
    DataTable dt = new DataTable();
    String strConnString =
System.Configuration.ConfigurationManager.ConnectionStrings["DowConectingString"].Con
nectionString;
    SqlConnection con = new SqlConnection(strConnString);

```



```

SqlDataAdapter sda = new SqlDataAdapter();
cmd.CommandType = CommandType.Text;
cmd.Connection = con;
try
{
    con.Open();
    sda.SelectCommand = cmd;
    sda.Fill(dt);
    return dt;
}
catch
{
    return null;
}
finally
{
    con.Close();
    sda.Dispose();
    con.Dispose();
}
}

```

```

private void ManageNavigations(int pageid, int userid, int groupid, string PageType)
{
    bool IsLeftBar = false;
    IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
myBtnClickHandler);

    if (IsLeftBar == true)
    {
        LeftSideBar.Attributes["class"] = "span2";
        MainContent.Attributes["class"] = "span10";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;"); //style="display:
block;"
        PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
    else
    {
        LeftSideBar.Attributes["class"] = "";
        MainContent.Attributes["class"] = "span12";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
        PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
}

```

// This is Event Handler

```

public byte[] GetPhtoStream()
{

```

```
byte[] bufferPhoto = new byte[FileUpload1.PostedFile.ContentLength];
Stream photoStream = FileUpload1.PostedFile.InputStream;
photoStream.Read(bufferPhoto, 0, FileUpload1.PostedFile.ContentLength);
```

```
    return bufferPhoto;
}
protected void SaveAdd_Click(object sender, EventArgs e)
{
    Address.Visible = false;
}
}
```

```
protected void myBtnClickHandler(object sender, EventArgs e)
{
```

```
    LinkButton lnk = (LinkButton)sender;
    string ID = lnk.ID;
```

```
    DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
    if (ds != null)
    {
        if (ds.Tables[0].Rows.Count > 0)
        {
            DataTable dtdll = ds.Tables[0];
            foreach (DataRow drnew in dtdll.Rows)
            {
                switch (drnew["lbtncode"].ToString())
                {
                    case "SAVE":
                        // Call SAVE Method
                        ContentSaveControlClicked();
                        break;
                    case "AUDIT":
                        Response.Redirect(drnew["pageurl"].ToString() + "&frompage=" +
Request.QueryString["pageid"]);
                        break;
                    case "URL":
                        Response.Redirect(drnew["pageurl"].ToString());
                        break;
                    case "CANCEL":
                        ContentCancelControlClicked();
                        break;
                    case "ADDADDRESS":
                        Address.Visible = true;

                        break;
                    case "UPLOAD":
                        upload_image();
                        break;
                    default:
                        break;
                }
            }
        }
    }
}
```

```
    }  
  }  
}  
}
```

```
private void ContentCancelControlClicked()
```

```
{  
    Common.ShowMessageHtml("", 0, false, dvMsgBox);
```

```
    PageType = Session["PageType"].ToString();
```

```
    if (PageType == "Add")
```

```
    {  
        Response.Redirect("~/home.aspx?pageid=1");
```

```
    }  
    else if (PageType == "Edit")
```

```
    {  
        Response.Redirect("~/home.aspx?pageid=1");
```

```
    }  
}  
private void ContentSaveControlClicked()
```

```
{  
  
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];  
    UserId = Convert.ToInt32(Clsusr.UserId);
```

```
    try  
    {  
        if (Address.Visible == true)  
        {  
            //Save Address first check data Existing  
            if (ddl_addtype.SelectedIndex == 0) {  
                Common.ShowMessageHtml("Address Type cannot be blank", 0, true,
```

```
dvMsgBox);
```

```
            }  
            else if (txt_add1.Text == "")
```

```
            {  
                Common.ShowMessageHtml("Address cannot be blank", 0, true, dvMsgBox);
```

```
            }  
            else if (txt_city.Text == "")  
            {  
                Common.ShowMessageHtml("City cannot be blank", 0, true, dvMsgBox);
```

```
            }  
            else if (txt_pincode.Text == "")  
            {  
                Common.ShowMessageHtml("Pincode cannot be blank", 0, true, dvMsgBox);
```

```
            }  
            else if (txt_contactno.Text == "")  
            {
```

```

        Common.ShowMessageHtml("Contact No cannot be blank", 0, true,
dvMsgBox);
    }
    else
    {
        Etetbluser_address et = new Etetbluser_address();
        et.address_type = ddl_addtype.SelectedIndex;
        et.address1 = txt_add1.Text;
        et.address3 = txt_add3.Text;
        et.address2 = txt_add2.Text;
        et.city = txt_city.Text;
        et.country = 1;
        et.countryname = "India";
        et.pincode = txt_pincode.Text;
        et.ContactNo = txt_contactno.Text;
        bool ret =new tblUsers().InsertAddress(et);
        if (ret)
        {
            Common.ShowMessageHtml("Address Updated", 0, true, dvMsgBox);
            List<Etetbluser_address> Lst = (new
tblUsers()).getusersaddressbyuserid(UserId, "");

            if (Lst != null && Lst.Count > 0)
            {
                WDGrid_UserAddress.DataSource = Lst;
                WDGrid_UserAddress.DataBind();
                WDGrid_UserAddress.Visible = true;
            }
            else
            {
                WDGrid_UserAddress.Visible = false;
            }
        }
    }
}
else if (Details.Text == "")
{
    Common.ShowMessageHtml("Company Description cannot be blank", 0, true,
dvMsgBox);

}

else
{
    Etetbluser_profile Prd = new Etetbluser_profile();
    Prd.userid = Convert.ToInt32(Request.QueryString["Id"]);

    bool Result = false;
    PageType = Session["PageType"].ToString();

    Prd.userid = Convert.ToInt32(Request.QueryString["ProductId"]);
    Prd.Company_Description = Details.Text;

```

```

Prd.Business_Type = txt_BusinessType.Text;
Prd.UOMId = ddl_uom.SelectedIndex;
Prd.Legal_Structure = ddl_legalstr.SelectedValue;
Prd.CID = ddl_currency.SelectedIndex;
Prd.Year_Established = txt_yrest.Text;
Prd.Number_of_Employees = txt_noempl.Text;
Prd.Turnover_per_Year = txt_turnover.Text;
Prd.CID = ddl_currency.SelectedIndex;
Prd.Quality_Assurance_Accreditation = txt_qa.Text;
Prd.Registration_Number = txt_regno.Text;
Prd.D_U_N_S_Number = txt_dunsno.Text;
Prd.Specializing_in = txt_special.Text;
Prd.Monthly_Production_Capacity = txt_prodcap.Text;
Prd.Company_Website_URL = txt_website.Text;
Prd.UpdatedOn = DateTime.Now;
Prd.CreatedOn = DateTime.Now;
Prd.userid = Convert.ToInt16(Clsusr.UserId);
//Prd.Company_logo = Photo;
if (PageType == "Add")
{
    Result = new tblUsers().InsertProfile(Prd);
}
else
{
    Result = new tblUsers().UpdateProfile(Prd);
}

if (Result == true)
{
    if (PageType == "Add")
    {
        Common.ShowMessageHtml("Adding of Profile Sucessful ", 1, true,
dvMsgBox);
    }
    else
    {
        Common.ShowMessageHtml("Profile Edited Sucessfully ", 1, true,
dvMsgBox);
    }
    Response.Redirect("~/Home.aspx?pageid=1&msg=Product Updated
Sucessfully");
}
}
}

catch (Exception exc)
{
    objlog.Trace_Error(4, "DateOrWed", "Profile", "Load", "Exception:" + exc.ToString());
    Common.ShowMessageHtml(exc.ToString(), 0, true, dvMsgBox);
}
}

```

```

        //objlog.Trace_Log(0, "DateOrWed", "MangeDealer", "SaveClick", "Start Here..");
    }

    protected void btn_SaveAddress_Click(object sender, System.EventArgs e)
    {
        //save code
        Address.Visible = false;
    }

    protected void Button2_Click(object sender, System.EventArgs e)
    {
        Address.Visible = false;
    }

    protected void btn_saveimage_Click(object sender, System.EventArgs e)
    {
        string path = FileUpload1.FileName;
        byte[] Photo = GetPhtoStream();
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        UserId = Convert.ToInt16(Clsusr.UserId);
        EtetblUserImages ep = new EtetblUserImages();

        ep.UserId = UserId;

        ep.Images = Photo;
        bool ret = new tblUsers().InsertImages(ep);

        List<EtetblUserImages> Pics = (new tblUsers()).getuserImagesbyuserid(UserId, "");
        foreach (var pic in Pics)
        {
            switch (pic.srno)
            {
                case 1:
                    Image2.ImageUrl = pic.ImageUrl;
                    Image2.Visible = true;
                    break;
                case 2:
                    Image3.ImageUrl = pic.ImageUrl;
                    Image3.Visible = true;
                    break;
                case 3:
                    Image4.ImageUrl = pic.ImageUrl;
                    Image4.Visible = true;
                    break;
                case 4:
                    Image5.ImageUrl = pic.ImageUrl;
                    Image5.Visible = true;
                    break;
                default:
                    break;
            }
        }
        //if (Pics != null && Pics.Count > 0)
        //{
        //    WebDataPictures.DataSource = Pics;
        //    WebDataPictures.DataBind();
        //}
    }

```

```

// WebDataPictures.Visible = true;
//}
//else
//{{
// WebDataPictures.Visible = false;
//}

}
protected void upload_image()
{
    string path = FileUpload1.FileName;
    byte[] Photo = GetPhtoStream();
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    UserId = Convert.ToInt16(Clsusr.UserId);
    EtetblUserImages ep = new EtetblUserImages();

    ep.UserId = UserId;

    ep.Images = Photo;
    bool ret = new tblUsers().InsertImages(ep);

    List<EtetblUserImages> Pics = (new tblUsers()).getuserImagesbyuserid(UserId, "");
    foreach (var pic in Pics)
    {
        switch (pic.srno)
        {
            case 1:
                Image2.ImageUrl = pic.ImageUrl;
                Image2.Visible = true;
                break;
            case 2:
                Image3.ImageUrl = pic.ImageUrl;
                Image3.Visible = true;
                break;
            case 3:
                Image4.ImageUrl = pic.ImageUrl;
                Image4.Visible = true;
                break;
            case 4:
                Image5.ImageUrl = pic.ImageUrl;
                Image5.Visible = true;
                break;
            default:
                break;
        }
    }
    //if (Pics != null && Pics.Count > 0)
    //{
    // WebDataPictures.DataSource = Pics;
    // WebDataPictures.DataBind();
    // WebDataPictures.Visible = true;
    //}
    //else
    //{
    // WebDataPictures.Visible = false;

```

```

    //}

}
private void download(DataTable dt)
{
    foreach (DataRow row in dt.Rows)
    {
//      Byte[] bytes = (Byte[])dt.Rows[0]["Data"];
        Byte[] bytes = (Byte[])row["Images"];
        Response.Buffer = true;
        Response.Charset = "";
        Response.Cache.SetCacheability(HttpCacheability.NoCache);
        Response.ContentType = row["ContentType"].ToString();
//      Response.ContentType = dt.Rows[0]["ContentType"].ToString();
//      Response.AddHeader("content-disposition", "attachment;filename=" +
dt.Rows[0]["Name"].ToString());
//      var path = Server.MapPath(row["Name"].ToString());
        string path = Server.MapPath("");
        path = path + @row["Name"].ToString();
        Response.AddHeader("content-disposition", "attachment;filename=" + path);

        Response.BinaryWrite(bytes);
        Response.TransmitFile(path);
//Response.WriteFile(path);
        Response.Flush();
        Response.End();
    }
}

}
***** ManageProducts.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="ManageProducts.aspx.cs" Inherits="ManageProducts"
%>

<%@ Register Assembly="Infragistics4.Web.v14.2, Version=14.2.20142.1028,
Culture=neutral, PublicKeyToken=7dd5c3163f2cd0cb"
Namespace="Infragistics.Web.UI.ListControls" TagPrefix="ig" %>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
    <script type="text/javascript">
        function checkTextAreaMaxLength(textBox, e, length) {

            var mLen = textBox["MaxLength"];
            if (null == mLen)
                mLen = length;

            var maxLength = parseInt(mLen);
            if (!checkSpecialKeys(e)) {
                if (textBox.value.length > maxLength - 1) {
                    if (window.event)//IE
                        e.returnValue = false;
                }
            }
        }
    </script>

```



```

        else
            e.preventDefault();
    }
}
}

```

```

function isNumberKey(evt) {
    var charCode = (evt.which) ? evt.which : event.keyCode;
    if (charCode != 46 && charCode > 31 && (charCode < 48 || charCode > 57)) {
        return false;
    } else {
        // If the number field already has . then don't allow to enter . again.
        if (evt.target.value.search(/\./) > -1 && charCode == 46) {
            return false;
        }
        return true;
    }
}

```

```

</script>

```

```

<style type="text/css">

```

```

    .auto-style1 {
        width: 220px;
        height: 48px;
    }

```

```

    .auto-style2 {
        height: 48px;
    }

```

```

</style>

```

```

</asp:Content>

```

```

<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" runat="Server">

```

```

    <div class="container-fluid">

```

```

        <div class="row-fluid" style="padding-top: 20px;">

```

```

            <div class="span2" id="dvlogo" runat="server" style="display: block;">

```

```

                <%----%>

```

```

            </div>

```

```

            <div class="span10">

```

```

                <div class="row">

```

```

                    <div id="dvMsgBox" class="span12 ordersave" runat="server" style="display:
none;">

```

```

                        <asp:Label ID="Label1" runat="server" Text=""></asp:Label>

```

```

                    </div>

```

```

                </div>

```

```

            </div>

```

```

        </div>

```

```

    </asp:Content>

```

```

<asp:Content ID="Content4" ContentPlaceHolderID="CPH_LeftSideBar" runat="Server">

```

```

    <div id="LeftSideBar" runat="server" class="span2">

```

```

        <asp:Table ID="Table1" runat="server">

```

```

            </asp:Table>

```

```

    </div>

```

```

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="CPH_MainContent" runat="Server">

    <div class="span10" id="MainContent" runat="server">
        <div id="ShowPageHeader" class="row" style="display: block;" runat="server">
            <div id="PageName" class="span12 customer" runat="server">
                </div>
            </div>

            <div id="Div1" class="row" style="display: block;" >
                <div id="Div2" class="span12 customertable">

                    <div class="tabs_holder tabs_wrapper skin7 top" style="display: block; overflow:
hidden; position: relative;">

                        <div class="content_holder">
                            <div style="overflow: hidden; position: relative; z-index: 1; padding: 10px
0px;" class="content_holder_inner">
                                <div id="your-tab-id-1" class="tab_content" style="display: block;">
                                    <div class="inner_content">
                                        <div style="background-color: #f8f9fa;" class="span12
customertable">

                                            <div class="row-fluid">
                                                <div style="background-color: #ffffff; border: 1px solid #eefef;
border-top: none; padding: 15px 5px;" class="span12 detailview">
                                                    <div class="control-group">
                                                        <table id="Table5" class="table table-striped table-detail
span12" runat="server">

                                                            <tbody>

                                                                <tr>
                                                                    <td class="span3" dir="auto">Group Name :</td>
                                                                    <td>
                                                                        <asp:DropDownList ID="ddl_productgroup" runat="server"></asp:DropDownList>
                                                                    </td>
                                                                </tr>
                                                                <tr>
                                                                    <td class="span3" dir="auto">Product Description :</td>
                                                                    <td>
                                                                        <asp:TextBox ID="Details" runat="server" Width="600px" Height="100px"
Rows="7" style="resize:none" TextMode="MultiLine" MaxLength="250"></asp:TextBox>
                                                                    </td>
                                                                </tr>

                                                                <tr>
                                                                    <td class="auto-style1" dir="auto">Seller</td>
                                                                    <td class="auto-style2">
                                                                        <asp:DropDownList ID="ddl_seller" runat="server"></asp:DropDownList>
                                                                    </td>
                                                                </tr>
                                                            </tbody>
                                                        </table>
                                                    </div>
                                                </div>
                                            </div>
                                        </div>
                                    </div>
                                </div>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>

```

```

        </tr>
        <tr>
            <td class="auto-style1" dir="auto">Price:</td>
            <td>
                <asp:TextBox ID="txt_tasksummary" runat="server" MaxLength="40" Width="100px"
                ></asp:TextBox>

                <asp:Label ID="Label2" runat="server" Text="Currency: "
                Width="80px"></asp:Label>

                <asp:DropDownList ID="ddl_currency" runat="server"
                Width="100px"></asp:DropDownList>

                <asp:Label ID="Label5" runat="server" Text=" Packing : "
                Width="80px"></asp:Label>

                <asp:DropDownList ID="ddl_packing" runat="server"
                Width="100px"></asp:DropDownList>

            </td>
        </tr>
        <tr>
            <td class="span3" dir="auto">Keywords</td>
            <td>
                <asp:TextBox ID="txt_taskdetails" runat="server" Width="600px"
                MaxLength="40" ></asp:TextBox>
            </td>
        </tr>
        <tr>
            <td class="span3" dir="auto">Min Order:</td>
            <td>
                <asp:TextBox ID="txt_minorder" runat="server" MaxLength="40"
                Width="250px">1</asp:TextBox>

                <asp:Label ID="Label3" runat="server" Text=" Lead Time: "></asp:Label>
                <asp:TextBox ID="txt_LeadTime" runat="server" MaxLength="40"
                Width="250px">1</asp:TextBox>

            </td>
        </tr>
    </tbody>
</table>
<div class="u-1-2" runat="server">
<asp:Label>Select an Option :</asp:Label>
    <asp:DropDownList ID="ddl_optlist"
    OnSelectedIndexChanged="ddl_optlist_SelectedIndexChanged" runat="server">

        </asp:DropDownList>
    <div id="ExtOpt" class="nav-list" runat="server" >

        </div>
</div>
</table>

```

```

<tbody>
<tr>
<td class="span3" dir="auto">Pictures</td>
<td>

</td>
</tr>
<tr>

<td > </td><td>
<asp:FileUpload ID="FileUpload1" runat="server" />

<asp:Button ID="btnupload" runat="server" OnClick="btnupload_Click" Text="Upload"
Height="22px" Width="75px" />&nbsp;
</td>
</tr>

<tr>
<td>
<asp:image ToolTip = "ASP Image Control" ID="Image1" runat="server" ImageUrl
="ManageProducts.aspx?ImageID=1" Height="156px" Width="174px"
Visible="false"></asp:image> </td>
<td> <asp:image ToolTip = "ASP Image Control" ID="Image2" runat="server" ImageUrl
="ManageProducts.aspx?ImageID=2" Height="156px" Width="174px"
Visible="false"></asp:image>
<asp:image ToolTip = "ASP Image Control" ID="Image3" runat="server" ImageUrl
="ManageProducts.aspx?ImageID=3" Height="156px" Width="174px" Visible="false">
</asp:image>
<asp:image ToolTip = "ASP Image Control" ID="Image4" runat="server" ImageUrl
="ManageProducts.aspx?ImageID=4" Height="156px" Width="174px" Visible="false">
</asp:image>

</td>

</tr>
</tbody>
</table>

<asp:SqlDataSource ID="SqlDataSource1" runat="server"
ConnectionString="<%= $ ConnectionStrings:DowConectingString %>"
SelectCommand="SELECT * FROM [productSellerImages]"></asp:SqlDataSource>
</div>
</div>
<!-- /.content_holder -->
<div style="height: 0px; font-size: 1px; line-height: 1px; clear:
both;"></div>
</div>
</div>

```

```

        </div>
    </div>
</div>
</div>

</div>
</div>
</div>
</asp:Content>
***** ManageProducts.aspx.cs *****
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.Security;
using DowBAL;
using System.Data;

using System.Net;
using DowBEntity;
using System.Data.SqlClient;
using System.Configuration;

using System.Xml;
using System.Xml.Serialization;
using System.IO;

public partial class ManageProducts : System.Web.UI.Page
{
    public string PageType;
    string pageid;
    int Id;
    DDLogTracer objlog = new DDLogTracer();

    protected void Page_Load(object sender, EventArgs e)
    {
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr != null)
        {
            PageType = Page.Request.QueryString["PageType"];
            if (PageType == null)
            {
                PageType = "Add";
            }
            Session["PageType"] = PageType;
            pageid = "4";// Request.QueryString["pageid"];
            ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
            Convert.ToInt32(Clsusr.GroupId), PageType);
        }
        int PId=Convert.ToInt16( Request.QueryString["ProductId"]);
    }
}

```

```

if (!IsPostBack)
{
    PopDlIs ObjPD = new PopDlIs();
    ObjPD.BindDropDowns(ddl_productgroup, (new PopDlIs()).GetproductGroups());
    ObjPD.BindDropDowns(ddl_seller, (new PopDlIs()).GetUserCompany());
    ObjPD.BindDropDowns(ddl_currency, (new PopDlIs()).GetCurrency());
    ObjPD.BindDropDowns(ddl_optlist, (new PopDlIs()).GetExtProp());
    if (PageType=="Add")
    {
        ddl_productgroup.SelectedValue = Request.QueryString["ProductId"];
    }
}

if (PageType == "Edit")
{
    Id = Convert.ToInt32(Request.QueryString["ProductId"]);
    Eteproduct Prd = new Products().GetProduct(Id, "manageproduct");

    if (Prd != null)
    {
        Details.Text = Prd.ProductName;
        ddl_productgroup.SelectedIndex =(int) Prd.ProductGroupId;
        ddl_seller.SelectedIndex =(int) Prd.ProductGroupCategory;
        txt_tasksummary.Text = Prd.Price.ToString();
        ddl_seller.SelectedIndex = Prd.SellerId;
        int CurrencyId = (int)Prd.CurrencyId;
        txt_taskdetails.Text = Prd.AnalysisCode;
        //CreatedOn = Prd.CreatedOn;
        //UpdatedOn = Prd.UpdatedOn;
        //UserId = Prd.UserId;
        //PrId = Prd.PrId;
        //MinOrder = (int)Prd.MinOrder;
        //DelTlme = (int)Prd.DelTlme;
        //List<EteproductImages> pics = new Products().GetProductSellerImage(Id,
"manageproduct");
        //List<ImageItem> ll = new List<ImageItem>();
        int SId = ddl_seller.SelectedIndex;
        string conn =
ConfigurationManager.ConnectionStrings["DowConectingString"].ConnectionString;
        SqlConnection connection = new SqlConnection(conn);
        string sql = "SELECT
name='~\\images\\FILE'+convert(varchar(5),productid)+' .jpg',ContentType='IMAGE/JPEG',
Images FROM productSellerImages WHERE ProductId = @PID and Sellerid=@SID";
        SqlCommand cmd = new SqlCommand(sql, connection);
        cmd.CommandType = CommandType.Text;
        cmd.Parameters.AddWithValue("@PID", Id);
        cmd.Parameters.AddWithValue("@SID", SId);
        SqlDataAdapter sda = new SqlDataAdapter();
        sda.SelectCommand = cmd;
        connection.Open();
        DataTable img = new DataTable();
        sda.SelectCommand = cmd;
        sda.Fill(img);
        SqlDataReader sqlDataReader = cmd.ExecuteReader();
        int sno = 0;

```

```

while (sqlDataReader.Read())
{
    sno++;
    byte[] bytes = (byte[])sqlDataReader["Images"];
    string base64String = Convert.ToBase64String(bytes, 0, bytes.Length);

    if (sno == 1) {
        Image1.ImageUrl = "data:image/png;base64," + base64String;
        Image1.Visible = true;
    }
    if (sno == 3)
    {
        Image3.ImageUrl = "data:image/png;base64," + base64String;
        Image3.Visible = true;
    }
    if (sno == 2)
    {
        Image2.ImageUrl = "data:image/png;base64," + base64String;
        Image2.Visible = true;
    }
    if (sno == 4)
    {
        Image4.ImageUrl = "data:image/png;base64," + base64String;
        Image4.Visible = true;
    }
}

```

```

connection.Close();
sda.Dispose();
connection.Dispose();

```

```

}

```

```

}

```

```

}

```

```

}

```

```

private DataTable GetData(SqlCommand cmd)

```

```

{

```

```

    DataTable dt = new DataTable();

```

```

    String strConnString =

```

```

System.Configuration.ConfigurationManager.ConnectionStrings["DowConectingString"].Con
nectionString;

```

```

    SqlConnection con = new SqlConnection(strConnString);

```

```

    SqlDataAdapter sda = new SqlDataAdapter();

```

```

    cmd.CommandType = CommandType.Text;

```

```

    cmd.Connection = con;

```

```

    try

```

```

    {

```

```

        con.Open();
        sda.SelectCommand = cmd;
        sda.Fill(dt);
        return dt;
    }
    catch
    {
        return null;
    }
    finally
    {
        con.Close();
        sda.Dispose();
        con.Dispose();
    }
}

```

```

private void ManageNavigations(int pageid, int userid, int groupid, string PageType)
{
    bool IsLeftBar = false;
    IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
myBtnClickHandler);

    if (IsLeftBar == true)
    {
        LeftSideBar.Attributes["class"] = "span2";
        MainContent.Attributes["class"] = "span10";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;"); //style="display:
block;"
        PageName.InnerText = "Manage Products";// new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
    else
    {
        LeftSideBar.Attributes["class"] = "";
        MainContent.Attributes["class"] = "span12";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
        PageName.InnerText = "Manage Products";// new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
}

```

// This is Event Handler

```

public byte[] GetPhtoStream()
{
    byte[] bufferPhoto = new byte[FileUpload1.PostedFile.ContentLength];
    Stream photoStream = FileUpload1.PostedFile.InputStream;
    photoStream.Read(bufferPhoto, 0, FileUpload1.PostedFile.ContentLength);

    return bufferPhoto;
}

```



```

}
protected void btnupload_Click(object sender, EventArgs e)
{
    string path = FileUpload1.FileName;
    byte[] Photo = GetPhtoStream();
    Id = Convert.ToInt32(Request.QueryString["ProductId"]);

    EteproductImages ep = new EteproductImages();

    ep.ProductId = Id;
    ep.SellerId = ddl_seller.SelectedIndex;
    ep.srno = 1;
    ep.Images = Photo;
    string Photold = new Products().InsertPhoto(ep);
}

protected void myBtnClickHandler(object sender, EventArgs e)
{

    LinkButton lnk = (LinkButton)sender;
    string ID = lnk.ID;

    DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
    if (ds != null)
    {
        if (ds.Tables[0].Rows.Count > 0)
        {
            DataTable dtdll = ds.Tables[0];
            foreach (DataRow drnew in dtdll.Rows)
            {
                switch (drnew["lbtncode"].ToString())
                {
                    case "SAVE":
                        // Call SAVE Method
                        ContentSaveControlClicked();
                        break;
                    case "AUDIT":
                        Response.Redirect(drnew["pageurl"].ToString() + "&frompage=" +
Request.QueryString["pageid"]);
                        break;
                    case "URL":
                        Response.Redirect(drnew["pageurl"].ToString());
                        break;
                    case "CANCEL":
                        ContentCancelControlClicked();
                        break;
                    default:
                        break;
                }
            }
        }
    }
}

```

```

    }
}

private void ContentCancelControlClicked()
{
    Common.ShowMessageHtml("", 0, false, dvMsgBox);

    PageType = Session["PageType"].ToString();
    if (PageType == "Add")
    {
        Response.Redirect("~/home.aspx?pageid=1");
    }
    else if (PageType == "Edit")
    {
        Response.Redirect("~/home.aspx?pageid=1");
    }
}
private void ContentSaveControlClicked()
{
    //objlog.Trace_Log(0, "DateOrWed", "MangeDealer", "SaveClick", "Start Here..");

    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    try
    {
        if (Details.Text == "")
        {
            Common.ShowMessageHtml("Product Description cannot be blank", 0, true,
dvMsgBox);

        }
        else if (txt_taskdetails.Text == "")
        {

            Common.ShowMessageHtml("Keywords cannot be blank", 0, true, dvMsgBox);

        }
        else
        {
            Eteproduct Prd = new Eteproduct();
            Prd.ProductId = Convert.ToInt32(Request.QueryString["Id"]);
            Prd.CreatedOn = DateTime.Now;
            Prd.UserId = Convert.ToInt32(Clsusr.UserId);
            Prd.ProductName = Details.Text.ToString() ;
            Prd.ProductGroupId = ddl_productgroup.SelectedIndex
;
            Prd.ProductGroupCategory = ddl_seller.SelectedIndex
;
            Prd.Price = Convert.ToDecimal(txt_tasksummary.Text) ;
            Prd.SellerId = ddl_seller.SelectedIndex ;
            Prd.CurrencyId = ddl_currency.SelectedIndex ;
        }
    }
}

```

```

Prd.AnalysisCode = txt_taskdetails.Text.ToString();
Prd.PrId = ddl_optlist.SelectedValue;
    bool Result = false;
    PageType = Session["PageType"].ToString();

    if (PageType == "Add")
    {
        Result = new Products().Insert(Prd);
    }
    else
    {
        Prd.ProductId = Convert.ToInt32(Request.QueryString["ProductId"]);

        Result = new Products().Update(Prd);
    }
    if (Result == true)
    {
        if (PageType == "Add")
        {
            Common.ShowMessageHtml("Adding of Product Successful ", 1, true,
dvMsgBox);
        }
        else
        {
            Common.ShowMessageHtml("Ceancel Reason Edited Sucessfully ", 1, true,
dvMsgBox);
        }
        Response.Redirect("~/ProductsList.aspx?pageid=29&msg=Product Updated
Sucessfully");
    }
}
}

catch (Exception exc)
{
    objlog.Trace_Error(4, "DateOrWed", "MangeDealer", "Load", "Exception:" +
exc.ToString());
    Common.ShowMessageHtml(exc.ToString(), 0, true, dvMsgBox);
}
//objlog.Trace_Log(0, "DateOrWed", "MangeDealer", "SaveClick", "Start Here..");
}
protected void Lnkbtn_PrintPDF_Click(object sender, EventArgs e)
{
    LinkButton lnkbutton = (LinkButton)sender;
    string strcustid = lnkbutton.CommandName;
    string filepath = "filepath";// Ist.WelcomePackFile;
    string fileName = Path.GetFileName(filepath);

    string FilePath = Server.MapPath("/PDFs") + "/" + fileName;

```

```

WebClient User = new WebClient();

Byte[] FileBuffer = User.DownloadData(FilePath);
if (FileBuffer != null)
{
    Response.Clear();
    Response.ContentType = "application/pdf";
    Response.AddHeader("content-length", FileBuffer.Length.ToString());
    Response.BinaryWrite(FileBuffer);
    Response.Flush();
    Response.Close();
}
}

//public class Handler : IHttpHandler
//{

//    public void ProcessRequest(HttpContext context)
//    {
//        string Id = context.Request.QueryString["ProductId"];
//        string conn =
ConfigurationManager.ConnectionStrings["DowConectingString"].ConnectionString;
//        SqlConnection connection = new SqlConnection(conn);
//        string sql = "SELECT Images FROM productSellerImages WHERE ProductId =
@PID";
//        SqlCommand cmd = new SqlCommand(sql, connection);
//        cmd.CommandType = CommandType.Text;
//        cmd.Parameters.AddWithValue("@PID", Id);
//        SqlDataReader sda = cmd.ExecuteReader();
//        sda.Read();
//        context.Response.BinaryWrite((Byte[])sda[0]);
//        context.Response.End();

//    }

//}

protected void ddl_optlist_SelectedIndexChanged(object sender, System.EventArgs e)
{
}
}
***** ManageProductsource.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="ManageProductsource.aspx.cs"
Inherits="ManageProductsource" %>

<!--<%@ Register Assembly="AjaxControlToolkit" Namespace="AjaxControlToolkit"
TagPrefix="cc1" %>-->

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">

```

```
<meta http-equiv="content-type" content="text/html; charset=UTF-8" />
<meta charset="UTF-8" />
<meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1" />
<title> </title>
```

```
<meta name="viewport" content="width=device-width, height=device-height, initial-
scale=1, minimum-scale=1, maximum-scale=1, user-scalable=0" />
```

```
<script type="text/javascript"
src="http://ajax.googleapis.com/ajax/libs/jquery/1.4.2/jquery.min.js"></script>
```

```
<style>
* {
  box-sizing: border-box;
}

.img-zoom-container {
  position: relative;
}

.img-zoom-lens {
  position: absolute;
  border: 1px solid #d4d4d4;
  /*set the size of the lens:*/
  width: 40px;
  height: 40px;
}

.img-zoom-result {
  border: 1px solid #d4d4d4;
  /*set the size of the result div:*/
  width: 200px;
  height: 200px;
}
</style>
```

```
<script>
function imageout() {
  //document.getElementById('myresult').style.display = "none";
  //document.getElementById('prodkart').style.display = "block";
}

function imageZoom(imgID, resultID) {
  document.getElementById(resultID).style.display = "block";
  //document.getElementById('prodkart').style.display = "none";

  var img, lens, result, cx, cy;
  img = document.getElementById(imgID);
  result = document.getElementById(resultID);
  /*create lens:*/
  lens = document.createElement("DIV");
  lens.setAttribute("class", "img-zoom-lens");
  /*insert lens:*/
  img.parentElement.insertBefore(lens, img);
  /*calculate the ratio between result DIV and lens:*/
```

```

cx = result.offsetWidth / lens.offsetWidth;
cy = result.offsetHeight / lens.offsetHeight;
/*set background properties for the result DIV:*/
result.style.backgroundImage = "url(" + img.src + ")";
result.style.backgroundSize = (img.width * cx) + "px " + (img.height * cy) + "px";
/*execute a function when someone moves the cursor over the image, or the lens:*/
lens.addEventListener("mousemove", moveLens);
img.addEventListener("mousemove", moveLens);
/*and also for touch screens:*/
lens.addEventListener("touchmove", moveLens);
img.addEventListener("touchmove", moveLens);
function moveLens(e) {
    var pos, x, y;
    /*prevent any other actions that may occur when moving over the image:*/
    e.preventDefault();
    /*get the cursor's x and y positions:*/
    pos = getCursorPos(e);
    /*calculate the position of the lens:*/
    x = pos.x - (lens.offsetWidth / 2);
    y = pos.y - (lens.offsetHeight / 2);
    /*prevent the lens from being positioned outside the image:*/
    if (x > img.width - lens.offsetWidth) { x = img.width - lens.offsetWidth; }
    if (x < 0) { x = 0; }
    if (y > img.height - lens.offsetHeight) { y = img.height - lens.offsetHeight; }
    if (y < 0) { y = 0; }
    /*set the position of the lens:*/
    lens.style.left = x + "px";
    lens.style.top = y + "px";
    /*display what the lens "sees":*/
    result.style.backgroundPosition = "-" + (x * cx) + "px -" + (y * cy) + "px";
}
function getCursorPos(e) {
    var a, x = 0, y = 0;
    e = e || window.event;
    /*get the x and y positions of the image:*/
    a = img.getBoundingClientRect();
    /*calculate the cursor's x and y coordinates, relative to the image:*/
    x = e.pageX - a.left;
    y = e.pageY - a.top;
    /*consider any page scrolling:*/
    x = x - window.pageXOffset;
    y = y - window.pageYOffset;
    return { x: x, y: y };
}
}
</script>

```

</asp:Content>

```

<asp:Content ID="Content2" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
<div class="container-fluid">
<div class="row-fluid" style="padding-top: 20px;">
<div class="span2" id="dvlogo" runat="server" style="display: block;">
<%----%>

```

```

</div>
<div class="span10">
  <div class="row">
    <div id="dvMsgBox" class="span12 ordersave" runat="server" style="display:
none;">
      <asp:Label ID="Label2" runat="server" Text="Message"></asp:Label>
    </div>
  </div>
</div>
</div>
</div>
</div>
</div>
</div>
</asp:Content>
<asp:Content ID="Content3" ContentPlaceHolderID="CPH_LeftSideBar" runat="Server" >
  <div id="LeftSideBar" runat="server" class="span2" style="display:block">
<%-- class="span2"--%>
  <asp:Table ID="Table1" runat="server" Width="250px">
    <asp:TableRow>
      <asp:TableCell CssClass="customer" Text="Zoomed Image">
        </asp:TableCell>
      </asp:TableRow>
    </asp:Table>
    <%-- zoom image below--%>
    <div id="myresult" class="img-zoom-result" style="display:block"></div>
  </div>
</asp:Content>

<asp:Content ID="Content4" ContentPlaceHolderID="CPH_MainContent" runat="Server">
  <div class="span10" id="MainContent" runat="server">
    <div id="ShowPageHeader" class="row" style="display: block;" runat="server">
      <div id="PageName" class="span12 customer" runat="server">
        </div>
      </div>
    </div>
    <div class="container-fluid product-slide" id="product-details-wrapper">
      <div class="row product-details">
        <div class="col-xs-12 col-sm-12 col-md-6 product-image">
          <%--
            <div class="product-carousel zoom-view carousel slick-initialized slick-
slider" data-fade="true" data-load-on-demand="true" data-arrows="true" style="height:
auto;">--%>
            <div id="test" runat="server">
<div id='gallery'>
  <div id='panel'><img id='largelImage' src='images/vendor/2-6-1.png' alt='1'
onmouseout='imageout()' style='height:400px;width:600px;'
onmouseover="imageZoom('largelImage', 'myresult')" />
</div>
<script>
  imageZoom('largelImage', 'myresult') </script>

```

```
<div id='thumbs'>
  <img src='images/vendor/2-6-1.png' class='thumb' alt='1' style='height:100px;width:80px;'/>
  <img src='images/Makeup.png' class='thumb' alt='2' style='height:100px;width:80px;'/>
  <img src='images/makeup2.png' class='thumb' alt='3' style='height:100px;width:80px;'/>
  <img src='images/Jewellery1.png' class='thumb' alt='4' style='height:100px;width:80px;'/>
  <img src='images/Mehendi.png' class='thumb' alt='5' style='height:100px;width:80px;'/>
</div>
<script>
  $('#thumbs').delegate('img', 'click', function () {
    $('#largelImage').attr('src', $(this).attr('src').replace('thumb', 'large'));
  });
</script>
</div>
```

```
</div>
<!-- </div>-->
</div>
<!-- product details price ect begins here-->
<div class="full-width col-xs-12 col-sm-12 col-md-5 product-info">
  <div class="info-block">
    <div class="content" id="prodkart" runat="server">

</div>
    <!--<--prodkart ends-->-->
    <!-- <--Sizes selection starts-->-->
    <div class="swatches clearfix">
      <div class="swatch size one-dropdown " id="size_block">
        <a href="javascript:void(0);" class="help dialogify" id="guidechartlink" data-
mobile-url="sizePopup" data-target="#size-guide" data-custom-class="size-guide-modal"
data-
chart="{&quot;measurementHeaders&quot;:[&quot;Bust&quot;],[&quot;Waist&quot;],[&quot;
Hips&quot;]],&quot;sizeHeaders&quot;:[&quot;Size&quot;]],&quot;sizes&quot;:{&quot;inches
Array&quot;:[[[&quot;S&quot;],[&quot;36.0&quot;],[&quot;32.0&quot;],[&quot;38.0&quot;],[&quot;
M&quot;],[&quot;38.0&quot;],[&quot;34.0&quot;],[&quot;40.0&quot;],[&quot;L&quot;],[&quot;40.
0&quot;],[&quot;36.0&quot;],[&quot;42.0&quot;],[&quot;XL&quot;],[&quot;42.0&quot;],[&quot;38.0
&quot;],[&quot;44.0&quot;],[&quot;XXL&quot;],[&quot;44.0&quot;],[&quot;40.0&quot;],[&quot;46.
0&quot;],[&quot;3XL&quot;],[&quot;48.0&quot;],[&quot;44.0&quot;],[&quot;50.0&quot;]],&quot;c
mArray&quot;:[[[&quot;S&quot;],[&quot;91.4&quot;],[&quot;81.3&quot;],[&quot;96.5&quot;],[&quot;
ot;M&quot;],[&quot;96.5&quot;],[&quot;86.4&quot;],[&quot;101.6&quot;],[&quot;L&quot;],[&quot;
101.6&quot;],[&quot;91.4&quot;],[&quot;106.7&quot;],[&quot;XL&quot;],[&quot;106.7&quot;],[&quot;
ot;96.5&quot;],[&quot;111.8&quot;],[&quot;XXL&quot;],[&quot;111.8&quot;],[&quot;101.6&quot;],[
&quot;116.8&quot;],[&quot;3XL&quot;],[&quot;121.9&quot;],[&quot;111.8&quot;],[&quot;127.0&
quot;]]],&quot;showSizeChart&quot;:true}" data-gaq-event="pdp~$~Size
guide~$~3714337">View Size Chart</a><span id="size-heading" class="hidden-
xs">Size</span>
      <div class="pdp-size-fit"></div>
      <div class="size-desktop clearfix">
        <ul class="options">
          <li class="first popover-options "><a data-gaq-
event="PDP~$~Size~$~20338762|" href="#" data-trigger="hover" class="btn-popover
swatch-item " data-placement="top" data-price="1899" data-special-price="854" data-simple-
sku="20338762" data-simple-id="3714337" data-discount="(55% OFF)" data-quantity="50"
data-size="S" data-original-title="" title=""><span>S</span></a></li>
          <li class="first popover-options "><a data-gaq-
event="PDP~$~Size~$~20338763|" href="#" data-trigger="hover" class="btn-popover
```



```
swatch-item " data-placement="top" data-price="1899" data-special-price="854" data-simple-
sku="20338763" data-simple-id="3714337" data-discount="(55% OFF)" data-quantity="50"
data-size="M" data-original-title="" title=""><span>M</span></a></li>
    <li class="first popover-options "><a data-gaq-
event="PDP~$~Size~$~20338764|" href="#" data-trigger="hover" class="btn-popover
swatch-item " data-placement="top" data-price="1899" data-special-price="854" data-simple-
sku="20338764" data-simple-id="3714337" data-discount="(55% OFF)" data-quantity="50"
data-size="L" data-original-title="" title=""><span>L</span></a></li>
    <li class="first popover-options "><a data-gaq-
event="PDP~$~Size~$~20338765|" href="#" data-trigger="hover" class="btn-popover
swatch-item " data-placement="top" data-price="1899" data-special-price="854" data-simple-
sku="20338765" data-simple-id="3714337" data-discount="(55% OFF)" data-quantity="50"
data-size="XL" data-original-title="" title=""><span>XL</span></a></li>
    <li class="first popover-options "><a data-gaq-
event="PDP~$~Size~$~20338767|" href="#" data-trigger="hover" class="btn-popover
swatch-item " data-placement="top" data-price="1899" data-special-price="854" data-simple-
sku="20338767" data-simple-id="3714337" data-discount="(55% OFF)" data-quantity="50"
data-size="XXL" data-original-title="" title=""><span>XXL</span></a></li>
    <li class="first popover-options "><a data-gaq-
event="PDP~$~Size~$~20338768|" href="#" data-trigger="hover" class="btn-popover
swatch-item " data-placement="top" data-price="1899" data-special-price="854" data-simple-
sku="20338768" data-simple-id="3714337" data-discount="(55% OFF)" data-quantity="95"
data-size="3XL" data-original-title="" title=""><span>3XL</span></a></li>
</ul>
</div>
<div class="stock-alert hide hidden-xs"><span class="label">Low Stock:
</span><span class="text">Only <span class="value"></span>left.</span></div>
<div class="visible-xs size-extra">
<div class="select-box mlb-dropdown " data-err-
container=".swatch.size">
    <div class="custom-select-boxes">Size<span class="change-
value">Select Size</span></div>
    <select data-rule="" class="form-control size-drop-down"
name="productSize" data-valid="" id="size-drop-down">
    <option value="" disabled="disabled" selected="selected"
style="visibility: none;">Select Size </option>
    <option value="S" data-qty="50" data-price="" data-special-price=""
data-simple-sku="20338762" data-simple-id="3714337" data-discount="" class="swatch-item
" data-quantity="50">S</option>
    <option value="M" data-qty="50" data-price="" data-special-price=""
data-simple-sku="20338763" data-simple-id="3714337" data-discount="" class="swatch-item
" data-quantity="50">M</option>
    <option value="L" data-qty="50" data-price="" data-special-price=""
data-simple-sku="20338764" data-simple-id="3714337" data-discount="" class="swatch-item
" data-quantity="50">L</option>
    <option value="XL" data-qty="50" data-price="" data-special-
price="" data-simple-sku="20338765" data-simple-id="3714337" data-discount=""
class="swatch-item " data-quantity="50">XL</option>
    <option value="XXL" data-qty="50" data-price="" data-special-
price="" data-simple-sku="20338767" data-simple-id="3714337" data-discount=""
class="swatch-item " data-quantity="50">XXL</option>
    <option value="3XL" data-qty="95" data-price="" data-special-
price="" data-simple-sku="20338768" data-simple-id="3714337" data-discount=""
class="swatch-item " data-quantity="95">3XL</option>
</select><p class="error"></p>
```

```
</div>
</div>
<span class="arrow visible-xs"></span>
<p id="pdp-size-error" class="error"></p>
</div>
</div>
<%-- <--Sizes selection ends-->--%>
  <%-- <--color selection starts-->--%>
  <script type="text/javascript">ga("send", "event", "Colour Picker", "Color Picker
Shown", "3714337", { nonInteraction: 1 });</script>
  <p class="size-suggestion hide" id="size-suggestion" data-csrf="9YIHpFI0-
BPzXmtwAwoMwicTxRPxRzfaHeq4" data-sku="3714337" data-articletype="Kurtas"><span
class="size-suggestion-text"></span><span class="size-suggestion-value"></span></p>
  <div class="actions ">
    <div class="primary-action">

</div>
</div>
<%-- <--color selection ends-->--%>
<div class="additional-text">
  <h4 class="delivery-hd visible-xs text-center">DELIVERY OPTIONS</h4>
  <div class="prod-delivery">
    <div id="cod-label" class="">Cash on delivery might be available</div>
    <div class="popover-options">
      <div class="btn-popover delivery-popover title zip-data delivery-info
hidden" id="pdp-tat-tool-tip" data-trigger="hover" data-container="body" data-
placement="left" data-content="&lt;span class=&quot;popover-close hidden-
xs&quot;&gt;&lt;/span&gt;&lt;span id=&quot;pdp-holiday-
reason&quot;&gt;&lt;/span&gt;&lt;span id=&quot;pdp-tat-
msg&quot;&gt;Loading&lt;/span&gt;" data-original-title="" title="">DISPATCH in 15 days |
COD available </div>
      <div class="content" id="pdp-tat-content"><span class="popover-close
hidden-xs"></span><span id="pdp-holiday-reason"></span><span id="pdp-tat-
msg">Loading</span></div>
      <div class="zip-code clearfix">
        <div class="zip-data" id="pdp-zip-link">
          <span class="text"></span><span class="zip"></span>
          <button class="zip-edit-btn" data-gaq-
event="pdp-~-check_delivery_details~$~3714337">Check Delivery Info </button>
        </div>
        <div class="delivery-options hide"></div>
        <div class="input-group change-pin-fields hide" id="pdp-tat" value="">
          <div class="form-group">
            <div class="title visible-xs">Check Delivery Info</div>
            <input class="form-control" placeholder="Enter Pincode" data-
rule="postcode" idx="tat" maxlength="6" data-placeholder="Enter Pincode"
type="text"><span class="input-group-addon btn-primary change-pincode" data-gaq-
event="pdp-~-pincode_check~$~3714337">Check</span><span class="input-group-addon
pincode-cancel change-pincode">Cancel</span><input name="_csrf" value="9YIHpFI0-
BPzXmtwAwoMwicTxRPxRzfaHeq4" type="hidden"><input name="tatPayload"
value="{&quot;price&quot;:854,&quot;mrp&quot;:1899,&quot;warehouses&quot;:[&quot;219
&quot;],&quot;leadTime&quot;:3,&quot;launchDate&quot;:&quot;&quot;,&quot;returnPeriod&
quot;:15,&quot;flags&quot;:{&quot;isHazmat&quot;:false,&quot;isFragile&quot;:false,&quot;is
Jewellery&quot;:false,&quot;isExchangeable&quot;:true,&quot;pickupEnabled&quot;:true,&quot;
```

```

    uot;isTryable&quot;;true,&quot;isLarge&quot;;false,&quot;isReturnable&quot;;true,&quot;cod
    Enabled&quot;;true}}" type="hidden"><p id="pdp-not-valid-pin" class="error hidden"></p>
        </div>
    </div>
</div>
</div>
</div>
</div>
<div class="policy-info">
    <div class="popover-options">
        <div class="btn-popover delivery-popover title delivery-info" id="pdp-
policy" data-trigger="hover" data-container="body" data-placement="left" data-
content="&lt;ul class=&quot;details&quot;&gt;&lt;li&gt;&lt;span&gt;&lt;/span&gt; 15 days free
return/exchange &lt;/li&gt;&lt;li&gt;&lt;span&gt;&lt;/span&gt; 100% Authentic
product&lt;/li&gt;&lt;/ul&gt;" data-original-title="" title="">Policy and Authenticity</div>
        <div class="content" id="pdp-policy-content">
            <ul class="details">
                <li><span></span>15 days free return/exchange </li>
                <li><span></span>100% Authentic product</li>
            </ul>
        </div>
        <span class="hidden policy-default-text" data-text="true"></span>
    </div>
</div>
</div>
</div>
<asp:ImageButton ID="ImageButton1" ImageUrl="images/addtocart.jpg"
OnClick="ImageButton1_Click" Width="200px" runat="server" />
<asp:ImageButton ID="ImageButton2" CssClass="pure-form-aligned"
ImageUrl="images/addtowishlist.png" OnClick="ImageButton2_Click" Height="75px"
Width="75px" runat="server" />
</div>

</div>
<%-- product details price ext ends here--%>

<div class="row off-screen" id="zoom-layer">
    <div class="col-md-12">
        <div class="img-wrapper">
            <div class="carousel slick-initialized slick-slider" data-dots="false" data-
fade="true" data-load-on-demand="true" data-arrows="true" style="height: auto;">
                <div class="slick-list draggable" tabindex="0">
                    </div>
            </div>
        </div>
    </div>
</div>

</div>

</div>
<%--section for product info begins--%>
<section class="row-fluid">

```

```

<section class="tabs-wrapper clearfix">
  <div role="tabpanel" class="col-sm-12 col-md-9 col-lg-9">
    <ul class="clearfix review" id="review-accordion">
      <li>
        <div class="accord-head accord-info tab-height ">PRODUCT INFO &
        CARE<span class="common-sprite expand-collapse expand-icon pull-right"></span></div>
        <div class="container-fluid accord-discription">
          <div class="row content-product">
            <div class="prod-tab-desc product-info-care col-md-9 col-xxs-12"
            id="productInfo">
              <div class="info-title hidden-xs content-heading">Product Info &
              Care</div>
              <%-- <a href="" data-gaq-event="pdp~$~Brand_Logo~$~La Firangi">-
              -%>
                </a>
                <section class="prod-info">
                  <div class="detail">
                    <h2 class="prod-disc product-info-right"
                    itemprop="description">Heading description here</h2>
                    </div>
                    <ul class="prod-main-wrapper">
                      <li><span class="product-info-left">Material & Care</span>
                      <span class="product-info-right">Regular Hand Wash At 30
                      Degree Temperature, Do Not Bleach, Mild Iron, Do Not Tumble Dry.</span></li>
                      <li class="authorized-brand"><span class="product-info-
                      left">Authorization</span>
                        <span>La Firangi authorized online sales partner. </span>
                        <div class="brand-authorization"><a href="javascript:void()"
                        class="authorized-link" target="_blank" id="certificate">View Certificate</a></div>
                      </li>
                      <li><span class="manufacture-txt">Manufacturer
                      Detail</span><p class="manufacture-data">MALHOTRA INTEGRATORS B-72
                      SHAHZADABAGH NEAR RAM DHARAM KANTA NEW DELHI 110035</p>
                      </li>
                      <li><span class="origin-txt">Country Of Origin</span><p
                      class="origin-data">INDIA</p>
                      </li>
                      <li><span class="article-attributes-key">Sleeve
                      Length</span><p class="article-attributes-val">Short Sleeves</p>
                      </li>
                      <li><span class="article-attributes-key">Length</span><p
                      class="article-attributes-val">Ankle Length</p>
                      </li>
                      <li><span class="article-attributes-key">Fabric</span><p
                      class="article-attributes-val">Silk</p>
                      </li>
                      <li><span class="product-code-txt">Product Code</span><p
                      class="product-code">3714337</p>
                      </li>
                    </ul>
                    <p class="seller-pop-p">
                    Sold by -
                    <button class="btn-seller-info" name="SellerInfo" id="seller-
                    info"><span><a href="javascript:void(0);">Elixirnet </a></span></button>

```

```

        </p>
    </section>
</div>
</div>
</div>
</li>
</ul>

</div>

</section>
</section>

<%-- section for product info ends--%>

<%--copied conent--%>
<%-- </div>
</div>
</div>
</div>
</div>--%>
<%--copied conent--%>
</div>

</asp:Content>

***** ManageProductsource.aspx.cs *****
using DowBAL;
using DowBEntity;
using DowDAL;
using System;
using System.Collections.Generic;
using System.Data;
using System.Linq;
using System.Text;
using System.Text.RegularExpressions;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.Security;
using System.IO;
using System.Data.SqlClient;
using System.Configuration;

public partial class ManageProductsource : System.Web.UI.Page
{
    DDLogTracer objlog = new DDLogTracer();

    List<Eteproduct> LstChecked = new List<Eteproduct>();
    string Msg = "";
    string emailtype = "Email";
    protected void Page_Load(object sender, EventArgs e)
    {

```

```

ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
if (Clsusr != null)
{
    string pageid = Request.QueryString["pageid"];
    string searchtype = Request.QueryString["searchtype"];
    string bytype = Request.QueryString["bytype"];
    ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
Convert.ToInt32(Clsusr.GroupId));
    string msg = Request.QueryString["msg"];
    string fname = this.ClientQueryString;
    if (msg != null)

        {
            //Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
        }
}
}
private void BindProductList(string pagetype, decimal ProductId)
{
    int prid = 2;
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    IList<Eteproductbuyers> Lst = (new Products()).GetProductBuyersByPID(ProductId,
prid, "ManageProductSource");
    foreach (var lst in Lst)
    {

    }
    if (Lst != null && Lst.Count > 0)
    {

        //if (WDGrid_ProductList.HasDataSource == true) {
WDGrid_ProductList.ClearDataSource(); }
        //WDGrid_ProductList.DataSource = Lst;
        //WDGrid_ProductList.DataBind();
        //WDGrid_ProductList.Visible = true;
        //Msg += "(" + Lst.Count.ToString() + ") Products Matched. ";
    }
    else
    {
        //Common.ShowMessageHtml("No records found", 0, true, dvMsgBox);
        Msg += "(0) PRODUCTS MATCHED. ";
        //WDGrid_ProductList.Visible = false;
    }
    if (Clsusr.GroupId == "3" || Clsusr.GroupId == "1")
    {
        IList<Eteproductbuyers> Lst1 = (new
Products()).GetProductBuyersByPID(ProductId, 2, "ManageProductSource");
        if (Lst1 != null && Lst1.Count > 0)
        {

            //if (WDGrid_ProductList.HasDataSource == true) {
WDGrid_ProductList.ClearDataSource(); }
            //WebDataGrid1.DataSource = Lst1;
            //WebDataGrid1.DataBind();

```

```

        //WebDataGrid1.Visible = true;
    }
    else
    {
        //WebDataGrid1.Visible = false;
    }
}
}
protected void btn_email_Click(object sender, EventArgs e)
{ //send email from buy offer
    //ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    //int UserId = Convert.ToInt16(Clsusr.UserId);
    //string emailstr = "";
    //emailstr = " Your Buy Offer for " + txt_Productdesc.Text + " Qty. requirement of " +
txt_qty.Text + " \n\r";
    //emailstr = emailstr + "Price :" + ddl_currency3.SelectedItem.ToString() +
txt_price3.Text + " \n\r";

    //emailstr = emailstr + "Qty.Available :" + txt_qty_avail.Text + " In Units:" +
ddl_uom.SelectedItem.ToString() + "\n\r";
    //emailstr = emailstr + "Lead Time :" + txt_leadtime.Text +
ddl_leadtime.SelectedItem.ToString() + "\n\r";

    //emailstr = emailstr + "Contact Details of the Sender :" + "\n\r";
    //emailstr = emailstr + "Email :" + txt_email3.Text + "\n\r";
    //emailstr = emailstr + "Company Name :" + txt_Iname3.Text + "\n\r";
    //emailstr = emailstr + "Contact Peron Name :" + txt_fname3.Text + "\n\r";
    //emailstr = emailstr + "Contact No :" + txt_urcontactno.Text + "\n\r";

    //Email email = new Email();
    //email.fromEmail = "emailservice@DateOrWed.com";
    //email.toList = txt_email.Text;
    //email.ccList = "admin@DateOrWed.com";
    //email.bccList = "owner@DateOrWed.com";
    //email.messageBody = emailstr;
    //email.subject = "Offer against your Buy Request";
    //email.sendEmail();
    //EteTblUser_emails em = new EteTblUser_emails();
    //em.EmailId = 0;
    //em.EMSentUserId = Convert.ToInt16(Clsusr.UserId);
    //em.EMRcvdUserId = Convert.ToInt16(txt_userid.Text);
    //em.EmSubject = email.subject;
    //em.EmBody = emailstr;
    //em.UserId = UserId;
    //em.CreatedOn = DateTime.Now;
    //bool ret = new Products().InsertEmail(em);
    //popup.Hide();
    //pnl_OnHoldDescription.Visible = false;
    //Response.Redirect("~/Home.aspx?pageid=1");
}

protected void LinkButton2_Click(object sender, EventArgs e)
{
    LinkButton lnkbutton = (LinkButton)sender; ;
    string OrdId = lnkbutton.CommandName;

```

```

ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];

//BindetailsGroup(OrdId);

}
protected void LinkButton1_Click(object sender, EventArgs e)
{
    LinkButton lnkbutton = (LinkButton)sender; ;
    decimal ProductId = Convert.ToDecimal(lnkbutton.CommandName);
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    int SellerId = Convert.ToInt16(Clsusr.UserId);
    bool ret = new Products().UpdateSellerProduct(ProductId, SellerId);

}
private void ManageNavigations(int pageid, int userid, int groupid)
{
    int psid = Convert.ToInt16(Request.QueryString["PSId"]);
    if (psid == 0) { psid = 1; }
    StringBuilder sb = new StringBuilder();
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];

    var list=new Products().GetProductSellerImage(psid,"").ToList();
    string path=list[0].ImagePath;
    sb.Append("<div id='gallery'> \n <div id='panel'>");

    sb.Append("<img id='largelImage' src='" + path + "' alt='" + psid + "'
style='height:500px;width:550px;' onmouseout='imageout()' onmouseover='\"imageZoom("
+ "largelImage" + ", 'myresult" + ")\" /> \n"); //style='height:400px;width:600px;'
    sb.Append("</div>\n ");
    sb.Append("<script>\n ");
    // Initiate zoom effect:
    sb.Append("imageZoom('largelImage', 'myresult')");
    sb.Append("</script>\n");
    sb.Append("<div id='thumbs'>\n ");
    foreach (var dr in list)
    {
        sb.Append("<img src='" + dr.ImagePath + "' class='thumb' alt='" + dr.srno + "'
style='height:100px;width:80px;' />\n ");
    }
    sb.Append("</div>\n ");
    sb.Append("<script>\n ");
    sb.Append(" $('#thumbs').delegate('img', 'click', function () { $('#largelImage').attr('src',
$(this).attr('src').replace('thumb', 'large'));});\n ");
    sb.Append("</script>\n</div>\n ");
    string PreHtml = sb.ToString();
    if (PreHtml.Length > 0)
    {
        //objlog.Trace_Log(4, "Gordersearchlist", "1", "SliderList", PreHtml);

        test.InnerHtml = PreHtml;
    }
    sb = new StringBuilder();
}

```



```

var lst = new Products().GetProductSeller(psid, "");
sb.Append("<h1 class='brand'>" + lst.BrandName + "</h1>");
sb.Append("<h1 class='product-title'>" + lst.SellerDescription + "</h1>");
sb.Append("<h2 class='price discounts' >Price:Rs." + lst.Price + "</h2>");
//sb.Append(" <button class='btn btn-primary' name='AddToCart' type='submit'
onclick='ImageButton1_Click' data-placement='top' id='add-to-cart' > Add To Cart </button>
");
//sb.Append(" <button class='btn btn-payment add-to-cart' name='AddToCart'
type='submit' data-placement='top' id='add-to-cart' > Add To Bag </button> ");
//sb.Append("<button class='btn-fav app-vendors-item-guardar no-hover'
name='SaveForLater' id='save-for-later' ><span class=' save-icon ' ></span><span
class='save-msg' > </span></button>");
// sb.Append("<div class='size-desktop clearfix'>\n ");
//sb.Append("<ul class='options'>\n ");
//sb.Append("<li class='first popover-options ' ><a href='#' data-trigger='hover' class='btn-
popover swatch-item ' data-placement='top' data-price='1899' data-special-price='854' data-
simple-sku='20338762' data-simple-id='3714337' data-discount='(55% OFF)' data-
quantity='50' data-size='S' data-original-title=' ' title=' '><span>S</span></a></li>\n ");
//sb.Append("<li class='first popover-options ' ><a href='#' data-trigger='hover' class='btn-
popover swatch-item ' data-placement='top' data-price='1899' data-special-price='854' data-
simple-sku='20338763' data-simple-id='3714337' data-discount='(55% OFF)' data-
quantity='50' data-size='M' data-original-title=' ' title=' '><span>M</span></a></li>\n ");
//sb.Append("<li class='first popover-options ' ><a href='#' data-trigger='hover' class='btn-
popover swatch-item ' data-placement='top' data-price='1899' data-special-price='854' data-
simple-sku='20338764' data-simple-id='3714337' data-discount='(55% OFF)' data-
quantity='50' data-size='L' data-original-title=' ' title=' '><span>L</span></a></li>\n ");
//sb.Append("<li class='first popover-options ' ><a href='#' data-trigger='hover' class='btn-
popover swatch-item ' data-placement='top' data-price='1899' data-special-price='854' data-
simple-sku='20338765' data-simple-id='3714337' data-discount='(55% OFF)' data-
quantity='50' data-size='XL' data-original-title=' ' title=' '><span>XL</span></a></li>\n ");
//sb.Append("<li class='first popover-options ' ><a href='#' data-trigger='hover' class='btn-
popover swatch-item ' data-placement='top' data-price='1899' data-special-price='854' data-
simple-sku='20338767' data-simple-id='3714337' data-discount='(55% OFF)' data-
quantity='50' data-size='XXL' data-original-title=' ' title=' '><span>XXL</span></a></li>\n ");
//sb.Append("<li class='first popover-options ' ><a href='#' data-trigger='hover' class='btn-
popover swatch-item ' data-placement='top' data-price='1899' data-special-price='854' data-
simple-sku='20338768' data-simple-id='3714337' data-discount='(55% OFF)' data-
quantity='95' data-size='3XL' data-original-title=' ' title=' '><span>3XL</span></a></li>\n ");
//sb.Append("</ul>\n ");
//sb.Append("</div>\n ");
sb.Append(" <div class='additional-text' > ");
sb.Append(" <h4 class='delivery-hd visible-xs text-center'>DELIVERY OPTIONS</h4>
");
sb.Append(" <div class='prod-delivery' > ");
sb.Append(" <div id='cod-label' class=' '>Cash on delivery might be available</div> ");
sb.Append(" <div class='popover-options' > ");
sb.Append(" <div class='btn-popover delivery-popover title zip-data delivery-info hidden'
id='pdp-tat-tool-tip' data-trigger='hover' data-container='body' data-placement='left' data-
content='&lt;span class='&quot;popover-close hidden-xs&quot;&gt;&lt;/span&gt;&lt;span
id='&quot;pdp-holiday-reason&quot;&gt;&lt;/span&gt;&lt;span id='&quot;pdp-tat-
msg&quot;&gt;Loading&lt;/span&gt;' data-original-title=' ' title=' '>DISPATCH in 15 days |
COD available </div> ");
sb.Append(" <div class='content' id='pdp-tat-content'><span class='popover-close
hidden-xs'></span><span id='pdp-holiday-reason'></span><span id='pdp-tat-
msg'>Loading</span></div> ");

```

```

sb.Append(" <div class='zip-code clearfix'> ");
sb.Append(" <div class='zip-data' id='pdp-zip-link'> ");
sb.Append(" <span class='text'></span><span class='zip'></span> ");
sb.Append(" <button class='zip-edit-btn' data-gaq-
event='pdp~$~check_delivery_details~$~3714337'>Check Delivery Info </button> ");
sb.Append(" </div> ");
sb.Append(" <div class='delivery-options hide'></div> ");
sb.Append(" <div class='input-group change-pin-fields hide' id='pdp-tat' value=''> ");
sb.Append(" <div class='form-group'> ");
sb.Append(" <div class='title visible-xs'>Check Delivery Info</div> ");
sb.Append(" <input class='form-control' placeholder='Enter Pincode' data-rule='postcode'
idx='tat' maxlength='6' data-placeholder='Enter Pincode' type='text'><span class='input-
group-addon btn-primary change-pincode' data-gaq-
event='pdp~$~pincode_check~$~3714337'>Check</span><span class='input-group-addon
pincode-cancel change-pincode'>Cancel</span><input name='_csrf' value='9YIHpFI0-
BPzXmtwAwOMWicTxRPxRzfaHeq4' type='hidden'><input name='tatPayload'
value='{&quot;price&quot;:854,&quot;mrp&quot;:1899,&quot;warehouses&quot;:[&quot;219&
quot;],&quot;leadTime&quot;:3,&quot;launchDate&quot;:&quot;&quot;,&quot;returnPeriod&
quot;:15,&quot;flags&quot;:{&quot;isHazmat&quot;:false,&quot;isFragile&quot;:false,&quot;isJ
ewellery&quot;:false,&quot;isExchangeable&quot;:true,&quot;pickupEnabled&quot;:true,&qu
ot;isTryable&quot;:true,&quot;isLarge&quot;:false,&quot;isReturnable&quot;:true,&quot;codE
nabled&quot;:true}}' type='hidden'><p id='pdp-not-valid-pin' class='error hidden'></p> ");
sb.Append(" </div> ");
sb.Append(" </div> ");
sb.Append(" </div> ");
sb.Append(" </div> ");
sb.Append(" </div> ");
sb.Append(" <div class='policy-info'> ");
sb.Append(" <div class='popover-options'> ");
sb.Append(" <div class='btn-popover delivery-popover title delivery-info' id='pdp-policy'
data-trigger='hover' data-container='body' data-placement='left' data-content='&lt;ul
class='&quot;details&quot;&gt;&lt;li&gt;&lt;span&gt;&lt;/span&gt; 15 days free
return/exchange &lt;/li&gt;&lt;li&gt;&lt;span&gt;&lt;/span&gt; 100% Authentic
product&lt;/li&gt;&lt;/ul&gt;' data-original-title='' title=''>Policy and Authenticity</div> ");
sb.Append(" <div class='content' id='pdp-policy-content'> ");
sb.Append(" <ul class='details'> ");
sb.Append(" <li><span></span>15 days free return/exchange </li> ");
sb.Append(" <li><span></span>100% Authentic product</li> ");
sb.Append(" </ul> ");
sb.Append(" </div> ");
sb.Append(" <span class='hidden policy-default-text' data-text='true'></span> ");
sb.Append(" </div> ");
sb.Append(" </div> ");
sb.Append(" </div> ");

```

```

PreHtml = sb.ToString();
if (PreHtml.Length > 0)
{
    //objlog.Trace_Log(2, "Gordersearchlist", "2", "SliderList", PreHtml);

    prodkart.InnerHtml = PreHtml;
}
//vendor icon
var vend = new tblUsers().getuserprofilebyid(Convert.ToInt16( CIsusr.UserId));
if (vend.Company_logo!=null)

```

```

    {
        byte[] bytes = (byte[])vend.Company_logo.ToArray();
        string base64String = Convert.ToBase64String(bytes, 0, bytes.Length);
        string imageUrl = "data:image/png;base64," + base64String;

        vendoricon.Src = imageUrl;    }
}

protected void Check_SelectedIndexChanged(object sender, EventArgs e)
{

    int selectedGroupid = Convert.ToInt32(0);

}

// This is Event Handler
protected void myBtnClickHandler(object sender, EventArgs e)
{

    LinkButton lnk = sender as LinkButton;

    if (lnk != null)
    {
        string ID = lnk.ID;
        Common.ShowMessageHtml("Please wait while refreshing the records", 1, true,
dvMsgBox);
        string pid = (Convert.ToInt16(Request.QueryString["pid"]) - 8).ToString();
        DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
        if (ds != null)
        {
            if (ds.Tables[0].Rows.Count > 0)
            {
                DataTable dtdll = ds.Tables[0];
                foreach (DataRow drnew in dtdll.Rows)
                {
                    switch (drnew["lbtncode"].ToString())
                    {
                        case "SAVE":
                            ContentSaveControlClicked();
                            // Call SAVE Method
                            break;
                        case "URL":
                            Response.Redirect(drnew["pageurl"].ToString() + "&pid=" +
drnew["lbtnid"]);
                            break;
                        case "AUDIT":
                            Response.Redirect(drnew["pageurl"].ToString() + "&frompage=" +
Request.QueryString["pageid"]);
                    }
                }
            }
        }
    }
}

```

```

        break;

        case "CANCEL":
            Response.Redirect("~/Home.aspx?pageid=1");
            break;
        case "ADDSUB":

Response.Redirect("~/ManageSubgroup.aspx?pageid=5&PageType=Add&ProductId=" +
pid);
            break;
        case "ADDPRODUCT":

Response.Redirect("~/ManageProducts.aspx?pageid=5&PageType=Add&ProductId=" +
pid);
            break;
        case "SENDQUOTE":
            //Div5.InnerHtml = "Your Offer";
            emailtype = "Email";
            //popup.Show();
            break;
        case "POSTOFFER":
            ContentPostControlClicked();
            break;
        case "SHORTLIST":
            ContentShortListControlClicked();
            break;
        case "EMAILCOUNTER":
            //Div5.InnerHtml = "Your Counter Offer";
            //emailtype = "EmailCounter";
            //popup.Show();
            break;
        case "PICTURE":
            break;

        default:
            break;
    }
}
}
}
}
}
}
}
DropDownList dropDownList = sender as DropDownList;
if (dropDownList != null)
{
    Common.ShowMessageHtml("", 0, false, dvMsgBox);
    Bindetails_Filter(Table1);
}
}
}
private void ContentShortListControlClicked()
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedIn_Specs"];

```

```

EteUserShortList etsl = new EteUserShortList();
etsl.PSId = Convert.ToInt32(Request.QueryString["PSId"]);
etsl.UserId = Convert.ToInt16(Clsusr.UserId);
etsl.UId = Convert.ToInt16(Clsusr.UserId);
etsl.CreatedOn = DateTime.Now;
etsl.Active = true;
bool ret = new Products().InsertShortList(etsl, true);
if (ret == true)
{

    Common.ShowMessageHtml("Shortlisting of Product Sucessful ", 1, true,
dvMsgBox);
    Response.Redirect("~/Home.aspx?pageid=1&msg=Shortlisting of Product
Sucessful");
}
else
{
    Common.ShowMessageHtml("Shortlisting of Product Not Sucessful ", 1, true,
dvMsgBox);

}

}

private void ContentPostControlClicked()
{
    //objlog.Trace_Log(0, "DateOrWed", "MangeDealer", "SaveClick", "Start Here..");

    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    try
    {
        //if (Details.Text == "")
        {
            Common.ShowMessageHtml("Product Description cannot be blank", 0, true,
dvMsgBox);

        }
        //else if (txt_taskdetails.Text == "")
        {

            Common.ShowMessageHtml("Keywords cannot be blank", 0, true, dvMsgBox);

        }
        //else
        {
            decimal PSId = Convert.ToInt32(Request.QueryString["PSId"]);
            List<Eteproductbuyers> eps = new Products().GetProductBuyersByPSID(PSId,
""");
            decimal ProductId = (from c in eps select c.ProductId).FirstOrDefault();
            Eteproduct Prd = new Products().GetProduct(ProductId, "");
            Prd.CreatedOn = DateTime.Now;
            Prd.UserId = Convert.ToInt32(Clsusr.UserId);

```

```

//Prd.ProductName = Details.Text.ToString();
//Prd.ProductGroupId = ddl_productgroup.SelectedIndex;
//Prd.Price = Convert.ToDecimal(txt_tasksummary.Text);
//Prd.SellerId = Convert.ToInt16(Clsusr.UserId);
//Prd.CurrencyId = ddl_currency.SelectedIndex;
//Prd.AnalysisCode = txt_taskdetails.Text.ToString();
//Prd.PriceUnits = ddl_uom.SelectedIndex;
//Prd.PriceUomId = ddl_uom.SelectedIndex;
//Prd.PricePackId = ddl_packing.SelectedIndex;
//Prd.MinOrderUom = ddl_uom.SelectedIndex;
//Prd.PrdId = 2;
//bool Result = false;

//Result = new Products().Update(Prd, false);

//if (Result == true)
//{

//    Common.ShowMessageHtml("Posting of Product Successful ", 1, true,
dvMsgBox);
//}

    Response.Redirect("~/Home.aspx?pageid=1&msg=Product Updated
Sucessfully");

}

}

catch (Exception exc)
{
    Common.ShowMessageHtml(exc.ToString(), 0, true, dvMsgBox);
}

//objlog.Trace_Log(0, "DateOrWed", "MangeDealer", "SaveClick", "Start Here..");
}
private void ContentSaveControlClicked()
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    int SellerId = Convert.ToInt16(Clsusr.UserId);
    foreach (var data in LstChecked)
    {
        if (data.Check == true)
        {
            decimal id = data.ProductId;
            bool ret = new Products().UpdateSellerProduct(id, SellerId);
        }
    }
}
}

```

```

private void Bindetails_Filter(Table table)
{
    try
    {
        int brandID = 0;
        int dealerID = 0;
        int orderStatusID = 0;
        int customerType = 0;
        int dateRange = 0;
        int OrderHold = 0;

        foreach (TableRow row in table.Rows)
        {
            foreach (TableCell rowcell in row.Cells)
            {

                foreach (WebControl contrl in rowcell.Controls)
                {
                    DropDownList listDrop = contrl as DropDownList;
                    if (listDrop != null)
                    {
                        if (listDrop.Attributes["LISTTYPE"] != null)
                        {
                            string rowcellHead = listDrop.Attributes["LISTTYPE"].ToUpper();
//rowcell.Text.ToUpper();
                            ExtractAll(ref orderStatusID, ref OrderHold, ref brandID, ref
customerType, ref dateRange, ref dealerID, rowcellHead, listDrop);
                        }
                    }
                    else
                    {
                        continue;
                    }
                }
            }
        }

        FilterEnum filterLst = new FilterEnum();
        filterLst.OrderStatus = orderStatusID;
        filterLst.OrderHold = OrderHold;
        filterLst.Brand = brandID;
        filterLst.CustomerType = customerType;
        filterLst.DateRange = dateRange;
        filterLst.Dealer = dealerID;
        //RefreshData(filterLst);
    }
    catch (Exception ex)
    {
    }
}

```

```
private static void ExtractAll(ref int orderStatusID, ref int OrderHold, ref int brandID, ref int
customerType, ref int dateRange, ref int dealerID, string rowcellHead, DropDownList
listDrop)
```

```
{
    //string strddl=rowcellHead;
    switch (rowcellHead)
    {
        case "BRAND":
            if (string.Compare(listDrop.SelectedItem.Text, "All", true) == 0)
            {
                brandID = 0;
            }
            else
            {
                brandID = Convert.ToInt32(listDrop.SelectedValue);
            }
            break;
        case "CUSTOMER TYPE":
            string strCustType = listDrop.SelectedItem.Text;
            if (string.Compare(strCustType, "Residential", true) == 0)
            {
                customerType = 1;
            }
            else if (string.Compare(strCustType, "Business", true) == 0)
            {
                customerType = 2;
            }
            else
            {
                customerType = 0;
            }
            break;
        case "DATE RANGE":
            dateRange = SelectDateRange(dateRange, listDrop);
            break;
        case "DEALER":
            if (string.Compare(listDrop.SelectedItem.Text, "All", true) == 0)
            {
                dealerID = 0;
            }
            else
            {
                dealerID = Convert.ToInt32(listDrop.SelectedValue);
            }

            break;
        case "STATUS":
            if (string.Compare(listDrop.SelectedItem.Text, "All", true) == 0)
            {
                orderStatusID = 0;
            }
            else
```



```

        {
            orderStatusID = Convert.ToInt32(listDrop.SelectedValue);
        }

        break;
    case "ON HOLD":
        string strOnHold = listDrop.SelectedItem.Text;
        if (string.Compare(strOnHold, "Yes", true) == 0)
        {
            OrderHold = 1;
        }
        else if (string.Compare(strOnHold, "No", true) == 0)
        {
            OrderHold = 2;
        }
        else
        {
            OrderHold = 0;
        }
        break;
    default:
        break;
}

}

private static void SetStatus(Table tbl, string LoggedinID, string FilterName)
{
    try
    {
        foreach (TableRow row in tbl.Rows)
        {
            foreach (TableCell rowcell in row.Cells)
            {
                foreach (WebControl contrl in rowcell.Controls)
                {
                    DropDownList listDrop = contrl as DropDownList;
                    if (listDrop != null)
                    {
                        if (listDrop.Attributes["LISTTYPE"] != null &&
listDrop.Attributes["LISTTYPE"].ToUpper() == FilterName)
                        {
                            listDrop.SelectedValue = LoggedinID;
                        }
                    }
                    else
                    {
                        continue;
                    }
                }
            }
        }
    }
}

```

```

    }
    catch (Exception exc)
    {
    }
}
private static int SelectDateRange(int dateRange, DropDownList listDrop)
{
    string strDateRange = listDrop.SelectedItem.Text.ToUpper();
    switch (strDateRange)
    {
        case "ALL":
        {
            dateRange = 0;
            break;
        }
        case "TODAY":
        {
            dateRange = 1;
            break;
        }

        case "YESTERDAY":
        {
            dateRange = 2;
            break;
        }
        case "THIS WEEK":
        {
            dateRange = 3;
            break;
        }
        case "LAST WEEK":
        {
            dateRange = 4;
            break;
        }

        case "LAST 7 DAYS":
        {
            dateRange = 5;
            break;
        }
        case "LAST 30 DAYS":
        {
            dateRange = 6;
            break;
        }
        case "THIS MONTH":
        {
            dateRange = 7;
            break;
        }
        case "LAST MONTH":
        {

```

```

        dateRange = 8;
        break;
    }
    case "THIS YEAR":
    {
        dateRange = 9;
        break;
    }
    case "LAST YEAR":
    {
        dateRange = 10;
        break;
    }
    default:
        dateRange = 0;
        break;
}
return dateRange;
}

```

```

protected void Lnkbtn_ManageOrder_Click(object sender, EventArgs e)
{
    LinkButton lnkbutton = (LinkButton)sender;
    string strcustid = lnkbutton.CommandName;

    if (strcustid.Length > 0)
    {
        string[] stringSeparators = new string[] { "&", "=" };
        string[] arr1;
        arr1 = strcustid.Split(stringSeparators, StringSplitOptions.RemoveEmptyEntries);
        bool bchkstatus = false;
        foreach (var word in arr1)
        {
            if (word == "status")
            {
                bchkstatus = true;
            }

            if (bchkstatus == true && word != "status")
            {
                Response.Redirect(Navigations.StatusWiseRedirect(Convert.ToInt32(word),
strcustid));
            }
        }
    }
}
//public byte[] GetPhtoStream()
//{
//    byte[] bufferPhoto = new byte[FileUpload1.PostedFile.ContentLength];

```

```

// Stream photoStream = FileUpload1.PostedFile.InputStream;
// photoStream.Read(bufferPhoto, 0, FileUpload1.PostedFile.ContentLength);

// return bufferPhoto;
//}

protected void btn_upload_Click(object sender, EventArgs e)
{
    //ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];

    //string path = FileUpload1.FileName;
    //byte[] Photo = GetPhtoStream();
    //int Id = Convert.ToInt32(ddl_productgroup.SelectedValue);
    //int sno = Convert.ToInt16(Session["SrNo"].ToString()) + 1;
    //string base64String = Convert.ToBase64String(Photo, 0, Photo.Length);
    //string imageUrl = "data:image/png;base64," + base64String;

    //if (sno == 1)
    //{
    //    Image2.ImageUrl = imageUrl;
    //    Image2.Visible = true;
    //}
    //if (sno == 2)
    //{
    //    Image3.ImageUrl = imageUrl;
    //    Image3.Visible = true;
    //}
    //if (sno == 3)
    //{
    //    Image4.ImageUrl = imageUrl;
    //    Image4.Visible = true;
    //}
    //if (sno == 4)
    //{
    //    Image5.ImageUrl = imageUrl;
    //    Image5.Visible = true;
    //}
    //Session["SrNo"] = sno.ToString();

    //EteproductImages ep = new EteproductImages();

    //ep.ProductId = Id;
    //ep.SellerId = Convert.ToInt16(Clsusr.UserId);
    //ep.sno = 1;
    //ep.Images = Photo;
    //string PhotoID = new Products().InsertPhoto(ep);
}

protected void ImageButton1_Click(object sender, ImageClickEventArgs e)
{
    //ImageButton lnkbutton = (ImageButton)sender; ;
    //int srno = Convert.ToInt16(lnkbutton.CommandName);
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
}

```

```

int psid = Convert.ToInt16(Request.QueryString["PSId"]);
EteUserCart cart = new EteUserCart();
cart.Active = true;
    cart.UserId=Convert.ToInt16( Clsusr.UserId) ;
    cart.PSId=psid;
cart.UId=    Convert.ToInt16( Clsusr.UserId) ;
cart.CreatedOn=DateTime.Today;
cart.PayId=0;
bool ret = new Products().InsertCartList(cart);
if (ret)
{
    Response.Redirect("~/MyCart.aspx");
}

}
protected void ImageButton2_Click(object sender, ImageClickEventArgs e)
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];

    int psid = Convert.ToInt16(Request.QueryString["PSId"]);
    EteUserShortList lst = new EteUserShortList();
    lst.UserId=    Convert.ToInt16(Clsusr.UserId);
    lst.PSId=psid;
    lst.UId=    Convert.ToInt16(Clsusr.UserId);

    lst.CreatedOn=DateTime.Today;
    lst.Active=true;
    bool result = new Products().InsertShortList(lst,true);

}
}

***** ManageProductsourced.aspx.cs *****
<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="ManageProductsourced.aspx.cs"
Inherits="ManageProductsourced" %>

<%-- <%@ Register Assembly="AjaxControlToolkit" Namespace="AjaxControlToolkit"
TagPrefix="cc1" %>--%>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
<link href="CSS/PopupModal.css" rel="stylesheet" type="text/css" />

<script type="text/javascript">
function Hidepopup() {
$find("Popup_PlayRecordings").hide();
$find("popup").hide();
$find("pnl_OnHoldDescription").hide();

return false;

```

```
}  
function Hidepopup1() {  
  //alert('Hidepopup1()');  
  //$find("Popup_PlayRecordings").hide();  
  $find("popup").hide();  
  $find("pnl_OnHoldDescription").hide();  
  return false;  
}  
</script>
```

```
<script type="text/javascript">
```

```
$(window).load(function () {  
  activeTab = getCookie('seltab');  
  //alert("." + activeTab);  
  $("." + activeTab).trigger('click');  
  setCookie('seltab', "", 2);  
})
```

```
function saveCookie(ckname) {  
  setCookie('seltab', ckname, 2);  
  //alert(ckname);  
}
```

```
function setCookie(cname, cvalue, exdays) {  
  var d = new Date();  
  d.setTime(d.getTime() + (exdays * 24 * 60 * 60 * 1000));  
  var expires = "expires=" + d.toUTCString();  
  document.cookie = cname + "=" + cvalue + ";" + expires;  
}
```

```
function getCookie(cname) {  
  var name = cname + "=";  
  var ca = document.cookie.split(';');  
  for (var i = 0; i < ca.length; i++) {  
    var c = ca[i];  
    while (c.charAt(0) == ' ') c = c.substring(1);  
    if (c.indexOf(name) == 0) return c.substring(name.length, c.length);  
  }  
  return "";  
}
```

```
</script>
```

```
<style type="text/css">  
.padding {  
  padding: 0px;  
}
```

```
.padding th {  
  padding: 0px !important;  
}
```

```
.span3 td {
```

```
padding: 0px;
}
```

```
.span5 td {
padding: 0px;
}
```

```
.span5 th {
padding: 0px;
}
```

```
</style>
</asp:Content>
```

```
<asp:Content ID="Content2" ContentPlaceHolderID="CPH_TopMessages"
runat="Server">
  <div class="container-fluid">
    <div class="row-fluid" style="padding-top: 20px;">
      <div class="span2" id="dvlogo" runat="server" style="display: block;">
        
      </div>
      <div class="span10">
        <div class="row">
          <%--<h3>Ravikiran Techno Soft</h3>--%>
          <div id="dvMsgBox" class="span12 ordersave" runat="server" style="display: none;">

          <asp:Label ID="Label2" runat="server" Text="Message"></asp:Label>
        </div>
      </div>
    </div>
  </div>
</asp:Content>
<asp:Content ID="Content3" ContentPlaceHolderID="CPH_LeftSideBar" runat="Server">
  <div id="LeftSideBar" runat="server" class="span2">
    <asp:Table ID="Table1" runat="server" Width="250px">
    </asp:Table>
  </div>
</asp:Content>
```

```
<asp:Content ID="Content4" ContentPlaceHolderID="CPH_MainContent" runat="Server"
>
  <div class="span10" id="MainContent" runat="server">
    <div id="ShowPageHeader" class="row" style="display: block;" runat="server">
      <div id="PageName" class="span12 customer" runat="server">
      </div>
    </div>
    <div id="Div1" class="row" style="display: block;">
      <div id="Div2" class="span12">

      <div class="content_holder">

      <div class="span12" style="background-color: #f8f9fa;" >
```

```
<div class="row-fluid">
<div class="span7" style="width: 100%">
<div class="row">
<%--          added code--%>
```

```
<%-- added code ends--%>
</div>
</div>
</div>
```

```
</div>
</div>
```

```
</div>
```

```
</div>
</div>
</div>
</asp:Content>
```

```
***** ManageProductsourced.aspx.cs *****
```

```
using DowBAL;
using DowBEntity;
using DowDAL;
using System;
using System.Collections.Generic;
using System.Data;
using System.Linq;
using System.Text;
using System.Text.RegularExpressions;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.Security;
using System.Data.SqlClient;
using System.Configuration;
public partial class ManageProductsourced: System.Web.UI.Page
{
    List<Eteproduct> LstChecked = new List<Eteproduct>();
    string Msg="";
```



```

string emailtype = "Email";

protected void Page_Load(object sender, EventArgs e)
{
    CIsLoggedInUser CIsusr = (CIsLoggedInUser)Session["LoggedInUser_Specs"];
    if (CIsusr != null)
    {
        string pageid = Request.QueryString["pageid"];
        string searchtype = Request.QueryString["searchtype"];
        string bytype = Request.QueryString["bytype"];
        ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(CIsusr.UserId),
Convert.ToInt32(CIsusr.GroupId));
        string msg = Request.QueryString["msg"];
        if (msg != null)
        {
            Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
        }
    }
    //((BoundCheckBoxField)this.WebDataGrid1.Columns["ProductId"]).ValueConverter =
new DecimalBooleanConverter();

}

private void ManageNavigations(int pageid, int userid, int groupid)
{
    bool IsLeftBar = false;
    IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
myBtnClickHandler);

    if (IsLeftBar == true)
    {
        LeftSideBar.Attributes["class"] = "span2";
        MainContent.Attributes["class"] = "span10";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;"); //style="display:
block;";
        PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
        string source = Request.QueryString["source"];
        string pagetype = Request.QueryString["PageType"];

    }
    else
    {
        LeftSideBar.Attributes["class"] = "";
        MainContent.Attributes["class"] = "span12";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
        PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
}

protected void Check_SelectedIndexChanged(object sender, EventArgs e)
{

```

```

int selectedGroupid = Convert.ToInt32(0);

}

// This is Event Handler
protected void myBtnClickHandler(object sender, EventArgs e)
{

    //LinkButton lnk = (LinkButton)sender;
    //string ID = lnk.ID;

    LinkButton lnk = sender as LinkButton;

    if (lnk != null)
    {
        string ID = lnk.ID;
        Common.ShowMessageHtml("Please wait while refreshing the records", 1, true,
dvMsgBox);
        string pid = (Convert.ToInt16(Request.QueryString["pid"]) - 8).ToString();
        DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
        if (ds != null)
        {
            if (ds.Tables[0].Rows.Count > 0)
            {
                DataTable dtdll = ds.Tables[0];
                foreach (DataRow drnew in dtdll.Rows)
                {
                    switch (drnew["lbtncode"].ToString())
                    {
                        case "SAVE":
                            ContentSaveControlClicked();
                            // Call SAVE Method
                            break;
                        case "URL":
                            Response.Redirect(drnew["pageurl"].ToString() + "&pid=" +
drnew["lbtnid"]);
                            break;
                        case "AUDIT":
                            Response.Redirect(drnew["pageurl"].ToString() + "&frompage=" +
Request.QueryString["pageid"]);
                            break;

                        case "CANCEL":
                            Response.Redirect("~/Home.aspx?pageid=1");
                            break;
                        case "ADDSUB":

                            Response.Redirect("~/ManageSubgroup.aspx?pageid=5&PageType=Add&ProductId=" +
pid);
                            break;
                        case "ADDPRODUCT":

```

```
Response.Redirect("~/ManageProducts.aspx?pageid=5&PageType=Add&ProductId=" + pid);
```

```
        break;
    case "SENDEMAIL":
        //Div5.InnerHtml = "Your Offer";
        ///ddl_Reasons
        //popup.Show();
        break;
    case "POSTOFFER":
        //ContentPostControlClicked();
        break;
    case "ADDCART":
        ContentAddCartControlClicked();
        break;
    case "ADDWISHLIST":
        ContentWishListControlClicked();
        break;
    case "EMAILCOUNTER":
        //Div5.InnerHtml = "Your Counter Offer";
        //emailtype = "EmailCounter";
        //popup.Show();
        break;

    default:
        break;
    }
}
}
}
}
}
}
}
private void ContentSaveControlClicked()
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    int SellerId = Convert.ToInt16(Clsusr.UserId);
    foreach (var data in LstChecked)
    {
        if (data.Check == true)
        {
            decimal id = data.ProductId;
            bool ret = new Products().UpdateSellerProduct(id, SellerId);
        }
    }
}
private void ContentAddCartControlClicked()
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];

    EteUserCart etsl = new EteUserCart();
```

```

etsl.PSId = Convert.ToInt32(Request.QueryString["PSId"]);
etsl.UserId = Convert.ToInt16(Clsusr.UserId);
etsl.UId = Convert.ToInt16(Clsusr.UserId);
etsl.CreatedOn = DateTime.Now;
etsl.Active = true;
etsl.PayId = 0;

bool ret = new Products().InsertCartList(etsl);
if (ret == true)
{
    Common.ShowMessageHtml("Shortlisting of Product Sucessful ", 1, true,
dvMsgBox);
    Response.Redirect("~/Home.aspx?pageid=1&msg=Shortlisting of Product
Sucessful");
}
else
{
    Common.ShowMessageHtml("Shortlisting of Product Not Sucessful ", 1, true,
dvMsgBox);
}
}

private void ContentWishListControlClicked()
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];

    EteUserShortList etsl = new EteUserShortList();
    etsl.PSId = Convert.ToInt32(Request.QueryString["PSId"]);
    etsl.UserId = Convert.ToInt16(Clsusr.UserId);
    etsl.UId = Convert.ToInt16(Clsusr.UserId);
    etsl.CreatedOn = DateTime.Now;
    etsl.Active = true;
    bool ret = new Products().InsertShortList(etsl,false);
    if (ret == true)
    {
        Common.ShowMessageHtml("Product Added to WishList Sucessfully ", 1, true,
dvMsgBox);
        Response.Redirect("~/Home.aspx?pageid=1&msg=Product Added to Cart
Sucessfully ");
    }
    else
    {
        Common.ShowMessageHtml("Shortlisting of Product Not Sucessful ", 1, true,
dvMsgBox);
    }
}
}

```



```

    }
  }
}
function validatealphanumeric(key) {
  var keycode = (key.which) ? key.which : key.keyCode
  var phn = document.getElementById('txtChar');
  //comparing pressed keycodes
  //alert("Test1")
  if (!(keycode == 8 || keycode == 32 || keycode == 46) && (keycode < 48 || keycode >
57) && (keycode < 65 || keycode > 90) && (keycode < 97 || keycode > 122))
  {
    return false;
  }
  else {
    return true;
  }
}
function validatealpha(key) {
  var keycode = (key.which) ? key.which : key.keyCode
  var phn = document.getElementById('txtChar');
  //comparing pressed keycodes
  //alert("Test1")
  if (!(keycode == 8 || keycode == 32 || keycode == 46) && (keycode < 65 || keycode
> 90) && (keycode < 97 || keycode > 122))
  {
    return false;
  }
  else {
    return true;
  }
}

function isNumberKey(evt) {
  var charCode = (evt.which) ? evt.which : event.keyCode;
  if (charCode != 46 && charCode > 31 && (charCode < 48 || charCode > 57)) {
    return false;
  } else {
    // If the number field already has . then don't allow to enter . again.
    if (evt.target.value.search(/\./) > -1 && charCode == 46) {
      return false;
    }
    return true;
  }
}
}
</script>
</asp:Content>

```

```

<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
  <div class="container-fluid">
    <div class="row-fluid" style="padding-top: 20px;">
      <div class="span2" id="dvlogo" runat="server" style="display: block;">
        <%----%>
      </div>
      <div class="span10">

```



```

        <tr>
            <td align="right" class="label_txt"><strong>Name
:*</strong></td>
            <td align="left">
                <asp:TextBox ID="txt_FirstName" runat="server"
MaxLength="100" onkeypress="return validatealpha(event)"></asp:TextBox>
            </td>
        </tr>
        <tr>
            <td align="right" class="label_txt"><strong>Company
Name :</strong></td>
            <td width="25%" align="left">
                <asp:TextBox ID="txt_LastName" runat="server"
MaxLength="40" onkeypress="return validatealpha(event)"></asp:TextBox>
            </td>
        </tr>

        <tr>
            <td align="right"
class="label_txt"><strong>Country:</strong></td>
            <td width="25%" align="left">
                <asp:DropDownList ID="ddl_country"
runat="server"></asp:DropDownList></td>
        </tr>

        <tr>
            <td align="right" class="label_txt"><strong>Mobile
:*</strong></td>
            <td width="25%" align="left">
                <asp:TextBox ID="txt_mobile" runat="server"
onkeypress="return isNumberKey(event)" MaxLength="12"
TextMode="Phone"></asp:TextBox>
            </td>
        </tr>
        <tr>
            <td align="right" class="label_txt"><strong>LandLine
:*</strong></td>
            <td width="25%" align="left">
                <asp:TextBox ID="txt_landline" runat="server"
onkeypress="return isNumberKey(event)" MaxLength="12"
TextMode="Phone"></asp:TextBox>
            </td>
        </tr>

        <%-- <tr>
            <td width="17%" align="right"
class="label_txt"><strong>Email :*</strong></td>
            <td width="44%" align="left">
                <asp:TextBox ID="email_name" runat="server"
MaxLength="50" TextMode="Email" CausesValidation="True"></asp:TextBox>

```



```

        </td>
</tr>--%>

<tr>
    <td align="right" class="label_txt"><strong>Role
:*</strong></td>

    <td width="25%" align="left">
        <asp:DropDownList ID="ddl_GroupId"
runat="server"></asp:DropDownList>
    </td>
</tr>
<%-- <tr>
    <td align="right" class="label_txt"><strong>Role
:*</strong></td>

    <td width="25%" align="left">
        <asp:DropDownList ID="ddl_RoleID" runat="server"
></asp:DropDownList>

    </td>
</tr>--%>
<tr>
    <td width="17%" align="right"
class="label_txt"><strong>Password:*</strong></td>
    <td width="25%" align="left">
        <asp:TextBox ID="txt_Password" runat="server"
MaxLength="40" CausesValidation="True" TextMode="Password"></asp:TextBox>
    </td>
</tr>

<tr>
    <td align="right"
class="label_txt">&nbsp;<strong>Password:*</strong></td>

    <td width="25%" align="left">
        <asp:TextBox ID="txt_Password1" runat="server"
MaxLength="40" CausesValidation="True" TextMode="Password"></asp:TextBox>
        <asp:TextBox ID="email_name" runat="server" MaxLength="50"
TextMode="Email" CausesValidation="True" Visible="false"></asp:TextBox>
    </td>
</tr>
</tbody>
</table>
</div>
</div>

</div>
<!-- /.content_holder -->
<div style="height: 0px; font-size: 1px; line-height: 1px; clear:
both;"></div>

</div>
</div>

</div>
</div>

```

```

        </div>
    </div>

    </div>
</div>
</div>
</asp:Content>

```

```

***** ManageUser.aspx.cs *****

```

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBAL;
using DowBEntity;
//using LogTracer;
using System.Data;

public partial class View_ManageUser : System.Web.UI.Page
{
    public string PageType;
    public string username;
    DDLogTracer objlog = new DDLogTracer();

    protected void Page_Load(object sender, EventArgs e)
    {
        ClsLoggedInUser ClsUser =
(ClsLoggedInUser)Session["LoggedUser_Specs"];
        if (ClsUser == null)
        {
            Response.Redirect("MLB_Login.aspx");
        }
        PageType = Page.Request.QueryString["PageType"];
        if (PageType == null)
        {
            PageType = "Add";
        }
        Session["PageType"] = PageType;
        EtetblUser Ist =new EtetblUser();
        string pageid = "4";// Request.QueryString["pageid"];
        ManageNavigations(Convert.ToInt32(pageid),
Convert.ToInt32(ClsUser.UserId), Convert.ToInt32(ClsUser.GroupId),PageType);

        if (!IsPostBack)
        {
            objlog.Trace_Log(0, "DateOrWed", "MangeUser", "Load", "Start
Here..");

            try
            {

                username = Page.Request.QueryString["username"];
                int userid =Convert.ToInt16( Page.Request.QueryString["UserId"]);

```

```

        PopDlIs ObjPD = new PopDlIs();
        //ObjPD.BindDropDowns(ddl_RoleID, (new PopDlIs()).GetRoles(ClsUser.RoleID));
        ObjPD.BindDropDowns(ddl_country, (new PopDlIs()).GetCountries());
        ObjPD.BindDropDowns(ddl_GroupId, (new
PopDlIs()).GetGroups(Convert.ToInt32(ClsUser.UserId),
Convert.ToInt32(ClsUser.GroupId)));
        if (PageType == "Edit" || PageType == "View")
        {
            if (username != null)
            {
                lst = (new tblUsers()).getuser(username);
            }
            else
            {
                lst = (new tblUsers()).getuserbyid(userid);
            }

            if (Convert.ToInt32(ClsUser.GroupId)==1 ||
lst.DealerId==Convert.ToInt32(ClsUser.DealerId ))
            {
                if (lst.GroupId >= Convert.ToInt32(ClsUser.GroupId))
                {
                    txt_username.Text = lst.UserId.ToString();
                    txt_username.Text = lst.UserName;
                    txt_FirstName.Text = lst.FirstName;
                    txt_LastName.Text = lst.LastName;
                    email_name.Text = lst.EmailID;
                    txt_mobile.Text = lst.ContactNum;
                    ddl_country.Text = lst.Region;
                    txt_mobile.Text = lst.ContactNum;
                    //ddl_RoleID.Text = lst.RoleID.ToString();
                    //ddl_DealerId.Text = lst.DealerId.ToString();
                    ddl_GroupId.Text = lst.GroupId.ToString();
                    txt_Password.Text = new BalCommon().EncryptPwd(lst.Password.ToString());
                    txt_Password1.Text = new BalCommon().EncryptPwd(lst.Password.ToString());
                    //ddl_GroupId.Enabled = false;
                    //ddl_DealerId.Enabled = false;
                    //ddl_RoleID.Enabled = false;

                }
                else
                {
                    Common.ShowMessageHtml("User can not
edit this user", 0, true, dvMsgBox);
                }
            }
            else
            {
                Common.ShowMessageHtml("User can not
edit this user", 0, true, dvMsgBox);
            }
        }
        else
        {

```

```

        txt_FirstName.Text = "";
        txt_LastName.Text = "";
        //email_name.Text = "";
        txt_mobile.Text = "";
        ddl_GroupId.Enabled = true;
        //ddl_DealerId.Enabled = true;
        //ddl_RoleID.Enabled = true;
        //chk_IsLoginAllowed.Checked = true;
        //if (ClsUser.DealerId > 1)
        //{
        //    ddl_DealerId.Text = Convert.ToString(ClsUser.DealerId);
        //    ddl_DealerId.Enabled = false;
        //}
        //else
        //{
        //    ddl_DealerId.Text = "";
        //}
    }
    Session["USERDATA"]=lst;
}
catch (Exception ex)
{
    Common.ShowMessageHtml(ex.Message, 1, true, dvMsgBox);
    ExceptionLogging.SendExcepToDB(ex);

    objlog.Trace_Error(5, "DateOrWed", "OrderSearchList", "Bindetails_Filter",
ex.Message);

}

objlog.Trace_Log(0, "DateOrWed", "MangeUser", "Load", "Ends
Here..");
}
}

private void ManageNavigations(int pageid, int userid, int groupid,string PageType)
{
    objlog.Trace_Log(0, "DateOrWed", "MangeUser", "ManageNavigations",
"Start Here..");
    try
    {
        bool IsLeftBar = false;
        IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid,
groupid, Table1, myBtnClickHandler);

        if (IsLeftBar == true)
        {
            LeftSideBar.Attributes["class"] = "span2";
            MainContent.Attributes["class"] = "span10";
            ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
//style="display: block;"
            if (PageType == "Add")

```

```

        {
            PageName.InnerText = "Add User";
        }
        else
        {
            PageName.InnerText = "Edit User";
        }
    }
    else
    {
        LeftSideBar.Attributes["class"] = "";
        MainContent.Attributes["class"] = "span12";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
    }
}
catch (Exception ex)
{
    Common.ShowMessageHtml(ex.Message, 1, true, dvMsgBox);
    ExceptionLogging.SendExcepToDB(ex);

    objlog.Trace_Error(5, "DateOrWed", "OrderSearchList", "Bindetails_Filter",
ex.Message);

}

objlog.Trace_Log(0, "DateOrWed", "MangeUser", "ManageNavigations",
"Ends Here..");
}

// This is Event Handler
protected void myBtnClickHandler(object sender, EventArgs e)
{
    objlog.Trace_Log(0, "DateOrWed", "MangeUser", "myBtnClickHandler",
"Starts Here..");
    try
    {
        LinkButton lnk = (LinkButton)sender;
        string ID = lnk.ID;

        DataSet ds =
Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
        if (ds != null)
        {
            if (ds.Tables[0].Rows.Count > 0)
            {
                DataTable dtdll = ds.Tables[0];
                foreach (DataRow drnew in dtdll.Rows)
                {
                    switch (drnew["lbtncode"].ToString())
                    {
                        case "SAVE":

```



```

        {
            //Common.ShowMessageHtml("Cancel Button clicked", 1, true,
dvMsgBox);
            Response.Redirect("~/UserList.aspx?pageid=153");
        }
        else if (PageType == "Add")
        {
            //Common.ShowMessageHtml("Cancel Button clicked -
Clearing Data Entry", 1, true, dvMsgBox);

            //ShowMessage("Cancel Button clicked - Clearing Data
Entry");
            txt_username.Text = "";
            txt_FirstName.Text = "";
            txt_LastName.Text = "";
            email_name.Text = "";
            txt_mobile.Text = "";
            ddl_GroupId.Enabled = true;
            //ddl_DealerId.Enabled = true;
            //ddl_RoleID.Enabled = true;
            //chk_IsLoginAllowed.Checked = true;
        }
    }
    catch (Exception exc)
    {
        objlog.Trace_Error(4, "DateOrWed", "MangeUser",
"ContentCancelControlClicked", "Exception:" + exc.ToString());
        Common.ShowMessageHtml(exc.ToString(), 0, true, dvMsgBox);
    }
    objlog.Trace_Log(0, "DateOrWed", "MangeUser",
"ContentCancelControlClicked", "Ends Here..");
}
private void ContentSaveControlClicked()
{
    objlog.Trace_Log(0, "DateOrWed", "MangeUser",
"ContentSaveControlClicked", "Start Here..");
    ClsLoggedInUser ClsUser =
(ClsLoggedInUser)Session["LoggedInUser_Specs"];

    try
    {
        if (BalCommon.IsValidEmail(txt_username.Text) == true)
        {
            bool chk = new tblUsers().UserNameExists(txt_username.Text);
            if (chk)
            {
                Common.ShowMessageHtml("User Name is Required.", 0, true, dvMsgBox);
            }
            email_name.Text = txt_username.Text;
        }
    }
}

```

```

        bool sresult;
        PageType = Convert.ToString(Session["PageType"]);
        if (ddl_GroupId.SelectedValue == "1" )
        {
            Common.ShowMessageHtml("Admin Group can not be
selected for Users", 0, true, dvMsgBox);
        }
        else if (txt_username.Text == "")
        {
            // ShowMessage( "User Name is Required");
            Common.ShowMessageHtml("User Name is Required.", 0,
true, dvMsgBox);
        }
        else if (txt_FirstName.Text == "")
        {
            //ShowMessage("Can not save First Name is Required");
            Common.ShowMessageHtml("First Name is Required.", 0,
true, dvMsgBox);
        }
        else if (txt_mobile.Text == "")
        {
            //ShowMessage("Can not save Mobile No is Required");
            Common.ShowMessageHtml("Mobile No is Required.", 0, true,
dvMsgBox);
        }
        else if (email_name.Text == "")
        {
            //ShowMessage("Can not save Email Id is Required");
            Common.ShowMessageHtml("Email Id is Required.", 0, true, dvMsgBox);
        }
        else if (txt_Password.Text == "")
        {
            //ShowMessage("Can not save password is Required");
            Common.ShowMessageHtml("Password is Required.", 0, true,
dvMsgBox);
        }
        else if (txt_Password.Text != txt_Password1.Text)
        {
            //ShowMessage("Can not save password is Required");
            Common.ShowMessageHtml("Passwords should be same.", 0, true, dvMsgBox);
        }
        else if (BalCommon.IsValidEmail(email_name.Text) == false)
        {
            Common.ShowMessageHtml("Valid Emailid is Required.", 0,
true, dvMsgBox);
        }
        else if (new FieldValidator().ValidateNumeric(txt_mobile)==false) {
            Common.ShowMessageHtml("Mobile No is Numeric only", 0,
true, dvMsgBox);
        }
        else if (new FieldValidator().ValidateNumeric(txt_landline) == false)

```



```
Response.Redirect("~/UserDetail.aspx?pageid=153&msg=User has been Created  
successfully.");
```

```
    }  
    else  
    {  
        Common.ShowMessageHtml("Create  
user failed.", 0, true, dvMsgBox);  
    }
```

```
    }  
    else  
    {  
        Common.ShowMessageHtml("User Already  
Exists.", 0, true, dvMsgBox);  
    }
```

```
    }  
    else  
    {  
        //_tbluser.UserId = Convert.ToInt32(txt_username.Text);  
        _tbluser.UpdatedOn = DateTime.Now;  
        if (_tbluser.UserId!=1)  
        {  
            sresult = (new tblUsers()).Update(_tbluser);  
        }  
        else  
        {  
            sresult = false;
```

```
Response.Redirect("~/UserList.aspx?pageid=153&msg=User Can not be edited.");
```

```
    }  
    if (sresult == true)  
    {  
        //ADD AUDIT TRAIL HERE  
        EtetblUser lst =  
(EtetblUser)Session["USERDATA"];  
  
        if(lst!=_tbluser)  
        {  
  
        }  
        //AUDIT TRAIL ENDS HERE  
        //ShowMessage("User has been edited  
successfully");  
        Common.ShowMessageHtml("User has been  
edited successfully.", 1, true, dvMsgBox);
```

```
Response.Redirect("~/UserList.aspx?pageid=153&msg=User has been edited  
successfully.");
```



```

jQuery.ajax({
    type: "POST",
    // data: "{}",
    data: "{GroupId:" + $("#CPH_MainContent_ddlper :selected").val() + "}",
    contentType: "application/json; charset=utf-8",
    dataType: "json",
    async: false,

    url: "ManageRoles.aspx/GetPerSetsForGroup",

    success: function (msg) {
        $("#ultopnav").html(msg.d);
    },
    error: function (msg) {
        $(document.body).html(msg.responseText);
    }
});
$("#div_msg").html("");
}

```

```
function GetPerSets() {
```

```

    jQuery.ajax({
        type: "POST",
        data: "{}",

        contentType: "application/json; charset=utf-8",
        dataType: "json",
        async: false,
        url: "ManageRoles.aspx/ChkPermissions",

        success: function (msg) {
            $('#DivPer').html(msg.d);

        },
        error: function (msg) {
            $(document.body).html(msg.responseText);
        }
    });
}

```

```
function chkPriv(txt) {
```

```

    if (txt == "AccessTosite") {

        if ($("#hdenSiteacc").val() == "True") {

            return true;

```

```

    }
    else {
        return false;
    }
}

if (txt == "View") {

    if ($("#hdenPerSetView").val() == txt) {

        return true;
    }
    else {
        return false;
    }
}
if (txt == "Edit") {
    if ($("#hdenPerSetEdit").val() == txt) {
        return true;
    }
    else {
        return false;
    }
}
if (txt == "Add") {

    if ($("#hdenPerSetADD").val() == txt) {
        return true;
    }
    else {
        return false;
    }
}
if (txt == "Delete") {

    if ($("#hdenPerSetDelete").val() == txt) {
        return true;
    }
    else {
        return false;
    }
}
}

```

```

function NavtoNoAccessPage() {

```

```

    GetPerSets();
    if (chkPriv('AccessTosite')) {

        if (!chkPriv('View')) {

```

```

        window.location.href = "NoPermission.aspx?PageName=" + "Edit
Roles&RedirectFlag=2 ";
        return false;
    }
    else {
        return true;
    }
}
else {
    window.location.href = "NoPermission.aspx?PageName=" + "Access Denied for the
Site. Plaese Contact Admin&RedirectFlag=1 ";
    return false;
}
}

```

```

function sendRegistration() {
    if (NavtoNoAccessPage()) {

        if (!chkPriv('Edit')) {
            alert('You donot have permissions to Edit Roles. Please contact Admin');
            return false;
        }
    }
    else {

        return false;
    }
    var rows = [];
    var ulld;
    var _gid, _add, _edit, _del, _view;
    var linenumber;
    $('[id^="ul_"]').each(function (i, n) {
        ulld = $(this).attr("id");
        ulld = ulld.substr(ulld.indexOf('_') + 1);

        _gid = ulld

        _view = $('#ChkView_' + _gid).attr("checked");
        _add = $('#ChkAdd_' + _gid).attr("checked");
        _edit = $('#ChkEdit_' + _gid).attr("checked");
        _del = $('#ChkDelete_' + _gid).attr("checked");
        rows.push({
            GroupID: _gid,
            Add: _add,
            View: _view,
            Edit: _edit,
            Delete: _del
        });
    });
    //alert(JSON.stringify(rows));
    $.ajax({

```

```

type: "POST",
url: "ManageRoles.aspx/UpdateXml",
data: "{ 'name': '" + JSON.stringify(rows) + "', grpId: '" + $('#CPH_MainContent_ddlper
:selected').val() + "'" },
contentType: "application/json; charset=utf-8",
dataType: "json",
success: function (html, status) {

    if (status) {
        //window.location.href = 'assignRoles.aspx';
        //$("#div_msg").html("Successfully Updated");
        alert("Successfully Updated");
    }
}

});

}
</script>
<div class="mid_block">
<h1><span>Assign ROLES</span></h1>
<div id="mid_panel">

<div class="mid_main_container">

<h2>Roles / permission groups</h2>
<hr style="background-color:#999">
<div class="lft_pad_block">
<table width="100%" border="0" cellspacing="0" cellpadding="0"
class="form_tab">
<tr>
<td width="20%" align="right" >Permission Group</td>
<td width="2%" align="center" >:</td>
<td width="78%"> <asp:DropDownList CssClass="drop_small width1"
ID="ddlper" runat="server" onchange="changePer();">

</asp:DropDownList></td>
</tr>
</table>
<ul id="ultopnav" class="topnav">

</ul>
<div id="div_msg"> </div>
<div class="actions mar_top">
<ul>
<li class="flt"><a href="ViewUsers.aspx"></a></li>
<li class="flt"><a href="#"></a></li>

</ul>

```

```
        <input id="hdnpriv" runat="server" type="hidden" value="1" />
    </div>
</div>
</div>
</div>
</div>
```

```
    <div id="DivPer"></div>
</asp:Content>
```

```
*****ManageRoles.aspx.cs *****
```

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.IO;
using System.Data;
using System.Text;
using DowBAL;
using System.Xml;
using DowBEntity;
public partial class ManageRoles : System.Web.UI.Page
{
    public static string Ids = ",";
    public static string ParentIds = ",";
    public static bool HasEditPriv = true;
    public static string FileName = string.Empty;
    protected void Page_Load(object sender, EventArgs e)
    {
        if (!IsPostBack)
        {
            DataSet dsPer = null;
            DowBAL.BALCreateUser Buser = new DowBAL.BALCreateUser();
            dsPer = Buser.GetAllPersets();
            ddlper.DataSource = dsPer;
            ddlper.DataTextField = "GroupName";
            ddlper.DataValueField = "GroupId";
            ddlper.DataBind();

            GetPerSetsForGroup(ddlper.SelectedValue.ToString());

            string AbsPath = Request.Url.AbsolutePath;
            FileName = Path.GetFileName(AbsPath);
        }
        if (!User.IsInRole("Edit"))
        {
            HasEditPriv = false;
            hdnpriv.Value = "0";
        }
        else
    }
}
```



```

    {
        HasEditPriv = true;
        hdnpriv.Value = "1";
    }
}
[System.Web.Services.WebMethod]
public static string ChkPermissions()
{
    return
Common.GetPermissionSets(FileName.TrimEnd(".").ToCharArray()).TrimEnd(".").ToCharArra
y());
}

[System.Web.Services.WebMethod]
public static string GetPerSetsForGroup(string GroupId)
{
    if (GroupId != null && GroupId == "" && GroupId=="undefined")
    {
        ClsPermissioGroup objclsper = new ClsPermissioGroup();
        DataTable dtGetPerSetsForGroup =
objclsper.PopulateDynamicTree(Convert.ToInt32(GroupId));

        StringBuilder sbuilder = new StringBuilder();
        DataRow[] parentMenus = dtGetPerSetsForGroup.Select("ParentId = 0");

        var sb = new StringBuilder();

        return GenerateUL(parentMenus, dtGetPerSetsForGroup, sb, 0);
    }
    else { return null; }
}
private static string GenerateUL(DataRow[] menu, DataTable table, StringBuilder sb, int
flag)
{
    string isdisabled = string.Empty;
    if (!HasEditPriv)
    {
        isdisabled = "disabled ='disabled'";
    }

    if (menu.Length > 0)
    {
        foreach (DataRow dr in menu)
        {
            string handler = dr["PageUrl"].ToString();
            string menuText = dr["PageName"].ToString();
            string level = dr["level"].ToString();
            string clscheckbox = "form_acording";
            string pid = dr["Pageld"].ToString();
            string line = string.Empty;

```

```

        if (menuText == "User Manager")
        {
            handler = handler.ToString();
        }

        if (dr["level"].ToString() == "0")
        {
            line = String.Format(@" <li class='bg1'><a href='#' class='bg'>{1}</a>", handler,
menuText);
        }
        if (dr["level"].ToString() == "1")
        {
            line = String.Format(@" <li class='bg2'><a href='#' class='bg2'>{1}</a>",
handler, menuText);
            clschkbox = "form_acording sub";
        }

        if (Convert.ToInt32(dr["level"].ToString()) >= 2)
        {
            line = String.Format(@" <li class='bg3'><a href='#' class='bg3'>{1}</a>",
handler, menuText);
            clschkbox = "form_acording sub_2";
        }
        sb.Append(line);
        DataRow[] subMenu = table.Select(String.Format("ParentId = {0}", pid));
        if (subMenu.Length > 0)
        {
            if (handler.ToLower() != "no-url")
            {
                string clsvchked = string.Empty;
                if (dr["view"].ToString().ToLower() == "true")
                {
                    clsvchked = "checked='checked'";
                }
                else
                {
                    clsvchked = "";
                }

                string clsechked = string.Empty;
                if (dr["edit"].ToString().ToLower() == "true")
                {
                    clsechked = "checked='checked'";
                }
                else
                {
                    clsechked = "";
                }

                string clsachked = string.Empty;
                if (dr["add"].ToString().ToLower() == "true")
                {
                    clsachked = "checked='checked'";
                }
                else
            }

```

```

    {
        clsachked = "";
    }

    string clsdchked = string.Empty;
    if (dr["delete"].ToString().ToLower() == "true")
    {
        clsdchked = "checked='checked'";
    }
    else
    {
        clsdchked = "";
    }
    sb.Append("<div id='ul_' + pid + '>");
    sb.Append("<div class='form_acording position'>");
    sb.Append("<label><input type='checkbox' " + isdisabled + " " + clsvchked + "
name='ChkView_' + pid + " runat='server' id='ChkView_' + pid + " />View</label>");
    sb.Append("<label><input type='checkbox' " + isdisabled + " " + clsachked + "
name='ChkAdd_' + pid + " runat='server' id='ChkAdd_' + pid + " />Add</label>");
    sb.Append("<label><input type='checkbox' " + isdisabled + " " + clsechked + "
name='ChkEdit_' + pid + " runat='server' id='ChkEdit_' + pid + " />Edit</label>");
    sb.Append("<label><input type='checkbox' " + isdisabled + " " + clsdchked + "
name='ChkDelete_' + pid + " runat='server' id='ChkDelete_' + pid + " />Delete</label>");
    sb.Append("</div>");
    sb.Append("</div>");

}
var subMenuBuilder = new StringBuilder();

if (level != "0")
{
    string concat = pid + " " + "-" + " " + level;
    DataRow[] drn = table.Select("concatenador LIKE '%" + concat + "%'");
    Ids = Ids + drn[drn.Length - 1]["Pageld"].ToString() + ",";
    sb.Append(" <ul class='no_mar'>");
}
else
{
    string concat = pid + " " + "-" + " " + level;
    DataRow[] drn = table.Select("concatenador LIKE '%" + concat + "%'");
    ParentIds = ParentIds + drn[drn.Length - 1]["Pageld"].ToString() + ",";
    sb.Append(" <ul class='no_mar'>");
}

if (ParentIds.Contains(", " + pid.ToString() + ", "))
{
    sb.Append(GenerateUL(subMenu, table, subMenuBuilder, 1));
}
else
{
    sb.Append(GenerateUL(subMenu, table, subMenuBuilder, 0));
}
}

```

```

else
{
    if (handler.ToLower() != "no-url")
    {
        string clsvchked = string.Empty;
        if (dr["view"].ToString().ToLower() == "true")
        {
            clsvchked = "checked='checked'";
        }
        else
        {
            clsvchked = "";
        }

        string clsechked = string.Empty;
        if (dr["edit"].ToString().ToLower() == "true")
        {
            clsechked = "checked='checked'";
        }
        else
        {
            clsechked = "";
        }

        string clsachked = string.Empty;
        if (dr["add"].ToString().ToLower() == "true")
        {
            clsachked = "checked='checked'";
        }
        else
        {
            clsachked = "";
        }

        string clsdchked = string.Empty;
        if (dr["delete"].ToString().ToLower() == "true")
        {
            clsdchked = "checked='checked'";
        }
        else
        {
            clsdchked = "";
        }

        sb.Append("<ul id='ul_' + pid + '>");
        sb.Append("<li>");
        sb.Append("<div class='" + clschkbox + "'>");
        sb.Append("<label><input type='checkbox' " + isdisabled + " " + clsvchked + " " +
name='ChkView_' + pid + " runat='server' id='ChkView_' + pid + " />View</label>");
        sb.Append("<label><input type='checkbox' " + isdisabled + " " + clsachked + " " +
name='ChkAdd_' + pid + " runat='server' id='ChkAdd_' + pid + " />Add</label>");
        sb.Append("<label><input type='checkbox' " + isdisabled + " " + clsechked +
" name='ChkEdit_' + pid + " runat='server' id='ChkEdit_' + pid + " />Edit</label>");

```

```

        sb.Append("<label><input type='checkbox' " + isdisabled + " " + clsdchked + "
name='ChkDelete_" + pid + "' runat='server' id='ChkDelete_" + pid + "' />Delete</label>");
        sb.Append(" </div>");
        sb.Append("</li>");
        sb.Append("</ul>");
    }
}
sb.Append("</li>");

if (Ids.Contains(", " + pid.ToString() + ",") || flag == 1)
{
    sb.Append("</ul>");
}
}
}

return sb.ToString();
}

```

```

[System.Web.Services.WebMethod()]
public static bool UpdateXml(string name, string grpId)
{
    try
    {
        string Data = "{/";
        Data += "root/";
        Data += ":{";
        Data += "/Data/";
        Data += ":";

        Data += name + "}}";

        Data = Data.Replace('/', "");

        XmlDocument xml = Newtonsoft.Json.JsonConvert.DeserializeXmlNode(Data);

        ClsPermissioGroup clsper = new ClsPermissioGroup();

        ClsLoggedInUser Clsusr =
(ClsLoggedInUser)HttpContext.Current.Session["LoggedInUser_Specs"];

        bool status = clsper.UpdatePermissionGrp(xml, Convert.ToInt32(grpId),
Convert.ToInt32(Clsusr.UserId), Clsusr.LoginID);
        if (status)
        {
            return true;
        }
        else
        {
            return false;
        }
    }
    catch (Exception ex)

```

```

    {
        return false;
    }
}

}

}
*****ManageSubgroup.aspx.cs *****
<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="ManageSubgroup.aspx.cs" Inherits="ManageSubgroup"
%>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
    <script type="text/javascript">
        function checkTextAreaMaxLength(textBox, e, length) {

            var mLen = textBox["MaxLength"];
            if (null == mLen)
                mLen = length;

            var maxLength = parseInt(mLen);
            if (!checkSpecialKeys(e)) {
                if (textBox.value.length > maxLength - 1) {
                    if (window.event)//IE
                        e.returnValue = false;
                    else
                        e.preventDefault();
                }
            }
        }

        function isNumberKey(evt) {
            var charCode = (evt.which) ? evt.which : event.keyCode;
            if (charCode != 46 && charCode > 31 && (charCode < 48 || charCode > 57)) {
                return false;
            } else {
                // If the number field already has . then don't allow to enter . again.
                if (evt.target.value.search(/\./) > -1 && charCode == 46) {
                    return false;
                }
                return true;
            }
        }
    </script>
</asp:Content>

<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
    <div class="container-fluid">
        <div class="row-fluid" style="padding-top: 20px;">
            <div class="span2" id="dvlogo" runat="server" style="display: block;">
                <%----%>
            </div>

```



```

using System.Web.UI;
using System.Web.UI.WebControls;
using System.IO;
using System.Data;
using System.Text;
using DowBAL;
using System.Xml;
using DowBEntity;
public partial class ManageSubgroup : System.Web.UI.Page
{
    public string PageType;
    string pageid;
    int Id;
    protected void Page_Load(object sender, EventArgs e)
    {
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr != null)
        {
            PageType = Page.Request.QueryString["PageType"];
            if (PageType == null)
            {
                PageType = "Add";
            }
            Session["PageType"] = PageType;
            pageid = "4";// Request.QueryString["pageid"];
            ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
            Convert.ToInt32(Clsusr.GroupId), PageType);
        }
        PopDlls ObjPD = new PopDlls();
        ObjPD.BindDropDowns(ddl_productgroup, (new PopDlls()).GetProductCats());

        if (!IsPostBack)
        {
            if (PageType == "Edit")
            {
                Id = Convert.ToInt32(Request.QueryString["Id"]);
                EteProductGroup Hold = new EteProductGroup();
                Hold = new Products().GetProductGroupbyId("", Id);
                if (Hold != null)
                {
                    ddl_productgroup.SelectedIndex =(int) Hold.PCatId;
                    Details.Text = Hold.pgrouppname;
                    Details1.Text = Hold.pgrouppdescription;
                }
                else
                {
                    Common.ShowMessageHtml("Missing Tasks Details ", 0, true, dvMsgBox);
                }
            }
        }
    }
    private void ManageNavigations(int pageid, int userid, int groupid, string PageType)

```

```

{
    bool IsLeftBar = false;
    IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
myBtnClickHandler);

    if (IsLeftBar == true)
    {
        LeftSideBar.Attributes["class"] = "span2";
        MainContent.Attributes["class"] = "span10";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;"); //style="display:
block;"
        PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
    else
    {
        LeftSideBar.Attributes["class"] = "";
        MainContent.Attributes["class"] = "span12";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
    }
}
}

```

// This is Event Handler

```

protected void myBtnClickHandler(object sender, EventArgs e)
{
    LinkButton lnk = (LinkButton)sender;
    string ID = lnk.ID;

    DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
    if (ds != null)
    {
        if (ds.Tables[0].Rows.Count > 0)
        {
            DataTable dtdll = ds.Tables[0];
            foreach (DataRow drnew in dtdll.Rows)
            {
                switch (drnew["lbtncode"].ToString())
                {
                    case "SAVE":
                        // Call SAVE Method
                        ContentSaveControlClicked();
                        break;
                    case "AUDIT":
                        Response.Redirect(drnew["pageurl"].ToString() + "&frompage=" +
Request.QueryString["pageid"]);
                        break;
                    case "URL":
                        Response.Redirect(drnew["pageurl"].ToString());
                        break;
                    case "CANCEL":
                        ContentCancelControlClicked();
                        break;
                }
            }
        }
    }
}

```

```

        default:
            break;
        }
    }
}

private void ContentCancelControlClicked()
{
    Common.ShowMessageHtml("", 0, false, dvMsgBox);

    PageType = Session["PageType"].ToString();
    if (PageType == "Add")
    {
    }
    else if (PageType == "Edit")
    {
        Response.Redirect("~/SubGroupsList.aspx?pageid=8");
    }
}

private void ContentSaveControlClicked()
{
    //objlog.Trace_Log(0, "DateOrWed", "MangeDealer", "SaveClick", "Start Here..");

    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    try
    {
        if (Details.Text == "")
        {
            Common.ShowMessageHtml("Name cannot be blank", 0, true, dvMsgBox);

        }
        else if (Details1.Text == "")
        {
            Common.ShowMessageHtml("Description cannot be blank", 0, true, dvMsgBox);

        }
        else
        {
            EteProductGroup ETask = new EteProductGroup();
            ETask.pgid = Convert.ToInt32(Request.QueryString["Id"]);
            ETask.pgroupname = Details.Text;
            ETask.PCatId = ddl_productgroup.SelectedIndex;
            ETask.pgroupdescription = Details1.Text;
            ETask.createdOn = DateTime.Now;
        }
    }
}

```



```
<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="ManageUOMTypes.aspx.cs"
Inherits="ManageUOMTypes" %>
```

```
<%@ Register Assembly="Infragistics4.Web.v14.2, Version=14.2.20142.2480,
Culture=neutral, PublicKeyToken=7dd5c3163f2cd0cb"
Namespace="Infragistics.Web.UI.GridControls" TagPrefix="ig" %>
```

```
<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
```

```
<script type="text/javascript">
```

```
function checkTextAreaMaxLength(textBox, e, length) {
```

```
    var mLen = textBox["MaxLength"];
```

```
    if (null == mLen)
```

```
        mLen = length;
```

```
    var maxLength = parseInt(mLen);
```

```
    if (!checkSpecialKeys(e)) {
```

```
        if (textBox.value.length > maxLength - 1) {
```

```
            if (window.event)//IE
```

```
                e.returnValue = false;
```

```
            else
```

```
                e.preventDefault();
```

```
        }
```

```
    }
```

```
}
```

```
function isNumberKey(evt) {
```

```
    var charCode = (evt.which) ? evt.which : event.keyCode;
```

```
    if (charCode != 46 && charCode > 31 && (charCode < 48 || charCode > 57)) {
```

```
        return false;
```

```
    } else {
```

```
        // If the number field already has . then don't allow to enter . again.
```

```
        if (evt.target.value.search(/\./) > -1 && charCode == 46) {
```

```
            return false;
```

```
        }
```

```
        return true;
```

```
    }
```

```
}
```

```
</script>
```

```
</asp:Content>
```

```
<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
```

```
<div class="container-fluid">
```

```
<div class="row-fluid" style="padding-top: 20px;">
```

```
<div class="span2" id="dvlogo" runat="server" style="display: block;">
```

```
<!---->
```

```
</div>
```

```
<div class="span10">
```

```
<div class="row">
```

```
<div id="dvMsgBox" class="span12 ordersave" runat="server" style="display:
none;">
```



```

public string PageType;
string pageid;
int Id;
protected void Page_Load(object sender, EventArgs e)
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    if (Clsusr != null)
    {
        PageType = Page.Request.QueryString["PageType"];
        if (PageType == null)
        {
            PageType = "Add";
        }
        Session["PageType"] = PageType;
        pageid = "4";// Request.QueryString["pageid"];
        ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
Convert.ToInt32(Clsusr.GroupId), PageType);
    }
    if (!IsPostBack)
    {
        if (PageType=="Edit")
        {
            Id =Convert.ToInt32( Request.QueryString["Id"]);
            Etetbluom Hold = new Etetbluom();
            Hold = new Products().GetProducttbluombyId(Id);
            if (Hold != null)
            {
                txt_holdname.Text = Hold.UomShort;
                Details.Text = Hold.UomDescription;
            }
            else
            {
                Common.ShowMessageHtml("Missing Tasks Details ", 0, true, dvMsgBox);
            }
        }
    }
}

private void ManageNavigations(int pageid, int userid, int groupid, string PageType)
{
    bool IsLeftBar = false;
    IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
myBtnClickHandler);

    if (IsLeftBar == true)
    {
        LeftSideBar.Attributes["class"] = "span2";
        MainContent.Attributes["class"] = "span10";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;"); //style="display:
block;"
        PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
}

```



```

    }
    else
    {
        LeftSideBar.Attributes["class"] = "";
        MainContent.Attributes["class"] = "span12";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
    }
}

// This is Event Handler
protected void myBtnClickHandler(object sender, EventArgs e)
{
    LinkButton lnk = (LinkButton)sender;
    string ID = lnk.ID;

    DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
    if (ds != null)
    {
        if (ds.Tables[0].Rows.Count > 0)
        {
            DataTable dtdll = ds.Tables[0];
            foreach (DataRow drnew in dtdll.Rows)
            {
                switch (drnew["lbtncode"].ToString())
                {
                    case "SAVE":
                        // Call SAVE Method
                        ContentSaveControlClicked();
                        break;
                    case "AUDIT":
                        Response.Redirect(drnew["pageurl"].ToString() + "&frompage=" +
Request.QueryString["pageid"]);
                        break;
                    case "URL":
                        Response.Redirect(drnew["pageurl"].ToString());
                        break;
                    case "CANCEL":
                        ContentCancelControlClicked();
                        break;
                    default:
                        break;
                }
            }
        }
    }
}
}

```

```

private void ContentCancelControlClicked()
{
    Common.ShowMessageHtml("", 0, false, dvMsgBox);

    PageType = Session["PageType"].ToString();
    if (PageType == "Add")
    {

    }
    else if (PageType == "Edit")
    {
        Response.Redirect("~/UOMTypesList.aspx");
    }
}
private void ContentSaveControlClicked()
{
    //objlog.Trace_Log(0, "DateOrWed", "MangeDealer", "SaveClick", "Start Here..");

    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    try
    {
        if (txt_holdname.Text == "")
        {
            Common.ShowMessageHtml("Name cannot be blank", 0, true, dvMsgBox);

        }
        else if (Details.Text == "")
        {

            Common.ShowMessageHtml("Description cannot be blank", 0, true, dvMsgBox);

        }
    }
    else
    {
        Etetbluom ETask = new Etetbluom();
        ETask.UomId = Convert.ToInt32(Request.QueryString["Id"]);
        ETask.UomShort = txt_holdname.Text;
        ETask.CreatedOn = DateTime.Now;
        ETask.UserId = Convert.ToInt32(Clsusr.UserId);

        ETask.UomDescription = Details.Text;

        bool Result = false;
        PageType = Session["PageType"].ToString();

        if (PageType == "Add")
        {
            Result = new Products().InsertUOM(ETask);
        }
        else
        {
            Result = new Products().UpdateUOM(ETask);
        }
    }
}

```

```

        if (Result == true)
        {
            if (PageType == "Add")
            {
                Common.ShowMessageHtml("Adding of UOM Type Sucessful ", 1, true,
dvMsgBox);
            }
            else
            {
                Common.ShowMessageHtml("UOM Type Edited Sucessfully ", 1, true,
dvMsgBox);
            }
            Response.Redirect("~/UOMTypesList.aspx?msg=Note Type Saved
Sucessfully");
        }
    }
}

catch (Exception exc)
{
    //objlog.Trace_Error(4, "DateOrWed", "MangeDealer", "Load", "Exception:" +
exc.ToString());
    Common.ShowMessageHtml(exc.ToString(), 0, true, dvMsgBox);
}
//objlog.Trace_Log(0, "DateOrWed", "MangeDealer", "SaveClick", "Start Here..");
}
}
}

```

```

}
***** MyCart.aspx*****
<%@ Page Title="" Language="C#" MasterPageFile="~/ShoppingMasterPage.master"
AutoEventWireup="true" CodeFile="MyCart.aspx.cs" Inherits="MyCart" %>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentMainPage" runat="Server">

<div class="span10" id="MainContent" runat="server">
<div id="ShowPageHeader" class="row" style="display: block;" runat="server">
<div id="PageName" class="span12 customer" runat="server">
</div>
</div>

```

```

<div id="Div1" class="row" style="display: block;">

<!-- Content Bodey Here -->

<section class="main-contain bg-white">
  <div id="CartAmt" runat="server" class="container">
    <div class="shopping-cart-block">
      <ul class="title clearfix">
        <li class="col-md-2 col-sm-2 col-xs-6">Image</li>
        <li class="col-md-3 col-sm-3 col-xs-6">Product Name</li>
        <li class="col-md-2 col-sm-2 col-xs-6">Quantity</li>
        <li class="col-md-2 col-sm-2 col-xs-6">Unite Price</li>
        <li class="col-md-2 col-sm-2 col-xs-6">Sub Total</li>
        <li class="col-md-1 col-sm-1 col-xs-6">Action</li>
      </ul>
      <ul class="shopping-cart-cd clearfix">
        <li class="col-md-2 col-sm-2 col-xs-4"><figure></figure></li>
        <li class="col-md-3 col-sm-3 col-xs-6 cd-shopping">
          <h6>Summer Dress</h6>
          <p><span>Color:</span> Brown</p>
          <p><span>Size:</span> 12</p>
        </li>
        <li class="col-md-2 col-sm-2 col-xs-4">
          <button type='button' class="quantity-button" name='subtract'
onclick='javascript: subtractQty1();' value='- '></button>
          <input type='text' class="quantity-field" name='qty1' value="5"
id='qty1' />
          <button type='button' class="quantity-button" name='add'
onclick='javascript: document.getElementById("qty1").value++;' value='+ '>+</button>
        </li>
        <li class="col-md-2 col-sm-2 col-xs-4">$520.00</li>
        <li class="col-md-2 col-sm-2 col-xs-6">$520.00</li>
        <li class="col-md-1 col-sm-1 col-xs-6"><a href="#">X</a></li>
      </ul>
      <ul class="shopping-cart-cd clearfix">
        <li class="col-md-2 col-sm-2 col-xs-4"><figure></figure></li>
        <li class="col-md-3 col-sm-3 col-xs-4 cd-shopping">
          <h6>Summer Dress</h6>
          <p><span>Color:</span> Brown</p>
          <p><span>Size:</span> 12</p>
        </li>
        <li class="col-md-2 col-sm-2 col-xs-4">
          <button type='button' class="quantity-button" name='subtract'
onclick='javascript: subtractQty3();' value='- '></button>
          <input type='text' class="quantity-field" name='qty3' value="1"
id='qty3' />
          <button type='button' class="quantity-button" name='add'
onclick='javascript: document.getElementById("qty3").value++;' value='+ '>+</button>
        </li>
        <li class="col-md-2 col-sm-2 col-xs-4">$520.00</li>
        <li class="col-md-2 col-sm-2 col-xs-4">$520.00</li>
        <li class="col-md-1 col-sm-1 col-xs-4"><a href="#">X</a></li>
      </ul>

```

```

        <ul class="shopping-cart-cd clearfix">
            <li class="col-md-2 col-sm-2 col-xs-4"><figure></figure></li>
            <li class="col-md-3 col-sm-3 col-xs-4 cd-shopping">
                <h6>Summer Dress</h6>
                <p><span>Color:</span> Brown</p>
                <p><span>Size:</span> 12</p>
            </li>
            <li class="col-md-2 col-sm-2 col-xs-4">
                <button type='button' class="quantity-button" name='subtract'
onclick='javascript: subtractQty4();' value='- '></button>
                <input type='text' class="quantity-field" name='qty4' value="7"
id='qty4' />
                <button type='button' class="quantity-button" name='add'
onclick='javascript: document.getElementById("qty4").value++;' value='+'>+</button>
            </li>
            <li class="col-md-2 col-sm-2 col-xs-4">$520.00</li>
            <li class="col-md-2 col-sm-2 col-xs-4">$520.00</li>
            <li class="col-md-1 col-sm-1 col-xs-2"><a href="#">X</a></li>
        </ul>
    </div>

```

```

<div class="shopping-cart-checkout">

```

```

    <div class="coupon-code">
        <div class="row">
            <div class="col-sm-4 sm-margin-bottom-30">
                <h3>Discount Code</h3>
                <p>Enter your coupon code</p>
                <input class="form-control margin-bottom-10" name="code" type="text">
                <button type="button" class="btn-check checkout-btn">Apply
Coupon</button>
            </div>
            <div class="col-sm-3 col-sm-offset-5">
                <ul class="list-inline total-result">
                    <li>
                        <h4>Subtotal:</h4>
                        <div class="total-result-in">
                            <span>$ 1280.00</span>
                        </div>
                    </li>
                    <li>
                        <h4>Shipping:</h4>
                        <div class="total-result-in">
                            <span class="text-right">- - - </span>
                        </div>
                    </li>
                    <li class="divider"></li>
                    <li class="total-price">
                        <h4>Total:</h4>
                        <div class="total-result-in">
                            <span>$ 1280.00</span>
                        </div>
                    </li>
                </ul>
            </div>
        </div>
    </div>

```



```

        Response.Redirect("~/MyCart.aspx");
    }
}
string pageid = "50";// Request.QueryString["pageid"];
ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
Convert.ToInt32(Clsusr.GroupId));
string msg = Request.QueryString["msg"];
if (msg != null)
{
    //Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
}
Session["Amt"] = "0";

Bindetails();

}
else
{
    Response.Redirect("MLB_Login.aspx");
}
}

private void ManageNavigations(int pageid, int userid, int groupid)
{
    //bool IsLeftBar = false;
    //IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
myBtnClickHandler);

    //if (IsLeftBar == true)
    //{
    //    LeftSideBar.Attributes["class"] = "span2";
    //    MainContent.Attributes["class"] = "span10";
    //    ShowPageHeader.Attributes.CssStyle.Add("display", "block;"); //style="display:
block;"
    //    PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    //}
    //else
    //{
    //    LeftSideBar.Attributes["class"] = "";
    //    MainContent.Attributes["class"] = "span12";
    //    ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
    //    PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    //}
}

// This is Event Handler
protected void myBtnClickHandler(object sender, EventArgs e)
{
    LinkButton Ink = (LinkButton)sender;
    string ID = Ink.ID;

```



```

sb.Append(" <li class='col-md-2 col-sm-2 col-xs-6'>Image</li> \n");
sb.Append(" <li class='col-md-3 col-sm-3 col-xs-6'>Product Name</li> \n");
sb.Append(" <li class='col-md-2 col-sm-2 col-xs-6'>Quantity</li>\n");
sb.Append(" <li class='col-md-2 col-sm-2 col-xs-6'>Unite Price</li> \n");
sb.Append(" <li class='col-md-2 col-sm-2 col-xs-6'>Sub Total</li> \n");
sb.Append(" <li class='col-md-1 col-sm-1 col-xs-6'>Action</li> \n");
sb.Append(" </ul> \n");
int sno = 0;
foreach (var list in Lst)
{
    sno++;
    sb.Append(" <ul class='shopping-cart-cd clearfix'> \n");
    sb.Append(" <li class='col-md-2 col-sm-2 col-xs-6'><figure><img src="" +
list.ImagePath + "" alt="" /></figure></li> \n");
    sb.Append(" <li class='col-md-3 col-sm-3 col-xs-6 cd-shopping'>
\n");
    sb.Append(" <h6>" + list.SellerDescription + "</h6> \n");
    sb.Append(" <p><span>Color:</span> Brown</p> \n");
    sb.Append(" <p><span>Size:</span> 12</p> \n");
    sb.Append(" </li> \n");
    sb.Append(" <li class='col-md-2 col-sm-2 col-xs-6'> \n");
    sb.Append(" <button type='button' class='quantity-button' name='subtract'
onclick='javascript: subtractQty1();' value='- '></button> \n");
    sb.Append(" <input type='text' class='quantity-field' name='qty'" + sno + ""
value="" + list.Qty + "" id='qty'" + sno + "" /> \n");
    sb.Append(" <button type='button' class='quantity-button' name='add'
onclick='javascript: document.getElementById('qty'" + sno + ").value++;'
value='+'></button> \n");
    sb.Append(" </li> \n");
    sb.Append(" <li class='col-md-2 col-sm-2 col-xs-6'>Rs." + list.Price + "</li>
\n");
    sb.Append(" <li class='col-md-2 col-sm-2 col-xs-6'>Rs." + list.Price + "</li>
\n");
    sb.Append(" <li class='col-md-1 col-sm-1 col-xs-6'><a href='#>X</a></li>
\n");
    sb.Append(" </ul> \n");
    price = price + list.Price;
}
sb.Append(" </div> \n");
sb.Append(" <div class='shopping-cart-checkout'> \n");
sb.Append(" <div class='coupon-code'> \n");
sb.Append(" <div class='row'> \n");
sb.Append(" <div class='col-sm-4 sm-margin-bottom-30'> \n");
sb.Append(" <h3>Discount Code</h3> \n");
sb.Append(" <p>Enter your coupon code</p> \n");
sb.Append(" <input class='form-control margin-bottom-10' name='code'
type='text'> \n");
sb.Append(" <button type='button' class='btn-check checkout-btn'>Apply
Coupon</button> \n");
sb.Append(" </div> \n");
sb.Append(" <div class='col-sm-3 col-sm-offset-5'> \n");
sb.Append(" <ul class='list-inline total-result'> \n");
sb.Append(" <li> \n");
sb.Append(" <h4>Subtotal:</h4> \n");
sb.Append(" <div class='total-result-in'> \n");

```

```

sb.Append(" <span>"+price+"</span> \n");
sb.Append(" </div> \n");
sb.Append(" </li> \n");
sb.Append(" <li> \n");
sb.Append(" <h4>Shipping:</h4> \n");
sb.Append(" <div class='total-result-in'> \n");
sb.Append(" <span class='text-right'>- - -</span> \n");
sb.Append(" </div> \n");
sb.Append(" </li> \n");
sb.Append(" <li class='divider'></li> \n");
sb.Append(" <li class='total-price'>\n");
sb.Append(" <h4>Total:</h4> \n");
sb.Append(" <div class='total-result-in'> \n");
sb.Append(" <span>"+price+"</span> \n");
sb.Append(" </div> \n");
sb.Append(" </li> \n");
sb.Append(" </ul> \n");
sb.Append(" <div class='mb30'></div> \n");
sb.Append(" <a href='#' class='btn btn-product pull-right'>Checkout</a>\n");
sb.Append(" </div> \n");
sb.Append(" </div> \n");
sb.Append(" </div> \n");
sb.Append(" </div> \n");

```

```

PreHtml = sb.ToString();
if (PreHtml.Length > 0)
{
    //objlog.Trace_Log(4, "Gordersearchlist", "1", "SliderList", PreHtml);
    //CartData.InnerHtml = PreHtml;

}

```

```

decimal discountprc =5;
decimal discount = Math.Round(price * discountprc / 100,0);
decimal amt = price - discount;
Session["Amt"] = amt.ToString();
CartAmt.InnerHtml = sb.ToString();

```

```

}
}
//protected override void OnInit(EventArgs e)
//{
//    base.OnInit(e);

//    if (this.Master is MainMaster)
//    {
//        ((MainMaster)this.Master).MasterControlClicked += new
EventHandler(ContentControlClicked);
//    }
//}
private void ContentControlClicked(object sender, EventArgs e)
{

```

```

        string pageid = "pageid=" + Session["pageid"].ToString();
        Response.Redirect("~/Managedealer.aspx?PageType=Add&dealername=0&" +
pageid);
    }
    protected void ImageButton1_Click(object sender, ImageClickEventArgs e)
    {
        decimal amt = Convert.ToDecimal( Session["Amt"].ToString());
        Response.Redirect("~/Checkout.aspx?PageType=Pay&Amt="+amt);

    }
}
*****MyProductsList.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="MyProductsList.aspx.cs" Inherits="MyProductsList" %>

<%@ Register Assembly="Infragistics4.Web.v14.2, Version=14.2.20142.2480,
Culture=neutral, PublicKeyToken=7dd5c3163f2cd0cb"
Namespace="Infragistics.Web.UI.GridControls" TagPrefix="ig" %>

<%--<%@ Register Assembly="Infragistics4.WebUI.WebSpellChecker.v14.2,
Version=14.2.20142.1028, Culture=neutral, PublicKeyToken=7dd5c3163f2cd0cb"
Namespace="Infragistics.WebUI.WebDataInput" TagPrefix="igtxt" %>
<%@ Register Assembly="Infragistics4.Web.v14.2, Version=14.2.20142.1028,
Culture=neutral, PublicKeyToken=7dd5c3163f2cd0cb"
Namespace="Infragistics.Web.UI.GridControls" TagPrefix="ig" %>--%>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
</asp:Content>

<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
    <div class="container-fluid">
        <div class="row-fluid" style="padding-top: 20px;">
            <div class="span2" id="dvlogo" runat="server" style="display: block;">
                <%----%>
            </div>
            <div class="span10">
                <div class="row">
                    <div id="dvMsgBox" class="span12 ordersave" runat="server" style="display:
none;">
                        <asp:Label ID="Label1" runat="server" Text=""></asp:Label>
                    </div>
                </div>
            </div>
        </div>
    </div>
</asp:Content>

<asp:Content ID="Content4" ContentPlaceHolderID="CPH_LeftSideBar" runat="Server">
    <div id="LeftSideBar" runat="server" class="span2">
        <asp:Table ID="Table1" runat="server">
            </asp:Table>
        </div>
    </div>

```

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="CPH_MainContent" runat="Server">

<div class="span10" id="MainContent" runat="server">

<div id="ShowPageHeader" class="row" style="display: block;" runat="server">

<div id="PageName" class="span12 customer" runat="server">

</div>

</div>

<div id="Div1" class="row" style="display: block;">

<div id="Div2" class="span12 customertable">

<!-- Main Content Comes here-->

<%--<asp:ScriptManager ID="ScriptManager1"
runat="server"></asp:ScriptManager>--%>

<ig:WebDataGrid ID="WDGrid_UserList" runat="server" CssClass="wdglist"
AutoGenerateColumns="False">

<Columns>

<ig:BoundDataField DataFieldName="CatName" Key="CatName">

<Header Text="Category">

</Header>

</ig:BoundDataField>

<ig:BoundDataField DataFieldName="GroupName" Key="GroupName ">

<Header Text="Sub-Category">

</Header>

</ig:BoundDataField>

<ig:BoundDataField DataFieldName="ProductName" Key="ProductName ">

<Header Text="Product Name">

</Header>

</ig:BoundDataField>

<ig:BoundDataField DataFieldName="SellerDescription"
Key="SellerDescription " Width="150px">

<Header Text="Product Desc">

</Header>

</ig:BoundDataField>

<ig:BoundDataField DataFieldName="CurPrice" Key="CurPrice ">

<Header Text="Price">

</Header>

</ig:BoundDataField>

<ig:BoundDataField DataFieldName="QtyUOM" Key="QtyUOM ">

<Header Text="Min Order">

</Header>

</ig:BoundDataField>

<ig:BoundDataField DataFieldName="DTTlmeType" Key="DTTlmeType ">

<Header Text="Lead Time">

</Header>

</ig:BoundDataField>

<ig:BoundDataField DataFieldName="CreatedOn" Key="CreatedOn"
DataFormatString="{0:dd/MM/yyyy}" Width="100px">

<Header Text="Posted On">

```

        </Header>

</ig:BoundDataField>

<ig:TemplateDataField Key="ProductId" Width="60px">
    <ItemTemplate>
        <asp:HyperLink ID="pcatid" runat="server" CssClass="btn btn-small
btn-primary" Text="Open"
            NavigateUrl='<% #Eval("ProductId",
"-~/ManageProducts.aspx?pageid=5&PageType=Edit&ProductId={0}")%>'>
        </asp:HyperLink>
    </ItemTemplate>
    <Header Text="Open" />
</ig:TemplateDataField>
</Columns>
<Behaviors>
    <ig:Paging CurrentPageLinkCssClass="Current" PageLinkCssClass="Link"
PagerCssClass="Pager" PagerMode="NumericFirstLast" PageSize="15" QuickPages="3"
FirstPageText="First" LastPageText="Last" NextPageText="Next"
PreviousPageText="Previous" PagerAppearance="bottom">
    </ig:Paging>

    <ig:Filtering>
</ig:Filtering>
    <ig:Sorting Enabled="true">
</ig:Sorting>
</Behaviors>
</ig:WebDataGrid>
    <asp:SqlDataSource ID="SqlDataSource1" runat="server"
ConnectionString="<%= $ConnectionStrings.DowConectingString %>"
SelectCommand="SELECT [UserId], [UserName], [FirstName], [LastName], [EmailID],
[ContactNum], [CreatedOn], [Status] FROM [tblUser]"></asp:SqlDataSource>

    </div>
</div>
</div>
</asp:Content>

```

***** MyProductsList.aspx.cs *****

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBEntity;
using DowBAL;
using System.Data;

public partial class MyProductsList: System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        //if (!IsPostBack)
        //{

```

```

ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
if (Clsusr != null)
{
    string pageid = "50";// Request.QueryString["pageid"];
    ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
Convert.ToInt32(Clsusr.GroupId));
    string msg = Request.QueryString["msg"];
    if (msg != null)
    {
        Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
    }
    if (Clsusr.GroupId == "1")
    {
        Bindetails();
    }
}
else
{
    Response.Redirect("MLB_Login.aspx");
}
}

```

```

private void ManageNavigations(int pageid, int userid, int groupid)
{
    bool IsLeftBar = false;
    IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
myBtnClickHandler);

    if (IsLeftBar == true)
    {
        LeftSideBar.Attributes["class"] = "span2";
        MainContent.Attributes["class"] = "span10";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;");//style="display:
block;"
        PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
    else
    {
        LeftSideBar.Attributes["class"] = "";
        MainContent.Attributes["class"] = "span12";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
        PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
}

```

```

// This is Event Handler
protected void myBtnClickHandler(object sender, EventArgs e)
{
    LinkButton Ink = (LinkButton)sender;
    string ID = Ink.ID;

```

```

DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
if (ds != null)
{
    if (ds.Tables[0].Rows.Count > 0)
    {
        DataTable dtdll = ds.Tables[0];
        foreach (DataRow drnew in dtdll.Rows)
        {
            switch (drnew["lbtncode"].ToString())
            {
                case "SAVE":
                    // Call SAVE Method
                    break;
                case "URL":
                    Response.Redirect(drnew["pageurl"].ToString());
                    break;
                case "AUDIT":
                    Response.Redirect(drnew["pageurl"].ToString() + "&frompage=" +
Request.QueryString["pageid"]);
                    break;
                case "CANCEL":
                    Response.Redirect("Home.aspx?pageid=1");
                    break;
                case "NEW":

Response.Redirect("ManageProductCat.aspx?pageid=17&pagetype=Add");
                    break;
                default:

                    break;
            }
        }
    }
}

```

```

private void Bindetails()
{
    //try
    //{
        int prid=0;
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        string pagetype=Request.QueryString["pagetype"];
        if(pagetype=="Buy"){prid=1;}else {prid=2;}
        List<ProductProductSeller> Lst = (new
Products()).GetProductSellerbyProductId(Convert.ToInt16(Clsusr.UserId), "");
        Lst=(from c in Lst where c.Prid==1 select c).ToList();
        if (Lst != null && Lst.Count > 0)
        {
            WDGrid_UserList.DataSource = Lst;
            WDGrid_UserList.DataBind();
        }
    }
}

```

```

        WDGrid_UserList.Visible = true;
    }
    else
    {
        WDGrid_UserList.Visible = false;
    }
}
//protected override void OnInit(EventArgs e)
//{
//    base.OnInit(e);

//    if (this.Master is MainMaster)
//    {
//        ((MainMaster)this.Master).MasterControlClicked += new
EventHandler(ContentControlClicked);
//    }
//}
private void ContentControlClicked(object sender, EventArgs e)
{
    string pageid = "pageid=" + Session["pageid"].ToString();
    Response.Redirect("~/Managedealer.aspx?PageType=Add&dealername=0&" +
pageid);
}
}
***** MyWishList.aspx*****
<%@ Page Title="" Language="C#" MasterPageFile="~/ShoppingMasterPage.master"
AutoEventWireup="true" CodeFile="MyWishList.aspx.cs" Inherits="MyWishList" %>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentMainPage" runat="Server">

<div class="span10" id="MainContent" runat="server">
<div id="ShowPageHeader" class="row" style="display: block;" runat="server">
<div id="PageName" class="span12 customer" runat="server">
</div>
</div>

<div id="Div1" class="row" style="display: block;">

<!-- Content Bodey Here -->

<section class="main-contain bg-white">
<div id="CartAmt" runat="server" class="container">
<div class="shopping-cart-block">
<ul class="title clearfix">
<li class="col-md-2 col-sm-2 col-xs-6">Image</li>
<li class="col-md-5 col-sm-5 col-xs-8">Product Name</li>
<li class="col-md-2 col-sm-2 col-xs-6">Unite Price</li>
<li class="col-md-1 col-sm-1 col-xs-6">Action</li>
</ul>
<ul class="shopping-cart-cd clearfix">

```



```

        <li class="col-md-2 col-sm-2 col-xs-6"><figure></figure></li>
        <li class="col-md-5 col-sm-5 col-xs-8 cd-shopping">
            <h6>Summer Dress</h6>
            <p><span>Color:</span> Brown</p>
            <p><span>Size:</span> 12</p>
        </li>
        <li class="col-md-2 col-sm-2 col-xs-6">$520.00</li>
        <li class="col-md-1 col-sm-1 col-xs-6"><a href="#">X</a></li>
    </ul>
    <ul class="shopping-cart-cd clearfix">
        <li class="col-md-2 col-sm-2 col-xs-6"><figure></figure></li>
        <li class="col-md-5 col-sm-5 col-xs-8 cd-shopping">
            <h6>Summer Dress</h6>
            <p><span>Color:</span> Brown</p>
            <p><span>Size:</span> 12</p>
        </li>
        <li class="col-md-2 col-sm-2 col-xs-6">$520.00</li>
        <li class="col-md-1 col-sm-1 col-xs-6"><a href="#">X</a></li>
    </ul>
    <ul class="shopping-cart-cd clearfix">
        <li class="col-md-2 col-sm-2 col-xs-6"><figure></figure></li>
        <li class="col-md-5 col-sm-5 col-xs-8 cd-shopping">
            <h6>Summer Dress</h6>
            <p><span>Color:</span> Brown</p>
            <p><span>Size:</span> 12</p>
        </li>
        <li class="col-md-2 col-sm-2 col-xs-6">$520.00</li>
        <li class="col-md-1 col-sm-1 col-xs-6"><a href="#">X</a></li>
    </ul>
</div>

```

```
<div class="shopping-cart-checkout">
```

```

</div>
</div>
</section>

```

```
<!-- ./Content Bodey Here -->
```

```

</div>
</div>
</asp:Content>
<asp:Content ID="Content4" ContentPlaceHolderID="ContentFooter" runat="Server">
</asp:Content>

```

```

***** MyWishList.aspx.cs*****
using System;
using System.Collections.Generic;
using System.Linq;

```

```

using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBEntity;
using DowBAL;
using System.Data;
using System.Text.RegularExpressions;
using System.Text;
public partial class MyWishList : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        //if (!IsPostBack)
        //{
            ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
            if (Clsusr != null)
            {
                string pageid = "50";// Request.QueryString["pageid"];
                ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
Convert.ToInt32(Clsusr.GroupId));
                string msg = Request.QueryString["msg"];
                if (msg != null)
                {
                    //Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
                }
                Session["Amt"] = "0";
                string psid =( Request.QueryString["psid"]);
                if(psid!=null ){
                    EteUserShortList lst = new EteUserShortList();
                    lst.UserId = Convert.ToInt16(Clsusr.UserId);
                    lst.PSId =Convert.ToInt32( psid);
                    lst.UId = Convert.ToInt16(Clsusr.UserId);

                    lst.CreatedOn = DateTime.Today;
                    lst.Active = true;
                    bool result = new Products().InsertShortList(lst, true);

                }
                Bindetails();

            }
            else
            {
                Response.Redirect("MLB_Login.aspx");
            }
        }

        private void ManageNavigations(int pageid, int userid, int groupid)
        {
            bool IsLeftBar = false;
            //IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
myBtnClickHandler);

            //if (IsLeftBar == true)

```

```

    //{
    // LeftSideBar.Attributes["class"] = "span2";
    // MainContent.Attributes["class"] = "span10";
    // ShowPageHeader.Attributes.CssStyle.Add("display", "block;"); //style="display:
block;"
    // PageName.InnerText = new
tblPageess().GettblPageNamebyId(Convert.ToInt32(pageid));
    //}
    //else
    //{
    // LeftSideBar.Attributes["class"] = "";
    // MainContent.Attributes["class"] = "span12";
    // ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
    // PageName.InnerText = new
tblPageess().GettblPageNamebyId(Convert.ToInt32(pageid));
    //}
}

```

```

// This is Event Handler
protected void myBtnClickHandler(object sender, EventArgs e)
{
    LinkButton lnk = (LinkButton)sender;
    string ID = lnk.ID;

    DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
    if (ds != null)
    {
        if (ds.Tables[0].Rows.Count > 0)
        {
            DataTable dtdll = ds.Tables[0];
            foreach (DataRow drnew in dtdll.Rows)
            {
                switch (drnew["lbtncode"].ToString())
                {
                    case "SAVE":
                        // Call SAVE Method
                        break;
                    case "URL":
                        Response.Redirect(drnew["pageurl"].ToString());
                        break;
                    case "AUDIT":
                        Response.Redirect(drnew["pageurl"].ToString() + "&frompage=" +
Request.QueryString["pageid"]);
                        break;
                    case "CANCEL":
                        Response.Redirect("Home.aspx?pageid=1");
                        break;
                    case "NEW":
                        Response.Redirect("ManageSubgroup.aspx?pageid=18&pagetype=Add");
                        break;
                    default:
                        break;
                }
            }
        }
    }
}

```

```

    }
    }
}
}
}

```

```

private void Bindetails()
{
    //try
    //{
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        string PreHtml = "";
        StringBuilder sb = new StringBuilder();

        List<EteMyUserCart> Lst = (new
Products()).GetShortListbyUserId(Convert.ToInt16(Clsusr.UserId),true);
        string title = "My WishList ";
        decimal price = 0;
        sb.Append(" <div class='shopping-cart-block'> \n");
        sb.Append(" <ul class='title clearfix'> \n");
        sb.Append(" <li class='col-md-2 col-sm-2 col-xs-6'>Image</li> \n");
        sb.Append(" <li class='col-md-5 col-sm-5 col-xs-8'>Product Name</li> \n");
        sb.Append(" <li class='col-md-2 col-sm-2 col-xs-6'>Unite Price</li> \n");
        sb.Append(" <li class='col-md-1 col-sm-1 col-xs-6'>Action</li> \n");
        sb.Append(" </ul> \n");
        int sno = 0;
        if (Lst != null && Lst.Count > 0)
        {
            foreach (var list in Lst)
            {
                sno++;
                sb.Append(" <ul class='shopping-cart-cd clearfix'> \n");
                sb.Append(" <li class='col-md-2 col-sm-2 col-xs-6'><figure><img src=\"" +
list.ImagePath + "\" alt=\"" /></figure></li> \n");
                sb.Append(" <li class='col-md-5 col-sm-5 col-xs-8 cd-shopping'>
\n");
                sb.Append(" <h6>" + list.SellerDescription + "</h6> \n");
                sb.Append(" <p><span>Color:</span> Brown</p> \n");
                sb.Append(" <p><span>Size:</span> 12</p> \n");
                sb.Append(" </li> \n");
                sb.Append(" <li class='col-md-2 col-sm-2 col-xs-6'>Rs." + list.Price + "</li>
\n");
                sb.Append(" <li class='col-md-1 col-sm-1 col-xs-6'><a href='#'>X</a></li>
\n");
                sb.Append(" <li class='col-md-1 col-sm-1 col-xs-6'> <a
href='MyCart.aspx?psid=" + list.PSId + ">Add to Cart</a></li> \n");
                sb.Append(" </ul> \n");
                price = price + list.Price;
            }
            sb.Append(" </div> \n");
            //sb.Append(" <div class='shopping-cart-checkout'> \n");
            //sb.Append(" <div class='coupon-code'> \n");

```

```

        //sb.Append(" <div class='row'> \n");
        //sb.Append(" <div class='col-sm-4 sm-margin-bottom-30'> \n");
        //sb.Append(" <h3>Discount Code</h3> \n");
        //sb.Append(" <p>Enter your coupon code</p> \n");
        //sb.Append(" <input class='form-control margin-bottom-10' name='code'
type='text'> \n");
        //sb.Append(" <button type='button' class='btn-check checkout-btn'>Apply
Coupon</button> \n");
        //sb.Append(" </div> \n");
        //sb.Append(" <div class='col-sm-3 col-sm-offset-5'> \n");
        //sb.Append(" <ul class='list-inline total-result'> \n");
        //sb.Append(" <li> \n");
        //sb.Append(" <h4>Subtotal:</h4> \n");
        //sb.Append(" <div class='total-result-in'> \n");
        //sb.Append(" <span>"+price+"</span> \n");
        //sb.Append(" </div> \n");
        //sb.Append(" </li> \n");
        //sb.Append(" <li> \n");
        //sb.Append(" <h4>Shipping:</h4> \n");
        //sb.Append(" <div class='total-result-in'> \n");
        //sb.Append(" <span class='text-right'>- - -</span> \n");
        //sb.Append(" </div> \n");
        //sb.Append(" </li> \n");
        //sb.Append(" <li class='divider'></li> \n");
        //sb.Append(" <li class='total-price'>\n");
        //sb.Append(" <h4>Total:</h4> \n");
        //sb.Append(" <div class='total-result-in'> \n");
        //sb.Append(" <span>"+price+"</span> \n");
        //sb.Append(" </div> \n");
        //sb.Append(" </li> \n");
        //sb.Append(" </ul> \n");
        //sb.Append(" <div class='mb30'></div> \n");
        //sb.Append(" <a href='#' class='btn btn-product pull-right'>Checkout</a>\n");
        //sb.Append(" </div> \n");
        //sb.Append(" </div> \n");
        //sb.Append(" </div> \n");
        //sb.Append(" </div> \n");
    }

```

```
PreHtml = sb.ToString();
```

```
if (PreHtml.Length > 0)
```

```
{
```

```
    //objlog.Trace_Log(4, "Gordersearchlist", "1", "SliderList", PreHtml);
```

```
    CartAmt.InnerHtml = PreHtml;
```

```
}
```

```
}
```

```
//private void ContentControlClicked(object sender, EventArgs e)
```

```
//{
```

```
//    string pageid = "pageid=" + Session["pageid"].ToString();
```

```
//Response.Redirect("~/Managedealer.aspx?PageType=Add&dealername=0&" + pageid);
```

```
//}
```

```
protected void ImageButton1_Click(object sender, ImageClickEventArgs e)
```

```

    {
        decimal amt =Convert.ToDecimal( Session["Amt"].ToString());
        Response.Redirect("~/Checkout.aspx?PageType=Pay&Amt="+amt);
    }
protected void RadioButton1_CheckedChanged(object sender, EventArgs e)
{
}
}
}
}
***** OrderSearchList.aspx*****
<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="OrderSearchList.aspx.cs" Inherits="OrderSearchList"
%>
<%@ Register Assembly="Infragistics4.Web.v14.2, Version=14.2.20142.1028,
Culture=neutral, PublicKeyToken=7dd5c3163f2cd0cb"
Namespace="Infragistics.Web.UI.GridControls" TagPrefix="ig" %>
<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
<%--<link rel="stylesheet" href="css/swiper.css"/>
<link rel="stylesheet" href="css/swiper.min.css"/>
<link rel="stylesheet" href="css/bootstrap.css"/>

<link rel="stylesheet" href="css/base.css"/>
<link rel="stylesheet" href="css/migrate.css"/>
<script src="js/swiper.js"></script>
<script src="js/swiper.min.js"></script>
<script src="swiper.esm.js"></script>
<script src="js/swiper.esm.bundle.js"></script>

<link rel="stylesheet" href="css/vendors.css"/>

<script src="js/common.js"></script>
<script src="js/jquery.countdown.js"></script>
<meta charset="utf-8"/>
<meta name="viewport" content="width=device-width, initial-scale=1"/>
<script src="js/jquery1.min.js"></script>
<script src="js/bootstrap1.min.js"></script>--%>

<script type="text/javascript">
function ImgClicked(Id) {
alert('ImgClicked-1');
//var fullURL = (location.pathname + location.search).substr(1);
<%--      var loc = document.getElementById("<%=ddl_location.ClientID%>").value;

var EnCodedFullURL = encodeURIComponent(fullURL);
var SearchString = document.getElementById("<%=txt_SearchString.ClientID%>").value; --
%>

var url = 'ViewSearchResult.aspx?pageid=198' + '&SrchStrng=' +
encodeURIComponent(Id.value) ;
//'+ '&loc=' + encodeURIComponent(loc);
//$(location).attr('href', url);
window.location.href = url;

```

```

event.preventDefault();
}
function submitButton(event) {
//alert(event.which);
if (event.which == 13) {
alert('Enter');
//$('#Button1').trigger('click');
GetURLString();
}
}

function TextName_OnKeyDown(e) {
//alert('TextName_OnKeyDown');
var keynum
if (window.event) // IE
{
keynum = e.keyCode
}
else if (e.which) // Netscape/Firefox/Opera
{
keynum = e.which
}

```

```

if (keynum == 13) {
//alert('Enter');
GetURLString()
}
}
function GetLoginString() {
// alert('GetLoginString-1');
var url = 'Login.aspx?';
//$(location).attr('href', url);
window.location.href = url;
event.preventDefault();
}
function GetRegisterString() {
// alert('GetLoginString-1');
var url = 'Register.aspx?';
//$(location).attr('href', url);
window.location.href = url;
event.preventDefault();
}

```

```

</script>
</asp:Content>

```

```

<asp:Content ID="Content2" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
<div class="container-fluid">
<div class="row-fluid" style="padding-top: 20px;">

<div class="homeSlider__content">
<h1 id="subt" class="homeSlider__title app-main-slider-home-title"
runat="server">Wedding Dresses</h1>

```



```
<div class="span10" id="MainContent" runat="server">
<div id="ShowPageHeader" class="row" style="display: block;" runat="server">
<div id="PageName" class="span10 customer" runat="server">
</div>
</div>

<div id="Div1" class="row" style="display: block;">
<div id="Div2" class="span10 "> <!--customertable-->
<div class="wrapper">
<div class="directory-structure clearfix" >
<div id="appvendors" class="span10 directory-structure-content" runat="server"> <!--
directory-structure-content-->
<!--data update-->
```

```
<!--data update-->
```

```
</div>
</div>
```

```
</div>
```

```
</div>
</div>
</div>
```

```
</asp:Content>
```

```
***** OrderSearchList.aspx.cs*****
using DowBAL;
using DowBEntity;

using System;
using System.Collections.Generic;
using System.Data;
using System.Diagnostics;
using System.IO;
```

```

using System.Linq;
using System.Net;
using System.Text.RegularExpressions;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Drawing;
using System.Text;
public partial class OrderSearchList : System.Web.UI.Page
{
    DDLogTracer objlog = new DDLogTracer();
    protected void Page_Load(object sender, EventArgs e)
    {
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr != null)
        {
            string pageid = "50";// Request.QueryString["pageid"];
            string searchtype = Request.QueryString["searchtype"];
            string bytype = Request.QueryString["bytype"];
            int pid = Convert.ToInt16(Request.QueryString["id"]);
            Eteproductcat cats = new Products().GetProductCat(pid);
            if (cats.pcatid == 0)
            {
                pid = Convert.ToInt16(Request.QueryString["pid"]);
                cats = new Products().GetProductCatbypid(pid);
            }
            img1.Src = cats.lbtimage;
            subt.InnerText = cats.pcatdesc;
            H1.InnerText = "";
            txt_SearchString.Text = cats.pcatdesc;

            ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
            Convert.ToInt32(Clsusr.GroupId));
            string msg = Request.QueryString["msg"];
            if (msg != null)
            {
                Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
            }

            if (searchtype == "1")
            {
                SetStatus(Table1, bytype, "STATUS");
            }
            else if (searchtype == "2")
            {
                SetStatus(Table1, bytype, "QUEUE");
            }
            //SetStatus(Table1, searchtype);
            Bindetails();
        }
        else
        {
            Response.Redirect("MLB_Login.aspx");
        }
    }
}

```

```

}
private void Bindetails()
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    string pt = "cat";
    int pid = Convert.ToInt16( Request.QueryString["id"]);
    int GroupId = Convert.ToInt16( Clsusr.GroupId);
    string PreHtml = "";
    StringBuilder sb = new StringBuilder();
    sb.Append(" <div class='directory-list'>");
    bool slider = true;
    if (pid==0) {
        pid = Convert.ToInt16(Request.QueryString["pid"]);
        pt = "prod";
    }

    try
    {
        IList<EteProductGroup> Lst;
        if(pt=="cat"){
            Lst = (new Products()).GetProductsList("SearchList", pid);
        }
        else{
            Lst = (new Products()).GetProductsListbypid("SearchList", (decimal) pid);
        }

        if (Lst != null && Lst.Count > 0)
        {
            foreach (var dt in Lst)
            {string href="",title;
                if(GroupId==5){
                    title = "Search ";
                    href = "ProductSearchList.aspx?id=" + dt.pgid;}
                else
                {
                    href = "PostProducts.aspx?id=" + dt.pgid;
                    title = "Post ";
                }
                //PostProducts.aspx

                //code
                if (dt.lbtnimage == null)
                {
                    continue;
                }
                if (slider)
                {
                    sb.Append(" <div class='app-ec-list app-internal-tracking-page' data-it-
page='1'>\n\r");
                    sb.Append(" <div id='esl_81030" + dt.PCatId + dt.pgid.ToString() + "
class='pure-g-r directory-list-item app-internal-tracking-item app-vendors-item app-vpromo
appGallery_35"");
                    sb.Append("      data-ec-brand='Free'>\n ");
                }
            }
        }
    }
}

```

```

thumb
sb.Append("<div class='pure-u-1-3 vendor-slider-thumb'>\n"); // vendor-slider-
thumb
sb.Append("<div class='vendor-slider'>\n");
sb.Append(" <div class='app-vendors-item-slider'></div>\n");
sb.Append(" <div class='vendor-slider-content'>\n");
sb.Append(" <div class='listing-caption'>\n");
sb.Append(" <span class='listing-caption-count'>\n");
sb.Append(" <i class='icon-vendor icon-vendor-count-pic icon-
left'></i>Nos:" + dt.PNos + " </span>\n");
sb.Append(" </div>\n");
sb.Append(" <figure class='homePremiumBox__figure img-zoom'>");
sb.Append("<a href='" + href + "'>");
sb.Append(" <img class='homePremiumBox__img' alt='Pic-" +
dt.pgroupdescription + "' src='" + dt.lbtnimage + "' />");
sb.Append("</a>");
sb.Append(" </figure>");

//sb.Append(" <img class='app-link' src='" + dt.lbtnimage + "' alt ='Pic-" +
dt.pgroupdescription + "' onclick='JavaScript:ImgClicked(\"+dt.pgid+\") height='300px'
width='300px'/>\n");
sb.Append("</div>\n");
sb.Append("</div>\n");

sb.Append("</div>\n");

sb.Append("<div class='pure-u-2-3'>\n");
sb.Append(" <div class='directory-item-content'>\n");

sb.Append("<a id='app_InkEmp_" + dt.pgid.ToString() + "' \n");
sb.Append(" class='item-title app_InkEsc_9880 app-ec-track-vendor-
click'\n");
sb.Append(" href='" + href + "'>" + dt.pgroupdescription + "</a>\n");

sb.Append("<div class='btn-fav app-vendors-item-guardar no-hover'");
sb.Append("data-id=" + dt.PCatId + dt.pgid.ToString() + """);
sb.Append("data-id-sector='122'>");
sb.Append("</div>\n");

sb.Append("<div class='item-subtitle'> <span>" + dt.pgroupdescription + "
<span> </div>\n");

sb.Append("<p class='item-desc'>");
sb.Append(" <span class='app-common-ellipsis'>" + dt.pgroupname +
"...</span>");
sb.Append(" <span>(" + dt.Catname + ")</span> </p>");

sb.Append("<div class='listing-item-faq-container'>\n");

sb.Append(" <div class='listing-item-faq'> ");
sb.Append(" <i class='icon-vendor icon-vendor-faq-price'></i>\n");
sb.Append(" <p>");

```

```

        sb.Append("    <span class='item-faqs-label notablet'>");
        sb.Append("    Price from    </span>");
        sb.Append("    <span class='item-faqs-value'>₹" + dt.Price.ToString() +
"</span>");
        sb.Append(" </p>\n");
        sb.Append(" <a title='" + title + "' class='btn btn-primary' rel='nofollow' href='"
+ href + "'>" + title + dt.pgrouppdescription + "</a>");
        sb.Append("    </div>\n");
        sb.Append(" </div>\n");
        sb.Append(" </div>\n");
        sb.Append(" </div>\n");
        sb.Append(" </div>\n");
        sb.Append(" </div>\n");
        sb.Append(" </div>\n");
        sb.Append(" <div class='clear'>                <p> </p>                </div>");

        //foreach code ends
    }
}
if (slider)
{
    //sb.Append("<div class='app-vendors-paginado nav pagination mt20 mb20'>");

    //sb.Append("<script>");
    //sb.Append("vendors_fncEmpLNL();");
    //sb.Append("</script>");
    sb.Append("</div>\n");

    PreHtml = sb.ToString();
    if (PreHtml.Length > 0)
    {
        //objlog.Trace_Log(4, "Gordersearchlist", "1", "SliderList", PreHtml);
        appvendors.InnerHtml = PreHtml;
    }
}
}
}
}
catch
{
    Common.ShowMessageHtml("No records found", 0, true, dvMsgBox);
}
}

private void ManageNavigations(int pageid, int userid, int groupid)
{
    bool IsLeftBar = false;
    IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
myBtnClickHandler);

    if (IsLeftBar == true)
    {

```

```

        LeftSideBar.Attributes["class"] = "span2";
        MainContent.Attributes["class"] = "span10";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;"); //style="display:
block;"
        PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
    else
    {
        LeftSideBar.Attributes["class"] = "";
        MainContent.Attributes["class"] = "span12";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
        PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
}

```

```

// This is Event Handler
protected void myBtnClickHandler(object sender, EventArgs e)
{
    //LinkButton lnk = (LinkButton)sender;
    //string ID = lnk.ID;

    LinkButton lnk = sender as LinkButton;

    if (lnk != null)
    {
        string ID = lnk.ID;
        Common.ShowMessageHtml("Please wait while refreshing the records", 1, true,
dvMsgBox);
        string pid = (Convert.ToInt16(Request.QueryString["pid"]) - 8).ToString();
        DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
        if (ds != null)
        {
            if (ds.Tables[0].Rows.Count > 0)
            {
                DataTable dtdll = ds.Tables[0];
                foreach (DataRow drnew in dtdll.Rows)
                {
                    switch (drnew["lbtncode"].ToString())
                    {
                        case "SAVE":
                            // Call SAVE Method
                            break;
                        case "URL":
                            Response.Redirect(drnew["pageurl"].ToString() + "&pid=" +
drnew["lbtnid"]);
                            break;
                        case "AUDIT":
                            Response.Redirect(drnew["pageurl"].ToString() + "&frompage=" +
Request.QueryString["pageid"]);
                            break;
                    }
                }
            }
        }
    }
}

```



```

        if (listDrop.Attributes["LISTTYPE"] != null)
        {
            string rowcellHead = listDrop.Attributes["LISTTYPE"].ToUpper();
//rowcell.Text.ToUpper();
            ExtractAll(ref orderStatusID, ref OrderHold, ref brandID, ref
customerType, ref dateRange, ref dealerID, rowcellHead, listDrop);
        }
    }
else
    {
        continue;
    }
}
}
}

FilterEnum filterLst = new FilterEnum();
filterLst.OrderStatus = orderStatusID;
filterLst.OrderHold = OrderHold;
filterLst.Brand = brandID;
filterLst.CustomerType = customerType;
filterLst.DateRange = dateRange;
filterLst.Dealer = dealerID;
//RefreshData(filterLst);
}
catch (Exception ex)
{
    objlog.Trace_Error(5, "DateOrWed", "OrderSearchList", "Bindetails_Filter",
ex.Message);
}
}
}

```

```

private static void ExtractAll(ref int orderStatusID, ref int OrderHold, ref int brandID, ref int
customerType, ref int dateRange, ref int dealerID, string rowcellHead, DropDownList
listDrop)
{
    //string strddl=rowcellHead;
    switch (rowcellHead)
    {
        case "LOCATION":
            brandID = Convert.ToInt32(listDrop.SelectedValue);
            break;
        case "PRODUCTS":
            brandID = Convert.ToInt32(listDrop.SelectedValue);

            break;
        //case "BRAND":
        // if (string.Compare(listDrop.SelectedItem.Text, "All", true) == 0)
        // {
        //     brandID = 0;
        // }
    }
}

```



```

// else
// {
//     brandID = Convert.ToInt32(listDrop.SelectedValue);
// }
// break;
//case "CUSTOMER TYPE":
//     string strCustType = listDrop.SelectedItem.Text;
//     if (string.Compare(strCustType, "Residential", true) == 0)
//     {
//         customerType = 1;
//     }
//     else if (string.Compare(strCustType, "Business", true) == 0)
//     {
//         customerType = 2;
//     }
//     else
//     {
//         customerType = 0;
//     }
//     break;
//case "DATE RANGE":
//     dateRange = SelectDateRange(dateRange, listDrop);
//     break;
//case "DEALER":
//     if (string.Compare(listDrop.SelectedItem.Text, "All", true) == 0)
//     {
//         dealerID = 0;
//     }
//     else
//     {
//         dealerID = Convert.ToInt32(listDrop.SelectedValue);
//     }

//     break;
//case "STATUS":
//     if (string.Compare(listDrop.SelectedItem.Text, "All", true) == 0)
//     {
//         orderStatusID = 0;
//     }
//     else
//     {
//         orderStatusID = Convert.ToInt32(listDrop.SelectedValue);
//     }

//     break;
//case "ON HOLD":
//     string strOnHold = listDrop.SelectedItem.Text;
//     if (string.Compare(strOnHold, "Yes", true) == 0)
//     {
//         OrderHold = 1;
//     }
//     else if (string.Compare(strOnHold, "No", true) == 0)
//     {
//         OrderHold = 2;
//     }

```

```

        // else
        // {
        //     OrderHold = 0;
        // }
        // break;
        default:
            break;
    }

}

private static void SetStatus(Table tbl, string LoggedinID, string FilterName)
{
    try
    {
        foreach (TableRow row in tbl.Rows)
        {
            foreach (TableCell rowcell in row.Cells)
            {
                foreach (WebControl contrl in rowcell.Controls)
                {
                    DropDownList listDrop = contrl as DropDownList;
                    if (listDrop != null)
                    {
                        if (listDrop.Attributes["LISTTYPE"] != null &&
listDrop.Attributes["LISTTYPE"].ToUpper() == FilterName)
                        {
                            listDrop.SelectedValue = LoggedinID;
                        }
                    }
                    else
                    {
                        continue;
                    }
                }
            }
        }
    }
    catch (Exception exc)
    {
    }
}

private static int SelectDateRange(int dateRange, DropDownList listDrop)
{
    string strDateRange = listDrop.SelectedItem.Text.ToUpper();
    switch (strDateRange)
    {
        case "ALL":
            {
                dateRange = 0;
                break;
            }
    }
}

```

```
}
case "TODAY":
{
    dateRange = 1;
    break;
}

case "YESTERDAY":
{
    dateRange = 2;
    break;
}

case "THIS WEEK":
{
    dateRange = 3;
    break;
}

case "LAST WEEK":
{
    dateRange = 4;
    break;
}

case "LAST 7 DAYS":
{
    dateRange = 5;
    break;
}

case "LAST 30 DAYS":
{
    dateRange = 6;
    break;
}

case "THIS MONTH":
{
    dateRange = 7;
    break;
}

case "LAST MONTH":
{
    dateRange = 8;
    break;
}

case "THIS YEAR":
{
    dateRange = 9;
    break;
}

case "LAST YEAR":
{
    dateRange = 10;
    break;
}

default:
    dateRange = 0;
```

```

        break;

    }
    return dateRange;
}

protected void Lnkbtn_ManageOrder_Click(object sender, EventArgs e)
{
    LinkButton lnkbutton = (LinkButton)sender;
    string strcustid = lnkbutton.CommandName;

    if (strcustid.Length > 0)
    {
        string[] stringSeparators = new string[] { "&", "=" };
        string[] arr1;
        arr1 = strcustid.Split(stringSeparators, StringSplitOptions.RemoveEmptyEntries);
        bool bchkstatus = false;
        foreach (var word in arr1)
        {
            if (word == "status")
            {
                bchkstatus = true;
            }

            if (bchkstatus == true && word != "status")
            {
                Response.Redirect(Navigations.StatusWiseRedirect(Convert.ToInt32(word),
strcustid));
            }

        }

    }
}

protected void ImageButton1_Click(object sender, ImageClickEventArgs e)
{
}
}
}

*****PackTypesList.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="PackTypesList.aspx.cs" Inherits="PackTypesList" %>

<%@ Register Assembly="Infragistics4.Web.v14.2, Version=14.2.20142.2480,
Culture=neutral, PublicKeyToken=7dd5c3163f2cd0cb"
Namespace="Infragistics.Web.UI.GridControls" TagPrefix="ig" %>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
</asp:Content>

```

```

<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
  <div class="container-fluid">
    <div class="row-fluid" style="padding-top: 20px;">
      <div class="span2" id="dvlogo" runat="server" style="display: block;">
        <%----%>
      </div>
      <div class="span10">
        <div class="row">
          <div id="dvMsgBox" class="span12 ordersave" runat="server" style="display:
none;">
            <asp:Label ID="Label1" runat="server" Text=""></asp:Label>
          </div>
        </div>
      </div>
    </div>
  </div>
</asp:Content>

```

```

<asp:Content ID="Content4" ContentPlaceHolderID="CPH_LeftSideBar" runat="Server">
  <div id="LeftSideBar" runat="server" class="span2">
    <asp:Table ID="Table1" runat="server">
      </asp:Table>
    </div>
</asp:Content>

```

```

<asp:Content ID="Content2" ContentPlaceHolderID="CPH_MainContent" runat="Server">

  <div class="span10" id="MainContent" runat="server">
    <div id="ShowPageHeader" class="row" style="display: block;" runat="server">
      <div id="PageName" class="span12 customer" runat="server">
        </div>
      </div>

    <div id="Div1" class="row" style="display: block;">
      <div id="Div2" class="span12 customertable">

        <!-- Main Content Comes here-->

        <%--<asp:ScriptManager ID="ScriptManager1"
runat="server"></asp:ScriptManager>--%>
        <ig:WebDataGrid ID="WDGrid_UserList" runat="server" CssClass="wdglist"
AutoGenerateColumns="False">
          <Columns>
            <ig:BoundDataField DataFieldName="PackShort" Key="PackShort">
              <Header Text="Name">
                </Header>
            </ig:BoundDataField>

            <ig:BoundDataField DataFieldName="PackDescription"
Key="PackDescription ">
              <Header Text="Description">
                </Header>
            </ig:BoundDataField>

```

```

        <ig:BoundDataField DataFieldName="CreatedOn" Key="CreatedOn"
DataFormatString="{0:dd/MM/yyyy}" Width="100px">
            <Header Text="Created On">
                </Header>
            <%-- </ig:BoundDataField>
            <ig:BoundDataField DataFieldName="DealerOperationProfileId"
Key="DealerOperationProfileId">
                <Header Text="Deater Type">
                    </Header>
                </ig:BoundDataField>
            <ig:BoundDataField DataFieldName="IsActive" Key="IsActive">
                <Header Text="Is Active">
                    </Header>--%>
            </ig:BoundDataField>

            <ig:TemplateDataField Key="PackID" Width="60px">
                <ItemTemplate>
                    <asp:HyperLink ID="PackID" runat="server" CssClass="btn btn-small
btn-primary" Text="Open"
                    NavigateUrl='<% #Eval("PackID",
"~/ManageNoteTypes.aspx?pageid=183&PageType=Edit&id={0}")%>'>
                        </asp:HyperLink>
                    </ItemTemplate>
                    <Header Text="Open" />
                </ig:TemplateDataField>
            </Columns>
            <Behaviors>
                <ig:Paging CurrentPageLinkCssClass="Current" PageLinkCssClass="Link"
PagerCssClass="Pager" PagerMode="NumericFirstLast" PageSize="15" QuickPages="3"
FirstPageText="First" LastPageText="Last" NextPageText="Next"
PreviousPageText="Previous" PagerAppearance="bottom">
                    </ig:Paging>

                <ig:Filtering>
                    </ig:Filtering>
                <ig:Sorting Enabled="true">
                    </ig:Sorting>
                </Behaviors>
            </ig:WebDataGrid>
            <asp:SqlDataSource ID="SqlDataSource1" runat="server"
ConnectionString="<%%$ ConnectionStrings:DowConectingString %>"
SelectCommand="SELECT [UserId], [UserName], [FirstName], [LastName], [EmailID],
[ContactNum], [CreatedOn], [Status] FROM [tblUser]"></asp:SqlDataSource>

        </div>
    </div>
</div>
</asp:Content>

```

```

***** PackTypesList.aspx.cs*****
using System;
using System.Collections.Generic;
using System.Linq;

```

```

using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBEntity;
using DowBAL;
using System.Data;

public partial class PackTypesList: System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        //if (!IsPostBack)
        //{
            ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
            if (Clsusr != null)
            {

                string pageid = "6";// Request.QueryString["pageid"];
                ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
Convert.ToInt32(Clsusr.GroupId));
                string msg = Request.QueryString["msg"];
                if (msg != null)
                {
                    Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
                }
                if (Clsusr.GroupId == "1")
                {
                    Bindetails();
                }
            }
            else
            {
                Response.Redirect("MLB_Login.aspx");
            }
        }

        private void ManageNavigations(int pageid, int userid, int groupid)
        {
            bool IsLeftBar = false;
            IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
myBtnClickHandler);

            if (IsLeftBar == true)
            {
                LeftSideBar.Attributes["class"] = "span2";
                MainContent.Attributes["class"] = "span10";
                ShowPageHeader.Attributes.CssStyle.Add("display", "block;"); //style="display:
block;"
                PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
            }
            else
            {
                LeftSideBar.Attributes["class"] = "";
                MainContent.Attributes["class"] = "span12";
            }
        }
    }
}

```

```

        ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
        PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
}

// This is Event Handler
protected void myBtnClickHandler(object sender, EventArgs e)
{

    LinkButton lnk = (LinkButton)sender;
    string ID = lnk.ID;

    DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
    if (ds != null)
    {
        if (ds.Tables[0].Rows.Count > 0)
        {
            DataTable dtdll = ds.Tables[0];
            foreach (DataRow drnew in dtdll.Rows)
            {
                switch (drnew["lbtncode"].ToString())
                {
                    case "SAVE":
                        // Call SAVE Method
                        break;
                    case "URL":
                        Response.Redirect(drnew["pageurl"].ToString());
                        break;
                    case "AUDIT":
                        Response.Redirect(drnew["pageurl"].ToString() + "&frompage=" +
Request.QueryString["pageid"]);
                        break;
                    case "CANCEL":
                        Response.Redirect("Home.aspx?pageid=1");
                        break;
                    case "NEW":

Response.Redirect("ManagePackTypes.aspx?pageid=15&pagetype=Add");
                        break;
                    default:

                        break;
                }
            }
        }
    }
}

private void Bindetails()
{

```



```

//try
//{

List<Etetblpacking> Lst = (new Products()).GetProducttblpackings("");

if (Lst != null && Lst.Count > 0)
{
    WDGrid_UserList.DataSource = Lst;
    WDGrid_UserList.DataBind();
    WDGrid_UserList.Visible = true;
}
else
{
    WDGrid_UserList.Visible = false;
}
}
//protected override void OnInit(EventArgs e)
//{
//    base.OnInit(e);

//    if (this.Master is MainMaster)
//    {
//        ((MainMaster)this.Master).MasterControlClicked += new
EventHandler(ContentControlClicked);
//    }
//}
private void ContentControlClicked(object sender, EventArgs e)
{

    string pageid = "pageid=" + Session["pageid"].ToString();
    Response.Redirect("~/Managedealer.aspx?PageType=Add&dealername=0&" +
pageid);
}
}
*****PayOut.aspx*****
<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="PayOut.aspx.cs" Inherits="PayOut" %>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
</asp:Content>

<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
<div class="container-fluid">
<div class="row-fluid" style="padding-top: 20px;">
<div class="span2" id="dvlogo" runat="server" style="display: block;">
<!----%>
</div>
<div class="span10">
<div class="row">
<div id="dvMsgBox" class="span12 ordersave" runat="server" style="display:
none;">
<asp:Label ID="Label1" runat="server" Text=""></asp:Label>

```

```

        </div>
    </div>
</div>
</div>
</div>
</div>
</asp:Content>

<asp:Content ID="Content4" ContentPlaceHolderID="CPH_LeftSideBar" runat="Server">
    <div id="LeftSideBar" runat="server" class="span2">
        <asp:Table ID="Table1" runat="server">
            </asp:Table>
        </div>
    </asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="CPH_MainContent" runat="Server">

    <div class="span10" id="MainContent" runat="server">
        <div id="ShowPageHeader" class="row" style="display: block;" runat="server">
            <div id="PageName" class="span12 customer" runat="server">
                </div>
            </div>

            <div id="Div1" class="row" style="display: block;">
                <div id="Div2" class="span12 customertable">

                    <!-- Main Content Comes here-->

                    <%--<asp:ScriptManager ID="ScriptManager1"
runat="server"></asp:ScriptManager>--%>

                    <div class="col-xs-12 col-sm-12 col-md-6 product-image">
                        <%--
                            <div class="product-carousel zoom-view carousel slick-initialized slick-
slider" data-fade="true" data-load-on-demand="true" data-arrows="true" style="height:
auto;">--%>

                            <div id="CartData" runat="server">

<div class='pure-u-1-3 vendor-slider-thumb'>
<div>
    <asp:ImageButton ID="ImageButton2" ImageUrl="images/PayPal.png"
OnClick="ImageButton2_Click" runat="server" />
    <asp:ImageButton ID="ImageButton3" ImageUrl="images/mobikwik.png"
OnClick="ImageButton3_Click" runat="server" />
    <asp:ImageButton ID="ImageButton4" ImageUrl="images/PayPal.png"
OnClick="ImageButton4_Click" runat="server" />
    <asp:ImageButton ID="ImageButton5" ImageUrl="images/PayPal.png"
OnClick="ImageButton5_Click" runat="server" />

</div>

                            </div>
                        <%-- </div>--%>
                    </div>
                </div>
            </div>
        </div>
    </div>

```

```

</div>
<!-- product details price ect begins here-->
<div class="full-width col-xs-12 col-sm-12 col-md-4 product-info">
  <div class="info-block">
    <div id="CartAmt" runat="server">

      </div>

    </div>
    <asp:ImageButton ID="ImageButton1" ImageUrl="images/continue.png"
  OnClick="ImageButton1_Click" Width="200px" runat="server" />
  </div>

</div>
</div>
</div>
</div>
</asp:Content>

```

***** PayOut.aspx.cs*****

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBEntity;
using DowBAL;
using System.Data;
using System.Text.RegularExpressions;
using System.Text;
//using LogTracer;
public partial class PayOut: System.Web.UI.Page
{
  DDLogTracer objlog = new DDLogTracer();

  protected void Page_Load(object sender, EventArgs e)
  {
    //if (!IsPostBack)
    //{
      ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
      if (Clsusr != null)
      {
        string pageid = "0";// Request.QueryString["pageid"];
        ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
        Convert.ToInt32(Clsusr.GroupId));
        string msg = Request.QueryString["msg"];
        if (msg != null)
        {
          Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
        }
        Session["Amt"] = "0";
      }
    }
  }
}

```

```

        Bindetails();
    }
    else
    {
        Response.Redirect("MLB_Login.aspx");
    }
}
private void Bindetails()
{
    //try
    //{
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];

        string PreHtml = "";
        StringBuilder sb = new StringBuilder();

        List<EteMyUserCart> Lst = (new
Products()).GetCartbyUserId(Convert.ToInt16(Clsusr.UserId));
        string title = "My Cart ";
        decimal price = 0;
        if (Lst != null && Lst.Count > 0)
        {
            string href = "ProductSearchList.aspx?id=" + Lst[0].UCId;

            foreach (var dt in Lst)
            {
                price += dt.Price;
            }
            bool chk = true;
            int srno = 1;
            sb.Clear();
            sb.Append("\n");
            List<Etetbluser_address> add = new
tblUsers().getusersaddressbyuserid(Convert.ToInt16(Clsusr.UserId), "");
            foreach (var dt in add)
            {
                if (dt.IsDefault==true) {

                    sb.Append("<div class='pure-u-2-3 vendor-slider-thumb'\>\n");
                    //sb.Append("<div>");
                    // sb.Append("<asp:RadioButton ID='RadioButton' + srno + \" Checked='\" + chk + \"
runat='server' Visible='true' Text='\" + dt.address_typedesc + \" width='200px' />\n");
                    sb.Append("<asp:CheckBox ID='CheckBox' + srno + \" runat='server'
CssClass='checkbox' Text='Office' OnCheckedChanged='javascript:ChkClick(\" + srno + \")'
/>");

                    sb.Append("<div class='item-subtitle'> <span>\" + dt.address_typedesc + \" </span>
</div>\n");
                    sb.Append("<div class='item-subtitle'> <span>\" + dt.address1 + \" </span>
</div>\n");
                    sb.Append("<div class='item-subtitle'> <span>\" + dt.address2 + \" </span>
</div>\n");
                }
            }
        }
    }
}

```

```

        sb.Append("<div class='item-subtitle'> <span>" + dt.address3 + " </span>
</div>\n");
        sb.Append("<div class='item-subtitle'> <span>" + dt.city + " </span> </div>\n");
        sb.Append("<div class='item-subtitle'> <span>" + dt.countryname + " </span>
</div>\n");
        sb.Append("<div class='item-subtitle'> <span>" + "Contact No:" + dt.ContactNo + "
</span> </div>\n");

        //sb.Append("</div>");

        sb.Append("</div>\n");
        srno++;
        chk = false;
    }

}
PreHtml = sb.ToString();
if (PreHtml.Length > 0)
{
    // objlog.Trace_Log(4, "Gordersearchlist", "1", "SliderList", PreHtml);
    //CartData.InnerHtml = PreHtml;
}

decimal discountprc = 5;
decimal discount = Math.Round(price * discountprc / 100, 0);
decimal amt = price - discount;
sb.Clear();
sb.Append("\n");
sb.Append("<table class='table'>");
sb.Append("<tr><td style='font-size: 12px; text-align: left; padding-left: 15px; border:
0px; padding-top: 5px; padding-bottom: 5px; color: blue; font-weight: 600; '> Total Amt :
</td> <td>" + price + " </td></tr>");
sb.Append("<tr><td style='font-size: 12px; text-align: left; padding-left: 15px; border:
0px; padding-top: 5px; padding-bottom: 5px; color: blue; font-weight: 600; '> Discount: @ " +
discountprc + "% </td> <td>" + discount + " </td></tr>");
sb.Append("<tr><td style='font-size: 12px; text-align: left; padding-left: 15px; border:
0px; padding-top: 5px; padding-bottom: 5px; color: blue; font-weight: 600; '>Amount to Pay:
</td> <td>" + amt + " </td></tr>");

sb.Append("</table>");
Session["Amt"] = amt.ToString();
CartAmt.InnerHtml = sb.ToString();

}
}

private void ManageNavigations(int pageid, int userid, int groupid)
{
    bool IsLeftBar = false;
    IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
myBtnClickHandler);
}

```



```

        break;
    }
}
}

//protected override void OnInit(EventArgs e)
//{
//    base.OnInit(e);

//    if (this.Master is MainMaster)
//    {
//        ((MainMaster)this.Master).MasterControlClicked += new
EventHandler(ContentControlClicked);
//    }
//}
private void ContentControlClicked(object sender, EventArgs e)
{
    string pageid = "pageid=" + Session["pageid"].ToString();
    Response.Redirect("~/Managedealer.aspx?PageType=Add&dealername=0&" +
pageid);
}
protected void ImageButton1_Click(object sender, ImageClickEventArgs e)
{
    decimal amt =Convert.ToDecimal( Session["Amt"].ToString());
    Response.Redirect("~/posttozaakpay.aspx?PageType=Pay&Amt=" + amt);
}
protected void ImageButton2_Click(object sender, ImageClickEventArgs e)
{
}
protected void ImageButton3_Click(object sender, ImageClickEventArgs e)
{
}
protected void ImageButton4_Click(object sender, ImageClickEventArgs e)
{
}
protected void ImageButton5_Click(object sender, ImageClickEventArgs e)
{
}
}
***** PostProducts.aspx*****
<%@ Page Title="" Language="C#" MasterPageFile="~/ProductMaster.master"
AutoEventWireup="true" CodeFile="PostProducts.aspx.cs" Inherits="PostProducts" %>

```

```

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
  <script type="text/javascript">
    function checkTextAreaMaxLength(textBox, e, length) {

      var mLen = textBox["MaxLength"];
      if (null == mLen)
        mLen = length;

      var maxLength = parseInt(mLen);
      if (!checkSpecialKeys(e)) {
        if (textBox.value.length > maxLength - 1) {
          if (window.event)//IE
            e.returnValue = false;
          else
            e.preventDefault();
        }
      }
    }

    function isNumberKey(evt) {
      var charCode = (evt.which) ? evt.which : event.keyCode;
      if (charCode != 46 && charCode > 31 && (charCode < 48 || charCode > 57)) {
        return false;
      } else {
        // If the number field already has . then don't allow to enter . again.
        if (evt.target.value.search(/\./) > -1 && charCode == 46) {
          return false;
        }
        return true;
      }
    }
  </script>
  <style type="text/css">
    .auto-style1 {
      width: 220px;
      height: 48px;
    }
    .auto-style2 {
      height: 48px;
    }
  </style>
  <script type="text/javascript" src="js/jquery-ui.min.js"></script>

  <style type="text/css">
    .line {
      /*line-height: 40px;*/
      background: #09F;
      /* word-spacing: 55px; */
      width: 100%;
      margin: 0 auto;
      color: #F5F20C;
    }
  </style>

```



```
font-family: Arial, Helvetica, sans-serif;
font-size: 9px;
font-weight: bold;
}

.line a {
  /*padding: 10px 0px 10px 0px;*/
  color: #FFF;
  text-decoration: underline;
  font-weight: normal;
}

.line a:hover {
  text-decoration: none;
}

.text {
  padding: 3px 0px;
}
</style>
```

```
<style type="text/css">
.line {
  /*line-height: 40px;*/
  background: #09F;
  /* word-spacing: 55px; */
  width: 100%;
  margin: 0 auto;
  color: #F5F20C;
  font-family: Arial, Helvetica, sans-serif;
  font-size: 9px;
  font-weight: bold;
}

.line a {
  /*padding: 10px 0px 10px 0px;*/
  color: #FFF;
  text-decoration: underline;
  font-weight: normal;
}

.line a:hover {
  text-decoration: none;
}

.text {
  padding: 3px 0px;
}
</style>
```

</asp:Content>

```
<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
  <asp:Label ID="Label2x" runat="server" Height="10px"
  Text=""></asp:Label>
```

```

<div class="container-fluid">
  <div class="row-fluid" style="padding-top: 20px;">
    <div>
      <h5 class="my-4 dark-grey-text font-weight-bold" style="padding-left:20px">Manage
Products
      <asp:Button ID="btn_save" runat="server" CssClass="btn btn-secondary btn-md btn-
rounded" Text="Save" OnClick="btn_save_Click" />
<%-- <a href="#" class="btn btn-secondary btn-md btn-rounded"> &nbsp;&nbsp;&nbsp;Edit</a> <%--
<i class="fas fa-edit"></i>
      <a href="#" class="btn btn-secondary btn-md btn-rounded"> &nbsp;&nbsp;&nbsp;Delete</a><%--<i
class="fas fa-trash"></i>--%>
    </h5>
    <!-- Navbar links -->

      </div>
    <div class="span10">
      <div class="row">
        <div id="dvMsgBox" class="span12 ordersave" runat="server" style="display:
none;">
          <asp:Label ID="Label1" runat="server" Text=""></asp:Label>
        </div>
      </div>
    </div>
  </div>

  <%-- <div id="LeftSideBar" runat="server" class="span2">
    <asp:Table ID="Table1" runat="server">
      </asp:Table>
    </div>--%>
  </div>
</asp:Content>
<%--<asp:Content ID="Content4" ContentPlaceHolderID="CPH_LeftSideBar"
runat="Server">
  <div id="Div3" runat="server" class="span2">
    <asp:Table ID="Table2" runat="server" Width="250px">
      </asp:Table>
    </div>
</asp:Content>--%>
<asp:Content ID="Content2" ContentPlaceHolderID="CPH_MainContent" runat="Server" >

  <div class="col-lg-12" id="MainContent" runat="server" style="padding-left:10px">
  <%-- <div id="ShowPageHeader" class="row" style="display: block;" runat="server">
    <div id="PageName" class="span10 alert-heading" runat="server">
      </div>
    </div>--%>
</div>
  <%--<div id="Div1" class="row" style="display: block;" >
    <div id="Div2" class="col-lg-12">--%>
    <div class="main-contain">
      <div class="container">
        <div class="row">

  <%-- <div class="tabs_holder tabs_wrapper skin7 top" style="display: block;
overflow: hidden; position: relative;">

```

```

<div class="content_holder">
  <div style="overflow: hidden; position: relative; z-index: 1; padding: 10px
0px;" class="content_holder_inner">
    <div id="your-tab-id-1" class="tab_content" style="display: block;">
      <div class="inner_content">
        <div style="background-color: #f8f9fa;" class="span12
customertable">
          <div class="row-fluid">
            <div style="background-color: #ffffff; border: 1px solid #eefef;
border-top: none; padding: 15px 5px;" class="span12 detailview">
--%>
          <aside>
            <div id="ActiveProducts" class=" p-1 sidebar" runat="server">

          </div>

        </aside>
        <div class="col-md-9 col-sm-12 col-xs-12">
          <div class="row hidden-xs">
            <div class="toolbar">

                                <div class=" col-lg-12 col-md-12 col-sm-10 category-
num">
                                  <table id="Table5" class="table table-striped table-detail
col-lg-12 col-md-12 col-sm-12" aria-disabled="false" runat="server">
                                    <tbody>
                                      <tr>
                                        <td class="span3" dir="auto" style="width:auto">Product
Category : </td>
                                        <td class="span3">
                                        <asp:DropDownList ID="ddl_productgroup" runat="server"
Visible="true" Enabled="false">
                                          <asp:ListItem Value="0">Blazers</asp:ListItem>
                                          <asp:ListItem Value="1" >Wedding</asp:ListItem>
                                        </asp:DropDownList>
                                        </td>
                                      </tr>
                                      <tr>
                                        <td class="span3" dir="auto">Product Name :</td>
                                        <td>
                                          <asp:TextBox ID="Details" runat="server" Width="600px"
Height="100px" Rows="7" style="resize:none" TextMode="MultiLine"
MaxLength="250"></asp:TextBox>
                                        </td>
                                      </tr>
                                      <tr>
                                        <td class="span3" dir="auto">Product Description :</td>
                                        <td>

```

```

        <asp:TextBox ID="Description" runat="server"
Width="600px" Height="100px" Rows="7" style="resize:none" TextMode="MultiLine"
MaxLength="250"></asp:TextBox>
        </td>

</tr>

<tr>
    <td class="span3" dir="auto">Seller</td>
    <td class="form-control search-slt ">
        <asp:DropDownList ID="ddl_seller"
runat="server"></asp:DropDownList>
    </td>
</tr>

<tr>
    <td class="span3" dir="auto">Price:</td>
    <td>
        <asp:TextBox ID="txt_price" runat="server" MaxLength="40"
Width="100px" ></asp:TextBox>

        <asp:Label ID="Label2" runat="server" Text="Currency: "
Width="80px"></asp:Label>

        <asp:DropDownList ID="ddl_currency" runat="server"
Width="100px"></asp:DropDownList>

        <asp:Label ID="Label5" runat="server" Text=" Packing : "
Width="80px"></asp:Label>

        <asp:DropDownList ID="ddl_packing" runat="server"
Width="100px"></asp:DropDownList>

    </td>
</tr>

<tr>
    <td class="span3" dir="auto">Keywords</td>
    <td>
        <asp:TextBox ID="txt_taskdetails" runat="server"
Width="600px" MaxLength="40" ></asp:TextBox>
    </td>
</tr>

<tr>
    <td class="span2" dir="auto">Min Order(in Units):</td>
    <td>
        <asp:TextBox ID="txt_minorder" runat="server" MaxLength="20"
Width="150px" Text="1"></asp:TextBox>

        <asp:Label ID="Label3" runat="server" Text=" Lead Time(in days):
"></asp:Label>
        <asp:TextBox ID="txt_LeadTime" runat="server" MaxLength="20"
Width="150px" Text="1"></asp:TextBox>

    </td>
</tr>

```

```

        </tr>
<tr>
        <td class="span2" dir="auto">Location:</td>
        <td>
                <asp:TextBox ID="Location" runat="server" MaxLength="50"
Width="150px" Text="1"></asp:TextBox>
        </td>
</tr>
        <tr>
                <td>
                        <asp:Label ID="txt_psid" runat="server" Text="0"
Visible="true"></asp:Label>
                </td>
</tr>
</tbody>
</table>

```

```

        </div>
</div>

```

```

        </div>
<!-- /.content_holder -->
<div style="height: 0px; font-size: 1px; line-height: 1px; clear:
both;"></div>
</div>

```

```

<%--                </div>

```

```

        </div>
</div>
</div>
</div>--%>

```

```

</div>
</div>
</div>
</asp:Content>

```

```

<asp:Content ID="cont3" ContentPlaceHolderID ="ContentPlaceHolder1" runat="server">
        <div class="col-md-9 col-sm-12 col-xs-12">
                <div class="row hidden-xs">
                        <div class="toolbar">

```

```

<h3 style="padding-left:100px" class='shop-product-title'>Enter Stocks</h3> <%--for each
Size and Color--%>
        <div id="Stocks" class="col-sm-4 category-num" runat="server" style="padding-
left:100px;" >
                <asp:Table ID="tstocks" runat="server" border="1" CssClass="align-items-sm-
center" >
                        </asp:Table>

```

```

</div>
<div id="Div1" class="col-sm-4 category-num" runat="server" style="padding-left:100px;"
>

<table id="pics" style="padding-left:50px" runat="server">
  <tbody>
    <tr>
      <td class="col-lg-12 col-md-12 col-sm-12 p-0" dir="auto">Pictures</td>
      <td>

</td>

          </tr>
          <tr>

              <td > </td><td>
                <asp:FileUpload ID="FileUpload1" runat="server" />

                <asp:Button ID="btnupload" runat="server"
OnClick="btnupload_Click" Text="Upload" />&nbsp; <!--Height="25px" Width="75px" --%>
                </td>
              </tr>

          <tr>

              <td>
                <asp:image ToolTip = "ASP Image Control" ID="Image1" runat="server"
ImageUrl = "ManageProducts.aspx?ImageID=1" Height="156px" Width="174px"
Visible="false"></asp:image> </td>
                <td> <asp:image ToolTip = "ASP Image Control" ID="Image2" runat="server"
ImageUrl = "ManageProducts.aspx?ImageID=2" Height="156px" Width="174px"
Visible="false"></asp:image>
                </td><td> <asp:image ToolTip = "ASP Image Control" ID="Image3"
runat="server" ImageUrl = "ManageProducts.aspx?ImageID=3" Height="156px"
Width="174px" Visible="false"> </asp:image>
                </td><td> <asp:image ToolTip = "ASP Image Control" ID="Image4"
runat="server" ImageUrl = "ManageProducts.aspx?ImageID=4" Height="156px"
Width="174px" Visible="false"> </asp:image>
                </td><td> <asp:image ToolTip = "ASP Image Control" ID="Image5"
runat="server" ImageUrl = "ManageProducts.aspx?ImageID=5" Height="156px"
Width="174px" Visible="false"> </asp:image>

              </td>
            </tr>
          <tr><td>Select files with Browse and Upload</td></tr>
        <!--
--%>
      </tbody>
    </table>
  </div>
</div>
  </div>
</div>
  <div class="copyright">
    <div class="text-center">

```

<p>© 2019 Copyrights All Rights Reserved @ DateOrWed.com</p>

</div>

</div>

</asp:Content>

***** PostProducts.aspx.cs*****

```
using System;
using System.Collections.Generic;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Text.RegularExpressions;
using System.Text;
using System.Security.Cryptography;
using System.Collections.Specialized;
using System.IO;
```

```
namespace Checksum
```

```
{
```

```
    public class ParamSanitizer
```

```
    {
```

```
        public static string sanitizeParam(string param)
```

```
        {
```

```
            String ret = null;
```

```
            if (param == null)
```

```
                return null;
```

```
            ret = param.Replace(">><>(){}?&* ~`!#$%^+=+|\\:'\";,\x5D\x5B]+", " ");
```

```
            return ret;
```

```
        }
```

```
        public static String SanitizeURLParam(String url)
```

```
        {
```

```
            if (url == null)
```

```
                return "";
```

```
            Match match = Regex.Match(url, "^(https?):/[a-zA-Z0-9+&@#/%?~_!|:,;]*[a-zA-Z0-9+&@#/%~_!|]", RegexOptions.IgnoreCase);
```

```
            if (match.Success)
```

```
                return url;
```

```
            else
```

```
                return "";
```

```
        }
```

```
    }
```

```
    public class ChecksumCalculator
```

```
    {
```

```

public static string toHex(byte[] bytes)
{
    StringBuilder hex = new StringBuilder(bytes.Length * 2);
    foreach (byte b in bytes)
        hex.AppendFormat("{0:x2}", b);
    return hex.ToString();
}

public static string calculateChecksum(string secretkey, string allparamvalues)
{
    byte[] dataToEncryptByte = Encoding.UTF8.GetBytes(allparamvalues);
    byte[] keyBytes = Encoding.UTF8.GetBytes(secretkey);
    HMACSHA256 hmacsha256 = new HMACSHA256(keyBytes);
    byte[] checksumByte = hmacsha256.ComputeHash(dataToEncryptByte);
    String checksum = toHex(checksumByte);

    return checksum;
}

public static Boolean verifyChecksum(String secretKey, String
allParamVauleExceptChecksum, String checksumReceived)
{
    byte[] dataToEncryptByte =
Encoding.UTF8.GetBytes(allParamVauleExceptChecksum);
    byte[] keyBytes = Encoding.UTF8.GetBytes(secretKey);
    HMACSHA256 hmacsha256 = new HMACSHA256(keyBytes);
    byte[] checksumCalculatedByte = hmacsha256.ComputeHash(dataToEncryptByte); ;
    String checksumCalculated = toHex(checksumCalculatedByte);

    if (checksumReceived.Equals(checksumCalculated))
    {
        return true;
    }
    else
    {
        return false;
    }
}

public static string getAllNotEmptyParamValue(HttpRequest Request)
{
    String allNonEmptyParamValue = "";

    NameValueCollection postedValues = Request.Form;

    String[] paramSeq = { "amount", "bankid", "buyerAddress",
"buyerCity", "buyerCountry", "buyerEmail", "buyerFirstName", "buyerLastName", "buyerPhoneN
umber", "buyerPincode",

```



```

"buyerState","currency","merchantIdentifier","merchantIpAddress","mode","orderId",
"product1Description","product2Description","product3Description","product4Description",
"productDescription","productInfo","purpose","returnUrl","shipToAddress","shipToCity","shipToCountry",
"shipToFirstname","shipToLastname","shipToPhoneNumber","shipToPincode","shipToState",
"showMobile","txnDate",
    "txnType","zpPayOption");

    foreach(String i in paramSeq)
    {

        try
        {
            String paramInArray = postedValues[i];

            if (!paramInArray.Equals("")) {
                //paramName = postedValues.AllKeys[i];

                String paramValue = ParamSanitizer.sanitizeParam(paramInArray);

                if (paramValue != null)
                {
                    allNonEmptyParamValue = allNonEmptyParamValue +i+"="+ paramValue
+ "&";
                }
            }
        }
        catch(Exception e)
        {
            Console.WriteLine("Exception caught: {0}", e);
        }
    }

    return allNonEmptyParamValue;

}
}

}
public partial class posttozaakpay : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {

    }
}
}
*****PostServices.aspx *****

```

```
<%@ Page Title="" Language="C#" MasterPageFile="~/ProductMaster.master"
AutoEventWireup="true" CodeFile="PostServices.aspx.cs" Inherits="PostServices" %>
```

```
<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
```

```
<script type="text/javascript">
```

```
function checkTextAreaMaxLength(textBox, e, length) {
```

```
    var mLen = textBox["MaxLength"];
```

```
    if (null == mLen)
```

```
        mLen = length;
```

```
    var maxLength = parseInt(mLen);
```

```
    if (!checkSpecialKeys(e)) {
```

```
        if (textBox.value.length > maxLength - 1) {
```

```
            if (window.event)//IE
```

```
                e.returnValue = false;
```

```
            else
```

```
                e.preventDefault();
```

```
        }
```

```
    }
```

```
}
```

```
function isNumberKey(evt) {
```

```
    var charCode = (evt.which) ? evt.which : event.keyCode;
```

```
    if (charCode != 46 && charCode > 31 && (charCode < 48 || charCode > 57)) {
```

```
        return false;
```

```
    } else {
```

```
        // If the number field already has . then don't allow to enter . again.
```

```
        if (evt.target.value.search(/\./) > -1 && charCode == 46) {
```

```
            return false;
```

```
        }
```

```
        return true;
```

```
    }
```

```
}
```

```
</script>
```

```
<style type="text/css">
```

```
    .auto-style1 {
```

```
        width: 220px;
```

```
        height: 48px;
```

```
    }
```

```
    .auto-style2 {
```

```
        height: 48px;
```

```
    }
```

```
</style>
```

```
<script type="text/javascript" src="js/jquery-ui.min.js"></script>
```

```
<style type="text/css">
```

```
    .line {
```

```
        /*line-height: 40px;*/
```

```
        background: #09F;
```

```
/* word-spacing: 55px; */
width: 100%;
margin: 0 auto;
color: #F5F20C;
font-family: Arial, Helvetica, sans-serif;
font-size: 9px;
font-weight: bold;
}

.line a {
  /*padding: 10px 0px 10px 0px;*/
  color: #FFF;
  text-decoration: underline;
  font-weight: normal;
}

.line a:hover {
  text-decoration: none;
}

.text {
  padding: 3px 0px;
}
</style>
```

```
<style type="text/css">
.line {
  /*line-height: 40px;*/
  background: #09F;
  /* word-spacing: 55px; */
  width: 100%;
  margin: 0 auto;
  color: #F5F20C;
  font-family: Arial, Helvetica, sans-serif;
  font-size: 9px;
  font-weight: bold;
}

.line a {
  /*padding: 10px 0px 10px 0px;*/
  color: #FFF;
  text-decoration: underline;
  font-weight: normal;
}

.line a:hover {
  text-decoration: none;
}

.text {
  padding: 3px 0px;
}
</style>

</asp:Content>
```

```

<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
    <asp:Label ID="Label2x" runat="server" Height="10px" Text=""></asp:Label>

    <div class="container-fluid">
        <div class="row-fluid" style="padding-top: 10px;">
            <div>
                <h5 class="my-4 dark-grey-text font-weight-bold" style="padding-left:20px">Manage
Products
                <asp:Button ID="btn_save" runat="server" CssClass="btn btn-secondary btn-md btn-
rounded" Text="Save" OnClick="btn_save_Click" />
                <%-- <a href="#" class="btn btn-secondary btn-md btn-rounded"> &nbsp;Edit</a> <%--
                <i class="fas fa-edit"></i>
                <a href="#" class="btn btn-secondary btn-md btn-rounded"> &nbsp;Delete</a><%--<i
class="fas fa-trash"></i>--%>
                </h5>
                <!-- Navbar links -->

                </div>
                <div class="span10">
                    <div class="row">
                        <div id="dvMsgBox" class="span12 ordersave" runat="server" style="display:
none;">
                            <asp:Label ID="Label1" runat="server" Text=""></asp:Label>
                        </div>
                    </div>
                </div>
            </div>

            <%-- <div id="LeftSideBar" runat="server" class="span2">
                <asp:Table ID="Table1" runat="server">
                    </asp:Table>
                </div>--%>
            </div>
        </asp:Content>
        <%--<asp:Content ID="Content4" ContentPlaceHolderID="CPH_LeftSideBar"
runat="Server">
            <div id="Div3" runat="server" class="span2">
                <asp:Table ID="Table2" runat="server" Width="250px">
                    </asp:Table>
                </div>
            </asp:Content>--%>
        <asp:Content ID="Content2" ContentPlaceHolderID="CPH_MainContent" runat="Server" >

            <div class="col-lg-12 col-md-12 col-sm-12" id="MainContent" runat="server"
style="padding-left:10px">
                <%-- <div id="ShowPageHeader" class="row" style="display: block;" runat="server">
                    <div id="PageName" class="span10 alert-heading" runat="server">
                        </div>
                    </div>--%>
            </div>
            <%--<div id="Div1" class="row" style="display: block;" >
                <div id="Div2" class="col-lg-12">--%>
                <div class="main-contain">
                    <div class="container">

```

```

<div class="row">

<%--      <div class="tabs_holder tabs_wrapper skin7 top" style="display: block;
overflow: hidden; position: relative;">

      <div class="content_holder">
        <div style="overflow: hidden; position: relative; z-index: 1; padding: 10px
0px;" class="content_holder_inner">
          <div id="your-tab-id-1" class="tab_content" style="display: block;">
            <div class="inner_content">
              <div style="background-color: #f8f9fa;" class="span12
customertable">
                <div class="row-fluid">
                  <div style="background-color: #ffffff; border: 1px solid #eefef;
border-top: none; padding: 15px 5px;" class="span12 detailview">
                    --%>
                  <aside>
                    <div id="ActiveProducts" class="widget mb30" runat="server">
                      <%-- col-md-4 col-sm-12 col-xs-12 sidebar --%>
                    </div>

                  </aside>
                  <div class="col-md-9 col-sm-12 col-xs-12">
                    <div class="row hidden-xs">
                      <div class="toolbar">

                                <div class=" col-lg-12 col-md-12 col-sm-9 category-
num">
                                <table id="Table5" class="table table-striped table-detail
col-lg-12" runat="server">
                                  <tbody>

                                    <tr>
                                      <td class="span3" dir="auto" style="width:auto">Service
Category : </td>

                                      <td class="span3">
                                      <asp:DropDownList ID="ddl_productgroup" runat="server"
Visible="true">
                                        <asp:ListItem Value="0">Venues</asp:ListItem>
                                        <asp:ListItem Value="1" >Catering</asp:ListItem>
                                      </asp:DropDownList> </td>
                                    </tr>
                                    <tr>
                                      <td class="span3" dir="auto">Service Name :</td>
                                      <td>
                                        <asp:TextBox ID="Details" runat="server"
Height="100px" Rows="7" style="resize:none" TextMode="MultiLine" MaxLength="250"
CssClass="col-lg-12 col-md-12 col-sm-12"></asp:TextBox>
                                      </td><%--Width="600px"--%>

                                    </tr>
                                  </tbody>
                                </table>
                                <tr>

```

```

                <td class="span3" dir="auto">Service Description :</td>
                <td>
                    <asp:TextBox ID="Description" runat="server"
Width="600px" Height="100px" Rows="7" style="resize:none" TextMode="MultiLine"
MaxLength="250"></asp:TextBox>
                </td>
            </tr>
            <tr>
                <td class="span3" dir="auto">Service Provider</td>
                <td class="form-control search-slt ">
                    <asp:DropDownList ID="ddl_seller"
runat="server"></asp:DropDownList>
                </td>
            </tr>
            <tr>
                <td class="span3" dir="auto">Rate:</td>
                <td>
                    <asp:TextBox ID="txt_price" runat="server" MaxLength="40"
Width="100px" ></asp:TextBox>
                    <asp:Label ID="Label2" runat="server" Text="Currency: "
Width="80px"></asp:Label>
                    <asp:DropDownList ID="ddl_currency" runat="server"
Width="100px"></asp:DropDownList>
                    <asp:Label ID="Label5" runat="server" Text=" Rate Per: "
Width="80px"></asp:Label>
                    <asp:DropDownList ID="ddl_packing" runat="server"
Width="100px"></asp:DropDownList>
                </td>
            </tr>
            <tr>
                <td class="span3" dir="auto">Keywords</td>
                <td>
                    <asp:TextBox ID="txt_taskdetails" runat="server"
Width="600px" MaxLength="40" ></asp:TextBox>
                </td>
            </tr>
            <tr>
                <td class="span2" dir="auto">Min Order(in Units):</td>
                <td>
                    <asp:TextBox ID="txt_minorder" runat="server" MaxLength="20"
Width="150px" Text="1"></asp:TextBox>
                    <asp:Label ID="Label3" runat="server" Text=" Lead Time(in days):
"></asp:Label>
                    <asp:TextBox ID="txt_LeadTime" runat="server" MaxLength="20"
Width="150px" Text="1"></asp:TextBox>

```

```

        </td>
    </tr> <tr>
        <td class="span2" dir="auto">Location:</td>
        <td>
            <asp:TextBox ID="Location" runat="server" MaxLength="50"
Width="150px" Text=" "></asp:TextBox>
        </td>
    </tr>
    <tr>
        <td> <asp:Label ID="txt_psid" runat="server" Text="0"
Visible="false"></asp:Label>
        </td>
    </tr>
</tbody>
</table>

```

```

    </div>
</div>
<div class="toolbar">
    <div class=" col-lg-12 col-md-12 col-sm-12 category-num">
        <h3>Select Services included in the package</h3>
        <asp:Table ID="FreeService" runat="server" border="2" CssClass="table table-
striped table-detail col-lg-6 align-items-sm-center" >
            </asp:Table>
    </div>

```

```

<h3 style="padding-left:100px" class='shop-product-title'>Available Dates</h3> <!--for
each Size and Color--%>
    <div id="Stocks" class="col-sm-4 category-num" runat="server" style="padding-
left:100px;" >
        <asp:Table ID="tstocks" runat="server" border="0" CssClass="align-items-sm-
center" >
            <asp:TableHeaderRow CssClass="mdb-color darken-3">
                <asp:TableCell>select</asp:TableCell>
                <asp:TableCell>Dates</asp:TableCell>
            </asp:TableHeaderRow>
            <asp:TableRow><asp:TableCell>
                <asp:Calendar ID="Calendar1" runat="server" >
            </asp:Calendar>
            </asp:TableCell></asp:TableRow>
        </asp:Table>
    </div>
</div>
</div>
    </div>
<!-- /.content_holder -->

```

```

        <div style="height: 0px; font-size: 1px; line-height: 1px; clear:
both;"></div>
        </div>

<%--                </div>

                    </div>
                    </div>
                    </div>
                    </div>--%>

                </div>
            </div>
        </div>
</asp:Content>

<asp:Content ID="cont3" ContentPlaceHolderID ="ContentPlaceHolder1" runat="server">
    <div class="col-md-9 col-sm-12 col-xs-12">
        <div class="row hidden-xs">
            <div class="toolbar">

                <div id="Div1" class="col-sm-4 category-num" runat="server" style="padding-
left:100px;" >

                    <table id="pics" style="padding-left:50px" runat="server">
                        <tbody>
                            <tr>
                                <td class="col-lg-12 col-md-12 col-sm-12 p-0" dir="auto">Pictures</td>
                                <td>

</td>

                                </tr>
                                <tr>

                                    <td > </td><td>
                                    <asp:FileUpload ID="FileUpload1" runat="server" />

                                    <asp:Button ID="btnupload" runat="server"
OnClick="btnupload_Click" Text="Upload" />&nbsp; <%--Height="25px" Width="75px" --%>
                                    </td>
                                </tr>

                                <tr>

                                    <td>
                                    <asp:image ToolTip = "ASP Image Control" ID="Image1" runat="server"
ImageUrl = "ManageProducts.aspx?ImageID=1" Height="156px" Width="174px"
Visible="false"></asp:image> </td>
                                    <td> <asp:image ToolTip = "ASP Image Control" ID="Image2" runat="server"
ImageUrl = "ManageProducts.aspx?ImageID=2" Height="156px" Width="174px"
Visible="false"></asp:image>

```



```

</td><td>                <asp:image ToolTip = "ASP Image Control" ID="Image3"
runat="server" ImageUrl ="ManageProducts.aspx?ImageID=3" Height="156px"
Width="174px" Visible="false"> </asp:image>
</td><td>                <asp:image ToolTip = "ASP Image Control" ID="Image4"
runat="server" ImageUrl ="ManageProducts.aspx?ImageID=4" Height="156px"
Width="174px" Visible="false"> </asp:image>
</td><td>                <asp:image ToolTip = "ASP Image Control" ID="Image5"
runat="server" ImageUrl ="ManageProducts.aspx?ImageID=5" Height="156px"
Width="174px" Visible="false"> </asp:image>

                </td>

        </tr>
<%--                <tr><td>Select files with Browse and Upload</td></tr>
--%>
    </tbody>
    </table>
</div>
</div>
    </div>
</div>
        <div class="copyright">
            <div class="text-center">
                <p>© 2019 Copyrights All Rights Reserved @ <a target="_blank"
href="Home.aspx">DateOrWed.com</a></p>
            </div>
        </div>
</asp:Content>
*****PostServices.aspx.cs *****
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.Security;
using DowBAL;
using System.Data;
using System.Text;
using System.Net;
using DowBEntity;
using System.Data.SqlClient;
using System.Configuration;
//using LogTracer;
using System.Xml;
using System.Xml.Serialization;
using System.IO;
using System.Drawing;

public partial class PostServices : System.Web.UI.Page
{
    public string PageType="New";
    string pageid;
    int Id=0;
    int UserId = 0;

```

```

DDLogTracer objlog = new DDLogTracer();
decimal psid = 0;
protected void Page_Load(object sender, EventArgs e)
{
    bool IsMobile = false;
    if (Request.Browser.IsMobileDevice)
    { IsMobile = true; }
    else { IsMobile = false; }
    var sb = new StringBuilder();

    //try
    {

        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr != null)
        {
            UserId = Convert.ToInt16(Clsusr.UserId);
            pageid = "4";// Request.QueryString["pageid"];
            ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
Convert.ToInt32(Clsusr.GroupId), PageType);
        }
        else
        {
            Response.Redirect("MLB_Login.aspx");

        }
        PageType = Page.Request.QueryString["pagetype"];
        if (PageType == null)
        {
            PageType = "Add";
            Id = Convert.ToInt16(Request.QueryString["id"]);
            if (Id == 0) { Table5.Disabled = false; }

        }
        //if (PageType == "Add")
        {
            Session["PageType"] = PageType;
            Id = Convert.ToInt16(Request.QueryString["id"]);
            PopDlIs ObjPD = new PopDlIs();
            ObjPD.BindDropDowns(ddl_productgroup, (new
PopDlIs()).GetProductbyUserIdType(Convert.ToInt16(Clsusr.UserId),2));
            ObjPD.BindDropDowns(ddl_seller, (new PopDlIs()).GetUserCompany());
            ObjPD.BindDropDowns(ddl_currency, (new PopDlIs()).GetCurrency());
            ObjPD.BindDropDowns(ddl_packing, (new PopDlIs()).Gettbltblpacking());
            ddl_packing.SelectedIndex = 1;
            Id = Convert.ToInt32(Request.QueryString["id"]);
            // vendors products selected starts
            sb = new StringBuilder();
            sb.Append("      <h6 class='text-uppercase bottom-line'>Select a Service</h6>
");
            sb.Append(" <ul class='icons-list'>      ");
            List<EteSellerProduct> actpr = new Products().GetSellerProducts(UserId,2);
            foreach (var apr in actpr)
            {

```

```
sb.Append(" <li><a href='/PostServices.aspx?id=" + apr.ProductId + ">" +  
apr.ProductDesc + " <span class='pull-right'></span></a></li> ");
```

```
    }  
    sb.Append("</ul>");  
    if (!IsMobile)  
    {  
        ActiveProducts.InnerHtml = sb.ToString();  
    }  
    // vendors products selected ends  
  
    if (PageType == "Add")  
    {  
        decimal PSId = new Products().GetNextPSId();  
        if (Convert.ToInt16(txt_psid.Text) == 0)  
        {  
            txt_psid.Text = PSId.ToString();  
        }  
        if (Id > 0)  
        {  
            Eteproduct ep = new Products().GetProduct(Id, "");  
            int epid = ep.DealerId;  
            ddl_productgroup.SelectedValue = Request.QueryString["id"]; // Id; //  
            ddl_productgroup.Enabled = true;  
            string sId = ddl_productgroup.SelectedValue;  
        }  
        else  
        {  
            ddl_productgroup.Enabled = true;  
        }  
        ddl_seller.SelectedValue = Clsusr.UserId;  
        //bool IsLeftBar = false;  
        //IsLeftBar = Navigations.Generate_Color_sizes(UserId, tstocks);  
        List<EteSellerProduct> freeser = new Products().GetSellerProducts(UserId,
```

2);

```
        TableRow tr = new TableRow();  
        TableCell td = new TableCell();  
        CheckBox chk = new CheckBox();  
        FreeService.Rows.Clear();  
        TableHeaderRow thr = new TableHeaderRow();  
        thr.CssClass = "mdb-color darken-3";  
        td.Text = "Select";  
        thr.Cells.Add(td);  
        td = new TableCell();  
        td.Text = "Service";  
        thr.Cells.Add(td);  
        FreeService.Rows.Add(thr);  
  
        foreach (var dt in freeser)  
        {  
            if (dt.ProductId != ddl_productgroup.SelectedIndex)  
            {  
                tr = new TableRow();
```

```

        td = new TableCell();
        chk = new CheckBox();
        chk.ID = "CheckBox" + dt.ProductId;
        chk.Checked = false;
        td.Controls.Add(chk);
        tr.Cells.Add(td);
        td = new TableCell();
        td.Text = "<span class='badge badge-primary'>" + dt.ProductDesc+
"PostProduct");
    "</span>";
        tr.Cells.Add(td);
        FreeService.Rows.Add(tr);
    } //if data

    } //for dt

    } //if add
    else
    if (PageType == "Edit")
    {
        ObjPD.BindDropDowns(ddl_productgroup, (new PopDlls()).GetProduct());

        psid =Convert.ToDecimal( Request.QueryString["psid"]);
        EteproductSeller Prd = new Products().GetProductSeller(psid,
        "PostProduct");

        if (Prd != null)
        {
            txt_psid.Text = Prd.PSId.ToString();

            Details.Text = Prd.SellerDescription;
            Description.Text = Prd.ProductDescription;
            ddl_productgroup.SelectedValue = Prd.ProductId.ToString();
            ddl_productgroup.Enabled = false;
            ddl_seller.SelectedValue = Prd.UserId.ToString();
            ddl_seller.Enabled = false;
            ddl_packing.SelectedIndex = (int)Prd.PricePackId;
            txt_price.Text = Prd.Price.ToString();
            int CurrencyId = (int)Prd.CurrencyId;
            txt_taskdetails.Text = Prd.KeyWords;
            txt_minorder.Text = Prd.MinOrder.ToString();
            txt_LeadTime.Text = Prd.DelTlme.ToString();
            int sno = 0;
            List<EteproductImages> imgs =new
            Products().GetProductSellerImage(psid, "");
            foreach (var img1 in imgs)
            {
                sno++;
                if (sno == 1)
                {
                    Image1.ImageUrl = img1.ImagePath;
                    Image1.Visible = true;
                }
                if (sno == 3)
                {

```

```

        Image3.ImageUrl = img1.ImagePath;
        Image3.Visible = true;
    }
    if (sno == 2)
    {
        Image2.ImageUrl = img1.ImagePath;
        Image2.Visible = true;
    }
    if (sno == 4)
    {
        Image4.ImageUrl = img1.ImagePath;
        Image4.Visible = true;
    }
} //for images

} //if not null daata

List<Eteproduct> freeser = new Products().GetSellerServicesbypsid(psid);
TableRow tr = new TableRow();
TableCell td = new TableCell();
CheckBox chk = new CheckBox();
FreeService.Rows.Clear();
TableHeaderRow thr = new TableHeaderRow();
thr.CssClass = "mdb-color darken-3";

td.Text = "Select";
thr.Cells.Add(td);
td = new TableCell();
td.Text = "Service";
thr.Cells.Add(td);
FreeService.Rows.Add(thr);
foreach (var dt in freeser)
{
    //if (dt.ProductId != ddl_productgroup.SelectedIndex)
    {
        tr = new TableRow();
        td = new TableCell();
        chk = new CheckBox();
        chk.ID = "CheckBox" + dt.ProductId;
        chk.Checked = true;
        td.Controls.Add(chk);
        tr.Cells.Add(td);
        td = new TableCell();
        td.Text = "<span class='badge badge-primary'>" + dt.ProductName +
"</span>";
        tr.Cells.Add(td);
        FreeService.Rows.Add(tr);
    }
} //foreach freeservices
if (freeser.Count() == 0)
{
    List<EteSellerProduct> freeser1 = new
Products().GetSellerProducts(UserId, 2);
    foreach (var dt in freeser1)

```

```

        {
            //if (dt.ProductId != ddl_productgroup.SelectedIndex)
            {
                tr = new TableRow();
                td = new TableCell();
                chk = new CheckBox();
                chk.ID = "CheckBox" + dt.ProductId;
                chk.Checked = false;
                td.Controls.Add(chk);
                tr.Cells.Add(td);
                td = new TableCell();
                td.Text = "<span class='badge badge-primary'>" + dt.ProductDesc +
"</span>";
                tr.Cells.Add(td);
                FreeService.Rows.Add(tr);
            }
        } //foreach freeservices
    }

} //if edit
}
}

//catch (Exception ex)
//{
//    Common.ShowMessageHtml(ex.Message, 1, true, dvMsgBox);
//    ExceptionLogging.SendExcepToDB(ex);

//    //objlog.Trace_Error(5, "DateOrWed", "PostProduts", "pageLoad", ex.Message);

//}
}
protected void btn_save_Click(object sender, EventArgs e)
{
    decimal PSId = 0;
    if (txt_psid.ToString() == "0") {
        PageType = "Add";
        PSId = new Products().GetNextPSId();
        txt_psid.Text = PSId.ToString();
    }
    //else { PageType = "Edit"; }
    PSId = Convert.ToDecimal(txt_psid.Text);

    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    try
    {
        if (Details.Text == "")

```

```

    {
        Common.ShowMessageHtml("Product Description cannot be blank", 0, true,
dvMsgBox);

    }
    else if (txt_taskdetails.Text == "")
    {

        Common.ShowMessageHtml("Keywords cannot be blank", 0, true, dvMsgBox);

    }
    else
    {
        bool Result = Insertupdate();

        if (Result == true)
        {
            if (PageType == "Add")
            {
                bool ret = false;
                CheckBox cbox = new CheckBox();
                cbox = (CheckBox)FreeService.FindControl("CheckBoxAll");
                List<EteSellerProduct> freeser = new Products().GetSellerProducts(UserId,
2);

                foreach (var list in freeser)
                {
                    cbox = (CheckBox)FreeService.FindControl("CheckBox" + list.ProductId);
                    if (cbox.Checked)
                    {
                        EteproductSellerServices data = new EteproductSellerServices();
                        data.PSId = PSId;
                        data.productid = list.ProductId;
                        data.Remark="";
                        data.PrdPrice=0;
                        data.PrdUnit=0;
                        data.active = true;
                        data.userid = Convert.ToInt16(Clsusr.UserId);
                        ret = new Products().InsertproductSellerServices(data);
                    }
                }

                Common.ShowMessageHtml("Adding of Service Sucessful ", 1, true,
dvMsgBox);

            }
            else
            {
                Common.ShowMessageHtml("Service Details Edited Sucessfully ", 1, true,
dvMsgBox);

            }
        }
    }
}

```

```

        Response.Redirect("~/VendorHome.aspx?pageid=1&msg=Product Updated
Sucessfully");
    }
}
}
catch (Exception exc)
{
    //objlog.Trace_Error(4, "DateOrWed", "MangeDealer", "Load", "Exception:" +
exc.ToString());
    Common.ShowMessageHtml(exc.ToString(), 0, true, dvMsgBox);
}
}
}

private void ManageNavigations(int pageid, int userid, int groupid, string PageType)
{
}

public byte[] GetPhtoStream()
{
    byte[] bufferPhoto = new byte[FileUpload1.PostedFile.ContentLength];
    Stream photoStream = FileUpload1.PostedFile.InputStream;
    photoStream.Read(bufferPhoto, 0, FileUpload1.PostedFile.ContentLength);

    return bufferPhoto;
}

protected void btnupload_Click(object sender, EventArgs e)
{
    string path = FileUpload1.PostedFile.FileName;// FileUpload1.FileName;
    if (path == "") { return; }
    byte[] Photo = GetPhtoStream();
    Stream strm = FileUpload1.PostedFile.InputStream;

    Id = Convert.ToInt32(Request.QueryString["id"]);
    string base64String = Convert.ToBase64String(Photo, 0, Photo.Length);
    string imageUrl = "data:image/png;base64," + base64String;
    if (txt_psid.Text.ToString() == "0")
    {
        Insertupdate();
    }
    EteproductImages ep = new EteproductImages();
    ep.PSId =Convert.ToDecimal( txt_psid.Text);
    ep.ProductId = ddl_productgroup.SelectedIndex; ;
    if (ep.ProductId == 0) { return; }
    ep.SellerId = ddl_seller.SelectedIndex;
}

```



```

ep.srno = 1;
ep.Images = Photo;
ep.ImageUrls = ImageUrl;
ep.ImagePath = path;

string Photold = new Products().InsertPhoto(ep);
if (Photold != "")
{
    string server = Server.MapPath(Photold);
    FileUpload1.SaveAs(server);
    string targetPath = Server.MapPath("thumb/"+Photold);

    ReducelImageSize(0.2, strm, targetPath);

}
string Pid = Session["Pid"].ToString();
if(Pid=="1")
{
    Image1.ImageUrl = ImageUrl;
    Image1.Visible = true;
    Session["Pid"] = "2";

}
if (Pid == "2")
{
    Image2.ImageUrl = ImageUrl;
    Image2.Visible = true;
    Session["Pid"] = "3";

}
if (Pid == "3")
{
    Image3.ImageUrl = ImageUrl;
    Image3.Visible = true;
    Session["Pid"] = "4";

}
if (Pid == "4")
{
    Image4.ImageUrl = ImageUrl;
    Image4.Visible = true;
    Session["Pid"] = "5";

}
}

private bool Insertupdate()
{
    decimal ProductId = Convert.ToDecimal(ddl_productgroup.SelectedValue);
    if (ProductId == 0)
    {
        Common.ShowMessageHtml("Please select a Service to continue", 0, true,
dvMsgBox);
    }
}

```

```

    return false; }
    PageType = Session["PageType"].ToString();
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    decimal PSId = 0;
    EteproductSeller Prd = new EteproductSeller();
    if (PageType=="Add")
    {
        PSId = new Products().GetNextPSId();
        txt_psid.Text = PSId.ToString();
        Prd = new EteproductSeller();
    }
    else
    {
        PSId = Convert.ToDecimal(txt_psid.Text.ToString());

        Prd = new Products().GetProductSeller((int)PSId, "PostProdut");
    }
    PSId =Convert.ToDecimal( txt_psid.Text.ToString());

//Eteproduct Prd = new Eteproduct();
Prd.PSId = PSId;
Prd.ProductId =Convert.ToDecimal( ddl_productgroup.SelectedValue);
Prd.CreatedOn = DateTime.Now;
Prd.UserId = Convert.ToInt32(Clsusr.UserId);
Prd.SellerDescription = Details.Text.ToString();
Prd.ProductDescription = Description.Text;
Prd.SellerId =Convert.ToInt16( ddl_seller.SelectedValue);
if (txt_price.Text == "")
{
    Prd.Price = 0;
}
else
{
    Prd.Price = Convert.ToDecimal(txt_price.Text);
}
Prd.PriceUnits = 1;
Prd.PriceUomId = 1;
Prd.PricePackId = 1;

Prd.CurrencyId = ddl_currency.SelectedIndex;
if (Prd.CurrencyId <1) { Prd.CurrencyId = 1; }
Prd.PricePackId = ddl_packing.SelectedIndex;
if (Prd.PricePackId <1) { Prd.PricePackId = 1; }
Prd.MinOrder = Convert.ToInt16(txt_minorder.Text);
Prd.MinOrderUom = 1;
Prd.DelTime = Convert.ToInt16(txt_LeadTime.Text);
Prd.DelTimeType = 1;
Prd.PrId = 1;
Prd.LinkPSId = PSId;
Prd.MadeinCountry = 1;

Prd.AnalysisCode = txt_taskdetails.Text.ToString();
//-----
Prd.DiscountLink = 1;

```

```

Prd.BrandName = "";
Prd.MaterialCare = "";
Prd.MaterialType = "";
Prd.Manufacturer = "";
Prd.Length = "";
Prd.Location = Location.Text;
Prd.GeoLocation = " ";
bool Result = false;

if (PageType == "Add")
{
    Result = new Products().InsertProductSeller(Prd,false);
    Session["PageType"] = "Edit";
}
else
{
    Prd.ProductId = Convert.ToInt32(Request.QueryString["ProductId"]);

    //Result = new Products().InsertProductSeller(Prd, true);
    Session["PageType"] = "Edit";
}

return Result;
}

private void ReduceImageSize(double scaleFactor, Stream sourcePath, string targetPath)
{
    using (var image = System.Drawing.Image.FromStream(sourcePath))
    {
        var newWidth = (int)(image.Width * scaleFactor);
        var newHeight = (int)(image.Height * scaleFactor);
        var thumbnailImg = new Bitmap(newWidth, newHeight);
        var thumbGraph = Graphics.FromImage(thumbnailImg);
        thumbGraph.CompositingQuality =
System.Drawing.Drawing2D.CompositingQuality.HighQuality;
        thumbGraph.SmoothingMode =
System.Drawing.Drawing2D.SmoothingMode.HighQuality;
        thumbGraph.InterpolationMode =
System.Drawing.Drawing2D.InterpolationMode.HighQualityBicubic;
        var imageRectangle = new Rectangle(0, 0, newWidth, newHeight);
        thumbGraph.DrawImage(image, imageRectangle);
        thumbnailImg.Save(targetPath, image.RawFormat);
    }
}

}

***** PostServicesUser.aspx*****
<%@ Page Title="" Language="C#" MasterPageFile="~/ProductMaster.master"
AutoEventWireup="true" CodeFile="PostServicesUser.aspx.cs" Inherits="PostServicesUser"
%>

```

```

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
  <script type="text/javascript">
    function checkTextAreaMaxLength(textBox, e, length) {

      var mLen = textBox["MaxLength"];
      if (null == mLen)
        mLen = length;

      var maxLength = parseInt(mLen);
      if (!checkSpecialKeys(e)) {
        if (textBox.value.length > maxLength - 1) {
          if (window.event)//IE
            e.returnValue = false;
          else
            e.preventDefault();
        }
      }
    }

    function isNumberKey(evt) {
      var charCode = (evt.which) ? evt.which : event.keyCode;
      if (charCode != 46 && charCode > 31 && (charCode < 48 || charCode > 57)) {
        return false;
      } else {
        // If the number field already has . then don't allow to enter . again.
        if (evt.target.value.search(/\./) > -1 && charCode == 46) {
          return false;
        }
        return true;
      }
    }
  </script>
  <style type="text/css">
    .auto-style1 {
      width: 220px;
      height: 48px;
    }
    .auto-style2 {
      height: 48px;
    }
  </style>
  <script type="text/javascript" src="js/jquery-ui.min.js"></script>

  <style type="text/css">
    .line {
      /*line-height: 40px;*/
      background: #09F;
      /* word-spacing: 55px; */
      width: 100%;
      margin: 0 auto;
      color: #F5F20C;
    }
  </style>

```

```
font-family: Arial, Helvetica, sans-serif;
font-size: 9px;
font-weight: bold;
}

.line a {
  /*padding: 10px 0px 10px 0px;*/
  color: #FFF;
  text-decoration: underline;
  font-weight: normal;
}

.line a:hover {
  text-decoration: none;
}

.text {
  padding: 3px 0px;
}
</style>
```

```
<style type="text/css">
.line {
  /*line-height: 40px;*/
  background: #09F;
  /* word-spacing: 55px; */
  width: 100%;
  margin: 0 auto;
  color: #F5F20C;
  font-family: Arial, Helvetica, sans-serif;
  font-size: 9px;
  font-weight: bold;
}

.line a {
  /*padding: 10px 0px 10px 0px;*/
  color: #FFF;
  text-decoration: underline;
  font-weight: normal;
}

.line a:hover {
  text-decoration: none;
}

.text {
  padding: 3px 0px;
}
</style>
```

</asp:Content>

```
<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
  <asp:Label ID="Label2x" runat="server" Height="10px" Text=""></asp:Label>
```

```

<div class="container-fluid">
  <div class="row-fluid" style="padding-top: 10px;">
    <div>
      <h5 class="my-4 dark-grey-text font-weight-bold" style="padding-left:20px">Manage
Products
      <asp:Button ID="btn_save" runat="server" CssClass="btn btn-secondary btn-md btn-
rounded" Text="Save" OnClick="btn_save_Click" />
<%--      <a href="#" class="btn btn-secondary btn-md btn-rounded"> &nbsp;&nbsp;&nbsp;Edit</a> <%--
<i class="fas fa-edit"></i>
      <a href="#" class="btn btn-secondary btn-md btn-rounded"> &nbsp;&nbsp;&nbsp;Delete</a><%--<i
class="fas fa-trash"></i>--%>
    </h5>
    <!-- Navbar links -->

      </div>
    <div class="span10">
      <div class="row">
        <div id="dvMsgBox" class="span12 ordersave" runat="server" style="display:
none;">
          <asp:Label ID="Label1" runat="server" Text=""></asp:Label>
        </div>
      </div>
    </div>
  </div>

  <%-- <div id="LeftSideBar" runat="server" class="span2">
    <asp:Table ID="Table1" runat="server">
      </asp:Table>
    </div>--%>
  </div>
</asp:Content>
<%--<asp:Content ID="Content4" ContentPlaceHolderID="CPH_LeftSideBar"
runat="Server">
  <div id="Div3" runat="server" class="span2">
    <asp:Table ID="Table2" runat="server" Width="250px">
      </asp:Table>
    </div>
</asp:Content>--%>
<asp:Content ID="Content2" ContentPlaceHolderID="CPH_MainContent" runat="Server" >

  <div class="col-lg-12 col-md-12 col-sm-12" id="MainContent" runat="server"
style="padding-left:10px">
  <%--  <div id="ShowPageHeader" class="row" style="display: block;" runat="server">
    <div id="PageName" class="span10 alert-heading" runat="server">
      </div>
    </div>--%>
</div>
  <%--<div id="Div1" class="row" style="display: block;" >
    <div id="Div2" class="col-lg-12">--%>
    <div class="main-contain">
      <div class="container">
        <div class="row">

  <%--      <div class="tabs_holder tabs_wrapper skin7 top" style="display: block;
overflow: hidden; position: relative;">

```

```

<div class="content_holder">
  <div style="overflow: hidden; position: relative; z-index: 1; padding: 10px
0px;" class="content_holder_inner">
    <div id="your-tab-id-1" class="tab_content" style="display: block;">
      <div class="inner_content">
        <div style="background-color: #f8f9fa;" class="span12
customertable">
          <div class="row-fluid">
            <div style="background-color: #ffffff; border: 1px solid #eefef;
border-top: none; padding: 15px 5px;" class="span12 detailview">
--%>
          <aside>
            <div id="ActiveProducts" class="widget mb30" runat="server">
              <%-- col-md-4 col-sm-12 col-xs-12 sidebar --%>
            </div>

          </aside>
          <div class="col-md-9 col-sm-12 col-xs-12">
            <div class="row hidden-xs">
              <div class="toolbar">

                <div class=" col-lg-12 col-md-12 col-sm-9 category-
num">
                  <table id="Table5" class="table table-striped table-detail
col-lg-12" runat="server">
                    <tbody>

                      <tr>
                        <td class="span3" dir="auto" style="width:auto">Service
Category : </td>

                        <td class="span3">
                          <asp:DropDownList ID="ddl_productgroup" runat="server"
Visible="true">
                            <asp:ListItem Value="0">Venues</asp:ListItem>
                            <asp:ListItem Value="1" >Catering</asp:ListItem>
                          </asp:DropDownList> </td>
                        </tr>
                      <tr>
                        <td class="span3" dir="auto">Service Name :</td>
                        <td>
                          <asp:TextBox ID="Details" runat="server"
Height="100px" Rows="7" style="resize:none" TextMode="MultiLine" MaxLength="250"
CssClass="col-lg-12 col-md-12 col-sm-12"></asp:TextBox>
                        </td><%--Width="600px"--%>

                      </tr>
                      <tr>
                        <td class="span3" dir="auto">Service Description :</td>
                        <td>

```

```

        <asp:TextBox ID="Description" runat="server"
Width="600px" Height="100px" Rows="7" style="resize:none" TextMode="MultiLine"
MaxLength="250"></asp:TextBox>
        </td>

</tr>

<tr>
    <td class="span3" dir="auto">Service Provider</td>
    <td class="form-control search-slt ">
        <asp:DropDownList ID="ddl_seller"
runat="server"></asp:DropDownList>
    </td>
</tr>

<tr>
    <td class="span3" dir="auto">Rate:</td>
    <td>
        <asp:TextBox ID="txt_price" runat="server" MaxLength="40"
Width="100px" ></asp:TextBox>

        <asp:Label ID="Label2" runat="server" Text="Currency: "
Width="80px"></asp:Label>

        <asp:DropDownList ID="ddl_currency" runat="server"
Width="100px"></asp:DropDownList>

        <asp:Label ID="Label5" runat="server" Text=" Rate Per: "
Width="80px"></asp:Label>

        <asp:DropDownList ID="ddl_packing" runat="server"
Width="100px"></asp:DropDownList>

    </td>
</tr>

<tr>
    <td class="span3" dir="auto">Keywords</td>
    <td>
        <asp:TextBox ID="txt_taskdetails" runat="server"
Width="600px" MaxLength="40" ></asp:TextBox>
    </td>
</tr>

<tr>
    <td class="span2" dir="auto">Min Order(in Units):</td>
    <td>
        <asp:TextBox ID="txt_minorder" runat="server" MaxLength="20"
Width="150px" Text="1"></asp:TextBox>

        <asp:Label ID="Label3" runat="server" Text=" Lead Time(in days):
"></asp:Label>
        <asp:TextBox ID="txt_LeadTime" runat="server" MaxLength="20"
Width="150px" Text="1"></asp:TextBox>

    </td>
</tr>

```



```

        </tr> <tr>
                <td class="span2" dir="auto">Location:</td>
                <td>
                        <asp:TextBox ID="Location" runat="server" MaxLength="50"
Width="150px" Text=" "></asp:TextBox>
                </td>
        </tr>
        <tr>
                <td>
                        <asp:Label ID="txt_psid" runat="server" Text="0"
Visible="false"></asp:Label>
                </td>
        </tr>
</tbody>
</table>

```

```

        </div>
</div>
<div class="toolbar">
        <div class=" col-lg-12 col-md-12 col-sm-12 category-num">
                <h3>Select Services included in the package</h3>
                <asp:Table ID="FreeService" runat="server" border="2" CssClass="table table-
striped table-detail col-lg-6 align-items-sm-center" >
                </asp:Table>

```

```

<h3 style="padding-left:100px" class='shop-product-title'>Available Dates</h3> <%--for
each Size and Color--%>

```

```

        <div id="Stocks" class="col-sm-4 category-num" runat="server" style="padding-
left:100px;" >
                <asp:Table ID="tstocks" runat="server" border="0" CssClass="align-items-sm-
center" >

```

```

                <asp:TableHeaderRow CssClass="mdb-color darken-3">
                        <asp:TableCell>select</asp:TableCell>
                        <asp:TableCell>Dates</asp:TableCell>

```

```

                </asp:TableHeaderRow>
                <asp:TableRow><asp:TableCell>
                        <asp:Calendar ID="Calendar1" runat="server" >

```

```

                </asp:Calendar>
                </asp:TableCell></asp:TableRow>
        </asp:Table>
</div>

```

```

</div>
</div>

```

```

        </div>
        <!-- /.content_holder -->
        <div style="height: 0px; font-size: 1px; line-height: 1px; clear:
both;"></div>
</div>

```

```

<!--                                </div>

                                </div>
                                </div>
                                </div>
                                </div>--%>

                                </div>
                                </div>
                                </div>
</asp:Content>

<asp:Content ID="cont3" ContentPlaceHolderID ="ContentPlaceHolder1" runat="server">
    <div class="col-md-9 col-sm-12 col-xs-12">
        <div class="row hidden-xs">
            <div class="toolbar">

                <div id="Div1" class="col-sm-4 category-num" runat="server" style="padding-
left:100px;" >

                    <table id="pics" style="padding-left:50px" runat="server">
                        <tbody>
                            <tr>
                                <td class="col-lg-12 col-md-12 col-sm-12 p-0" dir="auto">Pictures</td>
                                <td>

</td>

                                </tr>
                                <tr>

                                    <td > </td><td>
                                    <asp:FileUpload ID="FileUpload1" runat="server" />

                                    <asp:Button ID="btnupload" runat="server"
OnClick="btnupload_Click" Text="Upload" />&nbsp; <!--Height="25px" Width="75px" --%>
                                    </td>
                                </tr>

                                <tr>

                                    <td>
                                    <asp:image ToolTip = "ASP Image Control" ID="Image1" runat="server"
ImageUrl ="ManageProducts.aspx?ImageID=1" Height="156px" Width="174px"
Visible="false"></asp:image> </td>
                                    <td> <asp:image ToolTip = "ASP Image Control" ID="Image2" runat="server"
ImageUrl ="ManageProducts.aspx?ImageID=2" Height="156px" Width="174px"
Visible="false"></asp:image>
                                    </td><td> <asp:image ToolTip = "ASP Image Control" ID="Image3"
runat="server" ImageUrl ="ManageProducts.aspx?ImageID=3" Height="156px"
Width="174px" Visible="false"> </asp:image>

```

```

</td><td>                <asp:image ToolTip = "ASP Image Control" ID="Image4"
runat="server" ImageUrl ="ManageProducts.aspx?ImageID=4" Height="156px"
Width="174px" Visible="false"> </asp:image>
</td><td>                <asp:image ToolTip = "ASP Image Control" ID="Image5"
runat="server" ImageUrl ="ManageProducts.aspx?ImageID=5" Height="156px"
Width="174px" Visible="false"> </asp:image>

                </td>

        </tr>
<%--                <tr><td>Select files with Browse and Upload</td></tr>
--%>
    </tbody>
    </table>
</div>
</div>
    </div>
    </div>
        <div class="copyright">
            <div class="text-center">
                <p>© 2019 Copyrights All Rights Reserved @ <a target="_blank"
href="Home.aspx">DateOrWed.com</a></p>
            </div>
        </div>
</asp:Content>
***** PostServicesUser.aspx.cs*****
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.Security;
using DowBAL;
using System.Data;
using System.Text;
using System.Net;
using DowBEntity;
using System.Data.SqlClient;
using System.Configuration;
//using LogTracer;
using System.Xml;
using System.Xml.Serialization;
using System.IO;
using System.Drawing;

public partial class PostServicesUser : System.Web.UI.Page
{
    public string PageType="New";
    string pageid;
    int Id=0;
    int UserId = 0;
    DDLogTracer objlog = new DDLogTracer();
    decimal psid = 0;
    protected void Page_Load(object sender, EventArgs e)

```

```

{
    bool IsMobile = false;
    if (Request.Browser.IsMobileDevice)
    { IsMobile = true; }
    else { IsMobile = false; }
    var sb = new StringBuilder();

    //try
    {

        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr != null)
        {
            UserId = Convert.ToInt16(Clsusr.UserId);
            pageid = "4";// Request.QueryString["pageid"];
            ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
Convert.ToInt32(Clsusr.GroupId), PageType);
        }
        else
        {
            Response.Redirect("MLB_Login.aspx");
        }

        PageType = Page.Request.QueryString["pagetype"];
        if (PageType == null)
        {
            PageType = "Add";
            Id = Convert.ToInt16(Request.QueryString["id"]);
            if (Id == 0) { Table5.Disabled = false; }
        }

        //if (PageType == "Add")
        {
            Session["PageType"] = PageType;
            Id = Convert.ToInt16(Request.QueryString["id"]);
            PopDlIs ObjPD = new PopDlIs();
            ObjPD.BindDropDowns(ddl_productgroup, (new
PopDlIs()).GetProductbyUserIdType(Convert.ToInt16(Clsusr.UserId),2));
            ObjPD.BindDropDowns(ddl_seller, (new PopDlIs()).GetUserCompany());
            ObjPD.BindDropDowns(ddl_currency, (new PopDlIs()).GetCurrency());
            ObjPD.BindDropDowns(ddl_packing, (new PopDlIs()).Gettbltblpacking());
            ddl_packing.SelectedIndex = 1;
            Id = Convert.ToInt32(Request.QueryString["id"]);
            // vendors products selected starts
            sb = new StringBuilder();
            sb.Append("      <h6 class='text-uppercase bottom-line'>Select a Service</h6>
");
            sb.Append(" <ul class='icons-list'>      ");
            List<EteSellerProduct> actpr = new
Products().GetSellerProducts(Convert.ToInt16(Session["VendorCode"]), 2);
            foreach (var apr in actpr)
            {
                sb.Append("      <li><a href='/PostServices.aspx?id=" + apr.ProductId + ">" +
apr.ProductDesc + " <span class='pull-right'></span></a></li>      ");
            }
        }
    }
}

```

```

}
sb.Append("</ul>");
if (!IsMobile)
{
    ActiveProducts.InnerHtml = sb.ToString();
}
// vendors products selected ends

if (PageType == "Add")
{
    decimal PSId = new Products().GetNextPSId();
    if(Convert.ToInt16(txt_psid.Text)==0)
    {
        txt_psid.Text = PSId.ToString();
    }
    if (Id > 0)
    {
        Eteproduct ep = new Products().GetProduct(Id, "");
        int epid = ep.DealerId;
        ddl_productgroup.SelectedValue = Request.QueryString["id"]; // Id; //
        ddl_productgroup.Enabled = true;
        string sId = ddl_productgroup.SelectedValue;
    }
    else
    {
        ddl_productgroup.Enabled = true;
    }

    ddl_seller.SelectedValue = Clsusr.UserId;
    //bool IsLeftBar = false;
    //IsLeftBar = Navigations.Generate_Color_sizes(UserId, tstocks);
    List<EteSellerProduct> freeser = new Products().GetSellerProducts(UserId,
2);

    TableRow tr = new TableRow();
    TableCell td = new TableCell();
    CheckBox chk = new CheckBox();
    FreeService.Rows.Clear();
    TableHeaderRow thr = new TableHeaderRow();
    thr.CssClass = "mdb-color darken-3";
    td.Text = "Select";
    thr.Cells.Add(td);
    td = new TableCell();
    td.Text = "Service";
    thr.Cells.Add(td);
    FreeService.Rows.Add(thr);

    foreach (var dt in freeser)
    {
        if (dt.ProductId != ddl_productgroup.SelectedIndex)
        {
            tr = new TableRow();
            td = new TableCell();
            chk = new CheckBox();
            chk.ID = "CheckBox" + dt.ProductId;

```

```

        chk.Checked = false;
        td.Controls.Add(chk);
        tr.Cells.Add(td);
        td = new TableCell();
        td.Text = "<span class='badge badge-primary'>" + dt.ProductDesc+
"/span>";
        tr.Cells.Add(td);
        FreeService.Rows.Add(tr);
    } //if data

} //for dt

} //if add
else
if (PageType == "Edit")
{
    ObjPD.BindDropDowns(ddl_productgroup, (new PopDIIs()).GetProduct());

    psid =Convert.ToDecimal( Request.QueryString["psid"]);
    EteproductSeller Prd = new Products().GetProductSeller(psid,
"PostProduct");

    if (Prd != null)
    {
        txt_psid.Text = Prd.PSId.ToString();

        Details.Text = Prd.SellerDescription;
        Description.Text = Prd.ProductDescription;
        ddl_productgroup.SelectedValue = Prd.ProductId.ToString();
        ddl_productgroup.Enabled = false;
        ddl_seller.SelectedValue = Prd.UserId.ToString();
        ddl_seller.Enabled = false;
        ddl_packing.SelectedIndex = (int)Prd.PricePackId;
        txt_price.Text = Prd.Price.ToString();
        int CurrencyId = (int)Prd.CurrencyId;
        txt_taskdetails.Text = Prd.KeyWords;
        txt_minorder.Text = Prd.MinOrder.ToString();
        txt_LeadTime.Text = Prd.DelTlme.ToString();
        int sno = 0;
        List<EteproductImages> imgs =new
Products().GetProductSellerImage(psid, "");
        foreach (var img1 in imgs)
        {
            sno++;
            if (sno == 1)
            {
                Image1.ImageUrl = img1.ImagePath;
                Image1.Visible = true;
            }
            if (sno == 3)
            {
                Image3.ImageUrl = img1.ImagePath;
                Image3.Visible = true;
            }
        }
    }
}

```

```

        if (sno == 2)
        {
            Image2.ImageUrl = img1.ImagePath;
            Image2.Visible = true;
        }
        if (sno == 4)
        {
            Image4.ImageUrl = img1.ImagePath;
            Image4.Visible = true;
        }
    } //for images

} //if not null daata

List<Eteproduct> freeser = new Products().GetSellerServicesbypsid(psid);
TableRow tr = new TableRow();
TableCell td = new TableCell();
CheckBox chk = new CheckBox();
FreeService.Rows.Clear();
TableHeaderRow thr = new TableHeaderRow();
thr.CssClass = "mdb-color darken-3";

td.Text = "Select";
thr.Cells.Add(td);
td = new TableCell();
td.Text = "Service";
thr.Cells.Add(td);
FreeService.Rows.Add(thr);
foreach (var dt in freeser)
{
    //if (dt.ProductId != ddl_productgroup.SelectedIndex)
    {
        tr = new TableRow();
        td = new TableCell();
        chk = new CheckBox();
        chk.ID = "CheckBox" + dt.ProductId;
        chk.Checked = true;
        td.Controls.Add(chk);
        tr.Cells.Add(td);
        td = new TableCell();
        td.Text = "<span class='badge badge-primary'>" + dt.ProductName +
"</span>";
        tr.Cells.Add(td);
        FreeService.Rows.Add(tr);
    }
} //foreach freeservices
if (freeser.Count() == 0)
{
    List<EteSellerProduct> freeser1 = new
Products().GetSellerProducts(UserId, 2);
    foreach (var dt in freeser1)
    {
        //if (dt.ProductId != ddl_productgroup.SelectedIndex)
        {

```

```

        tr = new TableRow();
        td = new TableCell();
        chk = new CheckBox();
        chk.ID = "CheckBox" + dt.ProductId;
        chk.Checked = false;
        td.Controls.Add(chk);
        tr.Cells.Add(td);
        td = new TableCell();
        td.Text = "<span class='badge badge-primary'>" + dt.ProductDesc +
"/span>";
        tr.Cells.Add(td);
        FreeService.Rows.Add(tr);
    }
} //foreach freeservices

}

} //if edit

}

}

//catch (Exception ex)
//{
//    Common.ShowMessageHtml(ex.Message, 1, true, dvMsgBox);
//    ExceptionLogging.SendExcepToDB(ex);

//    //objlog.Trace_Error(5, "DateOrWed", "PostProduts", "pageLoad", ex.Message);

//}
}
protected void btn_save_Click(object sender, EventArgs e)
{
    decimal PSId = 0;
    if (txt_psid.ToString() == "0") {
        PageType = "Add";
        PSId = new Products().GetNextPSId();
        txt_psid.Text = PSId.ToString();
    }
    //else { PageType = "Edit"; }
    PSId = Convert.ToDecimal(txt_psid.Text);

    CIsLoggedInUser CIsusr = (CIsLoggedInUser)Session["LoggedInUser_Specs"];
    try
    {
        if (Details.Text == "")
        {
            Common.ShowMessageHtml("Product Description cannot be blank", 0, true,
dvMsgBox);

```



```

}
else if (txt_taskdetails.Text == "")
{
    Common.ShowMessageHtml("Keywords cannot be blank", 0, true, dvMsgBox);

}
else
{
    bool Result = Insertupdate();

    if (Result == true)
    {
        if (PageType == "Add")
        {
            bool ret = false;
            CheckBox cbox = new CheckBox();
            cbox = (CheckBox)FreeService.FindControl("CheckBoxAll");
            List<EteSellerProduct> freeser = new
Products().GetSellerProducts(Convert.ToInt16(Session["VendorCode"]), 2);
            foreach (var list in freeser)
            {
                cbox = (CheckBox)FreeService.FindControl("CheckBox" + list.ProductId);
                if (cbox.Checked)
                {
                    EteproductSellerServices data = new EteproductSellerServices();
                    data.PSId = PSId;
                    data.productid = list.ProductId;
                    data.Remark="";
                    data.PrdPrice=0;
                    data.PrdUnit=0;
                    data.active = true;
                    data.userid = Convert.ToInt16(Clsusr.UserId);
                    ret = new Products().InsertproductSellerServices(data);
                }
            }
        }

        Common.ShowMessageHtml("Adding of Service Sucessful ", 1, true,
dvMsgBox);
    }
    else
    {
        Common.ShowMessageHtml("Service Details Edited Sucessfully ", 1, true,
dvMsgBox);
    }
    Response.Redirect("~/VendorHome.aspx?pageid=1&msg=Product Updated
Sucessfully");
}

```

```

    }
}
}

catch (Exception exc)
{
    //objlog.Trace_Error(4, "DateOrWed", "MangeDealer", "Load", "Exception:" +
exc.ToString());
    Common.ShowMessageHtml(exc.ToString(), 0, true, dvMsgBox);
}
}

private void ManageNavigations(int pageid, int userid, int groupid, string PageType)
{
}

```

```

public byte[] GetPhtoStream()
{
    byte[] bufferPhoto = new byte[FileUpload1.PostedFile.ContentLength];
    Stream photoStream = FileUpload1.PostedFile.InputStream;
    photoStream.Read(bufferPhoto, 0, FileUpload1.PostedFile.ContentLength);

    return bufferPhoto;
}

protected void btnupload_Click(object sender, EventArgs e)
{
    string path = FileUpload1.PostedFile.FileName;// FileUpload1.FileName;
    if (path == "") { return; }
    byte[] Photo = GetPhtoStream();
    Stream strm = FileUpload1.PostedFile.InputStream;

    Id = Convert.ToInt32(Request.QueryString["id"]);
    string base64String = Convert.ToBase64String(Photo, 0, Photo.Length);
    string imageUrl = "data:image/png;base64," + base64String;
    if (txt_psid.Text.ToString() == "0")
    {
        Insertupdate();
    }
    EteproductImages ep = new EteproductImages();
    ep.PSId = Convert.ToDecimal( txt_psid.Text);
    ep.ProductId = ddl_productgroup.SelectedIndex; ;
    if (ep.ProductId == 0) { return; }
    ep.SellerId = ddl_seller.SelectedIndex;
    ep.srno = 1;
    ep.Images = Photo;
    ep.ImageUrls = imageUrl;
}

```

```

ep.ImagePath = path;

string Photold = new Products().InsertPhoto(ep);
if (Photold != "")
{
    string server = Server.MapPath(Photold);
    FileUpload1.SaveAs(server);
    string targetPath = Server.MapPath("thumb/"+Photold);

    ReducelImageSize(0.2, strm, targetPath);

}
string Pid = Session["Pid"].ToString();
if(Pid=="1")
{
    Image1.ImageUrl = ImageUrl;
    Image1.Visible = true;
    Session["Pid"] = "2";

}
if (Pid == "2")
{
    Image2.ImageUrl = ImageUrl;
    Image2.Visible = true;
    Session["Pid"] = "3";

}
if (Pid == "3")
{
    Image3.ImageUrl = ImageUrl;
    Image3.Visible = true;
    Session["Pid"] = "4";

}
if (Pid == "4")
{
    Image4.ImageUrl = ImageUrl;
    Image4.Visible = true;
    Session["Pid"] = "5";

}
}

private bool Insertupdate()
{
    decimal ProductId = Convert.ToDecimal(ddl_productgroup.SelectedValue);
    if (ProductId == 0)
    {
        Common.ShowMessageHtml("Please select a Service to continue", 0, true,
dvMsgBox);

        return false; }
    PageType = Session["PageType"].ToString();
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];

```

```

decimal PSId = 0;
EteproductSeller Prd = new EteproductSeller();
if (PageType=="Add")
{
    PSId = new Products().GetNextPSId();
    txt_psid.Text = PSId.ToString();
    Prd = new EteproductSeller();
}
else
{
    PSId = Convert.ToDecimal(txt_psid.Text.ToString());

    Prd = new Products().GetProductSeller((int)PSId, "PostProduct");
}
PSId =Convert.ToDecimal( txt_psid.Text.ToString());

//Eteproduct Prd = new Eteproduct();
Prd.PSId = PSId;
Prd.ProductId =Convert.ToDecimal( ddl_productgroup.SelectedValue);
Prd.CreatedOn = DateTime.Now;
Prd.UserId = Convert.ToInt32(Clsusr.UserId);
Prd.SellerDescription = Details.Text.ToString();
Prd.ProductDescription = Description.Text;
Prd.SellerId =Convert.ToInt16( ddl_seller.SelectedValue);
if (txt_price.Text == "")
{
    Prd.Price = 0;
}
else
{
    Prd.Price = Convert.ToDecimal(txt_price.Text);
}
Prd.PriceUnits = 1;
Prd.PriceUomId = 1;
Prd.PricePackId = 1;

Prd.CurrencyId = ddl_currency.SelectedIndex;
if (Prd.CurrencyId <1) { Prd.CurrencyId = 1; }
Prd.PricePackId = ddl_packing.SelectedIndex;
if (Prd.PricePackId <1) { Prd.PricePackId = 1; }
Prd.MinOrder = Convert.ToInt16(txt_minorder.Text);
Prd.MinOrderUom = 1;
Prd.DelTlme = Convert.ToInt16(txt_LeadTime.Text);
Prd.DelTimeType = 1;
Prd.PrId = 1;
Prd.LinkPSId = PSId;
Prd.MadeinCountry = 1;

Prd.AnalysisCode = txt_taskdetails.Text.ToString();
//-----
Prd.DiscountLink = 1;
Prd.BrandName = "";
Prd.MaterialCare = "";
Prd.MaterialType = "";

```

```

Prd.Manufacturer = "";
Prd.Length = "";
Prd.Location = Location.Text;
Prd.GeoLocation = " ";
bool Result = false;

if (PageType == "Add")
{
    Result = new Products().InsertProductSeller(Prd,false);
    Session["PageType"] = "Edit";
}
else
{
    Prd.ProductId = Convert.ToInt32(Request.QueryString["ProductId"]);

    //Result = new Products().InsertProductSeller(Prd, true);
    Session["PageType"] = "Edit";
}

return Result;
}

private void ReduceImageSize(double scaleFactor, Stream sourcePath, string targetPath)
{
    using (var image = System.Drawing.Image.FromStream(sourcePath))
    {
        var newWidth = (int)(image.Width * scaleFactor);
        var newHeight = (int)(image.Height * scaleFactor);
        var thumbnailImg = new Bitmap(newWidth, newHeight);
        var thumbGraph = Graphics.FromImage(thumbnailImg);
        thumbGraph.CompositingQuality =
System.Drawing.Drawing2D.CompositingQuality.HighQuality;
        thumbGraph.SmoothingMode =
System.Drawing.Drawing2D.SmoothingMode.HighQuality;
        thumbGraph.InterpolationMode =
System.Drawing.Drawing2D.InterpolationMode.HighQualityBicubic;
        var imageRectangle = new Rectangle(0, 0, newWidth, newHeight);
        thumbGraph.DrawImage(image, imageRectangle);
        thumbnailImg.Save(targetPath, image.RawFormat);
    }
}

}

*****ProductSearchList.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="ProductSearchList.aspx.cs"
Inherits="ProductSearchList" %>

<asp:Content ID="Content5" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
<<!--link rel="stylesheet" href="css/swiper.css"/>
<link rel="stylesheet" href="css/swiper.min.css"/>

```

```
<link rel="stylesheet" href="css/bootstrap.css"/>
```

```
<link rel="stylesheet" href="css/base.css"/>  
<link rel="stylesheet" href="css/migrate.css"/>  
<script src="js/swiper.js"></script>  
<script src="js/swiper.min.js"></script>  
<script src="swiper.esm.js"></script>  
<script src="js/swiper.esm.bundle.js"></script>
```

```
<link rel="stylesheet" href="css/vendors.css"/>
```

```
<script src="js/common.js"></script>  
<script src="js/jquery.countdown.js"></script>  
<meta charset="utf-8"/>  
<meta name="viewport" content="width=device-width, initial-scale=1"/>  
<script src="js/jquery1.min.js"></script>  
<script src="js/bootstrap1.min.js"></script>--%>
```

```
<script type="text/javascript">  
function ImgClicked(Id) {  
    alert('ImgClicked-1');  
    //var fullURL = (location.pathname + location.search).substr(1);  
<%--    var loc = document.getElementById("<%=ddl_location.ClientID%>").value;  
  
    var EnCodedFullURL = encodeURIComponent(fullURL);  
    var SearchString = document.getElementById("<%=txt_SearchString.ClientID%>").value; --  
    %>
```

```
    var url = 'ViewSearchResult.aspx?pageid=198' + '&SrchStrng=' +  
    encodeURIComponent(Id.value) ;  
    //' + '&loc=' + encodeURIComponent(loc);  
    //$(location).attr('href', url);  
    window.location.href = url;  
    event.preventDefault();  
    }  
    function submitButton(event) {  
        //alert(event.which);  
        if (event.which == 13) {  
            alert('Enter');  
            //$('#Button1').trigger('click');  
            GetURLSrng();  
        }  
    }  
}
```

```
function TextName_OnKeyDown(e) {  
    //alert('TextName_OnKeyDown');  
    var keynum  
    if (window.event) // IE  
    {  
        keynum = e.keyCode  
    }  
    else if (e.which) // Netscape/Firefox/Opera  
    {  
        keynum = e.which
```

```

}

if (keynum == 13) {
//alert('Enter');
GetURLString()
}
}
function GetLoginString() {
// alert('GetLoginString-1');
var url = 'Login.aspx?';
//$(location).attr('href', url);
window.location.href = url;
event.preventDefault();
}
function GetRegisterString() {
// alert('GetLoginString-1');
var url = 'Register.aspx?';
//$(location).attr('href', url);
window.location.href = url;
event.preventDefault();
}
}
</script>

<style type="text/css">
.padding {
padding: 0px;
}

.padding th {
padding: 0px !important;
}

.span3 td {
padding: 0px;
}

.span5 td {
padding: 0px;
}

.span5 th {
padding: 0px;
}
</style>
</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
<div class="container-fluid">
<div class="row-fluid" style="padding-top: 20px;">
<div class="span2" id="dvlogo" runat="server" style="display: block;">
<%-- --%>
</div>
<div class="span10">
<div class="row">

```

```
<%--<h3>Ravikiran Techno Soft</h3>--%>
<div id="dvMsgBox" class="span12 ordersave" runat="server" style="display:
block;">
    <asp:Label ID="Label1" runat="server" Text="Message"></asp:Label>
</div>
</div>
</div>
</div>
</div>
</asp:Content>
<asp:Content ID="Content3" ContentPlaceHolderID="CPH_LeftSideBar" runat="Server">
    <div id="LeftSideBar" runat="server" class="span2">
        <asp:Table ID="Table1" runat="server" Width="250px">
            </asp:Table>
        </div>
    </asp:Content>

<asp:Content ID="Content4" ContentPlaceHolderID="CPH_MainContent" runat="Server">
<div class="span10" id="MainContent" runat="server">
<div id="ShowPageHeader" class="row" style="display: block;" runat="server">
<div id="PageName" class="span10 customer" runat="server">
</div>
</div>

<div id="Div1" class="row" style="display: block;">
<div id="Div2" class="span10 "> <%--customertable--%>
<div class="wrapper">
<div class="directory-structure clearfix" >
<div id="appvendors" class="span10 directory-structure-content" runat="server"> <%--
directory-structure-content--%>
<%--data update--%>

<%--data update--%>

</div>
</div>

</div>
```



```
</div>
</div>
</div>
```

```
</asp:Content>
```

```
***** ProductSearchList.aspx.cs*****
```

```
using DowBAL;
using DowBEntity;
using DowDAL;
using System;
using System.Collections.Generic;
using System.Data;
using System.Linq;
using System.Text;
using System.Text.RegularExpressions;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.Security;
using System.Data.SqlClient;
using System.Configuration;
//using LogTracer;
```

```
public partial class ProductSearchList : System.Web.UI.Page
{
    DataTable dtSource = null;
    List<Eteproduct> LstChecked = new List<Eteproduct>();
    DDLogTracer objlog = new DDLogTracer();
    protected void Page_Load(object sender, EventArgs e)
    {
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr != null)
        {
            string pageid = "50";// Request.QueryString["pageid"];

            // string pageid = Request.QueryString["pageid"];
            string searchtype = Request.QueryString["searchtype"];
            string bytype = Request.QueryString["bytype"];
            string id = Request.QueryString["Id"];
            ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
            Convert.ToInt32(Clsusr.GroupId));
            string msg = Request.QueryString["msg"];
            if (msg != null)
            {
                Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
            }
            //CreateTreeNode(Convert.ToInt16( id));
            //Createtree(id);
            Bindetails();
        }
    }
}
```

```

    }
    else
    {
        Response.Redirect("MLB_Login.aspx");
    }
}

private void Bindetails()
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    int pid = Convert.ToInt16(Request.QueryString["id"]);
    int GroupId = Convert.ToInt16(Clsusr.GroupId);
    string PreHtml = "";
    StringBuilder sb = new StringBuilder();
    sb.Append(" <div class='directory-list'>");
    bool slider = true;

    //try
    {
        IList<ProductProductSeller> Lst = (new
        Products()).GetProductSellerbyProductId(pid, "SearchList");
        if (Lst != null && Lst.Count > 0)
        {
            foreach (var dt in Lst)
            {
                string href = "", title;
                if (GroupId == 5)
                {
                    title = "More info " ;
                    href = "ManageProductsSource.aspx?PSId=" + dt.PSId;
                }
                else
                {
                    href = "PostProducts.aspx?psid=" + dt.PSId+"&PageType='Edit'";
                    title = "Post-Edit " ;
                }
            }

            //code
            //if (dt.lbtnimage == null)
            //{
            //    continue;
            //}
            if (slider)
            {
                sb.Append(" <div class='app-ec-list app-internal-tracking-page' data-it-
                page='1'>\n\r");
                sb.Append(" <div id='esl_81030" + dt.ProductId + dt.PSId.ToString() + "
                class='pure-g-r directory-list-item app-internal-tracking-item app-vendors-item app-vpromo
                appGallery_35'");
                sb.Append("      data-ec-brand='Free'>\n ");
            }
        }
    }
}

```

```

thumb
sb.Append("<div class='pure-u-1-3 vendor-slider-thumb'>\n"); // vendor-slider-
thumb
sb.Append("<div class='vendor-slider'>\n");
sb.Append(" <div class='app-vendors-item-slider'></div>\n");
sb.Append(" <div class='vendor-slider-content'>\n");
sb.Append(" <div class='listing-caption'>\n");
sb.Append(" <span class='listing-caption-count'>\n");
//sb.Append(" <i class='icon-vendor icon-vendor-count-pic icon-
left'></i>Nos:1 </span>\n");
sb.Append(" </div>\n");
sb.Append(" <figure class='homePremiumBox__figure img-zoom'>");
sb.Append("<a href='"+href+"'>");
sb.Append(" <img class='homePremiumBox__img' alt='Pic-
"+dt.ProductName+" src='"+dt.ImagePath+"' />");
sb.Append("</a>");
sb.Append(" </figure>");

// sb.Append(" <img class='app-link' src='"+ dt.ImagePath + "' alt ='Pic-" +
dt.ProductName + "' onclick='JavaScript:ImgClicked("'" + dt.PSId + "'') />\n"); //height='300px'
width='250px'
sb.Append("</div>\n");
sb.Append("</div>\n");

sb.Append("</div>\n");

sb.Append("<div class='pure-u-2-3'>\n");
sb.Append(" <div class='directory-item-content'>\n");

sb.Append("<a id='app_InkEmp_" + dt.PSId.ToString() + "' \n");
sb.Append(" class='item-title app_InkEsc_9880 app-ec-track-vendor-
click'\n");
sb.Append(" href='"+ href + "'>" + dt.SellerDescription + "</a>\n");

sb.Append("<div class='btn-fav app-vendors-item-guardar no-hover'");
sb.Append("data-id=" + dt.PSId + dt.ProductId.ToString() + "'");
sb.Append("data-id-sector='122'>");
sb.Append("</div>\n");

sb.Append("<div class='item-subtitle'> <span>" + dt.ProductName + " <span>
</div>\n");

sb.Append("<p class='item-desc'>");
sb.Append(" <span class='app-common-ellipsis'>" + dt.ProductName +
"...</span>");
sb.Append(" <span>(" + dt.SellerName + ")</span> </p>");

sb.Append("<div class='listing-item-faq-container'>\n");

sb.Append(" <div class='listing-item-faq'> ");
sb.Append(" <i class='icon-vendor icon-vendor-faq-price'></i>\n");
sb.Append(" <p>");
sb.Append(" <span class='item-faqs-label notablet'>");
sb.Append(" Price </span>");

```

```

        sb.Append("    <span class='item-faqs-value'>₹" + dt.Price.ToString() +
"</span>");
        sb.Append(" </p>\n");
        sb.Append(" <a title='" + title + "' class='btn btn-primary' rel='nofollow' href='"
+ href + ">" + title + "</a>");
        sb.Append("    </div>\n");
        sb.Append(" </div>\n");
        sb.Append(" </div>\n");
        sb.Append(" </div>\n");
        sb.Append(" </div>\n");
        sb.Append(" </div>\n");
        sb.Append(" </div>\n");
        sb.Append(" </div>\n");
        sb.Append(" <div class='clear'>        <p> </p>        </div>");

        //foreach code ends
    }
}

if (slider)
{
    //sb.Append("<div class='app-vendors-paginado nav pagination mt20 mb20'>");

    //sb.Append("<script>");
    //sb.Append("vendors_fncEmpLNL();");
    //sb.Append("</script>");
    sb.Append("</div>\n");

    PreHtml = sb.ToString();
    if (PreHtml.Length > 0)
    {
        // objlog.Trace_Log(4, "Gordersearchlist", "1", "SliderList", PreHtml);
        appvendors.InnerHtml = PreHtml;
    }
}
else
{
    Common.ShowMessageHtml("Sorry no Matching Products found", 1, true,
dvMsgBox);
}

}
}
//catch (Exception ex)
//{{
// Common.ShowMessageHtml(ex.Message, 1, true, dvMsgBox);

//}
}

```

protected void LinkButton2_Click(object sender, EventArgs e)

```

{
    LinkButton Inkbutton = (LinkButton)sender; ;
    string OrdId = Inkbutton.CommandName;

    CIsLoggedInUser CIsusr = (CIsLoggedInUser)Session["LoggedInUser_Specs"];

    //BindetailsGroup(OrdId);
    //Bindetailsselected(Convert.ToInt16(CIsusr.UserId));
}
protected void LinkButton1_Click(object sender, EventArgs e)
{
    LinkButton Inkbutton = (LinkButton)sender; ;
    decimal ProductId = Convert.ToDecimal(Inkbutton.CommandName);
    CIsLoggedInUser CIsusr = (CIsLoggedInUser)Session["LoggedInUser_Specs"];
    int SellerId = Convert.ToInt16(CIsusr.UserId);
    bool ret = new Products().UpdateSellerProduct(ProductId, SellerId);

    //Bindetailsselected(SellerId);
}
private void ManageNavigations(int pageid, int userid, int groupid)
{
    bool IsLeftBar = false;
    IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
myBtnClickHandler);

    if (IsLeftBar == true)
    {
        LeftSideBar.Attributes["class"] = "span2";
        MainContent.Attributes["class"] = "span10";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;"); //style="display:
block;"
        PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
    else
    {
        LeftSideBar.Attributes["class"] = "";
        MainContent.Attributes["class"] = "span12";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
        PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
}
protected void Check_SelectedIndexChanged(object sender, EventArgs e)
{
    int selectedGroupid = Convert.ToInt32(0);
}

```

```

// This is Event Handler
protected void myBtnClickHandler(object sender, EventArgs e)
{

    //LinkButton lnk = (LinkButton)sender;
    //string ID = lnk.ID;

    LinkButton lnk = sender as LinkButton;

    if (lnk != null)
    {
        string ID = lnk.ID;
        Common.ShowMessageHtml("Please wait while refreshing the records", 1, true,
dvMsgBox);
        string pid = (Convert.ToInt16(Request.QueryString["pid"]) - 8).ToString();
        DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
        if (ds != null)
        {
            if (ds.Tables[0].Rows.Count > 0)
            {
                DataTable dtDll = ds.Tables[0];
                foreach (DataRow drnew in dtDll.Rows)
                {
                    switch (drnew["lbtncode"].ToString())
                    {
                        case "SAVE":
                            ContentSaveControlClicked();
                            // Call SAVE Method
                            break;
                        case "URL":
                            Response.Redirect(drnew["pageurl"].ToString() + "&pid=" +
drnew["lbtnid"]);
                            break;
                        case "AUDIT":
                            Response.Redirect(drnew["pageurl"].ToString() + "&frompage=" +
Request.QueryString["pageid"]);
                            break;

                        case "CANCEL":
                            // Call Cancel Method
                            break;
                        case "ADDSUB":

                            Response.Redirect("~/ManageSubgroup.aspx?pageid=5&PageType=Add&ProductId=" +
pid);
                            break;
                        case "ADDPRODUCT":

                            Response.Redirect("~/ManageProducts.aspx?pageid=5&PageType=Add&ProductId=" +
pid);
                            break;
                        case "MYBUYOFFER":
                            Response.Redirect("~/MyProductsList.aspx?pageid=5&pagetype=Buy");
                            break;
                        case "MYSALEOFFER":

```

```

Response.Redirect("~/MyProductsList.aspx?pageid=5&pagetype=Sale");
        break;
    default:
        break;
    }
}
}
}
}
}
DropDownList dropDownList = sender as DropDownList;
if (dropDownList != null)
{
    Common.ShowMessageHtml("", 0, false, dvMsgBox);
    Bindetails_Filter(Table1);
}
}
private void ContentSaveControlClicked()
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    int SellerId = Convert.ToInt16(Clsusr.UserId);
    foreach (var data in LstChecked)
    {
        if (data.Check == true)
        {
            decimal id = data.ProductId;
            bool ret = new Products().UpdateSellerProduct(id, SellerId);
        }
    }
}

}

private void Bindetails_Filter(Table table)
{
    try
    {
        int brandID = 0;
        int dealerID = 0;
        int orderStatusID = 0;
        int customerType = 0;
        int dateRange = 0;
        int OrderHold = 0;

        foreach (TableRow row in table.Rows)
        {
            foreach (TableCell rowcell in row.Cells)
            {
                foreach (WebControl contrl in rowcell.Controls)

```

```

        {
            DropDownList listDrop = contrl as DropDownList;
            if (listDrop != null)
            {
                if (listDrop.Attributes["LISTTYPE"] != null)
                {
                    string rowcellHead = listDrop.Attributes["LISTTYPE"].ToUpper();
//rowcell.Text.ToUpper());
                    ExtractAll(ref orderStatusID, ref OrderHold, ref brandID, ref
customerType, ref dateRange, ref dealerID, rowcellHead, listDrop);
                }
            }
            else
            {
                continue;
            }
        }
    }
}

FilterEnum filterLst = new FilterEnum();
filterLst.OrderStatus = orderStatusID;
filterLst.OrderHold = OrderHold;
filterLst.Brand = brandID;
filterLst.CustomerType = customerType;
filterLst.DateRange = dateRange;
filterLst.Dealer = dealerID;
//RefreshData(filterLst);
}
catch (Exception ex)
{
    Common.ShowMessageHtml(ex.Message, 1, true, dvMsgBox);
}
}
}

```

```

private static void ExtractAll(ref int orderStatusID, ref int OrderHold, ref int brandID, ref int
customerType, ref int dateRange, ref int dealerID, string rowcellHead, DropDownList
listDrop)

```

```

{
    //string strddl=rowcellHead;
    switch (rowcellHead)
    {
        case "BRAND":
            if (string.Compare(listDrop.SelectedItem.Text, "All", true) == 0)
            {
                brandID = 0;
            }
            else
            {
                brandID = Convert.ToInt32(listDrop.SelectedValue);
            }
        }
    }
}

```



```

    }
    break;
case "CUSTOMER TYPE":
    string strCustType = listDrop.SelectedItem.Text;
    if (string.Compare(strCustType, "Residential", true) == 0)
    {
        customerType = 1;
    }
    else if (string.Compare(strCustType, "Business", true) == 0)
    {
        customerType = 2;
    }
    else
    {
        customerType = 0;
    }
    break;
case "DATE RANGE":
    dateRange = SelectDateRange(dateRange, listDrop);
    break;
case "DEALER":
    if (string.Compare(listDrop.SelectedItem.Text, "All", true) == 0)
    {
        dealerID = 0;
    }
    else
    {
        dealerID = Convert.ToInt32(listDrop.SelectedValue);
    }

    break;
case "STATUS":
    if (string.Compare(listDrop.SelectedItem.Text, "All", true) == 0)
    {
        orderStatusID = 0;
    }
    else
    {
        orderStatusID = Convert.ToInt32(listDrop.SelectedValue);
    }

    break;
case "ON HOLD":
    string strOnHold = listDrop.SelectedItem.Text;
    if (string.Compare(strOnHold, "Yes", true) == 0)
    {
        OrderHold = 1;
    }
    else if (string.Compare(strOnHold, "No", true) == 0)
    {
        OrderHold = 2;
    }
    else
    {
        OrderHold = 0;
    }

```

```

        }
        break;
    default:
        break;
    }
}

private static void SetStatus(Table tbl, string LoggedinID, string FilterName)
{
    try
    {
        foreach (TableRow row in tbl.Rows)
        {
            foreach (TableCell rowcell in row.Cells)
            {
                foreach (WebControl contrl in rowcell.Controls)
                {
                    DropDownList listDrop = contrl as DropDownList;
                    if (listDrop != null)
                    {
                        if (listDrop.Attributes["LISTTYPE"] != null &&
listDrop.Attributes["LISTTYPE"].ToUpper() == FilterName)
                        {
                            listDrop.SelectedValue = LoggedinID;
                        }
                    }
                    else
                    {
                        continue;
                    }
                }
            }
        }
    }
    catch (Exception ex)
    {
        //Common.ShowMessageHtml(ex.Message, 1, true, dvMsgBox);
    }
}

private static int SelectDateRange(int dateRange, DropDownList listDrop)
{
    string strDateRange = listDrop.SelectedItem.Text.ToUpper();
    switch (strDateRange)
    {
        case "ALL":
            {
                dateRange = 0;
                break;
            }
    }
}

```

```
case "TODAY":
  {
    dateRange = 1;
    break;
  }

case "YESTERDAY":
  {
    dateRange = 2;
    break;
  }
case "THIS WEEK":
  {
    dateRange = 3;
    break;
  }
case "LAST WEEK":
  {
    dateRange = 4;
    break;
  }

case "LAST 7 DAYS":
  {
    dateRange = 5;
    break;
  }
case "LAST 30 DAYS":
  {
    dateRange = 6;
    break;
  }
case "THIS MONTH":
  {
    dateRange = 7;
    break;
  }
case "LAST MONTH":
  {
    dateRange = 8;
    break;
  }
case "THIS YEAR":
  {
    dateRange = 9;
    break;
  }
case "LAST YEAR":
  {
    dateRange = 10;
    break;
  }
default:
  dateRange = 0;
  break;
```

```

    }
    return dateRange;
}

protected void Lnkbtn_ManageOrder_Click(object sender, EventArgs e)
{
    LinkButton lnkbutton = (LinkButton)sender;
    string strcustid = lnkbutton.CommandName;

    if (strcustid.Length > 0)
    {
        string[] stringSeparators = new string[] { "&", "=" };
        string[] arr1;
        arr1 = strcustid.Split(stringSeparators, StringSplitOptions.RemoveEmptyEntries);
        bool bchkstatus = false;
        foreach (var word in arr1)
        {
            if (word == "status")
            {
                {
                    bchkstatus = true;
                }
            }

            if (bchkstatus == true && word != "status")
            {
                Response.Redirect(Navigations.StatusWiseRedirect(Convert.ToInt32(word),
strcustid));
            }
        }
    }
}

}

}

}

}

}

}

*****ProductSelectList.aspx*****
<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="ProductSelectList.aspx.cs" Inherits="ProductSelectList"
%>

<%@ Register Assembly="Infragistics4.Web.v14.2, Version=14.2.20142.1028,
Culture=neutral, PublicKeyToken=7dd5c3163f2cd0cb"
Namespace="Infragistics.Web.UI.GridControls" TagPrefix="ig" %>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
<div class="container-fluid">

```

```

<div class="row-fluid" style="padding-top: 20px;">
  <div class="span2" id="dvlogo" runat="server" style="display: block;">
    <%----%>
  </div>
  <div class="span10">
    <div class="row">
      <div id="dvMsgBox" class="span12 ordersave" runat="server" style="display:
none;">
        <asp:Label ID="Label1" runat="server" Text="Label"></asp:Label>
      </div>
    </div>
  </div>
</div>
</div>
</div>
</div>
</div>
</div>
</asp:Content>

<asp:Content ID="Content3" ContentPlaceHolderID="CPH_LeftSideBar" runat="Server">
  <div id="LeftSideBar" runat="server" class="span2">
    <asp:Table ID="Table1" runat="server">
    </asp:Table>
  </div>
</asp:Content>

<asp:Content ID="Content4" ContentPlaceHolderID="CPH_MainContent" runat="Server">
  <div class="span10" id="MainContent" runat="server">
    <div id="ShowPageHeader" class="row" style="display: block;" runat="server">
      <div id="PageName" class="span12 customer" runat="server">
      </div>
    </div>
  </div>

  <div id="Div1" class="row" style="display: block;">
    <div id="Div2" class="span12 customertable">
    <%--data upate --%>

    <%-- data upate--%>
    <table id="Tablex" class="table table-detail span12" runat="server" visible="false">

    <tbody>
    <tr>
    <td>

      <ig:WebDataGrid ID="WDGrid_UserList" runat="server" CssClass="wdglist"
AutoGenerateColumns="False" Width="360PX" ShowHeader="false" BorderColor ="White">
        <Columns>

          <ig:BoundDataField DataFieldName="pgroupname" Key="pgroupname"
Width="260px" >
            <Header Text="Product Group">
            </Header>

          </ig:BoundDataField>

```

```

                <ig:TemplateDataField Key="pgid" Width="100px" >
                    <ItemTemplate>
<%--                <asp:LinkButton ID="LinkButton2" runat="server"
OnClick="LinkButton2_Click" CommandName='<% #Eval("pgid", "{0}")%'
>>>></asp:LinkButton>--%>
                <asp:HyperLink ID="ProductId" runat="server" Text=">>>>"
Width="50px"
                NavigateUrl='<% #Eval("pgid",
"-~/ProductSearchList.aspx?pageid=26&PageType=Retry&PgId={0}")%'>
                </asp:HyperLink>

                    </ItemTemplate>
                <Header Text="Show List" />
            </ig:TemplateDataField>

        </Columns>
        <Behaviors>
            <ig:Paging CurrentPageLinkCssClass="Current"
PageLinkCssClass="Link" PagerCssClass="Pager" PagerMode="NumericFirstLast"
PageSize="38" QuickPages="3" FirstPageText="First" LastPageText="Last"
NextPageText="Next" PreviousPageText="Previous" PagerAppearance="bottom">
            </ig:Paging>

            <ig:Filtering>
            </ig:Filtering>
            <ig:Sorting Enabled="true">
            </ig:Sorting>

        </Behaviors>
    </ig:WebDataGrid>
    <asp:SqlDataSource ID="SqlDataSource2" runat="server"
ConnectionString="<%= $ ConnectionStrings:DownConectingString %>"
SelectCommand="SELECT * FROM [ProductCat]"></asp:SqlDataSource>

</td>

<td>
        <ig:WebDataGrid ID="WebGroupList" runat="server" CssClass="wdglist"
AutoGenerateColumns="False" Width="360PX" ShowHeader="false" BorderColor="White">
            <Columns>
                <ig:BoundDataField DataFieldName="ProductId" Key="ProductId"
Width="1px">
                    <Header Text="Id">
                    </Header>
                </ig:BoundDataField>
                <ig:BoundDataField DataFieldName="ProductName"
Key="ProductName" Width="260px">
                    <Header Text="Desc">
                    </Header>
                </ig:BoundDataField>
<%--                <ig:TemplateDataField Key="Check" Width="100px">
                <ItemTemplate>
                    <asp:CheckBox ID="checkbox" runat="server" />

```

```

        </ItemTemplate>
        <Header Text="Select List" />
    </ig:TemplateDataField>--%>
    <ig:TemplateDataField Key="Check" Width="100px">
        <ItemTemplate>
            <asp:LinkButton ID="LinkButton1" runat="server" OnClick="LinkButton1_Click"
CommandName='<% #Eval("ProductId", "{0}")%>' >>>></asp:LinkButton>
        </ItemTemplate>
    </ig:TemplateDataField>

    </Columns>
    <Behaviors>

<%--
    <ig:Activation Enabled="true" />
    <ig:Selection RowSelectType="Multiple" Enabled="true" />
    <ig:RowSelectors RowNumbering="false" Enabled="true" />

--%>

    <ig:Paging CurrentPageLinkCssClass="Current"
PageLinkCssClass="Link" PagerCssClass="Pager" PagerMode="NumericFirstLast"
PageSize="15" QuickPages="3" FirstPageText="First" LastPageText="Last"
NextPageText="Next" PreviousPageText="Previous" PagerAppearance="bottom">
    </ig:Paging>

    <ig:Filtering Enabled="false">
    </ig:Filtering>
    <ig:Sorting Enabled="true">
    </ig:Sorting>
    </Behaviors>
    </ig:WebDataGrid>
    <asp:SqlDataSource ID="SqlDataSource1" runat="server"
ConnectionString="<%= $ConnectionStrings.DowConectingString %>"
SelectCommand="SELECT * FROM [Product]"></asp:SqlDataSource>

    <ig:WebDataGrid ID="WebDataGrid1" runat="server" CssClass="wdglist"
AutoGenerateColumns="False" ShowHeader="true" BorderColor="White">
    <Columns>

    <%--
        <ig:BoundDataField DataFieldName="ProductId" Key="ProductId"
Width="260px">
            <Header Text="Id">
            </Header>
        </ig:BoundDataField> --%>

        <ig:BoundDataField DataFieldName="ProductName"
Key="ProductName" Width="260px">
            <Header Text="Selected Products">
            </Header>
        </ig:BoundDataField>

    </Columns>
    <Behaviors>
    <ig:Paging CurrentPageLinkCssClass="Current"
PageLinkCssClass="Link" PagerCssClass="Pager" PagerMode="NumericFirstLast"

```

```
PageSize="15" QuickPages="3" FirstPageText="First" LastPageText="Last"
NextPageText="Next" PreviousPageText="Previous" PagerAppearance="bottom">
    </ig:Paging>
```

```
        <ig:Filtering>
        </ig:Filtering>
        <ig:Sorting Enabled="true">
        </ig:Sorting>
    </Behaviors>
</ig:WebDataGrid>
```

```
    </td>
</tr>
</tbody>
</table>
```

```
</div>
```

```
</div>
</div>
```

```
</asp:Content>
```

```
*****ProductSelectList.aspx.cs *****
```

```
using DowBAL;
using DowBEntity;
```

```
using System;
using System.Collections.Generic;
using System.Data;
using System.Diagnostics;
using System.IO;
using System.Linq;
using System.Net;
using System.Text.RegularExpressions;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using Infragistics.Web.UI.GridControls;
public partial class ProductSelectList: System.Web.UI.Page
```

```
{
    DDLogTracer objlog = new DDLogTracer();
    List<Eteproduct> LstChecked = new List<Eteproduct>();
    protected void Page_Load(object sender, EventArgs e)
    {
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr != null)
        {
            string pageid = Request.QueryString["pageid"];
            string searchtype = Request.QueryString["searchtype"];
            string bytype = Request.QueryString["bytype"];
        }
    }
}
```



```

        ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
Convert.ToInt32(Clsusr.GroupId));
        string msg = Request.QueryString["msg"];
        if (msg != null)
        {
            Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
        }

        Bindetails();

    }
    else
    {
        Response.Redirect("MLB_Login.aspx");
    }
    ///((BoundCheckBoxField)this.WebDataGrid1.Columns["ProductId"]).ValueConverter =
new DecimalBooleanConverter();

}
private void Bindetails()
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    int pid =Convert.ToInt16( Request.QueryString["pid"])-8;
    int Pgld = Convert.ToInt16(Request.QueryString["Pgld"]);
    if (pid < 1)
    {
        pid = new Products().GetProductCatIdfromGId(Pgld);
    }
    try
    {
        IList<EteProductGroup> Lst = (new
Products()).GetProductGroupsbyCatId("SearchList", pid);
        if (Lst != null && Lst.Count > 0)
        {
            if (WDGrid_UserList.HasDataSource == true) {
WDGrid_UserList.ClearDataSource(); }
            WDGrid_UserList.DataSource = Lst;
            WDGrid_UserList.DataBind();
            WDGrid_UserList.Visible = true;

        }
        else
        {
            Common.ShowMessageHtml("No records found", 0, true, dvMsgBox);

            WDGrid_UserList.Visible = false;
        }
        if(Pgld>0){
            BindetailsGroup(Pgld);
        }
        Bindetailselected(Convert.ToInt16( Clsusr.UserId));
    }
    catch
    {

```

```

    }
}
protected void LinkButton2_Click(object sender, EventArgs e)
{
    LinkButton lnkbutton = (LinkButton)sender; ;
    string OrdId = lnkbutton.CommandName;

    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];

    //BinddetailsGroup(OrdId);
    Binddetailsselected(Convert.ToInt16(Clsusr.UserId));
}
protected void LinkButton1_Click(object sender, EventArgs e)
{
    LinkButton lnkbutton = (LinkButton)sender; ;
    decimal ProductId =Convert.ToDecimal( lnkbutton.CommandName);
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    int SellerId = Convert.ToInt16(Clsusr.UserId);
    bool ret = new Products().UpdateSellerProduct(ProductId, SellerId);

    Binddetailsselected(SellerId);
}
private void BinddetailsGroup(int OrdId)
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];

    try
    {
        LstChecked = (new Products()).GetProductbyGroup(OrdId, "ProductSearchList");
        if (LstChecked != null && LstChecked.Count > 0)
        {
            if (WebGroupList.HasDataSource == true) { WebGroupList.ClearDataSource(); }
            WebGroupList.DataSource = LstChecked;
            WebGroupList.DataBind();
            WebGroupList.Visible = true;

        }
        else
        {
            //Common.ShowMessageHtml("No records found", 0, true, dvMsgBox);

            WebGroupList.Visible = false;
        }
    }
    catch
    {
    }
}
private void Binddetailsselected(int SellerId)
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];

```

```

try
{
    List<Eteproduct> Lst = (new Products()).GetProductbySellerId(SellerId,
"ProductSearchList");
    if (Lst != null && Lst.Count > 0)
    {
        if (WebDataGrid1.HasDataSource == true) { WebDataGrid1.ClearDataSource(); }
        WebDataGrid1.DataSource = Lst;
        WebDataGrid1.DataBind();
        WebDataGrid1.Visible = true;

    }
    else
    {
        //Common.ShowMessageHtml("No records found", 0, true, dvMsgBox);

        WebGroupList.Visible = false;
    }
}
catch
{
}
}
private void ManageNavigations(int pageid, int userid, int groupid)
{
    bool IsLeftBar = false;
    IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
myBtnClickHandler);

    if (IsLeftBar == true)
    {
        LeftSideBar.Attributes["class"] = "span2";
        MainContent.Attributes["class"] = "span10";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;"); //style="display:
block;"
        PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
    else
    {
        LeftSideBar.Attributes["class"] = "";
        MainContent.Attributes["class"] = "span12";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
        PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
}
protected void Check_SelectedIndexChanged(object sender, EventArgs e)
{
    int selectedGroupid = Convert.ToInt32(0);

```

```

}

// This is Event Handler
protected void myBtnClickHandler(object sender, EventArgs e)
{

    //LinkButton lnk = (LinkButton)sender;
    //string ID = lnk.ID;

    LinkButton lnk = sender as LinkButton;

    if (lnk != null)
    {
        string ID = lnk.ID;
        Common.ShowMessageHtml("Please wait while refreshing the records", 1, true,
dvMsgBox);
        string pid = (Convert.ToInt16(Request.QueryString["pid"]) - 8).ToString();
        DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
        if (ds != null)
        {
            if (ds.Tables[0].Rows.Count > 0)
            {
                DataTable dtdll = ds.Tables[0];
                foreach (DataRow drnew in dtdll.Rows)
                {
                    switch (drnew["lbtncode"].ToString())
                    {
                        case "SAVE":
                            ContentSaveControlClicked();
                            // Call SAVE Method
                            break;
                        case "URL":
                            Response.Redirect(drnew["pageurl"].ToString() + "&pid=" +
drnew["lbtnid"]);
                            break;
                        case "AUDIT":
                            Response.Redirect(drnew["pageurl"].ToString() + "&frompage=" +
Request.QueryString["pageid"]);
                            break;

                        case "CANCEL":
                            // Call Cancel Method
                            break;
                        case "ADDSUB":

                            Response.Redirect("~/ManageSubgroup.aspx?pageid=5&PageType=Add&ProductId=" +
pid);
                            break;
                        case "ADDPRODUCT":

```

```
Response.Redirect("~/ManageProducts.aspx?pageid=5&PageType=Add&ProductId=" + pid);
```

```
        break;  
    default:
```

```
        break;
```

```
    }
```

```
    }
```

```
    }
```

```
    }
```

```
    }
```

```
    DropDownList dropDownList = sender as DropDownList;
```

```
    if (dropDownList != null)
```

```
    {
```

```
        Common.ShowMessageHtml("", 0, false, dvMsgBox);
```

```
        Bindetails_Filter(Table1);
```

```
    }
```

```
}
```

```
private void ContentSaveControlClicked()
```

```
{
```

```
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
```

```
    int SellerId = Convert.ToInt16( Clsusr.UserId);
```

```
    foreach(var data in LstChecked)
```

```
    {
```

```
        if (data.Check == true)
```

```
        {
```

```
            decimal id = data.ProductId;
```

```
            bool ret= new Products().UpdateSellerProduct(id, SellerId);
```

```
        }
```

```
    }
```

```
}
```

```
private void Bindetails_Filter(Table table)
```

```
{
```

```
    try
```

```
    {
```

```
        int brandID = 0;
```

```
        int dealerID = 0;
```

```
        int orderStatusID = 0;
```

```
        int customerType = 0;
```

```
        int dateRange = 0;
```

```
        int OrderHold = 0;
```

```
        foreach (TableRow row in table.Rows)
```

```
        {
```

```
            foreach (TableCell rowcell in row.Cells)
```

```
            {
```

```

        foreach (WebControl contrl in rowcell.Controls)
        {
            DropDownList listDrop = contrl as DropDownList;
            if (listDrop != null)
            {
                if (listDrop.Attributes["LISTTYPE"] != null)
                {
                    string rowcellHead = listDrop.Attributes["LISTTYPE"].ToUpper();
//rowcell.Text.ToUpper();
                    ExtractAll(ref orderStatusID, ref OrderHold, ref brandID, ref
customerType, ref dateRange, ref dealerID, rowcellHead, listDrop);
                }
            }
            else
            {
                continue;
            }
        }
    }
}

```

```

        FilterEnum filterLst = new FilterEnum();
        filterLst.OrderStatus = orderStatusID;
        filterLst.OrderHold = OrderHold;
        filterLst.Brand = brandID;
        filterLst.CustomerType = customerType;
        filterLst.DateRange = dateRange;
        filterLst.Dealer = dealerID;
        //RefreshData(filterLst);
    }
    catch (Exception ex)
    {
        objlog.Trace_Error(5, "DateOrWed", "OrderSearchList", "Bindetails_Filter",
ex.Message);
    }
}

```

```

private static void ExtractAll(ref int orderStatusID, ref int OrderHold, ref int brandID, ref int
customerType, ref int dateRange, ref int dealerID, string rowcellHead, DropDownList
listDrop)
{

```

```

    //string strddl=rowcellHead;
    switch (rowcellHead)
    {
        case "BRAND":
            if (string.Compare(listDrop.SelectedItem.Text, "All", true) == 0)
            {
                brandID = 0;
            }
            else
            {

```

```

        brandID = Convert.ToInt32(listDrop.SelectedValue);
    }
    break;
case "CUSTOMER TYPE":
    string strCustType = listDrop.SelectedItem.Text;
    if (string.Compare(strCustType, "Residential", true) == 0)
    {
        customerType = 1;
    }
    else if (string.Compare(strCustType, "Business", true) == 0)
    {
        customerType = 2;
    }
    else
    {
        customerType = 0;
    }
    break;
case "DATE RANGE":
    dateRange = SelectDateRange(dateRange, listDrop);
    break;
case "DEALER":
    if (string.Compare(listDrop.SelectedItem.Text, "All", true) == 0)
    {
        dealerID = 0;
    }
    else
    {
        dealerID = Convert.ToInt32(listDrop.SelectedValue);
    }

    break;
case "STATUS":
    if (string.Compare(listDrop.SelectedItem.Text, "All", true) == 0)
    {
        orderStatusID = 0;
    }
    else
    {
        orderStatusID = Convert.ToInt32(listDrop.SelectedValue);
    }

    break;
case "ON HOLD":
    string strOnHold = listDrop.SelectedItem.Text;
    if (string.Compare(strOnHold, "Yes", true) == 0)
    {
        OrderHold = 1;
    }
    else if (string.Compare(strOnHold, "No", true) == 0)
    {
        OrderHold = 2;
    }
    else
    {

```

```

        OrderHold = 0;
    }
    break;
default:
    break;
}

}

private static void SetStatus(Table tbl, string LoggedinID, string FilterName)
{
    try
    {
        foreach (TableRow row in tbl.Rows)
        {
            foreach (TableCell rowcell in row.Cells)
            {
                foreach (WebControl contrl in rowcell.Controls)
                {
                    DropDownList listDrop = contrl as DropDownList;
                    if (listDrop != null)
                    {
                        if (listDrop.Attributes["LISTTYPE"] != null &&
listDrop.Attributes["LISTTYPE"].ToUpper() == FilterName)
                        {
                            listDrop.SelectedValue = LoggedinID;
                        }
                    }
                    else
                    {
                        continue;
                    }
                }
            }
        }
    }
    catch (Exception exc)
    {
    }
}

private static int SelectDateRange(int dateRange, DropDownList listDrop)
{
    string strDateRange = listDrop.SelectedItem.Text.ToUpper();
    switch (strDateRange)
    {
        case "ALL":
            {
                dateRange = 0;
                break;
            }
        case "TODAY":

```



```
    {
      dateRange = 1;
      break;
    }

  case "YESTERDAY":
    {
      dateRange = 2;
      break;
    }
  case "THIS WEEK":
    {
      dateRange = 3;
      break;
    }
  case "LAST WEEK":
    {
      dateRange = 4;
      break;
    }

  case "LAST 7 DAYS":
    {
      dateRange = 5;
      break;
    }
  case "LAST 30 DAYS":
    {
      dateRange = 6;
      break;
    }
  case "THIS MONTH":
    {
      dateRange = 7;
      break;
    }
  case "LAST MONTH":
    {
      dateRange = 8;
      break;
    }
  case "THIS YEAR":
    {
      dateRange = 9;
      break;
    }
  case "LAST YEAR":
    {
      dateRange = 10;
      break;
    }
  default:
    dateRange = 0;
    break;
}
```

```
    }  
    return dateRange;  
}
```

```
protected void Lnkbtn_ManageOrder_Click(object sender, EventArgs e)  
{  
    LinkButton lnkbutton = (LinkButton)sender;  
    string strcustid = lnkbutton.CommandName;  
  
    if (strcustid.Length > 0)  
    {  
        string[] stringSeparators = new string[] { "&", "=" };  
        string[] arr1;  
        arr1 = strcustid.Split(stringSeparators, StringSplitOptions.RemoveEmptyEntries);  
        bool bchkstatus = false;  
        foreach (var word in arr1)  
        {  
            if (word == "status")  
            {  
                bchkstatus = true;  
            }  
  
            if (bchkstatus == true && word != "status")  
            {  
                Response.Redirect(Navigations.StatusWiseRedirect(Convert.ToInt32(word),  
strcustid));  
            }  
  
        }  
    }  
}  
  
public class DecimalBooleanConverter : IBooleanConverter  
{  
  
    public object DefaultFalseValue  
    {  
        get { return 0; }  
    }  
  
    public object DefaultTrueValue  
    {  
        get { return 1; }  
    }  
  
    public bool IsFalse(object value)  
    {  
        if (value == null)  
            return true;  
  
        return (int)value <= 1;  
    }  
}
```

```

public bool IsTrue(object value)
{
    if (value == null)
        return false;

    return (int)value > 0;
}
}

}
*****ProductsList.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="ProductsList.aspx.cs" Inherits="ProductsList" %>

<%@ Register Assembly="Infragistics4.Web.v14.2, Version=14.2.20142.2480,
Culture=neutral, PublicKeyToken=7dd5c3163f2cd0cb"
Namespace="Infragistics.Web.UI.GridControls" TagPrefix="ig" %>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
</asp:Content>

<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
    <div class="container-fluid">
        <div class="row-fluid" style="padding-top: 20px;">
            <div class="span2" id="dvlogo" runat="server" style="display: block;">
                <!----%>
            </div>
            <div class="span10">
                <div class="row">
                    <div id="dvMsgBox" class="span12 ordersave" runat="server" style="display:
none;">
                        <asp:Label ID="Label1" runat="server" Text=""></asp:Label>
                    </div>
                </div>
            </div>
        </div>
    </div>
</asp:Content>

<asp:Content ID="Content4" ContentPlaceHolderID="CPH_LeftSideBar" runat="Server">
    <div id="LeftSideBar" runat="server" class="span2">
        <asp:Table ID="Table1" runat="server">
            </asp:Table>
        </div>
    </asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="CPH_MainContent" runat="Server">

    <div class="span10" id="MainContent" runat="server">
        <div id="ShowPageHeader" class="row" style="display: block;" runat="server">
            <div id="PageName" class="span12 customer" runat="server">

```

```

    </div>
</div>

<div id="Div1" class="row" style="display: block;">
  <div id="Div2" class="span12 customertable">

    <!-- Main Content Comes here-->

    <%--<asp:ScriptManager ID="ScriptManager1"
runat="server"></asp:ScriptManager>--%>
    <ig:WebDataGrid ID="WDGrid_UserList" runat="server" CssClass="wdglist"
AutoGenerateColumns="False">
      <Columns>
        <ig:BoundDataField DataFieldName="CatName" Key="CatName">
          <Header Text="Category">
            </Header>
          </ig:BoundDataField>

          <ig:BoundDataField DataFieldName="GroupName" Key="GroupName ">
            <Header Text="Sub-Category">
              </Header>
            </ig:BoundDataField>
          <ig:BoundDataField DataFieldName="ShortCode" Key="ShortCode ">
            <Header Text="Product Name">
              </Header>
            </ig:BoundDataField>
          <ig:BoundDataField DataFieldName="ProductName" Key="ProductName ">
            <Header Text="Product Desc">
              </Header>
            </ig:BoundDataField>
          <ig:BoundDataField DataFieldName="AnalysisCode" Key="AnalysisCode
">
            <Header Text="Search code">
              </Header>
            </ig:BoundDataField>

            <ig:BoundDataField DataFieldName="CreatedOn" Key="CreatedOn"
DataFormatString="{0:dd/MM/yyyy}" Width="100px">
              <Header Text="Created On">
                </Header>
            </ig:BoundDataField>

            <ig:TemplateDataField Key="ProductId" Width="60px">
              <ItemTemplate>
                <asp:HyperLink ID="pcatid" runat="server" CssClass="btn btn-small
btn-primary" Text="Open"
                  NavigateUrl='<% #Eval("ProductId",
~/ManageProducts.aspx?pageid=5&PageType=Edit&ProductId={0})%>'>
                </asp:HyperLink>
              </ItemTemplate>
              <Header Text="Open" />
            </ig:TemplateDataField>
          </Columns>

```

```

        <Behaviors>
            <ig:Paging CurrentPageLinkCssClass="Current" PageLinkCssClass="Link"
PagerCssClass="Pager" PagerMode="NumericFirstLast" PageSize="15" QuickPages="3"
FirstPageText="First" LastPageText="Last" NextPageText="Next"
PreviousPageText="Previous" PagerAppearance="bottom">
            </ig:Paging>

            <ig:Filtering>
            </ig:Filtering>
            <ig:Sorting Enabled="true">
            </ig:Sorting>
        </Behaviors>
    </ig:WebDataGrid>
    <asp:SqlDataSource ID="SqlDataSource1" runat="server"
ConnectionString="<%= $ ConnectionStrings:DownConnectingString %>"
SelectCommand="SELECT [UserId], [UserName], [FirstName], [LastName], [EmailID],
[ContactNum], [CreatedOn], [Status] FROM [tblUser]"></asp:SqlDataSource>

    </div>
</div>
</div>
</asp:Content>

```

*****ProductsList.aspx.cs *****

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBEntity;
using DowBAL;
using System.Data;

public partial class ProductsList : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        //if (!IsPostBack)
        //{
            ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
            if (Clsusr != null)
            {
                string pageid = "6";// Request.QueryString["pageid"];
                ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
Convert.ToInt32(Clsusr.GroupId));
                string msg = Request.QueryString["msg"];
                if (msg != null)
                {
                    Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
                }
                if (Clsusr.GroupId == "1")
                {
                    Bindetails();
                }
            }
        }
    }
}

```

```

    }
}
else
{
    Response.Redirect("MLB_Login.aspx");
}
}

private void ManageNavigations(int pageid, int userid, int groupid)
{
    bool IsLeftBar = false;
    IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
myBtnClickHandler);

    if (IsLeftBar == true)
    {
        LeftSideBar.Attributes["class"] = "span2";
        MainContent.Attributes["class"] = "span10";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;"); //style="display:
block;";
        PageName.InnerText = "Products List";// new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
    else
    {
        LeftSideBar.Attributes["class"] = "";
        MainContent.Attributes["class"] = "span12";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
        PageName.InnerText = "Products List";// new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
}
}

```

// This is Event Handler

```

protected void myBtnClickHandler(object sender, EventArgs e)
{

```

```

    LinkButton lnk = (LinkButton)sender;
    string ID = lnk.ID;

```

```

    DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));

```

```

    if (ds != null)

```

```

    {

```

```

        if (ds.Tables[0].Rows.Count > 0)

```

```

        {

```

```

            DataTable dtdll = ds.Tables[0];

```

```

            foreach (DataRow drnew in dtdll.Rows)

```

```

            {

```

```

                switch (drnew["lbtncode"].ToString())

```

```

                {

```

```

                    case "SAVE":

```

```

                        // Call SAVE Method

```

```

                        break;

```

```

                    case "URL":

```



```

        string pageid = "pageid=" + Session["pageid"].ToString();
        Response.Redirect("~/Managedealer.aspx?PageType=Add&dealername=0&" +
pageid);
    }
}
*****ProductCatList.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="ProductCatList.aspx.cs" Inherits="ProductCatList" %>

<%@ Register Assembly="Infragistics4.WebUI.WebSpellChecker.v14.2,
Version=14.2.20142.1028, Culture=neutral, PublicKeyToken=7dd5c3163f2cd0cb"
Namespace="Infragistics.WebUI.WebDataInput" TagPrefix="igtxt" %>
<%@ Register Assembly="Infragistics4.Web.v14.2, Version=14.2.20142.1028,
Culture=neutral, PublicKeyToken=7dd5c3163f2cd0cb"
Namespace="Infragistics.Web.UI.GridControls" TagPrefix="ig" %>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
</asp:Content>

<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
    <div class="container-fluid">
        <div class="row-fluid" style="padding-top: 20px;">
            <div class="span2" id="dvlogo" runat="server" style="display: block;">
                <!---->
            </div>
            <div class="span10">
                <div class="row">
                    <div id="dvMsgBox" class="span12 ordersave" runat="server" style="display:
none;">
                        <asp:Label ID="Label1" runat="server" Text=""></asp:Label>
                    </div>
                </div>
            </div>
        </div>
    </div>
</asp:Content>

<asp:Content ID="Content4" ContentPlaceHolderID="CPH_LeftSideBar" runat="Server">
    <div id="LeftSideBar" runat="server" class="span2">
        <asp:Table ID="Table1" runat="server">
            </asp:Table>
        </div>
    </asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="CPH_MainContent" runat="Server">

    <div class="span10" id="MainContent" runat="server">
        <div id="ShowPageHeader" class="row" style="display: block;" runat="server">
            <div id="PageName" class="span12 customer" runat="server">
                </div>
            </div>
        </div>

```



```

<div id="Div1" class="row" style="display: block;">
  <div id="Div2" class="span12 customertable">

    <!-- Main Content Comes here-->

    <%--<asp:ScriptManager ID="ScriptManager1"
runat="server"></asp:ScriptManager>--%>

    <ig:WebDataGrid ID="WDGrid_UserList" runat="server" CssClass="wdglist"
AutoGenerateColumns="False">
      <Columns>
        <ig:BoundDataField DataFieldName="pcatdesc" Key="pcatdesc">
          <Header Text="Name">
            </Header>
          </ig:BoundDataField>

          <ig:BoundDataField DataFieldName="pcatdesc" Key="pcatdesc ">
            <Header Text="Description">
              </Header>
            </ig:BoundDataField>

          <ig:BoundDataField DataFieldName="CreatedOn" Key="CreatedOn"
DataFormatString="{0:dd/MM/yyyy}" Width="100px">
            <Header Text="Created On">
              </Header>
            <%-- </ig:BoundDataField>
            <ig:BoundDataField DataFieldName="DealerOperationProfileId"
Key="DealerOperationProfileId">
              <Header Text="Deater Type">
                </Header>
              </ig:BoundDataField>
              <ig:BoundDataField DataFieldName="IsActive" Key="IsActive">
                <Header Text="Is Active">
                  </Header>--%>
              </ig:BoundDataField>

              <ig:TemplateDataField Key="pcatid" Width="60px">
                <ItemTemplate>
                  <asp:HyperLink ID="pcatid" runat="server" CssClass="btn btn-small
btn-primary" Text="Open"
                    NavigateUrl='<% #Eval("pcatid",
"~/ManageProductCat.aspx?pageid=17&PageType=Edit&id={0}")%>'>
                  </asp:HyperLink>
                </ItemTemplate>
                <Header Text="Open" />
              </ig:TemplateDataField>
            </Columns>
            <Behaviors>
              <ig:Paging CurrentPageLinkCssClass="Current" PageLinkCssClass="Link"
PagerCssClass="Pager" PagerMode="NumericFirstLast" PageSize="15" QuickPages="3"
FirstPageText="First" LastPageText="Last" NextPageText="Next"
PreviousPageText="Previous" PagerAppearance="bottom">
                </ig:Paging>

```

```

        <ig:Filtering>
        </ig:Filtering>
        <ig:Sorting Enabled="true">
        </ig:Sorting>
        </Behaviors>
    </ig:WebDataGrid>
    <asp:SqlDataSource ID="SqlDataSource1" runat="server"
    ConnectionString="<%= $ ConnectionStrings:DowConectingString %>"
    SelectCommand="SELECT [UserId], [UserName], [FirstName], [LastName], [EmailID],
    [ContactNum], [CreatedOn], [Status] FROM [tblUser]"></asp:SqlDataSource>

    </div>
</div>
</div>
</asp:Content>

```

***** ProductCatList.aspx.cs *****

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBEntity;
using DowBAL;
using System.Data;

public partial class ProductCatList : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        //if (!IsPostBack)
        //{
            ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
            if (Clsusr != null)
            {
                string pageid = "6";//Request.QueryString["pageid"];
                ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
                Convert.ToInt32(Clsusr.GroupId));
                string msg = Request.QueryString["msg"];
                if (msg != null)
                {
                    Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
                }
                if (Clsusr.GroupId == "1")
                {
                    Bindetails();
                }
            }
            else
            {
                Response.Redirect("MLB_Login.aspx");
            }
        }
    }
}

```

```

private void ManageNavigations(int pageid, int userid, int groupid)
{
    bool IsLeftBar = false;
    IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
myBtnClickHandler);

    if (IsLeftBar == true)
    {
        LeftSideBar.Attributes["class"] = "span2";
        MainContent.Attributes["class"] = "span10";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;"); //style="display:
block;"
        PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
    else
    {
        LeftSideBar.Attributes["class"] = "";
        MainContent.Attributes["class"] = "span12";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
        PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
}

```

// This is Event Handler

```

protected void myBtnClickHandler(object sender, EventArgs e)
{

    LinkButton lnk = (LinkButton)sender;
    string ID = lnk.ID;

    DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
    if (ds != null)
    {
        if (ds.Tables[0].Rows.Count > 0)
        {
            DataTable dtdll = ds.Tables[0];
            foreach (DataRow drnew in dtdll.Rows)
            {
                switch (drnew["lbtncode"].ToString())
                {
                    case "SAVE":
                        // Call SAVE Method
                        break;
                    case "URL":
                        Response.Redirect(drnew["pageurl"].ToString());
                        break;
                    case "AUDIT":
                        Response.Redirect(drnew["pageurl"].ToString() + "&frompage=" +
Request.QueryString["pageid"]);
                        break;
                    case "CANCEL":

```

```

        Response.Redirect("Home.aspx?pageid=1");
        break;
    case "NEW":
Response.Redirect("ManageProductCat.aspx?pageid=17&pagetype=Add");
        break;
    default:
        break;
    }
}
}
}
}

private void Bindetails()
{
    //try
    //{

        List<Eteproductcat> Lst = (new Products()).GetProductCats("");

        if (Lst != null && Lst.Count > 0)
        {
            WDGrid_UserList.DataSource = Lst;
            WDGrid_UserList.DataBind();
            WDGrid_UserList.Visible = true;
        }
        else
        {
            WDGrid_UserList.Visible = false;
        }
    }
    //protected override void OnInit(EventArgs e)
    //{
    //    base.OnInit(e);

    //    if (this.Master is MainMaster)
    //    {
    //        ((MainMaster)this.Master).MasterControlClicked += new
EventHandler(ContentControlClicked);
    //    }
    //}
    private void ContentControlClicked(object sender, EventArgs e)
    {
        string pageid = "pageid=" + Session["pageid"].ToString();
        Response.Redirect("~/Managedealer.aspx?PageType=Add&dealername=0&" +
pageid);
    }
}
}

```

*****ProductDisplay.aspx *****

```
<%@ Page Title="B2BPortal" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="ProductDisplay.aspx.cs" Inherits="ProductDisplay" %>
```

```
<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0"/>
```

```
<link rel="stylesheet" href="styles.css" type="text/css" />
```

```
<script type="text/javascript"
```

```
src="http://ajax.googleapis.com/ajax/libs/jquery/1.4.2/jquery.min.js"></script>
```

```
</asp:Content>
```

```
<asp:Content ID="Content2" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
```

```
<div class="container-fluid">
```

```
<div class="row-fluid" style="padding-top: 20px;">
```

```
<div class="span2" id="dvlogo" runat="server" style="display: block;">
```

```

```

```
</div>
```

```
<div class="span10">
```

```
<div class="row">
```

```
<%--<h3>Ravikiran Techno Soft</h3>--%>
```

```
<div id="dvMsgBox" class="span12 ordersave" runat="server" style="display:
none;">
```

```
<asp:Label ID="Label2" runat="server" Text="Message"></asp:Label>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</asp:Content>
```

```
<asp:Content ID="Content3" ContentPlaceHolderID="CPH_LeftSideBar" runat="Server">
```

```
<div id="LeftSideBar" runat="server" class="span2">
```

```
<asp:Table ID="Table1" runat="server" Width="250px">
```

```
</asp:Table>
```

```
</div>
```

```
</asp:Content>
```

```
<asp:Content ID="Content4" ContentPlaceHolderID="CPH_MainContent" runat="Server" >
```

```
<div class="span10" id="MainContent" runat="server">
```

```
<div id="ShowPageHeader" class="row" style="display: block;" runat="server">
```

```
<div id="PageName" class="span12 customer" runat="server">
```

```
</div>
```

```
</div> .
```

```
<%--<asp:HiddenField ID="HF_HasStoppedService" runat="server" />--%>
```

```

    <div id="Div1" class="row" style="display: block;">
      <div id="Div2" class="span12">
<div id="page">

```

<h1>Image gallery with description in 4 lines of jQuery code</h1>

```

<div id="gallery" runat="server">
  <div id="panel">
    
    <div id="description">1st image description</div>
  </div>
  <div id="thumbs" >

```

```

    <img src='images/vendor/2-6-1.png' class='thumb' alt='1'
style="height:100px;width:100px;" />
<img src='images/Makeup.png' class='thumb' alt='2' height='100px' width='100px'/>
<img src='images/makeup2.png' class='thumb' alt='3' height='100px' width='100px'/>
<img src='images/Jewellery1.png' class='thumb' alt='4' height='100px' width='100px'/>
<img src='images/Mehendi.png' class='thumb' alt='5' height='100px' width='100px'/>

  </div>

```

```

</div>
<script>

```

```

  $('#thumbs').delegate('img', 'click', function () {
    $('#largeImage').attr('src', $(this).attr('src').replace('thumb', 'large'));
    $('#description').html($(this).attr('alt'));
  });

```

```

</script>
</div>

```

```

<div class="content_holder">

```

```

<%--      <h5>Ravikiran Techno Soft </h5>--%>
<div class="span12" style="background-color: #f8f9fa;" >

```

```

  <div class="row-fluid">

```

```

    <div class="span7" style="width: 140%">

```

```

      <div class="row">

```

```

        <div class="span1" style="width: 1%" >

```

```

          <p class="customer" > <a href='DraftOrders.aspx?pageid=38'></a>

```

```

</p>

```

```

      </div>

```

```

        <div class="span4" style="width: 17%" >

```

```

      </div>

```

```

        <div class="span3" style="width: 17%">

```

```

        </div>
        <div class="span3" style="width: 30%">
        <div class="span3" id="dv_RecentNews" runat="server" style="width:
90%" >
            <table class="table">
            <%-- display product details--%>

                </table>
            </div>

        </div>

    </div>
</div>

    </div>
</div>

    </div>
</div>

    </div>
</div>

    </div>
</div>
</asp:Content>

```

```
*****ProductDisplay.aspx.cs *****
```

```

using DowBAL;
using DowBEntity;
using DowDAL;
using System;
using System.Collections.Generic;
using System.Data;
using System.Linq;
using System.Text;
using System.Text.RegularExpressions;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.Security;

```

```

public partial class ProductDisplay : System.Web.UI.Page
{
    DDLogTracer objlog = new DDLogTracer();
    public string UserName = "";

```

```

public string CancellationPerLast7DaysWithPerCent = "",
CancellationPerLast28DaysWithPerCent = "";
protected void Page_Load(object sender, EventArgs e)
{
    if (Request.Browser.IsMobileDevice)
    {
        Response.Redirect("~/M.Home.aspx?pageid=1", true);
    }

    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    if (Clsusr != null)
    {
        UserName = Clsusr.UserName;
    }
    else
    {

    }

    string pageid = Request.QueryString["pageid"];
    if (pageid == null) { pageid = "1"; }
    Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    if (Clsusr != null)
    {

        ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
Convert.ToInt32(Clsusr.GroupId));
    }
    string msg = Request.QueryString["msg"];
    if (msg != null)
    {
        Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
    }

    Bindetails(Convert.ToInt16( Clsusr.UserId));
    FillRecentNews();
}
private void ManageNavigations(int pageid, int userid, int groupid)
{
    bool IsLeftBar = false;
    IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
myBtnClickHandler);

    if (IsLeftBar == true)
    {
        LeftSideBar.Attributes["class"] = "span2";
        MainContent.Attributes["class"] = "span10";
        ShowPageHeader.Attributes.CssStyle.Add("display", "none;"); //style="display:
block;"
        PageName.InnerText = new
tblPagess().GettblPageNameById(Convert.ToInt32(pageid));
    }
    else

```



```

    {
        LeftSideBar.Attributes["class"] = "";
        MainContent.Attributes["class"] = "span12";
        ShowPageHeader.Attributes.CssStyle.Add("display", "none;");
    }
}

protected void guest_login(string user)
{

    string City = "Hyderabad";// _City;
    string Country = "India";// _Country;

    BALCreateUser objlogin = new BALCreateUser();
    DataTable status_login = objlogin.LoginGuest(user, user,
Request.ServerVariables["REMOTE_ADDR"].ToString(), "", City, Country);
    string login_status = status_login.Rows[0]["UserStatus"].ToString();

    ClsLoggedInUser objLoggedInUser;
    if (login_status == "True")
    {
        objLoggedInUser = new ClsLoggedInUser();

        objLoggedInUser.UserId = status_login.Rows[0]["UserId"].ToString();
        objLoggedInUser.UserName = status_login.Rows[0]["UserName"].ToString();
        objLoggedInUser.GroupId = status_login.Rows[0]["GroupId"].ToString();
        objLoggedInUser.LoginStatus = "True";
        objLoggedInUser.LoginID = Convert.ToInt32(status_login.Rows[0]["LoginId"]);
        objLoggedInUser.DealerId = Convert.ToInt32(status_login.Rows[0]["DealerId"]);
        objLoggedInUser.RoleId = Convert.ToInt32(status_login.Rows[0]["RoleId"]);
        objLoggedInUser.BrandId = 1; //Convert.ToInt32(status_login.Rows[0]["BrandId"]);

        Session["LoggedUser_Specs"] = objLoggedInUser;

        FormsAuthentication.SetAuthCookie(objLoggedInUser.UserName, false);
        HttpCookie myCookie = new HttpCookie("B2BPortalCookie");
        DateTime now = DateTime.Now;
        myCookie.Value = objLoggedInUser.UserName;
        myCookie.Expires = now.AddYears(1);

        Response.Cookies.Add(myCookie);

        ClsPermissioGroup clsperGrp = new ClsPermissioGroup();
        UserPermissions Entuserper =
clsperGrp.GetUserLandingPage(objLoggedInUser.UserName);

        if (Entuserper.Pageld != 0)
        {
            Entuserper.Add = true;
            Entuserper.Edit = true;
            Entuserper.Delete = true;
            Entuserper.View = true;
            //BalRecentPlacess objRemoveRcntPlc = new BalRecentPlacess();

```

```

        //bool rst =
objRemoveRcntPlc.DeleteRecentPlaces(Convert.ToInt32(status_login.Rows[0]["UserId"].ToString()));
        Response.Redirect("~/ " + Entuserper.PageUrl);

    }

}
// This is Event Handler
private void Bindetails(int UserId)
{
    StringBuilder sb = new StringBuilder();

    var list = new Products().GetProductSellerImage(1, "").ToList();
    foreach (var dr in list)
    {
        sb.Append("\n");
    }
    string PreHtml = sb.ToString();
    if (PreHtml.Length > 0)
    {
        objlog.Trace_Log(4, "Gordersearchlist", "1", "SliderList", PreHtml);
        //thumbs.InnerHtml = PreHtml;
    }

}

protected void myBtnClickHandler(object sender, EventArgs e)
{
    LinkButton lnk = sender as LinkButton;

//    LinkButton lnk = (LinkButton)sender;
    if (lnk != null)
    {

        string ID = lnk.ID;

        DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
        if (ds != null)
        {
            if (ds.Tables[0].Rows.Count > 0)
            {
                DataTable dtdll = ds.Tables[0];
                foreach (DataRow drnew in dtdll.Rows)
                {
                    switch (drnew["lbtncode"].ToString())
                    {
                        case "SAVE":
                            // Call SAVE Method
                            break;
                        case "URL":
                    }
                }
            }
        }
    }
}

```



```

    }
    PreHtml = sb.ToString();
    if (PreHtml.Length > 0)
    {
        PreHtml += "</table>";
        dv_RecentNews.InnerHtml = PreHtml;
    }
}

}

}
***** response.aspx *****
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="response.aspx.cs"
Inherits="response" %>
    <%@ Import Namespace = "System.IO" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Zaakpay</title>
<link href=" ../stylesheets/styles-payflow.css" rel="stylesheet" type="text/css" />

<link href=" ../javascripts/txnpage/sort.css" rel="stylesheet" type="text/css" />
<script type="text/javascript" src=" ../javascripts/txnpage/sort_files/jquery.js"></script>
    <script type="text/javascript"
src=" ../javascripts/txnpage/sort_files/interface.js"></script>
<!--this is use for remove captchasection color or image-->

<!--popup_layer-->

</head>
<body>

<%
    String secretKey = "Insert your secret key here";
    String allParamValue = null;
    Boolean isChecksumValid = false;
    allParamValue =
ChecksumResponse.ChecksumCalculatorResponse.GetAllNotEmptyParamValue(Request).
Trim();
    String checksum =
ChecksumResponse.ChecksumCalculatorResponse.calculateChecksum(secretKey,
allParamValue);
    System.Diagnostics.Debug.WriteLine("allParamValue Response : " + allParamValue);
    System.Diagnostics.Debug.WriteLine("secretKey Response : " + secretKey);
    if(checksum != null){

```

```

        isChecksumValid =
ChecksumResponse.ChecksumCalculatorResponse.verifyChecksum(secretKey,
allParamValue, checksum);

    }

    %>
<center>

<table>

<tr>
    <td align = "center">OrderId</td>
    <td align = "center"><%= Request.Params.Get("orderId")%> </td>
</tr>
<tr>
    <td align = "center">Response Code</td>
    <%if (isChecksumValid)
    { %>
    <td align = "center"> <%=Request.Params.Get("responseCode")%></td>
    <%}
    else
    { %>
    <td align ="center"><font color ="red">***</font></td>
    <%} %>

</tr>
<tr>
    <td align = "center">Response Description</td>
    <%if (isChecksumValid)
    { %>
    <td align = "center"> <%=Request.Params.Get("responseDescription")%></td>
    <%}
    else
    { %>
    <td align ="center"><font color ="red">The Response is Compromised.</font></td>
    <%} %>

</tr>

<tr>
    <td align = "center">Checksum Valid</td>
    <%if (isChecksumValid)
    { %>
    <td align = "center">Yes</td>
    <%}
    else
    { %>
    <td align ="center"><font color ="red">No </font><br /><br />The Transaction might
have been Successfull.</td>

    <%} %>

</tr>

```

</table>

</center>

</body>

</html>

*****response.aspx.cs *****

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBAL;
using DowBEntity;
using System.Text;
using System.Security;
using System.Net;
using System.Data;
using System.Web.Security;
```

```
public partial class Services : System.Web.UI.Page
```

```
{
```

```
    DDLogTracer objlog = new DDLogTracer();
```

```
    protected void Page_Load(object sender, EventArgs e)
```

```
    {
```

```
        Session["page"] = "1";
```

```
        string user;
```

```
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];;
```

```
        if (Clsusr == null)
```

```
        {
```

```
            HttpCookie myCookie = Request.Cookies["DateOrWedCookie"];;
```

```
            if (myCookie != null)
```

```
            {
```

```
                user = myCookie.Value.ToString();
```

```
                guest_login(user);
```

```
                Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];;
```

```
            }
```

```
        }
```

```
        var sb = new StringBuilder();
```

```
        int srno = 0;
```

```
        decimal amt=0;
```

```
        List<EteMenuList> Lst = new tblPagess().GetMenuListbystype(2);//new
```

```
List<EteMenuList>();//
```

```
        var lst = (from c in Lst where c.Mlevel == 1 orderby c.priority select c);
```

```
        foreach (var dt in lst)
```

```
        {
```

```
            srno++;
```

```

        sb.Append(" <a href='#menu' + dt.pageid + " class='list-group-item collapsed' data-
toggle='collapse' data-parent='#sidebar' aria-expanded='false'> <span class='hidden-sm-
down'><i class='fas fa-chevron-right'
style='font-size: 12px;'></i></span></a>\n");
        sb.Append("<div class='collapse' id='menu' + dt.pageid + ">\n");
        var data = (from d in Lst where d.parentid == dt.pageid orderby d.priority select d);
        foreach (var data1 in data)
        {
            sb.Append("<a href='" + data1.pageurl + " class='list-group-item' data-
parent='#menu' + dt.pageid + "><i class='fas " + data1.BtnImg + "></i> " + data1.pagename
+ "</a>\n");
        }
        sb.Append("</div>\n");
    }
    //menu.InnerHtml = sb.ToString();
    //objlog.Trace_Error(5, "DateOrWed", "services", "pageLoad", sb.ToString());
    sb = new StringBuilder();

}
protected void guest_login(string user)
{
    string City = "Hyderabad";// _City;
    string Country = "India";// _Country;
    string ip = System.Net.Dns.GetHostEntry
        (System.Net.Dns.GetHostName()).AddressList.GetValue(0).ToString(); string
strHostName = System.Net.Dns.GetHostName();
    IPHostEntry ipEntry = System.Net.Dns.GetHostEntry(strHostName);
    IPAddress[] addr = ipEntry.AddressList;
    string ip1 = addr[1].ToString();

    BALCreateUser objlogin = new BALCreateUser();
    DataTable status_login = objlogin.LoginGuest(user, user,
Request.ServerVariables["REMOTE_ADDR"], "", City, Country);
    string login_status = status_login.Rows[0]["UserStatus"].ToString();

    CIsLoggedInUser objLoggedInUser;
    if (login_status == "True")
    {
        objLoggedInUser = new CIsLoggedInUser();

        objLoggedInUser.UserId = status_login.Rows[0]["UserId"].ToString();
        objLoggedInUser.UserName = status_login.Rows[0]["UserName"].ToString();
        objLoggedInUser.GroupId = status_login.Rows[0]["GroupId"].ToString();
        objLoggedInUser.LoginStatus = "True";
        objLoggedInUser.LoginID = Convert.ToInt32(status_login.Rows[0]["LoginId"]);
        objLoggedInUser.DealerId = Convert.ToInt32(status_login.Rows[0]["DealerId"]);
        objLoggedInUser.RoleId = Convert.ToInt32(status_login.Rows[0]["RoleId"]);
        objLoggedInUser.BrandId = 1; //Convert.ToInt32(status_login.Rows[0]["BrandId"]);

        Session["LoggedUser_Specs"] = objLoggedInUser;

        FormsAuthentication.SetAuthCookie(objLoggedInUser.UserName, false);
    }
}

```

```

    HttpCookie myCookie = new HttpCookie("DateOrWedCookie");
    DateTime now = DateTime.Now;
    myCookie.Value = objLoggedInUser.UserName;
    myCookie.Expires = now.AddYears(1);

    Response.Cookies.Add(myCookie);

}
}

}
***** ServicesList.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/ServicesMasterPage.master"
AutoEventWireup="true" CodeFile="ServicesList.aspx.cs" Inherits="ServicesList" %>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
Runat="Server">
</asp:Content>
<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" Runat="Server">
    <div class="col-md-9 col-sm-12 col-xs-12">
        <div class="row">
            <div id="dvMsgBox" class="span10 ordersave" runat="server"
style="display: block; height:100px">
                <asp:Label ID="Label1" runat="server" Text=""></asp:Label>
            </div>
        </div></div>

</asp:Content>

<asp:Content ID="Content5" ContentPlaceHolderID="CPH_MainContent" Runat="Server">
<!-- Content Bodey Here -->

    <!-- START MAIN CONTAIN -->
    <div class="main-contain shop-product bg-white">
        <div class="container">
            <div class="row">
                <aside>
                    <div id="sidebar" class="col-md-3 col-sm-12 col-xs-12 sidebar" runat="server">
                        <!-- CATEGORIES WIDGET -->
                        <div id="Filter" class="widget mb30" runat="server">
                            <h6 class="text-uppercase bottom-line">Show Results for</h6>
                            <ul class="icons-list">
                                <%--
                                right">12</span></a></li>
                                    <li><a href="#">Sarees <span class="pull-
                                    <li><a href="#">Lehengas <span class="pull-right">11</span></a></li>
                                    <li><a href="#">Indo Westers <span class="pull-right">10</span></a></li>
                                    <li><a href="#">Salwar Kameez <span class="pull-right">14</span></a></li>
                                    <li><a href="#">Kurtees <span class="pull-right">15</span></a></li>--%>
                                </ul>
                            </div>
                        <!-- END CATEGORIES WIDGET -->

                        <!-- SIDE-BAR PRODUCT WIDGET -->
                        <div id="toprate" class="" runat="server">

```



```

<div class="widget mb30">
  <h6 class="text-uppercase bottom-line">Top Rated</h6>
</div>
<%--
  <div class="small-thumb-proudct mb30">
    <div class="thumb-product">
      
      <div class="thumb-product-inner">
        <h6><a href="shoppingdetail.aspx">Printed Su..</a></h6>
        <span class="thumb-product-cat">Fashion</span>
      </div>
      <ul class="list-inline overflow-h">
        <li class="thumb-product-price line-through">$60.00</li>
        <li class="thumb-product-price">$30.00</li>
      </ul>
    </div>
  </div>
  <div class="small-thumb-proudct mb30">
    <div class="thumb-product">
      
      <div class="thumb-product-inner">
        <h6><a href="shoppingdetail.aspx">Printed Su..</a></h6>
        <span class="thumb-product-cat">Fashion</span>
      </div>
      <ul class="list-inline overflow-h">
        <li class="thumb-product-price">$65.00</li>
      </ul>
    </div>
  </div>
  <div class="small-thumb-proudct mb30">
    <div class="thumb-product">
      
      <div class="thumb-product-inner">
        <h6><a href="shoppingdetail.aspx">Printed Su..</a></h6>
        <span class="thumb-product-cat">Fashion</span>
      </div>
      <ul class="list-inline overflow-h">
        <li class="thumb-product-price line-through">$75.00</li>
        <li class="thumb-product-price">$65.00</li>
      </ul>
    </div>
  </div>--%>
</div>
<!-- END SIDE-BAR PRODUCT WIDGET -->

<!-- SIDE-BAR PRODUCT WIDGET -->
<div id="bestseller" class="" runat="server">
  <div class="widget mb30">
    <h6 class="text-uppercase bottom-line">Best Sellers</h6>
  </div>
  <%--
    <div class="small-thumb-proudct mb30">
      <div class="thumb-product">

```

```

alt="">
    
        <h6><a href="shoppingdetail.aspx">Printed Su..</a></h6>
        <span class="thumb-product-cat">Fashion</span>
    </div>
    <ul class="list-inline overflow-h">
        <li class="thumb-product-price line-through">$95.00</li>
        <li class="thumb-product-price">$75.00</li>
    </ul>
</div>
</div>
<div class="small-thumb-proudct mb30">
    <div class="thumb-product">
        
        <div class="thumb-product-inner">
            <h6><a href="shoppingdetail.aspx">Printed Su..</a></h6>
            <span class="thumb-product-cat">Fashion</span>
        </div>
        <ul class="list-inline overflow-h">
            <li class="thumb-product-price line-through">$70.00</li>
            <li class="thumb-product-price">$60.00</li>
        </ul>
    </div>
</div>
<div class="small-thumb-proudct mb30">
    <div class="thumb-product">
        
        <div class="thumb-product-inner">
            <h6><a href="shoppingdetail.aspx">Printed Su..</a></h6>
            <span class="thumb-product-cat">Fashion</span>
        </div>
        <ul class="list-inline overflow-h">
            <li class="thumb-product-price">$65.00</li>
        </ul>
    </div>
</div>
--%>
    </div>

```

<!-- END SIDE-BAR PRODUCT WIDGET -->

```

<!-- TAGS WIDGET -->
<div class="widget">
    <h6 class="text-uppercase bottom-line">Tags</h6>
    <div class="tags">
        <a href="#">Fashion</a>
        <a href="#">Electronics</a>
        <a href="#">Services</a>
        <a href="#">Businesses</a>
        <a href="#">Shops</a>
        <a href="#">Markets</a>
    </div>
</div>

```

```

        </div>
    </div>
    <!-- END TAGS WIDGET -->
</div>
</aside>

<div class="col-md-9 col-sm-12 col-xs-12">

        <div class="row hidden-xs">
<%--
            <div class="toolbar">
                <div id="results" class="col-sm-4 category-num" runat="server">
                    <h2>Designer</h2>
                    <small class="category-num-coun">9 Results</small>
                </div>
            </div>--%>
            <div class="col-sm-8 pull-right">
                <ul class="view-mode">
                    <li class="grid-list-icons">
<%--
                        <asp:Button ID="btn1" runat="server" OnClick="btn1_Click"
type="button" class="btn btn-floating color-green " Text="List" /><i class='fa fa-th-list'></i>
                        <asp:Button ID="btn2" runat="server" OnClick="btn2_Click" type="button"
class="btn btn-floating color-green " Text="Grid" /><i class='fa fa-th'></i> wrn-btn--%>
                            <asp:ImageButton ID="ImgBtn1" runat="server"
ImageUrl="images/small/list.png" OnClick="ImgBtn1_Click" />
                            <asp:ImageButton ID="ImgBtn2" runat="server"
ImageUrl="images/small/grid.png" OnClick="ImgBtn2_Click" />
<%--
                        <button id="Button1" type="button" class="btn btn-secondary wrn-
btn mt-0 ml-0" onclick="btn1_Click" ><i class='fa fa-th-list'></i>&nbsp;&nbsp;&nbsp;List</button>
                        <button id="Button2" type="button" style="width:10px" class="btn btn-
product " onclick="btn2_Click" ><i class='fa fa-th'></i></button>--%>
                    </li>
                    <li class="sort-list-btn">
                        <h6>Sort By :</h6>
                        <div id="sortbtn" class="btn-group" runat="server">
                            <button type="button" class="btn btn-default dropdown-toggle"
data-toggle="dropdown">
                                Relevent <span class="caret"></span>
                            </button>
                            <ul class="dropdown-menu" role="menu">
                                <li><a href="#">Relevent</a></li>
                                <li><a href="#">Price High to Low</a></li>
                                <li><a href="#">Price Low to High</a></li>
                                <li><a href="#">New Arrived</a></li>
                            </ul>
                        </div>
                    </li>
                    <li class="sort-list-btn">
                        <h6>Show :</h6>
                        <div class="btn-group">
                            <button type="button" class="btn btn-default dropdown-toggle"
data-toggle="dropdown">
                                30 <span class="caret"></span>
                            </button>
                            <ul class="dropdown-menu" role="menu">
                                <li><a href="#">30</a></li>

```

```

                <li><a href="#">15</a></li>
                <li><a href="#">9</a></li>
                <li><a href="#">6</a></li>
            </ul>
        </div>
    </li>
</ul>
</div>

</div>
<div class="col-md-9 col-sm-12 col-xs-12">
    <div class="row hidden-xs">
        <div class="toolbar">
            <div id="results" class="col-sm-8 category-num" runat="server">
                <h2>Venues</h2>
                <small class="category-num-coun">9 Results</small>
            </div>
        </div>
    </div>
</div>
</div>
<!--dynamic code from here-->
<div id="appvendors" class="category-products" runat="server" style="display:block">

        </div>
<!--dynamic code ends here-->

    <div id="AppGrid" class="category-products col-grid-style" runat="server"
style="display:none">
</div>
        <!--dynamic code ends here-->

<!-- Start Pagination-->
    <div id="pagenos" class="cd-pagination" runat="server">
        <ul class="pagination">
            <li><a href="#"><i class="fa fa-angle-left" aria-hidden="true"></i></a></li>
            <li class="active"><a href="#">1</a></li>
            <li><a href="#">2</a></li>
            <li><a href="#">3</a></li>
            <li><a href="#">4</a></li>
            <li><a href="#">5</a></li>
            <li><a href="#"><i class="fa fa-angle-right" aria-hidden="true"></i></a></li>
        </ul>
    </div>
<!-- End Pagination-->
</div>
</div>
        </div>

<!-- END MAIN CONTAIN -->

<!-- ./Content Bodey Here -->

```

```
</asp:Content>
<asp:Content ID="Content6" ContentPlaceHolderID="ContentFooter" Runat="Server">
</asp:Content>
```

```
***** ServicesList.aspx.cs*****
```

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBAL;
using DowBEntity;
using System.Text;
using System.Security;
using System.Net;
using System.Data;
using System.Web.Security;

public partial class Services : System.Web.UI.Page
{
    DDLogTracer objlog = new DDLogTracer();

    protected void Page_Load(object sender, EventArgs e)
    {
        Session["page"] = "1";

        string user;
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr == null)
        {
            HttpCookie myCookie = Request.Cookies["DateOrWedCookie"];
            if (myCookie != null)
            {
                user = myCookie.Value.ToString();
                guest_login(user);
                Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
            }

        }

        var sb = new StringBuilder();
        int srno = 0;
        decimal amt=0;
        List<EteMenuList> Lst = new tblIPagess().GetMenuListbystype(2);//new
        List<EteMenuList>();//
        var lst = (from c in Lst where c.Mlevel == 1 orderby c.priority select c);
        foreach (var dt in lst)
        {
            srno++;
        }
    }
}
```

```

        sb.Append(" <a href='#menu' + dt.pageid + "" class='list-group-item collapsed' data-
toggle='collapse' data-parent='#sidebar' aria-expanded='false'> <span class='hidden-sm-
down'><i class='fas fa-chevron-right'
style='font-size: 12px;'></i></span></a>\n");
        sb.Append("<div class='collapse' id='menu' + dt.pageid + "">\n");
        var data = (from d in Lst where d.parentid == dt.pageid orderby d.priority select d);
        foreach (var data1 in data)
        {
            sb.Append("<a href="" + data1.pageurl + "" class='list-group-item' data-
parent='#menu' + dt.pageid + ""><i class='fas " + data1.BtnImg + ""></i> " + data1.pagename
+ ""</a>\n");
        }
        sb.Append("</div>\n");
    }
    //menu.InnerHtml = sb.ToString();
    //objlog.Trace_Error(5, "DateOrWed", "services", "pageLoad", sb.ToString());
    sb = new StringBuilder();

}
protected void guest_login(string user)
{
    string City = "Hyderabad";// _City;
    string Country = "India";// _Country;
    string ip = System.Net.Dns.GetHostEntry
        (System.Net.Dns.GetHostName()).AddressList.GetValue(0).ToString(); string
strHostName = System.Net.Dns.GetHostName();
    IPHostEntry ipEntry = System.Net.Dns.GetHostEntry(strHostName);
    IPAddress[] addr = ipEntry.AddressList;
    string ip1 = addr[1].ToString();

    BALCreateUser objlogin = new BALCreateUser();
    DataTable status_login = objlogin.LoginGuest(user, user,
Request.ServerVariables["REMOTE_ADDR"], "", City, Country);
    string login_status = status_login.Rows[0]["UserStatus"].ToString();

    CIsLoggedInUser objLoggedInUser;
    if (login_status == "True")
    {
        objLoggedInUser = new CIsLoggedInUser();

        objLoggedInUser.UserId = status_login.Rows[0]["UserId"].ToString();
        objLoggedInUser.UserName = status_login.Rows[0]["UserName"].ToString();
        objLoggedInUser.GroupId = status_login.Rows[0]["GroupId"].ToString();
        objLoggedInUser.LoginStatus = "True";
        objLoggedInUser.LoginID = Convert.ToInt32(status_login.Rows[0]["LoginId"]);
        objLoggedInUser.DealerId = Convert.ToInt32(status_login.Rows[0]["DealerId"]);
        objLoggedInUser.RoleId = Convert.ToInt32(status_login.Rows[0]["RoleId"]);
        objLoggedInUser.BrandId = 1; //Convert.ToInt32(status_login.Rows[0]["BrandId"]);

        Session["LoggedUser_Specs"] = objLoggedInUser;

        FormsAuthentication.SetAuthCookie(objLoggedInUser.UserName, false);
    }
}

```

```

    HttpCookie myCookie = new HttpCookie("DateOrWedCookie");
    DateTime now = DateTime.Now;
    myCookie.Value = objLoggedInUser.UserName;
    myCookie.Expires = now.AddYears(1);

    Response.Cookies.Add(myCookie);

}
}

}
*****Services.aspx *****
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Services.aspx.cs"
Inherits="Services" %>

<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
  <meta http-equiv="x-ua-compatible" content="ie=edge">
  <title>DateOrWed</title>
  <link rel="icon" href="img/favicon.png" type="image/gif">
  <!-- Font Awesome -->
  <link rel="stylesheet" href="font/fontawesome/css/all.css">
  <!-- Bootstrap core CSS -->
  <link href="css/bootstrap.min.css" rel="stylesheet">
  <!-- Material Design Bootstrap -->
  <link href="css/mdb.min.css" rel="stylesheet">
  <!-- Your custom styles (optional) -->
  <link href="css/style.css" rel="stylesheet">
  <link rel="stylesheet" type="text/css" href="shopping/css/slick.css"/>
  <link rel="stylesheet" type="text/css" href="shopping/css/slick-theme.css"/>
  <script type="text/javascript" src="shopping/js/slick.js"></script>
  <!-- slick-slider-->
  <script type="text/javascript" src="shopping/js/slick-slider.js"></script>

</head>

<body>
  <form runat="server">
    <asp:ScriptManager ID="ScriptManager1" runat="server">
      </asp:ScriptManager>
    <!-- Start your project here-->
    <!--Navbar -->
    <nav class="mb-1 navbar fixed-top navbar-expand-lg navbar-light bg-light">
      <a class="navbar-brand" href="Home.aspx"></a>
      <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent-4"
        aria-controls="navbarSupportedContent-4" aria-expanded="false" aria-label="Toggle navigation">
        <span class="navbar-toggler-icon"></span>

```

```

</button>
<div class="collapse navbar-collapse" id="navbarSupportedContent-4">
  <ul class="navbar-nav ml-auto">
    <li class="nav-item">
      <a class="nav-link text-center" href="VendorHome.aspx">
        <i class="fas fa-store-alt"></i><br> Sell</a>
      </li>
    <li class="nav-item">
      <a class="nav-link text-center" href="#">
        <i class="far fa-question-circle"></i><br> Help</a>
      </li>
    <li class="nav-item">
      <a class="nav-link text-center" href="MyMessages.aspx">
        <i class="fas fa-envelope-open-text"></i><br> Messages</a>
      </li>
    <li class="nav-item">
      <a class="nav-link text-center" href="MyCart.aspx">
        <i class="fas fa-shopping-bag"></i><br> My Cart</a>
      </li>
    <li class="nav-item">
      <a class="nav-link text-center" href="MyItinerary.aspx">
        <i class="fas fa-calendar-alt"></i><br> My Itinerary</a>
      </li>

    <li class="nav-item dropdown">
      <a class="nav-link dropdown-toggle text-center" id="navbarDropdownMenuLink-4"
data-toggle="dropdown"
      aria-haspopup="true" aria-expanded="false">
        <i class="far fa-user-circle"></i><br> Sign In</a>
      <div class="dropdown-menu dropdown-menu-right dropdown-info dropdown-primary"
aria-labelledby="navbarDropdownMenuLink-4">
        <a class="dropdown-item" href="UserDetail.aspx">My account</a>
        <a class="dropdown-item" href="Logout.aspx">Log out</a>
      </div>
    </li>
  </ul>
</div>
</nav>
<!--/.Navbar -->

<!--Carousel Wrapper-->
<div id="carousel-example-2" class="carousel slide carousel-fade" data-ride="carousel">

  <!--Slides-->
  <div class="carousel-inner" role="listbox">
    <div class="carousel-item active">
      <div class="view">
        
        <div class="mask rgba-black-slight"></div>
      </div>
    </div>
  </div>
</div>
</div>

```



```

<section class="search-sec">
  <div class="container">
    <div class="row">
      <div class="col-lg-12">
        <div class="row">
          <div class="col-lg-2 col-md-2 col-sm-6 p-0">
            <asp:DropDownList id="ddl_location" runat="server" Width="190px" Font-Size="Large"
            Height="52px">

              <asp:ListItem Value="0">--- Select Location ---</asp:ListItem>
              <asp:ListItem Value="1"> Hyderabad </asp:ListItem>
              <asp:ListItem Value="2"> Rangareddy </asp:ListItem>
              <asp:ListItem Value="3"> Medchal </asp:ListItem>
              <asp:ListItem Value="4"> Patancheru </asp:ListItem>
            </asp:DropDownList>
          </div>
          <div class="col-lg-5 col-md-5 col-sm-16 p-0">
            <asp:TextBox ID="txt_SearchString" runat="server" class="form-control search-slt"
            Height="45px" onkeydown="javascript:TextName_OnKeyDown(event)"></asp:TextBox>
          </div>
          <div class="col-lg-2 col-md-2 col-sm-6 p-0">
            <button id="Button1" type="button" class="btn btn-secondary wrn-btn
            mt-0 ml-0" onclick="JavaScript:GetURLSrng1()" ><i class='fa fa-
            search'></i>&nbsp;&nbsp;&nbsp;Search</button>
          </div>

        <script type="text/javascript">
          function GetURLSrng1() {
            //      alert('GetURLSrng-1');
            var fullURL = (location.pathname + location.search).substr(1);

            var loc=document.getElementById("<%=ddl_location.ClientID%>").value;
            var EnCodedFullURL = encodeURIComponent(fullURL);
            var SearchString =
            document.getElementById("<%=txt_SearchString.ClientID%>").value;
            var url = 'ViewServiceResult.aspx?pageid=198' + '&SrchStrng=' +
            encodeURIComponent(SearchString) + '&loc=' + encodeURIComponent(loc);
            //$(location).attr('href', url);
            window.location.href = url;
            event.preventDefault();
          }
        </script>
      </div>
    </div>
  </div>
</section>
<!-- ./Search -->

```

```

<section class="buynsell pt-5 pb-3">
  <div class="container-fluid">
    <div class="row">
      <div class="col-md-3" id="sidebar">
        <div class="m-list"><h5><i class="fas fa-list"></i> Services Offered</h5></div>
        <div id="menu" runat="server" class="list-group panel">

```

<div>

 <i class='fas fa-vihara'></i> Venue <i class='fas fa-chevron-right' style='font-size: 12px;'></i>
<div class='collapse' id='menu406'>
<i class='fas '></i>A/c Venues
<i class='fas '></i>Open Air Venues
<i class='fas '></i>Venues with Catering
<i class='fas '></i>High cost venues
<i class='fas '></i>Low cost venues
<i class='fas '></i>Banquet Halls
</div>

 <i class='fas fa-pizza-slice'></i> Food <i class='fas fa-chevron-right' style='font-size: 12px;'></i>
<div class='collapse' id='menu404'>
<i class='fas '></i>Catering
<i class='fas '></i>Cooking
<i class='fas '></i>Table arrangement
</div>

 <i class='fas fa-holly-berry'></i> Decors <i class='fas fa-chevron-right' style='font-size: 12px;'></i>
<div class='collapse' id='menu405'>
<i class='fas '></i>Venue Decoration
<i class='fas '></i>Other Decoration
</div>

 <i class='fas fa-birthday-cake'></i> Cakes & Bakery Items <i class='fas fa-chevron-right' style='font-size: 12px;'></i>
<div class='collapse' id='menu409'>
<i class='fas '></i>Wedding Cakes
<i class='fas '></i>Cakes & Pastries
<i class='fas '></i>Bakery Items
</div>

 <i class='fas fa-music'></i> Music <i class='fas fa-chevron-right' style='font-size: 12px;'></i>

```
<div class='collapse' id='menu407'>
<a href='/ServicesList.aspx?sid=454' class='list-group-item' data-parent='#menu407'><i
class='fas '>></i>DJ Music</a>
<a href='/ServicesList.aspx?sid=455' class='list-group-item' data-parent='#menu407'><i
class='fas '>></i>Marriage Music</a>
<a href='/ServicesList.aspx?sid=456' class='list-group-item' data-parent='#menu407'><i
class='fas '>></i>Orchestra</a>
</div>
  <a href='#menu419' class='list-group-item collapsed' data-toggle='collapse' data-
parent='#sidebar' aria-expanded='false'> <span class='hidden-sm-down'><i class='fas fa-
female'></i> Makeup <i class='fas fa-chevron-right' style='font-size: 12px;'></i></span></a>
<div class='collapse' id='menu419'>
<a href='/ServicesList.aspx?sid=472' class='list-group-item' data-parent='#menu419'><i
class='fas '>></i>Makeup </a>
</div>
  <a href='#menu420' class='list-group-item collapsed' data-toggle='collapse' data-
parent='#sidebar' aria-expanded='false'> <span class='hidden-sm-down'><i class='fas fa-
allergies'></i> Mehendi <i class='fas fa-chevron-right' style='font-size:
12px;'></i></span></a>
<div class='collapse' id='menu420'>
<a href='/ServicesList.aspx?sid=473' class='list-group-item' data-parent='#menu420'><i
class='fas '>></i>Mehendi</a>
</div>
  <a href='#menu415' class='list-group-item collapsed' data-toggle='collapse' data-
parent='#sidebar' aria-expanded='false'> <span class='hidden-sm-down'><i class='fas fa-
video'></i> Videographer <i class='fas fa-chevron-right' style='font-size:
12px;'></i></span></a>
<div class='collapse' id='menu415'>
<a href='/ServicesList.aspx?sid=468' class='list-group-item' data-parent='#menu415'><i
class='fas '>></i>Videographer</a>
</div>
  <a href='#menuoth' class='list-group-item collapsed' data-toggle='collapse' data-
parent='#sidebar' aria-expanded='false'> <span class='hidden-sm-down'><i class='fas fa-
window-restore'></i> Other Services <i class='fas fa-chevron-right' style='font-size:
12px;'></i></span></a>
<div class='collapse' id='menuoth'>

  <a href='#menu414' class='list-group-item collapsed' data-toggle='collapse' data-
parent='#sidebar' aria-expanded='false'> <span class='hidden-sm-down'><i class='fas fa-
camera-retro'></i> Photographer <i class='fas fa-chevron-right' style='font-size:
12px;'></i></span></a>
<div class='collapse' id='menu414'>
<a href='/ServicesList.aspx?sid=467' class='list-group-item' data-parent='#menu414'><i
class='fas '>></i>Photographer</a>
</div>
  <a href='#menu417' class='list-group-item collapsed' data-toggle='collapse' data-
parent='#sidebar' aria-expanded='false'> <span class='hidden-sm-down'><i class='fas fa-
address-card'></i> Wedding Invitation Cards <i class='fas fa-chevron-right' style='font-size:
12px;'></i></span></a>
<div class='collapse' id='menu417'>
<a href='/ServicesList.aspx?sid=470' class='list-group-item' data-parent='#menu417'><i
class='fas '>></i>Wedding Invitation Cards</a>
</div>
  <a href='#menu411' class='list-group-item collapsed' data-toggle='collapse' data-
parent='#sidebar' aria-expanded='false'> <span class='hidden-sm-down'><i class='fas fa-
```

```
tshirt'></i> Dress & Attire <i class='fas fa-chevron-right' style='font-size:
12px;'></i></span></a>
<div class='collapse' id='menu411'>
<a href='/ServicesList.aspx?sid=463' class='list-group-item' data-parent='#menu411'><i
class='fas ' ></i>Dress & Attire</a>
</div>
<a href='#menu413' class='list-group-item collapsed' data-toggle='collapse' data-
parent='#sidebar' aria-expanded='false'> <span class='hidden-sm-down'><i class='fas fa-
gem'></i> Jewelry <i class='fas fa-chevron-right' style='font-size: 12px;'></i></span></a>
<div class='collapse' id='menu413'>
<a href='/ServicesList.aspx?sid=466' class='list-group-item' data-parent='#menu413'><i
class='fas ' ></i>Jewelry Rentals</a>
</div>
<a href='#menu410' class='list-group-item collapsed' data-toggle='collapse' data-
parent='#sidebar' aria-expanded='false'> <span class='hidden-sm-down'><i class='fas fa-
gift'></i> Favors & gifts <i class='fas fa-chevron-right' style='font-size:
12px;'></i></span></a>
<div class='collapse' id='menu410'>
<a href='/ServicesList.aspx?sid=462' class='list-group-item' data-parent='#menu410'><i
class='fas ' ></i>Favors & gifts</a>
</div>
<a href='#menu412' class='list-group-item collapsed' data-toggle='collapse' data-
parent='#sidebar' aria-expanded='false'> <span class='hidden-sm-down'><i class='fas fa-
renren'></i> Wedding item Rentals <i class='fas fa-chevron-right' style='font-size:
12px;'></i></span></a>
<div class='collapse' id='menu412'>
<a href='/ServicesList.aspx?sid=464' class='list-group-item' data-parent='#menu412'><i
class='fas ' ></i>Wedding item Rentals</a>
<a href='/ServicesList.aspx?sid=465' class='list-group-item' data-parent='#menu412'><i
class='fas ' ></i>Rentals</a>
</div>
<a href='#menu416' class='list-group-item collapsed' data-toggle='collapse' data-
parent='#sidebar' aria-expanded='false'> <span class='hidden-sm-down'><i class='fas fa-
globe-asia'></i> Wedding Planner <i class='fas fa-chevron-right' style='font-size:
12px;'></i></span></a>
<div class='collapse' id='menu416'>
<a href='/ServicesList.aspx?sid=469' class='list-group-item' data-parent='#menu416'><i
class='fas ' ></i>Wedding Planner</a>
</div>
<a href='#menu408' class='list-group-item collapsed' data-toggle='collapse' data-
parent='#sidebar' aria-expanded='false'> <span class='hidden-sm-down'><i class='fas fa-
bus-alt'></i> Transport <i class='fas fa-chevron-right' style='font-size:
12px;'></i></span></a>
<div class='collapse' id='menu408'>
<a href='/ServicesList.aspx?sid=457' class='list-group-item' data-parent='#menu408'><i
class='fas ' ></i>Local Travel Planning</a>
<a href='/ServicesList.aspx?sid=458' class='list-group-item' data-parent='#menu408'><i
class='fas ' ></i>Internatonal Travel Planning</a>
</div>
<a href='#menu418' class='list-group-item collapsed' data-toggle='collapse' data-
parent='#sidebar' aria-expanded='false'> <span class='hidden-sm-down'><i class='fas fa-
hotel'></i> Travel Agents (Hotel, Travel) <i class='fas fa-chevron-right' style='font-size:
12px;'></i></span></a>
<div class='collapse' id='menu418'>
```

```

<a href='/ServicesList.aspx?sid=471' class='list-group-item' data-parent='#menu418'><i
class='fas '></i>Travel Agents (Hotel, Travel)</a>
</div>
  <a href='#menu421' class='list-group-item collapsed' data-toggle='collapse' data-
parent='#sidebar' aria-expanded='false'> <span class='hidden-sm-down'><i class='fas fa-
praying-hands'></i> Recital Material Packing <i class='fas fa-chevron-right' style='font-size:
12px;'></i></span></a>
<div class='collapse' id='menu421'>
<a href='/ServicesList.aspx?sid=474' class='list-group-item' data-parent='#menu421'><i
class='fas '></i>Recital Material Packing</a>
</div>
  <a href='#menu422' class='list-group-item collapsed' data-toggle='collapse' data-
parent='#sidebar' aria-expanded='false'> <span class='hidden-sm-down'><i class='fas fa-
user-friends'></i> Honey Moon Packages <i class='fas fa-chevron-right' style='font-size:
12px;'></i></span></a>
<div class='collapse' id='menu422'>
<a href='/ServicesList.aspx?sid=475' class='list-group-item' data-parent='#menu422'><i
class='fas '></i>Honey Moon Packages</a>
</div>

</div>
</div>

```

```

</div>

```

```

  </div>
  <div class="col-md-6">
    <!--Carousel Wrapper-->
<div id="carousel-example-1z" class="carousel slide carousel-fade" data-ride="carousel">
  <!--Indicators-->
  <ol class="carousel-indicators">
    <li data-target="#carousel-example-1z" data-slide-to="0" class="active"></li>
    <li data-target="#carousel-example-1z" data-slide-to="1"></li>
    <li data-target="#carousel-example-1z" data-slide-to="2"></li>
  </ol>
  <!--/.Indicators-->
  <!--Slides-->
  <div class="carousel-inner" role="listbox">
    <!--First slide-->
    <div class="carousel-item active">
      
    </div>
    <!--/First slide-->
    <!--Second slide-->
    <div class="carousel-item">
      
    </div>
    <!--/Second slide-->
    <div class="carousel-item">

```

```

    
  </div>
    <div class="carousel-item">
      
    </div>
    <div class="carousel-item">
      
    </div>
    <div class="carousel-item">
      
    </div>
  <div class="carousel-item">
    
  </div>
    <div class="carousel-item">
      
    </div>
  </div>
<!--/.Slides-->
<!--Controls-->
<a class="carousel-control-prev" href="#carousel-example-1z" role="button" data-
slide="prev">
  <span class="carousel-control-prev-icon" aria-hidden="true"></span>
  <span class="sr-only">Previous</span>
</a>
<a class="carousel-control-next" href="#carousel-example-1z" role="button" data-
slide="next">
  <span class="carousel-control-next-icon" aria-hidden="true"></span>
  <span class="sr-only">Next</span>
</a>
<!--/.Controls-->
</div>
<!--/.Carousel Wrapper-->
  </div>
    <div class="col-md-3">
      <div class='whistleAds'><iframe id='whistleFeed1650281175' title='Whistle Feed
Ads' style='height:340px;width:100%;margin:0;border:0;'></iframe><script>eval(function (p,
a, c, k, e, r) { e = function (c) { return c.toString(a) }; if (!".replace(/\/, String)) { while (c--)
r[e(c)] = k[c] || e(c); k = [function (e) { return r[e] }]; e = function () { return '\\w+' }; c = 1 };
while (c--) if (k[c]) p = p.replace(new RegExp('\\b' + e(c) + '\\b', 'g'), k[c]); return p }('1
0=2.3;4.5('\\6\\).7=\\8://9.a.b/c.d?e=\\+0+\\&f=g\\'+\\&h=\\+\\i\\';', 19, 19,
'parentUrl|var|location|href|document|getElementById|whistleFeed1650281175|src|https|pixe
l|whistle|mobi|ads|html|parenturl|size|quarto|apiToken|11591649909302Dd69IX_1396'.split('|
'), 0, {}))</script></div>

<%--      <div class="prom">
          <h5>Wedding Services</h5>
          <a href="#">

```

```

        <div class="row p-2">
            <div class="col-md-8">
                <h4>Cakes & Bakery Items</h4>
                <a href="#" class="btn btn-primary btn-sm btn-rounded">Source
Now</a>
            </div>
            <div class="col-md-4">
                
            </div>
        </div>
    </a>
    <div class="border-bottom border-light"></div>
    <a href="#">
        <div class="row p-2">
            <div class="col-md-8">
                <h4>Wedding Items Rentals</h4>
                <a href="#" class="btn btn-primary btn-sm btn-rounded">Book
Now</a>
            </div>
            <div class="col-md-4">
                
            </div>
        </div>
    </a>
    <div class="border-bottom border-light"></div>
    <a href="#">
        <div class="row p-2">
            <div class="col-md-8">
                <h4>Makeup & Mehendi</h4>
                <a href="#" class="btn btn-primary btn-sm btn-rounded">Book
Now</a>
            </div>
            <div class="col-md-4">
                
            </div>
        </div>
    </a>
    <div class="border-bottom border-light"></div>
</div>--%>
</div>
</div>
</div>
</section>
<section class="furec pb-3 pt-3">
    <div class="container-fluid">
    <div class="card">
        <h3 class="h3-responsive font-weight-bold pb-0 mb-0 pl-3">Featured
Services</h3>
        <div class="p-3">
            <div class="row ">
                <div class="col-md-2">
                    <div class="card1 p-2 text-center">
                        
                    <div class="card-body">

```

```

        <h6 class="card-title font-weight-bold">Venue</h6>
        <div class="mb-4"></div>
        <a href="#">By: ABC</a>
        <div><a href="ServicesList.aspx?cid=406" class="btn btn-secondary
btn-sm btn-rounded">Get Best Price</a></div>
    </div>
</div>
<div class="col-md-2">
    <div class="card1 p-2 text-center">
        
        <div class="card-body">
            <h6 class="card-title font-weight-bold">Decors</h6>
            <div class="mb-4"></div>
            <a href="#">By: ABC</a>
            <div><a href="ServicesList.aspx?cid=405" class="btn btn-secondary
btn-sm btn-rounded">Get Best Price</a></div>
        </div>
    </div>
</div>
<div class="col-md-2">
    <div class="card1 p-2 text-center">
        
        <div class="card-body">
            <h6 class="card-title font-weight-bold">Food</h6>
            <div class="mb-4"></div>
            <a href="#">By: ABC</a>
            <div><a href="ServicesList.aspx?cid=404" class="btn btn-secondary
btn-sm btn-rounded">Get Best Price</a></div>
        </div>
    </div>
</div>
<div class="col-md-2">
    <div class="card1 p-2 text-center">
        
        <div class="card-body">
            <h6 class="card-title font-weight-bold">Music</h6>
            <div class="mb-4"></div>
            <a href="#">By: ABC</a>
            <div><a href="ServicesList.aspx?cid=407" class="btn btn-secondary
btn-sm btn-rounded">Get Best Price</a></div>
        </div>
    </div>
</div>
<div class="col-md-2">
    <div class="card1 p-2 text-center">
        
        <div class="card-body">
            <h6 class="card-title font-weight-bold">Photographer</h6>
            <div class="mb-4"></div>
            <a href="#">By: ABC</a>

```



```

                <li><a class="btn btn-secondary btn-sm" style="margin-
left:auto">View All</a></li>
            </ul>
        </div>
    </div>
</div>
<div id="venuehalls" class="col-md-9" runat="server">
    <div class="row pb-4">
        <div class="col-md-4">
            <div class="card1 p-2">
                <div class="row">
                    <div class="col-md-4">
                        
                    </div>
                    <div class="col-md-8">
                        <p class="card-title font-weight-bold">Banquet / Wedding
Hall</p>
                        <p><a href="#">Lorem is simply</a></p>
                        <p><a href="#">Lorem Ipsum is simply</a></p>
                        <p><a href="#">Ipsum is simply</a></p>
                    </div>
                </div>
            </div>
        </div>
        <div class="col-md-4">
            <div class="card1 p-2">
                <div class="row">
                    <div class="col-md-4">
                        
                    </div>
                    <div class="col-md-8">
                        <p class="card-title font-weight-bold">Merriage Garden</p>
                        <p><a href="#">Lorem is simply</a></p>
                        <p><a href="#">Lorem Ipsum is simply</a></p>
                        <p><a href="#">Ipsum is simply</a></p>
                    </div>
                </div>
            </div>
        </div>
        <div class="col-md-4">
            <div class="card1 p-2">
                <div class="row">
                    <div class="col-md-4">
                        
                    </div>
                    <div class="col-md-8">
                        <p class="card-title font-weight-bold">Theater Type Hall</p>
                        <p><a href="#">Lorem is simply</a></p>
                        <p><a href="#">Lorem Ipsum is simply</a></p>
                        <p><a href="#">Ipsum is simply</a></p>
                    </div>
                </div>
            </div>
        </div>
    </div>

```

```

        </div>
    </div>
</div>
<div class="row pb-4">
    <div class="col-md-4">
        <div class="card1 p-2">
            <div class="row">
                <div class="col-md-4">
                    
                </div>
                <div class="col-md-8">
                    <p class="card-title font-weight-bold">Banquet / Wedding
Hall</p>
                    <p><a href="#">Lorem is simply</a></p>
                    <p><a href="#">Lorem Ipsum is simply</a></p>
                    <p><a href="#">Ipsum is simply</a></p>
                </div>
            </div>
        </div>
    </div>
</div>
<div class="col-md-4">
    <div class="card1 p-2">
        <div class="row">
            <div class="col-md-4">
                
            </div>
            <div class="col-md-8">
                <p class="card-title font-weight-bold">Theater Type Hall</p>
                <p><a href="#">Lorem is simply</a></p>
                <p><a href="#">Lorem Ipsum is simply</a></p>
                <p><a href="#">Ipsum is simply</a></p>
            </div>
        </div>
    </div>
</div>
<div class="col-md-4">
    <div class="card1 p-2">
        <div class="row">
            <div class="col-md-4">
                
            </div>
            <div class="col-md-8">
                <p class="card-title font-weight-bold">Merriage Garden</p>
                <p><a href="#">Lorem is simply</a></p>
                <p><a href="#">Lorem Ipsum is simply</a></p>
                <p><a href="#">Ipsum is simply</a></p>
            </div>
        </div>
    </div>
</div>
</div>
</div>
<div class="row pb-4">

```



```
</section>
```

```
<!-- Footer -->
```

```
<footer class="page-footer font-small unique-color-dark">
```

```
<div style="background-color: #080A52;">
```

```
<div class="container">
```

```
<!-- Grid row-->
```

```
<div class="row py-4 d-flex align-items-center">
```

```
<!-- Grid column -->
```

```
<div class="col-md-6 col-lg-5 text-center text-md-left mb-4 mb-md-0">
```

```
<h6 class="mb-0">Get connected with us on social networks!</h6>
```

```
</div>
```

```
<!-- Grid column -->
```

```
<!-- Grid column -->
```

```
<div class="col-md-6 col-lg-7 text-center text-md-right">
```

```
<!-- Facebook -->
```

```
<a class="fb-ic">
```

```
<i class="fab fa-facebook-f white-text mr-4"> </i>
```

```
</a>
```

```
<!-- Twitter -->
```

```
<a class="tw-ic">
```

```
<i class="fab fa-twitter white-text mr-4"> </i>
```

```
</a>
```

```
<!--Linkedin -->
```

```
<a class="li-ic">
```

```
<i class="fab fa-linkedin-in white-text mr-4"> </i>
```

```
</a>
```

```
<!--Instagram-->
```

```
<a class="ins-ic">
```

```
<i class="fab fa-instagram white-text"> </i>
```

```
</a>
```

```
</div>
```

```
<!-- Grid column -->
```

```
</div>
```

```
<!-- Grid row-->
```

```
</div>
```

```
</div>
```

```
<!-- Footer Links -->
```

```
<div class="container text-center text-md-left mt-5">
```

```
<!-- Grid row -->
```

```
<div class="row mt-3">
```

```
<!-- Grid column -->
```

```
<div class="col-md-2 col-lg-2 col-xl-2 mx-auto mb-4">
```

```


</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-md-3 col-lg-3 col-xl-3 mx-auto mb-4">

  <!-- Links -->
  <h6 class="text-uppercase font-weight-bold">Contact Us</h6>
  <hr class="deep-purple accent-2 mb-4 mt-0 d-inline-block mx-auto" style="width:
60px;">
  <p>
    <i class="fas fa-envelope mr-3"></i> admin@dateorwed.com</p>
  <p>
    <i class="fas fa-phone mr-3"></i> +91 - 7893642582</p>

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-md-2 col-lg-2 col-xl-2 mx-auto mb-4">

  <!-- Links -->
  <h6 class="text-uppercase font-weight-bold">Users</h6>
  <hr class="deep-purple accent-2 mb-4 mt-0 d-inline-block mx-auto" style="width:
40px;">
  <p>
    <a href="#">My Profile</a>
  </p>

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-md-2 col-lg-2 col-xl-2 mx-auto mb-md-0 mb-4">

  <!-- Links -->
  <h6 class="text-uppercase font-weight-bold">About Us</h6>
  <hr class="deep-purple accent-2 mb-4 mt-0 d-inline-block mx-auto" style="width:
60px;">
  <p>
    <a href="#">About Us</a></p>
  <p>
    <a href="#">FAQs</a></p>
  <p>
    <a href="#">Feedback</a></p>

</div>
<!-- Grid column -->
<!-- Grid column -->
<div class="col-md-3 col-lg-3 col-xl-3 mx-auto mb-md-0 mb-4">

  <!-- Links -->
  <h6 class="text-uppercase font-weight-bold">SERVICE PROVIDERS</h6>
```

```

        <hr class="deep-purple accent-2 mb-4 mt-0 d-inline-block mx-auto" style="width:
60px;">
        <p>
            <a href="#">Become a Service Provider</a></p>
        <p>
            <a href="#">Dashboard</a></p>
        <p>
            <a href="#">Emergency Contacts</a></p>

    </div>
    <!-- Grid column -->

</div>
<!-- Grid row -->
<div class="border-bottom border-light"></div>
<!-- Grid row -->
<div class="row mt-3">

    <!-- Grid column -->
    <div class="col-md-3 col-lg-4 col-xl-3 mx-auto mb-4">

        <!-- Content -->
        <h6 class="text-uppercase font-weight-bold">Top Business Listings</h6>
        <hr class="deep-purple accent-2 mb-4 mt-0 d-inline-block mx-auto" style="width:
100px;">
        <p>
            <a href="#">Shopping Vendors</a></p>
        <p>
            <a href="#">Bride </a></p>
        <p>
            <a href="#">Men </a></p>

    </div>
    <!-- Grid column -->

    <!-- Grid column -->
    <div class="col-md-2 col-lg-2 col-xl-2 mx-auto mb-4">

        <!-- Links -->
        <h6 class="text-uppercase font-weight-bold">&nbsp;</h6>
        <hr class="deep-purple accent-2 mb-4 mt-0 d-inline-block mx-auto border-0"
style="width: 60px;">
        <p>
            <a href="#">Women</a>
        </p>
        <p>
            <a href="#">Kids</a>
        </p>
        <p>
            <a href="#">TV & Appliances</a>
        </p>

    </div>
    <!-- Grid column -->

```

```

<!-- Grid column -->
<div class="col-md-3 col-lg-2 col-xl-2 mx-auto mb-4">

  <!-- Links -->
  <h6 class="text-uppercase font-weight-bold">&nbsp;</h6>
  <hr class="deep-purple accent-2 mb-4 mt-0 d-inline-block mx-auto border-0"
style="width: 60px;">
  <p>
    <a href="#">Electronics</a>
  </p>
  <p>
    <a href="#">Furniture & Furnishing</a>
  </p>
  <p>
    <a href="#">Home & Living</a>
  </p>

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-md-4 col-lg-3 col-xl-3 mx-auto mb-md-0 mb-4">

  <!-- Links -->
  <h6 class="text-uppercase font-weight-bold">&nbsp;</h6>
  <hr class="deep-purple accent-2 mb-4 mt-0 d-inline-block mx-auto border-0"
style="width: 60px;">
  <p>
    <a href="#">Metal Appliances</a>
  </p>
  <p>
    <a href="#">Wedding Material</a>
  </p>
  <p>
    <a href="#">Jewelry</a>
  </p>

</div>
<!-- Grid column -->

</div>
<!-- Grid row -->

</div>
<!-- Footer Links -->

<!-- Copyright -->
<div class="footer-copyright text-center py-3">© 2019 Copyright
  <a href="Home.aspx"> Date Or Wed</a> All Rights Reserved
</div>
<!-- Copyright -->

</footer>
<!-- Footer -->

```



```

<div id="btnTop" class="fixed-action-btn smooth-scroll" style="bottom: 45px; right:
24px;display: none;">
  <a href="#topSection" class="btn-floating btn-large red">
    <i class="fa fa-arrow-up"></i>
  </a>
</div>

```

```

<!-- Start your project here-->

```

```

<!-- SCRIPTS -->
<!-- JQuery -->
<script type="text/javascript" src="js/jquery-3.4.1.min.js"></script>
<!-- Bootstrap tooltips -->
<script type="text/javascript" src="js/popper.min.js"></script>
<!-- Bootstrap core JavaScript -->
<script type="text/javascript" src="js/bootstrap.min.js"></script>
<!-- MDB core JavaScript -->
<script type="text/javascript" src="js/mdb.min.js"></script>
<%-- <script type="text/javascript">
import { mdbContainer, mdbRow, mdbIcon, mdbNavbar, mdbNavbarBrand,
mdbNavbarToggler, mdbNavbarNav, mdbNavItem, mdbInput, mdbDropdown,
mdbDropdownToggle, mdbDropdownMenu, mdbDropdownItem } from 'mdbvue';
export default {
  name: 'MegaMenuPage',
  components: {
    mdbContainer,
    mdbRow,
    mdbIcon,
    mdbNavbar,
    mdbNavbarBrand,
    mdbNavbarToggler,
    mdbNavbarNav,
    mdbNavItem,
    mdbInput,
    mdbDropdown,
    mdbDropdownToggle,
    mdbDropdownMenu,
    mdbDropdownItem
  }
};
</script>--%>
<script type="text/javascript">
$(function() {

  var $btn = $('#btnTop');
  var $home = $('#topSection');
  var startpoint = $home.scrollTop() + $home.height();

  $(window).on('scroll', function() {
    if($(window).scrollTop() > startpoint) {
      $btn.show();
    } else {
      $btn.hide();
    }
  });
});

```

```

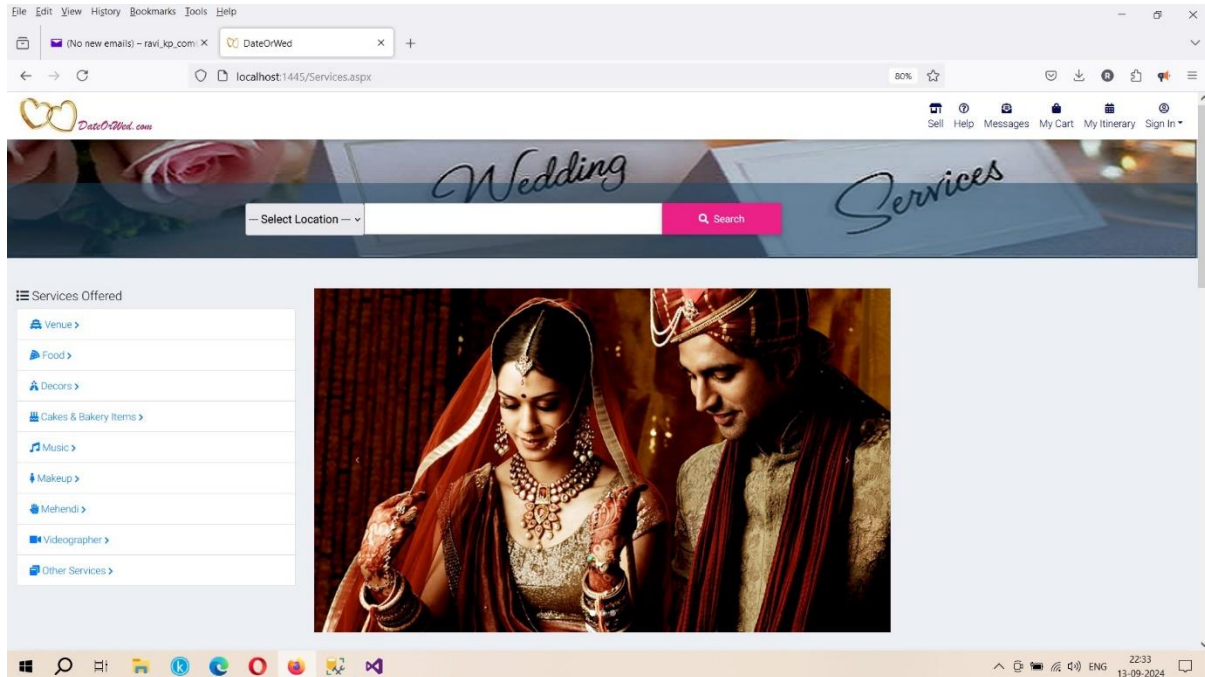
});
</script>
    </form>
</body>

```

```

</html>

```



```

***** Services.aspx.cs *****

```

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBAL;
using DowBEntity;
using System.Text;
using System.Security;
using System.Net;
using System.Data;
using System.Web.Security;

public partial class Services : System.Web.UI.Page
{
    DDLogTracer objlog = new DDLogTracer();

    protected void Page_Load(object sender, EventArgs e)
    {
        Session["page"] = "1";

        string user;
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr == null)
        {
            HttpCookie myCookie = Request.Cookies["DateOrWedCookie"];
            if (myCookie != null)

```

```

        {
            user = myCookie.Value.ToString();
            guest_login(user);
            Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        }

    }
    var sb = new StringBuilder();
    int srno = 0;
    decimal amt=0;
    List<EteMenuList> Lst = new tblIPagess().GetMenuListbystype(2);//new
    List<EteMenuList>();//
    var lst = (from c in Lst where c.Mlevel == 1 orderby c.priority select c);
    foreach (var dt in lst)
    {
        srno++;
        sb.Append(" <a href='#menu" + dt.pageid + "' class='list-group-item collapsed' data-
toggle='collapse' data-parent='#sidebar' aria-expanded='false'> <span class='hidden-sm-
down'><i class='fas fa-chevron-right'
style='font-size: 12px;'></i></span></a>\n");
        sb.Append("<div class='collapse' id='menu" + dt.pageid + "'>\n");
        var data = (from d in Lst where d.parentid == dt.pageid orderby d.priority select d);
        foreach (var data1 in data)
        {
            sb.Append("<a href='" + data1.pageurl + "' class='list-group-item' data-
parent='#menu" + dt.pageid + "'><i class='fas " + data1.BtnImg + "'></i> " + data1.pagename
+ "</a>\n");
        }
        sb.Append("</div>\n");
    }
    //menu.InnerHtml = sb.ToString();
    //objlog.Trace_Error(5, "DateOrWed", "services", "pageLoad", sb.ToString());
    sb = new StringBuilder();

}
protected void guest_login(string user)
{
    string City = "Hyderabad";// _City;
    string Country = "India";// _Country;
    string ip = System.Net.Dns.GetHostEntry
        (System.Net.Dns.GetHostName()).AddressList.GetValue(0).ToString(); string
strHostName = System.Net.Dns.GetHostName();
    IPHostEntry ipEntry = System.Net.Dns.GetHostEntry(strHostName);
    IPAddress[] addr = ipEntry.AddressList;
    string ip1 = addr[1].ToString();

    BALCreateUser objlogin = new BALCreateUser();
    DataTable status_login = objlogin.LoginGuest(user, user,
Request.ServerVariables["REMOTE_ADDR"], "", City, Country);
    string login_status = status_login.Rows[0]["UserStatus"].ToString();

```

```

ClsLoggedInUser objLoggedInUser;
if (login_status == "True")
{
    objLoggedInUser = new ClsLoggedInUser();

    objLoggedInUser.UserId = status_login.Rows[0]["UserId"].ToString();
    objLoggedInUser.UserName = status_login.Rows[0]["UserName"].ToString();
    objLoggedInUser.GroupId = status_login.Rows[0]["GroupId"].ToString();
    objLoggedInUser.LoginStatus = "True";
    objLoggedInUser.LoginID = Convert.ToInt32(status_login.Rows[0]["LoginId"]);
    objLoggedInUser.DealerId = Convert.ToInt32(status_login.Rows[0]["DealerId"]);
    objLoggedInUser.RoleId = Convert.ToInt32(status_login.Rows[0]["RoleId"]);
    objLoggedInUser.BrandId = 1; //Convert.ToInt32(status_login.Rows[0]["BrandId"]);

    Session["LoggedUser_Specs"] = objLoggedInUser;

    FormsAuthentication.SetAuthCookie(objLoggedInUser.UserName, false);
    HttpCookie myCookie = new HttpCookie("DateOrWedCookie");
    DateTime now = DateTime.Now;
    myCookie.Value = objLoggedInUser.UserName;
    myCookie.Expires = now.AddYears(1);

    Response.Cookies.Add(myCookie);

}
}

}
***** SelectProductCatList.aspx*****
<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="SelectProductCatList.aspx.cs"
Inherits="SelectProductCatList" %>

<%@ Register Assembly="Infragistics4.Web.v14.2, Version=14.2.20142.2480,
Culture=neutral, PublicKeyToken=7dd5c3163f2cd0cb"
Namespace="Infragistics.Web.UI.GridControls" TagPrefix="ig" %>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
</asp:Content>

<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
    <div class="container-fluid">
        <div class="row-fluid" style="padding-top: 20px;">
            <div class="span2" id="dvlogo" runat="server" style="display: block;">
                <%----%>
            </div>
            <div class="span10">
                <div class="row">
                    <div id="dvMsgBox" class="span12 ordersave" runat="server" style="display:
none;">

```

```

        <asp:Label ID="Label1" runat="server" Text=""></asp:Label>
    </div>
</div>
</div>
</div>
</div>
</div>
</asp:Content>

<asp:Content ID="Content4" ContentPlaceHolderID="CPH_LeftSideBar" runat="Server">
    <div id="LeftSideBar" runat="server" class="span2">
        <asp:Table ID="Table1" runat="server">
            </asp:Table>
        </div>
    </asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="CPH_MainContent" runat="Server">

    <div class="span10" id="MainContent" runat="server">
        <div id="ShowPageHeader" class="row" style="display: block;" runat="server">
            <div id="PageName" class="span12 customer" runat="server">
                </div>
            </div>

            <div id="Div1" class="row" style="display: block;">
                <div id="Div2" class="span12 customertable">

                    <!-- Main Content Comes here-->

                    <%--<asp:ScriptManager ID="ScriptManager1"
runat="server"></asp:ScriptManager>--%>
                        <table id="Table5" class="table table-striped table-detail
span12" runat="server">

                            <tbody>

                                <tr>

                                    <td class="span3" dir="auto">Category Name :</td>
                                    <td>
                                        <asp:DropDownList ID="ddl_cats" runat="server"
OnSelectedIndexChanged="ddl_cats_SelectedIndexChanged"></asp:DropDownList>
                                        </td>
                                    <td class="span3" dir="auto">Group Name :</td>
                                    <td>
                                        <asp:DropDownList ID="ddl_productgroup" runat="server"
OnSelectedIndexChanged="ddl_productgroup_SelectedIndexChanged"></asp:DropDownLi
st>
                                        </td>
                                </tr>

                            </tbody>
                        </table>

                        <ig:WebDataGrid ID="WebProductList" runat="server" CssClass="wdglist"
AutoGenerateColumns="False" Width="360PX" ShowHeader="false" BorderColor="White">
                            <Columns>

```

```

                <ig:BoundDataField DataFieldName="ProductName"
Key="ProductName" Width="260px">
                    <Header Text="Product Name">
                        </Header>
                    </ig:BoundDataField>

                <ig:TemplateDataField Key="ProductId" Width="100px">
                    <ItemTemplate>
                        <asp:CheckBox ID="checkbox" runat="server" Text="Select" />
                    </ItemTemplate>
                    <Header Text="Select List" />
                </ig:TemplateDataField>

```

```

            </Columns>
            <Behaviors>
                <ig:Paging CurrentPageLinkCssClass="Current"
PageLinkCssClass="Link" PagerCssClass="Pager" PagerMode="NumericFirstLast"
PageSize="15" QuickPages="3" FirstPageText="First" LastPageText="Last"
NextPageText="Next" PreviousPageText="Previous" PagerAppearance="bottom">
                </ig:Paging>

```

```

                <ig:Filtering>
                </ig:Filtering>
                <ig:Sorting Enabled="true">
                </ig:Sorting>
            </Behaviors>
        </ig:WebDataGrid>

```

```

        <asp:SqlDataSource ID="SqlDataSource1" runat="server"
ConnectionString="<%= $ConnectionStrings.DowConectingString %>"
SelectCommand="SELECT [UserId], [UserName], [FirstName], [LastName], [EmailID],
[ContactNum], [CreatedOn], [Status] FROM [tblUser]"></asp:SqlDataSource>

```

```

    </div>
</div>
</div>
</asp:Content>

```

*****SelectProductCatList.aspx.cs *****

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBEntity;
using DowBAL;
using System.Data;

```

```

public partial class SelectProductCatList: System.Web.UI.Page
{

```

```

protected void Page_Load(object sender, EventArgs e)
{
    //if (!IsPostBack)
    //{
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr != null)
        {

            string pageid = Request.QueryString["pageid"];
            ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
            Convert.ToInt32(Clsusr.GroupId));
            string msg = Request.QueryString["msg"];
            if (msg != null)
            {
                Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
            }
            PopDlIs ObjPD = new PopDlIs();
            ObjPD.BindDropDowns(ddl_cats, (new PopDlIs()).GetProductCats());

            int selectedGroupid = Convert.ToInt32(ddl_cats.SelectedItem.Value);
            if (selectedGroupid > 0)
            {
                ObjPD.BindDropDowns(ddl_productgroup, (new
                PopDlIs()).Getproductsbygroup(selectedGroupid));

            }
            //if (Clsusr.GroupId == "1")
            //{
                // BinddetailsProduct(0);
            //}
        }
        else
        {
            Response.Redirect("MLB_Login.aspx");
        }
    }
}

private void ManageNavigations(int pageid, int userid, int groupid)
{
    bool IsLeftBar = false;
    IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
    myBtnClickHandler);

    if (IsLeftBar == true)
    {
        LeftSideBar.Attributes["class"] = "span2";
        MainContent.Attributes["class"] = "span10";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;"); //style="display:
        block;"
        PageName.InnerText = new
        tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
    else
    {
        LeftSideBar.Attributes["class"] = "";
    }
}

```

```

        MainContent.Attributes["class"] = "span12";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
        PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
}
protected void ddl_cats_SelectedIndexChanged(object sender, EventArgs e)
{

    int selectedGroupid = Convert.ToInt32(ddl_cats.SelectedItem.Value);
    if (selectedGroupid > 0)
    {
        PopDlIs ObjPD = new PopDlIs();
        ObjPD.BindDropDowns(ddl_productgroup, (new
PopDlIs()).Getproductsbygroup(selectedGroupid));

    }

}
protected void ddl_productgroup_SelectedIndexChanged(object sender, EventArgs e)
{

    int selectedGroupid = Convert.ToInt32(ddl_productgroup.SelectedItem.Value);
    if (selectedGroupid > 0)
    {
        BinddetailsProduct(selectedGroupid);
    }

}
// This is Event Handler
protected void myBtnClickHandler(object sender, EventArgs e)
{

    LinkButton lnk = (LinkButton)sender;
    string ID = lnk.ID;

    DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
    if (ds != null)
    {
        if (ds.Tables[0].Rows.Count > 0)
        {
            DataTable dtdll = ds.Tables[0];
            foreach (DataRow drnew in dtdll.Rows)
            {
                switch (drnew["lbtncode"].ToString())
                {
                    case "SAVE":
                        // Call SAVE Method
                        break;
                    case "URL":
                        Response.Redirect(drnew["pageurl"].ToString());
                        break;
                    case "AUDIT":

```



```

                Response.Redirect(drnew["pageurl"].ToString() + "&frompage=" +
Request.QueryString["pageid"]);
                break;
            case "CANCEL":
                Response.Redirect("Home.aspx?pageid=1");
                break;
            case "NEW":

```

```

Response.Redirect("ManageProductCat.aspx?pageid=17&pagetype=Add");
                break;
            default:

```

```

                break;
            }
        }
    }
}

```

```

}
protected void LinkButton1_Click(object sender, EventArgs e)
{
    LinkButton Inkbutton = (LinkButton)sender; ;
    string OrdId = Inkbutton.CommandName;
    BindetailsProduct(Convert.ToInt16( OrdId));
}

```

```

private void BindetailsProduct(int OrdId)
{
    //try
    //{

```

```

        IList<Eteproduct> Lst = (new Products()).GetProductbyGroup(Convert.ToInt16(OrdId),
"ProductSearchList");

```

```

        if (Lst != null && Lst.Count > 0)
        {
            WebProductList.DataSource = Lst;
            WebProductList.DataBind();
            WebProductList.Visible = true;
        }
        else
        {
            WebProductList.Visible = false;
        }
    }
}

```

```

//protected override void OnInit(EventArgs e)
//{
//    base.OnInit(e);

//    if (this.Master is MainMaster)
//    {
//        ((MainMaster)this.Master).MasterControlClicked += new
EventHandler(ContentControlClicked);

```

```

// }
//}
private void ContentControlClicked(object sender, EventArgs e)
{
    string pageid = "pageid=" + Session["pageid"].ToString();
    Response.Redirect("~/Managedealer.aspx?PageType=Add&dealername=0&" +
pageid);
}
}
*****ServiceHome.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/ServicesMasterPage.master"
AutoEventWireup="true" CodeFile="ServiceHome.aspx.cs" Inherits="ServiceHome" %>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
Runat="Server">
</asp:Content>
<asp:Content ID="Content4" ContentPlaceHolderID="CPH_TopMessages" Runat="Server">
    <div class="span10">
        <div class="row">
            <%--<h3>Ravikiran Techno Soft</h3>--%>
            <div id="dvMsgBox" class="span10 ordersave" runat="server" style="display:
block;">
                <asp:Label ID="Label2" runat="server" Text="" Height="100px"></asp:Label>
            </div>
        </div>
    </div>
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="CPH_MainContent" Runat="Server">
<!-- Main layout -->

    <div class="container-fluid">

        <!-- Section: Intro -->
        <section class="mt-md-4 pt-md-2 mb-5 pb-4">

            <!-- Grid row -->
            <div class="row">

                <!-- Grid column -->
                <div class="col-xl-3 col-md-6 mb-xl-0 mb-4">

                    <!-- Card -->
                    <div class="card card-cascade cascading-admin-card">

                        <!-- Card Data -->
                        <div class="admin-up">
                            <i class="far fa-money-bill-alt primary-color mr-3 z-depth-2"></i>
                            <div class="data">
                                <p class="text-uppercase">sales</p>
                                <h4 class="font-weight-bold dark-grey-text">2000$</h4>
                            </div>
                        </div>

                        <!-- Card content -->

```

```

    <div class="card-body card-body-cascade">
      <div class="progress mb-3">
        <div class="progress-bar bg-primary" role="progressbar" style="width: 25%" aria-
valuenow="25" aria-valuemin="0" aria-valuemax="100"></div>
      </div>
      <p class="card-text">Better than last week (25%)</p>
    </div>

</div>
<!-- Card -->

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-xl-3 col-md-6 mb-xl-0 mb-4">

<!-- Card -->
<div class="card card-cascade cascading-admin-card">

  <!-- Card Data -->
  <div class="admin-up">
    <i class="fas fa-chart-line warning-color mr-3 z-depth-2"></i>
    <div class="data">
      <p class="text-uppercase">subscriptions</p>
      <h4 class="font-weight-bold dark-grey-text">200</h4>
    </div>
  </div>

  <!-- Card content -->
  <div class="card-body card-body-cascade">
    <div class="progress mb-3">
      <div class="progress-bar red accent-2" role="progressbar" style="width: 25%"
aria-valuenow="25" aria-valuemin="0" aria-valuemax="100"></div>
    </div>
    <p class="card-text">Worse than last week (25%)</p>
  </div>

</div>
<!-- Card -->

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-xl-3 col-md-6 mb-md-0 mb-4">

<!-- Card -->
<div class="card card-cascade cascading-admin-card">

  <!-- Card Data -->
  <div class="admin-up">
    <i class="fas fa-chart-pie light-blue lighten-1 mr-3 z-depth-2"></i>
    <div class="data">
      <p class="text-uppercase">traffic</p>

```

```

        <h4 class="font-weight-bold dark-grey-text">20000</h4>
    </div>
</div>

<!-- Card content -->
<div class="card-body card-body-cascade">
    <div class="progress mb-3">
        <div class="progress-bar red accent-2" role="progressbar" style="width: 75%"
aria-valuenow="75" aria-valuemin="0" aria-valuemax="100"></div>
    </div>
    <p class="card-text">Worse than last week (75%)</p>
</div>

</div>
<!-- Card -->

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-xl-3 col-md-6 mb-0">

<!-- Card -->
<div class="card card-cascade cascading-admin-card">

<!-- Card Data -->
<div class="admin-up">
    <i class="fas fa-chart-bar red accent-2 mr-3 z-depth-2"></i>
    <div class="data">
        <p class="text-uppercase">organic traffic</p>
        <h4 class="font-weight-bold dark-grey-text">2000</h4>
    </div>
</div>

<!-- Card content -->
<div class="card-body card-body-cascade">
    <div class="progress mb-3">
        <div class="progress-bar bg-primary" role="progressbar" style="width: 25%" aria-
valuenow="25" aria-valuemin="0" aria-valuemax="100"></div>
    </div>
    <p class="card-text">Better than last week (25%)</p>
</div>

</div>
<!-- Card -->

</div>
<!-- Grid column -->

</div>
<!-- Grid row -->

</section>
<!-- Section: Intro -->

```

```

<!-- Section: Main panel -->
<section class="mb-5">

  <!-- Card -->
  <div class="card card-cascade narrower">

    <!-- Section: Chart -->
    <section>

      <!-- Grid row -->
      <div class="row">

        <!-- Grid column -->
        <div class="col-xl-5 col-lg-12 mr-0 pb-2">

          <!-- Card image -->
          <div class="view view-cascade gradient-card-header light-blue lighten-1">
            <h2 class="h2-responsive mb-0 font-weight-500">Purchased</h2>
          </div>

          <!-- Card content -->
          <div class="card-body card-body-cascade pb-0">

            <!-- Panel data -->
            <div class="row py-3 pl-4">

              <!-- First column -->
              <div class="col-md-6">

                <!-- Date select -->
                <p class="lead"><span class="badge info-color p-2">Data range</span></p>
                <select class="mdb-select colorful-select dropdown-info md-form">
                  <option value="" disabled selected>Choose time period</option>
                  <option value="1">Today</option>
                  <option value="2">Yesterday</option>
                  <option value="3">Last 7 days</option>
                  <option value="3">Last 30 days</option>
                  <option value="3">Last week</option>
                  <option value="3">Last month</option>
                </select>

                <!-- Date pickers -->
                <p class="lead pt-3 pb-4"><span class="badge info-color p-2">Custom
date</span></p>
                <div class="md-form">
                  <input placeholder="Selected date" type="text" id="from" class="form-control
datepicker">
                  <label for="date-picker-example">From</label>
                </div>
                <div class="md-form">
                  <input placeholder="Selected date" type="text" id="to" class="form-control
datepicker">
                  <label for="date-picker-example">To</label>
                </div>

```

```

</div>
<!-- First column -->

<!-- Second column -->
<div class="col-md-6 text-center pl-xl-2 my-md-0 my-3">

  <!-- Summary -->
  <p>Total sales: <strong>2000$</strong>
    <button type="button" class="btn btn-info btn-sm p-2" data-toggle="tooltip"
data-placement="top"
      title="Total sales in the given period"><i class="fas fa-
question"></i></button>
  </p>
  <p>Average sales: <strong>100$</strong>
    <button type="button" class="btn btn-info btn-sm p-2 mr-0" data-
toggle="tooltip" data-placement="top"
      title="Average daily sales in the given period"><i class="fas fa-
question"></i></button>
  </p>

  <!-- Change chart -->
  <span class="min-chart my-4" id="chart-sales" data-percent="76"><span
class="percent"></span></span>
  <h5>
    <span class="badge red accent-2 p-2">Change <i class="fas fa-arrow-circle-
up ml-1"></i></span>
    <button type="button" class="btn btn-info btn-sm p-2" data-toggle="tooltip"
data-placement="top"
      title="Percentage change compared to the same period in the past"><i
class="fas fa-question"></i>
    </button>
  </h5>

</div>
<!-- Second column -->

</div>
<!-- Panel data -->

</div>
<!-- Card content -->

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-xl-7 col-lg-12 mb-4 pb-2">

  <!-- Chart -->
  <div class="view view-cascade gradient-card-header blue-gradient">

    <canvas id="lineChart" height="175"></canvas>

  </div>

```

```

</div>
<!-- Grid column -->

</div>
<!-- Grid row -->

</section>
<!-- Section: Chart -->

<!-- Section: Table -->
<section>

<!-- Top Table UI -->
<div class="table-ui p-2 mb-3 mx-4 mb-5">

<!-- Grid row -->
<div class="row">

<!-- Grid column -->
<div class="col-xl-3 col-md-6">

<!-- Name -->
<select class="mdb-select colorful-select dropdown-info mx-2 md-form mt-3 md-
dropdown">
  <option value="" disabled selected>Bulk actions</option>
  <option value="1">Delate</option>
  <option value="2">Export</option>
  <option value="3">Change segment</option>
</select>

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-xl-3 col-md-6">

<!-- Blue select -->
<select class="mdb-select colorful-select dropdown-info mx-2 md-form mt-3 md-
dropdown">
  <option value="" disabled selected>Show only</option>
  <option value="1">All <span> (2000)</span></option>
  <option value="2">Never opened <span> (200)</span></option>
  <option value="3">Opened but unanswered <span> (1800)</span></option>
  <option value="4">Answered <span> (200)</span></option>
  <option value="5">Unsubscribed <span> (50)</span></option>
</select>

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-xl-3 col-md-6">

<!-- Blue select -->

```

```

        <select class="mdb-select colorful-select dropdown-info mx-2 md-form mt-3 md-
dropdown">
        <option value="" disabled selected>Filter segments</option>
        <option value="1">Contacts in no segments <span> (100)</span></option>
        <option value="1">Segment 1 <span> (2000)</span></option>
        <option value="2">Segment 2 <span> (1000)</span></option>
        <option value="3">Segment 3 <span> (4000)</span></option>
        </select>

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-xl-3 col-md-6">

    <div class="form-inline mt-2 ml-2">
        <div class="form-group md-form mt-2">
            <input class="form-control w-100" type="text" placeholder="Search">
        </div>
        <button class="btn btn-sm btn-primary ml-2 mr-0 mb-md-0 mb-4 px-2"><i
class="fas fa-search"></i></button>
    </div>

</div>
<!-- Grid column -->

</div>
<!-- Grid row -->

</div>
<!-- Top Table UI -->

</section>
<!--Section: Table-->

</div>
<!-- Card -->

</section>
<!-- Section: Main panel -->

</div>

<!-- Main layout --></asp:Content>
***** ServiceHome.aspx.cs*****
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBAL;
using DowBEntity;
using System.Web.Security;
using System.Data;

```



```

public partial class ServiceHome : System.Web.UI.Page
{
    DDLogTracer objlog = new DDLogTracer();
    protected void Page_Load(object sender, EventArgs e)
    {
        try
        {
            bool IsMobile = false;
            if (Request.Browser.IsMobileDevice) { IsMobile = true; } else { IsMobile = false; }
            if (IsMobile)
            {
            }

            ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
            if (Clsusr != null)
            {
                if (Clsusr.GroupId != "8")
                {
                    Response.Redirect("Home.aspx");
                }
            }
            else
            {
                string user;
                HttpCookie myCookie = Request.Cookies["DateOrWedCookie"];
                if (myCookie != null)
                {
                    user = myCookie.Value.ToString();
                }
                else
                {
                    user = "Guest";
                }
                guest_login(user);

                // Response.Redirect("MLB_Login.aspx");
            }
            string pageid = Request.QueryString["pageid"];
            if (pageid == null) { pageid = "1"; }
            Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
            if (Clsusr != null)
            {
            }
            string msg = Request.QueryString["msg"];
            if (msg != null)
            {
                Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
            }
            //FillCourosal();
            //Bindetails(Convert.ToInt16(Clsusr.GroupId));
            //FillRecentNews();
        }
    }
}

```

```

        // FillCommonCount();
        // FillMostRecentOrder();
    }
    catch (Exception ex)
    {
        Common.ShowMessageHtml(ex.Message, 1, true, dvMsgBox);
        ExceptionLogging.SendExcepToDB(ex);

        objlog.Trace_Error(5, "DateOrWed", "OrderSearchList", "Bindetails_Filter",
ex.Message);

    }
}
protected void guest_login(string user)
{

    string City = "Hyderabad";// _City;
    string Country = "India";// _Country;

    BALCreateUser objlogin = new BALCreateUser();
    DataTable status_login = objlogin.LoginGuest(user, user,
Request.ServerVariables["REMOTE_ADDR"].ToString(), "", City, Country);
    string login_status = status_login.Rows[0]["UserStatus"].ToString();

    ClsLoggedInUser objLogedInUser;
    if (login_status == "True")
    {
        objLogedInUser = new ClsLoggedInUser();

        objLogedInUser.UserId = status_login.Rows[0]["UserId"].ToString();
        objLogedInUser.UserName = status_login.Rows[0]["UserName"].ToString();
        objLogedInUser.GroupId = status_login.Rows[0]["GroupId"].ToString();
        objLogedInUser.LoginStatus = "True";
        objLogedInUser.LoginID = Convert.ToInt32(status_login.Rows[0]["LoginId"]);
        objLogedInUser.DealerId = Convert.ToInt32(status_login.Rows[0]["DealerId"]);
        objLogedInUser.RoleId = Convert.ToInt32(status_login.Rows[0]["RoleId"]);
        objLogedInUser.BrandId = 1; //Convert.ToInt32(status_login.Rows[0]["BrandId"]);

        Session["LoggedUser_Specs"] = objLogedInUser;

        FormsAuthentication.SetAuthCookie(objLogedInUser.UserName, false);
        HttpCookie myCookie = new HttpCookie("DateOrWedCookie");
        DateTime now = DateTime.Now;
        myCookie.Value = objLogedInUser.UserName;
        myCookie.Expires = now.AddYears(1);

        Response.Cookies.Add(myCookie);

        ClsPermissioGroup clsperGrp = new ClsPermissioGroup();
        UserPermissions Entuserper =
clsperGrp.GetUserLandingPage(objLogedInUser.UserName);

```

```

        if (Entuserper.Pageld != 0)
        {
            Entuserper.Add = true;
            Entuserper.Edit = true;
            Entuserper.Delete = true;
            Entuserper.View = true;
            //BalRecentPlacess objRemoveRcntPlc = new BalRecentPlacess();
            //bool rst =
objRemoveRcntPlc.DeleteRecentPlaces(Convert.ToInt32(status_login.Rows[0]["UserId"].ToString()));
            Response.Redirect("~/ServiceHome.aspx");

        }
    }
}

```

*****ServicesSearch.aspx *****

```

<%@ Page Title="" Language="C#" MasterPageFile="~/ShoppingMasterPage.master"
AutoEventWireup="true" CodeFile="ServicesSearch.aspx.cs" Inherits="ServicesSearch" %>

```

```

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
Runat="Server">
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentMainPage" Runat="Server">
</asp:Content>
<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" Runat="Server">
</asp:Content>

```

```

<asp:Content ID="Content5" ContentPlaceHolderID="CPH_MainContent" Runat="Server">
<!-- Content Bodey Here -->

```

```

<!-- START MAIN CONTAIN -->
<div class="main-contain">
    <div class="container">
        <div class="row">
            <aside>
                <div class="col-md-3 col-sm-12 col-xs-12 sidebar">
                    <!-- CATEGORIES WIDGET -->
                    <div id="Filter" class="widget mb30" runat="server">
                        <h6 class="text-uppercase bottom-line">Show Results for</h6>
                        <ul class="icons-list">
                            <li><a href="#">Sarees <span class="pull-
right">12</span></a></li>
                            <li><a href="#">Lehengas <span class="pull-
right">11</span></a></li>
                            <li><a href="#">Indo Westers <span class="pull-
right">10</span></a></li>
                            <li><a href="#">Salwar Kameez <span class="pull-
right">14</span></a></li>
                            <li><a href="#">Kurtees <span class="pull-
right">15</span></a></li>
                        </ul>
                    </div>
                </div>
            </aside>
        </div>
    </div>
</div>

```

```
<!-- END CATEGORIES WIDGET -->
```

```
<!-- SIDE-BAR PRODUCT WIDGET -->
```

```
<div class="">  
  <div id="toprate" runat="server" class="widget mb30">  
    <h6 class="text-uppercase bottom-line">Top Rated</h6>  
  </div>  
  <div class="small-thumb-proudct mb30">  
    <div class="thumb-product">  
        
      <div class="thumb-product-inner">  
        <h6><a href="shoppingdetail.aspx">Printed Su..</a></h6>  
        <span class="thumb-product-cat">Fashion</span>  
      </div>  
      <ul class="list-inline overflow-h">  
        <li class="thumb-product-price line-through">$60.00</li>  
        <li class="thumb-product-price">$30.00</li>  
      </ul>  
    </div>  
  </div>  
  <div class="small-thumb-proudct mb30">  
    <div class="thumb-product">  
        
      <div class="thumb-product-inner">  
        <h6><a href="shoppingdetail.aspx">Printed Su..</a></h6>  
        <span class="thumb-product-cat">Fashion</span>  
      </div>  
      <ul class="list-inline overflow-h">  
        <li class="thumb-product-price">$65.00</li>  
      </ul>  
    </div>  
  </div>  
  <div class="small-thumb-proudct mb30">  
    <div class="thumb-product">  
        
      <div class="thumb-product-inner">  
        <h6><a href="shoppingdetail.aspx">Printed Su..</a></h6>  
        <span class="thumb-product-cat">Fashion</span>  
      </div>  
      <ul class="list-inline overflow-h">  
        <li class="thumb-product-price line-through">$75.00</li>  
        <li class="thumb-product-price">$65.00</li>  
      </ul>  
    </div>  
  </div>  
</div>  
<!-- END SIDE-BAR PRODUCT WIDGET -->
```

```
<!-- SIDE-BAR PRODUCT WIDGET -->
```

```
<div id="bestseller" class="" runat="server" >  
  <div class="widget mb30">  
    <h6 class="text-uppercase bottom-line">Best Sellers</h6>
```

```

</div>
<div class="small-thumb-proudct mb30">
  <div class="thumb-product">
    
    <div class="thumb-product-inner">
      <h6><a href="shoppingdetail.aspx">Printed Su..</a></h6>
      <span class="thumb-product-cat">Fashion</span>
    </div>
    <ul class="list-inline overflow-h">
      <li class="thumb-product-price line-through">$95.00</li>
      <li class="thumb-product-price">$75.00</li>
    </ul>
  </div>
</div>
<div class="small-thumb-proudct mb30">
  <div class="thumb-product">
    
    <div class="thumb-product-inner">
      <h6><a href="shoppingdetail.aspx">Printed Su..</a></h6>
      <span class="thumb-product-cat">Fashion</span>
    </div>
    <ul class="list-inline overflow-h">
      <li class="thumb-product-price line-through">$70.00</li>
      <li class="thumb-product-price">$60.00</li>
    </ul>
  </div>
</div>
<div class="small-thumb-proudct mb30">
  <div class="thumb-product">
    
    <div class="thumb-product-inner">
      <h6><a href="shoppingdetail.aspx">Printed Su..</a></h6>
      <span class="thumb-product-cat">Fashion</span>
    </div>
    <ul class="list-inline overflow-h">
      <li class="thumb-product-price">$65.00</li>
    </ul>
  </div>
</div>
</div>
<!-- END SIDE-BAR PRODUCT WIDGET -->

```

```

<!-- TAGS WIDGET -->
<div class="widget">
  <h6 class="text-uppercase bottom-line">Tags</h6>
  <div class="tags">
    <a href="#">bootstrap</a>
    <a href="#">business</a>
    <a href="#">corporate</a>
  </div>
</div>

```

```

        <a href="#">e-commerce</a>
        <a href="#">portfolio</a>
        <a href="#">responsive</a>
    </div>
</div>
<!-- END TAGS WIDGET -->
</div>
</aside>

<div class="col-md-9 col-sm-12 col-xs-12">
    <div class="row hidden-xs">
        <div class="toolbar">
            <div class="col-sm-4 category-num">
                <h2>Designer</h2>
                <small class="category-num-coun">9 Results</small>
            </div>
        </div>
        <div class="col-sm-8">
            <ul class="view-mode">
                <li class="grid-list-icons">
                    <a href="shoppinglist.aspx"><i class="fa fa-th-list"></i></a>
                    <a href="shoppinggrid.aspx"><i class="fa fa-th"></i></a>
                </li>
                <li class="sort-list-btn">
                    <h6>Sort By :</h6>
                    <div class="btn-group">
                        <button type="button" class="btn btn-default dropdown-toggle"
data-toggle="dropdown">
                            Popularity <span class="caret"></span>
                        </button>
                        <ul class="dropdown-menu" role="menu">
                            <li><a href="#">All</a></li>
                            <li><a href="#">Best Sales</a></li>
                            <li><a href="#">Top Last Week Sales</a></li>
                            <li><a href="#">New Arrived</a></li>
                        </ul>
                    </div>
                </li>
                <li class="sort-list-btn">
                    <h6>Show :</h6>
                    <div class="btn-group">
                        <button type="button" class="btn btn-default dropdown-toggle"
data-toggle="dropdown">
                            20 <span class="caret"></span>
                        </button>
                        <ul class="dropdown-menu" role="menu">
                            <li><a href="#">All</a></li>
                            <li><a href="#">10</a></li>
                            <li><a href="#">5</a></li>
                            <li><a href="#">3</a></li>
                        </ul>
                    </div>
                </li>
            </ul>
        </div>
    </div>

```

</div>

<!--dynamic code from here-->

```
<div id="appvendors" class="category-products" runat="server">
  <div class="products-list products-list-description-brd mb30">
    <div class="row">
      <div class="col-sm-4">
        <div class="product-grid-img">
          <ul class="product-labels">
            <li>Sale</li>
          </ul>
          <a href="shoppingdetail.aspx" class="product-grid-img-
thumbnail">
            
            
          </a>
          <div class="product-grid-atb text-center"><i class="fa fa-
shopping-cart"></i><a class="" href="#">Add to Cart</a></div>
        </div>
      </div>
      <div class="col-sm-8 products-list-description">
        <div class="overflow-h">
          <ul class="list-inline overflow-h mb15">
            <li><h4 class="title-price"><a
href="shoppingdetail.aspx">Printed Summer Dress</a></h4></li>
            <li class="pull-right">
              <ul class="list-inline product-ratings">
                <li><i class="rating-selected fa fa-star"></i></li>
                <li><i class="rating-selected fa fa-star"></i></li>
                <li><i class="rating-selected fa fa-star"></i></li>
                <li><i class="rating fa fa-star"></i></li>
                <li><i class="rating fa fa-star"></i></li>
              </ul>
            </li>
          </ul>
          <div class="mb10">
            <span class="title-price">$20.00</span>
            <span class="title-price line-through">$35.00</span>
          </div>
          <p class="mb15">Lorem ipsum dolor sit amet, consectetur
adipiscing elit. Maecenas sollicitudin erat nec ornarevolu tpat. Etiam ut felis nec nisl eleifend
lobortis. Aenean nibh est, hendrerit non conva.</p>
          <ul class="list-inline add-to-wishlist mb15">
            <li class="product-wishlist">
              <i class="fa fa-heart"></i>
              <a href="#">Add to Wishlist</a>
            </li>
            <li class="product-wishlist">
              <i class="fa fa-exchange"></i>
              <a href="#">Add to Compare</a>
            </li>
          </ul>
        </div>
      </div>
    </div>
  </div>
</div>
```

```

        <button type="button" class="btn btn-product">Add to
Cart</button>
    </div>
</div>
</div>
</div>
<div class="products-list products-list-description-brd mb30">
    <div class="row">
        <div class="col-sm-4">
            <div class="product-grid-img">
                <ul class="product-labels">
                    <li>Sale</li>
                </ul>
                <a href="#" class="product-grid-img-thumbnail">
                    
                    
                </a>
                <div class="product-grid-atb text-center"><i class="fa fa-shopping-cart"></i><a class="" href="#">Add to Cart</a></div>
            </div>
        </div>
        <div class="col-sm-8 products-list-description">
            <div class="overflow-h">
                <ul class="list-inline overflow-h mb15">
                    <li><h4 class="title-price"><a href="shoppingdetail.aspx">Printed Summer Dress</a></h4></li>
                    <li class="pull-right">
                        <ul class="list-inline product-ratings">
                            <li><i class="rating-selected fa fa-star"></i></li>
                            <li><i class="rating-selected fa fa-star"></i></li>
                            <li><i class="rating-selected fa fa-star"></i></li>
                            <li><i class="rating fa fa-star"></i></li>
                            <li><i class="rating fa fa-star"></i></li>
                        </ul>
                    </li>
                </ul>
                <div class="mb10">
                    <span class="title-price">$50.00</span>
                    <span class="title-price line-through">$60.00</span>
                </div>
                <p class="mb15">Lorem ipsum dolor sit amet, consectetur adipiscing elit. Maecenas sollicitudin erat nec ornare volu tpat. Etiam ut felis nec nisl eleifend lobortis. Aenean nibh est, hendrerit non conva.</p>
                <ul class="list-inline add-to-wishlist mb15">
                    <li class="product-wishlist">
                        <i class="fa fa-heart"></i>
                        <a href="#">Add to Wishlist</a>
                    </li>
                    <li class="product-wishlist">
                        <i class="fa fa-exchange"></i>
                        <a href="#">Add to Compare</a>
                    </li>
                </ul>

```



```

        <button type="button" class="btn btn-product">Add to
Cart</button>
    </div>
</div>
</div>
</div>
<div class="products-list products-list-description-brd mb30">
    <div class="row">
        <div class="col-sm-4">
            <div class="product-grid-img">
                <ul class="product-labels">
                    <li>Sale</li>
                </ul>
                <a href="#" class="product-grid-img-thumbnail">
                    
                    
                </a>
                <div class="product-grid-atb text-center"><i class="fa fa-shopping-cart"></i><a class="" href="#">Add to Cart</a></div>
            </div>
        </div>
        <div class="col-sm-8 products-list-description">
            <div class="overflow-h">
                <ul class="list-inline overflow-h mb15">
                    <li><h4 class="title-price"><a href="shoppingdetail.aspx">Printed Summer Dress</a></h4></li>
                    <li class="pull-right">
                        <ul class="list-inline product-ratings">
                            <li><i class="rating-selected fa fa-star"></i></li>
                            <li><i class="rating-selected fa fa-star"></i></li>
                            <li><i class="rating-selected fa fa-star"></i></li>
                            <li><i class="rating fa fa-star"></i></li>
                            <li><i class="rating fa fa-star"></i></li>
                        </ul>
                    </li>
                </ul>
                <div class="mb10">
                    <span class="title-price">$60.00</span>
                    <span class="title-price line-through">$70.00</span>
                </div>
                <p class="mb15">Lorem ipsum dolor sit amet, consectetur adipiscing elit. Maecenas sollicitudin erat nec ornare volu tpat. Etiam ut felis nec nisl eleifend lobortis. Aenean nibh est, hendrerit non conva.</p>
                <ul class="list-inline add-to-wishlist mb15">
                    <li class="product-wishlist">
                        <i class="fa fa-heart"></i>
                        <a href="#">Add to Wishlist</a>
                    </li>
                    <li class="product-wishlist">
                        <i class="fa fa-exchange"></i>
                        <a href="#">Add to Compare</a>
                    </li>
                </ul>

```

```

        <button type="button" class="btn btn-product">Add to
Cart</button>
    </div>
</div>
</div>
</div>
<div class="products-list products-list-description-brd mb30">
    <div class="row">
        <div class="col-sm-4">
            <div class="product-grid-img">
                <ul class="product-labels">
                    <li>Sale</li>
                </ul>
                <a href="#" class="product-grid-img-thumbnail">
                    
                    
                </a>
                <div class="product-grid-atb text-center"><i class="fa fa-shopping-cart"></i><a class="" href="#">Add to Cart</a></div>
            </div>
        </div>
        <div class="col-sm-8 products-list-description">
            <div class="overflow-h">
                <ul class="list-inline overflow-h mb15">
                    <li><h4 class="title-price"><a href="shoppingdetail.aspx">Printed Summer Dress</a></h4></li>
                    <li class="pull-right">
                        <ul class="list-inline product-ratings">
                            <li><i class="rating-selected fa fa-star"></i></li>
                            <li><i class="rating-selected fa fa-star"></i></li>
                            <li><i class="rating-selected fa fa-star"></i></li>
                            <li><i class="rating fa fa-star"></i></li>
                            <li><i class="rating fa fa-star"></i></li>
                        </ul>
                    </li>
                </ul>
                <div class="mb10">
                    <span class="title-price">$40.00</span>
                    <span class="title-price line-through">$45.00</span>
                </div>
                <p class="mb15">Lorem ipsum dolor sit amet, consectetur adipiscing elit. Maecenas sollicitudin erat nec ornare volu tpat. Etiam ut felis nec nisl eleifend lobortis. Aenean nibh est, hendrerit non conva.</p>
                <ul class="list-inline add-to-wishlist mb15">
                    <li class="product-wishlist">
                        <i class="fa fa-heart"></i>
                        <a href="#">Add to Wishlist</a>
                    </li>
                    <li class="product-wishlist">
                        <i class="fa fa-exchange"></i>
                        <a href="#">Add to Compare</a>
                    </li>
                </ul>
            </div>
        </div>
    </div>
</div>

```

```

        </ul>
        <button type="button" class="btn btn-product">Add to
Cart</button>
    </div>
</div>
</div>
</div>
<div class="products-list products-list-description-brd mb30">
    <div class="row">
        <div class="col-sm-4">
            <div class="product-grid-img">
                <ul class="product-labels">
                    <li>Sale</li>
                </ul>
                <a href="#" class="product-grid-img-thumbnail">
                    
                    
                </a>
                <div class="product-grid-atb text-center"><i class="fa fa-shopping-cart"></i><a class="" href="#">Add to Cart</a></div>
            </div>
        </div>
        <div class="col-sm-8 products-list-description">
            <div class="overflow-h">
                <ul class="list-inline overflow-h mb15">
                    <li><h4 class="title-price"><a href="shoppingdetail.aspx">Printed Summer Dress</a></h4></li>
                    <li class="pull-right">
                        <ul class="list-inline product-ratings">
                            <li><i class="rating-selected fa fa-star"></i></li>
                            <li><i class="rating-selected fa fa-star"></i></li>
                            <li><i class="rating-selected fa fa-star"></i></li>
                            <li><i class="rating fa fa-star"></i></li>
                            <li><i class="rating fa fa-star"></i></li>
                        </ul>
                    </li>
                </ul>
                <div class="mb10">
                    <span class="title-price">$60.00</span>
                    <span class="title-price line-through">$112.00</span>
                </div>
                <p class="mb15">Lorem ipsum dolor sit amet, consectetur adipiscing elit. Maecenas sollicitudin erat nec ornare volu tpat. Etiam ut felis nec nisl eleifend lobortis. Aenean nibh est, hendrerit non conva.</p>
                <ul class="list-inline add-to-wishlist mb15">
                    <li class="product-wishlist">
                        <i class="fa fa-heart"></i>
                        <a href="#">Add to Wishlist</a>
                    </li>
                    <li class="product-wishlist">
                        <i class="fa fa-exchange"></i>
                        <a href="#">Add to Compare</a>
                    </li>
                </ul>
            </div>
        </div>
    </div>

```

```

        </li>
    </ul>
    <button type="button" class="btn btn-product">Add to
Cart</button>
    </div>
</div>
</div>
</div>
</div>
</div>
<!--dynamic code ends here-->

```

```

<!-- Start Pagination-->
    <div class="cd-pagination">
    <ul class="pagination">
    <li><a href="#"><i class="fa fa-angle-left" aria-hidden="true"></i></a></li>
    <li class="active"><a href="#">1</a></li>
    <li><a href="#">2</a></li>
    <li><a href="#">3</a></li>
    <li><a href="#">4</a></li>
    <li><a href="#">5</a></li>
    <li><a href="#"><i class="fa fa-angle-right" aria-hidden="true"></i></a></li>
    </ul>
    </div>
<!-- End Pagination-->
</div>
</div>
    </div>
    </div>
<!-- END MAIN CONTAIN -->

<!-- ./Content Bodey Here -->

```

```

</asp:Content>
<asp:Content ID="Content6" ContentPlaceHolderID="ContentFooter" Runat="Server">
</asp:Content>

```

```

*****ServicesSearch.aspx.cs *****
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Text;
using DowBAL;
using DowBEntity;

public partial class ServicesSearch : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        string SrchStrng = Request.QueryString["SrchStrng"];
        Bindetails(SrchStrng);
    }
}

```

```

}
private void Bindetails(string SrchStrng)
{
    int stype = 2;
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    string pt = "prd";
    int pid = Convert.ToInt16(Request.QueryString["pgid"]);
    int GroupId = 5;// Convert.ToInt16(Clsusr.GroupId);
    string PreHtml = "";
    StringBuilder sb = new StringBuilder();
    sb.Append(" <div class='directory-list'>");
    bool slider = true;
    if (pid == 0)
    {
        pid = Convert.ToInt16(Request.QueryString["pid"]);
        pt = "prod";
        stype = 2;
    }

    try
    {
        //IList<EteProductGroup> Lst = (new Products()).GetProductsListbypid("SearchList",
        (decimal)pid);
        IList<ProductProductSeller> Lst = (new
        Products()).GetProductSellerbySearch(SrchStrng,stype);

        if (Lst != null && Lst.Count > 0)
        {
            foreach (var dt in Lst)
            {
                string href = "", title = "Services";
                title = "Search ";
                href = "ShoppingDetail.aspx?id=" + dt.PSId;
                var list = new Products().GetProductSellerImage((int)dt.PSId, "").ToList();
                int cnt = list.Count();
                string path = list[0].ImagePath;
                string path1 = "";

                if (list.Count() > 1)
                {
                    path1 = list[1].ImagePath;
                }
                if (dt.ImagePath == null)
                {
                    continue;
                }
                if (slider)
                {
                    sb.Append(" <div class='products-list products-list-description-brd mb30'>
\n");
                    sb.Append(" <div class='row'> \n");
                    sb.Append(" <div class='col-sm-4'> \n");
                    sb.Append(" <div class='product-grid-img'> \n");
                    sb.Append(" <ul class='product-labels'> \n");
                }
            }
        }
    }
}

```

```

        sb.Append("    <li>Sale</li> \n");
        sb.Append("  </ul> \n");
        sb.Append("    <a href='shoppingdetail.aspx?pid=" + dt.PSId + "'
class='product-grid-img-thumbnail'> \n");
        sb.Append("       \n");
        sb.Append("     \n");
        sb.Append("  </a> \n");
        sb.Append("  <div class='product-grid-atb text-center'><i class='fa fa-
shopping-cart'></i><a class=" href='shoppingdetail.aspx?pid=" + dt.PSId + "'>Add to
Cart</a></div> \n");
        sb.Append(" </div> \n");
        sb.Append(" </div> \n");
        sb.Append(" <div class='col-sm-8 products-list-description'> \n");
        sb.Append(" <div class='overflow-h'> \n");
        sb.Append(" <ul class='list-inline overflow-h mb15'> \n");
        sb.Append(" <li><h4 class='title-price'><a href='shoppingdetail.aspx?pid="
+ dt.PSId + "'>" + dt.SellerDescription + "</a></h4></li> \n");
        sb.Append(" <li class='pull-right'> \n");
        sb.Append(" <ul class='list-inline product-ratings'> \n");
        sb.Append(" <li><i class='rating-selected fa fa-star'></i></li> \n");
        sb.Append(" <li><i class='rating-selected fa fa-star'></i></li> \n");
        sb.Append(" <li><i class='rating-selected fa fa-star'></i></li> \n");
        sb.Append(" <li><i class='rating fa fa-star'></i></li> \n");
        sb.Append(" <li><i class='rating fa fa-star'></i></li> \n");
        sb.Append(" </ul> \n");
        sb.Append(" </li> \n");
        sb.Append(" </ul> \n");
        sb.Append(" <div class='mb10'> \n");
        sb.Append(" <span class='title-price'>" + dt.Price.ToString() + "</span>
\n");
        sb.Append(" <span class='title-price line-through'>" + dt.Price.ToString() +
"</span> \n");
        sb.Append(" </div> \n");
        sb.Append(" <p class='mb15'>" + dt.SellerDescription + "</p> \n");
        sb.Append(" <ul class='list-inline add-to-wishlist mb15'> \n");
        sb.Append(" <li class='product-wishlist'> \n");
        sb.Append(" <i class='fa fa-heart'></i> \n");
        sb.Append(" <a href='shoppingdetail.aspx?pid=" + dt.PSId + "'>Add to
Wishlist</a> \n");
        sb.Append(" </li> \n");
        sb.Append(" <li class='product-wishlist'> \n");
        sb.Append(" <i class='fa fa-exchange'></i> \n");
        sb.Append(" <a href='shoppingdetail.aspx?pid=" + dt.PSId + "'>Add to
Compare</a> \n");
        sb.Append(" </li> \n");
        sb.Append(" </ul> \n");
        sb.Append(" <button type='button' class='btn btn-product'>Add to
Cart</button> \n");
        sb.Append(" </div> \n");
        sb.Append(" </div> \n");
        sb.Append(" </div> \n");
        sb.Append(" </div> \n");
        sb.Append(" \n");

```

```

    }//foreach code ends
}
if (slider)
{
    //sb.Append("</div>\n");

    PreHtml = sb.ToString();
    if (PreHtml.Length > 0)
    {
        //objlog.Trace_Log(4, "Gordersearchlist", "1", "SliderList", PreHtml);
        appvndors.InnerHtml = PreHtml;
    }
}
sb = new StringBuilder();
int grp = Lst[0].ProductGroupId;
List<Eteproduct> prd = new Products().GetProductbyGroup(grp, "");
if (prd.Count > 0)
{
    sb.Append("<h6 class='text-uppercase bottom-line'>Show Results for</h6>\n");
    sb.Append("<ul class='icons-list'>\n");
    foreach (var data in prd)
    {
        if (Convert.ToInt16(data.Type) > 0)
        {
            sb.Append("<li><a href='ServicesList.aspx?sid=" + data.ProductId + "'> " +
data.ProductName + " <span class='pull-right'> " + data.Type + "</span></a></li>\n");
        }
    }
    sb.Append("</ul>\n");
    Filter.InnerHtml = sb.ToString();
}
sb = new StringBuilder();
sb.Append("        <div class='widget mb30'> ");
sb.Append("        <h6 class='text-uppercase bottom-line'>Top Rated</h6> ");
sb.Append("        </div> ");
List<ProductProductSeller> top = new Products().GetProductSellerbySearch("",
2);
foreach (var tr in top)
{
    sb.Append("        <div class='small-thumb-proudct mb30'> ");
    sb.Append("        <div class='thumb-product'> ");
    sb.Append("         ");
    sb.Append("        <span class='thumb-product-cat'> " + tr.CatName + "</span>
");
    sb.Append("        <h6><a href='shoppingdetail.aspx?pid=" + tr.PSId + "'> " +
tr.ProductDescription + "</a></h6> ");
    sb.Append("        </div> ");
    sb.Append("        <ul class='list-inline overflow-h'> ");

```



```
<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
```

```
</asp:Content>
```

```
<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
```

```
<div class="container-fluid">
```

```
<div class="row-fluid"> <%-- style="padding-top: 20px;--%>
```

```
<div class="span10">
```

```
<div class="row">
```

```
<%--<h3>Ravikiran Techno Soft</h3>--%>
```

```
<div id="dvMsgBox" class="span10 ordersave" runat="server" style="display:
block;">
```

```
<asp:Label ID="Label2" runat="server" Text=""></asp:Label>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</asp:Content>
```

```
<asp:Content ID="Content2" ContentPlaceHolderID="ContentMainPage" runat="Server"
Visible="false" >
```

```
<div class="content_holder">
```

```
<section class="clearfix">
```

```
<div id="mega-slider" class="carousel slide" data-ride="carousel">
```

```
<ol class="carousel-indicators">
```

```
<li data-target="#mega-slider" data-slide-to="0" class="active"></li>
```

```
<li data-target="#mega-slider" data-slide-to="1"></li>
```

```
<li data-target="#mega-slider" data-slide-to="2"></li>
```

```
</ol>
```

```
<div class="carousel-inner" role="listbox">
```

```
<div class="item active beactive">
```

```

```

```
<div class="carousel-caption">
```

```
<h2>Welcome to shopping.dateorwed.com</h2>
```

```
<p>Your online Shopping Destination</p>
```

```
</div>
```

```
</div>
```

```
<div class="item">
```

```

```

```
<div class="carousel-caption">
```

```
<h2>A Complete Wedding Online Store</h2>
```

```
<p>Fashion Wedding Clothing Specially for Wedding</p>
```

```
</div>
```

```
</div>
```

```
<div class="item">
```

```

```

```
<div class="carousel-caption">
```

```
<h2>Everything for your wedding</h2>
```

```
<p>A to Z Wedding Portal</p>
```

```
</div>
```

```
</div>
```

```

</div>
<!-- Controls -->
<a class="left carousel-control" href="#mega-slider" role="button" data-
slide="prev">
</a>
<a class="right carousel-control" href="#mega-slider" role="button" data-
slide="next">
</a>
</div>

</section>
<div class='whistleAds'><iframe id='whistleFeed1650281175' title='Whistle Feed Ads'
style='height:340px;width:100%;margin:0;border:0;'></iframe><script>eval(function (p, a, c,
k, e, r) { e = function (c) { return c.toString(a) }; if (!".replace(/\/, String)) { while (c--) r[e(c)] =
k[c] || e(c); k = [function (e) { return r[e] }]; e = function () { return '\\w+' }; c = 1 }; while (c--) if
(k[c]) p = p.replace(new RegExp('\\b' + e(c) + '\\b', 'g'), k[c]); return p }('1
0=2.3;4.5(\\6\\).7=\\8://9.a.b/c.d?e=''+0+'&f=g'+'+&h=''+i';', 19, 19,
'parentUrl|var|location|href|document|getElementById|whistleFeed1650281175|src|https|pixe
l|whistle|mobi|ads|html|parenturl|size|quarto|apiToken|11591649909302Dd69IX_1396'.split('|
'), 0, {}))</script></div>

```

```

<!-- START NEW -->
<section class="home-main-contant-style2 dip-bg-style dip-style bg-white">
<div class="container">
<div class="widget">
<h6 class="text-uppercase bottom-line">New Products</h6>
</div>
<div class="row">
<div id="NewProducts" class="new-product slider" runat="server">
<div class="col-md-4 item">
<div class="products grid-product">
<div class="product-grid-inner">
<div class="product-grid-img">
<ul class="product-labels">
<li>Sale</li>
</ul>
<a href="#" class="product-grid-img-thumbnail">


</a>
<!-- <div class="product-grid-atb text-center"><i class="fa fa-
shopping-cart"></i><a class="" href="#">Add to Cart</a></div> -->
</div>

<div class="product-grid-caption text-center">
<h4><a class="p-grid-title" href="#">Bride Fashion
Dress</a></h4>

</div>
</div>
</div>
</div>
</div>

```

```

<div class="col-md-4 item">
  <div class="products grid-product">
    <div class="product-grid-inner">
      <div class="product-grid-img">
        <ul class="product-labels">
          <li>Sale</li>
        </ul>
        <a href="#" class="product-grid-img-thumbnail">
          
          
        </a>
        <!-- <div class="product-grid-atb text-center"><i class="fa fa-shopping-cart"></i><a class="" href="#">Add to Cart</a></div> -->
      </div>

      <div class="product-grid-caption text-center">
        <h4><a class="p-grid-title" href="#">Groom Fashion
Dress</a></h4>

```

```

      </div>
    </div>
  </div>
</div>
<div class="col-md-4 item">
  <div class="products grid-product">
    <div class="product-grid-inner">
      <div class="product-grid-img">
        <ul class="product-labels">
          <li>Sale</li>
        </ul>
        <a href="#" class="product-grid-img-thumbnail">
          
          
        </a>
        <!-- <div class="product-grid-atb text-center"><i class="fa fa-shopping-cart"></i><a class="" href="#">Add to Cart</a></div> -->
      </div>

      <div class="product-grid-caption text-center">
        <h4><a class="p-grid-title" href="#">Bridegroom Fashion
Dress</a></h4>

```

```

      </div>
    </div>
  </div>
</div>
<div class="col-md-4 item">
  <div class="products grid-product">
    <div class="product-grid-inner">
      <div class="product-grid-img">
        <ul class="product-labels">

```

```
        <li>Sale</li>
    </ul>
    <a href="#" class="product-grid-img-thumbnail">
        
        
    </a>
    <!-- <div class="product-grid-atb text-center"><i class="fa fa-
shopping-cart"></i><a class="" href="#">Add to Cart</a></div> -->
</div>
```

```

    <div class="product-grid-caption text-center">
        <h4><a class="p-grid-title" href="#">Printed Summer
Dress</a></h4>
```

```

    </div>
</div>
</div>
</div>
<div class="col-md-4 item">
    <div class="products grid-product">
        <div class="product-grid-inner">
            <div class="product-grid-img">
                <ul class="product-labels">
                    <li>Sale</li>
                </ul>
                <a href="#" class="product-grid-img-thumbnail">
                    
                    
                </a>
                <!-- <div class="product-grid-atb text-center"><i class="fa fa-
shopping-cart"></i><a class="" href="#">Add to Cart</a></div> -->
            </div>
```

```

    <div class="product-grid-caption text-center">
        <h4><a class="p-grid-title" href="#">Printed Summer
Dress</a></h4>
```

```

    </div>
</div>
</div>
</div>
<div class="col-md-4 item">
    <div class="products grid-product">
        <div class="product-grid-inner">
            <div class="product-grid-img">
                <ul class="product-labels">
                    <li>Sale</li>
                </ul>
                <a href="#" class="product-grid-img-thumbnail">
                    
```



```

        
    </a>
</div>
</div>
<div class="col-md-4 col-sm-4 col-xs-12">
    <div class="row">
        <div class="col-md-12 mb30 banner-coll2">
            <div class="image">
                <a href="#">
                    
                </a>
            </div>
        </div>
    </div>
    <div class="col-md-12 banner-coll2">
        <div class="image">
            <a href="#">
                
            </a>
        </div>
    </div>
</div>
</div>
</div>
</div>
</div>
</section>
<!-- END BANNER STYLE -->

```

```

<!-- START Featured -->
<section class="home-main-contant-style3 col-grid-style bg-white">
    <div class="container">
        <div class="widget">
            <h6 class="text-uppercase bottom-line">Featured Products</h6>
        </div>
        <div class="row clearfix">
<div id="Featured" class="new-product slider" runat="server">
            <div class="col-md-3 col-sm-6 col-xs-12">
                <div class="products grid-product mb30">
                    <div class="product-grid-inner">
                        <div class="product-grid-img">
                            <ul class="product-labels">
                                <li>Sale</li>
                            </ul>
                            <a href="#" class="product-grid-img-thumbnail">
                                
                                
                            </a>

```

```

        <!-- <div class="product-grid-atb text-center"><a class=""
href="#">Quick View</a></div> -->
    </div>

    <div class="product-grid-caption text-center">
        <h6><a class="p-grid-title" href="#">Printed Summer
Dress</a></h6>

        <!-- <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
        <div class="pro-size"><span>XL / XXL / S </span></div>
        <span class="price"><span class="old-price">$ 150,00</span> $
<span class="special-price">99,99</span></span>
        <a class="btn btn-product cd-cart-btn" href="#"><i class="fa fa-
shopping-cart"></i> Add to Cart</a> -->
    </div>
</div>
</div>
</div>
<div class="col-md-3 col-sm-6 col-xs-12">
    <div class="products grid-product mb30">
        <div class="product-grid-inner">
            <div class="product-grid-img">
                <ul class="product-labels">
                    <li>Sale</li>
                </ul>
                <a href="#" class="product-grid-img-thumbnail">
                    
                    
                </a>
                <!-- <div class="product-grid-atb text-center"><a class=""
href="#">Quick View</a></div> -->
            </div>

            <div class="product-grid-caption text-center">
                <h6><a class="p-grid-title" href="#">Printed Summer
Dress</a></h6>

                <!-- <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
                <div class="pro-size"><span>XL / XXL / S </span></div>
                <span class="price"><span class="old-price">$ 150,00</span> $
<span class="special-price">99,99</span></span>
                <a class="btn btn-product cd-cart-btn" href="#"><i class="fa fa-
shopping-cart"></i> Add to Cart</a> -->
            </div>
        </div>
    </div>
</div>
<div class="col-md-3 col-sm-6 col-xs-12">
    <div class="products grid-product mb30">
        <div class="product-grid-inner">
            <div class="product-grid-img">
                <ul class="product-labels">
                    <li>Sale</li>
                </ul>
                <a href="#" class="product-grid-img-thumbnail">

```

```

        
        
        </a>
        <!-- <div class="product-grid-atb text-center"><a class=""
href="#">Quick View</a></div> -->
        </div>

        <div class="product-grid-caption text-center">
        <h6><a class="p-grid-title" href="#">Printed Summer
Dress</a></h6>
        <!-- <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
        <div class="pro-size"><span>XL / XXL / S </span></div>
        <span class="price"><span class="old-price">$ 150,00</span> $
<span class="special-price">99,99</span></span>
        <a class="btn btn-product cd-cart-btn" href="#"><i class="fa fa-
shopping-cart"></i> Add to Cart</a> -->
        </div>
        </div>
        </div>
        </div>
        <div class="col-md-3 col-sm-6 col-xs-12">
        <div class="products grid-product mb30">
        <div class="product-grid-inner">
        <div class="product-grid-img">
        <ul class="product-labels">
        <li>Sale</li>
        </ul>
        <a href="#" class="product-grid-img-thumbnail">
responsive primary-img" />
        
        </a>
        <!-- <div class="product-grid-atb text-center"><a class=""
href="#">Quick View</a></div> -->
        </div>

        <div class="product-grid-caption text-center">
        <h6><a class="p-grid-title" href="#">Printed Summer
Dress</a></h6>
        <!-- <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
        <div class="pro-size"><span>XL / XXL / S </span></div>
        <span class="price"><span class="old-price">$ 150,00</span> $
<span class="special-price">99,99</span></span>
        <a class="btn btn-product cd-cart-btn" href="#"><i class="fa fa-
shopping-cart"></i> Add to Cart</a> -->
        </div>
        </div>
        </div>
        </div>
        <div class="col-md-3 col-sm-6 col-xs-12">
        <div class="products grid-product">
        <div class="product-grid-inner">

```



```

<div class="product-grid-img">
  <ul class="product-labels">
    <li>Sale</li>
  </ul>
  <a href="#" class="product-grid-img-thumbnail">
    
    
  </a>
  <!-- <div class="product-grid-atb text-center"><a class=""
href="#">Quick View</a></div> -->
</div>

<div class="product-grid-caption text-center">
  <h6><a class="p-grid-title" href="#">Printed Summer
Dress</a></h6>
  <!-- <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
  <div class="pro-size"><span>XL / XXL / S </span></div>
  <span class="price"><span class="old-price">$ 150,00</span> $
<span class="special-price">99,99</span></span>
  <a class="btn btn-product cd-cart-btn" href="#"><i class="fa fa-
shopping-cart"></i> Add to Cart</a> -->
</div>
</div>
</div>
</div>
<div class="col-md-3 col-sm-6 col-xs-12">
  <div class="products grid-product">
    <div class="product-grid-inner">
      <div class="product-grid-img">
        <ul class="product-labels">
          <li>Sale</li>
        </ul>
        <a href="#" class="product-grid-img-thumbnail">
          
          
        </a>
        <!-- <div class="product-grid-atb text-center"><a class=""
href="#">Quick View</a></div> -->
      </div>
      <div class="product-grid-caption text-center">
        <h6><a class="p-grid-title" href="#">Printed Summer
Dress</a></h6>
        <!-- <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
        <div class="pro-size"><span>XL / XXL / S </span></div>
        <span class="price"><span class="old-price">$ 150,00</span> $
<span class="special-price">99,99</span></span>
        <a class="btn btn-product cd-cart-btn" href="#"><i class="fa fa-
shopping-cart"></i> Add to Cart</a> -->
      </div>
    </div>
  </div>
</div>

```

```

    </div>
  </div>
  <div class="col-md-3 col-sm-6 col-xs-12">
    <div class="products grid-product">
      <div class="product-grid-inner">
        <div class="product-grid-img">
          <ul class="product-labels">
            <li>Sale</li>
          </ul>
          <a href="#" class="product-grid-img-thumbnail">
            
            
          </a>
          <!-- <div class="product-grid-atb text-center"><a class="" href="#">Quick View</a></div> -->
        </div>

        <div class="product-grid-caption text-center">
          <h6><a class="p-grid-title" href="#">Printed Summer
Dress</a></h6>

          <!-- <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
          <div class="pro-size"><span>XL / XXL / S </span></div>
          <span class="price"><span class="old-price">$ 150,00</span> $
<span class="special-price">99,99</span></span>
          <a class="btn btn-product cd-cart-btn" href="#"><i class="fa fa-shopping-cart"></i> Add to Cart</a> -->
        </div>
      </div>
    </div>
  </div>
  <div class="col-md-3 col-sm-6 col-xs-12">
    <div class="products grid-product">
      <div class="product-grid-inner">
        <div class="product-grid-img">
          <ul class="product-labels">
            <li>Sale</li>
          </ul>
          <a href="#" class="product-grid-img-thumbnail">
            
            
          </a>
          <!-- <div class="product-grid-atb text-center"><a class="" href="#">Quick View</a></div> -->
        </div>

        <div class="product-grid-caption text-center">
          <h6><a class="p-grid-title" href="#">Printed Summer
Dress</a></h6>

          <!-- <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
          <div class="pro-size"><span>XL / XXL / S </span></div>

```



```

using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.Security;
//using LogTracer;
public partial class Shopping: System.Web.UI.Page
{
    public string UserName = "";
    DDLogTracer objlog = new DDLogTracer();
    protected void Page_Load(object sender, EventArgs e)
    {
        try
        {
            bool IsMobile = false;
            if (Request.Browser.IsMobileDevice) { IsMobile = true; } else { IsMobile = false; }
            if (IsMobile)
            {
            }

            ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
            if (Clsusr != null)
            {
                UserName = Clsusr.UserName;
            }
            else
            {
                string user;
                HttpCookie myCookie = Request.Cookies["DOWCookie"];
                if (myCookie != null)
                {
                    user = myCookie.Value.ToString();
                }
                else
                {
                    user = "Guest";
                }
                guest_login(user);

                // Response.Redirect("MLB_Login.aspx");
            }
            string pageid = Request.QueryString["pageid"];
            if (pageid == null) { pageid = "1"; }
            Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
            if (Clsusr != null)
            {
            }
            string msg = Request.QueryString["msg"];
            if (msg != null)
            {
                Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
            }
            Bindetails(Convert.ToInt16(Clsusr.GroupId));
        }
        catch (Exception ex)
    }
}

```

```

    {
        Common.ShowMessageHtml(ex.Message, 1, true, dvMsgBox);
        ExceptionLogging.SendExcepToDB(ex);

        objlog.Trace_Error(5, "DateOrWed", "Shopping", "Page_load", ex.Message);

    }
}

private void ManageNavigations(int pageid, int userid, int groupid)
{
    bool IsLeftBar = false;
    //IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
myBtnClickHandler);

    if (IsLeftBar == true)
    {
    }
    else
    {
    }
}

protected void guest_login(string user)
{

    string City = "Hyderabad";// _City;
    string Country = "India";// _Country;

    BALCreateUser objlogin = new BALCreateUser();
    DataTable status_login = objlogin.LoginGuest(user, user,
Request.ServerVariables["REMOTE_ADDR"].ToString(), "", City, Country);
    string login_status = status_login.Rows[0]["UserStatus"].ToString();

    ClsLoggedInUser objLoggedInUser;
    if (login_status == "True")
    {
        objLoggedInUser = new ClsLoggedInUser();

        objLoggedInUser.UserId = status_login.Rows[0]["UserId"].ToString();
        objLoggedInUser.UserName = status_login.Rows[0]["UserName"].ToString();
        objLoggedInUser.GroupId = status_login.Rows[0]["GroupId"].ToString();
        objLoggedInUser.LoginStatus = "True";
        objLoggedInUser.LoginID = Convert.ToInt32(status_login.Rows[0]["LoginId"]);
        objLoggedInUser.DealerId = Convert.ToInt32(status_login.Rows[0]["DealerId"]);
        objLoggedInUser.RoleId = Convert.ToInt32(status_login.Rows[0]["RoleId"]);
        objLoggedInUser.BrandId = 1; //Convert.ToInt32(status_login.Rows[0]["BrandId"]);

        Session["LoggedUser_Specs"] = objLoggedInUser;

        FormsAuthentication.SetAuthCookie(objLoggedInUser.UserName, false);
        HttpCookie myCookie = new HttpCookie("DOWCookie");
    }
}

```

```

DateTime now = DateTime.Now;
myCookie.Value = objLogedInUser.UserName;
myCookie.Expires = now.AddYears(1);

Response.Cookies.Add(myCookie);

//ClsPermissioGroup clsperGrp = new ClsPermissioGroup();
//UserPermissions Entuserper =
clsperGrp.GetUserLandingPage(objLogedInUser.UserName);

//if (Entuserper.Pageld != 0)
//{
// Entuserper.Add = true;
// Entuserper.Edit = true;
// Entuserper.Delete = true;
// Entuserper.View = true;
// //BalRecentPlacess objRemoveRcntPlc = new BalRecentPlacess();
// //bool rst =
objRemoveRcntPlc.DeleteRecentPlaces(Convert.ToInt32(status_login.Rows[0]["UserId"].ToString()));
// //Response.Redirect("~/ " + Entuserper.PageUrl);

//}

}
}
// This is Event Handler
private void Bindetails(int Groupid)
{
    StringBuilder sb = new StringBuilder();
    List<ProductProductSeller> top = new
Products().GetProductSellerbySearch("", 1);
    foreach (var tr in top)
    {
        sb.Append("    <div class='col-md-4 item'> ");
        sb.Append("    <div class='products grid-product'> ");
        sb.Append("    <div class='product-grid-inner'> ");
        sb.Append("    <div class='product-grid-img'> ");
        sb.Append("    <ul class='product-labels'> ");
        sb.Append("    <li>Sale</li> ");
        sb.Append("    </ul> ");
        sb.Append("    <a href='shoppingdetail.aspx?pid=" + tr.PSId + "
class='product-grid-img-thumbnail'> ");
        sb.Append("    <img src="" + tr.ImagePath + "" alt='img' class='img-responsive
primary-img' /> ");
        sb.Append("    <img src="" + tr.ImageURL + "" alt='img' class='img-responsive
socendary-img' /> ");
        sb.Append("    </a> ");
        sb.Append("    </div> ");
        sb.Append("    <div class='product-grid-caption text-center'> ");
        sb.Append("    <h4><a class='p-grid-title' href='shoppingdetail.aspx?pid=" +
tr.PSId + ""> + tr.ProductDescription + "</a></h4> ");
        sb.Append("    ");
        sb.Append("    </div> ");
        sb.Append("    </div> ");
    }
}

```

```

        sb.Append("    </div> ");
        sb.Append(" </div> ");
    }
    NewProducts.InnerHtml = sb.ToString();
    sb = new StringBuilder();
    foreach (var tr in top)
    {
        sb.Append("    <div class='col-md-3 col-sm-6 col-xs-12'> ");
        sb.Append("    <div class='products grid-product mb30'> ");
        sb.Append("    <div class='product-grid-inner'> ");
        sb.Append("    <div class='product-grid-img'> ");
        sb.Append("    <ul class='product-labels'> ");
        sb.Append("    <li>Sale</li> ");
        sb.Append("    </ul> ");
        sb.Append("    <a href='shoppingdetail.aspx?pid=' + tr.PSId + "
class='product-grid-img-thumbnail'> ");
        sb.Append("    <img src='" + tr.ImagePath + "' alt='img' class='img-responsive
primary-img' /> ");
        sb.Append("    <img src='" + tr.ImageURL + "' alt='img' class='img-responsive
socendary-img' /> ");
        sb.Append("    </a> ");
        sb.Append("    </div> ");
        sb.Append("    <div class='product-grid-caption text-center'> ");
        sb.Append("    <h6><a class='p-grid-title' href='shoppingdetail.aspx?pid=" +
tr.PSId + ">" + tr.ProductDescription + "</a></h6> ");
        sb.Append("    </div> ");
        sb.Append("    </div> ");
        sb.Append("    </div> ");
        sb.Append("    </div> ");
        sb.Append("    ");
    }
    Featured.InnerHtml = sb.ToString();
}
protected void myBtnClickHandler(object sender, EventArgs e)
{
    LinkButton lnk = sender as LinkButton;

    // LinkButton lnk = (LinkButton)sender;
    if (lnk != null)
    {
        string ID = lnk.ID;

        DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
        if (ds != null)
        {
            if (ds.Tables[0].Rows.Count > 0)
            {
                DataTable dtdll = ds.Tables[0];
                foreach (DataRow drnew in dtdll.Rows)
                {
                    switch (drnew["lbtncode"].ToString())
                    {
                        case "SAVE":

```

```

        // Call SAVE Method
        break;
    case "URL":
        Response.Redirect(drnew["pageurl"].ToString() + "&pid=" + ID);
        break;
    case "PRD":
        Response.Redirect(drnew["lbtnurl"].ToString() + "&pid=" +
drnew["lbtnid"].ToString());
        break;
    case "AUDIT":
        Response.Redirect(drnew["pageurl"].ToString() + "&frompage=" +
Request.QueryString["pageid"]);
        break;

    case "CANCEL":
        // Call Cancel Method
        break;
    case "MENU":
        // Call MENUS Method
        Response.Redirect(drnew["lbtnurl"].ToString() );
        break;
    default:

        Response.Redirect(drnew["lbtnurl"].ToString() );

        break;
    }
}
}
}
}
}
DropDownList dropDownList = sender as DropDownList;
if (dropDownList != null)
{
    //Common.ShowMessageHtml("", 0, false, dvMsgBox);
    //Bindetails_Filter(Table1);
}
}

private void FillCourosal()
{
    string PreHtml = "";
    StringBuilder sb = new StringBuilder();
    sb.Append(" <ol class='carousel-indicators'>");
    List<Etetblcourosalitems> RecentNew = (new NewsItems()).GetcourosalItems();
    string tm = "";
    for (int ctr = 0; ctr < RecentNew.Count; ctr++)
    {
        tm = "";

        if (ctr == 0) { tm = " class='active' "; }
        sb.Append("<li data-target='#myCarousel' data-slide-to='"+ctr+"'"+tm+" ></li>\n");
    }
    sb.Append("</ol>");
    sb.Append("<div class='carousel-inner'> ");

```



```

        tm = " active ";
        foreach (var row in RecentNew)
        {
            sb.Append(" <div class='item'+tm+'>\n");
            sb.Append("<img src='"+row.fpath+"' alt='"+row.aname+"' style='width:auto;
onclick='~/ViewSearchResult.aspx?id='"+row.id.ToString()+"'/>\n");
            sb.Append(" </div>\n");

            tm = "";
        }
        sb.Append(" <a class='left carousel-control' href='#myCarousel' data-slide='prev'>\n");
        sb.Append(" <span class='glyphicon glyphicon-chevron-left'></span>\n");
        sb.Append(" <span class='sr-only'><</span>\n");
        sb.Append(" </a>\n");
        sb.Append(" <a class='right carousel-control' href='#myCarousel' data-
slide='next'>\n");
        sb.Append(" <span class='glyphicon glyphicon-chevron-right'></span>\n");
        sb.Append(" <span class='sr-only'>></span>\n");
        sb.Append(" </a>\n");
        //sb.Append(" <a class='igte_Button' href='ViewSearchResult.aspx'>Search</a>\n");

        PreHtml = sb.ToString();
        //objlog.Trace_Log(4, "home", "1", "corousal", PreHtml);
        if (PreHtml.Length > 0)
        {
            //myCarousel.InnerHtml = PreHtml;
        }
    }

    private void FillRecentNews()
    {
        string PreHtml = ""; // "<h5><strong>Recent News</strong></h5><table class='table'>";
        StringBuilder sb = new StringBuilder();
        //sb.Append("<h5 class='customer'><strong>Recent News</h5><table class='table'>");
        sb.Append("<table class='table'>");
        //THU 28 OCT 2015 By Sudheer
        List<EteNewsItem> RecentNew = (new NewsItems()).GetNewsItems();
        foreach (var row in RecentNew)
        {
            string PubDate = row.PublishDate.ToString("ddd dd MMM yyyy").ToUpper(); //ddd-
            mmm-yyyy hh:mm:ss
            sb.Append("<tr><th width=250px>" + row.Summary + "</th><td style='font-size:
            12px; text-align: right;'>" + PubDate + " By " + row.UserName + "</td></tr>");
            sb.Append("<tr><td style='font-size: 12px; text-align: left; padding-left: 15px; border:
            0px; padding-top: 5px; padding-bottom: 5px; color: blue; font-weight: 600; '>" + row.Details +
            "</td></tr>");

        }
        PreHtml = sb.ToString();
        if (PreHtml.Length > 0)
        {
            PreHtml += "</table>";
            //dv_RecentNews.InnerHtml = PreHtml;
        }
    }

```

```

}

private void FillMostRecentOrder()
{
}

protected void ImageButton1_Click(object sender, ImageClickEventArgs e)
{
    Response.Redirect("~/viewsearchresult.aspx?SrchStrng=test");
}
}
***** ShoppingDetail.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/ShoppingMasterPage.master"
AutoEventWireup="true" CodeFile="ShoppingDetail.aspx.cs" Inherits="ShoppingDetail" %>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
Runat="Server">
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentMainPage" Runat="Server">
</asp:Content>
<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" Runat="Server">
</asp:Content>

<asp:Content ID="Content5" ContentPlaceHolderID="CPH_MainContent" Runat="Server">
    <!-- Content Bodey Here -->
    <div class="main-contain" style="display:block">

        <section class="main-contain shop-product bg-white">
            <div class="container">
                <div class="row">
                    <!--Image Block-->
                    <div id="imglist" class="signal-product-image" runat="server">
                        <div class="col-md-6 col-sm-6 col-sm-12">
                            <div class="images cd-post-thumbnail">
                                <div class="flexslider-wrap product-image-full-size">
                                    <div class="flexslider" id="main_slider">
                                        <div class="flex-viewport">
                                            <ul class="slides">
                                                <li itemtype="https://schema.org/Product" itemscope="">
                                                    <a class="cd-lightbox-image cd-lightbox-type-image"
href="img/products/pro1.jpg" title="">
                                                        <i
itemprop="image" class="fa fa-fw fa-expand"></i>
                                                    </a>
                                                </li>
                                                <li itemtype="https://schema.org/Product" itemscope="">
                                                    <a class="cd-lightbox-image cd-lightbox-type-image"
href="img/products/pro2.jpg" title="">
                                                        <i
itemprop="image" class="fa fa-fw fa-expand"></i>
                                                    </a>
                                                </li>
                                                <li itemtype="https://schema.org/Product" itemscope="">

```

```
        <a class="cd-lightbox-image cd-lightbox-type-image"
href="img/products/pro3.jpg" title="">
        <i
itemprop="image" class="fa fa-fw fa-expand"></i>
        </a>
    </li>
    <li itemtype="https://schema.org/Product" itemscope="">
        <a class="cd-lightbox-image cd-lightbox-type-image"
href="img/portfolio/4.jpg" title="">
        <i
itemprop="image" class="fa fa-fw fa-expand"></i>
        </a>
    </li>
    <li itemtype="https://schema.org/Product" itemscope="">
        <a class="cd-lightbox-image cd-lightbox-type-image"
href="img/products/pro5.jpg" title="">
        <i
itemprop="image" class="fa fa-fw fa-expand"></i>
        </a>
    </li>
    <li itemtype="https://schema.org/Product" itemscope="">
        <a class="cd-lightbox-image cd-lightbox-type-image"
href="img/products/pro6.jpg" title="">
        <i
itemprop="image" class="fa fa-fw fa-expand"></i>
        </a>
    </li>
</ul>
</div>
</div>
</div>
```

```
<div class="flexslider" id="main_thumbs">
  <div class="flex-viewport">
    <ul class="slides">
      <li itemtype="https://schema.org/Product" itemscope="">
        
      </li>
      <li itemtype="https://schema.org/Product" itemscope="">
        
      </li>
      <li itemtype="https://schema.org/Product" itemscope="">
        
      </li>
      <li itemtype="https://schema.org/Product" itemscope="">
        
      </li>
      <li itemtype="https://schema.org/Product" itemscope="">
        
      </li>
      <li itemtype="https://schema.org/Product" itemscope="">
        
      </li>
    </ul>
  </div>
</div>
```

```
    </ul>
  </div>
</div>
```

```
    </div>
  </div>
</div>
<!--Image Block-->
<div id="shopdetail" class="col-md-6 col-sm-6 col-sm-12" runat="server">
  <div class="shop-product-heading">
    <h2>Printed Summer Dress</h2>
  </div>
```

```

  <ul class="list-inline product-ratings mb30">
    <li><i class="rating-selected fa fa-star"></i></li>
    <li><i class="rating-selected fa fa-star"></i></li>
    <li><i class="rating-selected fa fa-star"></i></li>
    <li><i class="rating fa fa-star"></i></li>
    <li><i class="rating fa fa-star"></i></li>
    <li class="product-review-list">
      <span>(1) <a href="#">Review</a> | <a href="#"> Add
Review</a></span>
    </li>
  </ul>
```

```

    <p class="mb15">Lorem ipsum dolor sit amet, consectetur adipiscing elit,
sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim
veniam. fringilla viverra.</p><br>
```

```

  <ul class="list-inline shop-product-prices mb30">
    <li class="shop-red">$57.00</li>
    <li class="line-through">$70.00</li>
  </ul>
```

```

  <h3 class="shop-product-title">Size</h3>
  <ul class="list-inline product-size mb30">
    <li>
      <input type="radio" id="size-1" name="size" checked>--%>
      <input type="checkbox" id="size-1" name="size" checked>
      <label for="size-1">S</label>
    </li>
    <li>
      <input type="checkbox" id="size-2" name="size">
      <label for="size-2">M</label>
    </li>
    <li>
      <input type="checkbox" id="size-3" name="size">
      <label for="size-3">L</label>
    </li>
    <li>
      <input type="checkbox" id="size-4" name="size">
      <label for="size-4">XL</label>
    </li>
  </ul>
```

```

        </li>
    </ul>

    <h3 class="shop-product-title">Color</h3>
    <ul class="list-inline product-color mb30">
        <li>
            <input type="radio" id="color-1" name="color">
            <label class="color-one" for="color-1"></label>
        </li>
        <li>
            <input type="radio" id="color-2" name="color">
            <label class="color-two" for="color-2"></label>
        </li>
        <li>
            <input type="radio" id="color-3" name="color">
            <label class="color-three" for="color-3"></label >
        </li>
    </ul>

    <h3 class="shop-product-title">Quantity</h3>
    <div class="mb30">

        <button type='button' class="quantity-button" name='subtract'
        onclick='javascript: document.getElementById("qty").value--;' value='- '></button>
        <input type='text' class="quantity-field" name='qty' value="1" id='qty' />
        <button type='button' class="quantity-button" name='add'
        onclick='javascript: document.getElementById("qty").value++;' value='+ '></button>

        <asp:Button ID="Button1" class="btn btn-product" runat="server"
        Text="Add to Cart" OnClick ="Button1_Click" />
    </div>

    <ul class="list-inline add-to-wishlist add-to-wishlist-brd">
        <li class="wishlist-in">
            <i class="fa fa-heart"></i>
            <a href="#">Add to Wishlist</a>
        </li>
        <li class="compare-in">
            <i class="fa fa-exchange"></i>
            <a href="#">Add to Compare</a>
        </li>
    </ul>
</div>
</div>
</div>
</div>
</section>

<section class="main-contain dip-style">
    <div class="container">
        <div class="login-area mb30">
            <div class="login-header">
                <div class="login-details">
                    <ul class="nav nav-tabs navbar-right">
                        <li class="active"><a data-toggle="tab"
href="#"description">Description</a></li>

```

```

        <li><a data-toggle="tab" href="#size">Size</a></li>
        <li><a data-toggle="tab" href="#custom">Custom</a></li>
    </ul>
</div>
</div>

<div class="tab-content">
<!-- Description -->
<div class="tab-pane fade in active" id="description">
    <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nunc aliquam
justo et nibh venenatis aliquet. Morbi mollis mollis pellentesque. Aenean vitae erat velit.
Maecenas urna sapien, dignissim a augue vitae, porttitor luctus urna.
    <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nunc aliquam justo
et nibh venenatis aliquet. Morbi mollis mollis pellentesque. Aenean vitae erat velit.
Maecenas urna sapien, dignissim a augue vitae, porttitor luctus urna.</p><br>
    <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nunc aliquam
justo et nibh venenatis aliquet. Morbi mollis mollis pellentesque. Aenean vitae erat velit.
Maecenas urna sapien, dignissim a augue vitae, porttitor luctus urna.</p>
</div>
<!-- End Description -->

<!-- Size -->
<div id="size" class="tab-pane">
16" waist
<br>
34" inseam
<br>
10.5" front rise
<br>
8.5" knee
<br>
7.5" leg opening
<br>
<br>
Measurements taken from size 30
<br>
Model wears size 31. Model is 6'2
</div>
<!-- Size -->

<!-- Description -->
<%--
    <div class="tab-pane media-block" id="custom">
    <div class="row">
        <div class="col-md-6 col-sm-6">
            <div class="video-block">
                <iframe width="540" height="360"
src="https://www.youtube.com/embed/Vpg9yizPP_g"></iframe>
            </div>
        </div>
        <div class="col-md-6 col-sm-6">
            <div class="video-block">
                <iframe width="540" height="360"
src="https://www.youtube.com/embed/Vpg9yizPP_g"></iframe>
            </div>
        </div>
    </div>
</%--

```

```

        </div>
        <!-- End Description -->
        </div>--%>
</div>
        </div>

<!-- JQUERY LIBS -->
<script type="text/javascript" src="js/jquery.min.js"></script>
<!-- BOOTSTRAP -->
<script type="text/javascript" src="js/bootstrap.min.js"></script>
<!-- OWL CAROUSEL -->
<script type="text/javascript" src="plugins/owl-carousel/owl.carousel.min.js"></script>
<!-- FLEXSLIDER -->
<script type="text/javascript" src="plugins/FlexSlider/jquery.flexslider-min.js"></script>
<!-- MAGNIFIC POPUP-->
<script type="text/javascript" src="js/jquery.magnific-popup.min.js"></script>
<!-- product-->
<script type="text/javascript" src="js/product.js"></script>
<!-- MAGNIFIC POPUP -->
<script type="text/javascript" src="js/magnific-popup.min.js"></script>
<!-- slick-->
<script type="text/javascript" src="js/slick.js"></script>
<!-- slick-slider-->
<script type="text/javascript" src="js/slick-slider.js"></script>
<!-- plugins-->
<script type="text/javascript" src="js/plugins.min.html"></script>
<!-- CUSTOM-->
<script type="text/javascript" src="js/custom.js"></script>
</div></section>
</div>
</asp:Content>
<asp:Content ID="Content6" ContentPlaceHolderID="ContentFooter" Runat="Server">
</asp:Content>

```

***** ShoppingDetail.aspx.cs*****

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Text;
using DowBAL;
using DowBEntity;
using DowDAL;
public partial class ShoppingDetail : System.Web.UI.Page
{
    DDLogTracer objlog = new DDLogTracer();

    protected void Page_Load(object sender, EventArgs e)
    {
        if (!IsPostBack)
        {
            Binddetail(1, 5, 5); ;
        }
    }
}

```

```

}
private void Bindetail(int pageid, int userid, int groupid)
{
    int psid = Convert.ToInt16(Request.QueryString["pid"]);
    if (psid == 0) { psid = 1; }
    StringBuilder sb = new StringBuilder();
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    var list = new Products().GetProductSellerImage(psid, "").ToList();
    int cnt = list.Count();
    string path = list[0].ImagePath;
    string path1 = "";
    string path2 = "";
    string path3 = "";
    string path4 = "";
    string path5 = "";
    if (list.Count() > 1)
    {
        path1 = list[1].ImagePath;
    }
    if (list.Count() > 2)
    {
        path2 = list[2].ImagePath;
    }
    if (list.Count() > 3) {
        path3 = list[3].ImagePath;}
    if (list.Count() > 4) {
        path4 = list[4].ImagePath;}
    if (list.Count() > 5) {
        path5 = list[5].ImagePath;}

    sb.Append(" <div class='col-md-6 col-sm-6 col-sm-12'> \n");
    sb.Append(" <div class='images cd-post-thumbnail'> \n");
    sb.Append(" <div class='flexslider-wrap product-image-full-size'> \n");
    sb.Append(" <div class='flexslider' id='main_slider'> \n");
    sb.Append(" <div class='flex-viewport'> \n");
    sb.Append(" <ul class='slides'> \n");
    sb.Append(" <li itemtype='https://schema.org/Product' itemscope=''> \n");
    sb.Append(" <a class='cd-lightbox-image cd-lightbox-type-image' href='"+path+"' title='>
        \n");
    sb.Append(" <img title='' src='' + path + '' alt='' ><i itemprop='image' class='fa fa-fw fa-
    expand'></i> \n");
    sb.Append(" </a> \n");
    sb.Append(" </li> \n");
    sb.Append(" <li itemtype='https://schema.org/Product' itemscope=''> \n");
    sb.Append(" <a class='cd-lightbox-image cd-lightbox-type-image' href='"+path1+"' title='>
        \n");
    sb.Append(" <img title='' src='' + path1 + '' alt='' ><i itemprop='image' class='fa fa-fw fa-
    expand'></i> \n");
    sb.Append(" </a> \n");
    sb.Append(" </li> \n");
    sb.Append(" <li itemtype='https://schema.org/Product' itemscope=''> \n");
    sb.Append(" <a class='cd-lightbox-image cd-lightbox-type-image' href='"+path2+"' title='>
        \n");

```



```
sb.Append(" <i itemprop='image' class='fa fa-fw fa-  
expand'></i> \n");  
sb.Append(" </a> \n");  
sb.Append(" </li> \n");  
sb.Append(" <li itemprop='https://schema.org/Product' itemscope="> \n");  
sb.Append(" <a class='cd-lightbox-image cd-lightbox-type-image' href="+path3+" title=">  
 \n");  
sb.Append(" <i itemprop='image' class='fa fa-fw fa-  
expand'></i> \n");  
sb.Append(" </a> \n");  
sb.Append(" </li> \n");  
sb.Append(" <li itemprop='https://schema.org/Product' itemscope="> \n");  
sb.Append(" <a class='cd-lightbox-image cd-lightbox-type-image' href="+path4+" title=">  
 \n");  
sb.Append(" <i itemprop='image' class='fa fa-fw fa-  
expand'></i> \n");  
sb.Append(" </a> \n");  
sb.Append(" </li> \n");  
sb.Append(" <li itemprop='https://schema.org/Product' itemscope="> \n");  
sb.Append(" <a class='cd-lightbox-image cd-lightbox-type-image' href="+path5+" title=">  
 \n");  
sb.Append(" <i itemprop='image' class='fa fa-fw fa-  
expand'></i> \n");  
sb.Append(" </a> \n");  
sb.Append(" </li> \n");  
sb.Append(" </ul> \n");  
sb.Append(" </div> \n");  
sb.Append(" </div> \n");  
sb.Append(" </div> \n");  
sb.Append(" <div class='flexslider' id='main_thumbs'> \n");  
sb.Append(" <div class='flex-viewport'> \n");  
sb.Append(" <ul class='slides'> \n");  
sb.Append(" <li itemprop='https://schema.org/Product' itemscope="> \n");  
sb.Append("  \n");  
sb.Append(" </li> \n");  
sb.Append(" <li itemprop='https://schema.org/Product' itemscope="> \n");  
sb.Append("  \n");  
sb.Append(" </li> \n");  
sb.Append(" <li itemprop='https://schema.org/Product' itemscope="> \n");  
sb.Append("  \n");  
sb.Append(" </li> \n");  
sb.Append(" <li itemprop='https://schema.org/Product' itemscope="> \n");  
sb.Append("  \n");  
sb.Append(" </li> \n");  
sb.Append(" <li itemprop='https://schema.org/Product' itemscope="> \n");  
sb.Append("  \n");  
sb.Append(" </li> \n");  
sb.Append(" <li itemprop='https://schema.org/Product' itemscope="> \n");  
sb.Append("  \n");  
sb.Append(" </li> \n");  
sb.Append(" </ul> \n");  
sb.Append(" </div> \n");  
sb.Append(" </div> \n");  
sb.Append(" </div> \n");  
sb.Append(" </div> \n");
```

```

//sb.Append("      </div> \n");
string PreHtml = sb.ToString();
if (PreHtml.Length > 0)
{
    objlog.Trace_Error(4, "Gordersearchlist", "1", "SliderList", PreHtml);

    imglist.InnerHtml = PreHtml;
}

PreHtml = "";
var lst = new Products().GetProductSeller(psid, "");
List<EteproductSellerStocks> stk =new Products().GetSellerStocksbyPsId(psid);
var col = (from c in stk where c.Stock>0 orderby c.colid select c.colid ).Distinct();
var siz = (from d in stk where d.Stock>0 orderby d.EpOptId select d.EpOptId).Distinct();
sb = new StringBuilder();
sb.Append(" <div class='shop-product-heading'> \n");
sb.Append(" <h2>" + lst.SellerDescription + "</h2>      \n");
sb.Append(" </div> \n");
sb.Append(" <ul class='list-inline product-ratings mb30'> \n");
sb.Append(" <li><i class='rating-selected fa fa-star'></i></li>      \n");
sb.Append(" <li><i class='rating-selected fa fa-star'></i></li>      \n");
sb.Append(" <li><i class='rating-selected fa fa-star'></i></li>      \n");
sb.Append(" <li><i class='rating fa fa-star'></i></li>      \n");
sb.Append(" <li><i class='rating fa fa-star'></i></li>      \n");
sb.Append(" <li class='product-review-list'>      \n");
sb.Append(" <span>(1) <a href='#>Review</a> | <a href='#> Add Review</a></span>
      \n");
sb.Append(" </li> \n");
sb.Append(" </ul> \n");
sb.Append(" <p class='mb15'>" + lst.SellerDescription + "</p><br>      \n");
sb.Append(" <ul class='list-inline shop-product-prices mb30'> \n");
sb.Append(" <li class='shop-red'>+lst.Price+</li>      \n");
sb.Append(" <li class='line-through'>+lst.Price+</li>      \n");
sb.Append(" </ul> \n");
if (siz.Count() > 1)
{
    sb.Append("      <h3 class='shop-product-title'>Sizes</h3> \n");
    sb.Append(" <ul class='list-inline product-size mb30'> \n");
    string chk = "checked";
    foreach (var sz in siz)
    {
        string size = new Products().GetSizeByOptId(sz);
        sb.Append("      <li> \n");
        sb.Append("      <input type='radio' id='size-" + sz + "' name='size' "+chk+">
      \n");
        sb.Append("      <label for='size-"+sz+"'>"+size+</label> \n");

        sb.Append("      </li> \n");
        chk="";
    }
    sb.Append(" </ul> \n");
}
if (col.Count() > 1)
{
    sb.Append("      <h3 class='shop-product-title'>Colors</h3> \n");
}

```

```

sb.Append("      <ul class='list-inline product-color mb30'> \n");
foreach (var colr in col)
{
    Eteproduct cl = new Products().GetColorByColld(colr);

    sb.Append("      <li> \n");
    sb.Append("      <input type='Checkbox' id='color-" + cl.ProductId + "' name='color'> \n");
    sb.Append("      <img id='img" + cl.ProductId + "' src="" + cl.imageurl + "' />");
    //sb.Append("      <label class='color-" + cl.ProductName + "' for='color-" + cl.ProductId
+ "'>" + cl.ProductName + "</label> \n");

    sb.Append("      </li> \n");
}
sb.Append("      </ul> \n");
}
sb.Append(" <h3 class='shop-product-title'>Quantity</h3> \n");
sb.Append(" <div class='mb30'> \n");
sb.Append(" <button type='button' class='quantity-button' name='subtract'
onclick='javascript: document.getElementById('qty').value--;' value='- '></button> \n");
sb.Append(" <input type='text' class='quantity-field' name='qty' value='1' id='qty' /> \n");
sb.Append(" <button type='button' class='quantity-button' name='add' onclick='javascript:
document.getElementById('qty').value++;' value='+'></button> \n");
sb.Append(" <button type='button' class='btn btn-product' OnClick ='Button1_Click'>Add
to Cart</button> \n");
sb.Append(" </div> \n");
sb.Append(" <ul class='list-inline add-to-wishlist add-to-wishlist-brd'> \n");
sb.Append(" <li class='wishlist-in'>\n");
sb.Append(" <i class='fa fa-heart'></i> \n");
sb.Append(" <a href='MyWishList.aspx?psid="+psid+"'>Add to Wishlist</a> \n");
sb.Append(" </li> \n");
sb.Append(" <li class='compare-in'> \n");
sb.Append(" <i class='fa fa-exchange'></i> \n");
sb.Append(" <a href='MyCompareList.aspx?psid=" + psid + "'>Add to Compare</a> \n");
sb.Append(" </li> \n");
sb.Append(" </ul> \n");

```

```

    PreHtml = sb.ToString();
    if (PreHtml.Length > 0)
    {
        //objlog.Trace_Log(4, "Gordersearchlist", "1", "SliderList", PreHtml);

        shopdetail.InnerHtml = PreHtml;
    }
}

```

```

protected void Button1_Click(object sender, EventArgs e)
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];

    int psid = Convert.ToInt16(Request.QueryString["pid"]);
    EteUserCart cart = new EteUserCart();
    cart.Active = true;

```

```

    cart.UserId = Convert.ToInt16(Clsusr.UserId);
    cart.PSId = psid;
    cart.UId = Convert.ToInt16(Clsusr.UserId);
    cart.CreatedOn = DateTime.Today;
    cart.PayId = 0;
    bool ret = new Products().InsertCartList(cart);
    if (ret)
    {
        Response.Redirect("~/MyCart.aspx");
    }
}
protected void ImageButton2_Click(object sender, ImageClickEventArgs e)
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];

    int psid = Convert.ToInt16(Request.QueryString["pid"]);
    EteUserShortList lst = new EteUserShortList();
    lst.UserId = Convert.ToInt16(Clsusr.UserId);
    lst.PSId = psid;
    lst.UId = Convert.ToInt16(Clsusr.UserId);

    lst.CreatedOn = DateTime.Today;
    lst.Active = true;
    bool result = new Products().InsertShortList(lst, true);

}
}

```

***** ShoppingGrid.aspx*****

```

<%@ Page Title="" Language="C#" MasterPageFile="~/ShoppingMasterPage.master"
AutoEventWireup="true" CodeFile="ShoppingGrid.aspx.cs" Inherits="ShoppingGrid" %>

```

```

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
Runat="Server">
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentMainPage" Runat="Server">
</asp:Content>
<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" Runat="Server">
    <div class="col-md-9 col-sm-12 col-xs-12">
        <div class="row">
            <div id="dvMsgBox" class="span10 ordersave" runat="server"
style="display: block; height:100px">
                <asp:Label ID="Label1" runat="server" Text=""></asp:Label>
            </div>
        </div></div>
</asp:Content>

<asp:Content ID="Content5" ContentPlaceHolderID="CPH_MainContent" Runat="Server">
<!-- Content Bodey Here -->

    <!-- START MAIN CONTAIN -->
    <div class="main-contain shop-product bg-white">

```

```

<div class="container">
  <div class="row">
    <aside>
      <div id="sidebar" class="col-md-3 col-sm-12 col-xs-12 sidebar"
runat="server">
      <!-- CATEGORIES WIDGET -->
      <div id="Filter" class="widget mb30" runat="server">
        <h6 class="text-uppercase bottom-line">Show Results for</h6>
        <ul class="icons-list">
          <li><a href="#">Sarees <span class="pull-
right">12</span></a></li>
          <li><a href="#">Lehengas <span class="pull-
right">11</span></a></li>
          <li><a href="#">Indo Westers <span class="pull-
right">10</span></a></li>
          <li><a href="#">Salwar Kameez <span class="pull-
right">14</span></a></li>
          <li><a href="#">Kurtees <span class="pull-
right">15</span></a></li>
        </ul>
      </div>
      <!-- END CATEGORIES WIDGET -->

      <!-- SIDE-BAR PRODUCT WIDGET -->
      <div id="toprate" class="" runat="server">
        <div class="widget mb30">
          <h6 class="text-uppercase bottom-line">Top Rated</h6>
        </div>
      </div>
      <!-- END SIDE-BAR PRODUCT WIDGET -->

      <!-- SIDE-BAR PRODUCT WIDGET -->
      <div id="bestseller" class="" runat="server">
        <div class="widget mb30">
          <h6 class="text-uppercase bottom-line">Best Sellers</h6>
        </div>
        <div class="small-thumb-proudct mb30">
          <div class="thumb-product">
            
            <div class="thumb-product-inner">
              <h6><a href="shoppingdetail.aspx">Printed Su..</a></h6>
              <span class="thumb-product-cat">Fashion</span>
            </div>
            <ul class="list-inline overflow-h">
              <li class="thumb-product-price line-through">$95.00</li>
              <li class="thumb-product-price">$75.00</li>
            </ul>
          </div>
        </div>
        <div class="small-thumb-proudct mb30">
          <div class="thumb-product">
            
            <div class="thumb-product-inner">

```

```

        <h6><a href="shoppingdetail.aspx">Printed Su..</a></h6>
        <span class="thumb-product-cat">Fashion</span>
    </div>
    <ul class="list-inline overflow-h">
        <li class="thumb-product-price line-through">$70.00</li>
        <li class="thumb-product-price">$60.00</li>
    </ul>
</div>
</div>
<div class="small-thumb-proudct mb30">
    <div class="thumb-product">
        
        <div class="thumb-product-inner">
            <h6><a href="shoppingdetail.aspx">Printed Su..</a></h6>
            <span class="thumb-product-cat">Fashion</span>
        </div>
        <ul class="list-inline overflow-h">
            <li class="thumb-product-price">$65.00</li>
        </ul>
    </div>
</div>
</div>
</div>
<!-- END SIDE-BAR PRODUCT WIDGET -->

```

```

<!-- TAGS WIDGET -->
<div class="widget">
    <h6 class="text-uppercase bottom-line">Tags</h6>
    <div class="tags">
        <a href="#">Fashion</a>
        <a href="#">Electronics</a>
        <a href="#">Services</a>
        <a href="#">Businesses</a>
        <a href="#">Shops</a>
        <a href="#">Markets</a>
    </div>
</div>
<!-- END TAGS WIDGET -->
</div>
</aside>

```

```

<div class=" col-lg-9 col-md-9 col-sm-12 col-xs-12">
    <div class="row hidden-xs">
        <div class="toolbar">
            <div id="results" class="col-sm-4 category-num" runat="server">
                <h2>Designer</h2>
                <small class="category-num-coun">9 Results</small>
            </div>
        </div>--%>
        <div class="col-sm-8 pull-right">
            <ul class="view-mode">

```

```

        <li class="grid-list-icons">
<%--      <asp:Button ID="btn1" runat="server" OnClick="btn1_Click"
type="button" class="btn btn-floating color-green " Text="List" /><i class='fa fa-th-list'></i>
      <asp:Button ID="btn2" runat="server" OnClick="btn2_Click" type="button"
class="btn btn-floating color-green " Text="Grid" /><i class='fa fa-th'></i> wrn-btn--%>
      <asp:ImageButton ID="ImgBtn1" runat="server"
ImageUrl="images/small/list.png" OnClick="ImgBtn1_Click" />
      <asp:ImageButton ID="ImgBtn2" runat="server"
ImageUrl="images/small/grid.png" OnClick="ImgBtn2_Click" />
<%--      <button id="Button1" type="button" class="btn btn-secondary wrn-
btn mt-0 ml-0" onclick="btn1_Click" ><i class='fa fa-th-list'></i>&nbsp;&nbsp;&nbsp;List</button>
      <button id="Button2" type="button" style="width:10px" class="btn btn-
product " onclick="btn2_Click" ><i class='fa fa-th'></i></button>--%>
        </li>
        <li class="sort-list-btn">
          <h6>Sort By :</h6>
          <div class="btn-group">
            <button type="button" class="btn btn-default dropdown-toggle"
data-toggle="dropdown">
              Relevent <span class="caret"></span>
            </button>
            <ul class="dropdown-menu" role="menu">
              <li><a href="#">Relevent</a></li>
              <li><a href="#">Price High to Low</a></li>
              <li><a href="#">Price Low to High</a></li>
              <li><a href="#">New Arrived</a></li>
            </ul>
          </div>
        </li>
        <li class="sort-list-btn">
          <h6>Show :</h6>
          <div class="btn-group">
            <button type="button" class="btn btn-default dropdown-toggle"
data-toggle="dropdown">
              30 <span class="caret"></span>
            </button>
            <ul class="dropdown-menu" role="menu">
              <li><a href="#">30</a></li>
              <li><a href="#">45</a></li>
              <li><a href="#">60</a></li>
              <li><a href="#">90</a></li>
            </ul>
          </div>
        </li>
      </ul>
    </div>
  </div>
  <div class="col-md-9 col-sm-12 col-xs-12">
    <div class="row hidden-xs">
      <div class="toolbar">
        <div id="results" class="col-sm-8 category-num" runat="server">
          <h2>Designer</h2>

```

```

        <small class="category-num-coun">9 Results</small>
    </div>
</div>
</div>
</div>
</div>
<!--dynamic code from here-->
<div id="appvendors" class="category-products" runat="server" style="display:none">
<div>

</div>

</div>
<!--dynamic code ends here-->

<div id="AppGrid" class="category-products col-grid-style" runat="server"
style="display:block">
<div>

</div>

</div>
<!--dynamic code ends here-->

<!-- Start Pagination-->
<div class="cd-pagination"> <!--href="#"-->
<ul class="pagination">
<li><a href="#"><i class="fa fa-angle-left" aria-hidden="true"></i></a></li>
<li class="active"><a onclick="javascript:ShowPage('DivL1','DivG1')">1</a></li>
<li><a onclick="javascript:ShowPage('DivL2','DivG2')">2</a></li>
<li><a onclick="javascript:ShowPage('DivL3','DivG3')">3</a></li>
<li><a onclick="javascript:ShowPage('DivL4','DivG4')">4</a></li>
<li><a onclick="javascript:ShowPage('DivL5',DivG5')">5</a></li>
<li><a onclick="javascript:ShowPage('DivL6',Div6')"><i class="fa fa-angle-right"
aria-hidden="true"></i></a></li>
</ul>
</div>
</div>

<script type="text/javascript">
function ShowPage(Div,Div1)
{
    document.getElementById('DivL1').style.display = "none";
    document.getElementById('DivL2').style.display = "none";
    document.getElementById('DivL3').style.display = "none";
    document.getElementById('DivL4').style.display = "none";
    document.getElementById('DivL5').style.display = "none";
    document.getElementById('DivL6').style.display = "none";
    document.getElementById('DivG1').style.display = "none";
    document.getElementById('DivG2').style.display = "none";
    document.getElementById('DivG3').style.display = "none";
    document.getElementById('DivG4').style.display = "none";
}

```



```

document.getElementById('DivG5').style.display = "none";
document.getElementById('DivG6').style.display = "none";

document.getElementById(Div).style.display = "block";
document.getElementById(Div1).style.display = "block";

}
    </script>
    <!-- End Pagination-->
  </div>
  </div>

  <!-- END MAIN CONTAIN -->

  <!-- ./Content Bodey Here -->

```

```

</asp:Content>
<asp:Content ID="Content6" ContentPlaceHolderID="ContentFooter" Runat="Server">
</asp:Content>

```

```

*****ShoppingGrid.aspx.cs *****

```

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBAL;
using DowBEntity;
//using LogTracer;
using System.Text;
using System.Text.RegularExpressions;

```

```

public partial class ShoppingGrid: System.Web.UI.Page
{

```

```

    DDLogTracer objlog = new DDLogTracer();
    bool IsMobile = false;

```

```

    protected void Page_Load(object sender, EventArgs e)
    {

```

```

        CIsLoggedInUser CIsusr = (CIsLoggedInUser)Session["LoggedInUser_Specs"];
        if (CIsusr != null)
        {
            Bindetails();
        }
    }

```

```

}
private void Bindetails()
{

```

```

    CIsLoggedInUser CIsusr = (CIsLoggedInUser)Session["LoggedInUser_Specs"];
    string pt = "cat", title = "";
    ImgBtn1.Enabled = false;

```

```

ImgBtn2.Enabled = true;
appvndors.Attributes.CssStyle.Add("display", "block;");
AppGrid.Attributes.CssStyle.Add("display", "none;");
int srno = 1, perpage = 30, dvno = 1, sno = 1;
int pid = Convert.ToInt16(Request.QueryString["cid"]);
string PreHtml = "";
StringBuilder sb = new StringBuilder();
StringBuilder sbg = new StringBuilder();

sb.Append(" <div class='directory-list'>");
if (pid == 0)
{
    pid = Convert.ToInt16(Request.QueryString["pid"]);
    pt = "prod";
}

//try
{
    IList<ProductProductSeller> Lst = new List<ProductProductSeller>();
    if (pt == "prod")
    {
        Lst = (new Products()).GetProductSellerbyProductId(pid, "SearchList");
        if (Lst.Count() == 0 && pid != 0)
        {
            Eteproduct ep = new Products().GetProduct(pid, "");
            string pname = ep.ProductName;
            Lst = new Products().GetProductSellerbySearch(pname, 1);
        }
    }
    else
    {
        Lst = (new Products()).GetProductSellerbyCatId(pid, 1, "SearchList");
    }
    if (Lst.Count > 0)
    {
        ProductProductSeller pst = (from c in Lst select c).FirstOrDefault();

        if (pt == "prod")
        {
            title = pst.ProductName;
        }
        else
        {
            title = pst.CatName;
        }
        sb.Append("<h2>" + title + "</h2>\n <small class='category-num-coun'>" +
Convert.ToString(Lst.Count) + " Results</small>\n");
        results.InnerHtml = sb.ToString();
        sb = new StringBuilder();
        string path = "images/NoImage.jpg";
        string path1 = "images/NoImage.jpg";
        if (Lst != null && Lst.Count > 0)
        {
            sbg.Append("\n <div class='row clearfix'>\n");

```

```

sb.Append("\n <div class='row clearfix'>\n");

sb.Append("<div id='DivL" + dvno + "' class='products-list products-list-
description-brd mb30' style='display:block'>\n");

sbg.Append("<div id='DivG" + dvno + "' class='Row clearfix'
style='display:block'> \n");
sbg.Append(" <div class='row clearfix'>\n");
foreach (var dt in Lst)
{
    string href = "";
    href = "ShoppingDetail.aspx?id=" + dt.PSId;
    var list = new Products().GetProductSellerImage((int)dt.PSId, "").ToList();
    int cnt = list.Count();
    if (cnt > 1)
    {
        path = list[0].ImagePath;
        path1 = list[1].ImagePath; //"http://rk.bakhala.com/"+
    }
    else
        if (list.Count() > 0)
        {
            path = list[0].ImagePath;
        }

    if (path == null || path == "")
    {
        path = "images/NoImage.jpg";
    }
    if (path1 == null || path1 == "") { path1 = path; }
    List<EteproductSellerStocks> stk = new
Products().GetSellerStocksbyPsId((int)dt.PSId);
    var col = (from c in stk where c.Stock > 0 orderby c.colid select
c.colid).Distinct();
    var siz = (from d in stk where d.Stock > 0 orderby d.EpOptId select
d.EpOptId).Distinct();

    //list code
sb.Append(" <div class='products-list products-list-description-brd mb30'>
\n");
sb.Append(" <div class='row'> \n");
sb.Append(" <div class='col-sm-4'> \n");
sb.Append(" <div class='product-grid-img'> \n");
sb.Append(" <ul class='product-labels'> \n");
sb.Append(" <li>Sale</li> \n");
sb.Append(" </ul> \n");
sb.Append(" <a href='shoppingdetail.aspx?pid=" + dt.PSId + "'
class='product-grid-img-thumbnail'> \n");
sb.Append("  \n");
sb.Append("  \n");
sb.Append(" </a> \n");

```

```

        sb.Append(" <div class='product-grid-atb text-center'><i class='fa fa-
shopping-cart'></i><a class='" href='shoppingdetail.aspx?pid=" + dt.PSId + "'>Add to
Cart</a></div> \n");
        sb.Append(" </div> \n");
        sb.Append(" </div> \n");
        sb.Append(" <div class='col-sm-8 products-list-description'> \n");
        sb.Append(" <div class='overflow-h'> \n");
        sb.Append(" <ul class='list-inline overflow-h mb15'> \n");
        sb.Append(" <li><h4 class='title-price'><a href='shoppingdetail.aspx?pid="
+ dt.PSId + "'>" + dt.ProductDescription + "</a></h4></li> \n");
        sb.Append(" <li class='pull-right'> \n");
        sb.Append(" <ul class='list-inline product-ratings'> \n");
        sb.Append(" <li><i class='rating-selected fa fa-star'></i></li> \n");
        sb.Append(" <li><i class='rating-selected fa fa-star'></i></li> \n");
        sb.Append(" <li><i class='rating-selected fa fa-star'></i></li> \n");
        sb.Append(" <li><i class='rating fa fa-star'></i></li> \n");
        sb.Append(" <li><i class='rating fa fa-star'></i></li> \n");
        sb.Append(" </ul> \n");
        sb.Append(" </li> \n");
        sb.Append(" </ul> \n");
        sb.Append(" <div class='mb10'> \n");
        sb.Append(" <span class='title-price'>" + dt.Price.ToString() + "</span>
\n");
        sb.Append(" <span class='title-price line-through'>" + dt.Price.ToString() +
\n");
        sb.Append(" </span> \n");
        sb.Append(" </div> \n");
        sb.Append(" <p class='mb15'>" + dt.SellerDescription + "</p> \n");

//colors and sizes
if (siz.Count() > 1)
{
        sb.Append(" <h3 class='shop-product-title'>Sizes Available</h3>
\n");
        sb.Append(" <ul class='list-inline product-size mb30'> \n");
        string chk = "checked";

        foreach (var sz in siz)
        {
                string size = new Products().GetSizeByOptIId(sz);
                sb.Append(" <li> \n");
                sb.Append(" <input type='checkbox' id='size-" + sz + "' name='size'
" + chk + "> \n");
                sb.Append(" <label for='size-" + dt.PSId + sz + "'>" + size +
</label> \n");

                sb.Append(" </li> \n");
                chk = "";
        }
        sb.Append(" </ul> \n");
}
if (col.Count() > 1)
{
        sb.Append(" <h3 class='shop-product-title'>Colors Available</h3>
\n");
        sb.Append(" <ul class='list-inline product-color mb30'> \n");

```

```

        sb.Append("        <li> \n");
        foreach (var colr in col)
        {
            Eteproduct cl = new Products().GetColorByColld(colr);
            sb.Append("        <li> \n");
            sb.Append("        <input type='radio' id='color-" + cl.ProductId + "
name='color'> \n");
            sb.Append(" </li>");
        }
        sb.Append("        </ul> \n");
    }
    //colors and sizes ends
    sb.Append(" <ul class='list-inline add-to-wishlist mb15'> \n");
    sb.Append(" <li class='product-wishlist'> \n");
    sb.Append(" <i class='fa fa-heart'></i> \n");
    sb.Append(" <a href='MyWishList.aspx?psid=" + dt.PSId + "'>Add to
Wishlist</a> \n");
    sb.Append(" </li> \n");
    sb.Append(" <li class='product-wishlist'> \n");
    sb.Append(" <i class='fa fa-exchange'></i> \n");
    //sb.Append(" <a href='shoppingdetail.aspx?pid=" + dt.PSId + "'>Add to
Compare</a> \n");
    sb.Append(" </li> \n");
    sb.Append(" </ul> \n"); sb.Append("        <button type='button' class='btn
btn-product'>Add to Cart</button> \n");
    sb.Append(" </div> \n");
    sb.Append(" </div> \n");
    sb.Append(" </div> \n");
    sb.Append(" </div> \n");
    sb.Append(" \n");

    //grid code
    sbg.Append(" <div class='col-md-4 col-sm-6 col-xs-12'> \n");
    sbg.Append(" <div class='products grid-product mb30'> \n");
    sbg.Append(" <div class='product-grid-inner'> \n");
    sbg.Append(" <div class='product-grid-img'> \n");
    sbg.Append(" <ul class='product-labels'> \n");
    sbg.Append(" <li>Sale</li> \n");
    sbg.Append(" </ul> \n");
    sbg.Append(" <a href='shoppingdetail.aspx?pid=" + dt.PSId + "
class='product-grid-img-thumbnail'> \n");
    sbg.Append("  \n");
    sbg.Append("  \n");
    sbg.Append(" </a> \n");
    sbg.Append(" <div class='product-grid-atb text-center'><a class="
href='shoppingdetail.aspx?pid=" + dt.PSId + "'>Quick View</a></div> \n");
    sbg.Append(" </div> \n");
    sbg.Append(" <div class='product-grid-caption text-center'> \n");
    sbg.Append(" <h6><a class='p-grid-title' href='shoppingdetail.aspx?pid=" +
dt.PSId + "'>" + dt.ProductDescription + "</a></h6> \n");
    sbg.Append(" <p>" + dt.SellerDescription + "</p> \n");
    //colors and sizes

```

```

        if (siz.Count() > 1)
        {
            sbg.Append("      <h3 class='shop-product-title'>Sizes Available</h3>
\n");
            sbg.Append("      <ul class='list-inline product-size mb30'> \n");
            string chk = "checked";

            foreach (var sz in siz)
            {
                string size = new Products().GetSizeByOptlId(sz);
                sbg.Append("      <li> \n");
                sbg.Append("      <input type='radio' id='size-" + sz + "' name='size' " +
chk + "> \n");
                sbg.Append("      <label for='size-" + sz + "'> " + size + "</label>

\n");

                sbg.Append("      </li> \n");
                chk = "";
            }
            sbg.Append("      </ul> \n");
        }
        if (col.Count() > 1)
        {
            sbg.Append("      <h3 class='shop-product-title'>Colors Available</h3>
\n");
            sbg.Append("      <ul class='list-inline product-color mb30'> \n");
            sbg.Append("      <li> \n");
            foreach (var colr in col)
            {
                Eteproduct cl = new Products().GetColorByColId(colr);
                sbg.Append("      <li> \n");
                sbg.Append("      <input type='radio' id='color-" + cl.ProductId + "
name='color'> \n");
                sbg.Append("      <img id='img" + cl.ProductId + "' src="" + cl.imageurl + "
/></li>");
            }
            sbg.Append("      </ul> \n");
        }
        //colors and sizes ends
        sbg.Append(" <span class='price clearfix'><span class='old-price'>" +
dt.Price.ToString() + "</span> $ <span class='special-price'>" + dt.Price.ToString() +
"</span></span> \n");
        sbg.Append(" <a class='btn btn-product cd-cart-btn'
href='MyCart.aspx?psid=" + dt.PSId + "'><i class='fa fa-shopping-cart'></i> Add to Cart</a>
\n");
        sbg.Append(" </div> \n");
        sbg.Append(" </div> \n");
        sbg.Append(" </div> \n");
        sbg.Append(" </div> \n");
        srno++;
        sno++;
        if (sno > 3)
        {
            sbg.Append("      </div> \n");
            sbg.Append(" <div class='row clearfix'>\n");

```

```

        sno = 1;
    }
    if (srno > perpage)
    {
        dvno++;
        srno = 1;
        sb.Append("        </div> \n");
        sbg.Append("        </div> \n");
        sb.Append("<div id='DivL' + dvno + '" class='products-list products-list-
description-brd mb30' style='display:block'>\n");

        sbg.Append("<div id='DivG' + dvno + '" class='Row clearfix'
style='display:block'> \n");

    }

}

sb.Append("        </div> \n");
sb.Append("        </div> \n");
sbg.Append("        </div> \n");
sbg.Append("        </div> \n");
sbg.Append("        </div> \n");

PreHtml = sb.ToString();
if (PreHtml.Length > 0)
{
    objlog.Trace_Error(4, "ShoppingList", "1", "appvendors", PreHtml,
Session["Logpath"].ToString());
    appvendors.InnerHtml = PreHtml;
}
PreHtml = sbg.ToString();
if (PreHtml.Length > 0)
{
    objlog.Trace_Log(4, "ShoppingList", "1", "AppGrid", PreHtml,
Session["Logpath"].ToString());
    AppGrid.InnerHtml = PreHtml;
}

}

sb = new StringBuilder();
int grp = pst.ProductGroupId;
List<Eteproduct> prd = new Products().GetProductbyGroup(grp, "");
if (prd.Count > 0)
{
    sb.Append("<h6 class='text-uppercase bottom-line'>Show Results for</h6>\n");
    sb.Append("<ul class='icons-list'>\n");
    foreach (var data in prd)
    {
        if (Convert.ToInt16(data.Type) > 0)
        {

```

```

                sb.Append("<li><a href='ShoppingList.aspx?pgid=" + data.ProductId + ">"
+ data.ProductName + " <span class='pull-right'>" + data.Type + "</span></a></li>\n");
            }
        }
        sb.Append("</ul>\n");
        Filter.InnerHtml = sb.ToString();
    }
    sb = new StringBuilder();
    sb.Append("        <div class='widget mb30'> ");
    sb.Append("        <h6 class='text-uppercase bottom-line'>Top Rated</h6> ");
    sb.Append("        </div> ");
    List<ProductProductSeller> top = new Products().GetProductSellerbySearch("",
1);
    foreach (var tr in top)
    {
        sb.Append("        <div class='small-thumb-proudct mb30'> ");
        sb.Append("        <div class='thumb-product'> ");
        sb.Append("         ");
        //sb.Append("        <div class='thumb-product-inner'> ");
        sb.Append("        <span class='thumb-product-cat'>" + tr.CatName + "</span>
");
        sb.Append("        <h6><a href='shoppingdetail.aspx?pid=" + tr.PSId + ">" +
tr.ProductDescription + "</a></h6> ");
        sb.Append("        </div> ");
        sb.Append("        <ul class='list-inline overflow-h'> ");
        sb.Append("        <li class='thumb-product-price line-through'>" + tr.CurPrice +
"</li> ");
        sb.Append("        <li class='thumb-product-price'>" + tr.CurPrice + "</li> ");
        sb.Append("        </ul> ");
        //sb.Append("        </div> ");
        sb.Append("        </div> ");
    }
    topate.InnerHtml = sb.ToString();
    sb = new StringBuilder();
    sb.Append("        <div class='widget mb30'> ");
    sb.Append("        <h6 class='text-uppercase bottom-line'>Best Sellers</h6> ");
    sb.Append("        </div> ");

    //List<ProductProductSeller> best = new Products().GetProductSellerbySearch("");
    foreach (var tr in top)
    {
        sb.Append("        <div class='small-thumb-proudct mb30'> ");
        sb.Append("        <div class='thumb-product'> ");
        sb.Append("         ");
        //sb.Append("        <div class='thumb-product-inner'> ");
        sb.Append("        <span class='thumb-product-cat'>" + tr.CatName + "</span>
");
        sb.Append("        <h6><a href='shoppingdetail.aspx?pid=" + tr.PSId + ">" +
tr.ProductDescription + "</a></h6> ");
        sb.Append("        </div> ");
        sb.Append("        <ul class='list-inline overflow-h'> ");
        sb.Append("        <li class='thumb-product-price line-through'>" + tr.CurPrice +
"</li> ");
    }

```



```

        sb.Append("    <li class='thumb-product-price'>" + tr.CurPrice + "</li>    ");
        sb.Append("    </ul> ");
        //sb.Append(" </div> ");
        sb.Append(" </div> ");

    }
    bestseller.InnerHtml = sb.ToString();
    sb = new StringBuilder();
}
}
// catch (Exception ex)
//{

// Common.ShowMessageHtml(ex.Message, 1, true, this.dvMsgBox);
// ExceptionLogging.SendExcepToDB(ex);

//}
}

protected void btn1_Click(object sender, EventArgs e)
{
    appvendors.Attributes.CssStyle.Add("display", "block;");
    AppGrid.Attributes.CssStyle.Add("display", "none;");

}
protected void btn2_Click(object sender, EventArgs e)
{
    appvendors.Attributes.CssStyle.Add("display", "none;");
    AppGrid.Attributes.CssStyle.Add("display", "block;");

}
protected void ImgBtn1_Click(object sender, ImageClickEventArgs e)
{
    appvendors.Attributes.CssStyle.Add("display", "block;");
    AppGrid.Attributes.CssStyle.Add("display", "none;");
    ImgBtn1.Enabled = false;
    ImgBtn2.Enabled = true;

}
protected void ImgBtn2_Click(object sender, ImageClickEventArgs e)
{
    appvendors.Attributes.CssStyle.Add("display", "none;");
    AppGrid.Attributes.CssStyle.Add("display", "block;");
    ImgBtn1.Enabled = true;
    ImgBtn2.Enabled = false;

}
}
*****ShoppingHome.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/ShoppingMasterPage.master"
AutoEventWireup="true" CodeFile="ShoppingHome.aspx.cs" Inherits="ShoppingHome" %>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
Runat="Server">
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentMainPage" Runat="Server">

```

```
<!-- Main layout -->
```

```
<div class="container-fluid">
```

```
<!-- Section: Intro -->
```

```
<section class="mt-md-4 pt-md-2 mb-5 pb-4">
```

```
<!-- Grid row -->
```

```
<div class="row">
```

```
<!-- Grid column -->
```

```
<div class="col-xl-3 col-md-6 mb-xl-0 mb-4">
```

```
<!-- Card -->
```

```
<div class="card card-cascade cascading-admin-card">
```

```
<!-- Card Data -->
```

```
<div class="admin-up">
```

```
<i class="far fa-money-bill-alt primary-color mr-3 z-depth-2"></i>
```

```
<div class="data">
```

```
<p class="text-uppercase">sales</p>
```

```
<h4 class="font-weight-bold dark-grey-text">2000$</h4>
```

```
</div>
```

```
</div>
```

```
<!-- Card content -->
```

```
<div class="card-body card-body-cascade">
```

```
<div class="progress mb-3">
```

```
<div class="progress-bar bg-primary" role="progressbar" style="width: 25%" aria-  
valuenow="25" aria-valuemin="0" aria-valuemax="100"></div>
```

```
</div>
```

```
<p class="card-text">Better than last week (25%)</p>
```

```
</div>
```

```
</div>
```

```
<!-- Card -->
```

```
</div>
```

```
<!-- Grid column -->
```

```
<!-- Grid column -->
```

```
<div class="col-xl-3 col-md-6 mb-xl-0 mb-4">
```

```
<!-- Card -->
```

```
<div class="card card-cascade cascading-admin-card">
```

```
<!-- Card Data -->
```

```
<div class="admin-up">
```

```
<i class="fas fa-chart-line warning-color mr-3 z-depth-2"></i>
```

```
<div class="data">
```

```
<p class="text-uppercase">subscriptions</p>
```

```
<h4 class="font-weight-bold dark-grey-text">200</h4>
```

```
</div>
```

```
</div>
```

```

    <!-- Card content -->
    <div class="card-body card-body-cascade">
      <div class="progress mb-3">
        <div class="progress-bar red accent-2" role="progressbar" style="width: 25%"
aria-valuenow="25" aria-valuemin="0" aria-valuemax="100"></div>
      </div>
      <p class="card-text">Worse than last week (25%)</p>
    </div>

  <!-- Card -->

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-xl-3 col-md-6 mb-md-0 mb-4">

  <!-- Card -->
  <div class="card card-cascade cascading-admin-card">

    <!-- Card Data -->
    <div class="admin-up">
      <i class="fas fa-chart-pie light-blue lighten-1 mr-3 z-depth-2"></i>
      <div class="data">
        <p class="text-uppercase">traffic</p>
        <h4 class="font-weight-bold dark-grey-text">20000</h4>
      </div>
    </div>

    <!-- Card content -->
    <div class="card-body card-body-cascade">
      <div class="progress mb-3">
        <div class="progress-bar red accent-2" role="progressbar" style="width: 75%"
aria-valuenow="75" aria-valuemin="0" aria-valuemax="100"></div>
      </div>
      <p class="card-text">Worse than last week (75%)</p>
    </div>

  </div>
  <!-- Card -->

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-xl-3 col-md-6 mb-0">

  <!-- Card -->
  <div class="card card-cascade cascading-admin-card">

    <!-- Card Data -->
    <div class="admin-up">
      <i class="fas fa-chart-bar red accent-2 mr-3 z-depth-2"></i>
      <div class="data">

```

```

    <p class="text-uppercase">organic traffic</p>
    <h4 class="font-weight-bold dark-grey-text">2000</h4>
  </div>
</div>

<!-- Card content -->
<div class="card-body card-body-cascade">
  <div class="progress mb-3">
    <div class="progress-bar bg-primary" role="progressbar" style="width: 25%" aria-
valuenow="25" aria-valuemin="0" aria-valuemax="100"></div>
  </div>
  <p class="card-text">Better than last week (25%)</p>
</div>

</div>
<!-- Card -->

</div>
<!-- Grid column -->

</div>
<!-- Grid row -->

</section>
<!-- Section: Intro -->

<!-- Section: Main panel -->
<section class="mb-5">

  <!-- Card -->
  <div class="card card-cascade narrower">

    <!-- Section: Chart -->
    <section>

      <!-- Grid row -->
      <div class="row">

        <!-- Grid column -->
        <div class="col-xl-5 col-lg-12 mr-0 pb-2">

          <!-- Card image -->
          <div class="view view-cascade gradient-card-header light-blue lighten-1">
            <h2 class="h2-responsive mb-0 font-weight-500">Purchased</h2>
          </div>

          <!-- Card content -->
          <div class="card-body card-body-cascade pb-0">

            <!-- Panel data -->
            <div class="row py-3 pl-4">

              <!-- First column -->
              <div class="col-md-6">

```

```

<!-- Date select -->
<p class="lead"><span class="badge info-color p-2">Data range</span></p>
<select class="mdb-select colorful-select dropdown-info md-form">
  <option value="" disabled selected>Choose time period</option>
  <option value="1">Today</option>
  <option value="2">Yesterday</option>
  <option value="3">Last 7 days</option>
  <option value="3">Last 30 days</option>
  <option value="3">Last week</option>
  <option value="3">Last month</option>
</select>

<!-- Date pickers -->
<p class="lead pt-3 pb-4"><span class="badge info-color p-2">Custom
date</span></p>
<div class="md-form">
  <input placeholder="Selected date" type="text" id="from" class="form-control
datepicker">
  <label for="date-picker-example">From</label>
</div>
<div class="md-form">
  <input placeholder="Selected date" type="text" id="to" class="form-control
datepicker">
  <label for="date-picker-example">To</label>
</div>
</div>
<!-- First column -->

<!-- Second column -->
<div class="col-md-6 text-center pl-xl-2 my-md-0 my-3">

  <!-- Summary -->
  <p>Total sales: <strong>2000$</strong>
  <button type="button" class="btn btn-info btn-sm p-2" data-toggle="tooltip"
data-placement="top"
  title="Total sales in the given period"><i class="fas fa-
question"></i></button>
  </p>
  <p>Average sales: <strong>100$</strong>
  <button type="button" class="btn btn-info btn-sm p-2 mr-0" data-
toggle="tooltip" data-placement="top"
  title="Average daily sales in the given period"><i class="fas fa-
question"></i></button>
  </p>

  <!-- Change chart -->
  <span class="min-chart my-4" id="chart-sales" data-percent="76"><span
class="percent"></span></span>
  <h5>
  <span class="badge red accent-2 p-2">Change <i class="fas fa-arrow-circle-
up ml-1"></i></span>
  <button type="button" class="btn btn-info btn-sm p-2" data-toggle="tooltip"
data-placement="top"

```

```

        title="Percentage change compared to the same period in the past"><i
class="fas fa-question"></i>
      </button>
    </h5>

  </div>
  <!-- Second column -->

</div>
<!-- Panel data -->

</div>
<!-- Card content -->

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-xl-7 col-lg-12 mb-4 pb-2">

  <!-- Chart -->
  <div class="view view-cascade gradient-card-header blue-gradient">

    <canvas id="lineChart" height="175"></canvas>

  </div>

</div>
<!-- Grid column -->

</div>
<!-- Grid row -->

</section>
<!-- Section: Chart -->

<!-- Section: Table -->
<section>

  <!-- Top Table UI -->
  <div class="table-ui p-2 mb-3 mx-4 mb-5">

    <!-- Grid row -->
    <div class="row">

      <!-- Grid column -->
      <div class="col-xl-3 col-md-6">

        <!-- Name -->
        <select class="mdb-select colorful-select dropdown-info mx-2 md-form mt-3 md-
dropdown">
          <option value="" disabled selected>Bulk actions</option>
          <option value="1">Delate</option>
          <option value="2">Export</option>
          <option value="3">Change segment</option>

```

```

    </select>

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-xl-3 col-md-6">

    <!-- Blue select -->
    <select class="mdb-select colorful-select dropdown-info mx-2 md-form mt-3 md-
dropdown">
        <option value="" disabled selected>Show only</option>
        <option value="1">All <span> (2000)</span></option>
        <option value="2">Never opened <span> (200)</span></option>
        <option value="3">Opened but unanswered <span> (1800)</span></option>
        <option value="4">Answered <span> (200)</span></option>
        <option value="5">Unsubscribed <span> (50)</span></option>
    </select>

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-xl-3 col-md-6">

    <!-- Blue select -->
    <select class="mdb-select colorful-select dropdown-info mx-2 md-form mt-3 md-
dropdown">
        <option value="" disabled selected>Filter segments</option>
        <option value="1">Contacts in no segments <span> (100)</span></option>
        <option value="1">Segment 1 <span> (2000)</span></option>
        <option value="2">Segment 2 <span> (1000)</span></option>
        <option value="3">Segment 3 <span> (4000)</span></option>
    </select>

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-xl-3 col-md-6">

    <div class="form-inline mt-2 ml-2">
        <div class="form-group md-form mt-2">
            <input class="form-control w-100" type="text" placeholder="Search">
        </div>
        <button class="btn btn-sm btn-primary ml-2 mr-0 mb-md-0 mb-4 px-2"><i
class="fas fa-search"></i></button>
    </div>

</div>
<!-- Grid column -->

</div>
<!-- Grid row -->

```

```

        </div>
        <!-- Top Table UI -->

    </section>
    <!--Section: Table-->

    </div>
    <!-- Card -->

</section>
<!-- Section: Main panel -->

</div>

    <!-- Main layout --></asp:Content>
<asp:Content ID="Content3" ContentPlaceHolderID="ContentPlaceHolder1"
Runat="Server">
</asp:Content>
*****ShoppingHome.aspx.cs *****
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

public partial class ShoppingHome : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {

    }
}
***** ShoppingList.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/ShoppingMasterPage.master"
AutoEventWireup="true" CodeFile="ShoppingList.aspx.cs" Inherits="ShoppingList" %>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
Runat="Server">
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentMainPage" Runat="Server">
</asp:Content>
<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" Runat="Server">
    <div class="col-md-9 col-sm-12 col-xs-12">
        <div class="row">
            <div id="dvMsgBox" class="span10 ordersave" runat="server"
style="display: block; height:100px">
                <asp:Label ID="Label1" runat="server" Text=""></asp:Label>
            </div>
        </div>
    </div></div>

</asp:Content>

<asp:Content ID="Content5" ContentPlaceHolderID="CPH_MainContent" Runat="Server">
<!-- Content Bodey Here -->

```



```

<!-- START MAIN CONTAIN -->
<div class="main-contain shop-product bg-white">
  <div class="container">
    <div class="row">
      <aside>
        <div id="sidebar" class="col-md-3 col-sm-12 col-xs-12 sidebar"
runat="server">
          <!-- CATEGORIES WIDGET -->
          <div id="Filter" class="widget mb30" runat="server">
            <h6 class="text-uppercase bottom-line">Show Results for</h6>
            <ul class="icons-list">
              <li><a href="#">Sarees <span class="pull-
right">12</span></a></li>
              <li><a href="#">Lehengas <span class="pull-
right">11</span></a></li>
              <li><a href="#">Indo Westers <span class="pull-
right">10</span></a></li>
              <li><a href="#">Salwar Kameez <span class="pull-
right">14</span></a></li>
              <li><a href="#">Kurtees <span class="pull-
right">15</span></a></li>
            </ul>
          </div>
          <!-- END CATEGORIES WIDGET -->

          <!-- SIDE-BAR PRODUCT WIDGET -->
          <div id="toprate" class="" runat="server">
            <div class="widget mb30">
              <h6 class="text-uppercase bottom-line">Top Rated</h6>
            </div>
          </div>
          <!-- END SIDE-BAR PRODUCT WIDGET -->

          <!-- SIDE-BAR PRODUCT WIDGET -->
          <div id="bestseller" class="" runat="server">
            <div class="widget mb30">
              <h6 class="text-uppercase bottom-line">Best Sellers</h6>
            </div>
            <div class="small-thumb-proudct mb30">
              <div class="thumb-product">
                
                <div class="thumb-product-inner">
                  <h6><a href="shoppingdetail.aspx">Printed Su..</a></h6>
                  <span class="thumb-product-cat">Fashion</span>
                </div>
                <ul class="list-inline overflow-h">
                  <li class="thumb-product-price line-through">$95.00</li>
                  <li class="thumb-product-price">$75.00</li>
                </ul>
              </div>
            </div>
            <div class="small-thumb-proudct mb30">
              <div class="thumb-product">

```

```

        
        <div class="thumb-product-inner">
            <h6><a href="shoppingdetail.aspx">Printed Su..</a></h6>
            <span class="thumb-product-cat">Fashion</span>
        </div>
        <ul class="list-inline overflow-h">
            <li class="thumb-product-price line-through">$70.00</li>
            <li class="thumb-product-price">$60.00</li>
        </ul>
    </div>
</div>
<div class="small-thumb-proudct mb30">
    <div class="thumb-product">
        
        <div class="thumb-product-inner">
            <h6><a href="shoppingdetail.aspx">Printed Su..</a></h6>
            <span class="thumb-product-cat">Fashion</span>
        </div>
        <ul class="list-inline overflow-h">
            <li class="thumb-product-price">$65.00</li>
        </ul>
    </div>
</div>
</div>

```

```
<!-- END SIDE-BAR PRODUCT WIDGET -->
```

```
<!-- TAGS WIDGET -->
```

```

<div class="widget">
    <h6 class="text-uppercase bottom-line">Tags</h6>
    <div class="tags">
        <a href="#">Fashion</a>
        <a href="#">Electronics</a>
        <a href="#">Services</a>
        <a href="#">Businesses</a>
        <a href="#">Shops</a>
        <a href="#">Markets</a>
    </div>
</div>
<!-- END TAGS WIDGET -->

```

```

</div>
</aside>

```

```
<div class=" col-lg-9 col-md-9 col-sm-12 col-xs-12">
```

```

        <div class="row hidden-xs">
<%--
        <div class="toolbar">
        <div id="results" class="col-sm-4 category-num" runat="server">
            <h2>Designer</h2>
            <small class="category-num-coun">9 Results</small>
        </div>

```

```

</div>--%>
<div class="col-sm-8 pull-right">
  <ul class="view-mode">
    <li class="grid-list-icons">
      <asp:ImageButton ID="ImgBtn1" runat="server"
ImageUrl="images/small/list.png" OnClick="ImgBtn1_Click" />
      <asp:ImageButton ID="ImgBtn2" runat="server"
ImageUrl="images/small/grid.png" OnClick="ImgBtn2_Click" />
    </li>
    <li class="sort-list-btn">
      <h6>Sort By :</h6>
      <div id="sortbtn" class="btn-group" runat="server">
        <button type="button" class="btn btn-default dropdown-toggle"
data-toggle="dropdown">
          Relevent <span class="caret"></span>
        </button>
        <ul class="dropdown-menu" role="menu">
          <li><a href="#">Relevent</a></li>
          <li><a href="#">Price High to Low</a></li>
          <li><a href="#">Price Low to High</a></li>
          <li><a href="#">New Arrived</a></li>
        </ul>
      </div>
    </li>
    <li class="sort-list-btn">
      <h6>Show :</h6>
      <div id="pagenos" class="btn-group" runat="server">
        <button type="button" class="btn btn-default dropdown-toggle"
data-toggle="dropdown">
          30 <span class="caret"></span>
        </button>
        <ul class="dropdown-menu" role="menu" runat="server">
          <li><a href="#">30</a></li>
          <li><a href="#">45</a></li>
          <li><a href="#">60</a></li>
          <li><a href="#">90</a></li>
        </ul>
      </div>
    </li>
  </ul>
</div>

</div>
<div class="col-md-9 col-sm-12 col-xs-12">
  <div class="row hidden-xs">
    <div class="toolbar">
      <div id="results" class="col-sm-8 category-num" runat="server">
        <h2>Designer</h2>
        <small class="category-num-coun">9 Results</small>
      </div>
    </div>
  </div>
</div>
</div>
</div>

```

```
<%--dynamic code from here--%>
  <div id="appvendors" class="category-products col-grid-style slick-slider" runat="server"
style="display:block">
<div>
```

```
</div>
```

```
</div>
```

```
<%--dynamic code ends here--%>
```

```
<div id="AppGrid" class="category-products col-grid-style slider" runat="server"
style="display:none">
<div>
```

```
<div class='row clearfix'>
<div id='DivG1' class='Row clearfix' style='display:block'>
<div class='row clearfix'>
  <div class='col-md-4 col-sm-6 col-xs-12'>
    <div class='products grid-product mb30'>
      <div class='product-grid-inner'>
        <div class='product-grid-img'>
          <ul class='product-labels'>
            <li>Sale</li>
          </ul>
          <a href='shoppingdetail.aspx?pid=1' class='product-grid-img-thumbnail'>
            <img src='images/vendor/1-4-1.png' alt='img' class='img-responsive primary-img' />
            <img src='images/vendor/1-4-2.png' alt='img' class='img-responsive socendary-img'
/>
          </a>
          <div class='product-grid-atb text-center'><a class="
href='shoppingdetail.aspx?pid=1'>Quick View</a></div>
          </div>
          <div class='product-grid-caption text-center'>
            <h6><a class='p-grid-title' href='shoppingdetail.aspx?pid=1'>banaras silk saree teal
green with red golden border</a></h6>
            <p>banaras silk saree teal green with red golden border</p>
          <div><li><a href='ShoppingList.aspx?vendor=6'>Sold by : <span class='btn btn-default
special-price'>Vendor</span></li> </div>
          <div><li><a href='ShoppingList.aspx?brand=Brand'> Brand : <span class='fa glyphicon-bold
special-price'>Brand</span></li></div>
            <span class='price clearfix'><span class='old-price'>₹12344.00</span> ₹ <span
class='special-price'>12344.00</span></span>
            <a class='btn btn-product cd-cart-btn' href='MyCart.aspx?psid=1'><i class='fa fa-
shopping-cart'></i> Add to Cart</a>
          </div>
        </div>
      </div>
    </div>
  </div>
</div>
</div>
```

```

<div class='products grid-product mb30'>
<div class='product-grid-inner'>
<div class='product-grid-img'>
<ul class='product-labels'>
<li>Sale</li>
</ul>
<a href='shoppingdetail.aspx?pid=3' class='product-grid-img-thumbnail'>
<img src='images/vendor/3-1-1.png' alt='img' class='img-responsive primary-img' />

<img src='images/vendor/3-1-2.png' alt='img' class='img-responsive socendary-img'
/>
</a>
<div class='product-grid-atb text-center'><a class=""
href='shoppingdetail.aspx?pid=3'>Quick View</a></div>
</div>
<div class='product-grid-caption text-center'>
<h6><a class='p-grid-title' href='shoppingdetail.aspx?pid=3'>SALWAR KAMEEZ
BRIDES WEDDING GREEN AND RED</a></h6>
<p>SALWAR KAMEEZ BRIDES WEDDING GREEN AND RED</p>
<h3 class='shop-product-title'>Sizes Available</h3>
<ul class='list-inline product-size mb30'>
<li>
<input type='radio' id='size-34' name='sizeG3' checked >
<label for='sizeG-34'>M</label>
</li>
<li>
<input type='radio' id='size-35' name='sizeG3' >
<label for='sizeG-35'>L</label>
</li>
</ul>
<div><li><a href='ShoppingList.aspx?vendor=6'>Sold by : <span class='btn btn-default
special-price'>Vendor</span></li> </div>
<div><li><a href='ShoppingList.aspx?brand=Brand'> Brand : <span class='fa glyphicon-bold
special-price'>Brand</span></li></div>
<span class='price clearfix'><span class='old-price'>₹1234.00</span> ₹ <span
class='special-price'>1234.00</span></span>
<a class='btn btn-product cd-cart-btn' href='MyCart.aspx?psid=3'><i class='fa fa-
shopping-cart'></i> Add to Cart</a>
</div>
</div>
</div>
</div>
<div class='col-md-4 col-sm-6 col-xs-12'>
<div class='products grid-product mb30'>
<div class='product-grid-inner'>
<div class='product-grid-img'>
<ul class='product-labels'>
<li>Sale</li>
</ul>
<a href='shoppingdetail.aspx?pid=4' class='product-grid-img-thumbnail'>
<img src='images/vendor/4-1-1.png' alt='img' class='img-responsive primary-img' />

<img src='images/vendor/4-1-2.png' alt='img' class='img-responsive socendary-img'
/>
</a>

```

```

        <div class='product-grid-atb text-center'><a class=""
href='shoppingdetail.aspx?pid=4'>Quick View</a></div>
    </div>
    <div class='product-grid-caption text-center'>
    <h6><a class='p-grid-title' href='shoppingdetail.aspx?pid=4'>Western
Clothing</a></h6>
    <p>Western Clothing</p>
    <h3 class='shop-product-title'>Sizes Available</h3>
    <ul class='list-inline product-size mb30'>
    <li>
    <input type='radio' id='size-43' name='sizeG4' checked >
    <label for='sizeG-43'>S</label>
    </li>
    <li>
    <input type='radio' id='size-44' name='sizeG4' >
    <label for='sizeG-44'>M</label>
    </li>
    <li>
    <input type='radio' id='size-45' name='sizeG4' >
    <label for='sizeG-45'>L</label>
    </li>
    </ul>
    <div><li><a href='ShoppingList.aspx?vendor=6'>Sold by : <span class='btn btn-default
special-price'>Vendor</span></li> </div>
    <div><li><a href='ShoppingList.aspx?brand=Brand'> Brand : <span class='fa glyphicon-bold
special-price'>Brand</span></li></div>
    <span class='price clearfix'><span class='old-price'>₹12000.00</span> ₹ <span
class='special-price'>12000.00</span></span>
    <a class='btn btn-product cd-cart-btn' href='MyCart.aspx?psid=4'><i class='fa fa-
shopping-cart'></i> Add to Cart</a>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    <div class='row clearfix'>
    </div>
    <div id='DivG2' class='Row clearfix' style='display:none'>
    <div class='col-md-4 col-sm-6 col-xs-12'>
    <div class='products grid-product mb30'>
    <div class='product-grid-inner'>
    <div class='product-grid-img'>
    <ul class='product-labels'>
    <li>Sale</li>
    </ul>
    <a href='shoppingdetail.aspx?pid=5' class='product-grid-img-thumbnail'>
    <img src='images/vendor/5-1-1.png' alt='img' class='img-responsive primary-img' />

    <img src='images/vendor/5-1-2.png' alt='img' class='img-responsive socendary-img'
/>
    </a>
    <div class='product-grid-atb text-center'><a class=""
href='shoppingdetail.aspx?pid=5'>Quick View</a></div>
    </div>
    <div class='product-grid-caption text-center'>

```

```

        <h6><a class='p-grid-title' href='shoppingdetail.aspx?pid=5'>Indian wedding
dress</a></h6>
        <p>Indian wedding dress</p>
<div><li><a href='ShoppingList.aspx?vendor=6'>Sold by : <span class='btn btn-default
special-price'>Vendor</span></li> </div>
<div><li><a href='ShoppingList.aspx?brand=Brand'> Brand : <span class='fa glyphicon-bold
special-price'>Brand</span></li></div>
        <span class='price clearfix'><span class='old-price'>₹2341.00</span> ₹ <span
class='special-price'>2341.00</span></span>
        <a class='btn btn-product cd-cart-btn' href='MyCart.aspx?psid=5'><i class='fa fa-
shopping-cart'></i> Add to Cart</a>
        </div>
        </div>
        </div>
        </div>
<div class='col-md-4 col-sm-6 col-xs-12'>
<div class='products grid-product mb30'>
<div class='product-grid-inner'>
<div class='product-grid-img'>
<ul class='product-labels'>
<li>Sale</li>
</ul>
<a href='shoppingdetail.aspx?pid=6' class='product-grid-img-thumbnail'>
<img src='images/vendor/6-4-1.png' alt='img' class='img-responsive primary-img' />

<img src='images/vendor/6-4-2.png' alt='img' class='img-responsive socendary-img'
/>

</a>
<div class='product-grid-atb text-center'><a class=""
href='shoppingdetail.aspx?pid=6'>Quick View</a></div>
</div>
<div class='product-grid-caption text-center'>
<h6><a class='p-grid-title' href='shoppingdetail.aspx?pid=6'>SALWAR KAMEEZ
BRIDES WEDDING</a></h6>
<p>SALWAR KAMEEZ BRIDES WEDDING</p>
<div><li><a href='ShoppingList.aspx?vendor=6'>Sold by : <span class='btn btn-default
special-price'>Vendor</span></li> </div>
<div><li><a href='ShoppingList.aspx?brand=Brand'> Brand : <span class='fa glyphicon-bold
special-price'>Brand</span></li></div>
        <span class='price clearfix'><span class='old-price'>₹123.00</span> ₹ <span
class='special-price'>123.00</span></span>
        <a class='btn btn-product cd-cart-btn' href='MyCart.aspx?psid=6'><i class='fa fa-
shopping-cart'></i> Add to Cart</a>
        </div>
        </div>
        </div>
<div class='col-md-4 col-sm-6 col-xs-12'>
<div class='products grid-product mb30'>
<div class='product-grid-inner'>
<div class='product-grid-img'>
<ul class='product-labels'>
<li>Sale</li>
</ul>
<a href='shoppingdetail.aspx?pid=7' class='product-grid-img-thumbnail'>

```

```

<img src='images/vendor/7-1-1.png' alt='img' class='img-responsive primary-img' />

<img src='images/vendor/7-1-2.png' alt='img' class='img-responsive socendary-img'
/>
</a>
<div class='product-grid-atb text-center'><a class=""
href='shoppingdetail.aspx?pid=7'>Quick View</a></div>
</div>
<div class='product-grid-caption text-center'>
<h6><a class='p-grid-title' href='shoppingdetail.aspx?pid=7'>Indian wedding
dress</a></h6>
<p>Indian wedding dress</p>
<div><li><a href='ShoppingList.aspx?vendor=6'>Sold by : <span class='btn btn-default
special-price'>Vendor</span></li> </div>
<div><li><a href='ShoppingList.aspx?brand=Brand'> Brand : <span class='fa glyphicon-bold
special-price'>Brand</span></li></div>
<span class='price clearfix'><span class='old-price'>₹133.00</span> ₹ <span
class='special-price'>133.00</span></span>
<a class='btn btn-product cd-cart-btn' href='MyCart.aspx?psid=7'><i class='fa fa-
shopping-cart'></i> Add to Cart</a>
</div>
</div>
</div>
</div>
</div>
<div class='row clearfix'>
</div>
<div id='DivG3' class='Row clearfix' style='display:none'>
<div class='col-md-4 col-sm-6 col-xs-12'>
<div class='products grid-product mb30'>
<div class='product-grid-inner'>
<div class='product-grid-img'>
<ul class='product-labels'>
<li>Sale</li>
</ul>
<a href='shoppingdetail.aspx?pid=8' class='product-grid-img-thumbnail'>
<img src='images/vendor/8-2-1.png' alt='img' class='img-responsive primary-img' />

<img src='images/vendor/8-2-2.png' alt='img' class='img-responsive socendary-img'
/>
</a>
<div class='product-grid-atb text-center'><a class=""
href='shoppingdetail.aspx?pid=8'>Quick View</a></div>
</div>
<div class='product-grid-caption text-center'>
<h6><a class='p-grid-title' href='shoppingdetail.aspx?pid=8'>kurtaset for women and
wedding set</a></h6>
<p>kurtaset for women and wedding set</p>
<div><li><a href='ShoppingList.aspx?vendor=6'>Sold by : <span class='btn btn-default
special-price'>Vendor</span></li> </div>
<div><li><a href='ShoppingList.aspx?brand=Brand'> Brand : <span class='fa glyphicon-bold
special-price'>Brand</span></li></div>
<span class='price clearfix'><span class='old-price'>₹2332.00</span> ₹ <span
class='special-price'>2332.00</span></span>

```



```

        <a class='btn btn-product cd-cart-btn' href='MyCart.aspx?psid='8'><i class='fa fa-
shopping-cart'></i> Add to Cart</a>
    </div>
</div>
</div>
</div>
<div class='col-md-4 col-sm-6 col-xs-12'>
    <div class='products grid-product mb30'>
        <div class='product-grid-inner'>
            <div class='product-grid-img'>
                <ul class='product-labels'>
                    <li>Sale</li>
                </ul>
                <a href='shoppingdetail.aspx?pid=9' class='product-grid-img-thumbnail'>
                    <img src='images/vendor/9-2-1.png' alt='img' class='img-responsive primary-img' />

                    <img src='images/vendor/9-2-2.png' alt='img' class='img-responsive socendary-img'
/>
                </a>
                <div class='product-grid-atb text-center'><a class=""
href='shoppingdetail.aspx?pid=9'>Quick View</a></div>
                </div>
                <div class='product-grid-caption text-center'>
                    <h6><a class='p-grid-title' href='shoppingdetail.aspx?pid=9'>Indian wedding
dress</a></h6>
                    <p>Indian wedding dress</p>
                <div><li><a href='ShoppingList.aspx?vendor=6'>Sold by : <span class='btn btn-default
special-price'>Vendor</span></li> </div>
                <div><li><a href='ShoppingList.aspx?brand=Brand'> Brand : <span class='fa glyphicon-bold
special-price'>Brand</span></li></div>
                    <span class='price clearfix'><span class='old-price'>₹343.00</span> ₹ <span
class='special-price'>343.00</span></span>
                    <a class='btn btn-product cd-cart-btn' href='MyCart.aspx?psid='9'><i class='fa fa-
shopping-cart'></i> Add to Cart</a>
                </div>
            </div>
</div>
</div>
</div>
</div>
<div class='col-md-4 col-sm-6 col-xs-12'>
    <div class='products grid-product mb30'>
        <div class='product-grid-inner'>
            <div class='product-grid-img'>
                <ul class='product-labels'>
                    <li>Sale</li>
                </ul>
                <a href='shoppingdetail.aspx?pid=10' class='product-grid-img-thumbnail'>
                    <img src='images/vendor/10-2-1.png' alt='img' class='img-responsive primary-img' />

                    <img src='images/vendor/9-2-2.png' alt='img' class='img-responsive socendary-img'
/>
                </a>
                <div class='product-grid-atb text-center'><a class=""
href='shoppingdetail.aspx?pid=10'>Quick View</a></div>
                </div>
                <div class='product-grid-caption text-center'>

```

```

        <h6><a class='p-grid-title' href='shoppingdetail.aspx?pid=10'>Kanchvaram Pattu
saree Gold wedding</a></h6>
        <p>Kanchvaram Pattu saree Gold wedding</p>
<div><li><a href='ShoppingList.aspx?vendor=6'>Sold by : <span class='btn btn-default
special-price'>Vendor</span></li> </div>
<div><li><a href='ShoppingList.aspx?brand=Brand'> Brand : <span class='fa glyphicon-bold
special-price'>Brand</span></li></div>
        <span class='price clearfix'><span class='old-price'>₹1233.00</span> ₹ <span
class='special-price'>1233.00</span></span>
        <a class='btn btn-product cd-cart-btn' href='MyCart.aspx?psid='10'><i class='fa fa-
shopping-cart'></i> Add to Cart</a>
        </div>
        </div>
        </div>
        </div>
        </div>
<div class='row clearfix'>
        </div>
<div id='DivG4' class='Row clearfix' style='display:none'>
        <div class='col-md-4 col-sm-6 col-xs-12'>
        <div class='products grid-product mb30'>
        <div class='product-grid-inner'>
        <div class='product-grid-img'>
        <ul class='product-labels'>
        <li>Sale</li>
        </ul>
        <a href='shoppingdetail.aspx?pid=11' class='product-grid-img-thumbnail'>
        <img src='images/vendor/10-2-2.png' alt='img' class='img-responsive primary-img' />

        <img src='images/vendor/9-2-2.png' alt='img' class='img-responsive socendary-img'
/>
        </a>
        <div class='product-grid-atb text-center'><a class=""
href='shoppingdetail.aspx?pid=11'>Quick View</a></div>
        </div>
        <div class='product-grid-caption text-center'>
        <h6><a class='p-grid-title' href='shoppingdetail.aspx?pid=11'>Kanchvaram Pattu
saree Gold wedding</a></h6>
        <p>Kanchvaram Pattu saree Gold wedding</p>
<div><li><a href='ShoppingList.aspx?vendor=6'>Sold by : <span class='btn btn-default
special-price'>Vendor</span></li> </div>
<div><li><a href='ShoppingList.aspx?brand=Brand'> Brand : <span class='fa glyphicon-bold
special-price'>Brand</span></li></div>
        <span class='price clearfix'><span class='old-price'>₹1333.00</span> ₹ <span
class='special-price'>1333.00</span></span>
        <a class='btn btn-product cd-cart-btn' href='MyCart.aspx?psid='11'><i class='fa fa-
shopping-cart'></i> Add to Cart</a>
        </div>
        </div>
        </div>
        </div>
        </div>
<div class='col-md-4 col-sm-6 col-xs-12'>
        <div class='products grid-product mb30'>
        <div class='product-grid-inner'>
        <div class='product-grid-img'>

```

```

<ul class='product-labels'>
<li>Sale</li>
</ul>
<a href='shoppingdetail.aspx?pid=12' class='product-grid-img-thumbnail'>
<img src='images/vendor/12-5-1.png' alt='img' class='img-responsive primary-img' />

<img src='images/vendor/9-2-2.png' alt='img' class='img-responsive socendary-img'
/>
</a>
<div class='product-grid-atb text-center'><a class='
href='shoppingdetail.aspx?pid=12'>Quick View</a></div>
</div>
<div class='product-grid-caption text-center'>
<h6><a class='p-grid-title' href='shoppingdetail.aspx?pid=12'>Kanchvaram Pattu
saree Gold wedding</a></h6>
<p>Kanchvaram Pattu saree Gold wedding</p>
<div><li><a href='ShoppingList.aspx?vendor=6'>Sold by : <span class='btn btn-default
special-price'>Vendor</span></li> </div>
<div><li><a href='ShoppingList.aspx?brand=Brand'> Brand : <span class='fa glyphicon-bold
special-price'>Brand</span></li></div>
<span class='price clearfix'><span class='old-price'>₹2332.00</span> ₹ <span
class='special-price'>2332.00</span></span>
<a class='btn btn-product cd-cart-btn' href='MyCart.aspx?psid='12'><i class='fa fa-
shopping-cart'></i> Add to Cart</a>
</div>
</div>
</div>
</div>
<div class='col-md-4 col-sm-6 col-xs-12'>
<div class='products grid-product mb30'>
<div class='product-grid-inner'>
<div class='product-grid-img'>
<ul class='product-labels'>
<li>Sale</li>
</ul>
<a href='shoppingdetail.aspx?pid=13' class='product-grid-img-thumbnail'>
<img src='images/vendor/13-2-1.png' alt='img' class='img-responsive primary-img' />

<img src='images/vendor/13-2-2.png' alt='img' class='img-responsive socendary-
img' />
</a>
<div class='product-grid-atb text-center'><a class='
href='shoppingdetail.aspx?pid=13'>Quick View</a></div>
</div>
<div class='product-grid-caption text-center'>
<h6><a class='p-grid-title' href='shoppingdetail.aspx?pid=13'>kurtaset for women
and wedding set</a></h6>
<p>kurtaset for women and wedding set</p>
<h3 class='shop-product-title'>Colors Available</h3>
<ul class='list-inline product-color mb30'>
<li>
<li>
<input type='radio' id='colorG-13-1' name='colorG13'>
<img id='imgG13-1' src='images\Colors\Black.png' /></li> <li>
<input type='radio' id='colorG-13-2' name='colorG13'>

```

```

<img id='imgG13-2' src='images\Colors\white.png' /></li> <li>
  <input type='radio' id='colorG-13-3' name='colorG13'>
<img id='imgG13-3' src='images\Colors\Red.png' /></li> </ul>
<div><li><a href='ShoppingList.aspx?vendor=6'>Sold by : <span class='btn btn-default
special-price'>Vendor</span></li> </div>
<div><li><a href='ShoppingList.aspx?brand=Brand'> Brand : <span class='fa glyphicon-bold
special-price'>Brand</span></li></div>
  <span class='price clearfix'><span class='old-price'>₹2345.00</span> ₹ <span
class='special-price'>2345.00</span></span>
  <a class='btn btn-product cd-cart-btn' href='MyCart.aspx?psid='13'><i class='fa fa-
shopping-cart'></i> Add to Cart</a>
  </div>
  </div>
  </div>
  </div>
  </div>
  </div>
<div class='row clearfix'>
  </div>
<div id='DivG5' class='Row clearfix' style='display:none'>
  <div class='col-md-4 col-sm-6 col-xs-12'>
    <div class='products grid-product mb30'>
      <div class='product-grid-inner'>
        <div class='product-grid-img'>
          <ul class='product-labels'>
            <li>Sale</li>
          </ul>
          <a href='shoppingdetail.aspx?pid=14' class='product-grid-img-thumbnail'>
            <img src='images/vendor/14-2-1.png' alt='img' class='img-responsive primary-img' />

            <img src='images/vendor/14-2-2.png' alt='img' class='img-responsive socendary-
img' />
          </a>
          <div class='product-grid-atb text-center'><a class=""
href='shoppingdetail.aspx?pid=14'>Quick View</a></div>
          </div>
          <div class='product-grid-caption text-center'>
            <h6><a class='p-grid-title' href='shoppingdetail.aspx?pid=14'>kurtaset for women
and wedding set</a></h6>
            <p>kurtaset for women and wedding set</p>
            <h3 class='shop-product-title'>Colors Available</h3>
            <ul class='list-inline product-color mb30'>
              <li>
                <li>
                  <input type='radio' id='colorG-14-0' name='colorG14'>
<img id='imgG14-0' src='images\Colors\nocolor.png' /></li> <li>
                  <input type='radio' id='colorG-14-1' name='colorG14'>
<img id='imgG14-1' src='images\Colors\Black.png' /></li> <li>
                  <input type='radio' id='colorG-14-2' name='colorG14'>
<img id='imgG14-2' src='images\Colors\white.png' /></li> <li>
                  <input type='radio' id='colorG-14-3' name='colorG14'>
<img id='imgG14-3' src='images\Colors\Red.png' /></li> </ul>
                <div><li><a href='ShoppingList.aspx?vendor=6'>Sold by : <span class='btn btn-default
special-price'>Vendor</span></li> </div>
                <div><li><a href='ShoppingList.aspx?brand=Brand'> Brand : <span class='fa glyphicon-bold
special-price'>Brand</span></li></div>

```

```
<span class='price clearfix'><span class='old-price'>₹232.00</span> ₹ <span
class='special-price'>232.00</span></span>
<a class='btn btn-product cd-cart-btn' href='MyCart.aspx?psid='14'><i class='fa fa-
shopping-cart'></i> Add to Cart</a>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
```

```
</div>
```

```
</div>
```

```
<!--dynamic code ends here-->
```

```
<!-- Start Pagination-->
```

```
<div id="paging" class=" cd-pagination "" runat="server"> <!--href="#"row
clearfix-->
```

```
<ul class="pagination">
<li><a onclick="LB1_Click"><i class="fa fa-angle-left" aria-
hidden="true"></i></a></li>
<li id="pno" class="active"><a onclick="LB1_Click" runat="server">1</a></li>
<!--
<li><a onclick="javascript:ShowPage('DivL2','DivG2')">2</a></li>
<li><a onclick="javascript:ShowPage('DivL3','DivG3')">3</a></li>
<li><a onclick="javascript:ShowPage('DivL4','DivG4')">4</a></li>
<li><a onclick="javascript:ShowPage('DivL5','DivG5')">5</a></li>-->
<li><a onclick="LB2_Click"><i class="fa fa-angle-right" aria-
hidden="true"></i></a></li>

</ul>
```

```
<!--
<ul class="pagination">
<li><a href="#"><i class="fa fa-angle-left" aria-hidden="true"></i></a></li>
<li class="active"><a onclick="javascript:ShowPage('DivL1','DivG1')">1</a></li>
<li><a onclick="javascript:ShowPage('DivL2','DivG2')">2</a></li>
<li><a onclick="javascript:ShowPage('DivL3','DivG3')">3</a></li>
<li><a onclick="javascript:ShowPage('DivL4','DivG4')">4</a></li>
<li><a onclick="javascript:ShowPage('DivL5','DivG5')">5</a></li>
<li><a onclick="javascript:ShowPage('DivL6','Div6')"><i class="fa fa-angle-right"
aria-hidden="true"></i></a></li>
</ul>
```

```
--> </div>
```

```
</div>
```

```
</div>
```

```

<script type="text/javascript">
  function ShowPage(Div,Div1,pg,tpg)
  {
    alert("test");
    //document.getElementById('DivL1').style.display = "none";
    //document.getElementById('DivL2').style.display = "none";
    //document.getElementById('DivL3').style.display = "none";
    //document.getElementById('DivL4').style.display = "none";
    //document.getElementById('DivL5').style.display = "none";
    //document.getElementById('DivL6').style.display = "none";
    //document.getElementById('DivG1').style.display = "none";
    //document.getElementById('DivG2').style.display = "none";
    //document.getElementById('DivG3').style.display = "none";
    //document.getElementById('DivG4').style.display = "none";
    //document.getElementById('DivG5').style.display = "none";
    var i=1;
    for (i = 1; i <= tpg; i++) {
      document.getElementById("DivL"+i).style.display = "none";
      document.getElementById("DivG" + i).style.display = "none";
    }
    document.getElementById(Div).style.display = "block";
    document.getElementById(Div1).style.display = "block";
    document.getElementById(pg).classList.add("active");
    //document.getElementById(Div1).classList.add("active");
  }
</script>
<!-- End Pagination-->

</div>

<!-- END MAIN CONTAIN -->

<!-- ./Content Bodey Here -->

</asp:Content>
<asp:Content ID="Content6" ContentPlaceHolderID="ContentFooter" Runat="Server">
</asp:Content>

```

```

***** ShoppingList.aspx.cs*****
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBAL;
using DowBEntity;
//using LogTracer;
using System.Text;
using System.Text.RegularExpressions;

```

```

public partial class ShoppingList: System.Web.UI.Page
{
    static string key = "abcdefg123456";
    static string salt = "123456abcdefg";
    DDLogTracer objlog = new DDLogTracer();
    bool IsMobile = false;
    string urlqry = "";
    BalCommon blc = new BalCommon();
    protected void Page_Load(object sender, EventArgs e)
    {
        //string qry = Request.QueryString.ToString();
        //if (qry.ToString().Substring(0,3) != "pid")
        //{
            // Common.ShowMessageHtml(new BalCommon().Decrypt(qry), 1, true,
this.dvMsgBox);
            // urlqry = new BalCommon().Decrypt(qry).ToQueryString();
        //}

        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr != null)
        {
        }
        else
        {
            Response.Redirect("~/MLB_Login.aspx");
        }
        if (Request.Browser.IsMobileDevice) { IsMobile = true; } else { IsMobile = false; }
        if (IsMobile)
        {
            sidebar.Attributes.CssStyle.Add("display", "none;");
        }
        //if (!IsPostBack)
        { Bindetails(); }
    }
    private void Bindetails()
    {
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        string pt = "", title = "", vendor, qstring = "", pagesort = "";
        int pid = 0;
        var page = Session["page"].ToString();
        int curpage = Convert.ToInt16(page);

        ImgBtn1.Enabled = false;
        ImgBtn2.Enabled = true;
        appvendors.Attributes.CssStyle.Add("display", "block;");
        AppGrid.Attributes.CssStyle.Add("display", "none;");
        int srno = 1, perpage = 30, dvno = 1, sno = 1;
        string cat = Request.QueryString["cid"];
        pagesort= Request.QueryString["psort"];
        if (pagesort == null) { pagesort = "R"; }

        string ppage = Request.QueryString["ppage"];
        if (ppage != null) { perpage = Convert.ToInt16(ppage); }
    }
}

```

```

if (cat != "" && cat != null && cat!="0")
{
    pid = Convert.ToInt16(cat);
    pt = "cat";
}
string SrchStrng = Request.QueryString["SrchStrng"];
if (SrchStrng != null && SrchStrng != "")
{
    pt = "Search";
}
string PreHtml = "";
StringBuilder sb = new StringBuilder();
StringBuilder sbg = new StringBuilder();
vendor = Request.QueryString["vendor"];

sb.Append(" <div class='directory-list'>");
if (pid == 0)
{
    pid = Convert.ToInt16(Request.QueryString["pid"]);
    if (pid != 0)
    {
        pt = "prod";
        qstring = "pid=" + pid;
    }
}

//try
{
    IList<ProductProductSeller> Lst = new List<ProductProductSeller>();
    if (vendor != null)
    {
        Lst = (new Products()).GetProductSellerbySellerId(Convert.ToInt16(vendor), 1);
        qstring = "vendor=" + vendor;
        pt = "vendor";
    }
    else if (pt == "prod")
    {
        Lst = (new Products()).GetProductSellerbyProductId(pid, "SearchList");
        if (Lst.Count() == 0 && pid != 0)
        {
            Eteproduct ep = new Products().GetProduct(pid, "");
            string pname = ep.ProductName;
            Lst = new Products().GetProductSellerbySearch(pname, 1);
            qstring = "pid=" + pid;
        }
    }
    else if(pt=="Search")
    {
        Lst = (new Products()).GetProductSellerbySearch(SrchStrng,1);
        qstring = "SrchStrng=" + SrchStrng;
    }
    else if(pt=="cat")
    {

```



```

Lst = (new Products()).GetProductSellerbyCatId(pid, 1, "SearchList");
qstring = "cid=" + pid;

}
if (Lst.Count > 0)
{
    ProductProductSeller pst = (from c in Lst select c).FirstOrDefault();

    if (pt == "prod")
    {
        title = pst.ProductName;
    }
    else if (pt == "vendor")
    {
        title = Lst[0].SellerName;
    }
    else
    {
        title = pst.CatName;
    }
    sb.Append("<h2>" + title + "</h2>\n <small class='category-num-coun'>" +
Convert.ToString(Lst.Count) + " Results</small>\n");
    results.InnerHtml = sb.ToString();
    sb = new StringBuilder();
    string path = "images/NoImage.jpg";
    string path1 = "images/NoImage.jpg";
    if (Lst != null && Lst.Count > 0)
    {
        string pageshow = "block";
        sbg.Append("\n <div class='row clearfix'>\n");
        sb.Append("\n <div class='row clearfix'>\n");
        if (curpage != 1) { pageshow = "none"; }
        sb.Append("<div id='DivL" + dvno + "' class='products-list products-list-
description-brd mb30' style='display:" + pageshow + "'>\n");

        sbg.Append("<div id='DivG" + dvno + "' class='Row clearfix' style='display:" +
pageshow + "'> \n");
        sbg.Append(" <div class='row clearfix'>\n");
        IList<ProductProductSeller> Lst1 = new List<ProductProductSeller>();
        if (pagesort == "H")
        {
            Lst1 = (from c in Lst orderby c.Price descending select c).ToList();
        } else
        if (pagesort == "L")
        {
            Lst1 = (from c in Lst orderby c.Price ascending select c).ToList();
        }
        else
        if (pagesort == "N")
        {
            Lst1 = (from c in Lst orderby c.CreatedOn descending select c).ToList();
        }
        else
        {

```

```

        Lst1 = (from c in Lst  select c).ToList();

    }

foreach (var dt in Lst1)
{
    string href = "";
    href = "ShoppingDetail.aspx?id=" + dt.PSId;
    var list = new Products().GetProductSellerImage((int)dt.PSId, "").ToList();
    int cnt = list.Count();
    if (cnt > 1)
    {
        path = list[0].ImagePath;
        path1 = list[1].ImagePath; //"http://rk.bakhala.com/"+
    }
    else
        if (list.Count() > 0)
        {
            path = list[0].ImagePath;
        }

    if (path == null || path == "")
    {
        path = "images/NoImage.jpg";
    }
    if (path1 == null || path1 == "") { path1 = path; }
    List<EteproductSellerStocks> stk = new
Products().GetSellerStocksbyPsId((int)dt.PSId);
    var col = (from c in stk where c.Stock > 0 orderby c.colId select
c.colId).Distinct();
    var siz = (from d in stk where d.Stock > 0 orderby d.EpOptId select
d.EpOptId).Distinct();

    //list code
    sb.Append(" <div class='products-list products-list-description-brd mb30'>
\n");
    sb.Append(" <div class='row'> \n");
    sb.Append(" <div class='col-sm-4'> \n");
    sb.Append(" <div class='product-grid-img'> \n");
    sb.Append(" <ul class='product-labels'> \n");
    sb.Append(" <li>Sale</li> \n");
    sb.Append(" </ul> \n");
    sb.Append(" <a href='shoppingdetail.aspx?pid=" + dt.PSId + "
class='product-grid-img-thumbnail'> \n");
    sb.Append("  \n");
    sb.Append("  \n");
    sb.Append(" </a> \n");
    sb.Append(" </div> \n");
    sb.Append(" </div> \n");
    sb.Append(" <div class='col-sm-8 products-list-description'> \n");
    sb.Append(" <div class='overflow-h'> \n");

```

```

        sb.Append("    <ul class='list-inline overflow-h mb15'>        \n");
        sb.Append("    <li><h4 class='title-price'><a href='shoppingdetail.aspx?pid="
+ dt.PSId + "'>" + dt.ProductDescription + "</a></h4></li> \n");
        sb.Append("    <li class='pull-right'> \n");
        sb.Append("    <ul class='list-inline product-ratings'>        \n");
        sb.Append("    <li><i class='rating-selected fa fa-star'></i></li>        \n");
        sb.Append("    <li><i class='rating-selected fa fa-star'></i></li>        \n");
        sb.Append("    <li><i class='rating-selected fa fa-star'></i></li>        \n");
        sb.Append("    <li><i class='rating fa fa-star'></i></li>        \n");
        sb.Append("    <li><i class='rating fa fa-star'></i></li>        \n");
        sb.Append("    </ul> \n");
        sb.Append("    </li> \n");
        sb.Append("    </ul> \n");
        sb.Append("    <p class='mb15'>" + dt.SellerDescription + "</p> \n");

//colors and sizes
if (siz.Count() > 1)
{
    sb.Append("        <h3 class='shop-product-title'>Sizes Available</h3>
\n");
    sb.Append("        <ul class='list-inline product-size mb30'> \n");
    string chk = "checked";

    foreach (var sz in siz)
    {
        string size = new Products().GetSizeByOptId(sz);
        sb.Append("        <li> \n");
        sb.Append("        <input type='radio' id='sizeL-" + dt.PSId + "-" + sz + "
name='sizeL" + dt.PSId + "" + chk + "> \n");
        sb.Append("        <label for='sizeL-" + dt.PSId + "-" + sz + "'>" + size +
"</label> \n");

        sb.Append("        </li> \n");
        chk = "";
    }
    sb.Append("        </ul> \n");
}
if (col.Count() > 1)
{
    sb.Append("        <h3 class='shop-product-title'>Colors Available</h3>
\n");
    sb.Append("        <ul class='list-inline product-color mb30'> \n");
    sb.Append("        <li> \n");
    foreach (var colr in col)
    {
        Eteproduct cl = new Products().GetColorByColId(colr);
        sb.Append("        <li> \n");
        sb.Append("        <input type='radio' id='colorL-" + dt.PSId + "-" +
cl.ProductId + " name='colorL" + dt.PSId + "'> \n");
        sb.Append("        </li>");
    }
    sb.Append("        </ul> \n");
}
}
//colors and sizes ends

```

```

        sb.Append(" <div class='mb10'> \n");
        sb.Append(" <span class='title-price'>₹" + dt.Price.ToString() + "</span>
\n");
        sb.Append(" <span class='title-price line-through'>₹" + dt.Price.ToString() +
"</span> \n");
        sb.Append(" </div> \n");
        sb.Append("<div><a href='ShoppingList.aspx?vendor=" + dt.SellerId + "'> +
\"Sold by : <span class='btn btn-default special-price'>" + dt.SellerName + "</span> </div>
\n");
        sb.Append("<div><a href='ShoppingList.aspx?brand=" + dt.BrandName + "'> +
+ \" Brand : <span class='fa glyphicon-bold special-price'>" + dt.BrandName +
"</span></div> \n");
        sb.Append("<ul class='list-inline add-to-compare mb15'> <li
class='product-wishlist'> \n");
        sb.Append(" <i class='fa fa-exchange-alt'></i> \n");
        sb.Append(" <a href='shoppingdetail.aspx?pid=" + dt.PSId + "'>Add to
Compare</a> \n");
        sb.Append(" </li></ul> \n");
        sb.Append(" <ul class='list-inline add-to-wishlist mb15'> \n");
        sb.Append(" <li class='product-wishlist'> \n");
        sb.Append(" <i class='fa fa-heart'></i> \n");
        sb.Append(" <a href='MyWishList.aspx?psid=" + dt.PSId + "'>Add to
Wishlist</a> \n");
        sb.Append(" </li></ul> \n");
        sb.Append(" <div class='product-grid-atb text-center'><i class='fas fa-
shopping-cart'></i><a class=" href='shoppingdetail.aspx?pid=" + dt.PSId + "'>Add to
Cart</a></div> \n");
        //sb.Append(" </ul> \n"); sb.Append(" <button type='button' class='btn
btn-product'>Add to Cart</button> \n");
        sb.Append(" <a href='MyCart.aspx?pid=" + dt.PSId + "' class='btn btn-
product'>Add to Cart</a> \n");

        sb.Append(" </div> \n");
        sb.Append(" </div> \n");
        sb.Append(" </div> \n");
        sb.Append(" </div> \n");
        sb.Append(" \n");

        //grid code
        sbg.Append(" <div class='col-md-4 col-sm-6 col-xs-12'> \n");
        sbg.Append(" <div class='products grid-product mb30'> \n");
        sbg.Append(" <div class='product-grid-inner'> \n");
        sbg.Append(" <div class='product-grid-img'> \n");
        sbg.Append(" <ul class='product-labels'> \n");
        sbg.Append(" <li>Sale</li> \n");
        sbg.Append(" </ul> \n");
        sbg.Append(" <a href='shoppingdetail.aspx?pid=" + dt.PSId + "'
class='product-grid-img-thumbnail'> \n");
        sbg.Append("  \n");
        sbg.Append("  \n");
        sbg.Append(" </a> \n");
        sbg.Append(" <div class='product-grid-atb text-center'><a class="
href='shoppingdetail.aspx?pid=" + dt.PSId + "'>Quick View</a></div> \n");

```

```

        sbg.Append(" </div> \n");
        sbg.Append(" <div class='product-grid-caption text-center'> \n");
        sbg.Append(" <h6><a class='p-grid-title' href='shoppingdetail.aspx?pid=" +
dt.PSId + ">" + dt.ProductDescription + "</a></h6> \n");
        sbg.Append(" <p>" + dt.SellerDescription + "</p> \n");
        //colors and sizes
        if (siz.Count() > 1)
        {
            sbg.Append(" <h3 class='shop-product-title'>Sizes Available</h3>
\n");
            sbg.Append(" <ul class='list-inline product-size mb30'> \n");
            string chk = "checked";

            foreach (var sz in siz)
            {
                string size = new Products().GetSizeByOptId(sz);
                sbg.Append(" <li> \n");
                sbg.Append(" <input type='radio' id='sizeG-" + dt.PSId + "-" + sz + "
name='sizeG" + dt.PSId + "" + chk + "> \n");
                sbg.Append(" <label for='sizeG-" + dt.PSId + "-" + sz + ">" + size +
"</label> \n");
                //sbg.Append(" <input type='radio' id='size-" + dt.PSId+"-" + sz + "
name='sizeG" + dt.PSId + "" + chk + " > \n");
                //sbg.Append(" <label for='sizeG-" + dt.PSId+"-" + sz + ">" + size +
"</label> \n");

                sbg.Append(" </li> \n");
                chk = "";
            }
            sbg.Append(" </ul> \n");
        }
        if (col.Count() > 1)
        {
            sbg.Append(" <h3 class='shop-product-title'>Colors Available</h3>
\n");
            sbg.Append(" <ul class='list-inline product-color mb30'> \n");
            sbg.Append(" <li> \n");
            foreach (var colr in col)
            {
                Eteproduct cl = new Products().GetColorByColId(colr);
                sbg.Append(" <li> \n");
                sbg.Append(" <input type='radio' id='colorG-" + dt.PSId + "-" +
cl.ProductId + " name='colorG" + dt.PSId + "> \n");
                sbg.Append(" </li>");
            }
            sbg.Append(" </ul> \n");
        }
        //colors and sizes ends
        sbg.Append("<div><li><a href='ShoppingList.aspx?vendor=" + dt.SellerId +
">" + "Sold by : <span class='btn btn-default special-price'>" + dt.SellerName +
"</span></li> </div> \n");
        sbg.Append("<div><li><a href='ShoppingList.aspx?brand=" + dt.BrandName
+ ">" + " Brand : <span class='fa glyphicon-bold special-price'>" + dt.BrandName +
"</span></li></div> \n");

```

```

        //sbg.Append("<li><a href='ShoppingList.aspx?vendor=" + dt.SellerId + "'> +
"sold by: <span class='special-price'>" + dt.SellerName + "</span></a> \n");
        sbg.Append(" <span class='price clearfix'><span class='old-price'>₹" +
dt.Price.ToString() + "</span> ₹ <span class='special-price'>" + dt.Price.ToString() +
"</span></span> \n");
        sbg.Append(" <a class='btn btn-product cd-cart-btn'
href='MyCart.aspx?psid=" + dt.PSId + "'><i class='fa fa-shopping-cart'></i> Add to Cart</a>
\n");
        sbg.Append(" </div> \n");
        sbg.Append(" </div> \n");
        sbg.Append(" </div> \n");
        sbg.Append(" </div> \n");
        srno++;
        sno++;
        if (sno > 3)
        {
            sbg.Append(" </div> \n");
            sbg.Append(" <div class='row clearfix'>\n");
            sno = 1;
        }
        if (srno > perpage)
        {
            dvno++;
            srno = 1;
            if (curpage != dvno) { pageshow = "none"; } else { pageshow = "block"; }

            sb.Append(" </div> \n");
            sbg.Append(" </div> \n");
            sb.Append("<div id='DivL" + dvno + "' class='products-list products-list-
description-brd mb30' style='display:" + pageshow + "'>\n");

            sbg.Append("<div id='DivG" + dvno + "' class='Row clearfix'
style='display:" + pageshow + "'> \n");

        }

    }

    sb.Append(" </div> \n");
    sb.Append(" </div> \n");
    sbg.Append(" </div> \n");
    sbg.Append(" </div> \n");
    sbg.Append(" </div> \n");

    PreHtml = sb.ToString();
    if (PreHtml.Length > 0)
    {
        //objlog.Trace_Error(4, "ShoppingList", "1", "appvendors", PreHtml, "Log");
        appvendors.InnerHtml = PreHtml;
    }
    PreHtml = sbg.ToString();
    if (PreHtml.Length > 0)

```

```

        {
            objlog.Trace_Log(4, "ShoppingList", "1", "AppGrid", PreHtml,
"Log".ToString());
            AppGrid.InnerHtml = PreHtml;

        }
        string ccl;
        int i = 1;
        sb = new StringBuilder();
        sb.Append("        <ul class='pagination'>        \n");
        sb.Append("        <li><a href='#'><i class='fa fa-angle-left' ></i></a></li>
\n"); //aria-hidden='true
        sb.Append("        <li id='Pg" + i + "' class='active'><a
onclick=\"javascript:ShowPage('DivL" + i + "', 'DivG" + i + "', 'Pg" + i + "', '" + dvno + "')\">>" + i +
"</a></li>\n        ");
        for (i = 2; i <= dvno; i++)
        {
            if (curpage == i) { ccl = "active"; } else { ccl = ""; }
            sb.Append("        <li id='Pg" + i + "' class='"+ccl+"'><a
onclick=\"javascript:ShowPage('DivL" + i + "', 'DivG" + i + "', 'Pg" + i + "', '" + dvno + "')\">>" + i +
"</a></li>\n        ");
        }
        sb.Append("        <li><a href='#'><i class='fa fa-angle-right' aria-
hidden='true'></i></a></li>        \n");
        sb.Append("</ul>        \n        ");
        paging.InnerHtml = sb.ToString();
        //objlog.Trace_Log(4, "ShoppingList", "1", "paging", sb.ToString(), "Log");
    }

    sb = new StringBuilder();
    int grp = pst.ProductGroupId;
    List<Eteproduct> prd = new Products().GetProductbyGroup(grp, "");
    if (prd.Count > 0)
    {
        sb.Append("<h6 class='text-uppercase bottom-line'>Show Results for</h6>\n");
        sb.Append("<ul class='icons-list'>\n");
        foreach (var data in prd)
        {
            if (Convert.ToInt16(data.Type) > 0)
            {
                sb.Append("<li><a href='ShoppingList.aspx?pid='" + data.ProductId + "'>" +
data.ProductName + " <span class='pull-right'>" + data.Type + "</span></a></li>\n");
            }
        }
        sb.Append("</ul>\n");
        Filter.InnerHtml = sb.ToString();
    }
    sb = new StringBuilder();
    sb.Append("        <div class='widget mb30'>        ");
    sb.Append("        <h6 class='text-uppercase bottom-line'>Top Rated</h6>        ");
    sb.Append("        </div> ");
    List<ProductProductSeller> top1 = new
Products().GetProductSellerbySearch("all", 1);
    var top = (from c in top1 orderby c.CurPrice descending select c).Take(10);

```

```

foreach (var tr in top)
{
    sb.Append("\n <div class='row clearfix'>\n");

    sb.Append("    <div class='small-thumb-proudct mb30'>    ");
    sb.Append("    <div class='thumb-product'> ");
    sb.Append("    <img class='thumb-product-img img-responsive' src=\"" +
tr.ImagePath + "\" alt=\"">    ");
    //sb.Append("    <div class='thumb-product-inner'>    ");
    sb.Append("    <span class='thumb-product-cat'>" + tr.CatName + "</span>
");
    sb.Append("    <h6><a href='shoppingdetail.aspx?pid=\"" + tr.PSId + "\">" +
tr.ProductDescription + "</a></h6> ");
    sb.Append("    </div> ");
    sb.Append("    <ul class='list-inline overflow-h'>    ");
    sb.Append("    <li class='thumb-product-price line-through'>" + tr.CurPrice +
"</li> ");
    sb.Append("    <li class='thumb-product-price'>" + tr.CurPrice + "</li>    ");
    sb.Append("    </ul> ");
    sb.Append("    </div> ");
    sb.Append("    </div> ");
}
toprate.InnerHtml = sb.ToString();
sb = new StringBuilder();
sb.Append("    <div class='widget mb30'>    ");
sb.Append("    <h6 class='text-uppercase bottom-line'>Best Sellers</h6> ");
sb.Append("    </div> ");
top = (from c in top1 orderby c.CurPrice ascending select c).Take(10);

//List<ProductProductSeller> best = new Products().GetProductSellerbySearch("");
foreach (var tr in top)
{
    sb.Append("\n <div class='row clearfix'>\n");

    sb.Append("    <div class='small-thumb-proudct mb30'>    ");
    sb.Append("    <div class='thumb-product'> ");
    sb.Append("    <img class='thumb-product-img img-responsive' src=\"" +
tr.ImagePath + "\" alt=\"">    ");
    //sb.Append("    <div class='thumb-product-inner'>    ");
    sb.Append("    <span class='thumb-product-cat'>" + tr.CatName + "</span>
");
    sb.Append("    <h6><a href='shoppingdetail.aspx?pid=\"" + tr.PSId + "\">" +
tr.ProductDescription + "</a></h6> ");
    sb.Append("    </div> ");
    sb.Append("    <ul class='list-inline overflow-h'>    ");
    sb.Append("    <li class='thumb-product-price line-through'>" + tr.CurPrice +
"</li> ");
    sb.Append("    <li class='thumb-product-price'>" + tr.CurPrice + "</li>    ");
    sb.Append("    </ul> ");
    sb.Append("    </div> ");
    sb.Append("    </div> ");
}
bestseller.InnerHtml = sb.ToString();
}

```



```

// qstring = "ShoppingList.aspx?" + qstring;
string sort = "30";
if (perpage == 45)
{
    sort = "45";
}
else if (perpage == 60)
{
    sort = "60";
}
else if (perpage == 90)
{
    sort = "90";
}
sb = new StringBuilder();
sb.Append( "\n");
sb.Append("      <button type='button' class='btn btn-default dropdown-toggle'
data-toggle='dropdown'> \n");
sb.Append(sort+ "      <span class='caret'></span> \n");
sb.Append("      </button> \n");
sb.Append("<ul class='dropdown-menu' role='menu' runat='server'>\n");
sb.Append("      <li><a href='" + "ShoppingList.aspx?" + qstring + "&ppage=30"
+ ">30</a></li> \n");
sb.Append("      <li><a href='" + "ShoppingList.aspx?" + qstring + "&ppage=45"
+ ">45</a></li> \n");
sb.Append("      <li><a href='" + "ShoppingList.aspx?" + qstring + "&ppage=60"
+ ">60</a></li> \n");
sb.Append("      <li><a href='" + "ShoppingList.aspx?" + qstring + "&ppage=90"
+ ">90</a></li></ul> \n");
pagenos.InnerHtml = sb.ToString();
//objlog.Trace_Log(4, "ShoppingList", "1", "pagenos", sb.ToString(), "Log");
if (ppage != null)
{
    qstring = qstring + "&ppage="+ppage;
}
else
{
    qstring = qstring + "&ppage=30";
}
sort = "Relevant";
if (pagesort == "H") { sort = "Price High to Low"; }
else if (pagesort == "L") { sort = "Price Low to High"; }
else if (pagesort == "N") { sort = "New Arrived"; }
BalCommon blc = new BalCommon();
sb = new StringBuilder();
sb.Append("\n");
sb.Append("<button type='button' class='btn btn-default dropdown-toggle' data-
toggle='dropdown'> " + sort + " <span class='caret'></span></button>\n");
sb.Append("<ul class='dropdown-menu' role='menu' runat='server'>\n");
sb.Append("      <li><a href='" + "ShoppingList.aspx?" + (qstring + "&psort=R")
+ ">Relevant</a></li> \n");
sb.Append("      <li><a href='" + "ShoppingList.aspx?" + (qstring + "&psort=H" )
+ ">Price High to Low</a></li> \n");
sb.Append("      <li><a href='" + "ShoppingList.aspx?" + (qstring + "&psort=L" )
+ ">Price Low to High</a></li> \n");

```

```

        sb.Append("        <li><a href=\"" + "ShoppingList.aspx?" + (qstring + "&psort=N" )
+ ">New Arrived</a></li></ul>        \n");
        sortbtn.InnerHtml = sb.ToString();
        //objlog.Trace_Log(4, "ShoppingList", "1", "sortbtn", sb.ToString(), "Log");
        //blc.encrypt
    }
    // catch (Exception ex)
    //{

    // Common.ShowMessageHtml(ex.Message, 1, true, this.dvMsgBox);
    // ExceptionLogging.SendExcepToDB(ex);

    //}
}

protected void btn1_Click(object sender, EventArgs e)
{
    appvendors.Attributes.CssStyle.Add("display", "block;");
    AppGrid.Attributes.CssStyle.Add("display", "none;");
}

protected void btn2_Click(object sender, EventArgs e)
{
    appvendors.Attributes.CssStyle.Add("display", "none;");
    AppGrid.Attributes.CssStyle.Add("display", "block;");
}

protected void ImgBtn1_Click(object sender, ImageClickEventArgs e)
{
    appvendors.Attributes.CssStyle.Add("display", "block;");
    AppGrid.Attributes.CssStyle.Add("display", "none;");
    ImgBtn1.Enabled = false;
    ImgBtn2.Enabled = true;
}

protected void ImgBtn2_Click(object sender, ImageClickEventArgs e)
{
    appvendors.Attributes.CssStyle.Add("display", "none;");
    AppGrid.Attributes.CssStyle.Add("display", "block;");
    ImgBtn1.Enabled = true;
    ImgBtn2.Enabled = false;
}

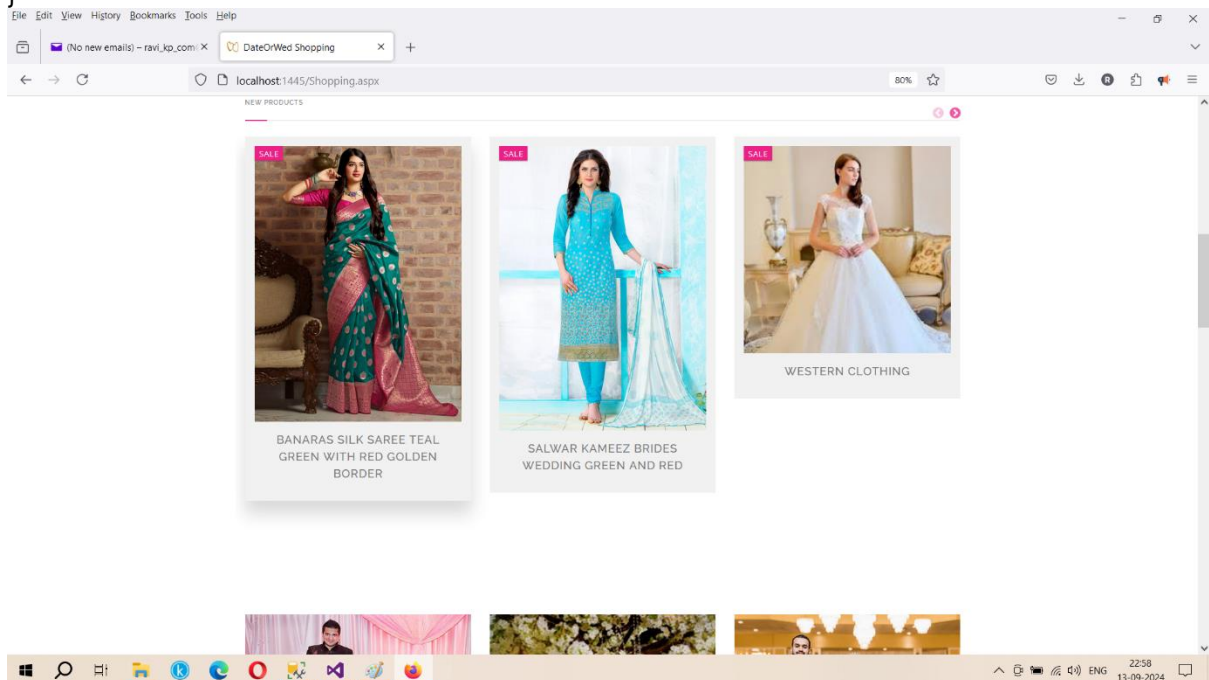
protected void LB1_Click(object sender, EventArgs e)
{
    LinkButton lnkbutton = (LinkButton)sender; ;
    //string pgid = lnkbutton.CommandName;
    string pgid = Session["page"].ToString();
    if (pgid == "" || pgid == null)
    {
        pgid = "1";// Session["page"].ToString();
        pgid = (Convert.ToInt16(pgid) + 1).ToString();
    }
    Session["page"] = pgid;
    Common.ShowMessageHtml("Page" + pgid + ", PS" + Session["page"], 1, true,
this.dvMsgBox);
}

```

```

}
protected void LB2_Click(object sender, EventArgs e)
{
    LinkButton lnkbutton = (LinkButton)sender; ;
    //string pgid = lnkbutton.CommandName;
    string pgid = Session["page"].ToString();
    if (pgid == "" || pgid == null)
    {
        pgid = "1"; //Session["page"].ToString();
        if (Convert.ToInt16(pgid) > 1)
        {
            pgid = (Convert.ToInt16(pgid) - 1).ToString();
        }
    }
    Session["page"] = pgid;
    Common.ShowMessageHtml("Page" + pgid + ", PS" + Session["page"], 1, true,
this.dvMsgBox);
}
}
}

```



***** ShoppingSearch.aspx.cs *****

```

<%@ Page Title="" Language="C#" MasterPageFile="~/ShoppingMasterPage.master"
AutoEventWireup="true" CodeFile="ShoppingSearch.aspx.cs" Inherits="ShoppingSearch"
%>

```

```

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
Runat="Server">

```

```

</asp:Content>

```

```

<asp:Content ID="Content2" ContentPlaceHolderID="ContentMainPage" Runat="Server">

```

```

</asp:Content>

```

```

<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" Runat="Server">

```

```

</asp:Content>

```

```
<asp:Content ID="Content5" ContentPlaceHolderID="CPH_MainContent" Runat="Server">
<!-- Content Bodey Here -->
```

```
    <!-- START MAIN CONTAIN -->
    <div class="main-contain">
        <div class="container">
            <div class="row">
                <aside>
                    <div class="col-md-3 col-sm-12 col-xs-12 sidebar">
                        <!-- CATEGORIES WIDGET -->
                        <div id="Filter" class="widget mb30">
                            <h6 class="text-uppercase bottom-line">Show Results for</h6>
                            <ul class="icons-list">
                                <li><a href="#">Sarees <span class="pull-
right">12</span></a></li>
                                <li><a href="#">Lehengas <span class="pull-
right">11</span></a></li>
                                <li><a href="#">Indo Westers <span class="pull-
right">10</span></a></li>
                                <li><a href="#">Salwar Kameez <span class="pull-
right">14</span></a></li>
                                <li><a href="#">Kurtees <span class="pull-
right">15</span></a></li>
                            </ul>
                        </div>
                        <!-- END CATEGORIES WIDGET -->

                        <!-- SIDE-BAR PRODUCT WIDGET -->
                        <div class="">
                            <div class="widget mb30">
                                <h6 class="text-uppercase bottom-line">Top Rated</h6>
                                </div>
                                <div class="small-thumb-proudct mb30">
                                    <div class="thumb-product">
                                        
                                        <div class="thumb-product-inner">
                                            <h6><a href="shoppingdetail.aspx">Printed Su..</a></h6>
                                            <span class="thumb-product-cat">Fashion</span>
                                        </div>
                                        <ul class="list-inline overflow-h">
                                            <li class="thumb-product-price line-through">$60.00</li>
                                            <li class="thumb-product-price">$30.00</li>
                                        </ul>
                                    </div>
                                </div>
                                <div class="small-thumb-proudct mb30">
                                    <div class="thumb-product">
                                        
                                        <div class="thumb-product-inner">
                                            <h6><a href="shoppingdetail.aspx">Printed Su..</a></h6>
                                            <span class="thumb-product-cat">Fashion</span>
                                        </div>
                                    </div>
                                </div>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>
```

```

        </div>
        <ul class="list-inline overflow-h">
            <li class="thumb-product-price">$65.00</li>
        </ul>
    </div>
</div>
<div class="small-thumb-proudct mb30">
    <div class="thumb-product">
        
        <div class="thumb-product-inner">
            <h6><a href="shoppingdetail.aspx">Printed Su..</a></h6>
            <span class="thumb-product-cat">Fashion</span>
        </div>
        <ul class="list-inline overflow-h">
            <li class="thumb-product-price line-through">$75.00</li>
            <li class="thumb-product-price">$65.00</li>
        </ul>
    </div>
</div>
</div>
<!-- END SIDE-BAR PRODUCT WIDGET -->

<!-- SIDE-BAR PRODUCT WIDGET -->
<div class="">
    <div class="widget mb30">
        <h6 class="text-uppercace bottom-line">Best Sellers</h6>
    </div>
    <div class="small-thumb-proudct mb30">
        <div class="thumb-product">
            
            <div class="thumb-product-inner">
                <h6><a href="shoppingdetail.aspx">Printed Su..</a></h6>
                <span class="thumb-product-cat">Fashion</span>
            </div>
            <ul class="list-inline overflow-h">
                <li class="thumb-product-price line-through">$95.00</li>
                <li class="thumb-product-price">$75.00</li>
            </ul>
        </div>
    </div>
    <div class="small-thumb-proudct mb30">
        <div class="thumb-product">
            
            <div class="thumb-product-inner">
                <h6><a href="shoppingdetail.aspx">Printed Su..</a></h6>
                <span class="thumb-product-cat">Fashion</span>
            </div>
            <ul class="list-inline overflow-h">
                <li class="thumb-product-price line-through">$70.00</li>
                <li class="thumb-product-price">$60.00</li>
            </ul>
        </div>
    </div>
</div>

```

```

</div>
<div class="small-thumb-proudct mb30">
  <div class="thumb-product">
    
    <div class="thumb-product-inner">
      <h6><a href="shoppingdetail.aspx">Printed Su..</a></h6>
      <span class="thumb-product-cat">Fashion</span>
    </div>
    <ul class="list-inline overflow-h">
      <li class="thumb-product-price">$65.00</li>
    </ul>
  </div>
</div>
</div>

```

<!-- END SIDE-BAR PRODUCT WIDGET -->

```

<!-- TAGS WIDGET -->
<div class="widget">
  <h6 class="text-uppercase bottom-line">Tags</h6>
  <div class="tags">
    <a href="#">bootstrap</a>
    <a href="#">business</a>
    <a href="#">corporate</a>
    <a href="#">e-commerce</a>
    <a href="#">portfolio</a>
    <a href="#">responsive</a>
  </div>
</div>
<!-- END TAGS WIDGET -->
</div>
</aside>

```

```

<div class="col-md-9 col-sm-12 col-xs-12">
  <div class="row hidden-xs">
    <div class="toolbar">
      <div class="col-sm-4 category-num">
        <h2>Designer</h2>
        <small class="category-num-coun">9 Results</small>
      </div>
    </div>
    <div class="col-sm-8">
      <ul class="view-mode">
        <li class="grid-list-icons">
          <a href="shoppinglist.aspx"><i class="fa fa-th-list"></i></a>
          <a href="shoppinggrid.aspx"><i class="fa fa-th"></i></a>
        </li>
        <li class="sort-list-btn">
          <h6>Sort By :</h6>
          <div class="btn-group">
            <button type="button" class="btn btn-default dropdown-toggle"
data-toggle="dropdown">

```

```

        Popularity <span class="caret"></span>
    </button>
    <ul class="dropdown-menu" role="menu">
        <li><a href="#">All</a></li>
        <li><a href="#">Best Sales</a></li>
        <li><a href="#">Top Last Week Sales</a></li>
        <li><a href="#">New Arrived</a></li>
    </ul>
</div>
</li>
<li class="sort-list-btn">
    <h6>Show :</h6>
    <div class="btn-group">
        <button type="button" class="btn btn-default dropdown-toggle"
data-toggle="dropdown">
            20 <span class="caret"></span>
        </button>
        <ul class="dropdown-menu" role="menu">
            <li><a href="#">All</a></li>
            <li><a href="#">10</a></li>
            <li><a href="#">5</a></li>
            <li><a href="#">3</a></li>
        </ul>
    </div>
</li>
</ul>
</div>
</div>
</div>

```

```

<!--dynamic code from here-->
    <div id="appvendors" class="category-products" runat="server">
        <div class="products-list products-list-description-brd mb30">
            <div class="row">
                <div class="col-sm-4">
                    <div class="product-grid-img">
                        <ul class="product-labels">
                            <li>Sale</li>
                        </ul>
                        <a href="shoppingdetail.aspx" class="product-grid-img-
thumbnail">
                            
                            
                        </a>
                        <div class="product-grid-atb text-center"><i class="fa fa-
shopping-cart"></i><a class="" href="#">Add to Cart</a></div>
                    </div>
                </div>
                <div class="col-sm-8 products-list-description">
                    <div class="overflow-h">
                        <ul class="list-inline overflow-h mb15">
                            <li><h4 class="title-price"><a
href="shoppingdetail.aspx">Printed Summer Dress</a></h4></li>

```

```

        <li class="pull-right">
            <ul class="list-inline product-ratings">
                <li><i class="rating-selected fa fa-star"></i></li>
                <li><i class="rating-selected fa fa-star"></i></li>
                <li><i class="rating-selected fa fa-star"></i></li>
                <li><i class="rating fa fa-star"></i></li>
                <li><i class="rating fa fa-star"></i></li>
            </ul>
        </li>
    </ul>
    <div class="mb10">
        <span class="title-price">$20.00</span>
        <span class="title-price line-through">$35.00</span>
    </div>
    <p class="mb15">Lorem ipsum dolor sit amet, consectetur
adipiscing elit. Maecenas sollicitudin erat nec ornarevolu tpat. Etiam ut felis nec nisl eleifend
lobortis. Aenean nibh est, hendrerit non conva.</p>
    <ul class="list-inline add-to-wishlist mb15">
        <li class="product-wishlist">
            <i class="fa fa-heart"></i>
            <a href="#">Add to Wishlist</a>
        </li>
        <li class="product-wishlist">
            <i class="fa fa-exchange"></i>
            <a href="#">Add to Compare</a>
        </li>
    </ul>
    <button type="button" class="btn btn-product">Add to
Cart</button>
    </div>
</div>
</div>
</div>
</div>
<div class="products-list products-list-description-brd mb30">
    <div class="row">
        <div class="col-sm-4">
            <div class="product-grid-img">
                <ul class="product-labels">
                    <li>Sale</li>
                </ul>
                <a href="#" class="product-grid-img-thumbnail">
                    
                    
                </a>
                <div class="product-grid-atb text-center"><i class="fa fa-
shopping-cart"></i><a class="" href="#">Add to Cart</a></div>
            </div>
        </div>
        <div class="col-sm-8 products-list-description">
            <div class="overflow-h">
                <ul class="list-inline overflow-h mb15">
                    <li><h4 class="title-price"><a
href="shoppingdetail.aspx">Printed Summer Dress</a></h4></li>

```



```

<li class="pull-right">
  <ul class="list-inline product-ratings">
    <li><i class="rating-selected fa fa-star"></i></li>
    <li><i class="rating-selected fa fa-star"></i></li>
    <li><i class="rating-selected fa fa-star"></i></li>
    <li><i class="rating fa fa-star"></i></li>
    <li><i class="rating fa fa-star"></i></li>
  </ul>
</li>
</ul>
<div class="mb10">
  <span class="title-price">$50.00</span>
  <span class="title-price line-through">$60.00</span>
</div>
<p class="mb15">Lorem ipsum dolor sit amet, consectetur
adipiscing elit. Maecenas sollicitudin erat nec ornarevolu tpat. Etiam ut felis nec nisl eleifend
lobortis. Aenean nibh est, hendrerit non conva.</p>
<ul class="list-inline add-to-wishlist mb15">
  <li class="product-wishlist">
    <i class="fa fa-heart"></i>
    <a href="#">Add to Wishlist</a>
  </li>
  <li class="product-wishlist">
    <i class="fa fa-exchange"></i>
    <a href="#">Add to Compare</a>
  </li>
</ul>
<button type="button" class="btn btn-product">Add to
Cart</button>
</div>
</div>
</div>
</div>
<div class="products-list products-list-description-brd mb30">
  <div class="row">
    <div class="col-sm-4">
      <div class="product-grid-img">
        <ul class="product-labels">
          <li>Sale</li>
        </ul>
        <a href="#" class="product-grid-img-thumbnail">
          
          
        </a>
        <div class="product-grid-atb text-center"><i class="fa fa-
shopping-cart"></i><a class="" href="#">Add to Cart</a></div>
      </div>
    </div>
    <div class="col-sm-8 products-list-description">
      <div class="overflow-h">
        <ul class="list-inline overflow-h mb15">
          <li><h4 class="title-price"><a
href="shoppingdetail.aspx">Printed Summer Dress</a></h4></li>

```

```

<li class="pull-right">
  <ul class="list-inline product-ratings">
    <li><i class="rating-selected fa fa-star"></i></li>
    <li><i class="rating-selected fa fa-star"></i></li>
    <li><i class="rating-selected fa fa-star"></i></li>
    <li><i class="rating fa fa-star"></i></li>
    <li><i class="rating fa fa-star"></i></li>
  </ul>
</li>
</ul>
<div class="mb10">
  <span class="title-price">$60.00</span>
  <span class="title-price line-through">$70.00</span>
</div>
<p class="mb15">Lorem ipsum dolor sit amet, consectetur
adipiscing elit. Maecenas sollicitudin erat nec ornarevolu tpat. Etiam ut felis nec nisl eleifend
lobortis. Aenean nibh est, hendrerit non conva.</p>
<ul class="list-inline add-to-wishlist mb15">
  <li class="product-wishlist">
    <i class="fa fa-heart"></i>
    <a href="#">Add to Wishlist</a>
  </li>
  <li class="product-wishlist">
    <i class="fa fa-exchange"></i>
    <a href="#">Add to Compare</a>
  </li>
</ul>
<button type="button" class="btn btn-product">Add to
Cart</button>
</div>
</div>
</div>
</div>
<div class="products-list products-list-description-brd mb30">
  <div class="row">
    <div class="col-sm-4">
      <div class="product-grid-img">
        <ul class="product-labels">
          <li>Sale</li>
        </ul>
        <a href="#" class="product-grid-img-thumbnail">
          
          
        </a>
        <div class="product-grid-atb text-center"><i class="fa fa-
shopping-cart"></i><a class="" href="#">Add to Cart</a></div>
      </div>
    </div>
    <div class="col-sm-8 products-list-description">
      <div class="overflow-h">
        <ul class="list-inline overflow-h mb15">

```

```

        <li><h4 class="title-price"><a
href="shoppingdetail.aspx">Printed Summer Dress</a></h4></li>
        <li class="pull-right">
            <ul class="list-inline product-ratings">
                <li><i class="rating-selected fa fa-star"></i></li>
                <li><i class="rating-selected fa fa-star"></i></li>
                <li><i class="rating-selected fa fa-star"></i></li>
                <li><i class="rating fa fa-star"></i></li>
                <li><i class="rating fa fa-star"></i></li>
            </ul>
        </li>
    </ul>
    <div class="mb10">
        <span class="title-price">$40.00</span>
        <span class="title-price line-through">$45.00</span>
    </div>
    <p class="mb15">Lorem ipsum dolor sit amet, consectetur
adipiscing elit. Maecenas sollicitudin erat nec ornarevolu tpat. Etiam ut felis nec nisl eleifend
lobortis. Aenean nibh est, hendrerit non conva.</p>
    <ul class="list-inline add-to-wishlist mb15">
        <li class="product-wishlist">
            <i class="fa fa-heart"></i>
            <a href="#">Add to Wishlist</a>
        </li>
        <li class="product-wishlist">
            <i class="fa fa-exchange"></i>
            <a href="#">Add to Compare</a>
        </li>
    </ul>
    <button type="button" class="btn btn-product">Add to
Cart</button>
    </div>
</div>
</div>
</div>
</div>
<div class="products-list products-list-description-brd mb30">
    <div class="row">
        <div class="col-sm-4">
            <div class="product-grid-img">
                <ul class="product-labels">
                    <li>Sale</li>
                </ul>
                <a href="#" class="product-grid-img-thumbnail">
                    
                    
                </a>
                <div class="product-grid-atb text-center"><i class="fa fa-
shopping-cart"></i><a class="" href="#">Add to Cart</a></div>
            </div>
        </div>
        <div class="col-sm-8 products-list-description">
            <div class="overflow-h">

```

```

        <ul class="list-inline overflow-h mb15">
            <li><h4 class="title-price"><a
href="shoppingdetail.aspx">Printed Summer Dress</a></h4></li>
            <li class="pull-right">
                <ul class="list-inline product-ratings">
                    <li><i class="rating-selected fa fa-star"></i></li>
                    <li><i class="rating-selected fa fa-star"></i></li>
                    <li><i class="rating-selected fa fa-star"></i></li>
                    <li><i class="rating fa fa-star"></i></li>
                    <li><i class="rating fa fa-star"></i></li>
                </ul>
            </li>
        </ul>
        <div class="mb10">
            <span class="title-price">$60.00</span>
            <span class="title-price line-through">$112.00</span>
        </div>
        <p class="mb15">Lorem ipsum dolor sit amet, consectetur
adipiscing elit. Maecenas sollicitudin erat nec ornarevolu tpat. Etiam ut felis nec nisl eleifend
lobortis. Aenean nibh est, hendrerit non conva.</p>
        <ul class="list-inline add-to-wishlist mb15">
            <li class="product-wishlist">
                <i class="fa fa-heart"></i>
                <a href="#">Add to Wishlist</a>
            </li>
            <li class="product-wishlist">
                <i class="fa fa-exchange"></i>
                <a href="#">Add to Compare</a>
            </li>
        </ul>
        <button type="button" class="btn btn-product">Add to
Cart</button>
    </div>
</div>
</div>
</div>
</div>
</div>
<%--dynamic code ends here--%>

<!-- Start Pagination-->
    <div class="cd-pagination">
        <ul class="pagination">
            <li><a href="#"><i class="fa fa-angle-left" aria-hidden="true"></i></a></li>
            <li class="active"><a href="#">1</a></li>
            <li><a href="#">2</a></li>
            <li><a href="#">3</a></li>
            <li><a href="#">4</a></li>
            <li><a href="#">5</a></li>
            <li><a href="#"><i class="fa fa-angle-right" aria-hidden="true"></i></a></li>
        </ul>
    </div>
<!-- End Pagination-->
</div>
</div>
</div>
</div>

```

```
</div>
<!-- END MAIN CONTAIN -->
```

```
<!-- ./Content Bodey Here -->
```

```
</asp:Content>
<asp:Content ID="Content6" ContentPlaceHolderID="ContentFooter" Runat="Server">
</asp:Content>
```

```
***** ShoppingSearch.aspx.cs*****
```

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Text;
using DowBAL;
using DowBEntity;
```

```
public partial class ShoppingSearch : System.Web.UI.Page
{
```

```
    protected void Page_Load(object sender, EventArgs e)
```

```
    {
        string SrchStrng = Request.QueryString["SrchStrng"];
        Bindetails(SrchStrng);
    }
```

```
    private void Bindetails(string SrchStrng)
```

```
    {
        int stype = 1;
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        string pt = "prd";
        int pid = Convert.ToInt16(Request.QueryString["pgid"]);
        int GroupId = 5;// Convert.ToInt16(Clsusr.GroupId);
        string PreHtml = "";
        StringBuilder sb = new StringBuilder();
        sb.Append(" <div class='directory-list'>");
        bool slider = true;
        if (pid == 0)
        {
            pid = Convert.ToInt16(Request.QueryString["pid"]);
            pt = "prod";
            stype = 1;
        }
    }
```

```
    try
    {
```

```
        //IList<EteProductGroup> Lst = (new Products()).GetProductsListbypid("SearchList",
(decimal)pid);
```

```
        IList<ProductProductSeller> Lst = (new
Products()).GetProductSellerbySearch(SrchStrng,stype);
```

```

if (Lst != null && Lst.Count > 0)
{
    foreach (var dt in Lst)
    {
        string href = "", title = "Shopping";
        title = "Search ";
        href = "ShoppingDetail.aspx?id=" + dt.PSId;
        var list = new Products().GetProductSellerImage((int)dt.PSId, "").ToList();
        int cnt = list.Count();
        string path = list[0].ImagePath;
        string path1 = "";

        if (list.Count() > 1)
        {
            path1 = list[1].ImagePath;
        }
        if (dt.ImagePath == null)
        {
            continue;
        }
        if (slider)
        {
            sb.Append(" <div class='products-list products-list-description-brd mb30'>
\n");
            sb.Append(" <div class='row'> \n");
            sb.Append(" <div class='col-sm-4'> \n");
            sb.Append(" <div class='product-grid-img'> \n");
            sb.Append(" <ul class='product-labels'> \n");
            sb.Append(" <li>Sale</li> \n");
            sb.Append(" </ul> \n");
            sb.Append(" <a href='shoppingdetail.aspx?pid=" + dt.PSId + "'
class='product-grid-img-thumbnail'> \n");
            sb.Append("  \n");
            sb.Append("  \n");
            sb.Append(" </a> \n");
            sb.Append(" <div class='product-grid-atb text-center'><i class='fa fa-
shopping-cart'></i><a class=" href='shoppingdetail.aspx?pid=" + dt.PSId + "'>Add to
Cart</a></div> \n");
            sb.Append(" </div> \n");
            sb.Append(" </div> \n");
            sb.Append(" <div class='col-sm-8 products-list-description'> \n");
            sb.Append(" <div class='overflow-h'> \n");
            sb.Append(" <ul class='list-inline overflow-h mb15'> \n");
            sb.Append(" <li><h4 class='title-price'><a href='shoppingdetail.aspx?pid="
+ dt.PSId + "'> + dt.SellerDescription + "</a></h4></li> \n");
            sb.Append(" <li class='pull-right'> \n");
            sb.Append(" <ul class='list-inline product-ratings'> \n");
            sb.Append(" <li><i class='rating-selected fa fa-star'></i></li> \n");
            sb.Append(" <li><i class='rating-selected fa fa-star'></i></li> \n");
            sb.Append(" <li><i class='rating-selected fa fa-star'></i></li> \n");
            sb.Append(" <li><i class='rating fa fa-star'></i></li> \n");
            sb.Append(" <li><i class='rating fa fa-star'></i></li> \n");
            sb.Append(" </ul> \n");

```

```

        sb.Append(" </li> \n");
        sb.Append(" </ul> \n");
        sb.Append(" <div class='mb10'> \n");
        sb.Append(" <span class='title-price'>" + dt.Price.ToString() + "</span>
\n");
        sb.Append(" <span class='title-price line-through'>" + dt.Price.ToString() +
"</span> \n");
        sb.Append(" </div> \n");
        sb.Append(" <p class='mb15'>" + dt.SellerDescription + "</p> \n");
        sb.Append(" <ul class='list-inline add-to-wishlist mb15'> \n");
        sb.Append(" <li class='product-wishlist'> \n");
        sb.Append(" <i class='fa fa-heart'></i> \n");
        sb.Append("<li><a href='ServicesList.aspx?vendor=" + dt.SellerId + "'> +
"sold by:" + dt.SellerName + "</a> \n");
        sb.Append(" <a href='shoppingdetail.aspx?pid=" + dt.PSId + "'>Add to
Wishlist</a> \n");
        sb.Append(" </li> \n");
        sb.Append(" <li class='product-wishlist'> \n");
        sb.Append(" <i class='fa fa-exchange'></i> \n");
        sb.Append(" <a href='shoppingdetail.aspx?pid=" + dt.PSId + "'>Add to
Compare</a> \n");
        sb.Append(" </li> \n");
        sb.Append(" </ul> \n");
        sb.Append(" <button type='button' class='btn btn-product'>Add to
Cart</button> \n");
        sb.Append(" </div> \n");
        sb.Append(" </div> \n");
        sb.Append(" </div> \n");
        sb.Append(" </div> \n");
        sb.Append(" \n");
    }

    } //foreach code ends
}
if (slider)
{
    //sb.Append("</div>\n");

    PreHtml = sb.ToString();
    if (PreHtml.Length > 0)
    {
        //objlog.Trace_Log(4, "Gordersearchlist", "1", "SliderList", PreHtml);
        appvendors.InnerHtml = PreHtml;
    }
}
}
}
catch
{
    //Common.ShowMessageHtml("No records found", 0, true, dvMsgBox);
}
}

```

```

}
}
*****SubGroupsList.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="SubGroupsList.aspx.cs" Inherits="SubGroupsList" %>

<%@ Register Assembly="Infragistics4.Web.v14.2, Version=14.2.20142.2480,
Culture=neutral, PublicKeyToken=7dd5c3163f2cd0cb"
Namespace="Infragistics.Web.UI.GridControls" TagPrefix="ig" %>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
</asp:Content>

<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
<div class="container-fluid">
<div class="row-fluid" style="padding-top: 20px;">
<div class="span2" id="dvlogo" runat="server" style="display: block;">
<%----%>
</div>
<div class="span10">
<div class="row">
<div id="dvMsgBox" class="span12 ordersave" runat="server" style="display:
none;">
<asp:Label ID="Label1" runat="server" Text=""></asp:Label>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
</asp:Content>

<asp:Content ID="Content4" ContentPlaceHolderID="CPH_LeftSideBar" runat="Server">
<div id="LeftSideBar" runat="server" class="span2">
<asp:Table ID="Table1" runat="server">
</asp:Table>
</div>
</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="CPH_MainContent" runat="Server">

<div class="span10" id="MainContent" runat="server">
<div id="ShowPageHeader" class="row" style="display: block;" runat="server">
<div id="PageName" class="span12 customer" runat="server">
</div>
</div>

<div id="Div1" class="row" style="display: block;">
<div id="Div2" class="span12 customertable">

<!-- Main Content Comes here-->

```



```

        <%--<asp:ScriptManager ID="ScriptManager1"
runat="server"></asp:ScriptManager>--%>
        <ig:WebDataGrid ID="WDGrid_UserList" runat="server" CssClass="wdglist"
AutoGenerateColumns="False">
        <Columns>

                <ig:BoundDataField DataFieldName="Catname" Key="Catname">
                <Header Text="Category Name">
                </Header>
                </ig:BoundDataField>

                <ig:BoundDataField DataFieldName="pgroupname" Key="pgroupname">
                <Header Text="Group Name">
                </Header>
                </ig:BoundDataField>

                <ig:BoundDataField DataFieldName="pgroupdescription"
Key="pgroupdescription ">
                <Header Text="Description">
                </Header>
                </ig:BoundDataField>

                <ig:BoundDataField DataFieldName="CreatedOn" Key="CreatedOn"
Width="100px">
                <Header Text="Created On">
                </Header>
                </ig:BoundDataField>

                <ig:BoundDataField DataFieldName="isActive" Key="isActive">
                <Header Text="Is Active">
                </Header>
                </ig:BoundDataField>

                <ig:TemplateDataField Key="pgid" Width="60px">
                <ItemTemplate>
                <asp:HyperLink ID="pgid" runat="server" CssClass="btn btn-small btn-
primary" Text="Open"
                NavigateUrl='<% #Eval("pgid",
"~/ManageSubgroup.aspx?pageid=18&PageType=Edit&id={0}")%>'>
                </asp:HyperLink>
                </ItemTemplate>
                <Header Text="Open" />
                </ig:TemplateDataField>
        </Columns>
        <Behaviors>
                <ig:Paging CurrentPageLinkCssClass="Current" PageLinkCssClass="Link"
PagerCssClass="Pager" PagerMode="NumericFirstLast" PageSize="15" QuickPages="3"
FirstPageText="First" LastPageText="Last" NextPageText="Next"
PreviousPageText="Previous" PagerAppearance="bottom">
                </ig:Paging>

                <ig:Filtering>
                </ig:Filtering>

```

```

        <ig:Sorting Enabled="true">
        </ig:Sorting>
    </Behaviors>
</ig:WebDataGrid>
    <asp:SqlDataSource ID="SqlDataSource1" runat="server"
ConnectionString="<%%$ ConnectionStrings:DowConectingString %>"
SelectCommand="SELECT [UserId], [UserName], [FirstName], [LastName], [EmailID],
[ContactNum], [CreatedOn], [Status] FROM [tblUser]"></asp:SqlDataSource>

    </div>
</div>
</div>
</asp:Content>

```

***** SubGroupsList.aspx.cs*****

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBEntity;
using DowBAL;
using System.Data;

public partial class SubGroupsList:System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        //if (!IsPostBack)
        //{
            ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
            if (Clsusr != null)
            {
                string pageid = "6";// Request.QueryString["pageid"];
                ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
Convert.ToInt32(Clsusr.GroupId));
                string msg = Request.QueryString["msg"];
                if (msg != null)
                {
                    Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
                }
                if (Clsusr.GroupId == "1")
                {
                    Bindetails();
                }
            }
            else
            {
                Response.Redirect("MLB_Login.aspx");
            }
        }

        private void ManageNavigations(int pageid, int userid, int groupid)

```

```

{
    bool IsLeftBar = false;
    IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
myBtnClickHandler);

    if (IsLeftBar == true)
    {
        LeftSideBar.Attributes["class"] = "span2";
        MainContent.Attributes["class"] = "span10";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;"); //style="display:
block;"
        PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
    else
    {
        LeftSideBar.Attributes["class"] = "";
        MainContent.Attributes["class"] = "span12";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
        PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
}

```

// This is Event Handler

```

protected void myBtnClickHandler(object sender, EventArgs e)
{

    LinkButton lnk = (LinkButton)sender;
    string ID = lnk.ID;

    DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
    if (ds != null)
    {
        if (ds.Tables[0].Rows.Count > 0)
        {
            DataTable dtdll = ds.Tables[0];
            foreach (DataRow drnew in dtdll.Rows)
            {
                switch (drnew["lbtncode"].ToString())
                {
                    case "SAVE":
                        // Call SAVE Method
                        break;
                    case "URL":
                        Response.Redirect(drnew["pageurl"].ToString());
                        break;
                    case "AUDIT":
                        Response.Redirect(drnew["pageurl"].ToString() + "&frompage=" +
Request.QueryString["pageid"]);
                        break;
                    case "CANCEL":
                        Response.Redirect("Home.aspx?pageid=1");
                        break;
                }
            }
        }
    }
}

```

```

        case "NEW":
            Response.Redirect("ManageSubgroup.aspx?pageid=18&pagetype=Add");
            break;
        default:
            break;
    }
}
}
}
}

private void Binddetails()
{
    //try
    //{

        List<EteProductGroup> Lst = (new Products()).GetProductGroups("");

        if (Lst != null && Lst.Count > 0)
        {
            WDGrid_UserList.DataSource = Lst;
            WDGrid_UserList.DataBind();
            WDGrid_UserList.Visible = true;
        }
        else
        {
            WDGrid_UserList.Visible = false;
        }
    }
    //protected override void OnInit(EventArgs e)
    //{
    //    base.OnInit(e);

    //    if (this.Master is MainMaster)
    //    {
    //        ((MainMaster)this.Master).MasterControlClicked += new
    EventHandler(ContentControlClicked);
    //    }
    //}
    private void ContentControlClicked(object sender, EventArgs e)
    {

        string pageid = "pageid=" + Session["pageid"].ToString();
        Response.Redirect("~/Managedealer.aspx?PageType=Add&dealername=0&" +
pageid);
    }
}
}
***** UOMTypesList.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="UOMTypesList.aspx.cs" Inherits="UOMTypesList" %>

```

```
<%--<%@ Register Assembly="Infragistics4.WebUI.WebSpellChecker.v14.2,
Version=14.2.20142.1028, Culture=neutral, PublicKeyToken=7dd5c3163f2cd0cb"
Namespace="Infragistics.WebUI.WebDataInput" TagPrefix="igtxt" %>--%>
<%@ Register Assembly="Infragistics4.Web.v14.2, Version=14.2.20142.1028,
Culture=neutral, PublicKeyToken=7dd5c3163f2cd0cb"
Namespace="Infragistics.Web.UI.GridControls" TagPrefix="ig" %>
```

```
<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
</asp:Content>
```

```
<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
<div class="container-fluid">
<div class="row-fluid" style="padding-top: 20px;">
<div class="span2" id="dvlogo" runat="server" style="display: block;">
<%----%>
</div>
<div class="span10">
<div class="row">
<div id="dvMsgBox" class="span12 ordersave" runat="server" style="display:
none;">
<asp:Label ID="Label1" runat="server" Text=""></asp:Label>
</div>
</div>
</div>
</div>
</div>
</div>
</asp:Content>
```

```
<asp:Content ID="Content4" ContentPlaceHolderID="CPH_LeftSideBar" runat="Server">
<div id="LeftSideBar" runat="server" class="span2">
<asp:Table ID="Table1" runat="server">
</asp:Table>
</div>
</asp:Content>
```

```
<asp:Content ID="Content2" ContentPlaceHolderID="CPH_MainContent" runat="Server">
<div class="span10" id="MainContent" runat="server">
<div id="ShowPageHeader" class="row" style="display: block;" runat="server">
<div id="PageName" class="span12 customer" runat="server">
</div>
</div>
<div id="Div1" class="row" style="display: block;">
<div id="Div2" class="span12 customertable">
<!-- Main Content Comes here-->
<%--<asp:ScriptManager ID="ScriptManager1"
runat="server"></asp:ScriptManager>--%>
```

```

        <ig:WebDataGrid ID="WDGrid_UserList" runat="server" CssClass="wdglist"
AutoGenerateColumns="False">
    <Columns>
        <ig:BoundDataField DataFieldName="UomShort" Key="UomShort">
            <Header Text="Name">
                </Header>
            </ig:BoundDataField>

            <ig:BoundDataField DataFieldName="UomDescription"
Key="UomDescription ">
                <Header Text="Description">
                    </Header>
                </ig:BoundDataField>

            <ig:BoundDataField DataFieldName="CreatedOn" Key="CreatedOn"
Width="100px">
                <Header Text="Created On"></Header>

            </ig:BoundDataField>

            <ig:TemplateDataField Key="UomId" Width="60px">
                <ItemTemplate>
                    <asp:HyperLink ID="UomId" runat="server" CssClass="btn btn-small
btn-primary" Text="Open"
                    NavigateUrl='<% #Eval("UomId",
"~/ManageUOMTypes.aspx?pageid=14&PageType=Edit&id={0}")%>'>
                </asp:HyperLink>
                </ItemTemplate>
                <Header Text="Open" />
            </ig:TemplateDataField>
        </Columns>
        <Behaviors>
            <ig:Paging CurrentPageLinkCssClass="Current" PageLinkCssClass="Link"
PagerCssClass="Pager" PagerMode="NumericFirstLast" PageSize="15" QuickPages="3"
FirstPageText="First" LastPageText="Last" NextPageText="Next"
PreviousPageText="Previous" PagerAppearance="bottom">
                </ig:Paging>

            <ig:Filtering>
                </ig:Filtering>
            <ig:Sorting Enabled="true">
                </ig:Sorting>
        </Behaviors>
    </ig:WebDataGrid>
    <asp:SqlDataSource ID="SqlDataSource1" runat="server"
ConnectionString="<%= $ ConnectionStrings:DownConectingString %>"
SelectCommand="SELECT [UserId], [UserName], [FirstName], [LastName], [EmailID],
[ContactNum], [CreatedOn], [Status] FROM [tblUser]"></asp:SqlDataSource>

    </div>
</div>
</div>
</asp:Content>

```

```

*****UOMTypesList.aspx.cs *****
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBEntity;
using DowBAL;
using System.Data;

public partial class UOMTypesList:System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        //if (!IsPostBack)
        //{
            ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
            if (Clsusr != null)
            {
                string pageid = Request.QueryString["pageid"];
                ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
                Convert.ToInt32(Clsusr.GroupId));
                string msg = Request.QueryString["msg"];
                if (msg != null)
                {
                    Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
                }
                if (Clsusr.GroupId == "1")
                {
                    Bindetails();
                }
            }
            else
            {
                Response.Redirect("MLB_Login.aspx");
            }
        }

        private void ManageNavigations(int pageid, int userid, int groupid)
        {
            bool IsLeftBar = false;
            IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
            myBtnClickHandler);

            if (IsLeftBar == true)
            {
                LeftSideBar.Attributes["class"] = "span2";
                MainContent.Attributes["class"] = "span10";
                ShowPageHeader.Attributes.CssStyle.Add("display", "block;"); //style="display:
                block;"
                PageName.InnerText = new
                tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
            }
        }
    }
}

```

```

else
{
    LeftSideBar.Attributes["class"] = "";
    MainContent.Attributes["class"] = "span12";
    ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
    PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
}
}

// This is Event Handler
protected void myBtnClickHandler(object sender, EventArgs e)
{
    LinkButton lnk = (LinkButton)sender;
    string ID = lnk.ID;

    DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
    if (ds != null)
    {
        if (ds.Tables[0].Rows.Count > 0)
        {
            DataTable dtdll = ds.Tables[0];
            foreach (DataRow drnew in dtdll.Rows)
            {
                switch (drnew["lbtncode"].ToString())
                {
                    case "SAVE":
                        // Call SAVE Method
                        break;
                    case "URL":
                        Response.Redirect(drnew["pageurl"].ToString());
                        break;
                    case "AUDIT":
                        Response.Redirect(drnew["pageurl"].ToString() + "&frompage=" +
Request.QueryString["pageid"]);
                        break;
                    case "CANCEL":
                        Response.Redirect("Home.aspx?pageid=1");
                        break;
                    case "NEW":
                        Response.Redirect("ManageUOMTypes.aspx?pageid=4&pagetype=Add");
                        break;
                    default:
                        break;
                }
            }
        }
    }
}
}

```



```

private void Bindetails()
{
    //try
    //{

        List<Etetbluom> Lst = (new Products()).GetProducttbluoms("");

        if (Lst != null && Lst.Count > 0)
        {
            WDGrid_UserList.DataSource = Lst;
            WDGrid_UserList.DataBind();
            WDGrid_UserList.Visible = true;
        }
        else
        {
            WDGrid_UserList.Visible = false;
        }
    }
    //protected override void OnInit(EventArgs e)
    //{
    //    base.OnInit(e);

    //    if (this.Master is MainMaster)
    //    {
    //        ((MainMaster)this.Master).MasterControlClicked += new
EventHandler(ContentControlClicked);
    //    }
    //}
    private void ContentControlClicked(object sender, EventArgs e)
    {

        string pageid = "pageid=" + Session["pageid"].ToString();
        Response.Redirect("~/Managedealer.aspx?PageType=Add&dealername=0&" +
pageid);
    }
}
*****UserDetail.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/ShoppingMasterPage.master"
AutoEventWireup="true" CodeFile="UserDetail.aspx.cs" Inherits="UserDetail" %>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
Runat="Server">

</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentMainPage" Runat="Server">
</asp:Content>
<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
<div class="container-fluid">
<div class="row-fluid" style="padding-top: 20px;">
<div class="span2" id="dvlogo" runat="server" style="display: block;">
<%----%>
</div>

```

```

        <div class="span10">
            <div class="row">
                <div id="dvMsgBox" class="span12 ordersave" runat="server" style="display:
none;">
                    <asp:Label ID="Label1" runat="server" Text=""></asp:Label>
                </div>
            </div>
        </div>
    </div>
</asp:Content>
<asp:Content ID="Content5" ContentPlaceHolderID="CPH_MainContent" Runat="Server">
    <!-- Content Bodey Here -->

```

```

<section class="main-contain dip-style">
    <div class="container">
        <div class="login-area mb30">
            <div class="login-header">
                <div class="login-details ">
                    <ul class="nav nav-tabs navbar-right sm-6">
                        <li class="active sm-6"><a data-toggle="tab"
href="#description">User Info</a></li>
                        <li class="sm-6"><a data-toggle="tab" href="#Profile">Profile
Details</a></li>
                        <li class="sm-6"><a data-toggle="tab"
href="#Address">Address</a></li>
                        <li class="sm-6"><a data-toggle="tab"
href="#Images">Images</a></li>
                    </ul>
                </div>
            </div>

            <div class="tab-content">
                <!-- Description -->
                <div class="tab-pane fade in active" id="description">
                    <div class="col-lg-8 col-md-8 col-sm-12 " style="background-color:
#f8f9fa;">
                        <div class="row-fluid">
                            <div class="col-lg-8 col-md-8 col-sm-12" style="text-align:
left;">BASIC INFORMATION</div>
                        </div>

                        <div class="row-fluid">
                            <div class="col-lg-8 col-md-8 col-sm-12 detailview"
style="background-color: #ffffff; border: 1px solid #eefef; padding: 15px 5px;"> <%--border-
top: none;--%>

                            <!--<form class="form-horizontal">-->
                            <div class="control-group">
                                <table class="table table-striped table-detail col-lg-8 col-md-8
col-sm-12">

                                    <tbody>

```

```

        <tr>
            <td class=" col-lg-2 col-md-2 col-sm-
3"><strong>Email/User Id :*</strong></td>
            <td align="left" class="auto-style1">
                <asp:TextBox ID="txt_username" runat="server"
MaxLength="50" TextMode="Email" class="col-lg-9 col-md-9 col-sm-12 pull-
left"></asp:TextBox>
            </td>
        </tr>
        <tr>
            <td align="right" class=" col-lg-3 col-md-3 col-sm-
3"><strong>Name :*</strong></td>
            <td align="left" ">
                <asp:TextBox ID="txt_FirstName" runat="server"
MaxLength="100" onkeypress="return validatealpha(event)" class="col-lg-9 col-md-9 col-sm-
12 pull-left"></asp:TextBox>
            </td>
        </tr>
        <tr>
            <td align="right" class=" col-lg-3 col-md-3 col-sm-
3"><strong>Company Name :</strong></td>
            <td align="left">
                <asp:TextBox ID="txt_LastName" runat="server"
MaxLength="40" onkeypress="return validatealpha(event)" class="col-lg-9 col-md-9 col-sm-
12 pull-left"></asp:TextBox>
            </td>
        </tr>
        <tr>
            <td align="right" class=" col-lg-3 col-md-3 col-sm-
3"><strong>Country:</strong></td>
            <td align="left">
                <asp:DropDownList ID="ddl_country"
runat="server"></asp:DropDownList></td>
            </tr>
        <tr>
            <td align="right" class=" col-lg-3 col-md-3 col-sm-
3"><strong>Mobile :*</strong></td>
            <td align="left">
                <asp:TextBox ID="txt_mobile" runat="server"
onkeypress="return isNumberKey(event)" MaxLength="12"
TextMode="Phone"></asp:TextBox>
            </td>
        </tr>
        <tr>
            <td align="right" class=" col-lg-3 col-md-3 col-sm-
3"><strong>LandLine :*</strong></td>

```

```

        <td width="25%" align="left">
            <asp:TextBox ID="txt_landline" runat="server"
onkeypress="return isNumberKey(event)" MaxLength="12"
TextMode="Phone"></asp:TextBox>
        </td>
    </tr>

    <tr>
        <td align="right" class=" col-lg-3 col-md-3 col-sm-
3"><strong>Role :*</strong></td>
        <td width="25%" align="left">
            <asp:DropDownList ID="ddl_GroupId"
runat="server" Enabled="false"></asp:DropDownList>
        </td>
    </tr>
    <tr>
        <td align="right" class=" col-lg-3 col-md-
3 col-sm-3"><strong>Password:*</strong></td>
        <td width="25%" align="left">
            <asp:TextBox ID="txt_Password" runat="server"
MaxLength="40" TextMode="Password"></asp:TextBox>
        </td>
    </tr>

    <tr>
        <td align="right" class=" col-lg-3 col-md-3 col-sm-
3">&nbsp;<strong>Password:*</strong></td>
        <td width="25%" align="left">
            <asp:TextBox ID="txt_Password1" runat="server"
MaxLength="40" TextMode="Password"></asp:TextBox>
            <asp:TextBox ID="email_name" runat="server" MaxLength="50"
TextMode="Email" Visible="false"></asp:TextBox>
        </td>
    </tr>
</tbody>
</table>

</div>

</div>

</div>
<!-- End Description -->

<!-- Size -->

```

```

<div id="Profile" class="tab-pane">
  <div class="col-lg-8 col-md-8 col-sm-12" style="background-color:
#f8f9fa;">
    <div class="row-fluid">
      <!--Profile Account----->
      <div class="row-fluid">
        <div class="col-lg-8 col-md-8 col-sm-12" style="text-align:
left;">Profile</div>
      </div>
      <div class="row-fluid">
        <div class="col-lg-8 col-md-8 col-sm-12" style="background-color:
#ffffff; border: 1px solid #eefef; border-top: none; padding: 5px;">
          <table class="table table-striped table-detail col-lg-8 col-md-8
col-sm-12">
            <tbody>
              <tr>
                <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">User
Name :</td>
                <td>
                  <asp:DropDownList ID="ddl_users"
runat="server"></asp:DropDownList>
                </td>
              </tr>
              <tr>
                <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Company
Description :</td>
                <td>
                  <asp:TextBox ID="Details" runat="server"
Height="100px" Rows="7" style="resize:both" TextMode="MultiLine" MaxLength="250"
class="col-lg-9 col-md-9 col-sm-12 pull-left"></asp:TextBox>
                </td>
              </tr>
              <tr>
                <td class="col-lg-3 col-md-3 col-sm-3 auto-style1"
dir="auto">Business Type</td>
                <td class="auto-style2">
                  <asp:TextBox ID="txt_BusinessType" runat="server"
MaxLength="40" class="col-lg-9 col-md-9 col-sm-12 pull-left"></asp:TextBox>
                </td>
              </tr>
              <tr>
                <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Legal
Structure:</td>
                <td>
                  <asp:DropDownList ID="ddl_legalstr"
runat="server"></asp:DropDownList>
                </td>
              </tr>
              <tr>
                <td>
            
```

```

Established :</td>
        <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Year
        <td>
            <asp:TextBox ID="txt_yrest" runat="server"
MaxLength="40" ></asp:TextBox>
        </td>
    </tr>
    <tr>
        <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">No of
Employees</td>
        <td>
            <asp:TextBox ID="txt_noempl" runat="server"
MaxLength="40" ></asp:TextBox>
        </td>
    </tr>
    <tr>
        <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Web-
URL</td>
        <td>
            <asp:TextBox ID="txt_website" runat="server"
MaxLength="40" ></asp:TextBox>
        </td>
    </tr>
    <tr>
        <td class="col-lg-3 col-md-3 col-sm-3"
dir="auto">Turnover Per Year:</td>
        <td >
            <asp:TextBox ID="txt_turnover" runat="server" MaxLength="40"
class="col-lg-3 col-md-3 col-sm-3" ></asp:TextBox>
            <%-- </td><td class="col-lg-3 col-md-3 col-sm-3" >
            --%>
                <asp:Label ID="curr" runat="server" class="col-lg-
3 col-md-3 col-sm-3" align="right" dir="right">In Currency:</asp:Label>
            <%--
                </td><td class="col-lg-3 col-md-3 col-sm-3" >
                --%>
                    <asp:DropDownList ID="ddl_currency" runat="server"
class="col-lg-3 col-md-3 col-sm-3" ></asp:DropDownList>
            </td>
        </tr>
    <tr>
        <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Quality
Assurance Accreditation :</td>
        <td>
            <asp:TextBox ID="txt_qa" runat="server"
MaxLength="40" ></asp:TextBox>
        </td>
    </tr>
    <tr>
        <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">
    </td>
    </tr>

```

```
  |
```

```

        </table>

    </div>
    <!-- Bank Account END -->
</div>

</div>
</div>

</div>
<!-- Size -->

<!-- Description -->
<div class="tab-pane media-block" id="Address">
    <div class="row-fluid">
        <div id="addlist" runat="server" class="col-lg-12 col-md-12 col-sm-12 "
span3="" style="text-align:center">Addresses</div>
        <asp:Button ID="btn_SaveAddress" runat="server" CssClass="btn btn-primary btn-
md btn-rounded" Text="Save Address" OnClick="btn_SaveAddress_Click" />
        <asp:Button ID="btn_addaddress" runat="server" CssClass="btn btn-primary btn-md
btn-rounded" Text="Add Address" OnClick="btn_addaddress_Click" />

    </div>
    <div class="row-fluid">
        <div class="col-lg-8 col-md-8 col-sm-12" style="background-color:
#ffffff; border: 1px solid #eefef; border-top: none; padding: 15px 5px;">
            <table id="Addressdet" class="table table-striped table-detail col-
lg-8 col-md-8 col-sm-12" runat="server" visible="true">
                <tbody>
                    <tr>
                        <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Address
Type :</td>
                        <td>
                            <asp:DropDownList ID="ddl_addtype"
runat="server"></asp:DropDownList>
                        </td>
                    </tr>
                    <tr>
                        <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Address1 :</td>
                        <td>
                            <asp:TextBox ID="txt_add1" runat="server" MaxLength="100"
Width="250px"></asp:TextBox>
                        </td>
                    </tr>
                    <tr>
                        <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Address2 :</td>
                        <td>
                            <asp:TextBox ID="txt_add2" runat="server" MaxLength="100"
Width="250px"></asp:TextBox>

```



```

        </td>
    </tr>
    <tr>
        <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Address3 :</td>

        <td>
            <asp:TextBox ID="txt_add3" runat="server" MaxLength="100"
Width="250px"></asp:TextBox>

        </td>
    </tr>

    <tr>
        <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">City :</td>

        <td>
            <asp:TextBox ID="txt_city" runat="server" MaxLength="100"
Width="250px"></asp:TextBox>

        </td>
    </tr>
    <tr>
        <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Pincode :</td>

        <td>
            <asp:TextBox ID="txt_pincode" runat="server" MaxLength="100"
Width="250px"></asp:TextBox>

        </td>
    </tr>
    <tr>
        <td align="right" class="label_txt"><strong>Country:</strong></td>
        <td width="25%" align="left">
            <asp:DropDownList ID="ddl_country1"
runat="server"></asp:DropDownList></td>
    </tr>

    <tr>
        <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Contact No :</td>

        <td>
            <asp:TextBox ID="txt_contactno" runat="server" MaxLength="100"
Width="250px"></asp:TextBox>

        </td>
    </tr>
</tbody>

</table>

<!--<form class="form-horizontal">-->

```

```

        </div>

        </div>
<!-- End Description -->
</div>
<div class="tab-pane media-block" id="Images">
    <div class="span12 customertable" style="background-color: #f8f9fa;">
        <div class="row-fluid">

            <!--Notes----->
            <div class="row-fluid">
                <div class="col-lg-12 col-md-12 col-sm-12" style="text-align:
left;">Pictures / Images</div>
                <table>
                    <tr>
                        <td>
                            <asp:image ToolTip = "ASP Image Control" ID="Image2" runat="server"
ImageUrl = "ManageProducts.aspx?ImageID=1" Height="156px" Width="174px"
Visible="true"></asp:image> </td>
                            <td> <asp:image ToolTip = "ASP Image Control" ID="Image3" runat="server"
ImageUrl = "ManageProducts.aspx?ImageID=2" Height="156px" Width="174px"
Visible="true"></asp:image>
                                </td><td> <asp:image ToolTip = "ASP Image Control" ID="Image4"
runat="server" ImageUrl = "ManageProducts.aspx?ImageID=3" Height="156px"
Width="174px" Visible="true"> </asp:image>
                                    </td><td> <asp:image ToolTip = "ASP Image Control" ID="Image5"
runat="server" ImageUrl = "ManageProducts.aspx?ImageID=4" Height="156px"
Width="174px" Visible="true"> </asp:image>

                                </td>
                            </tr>

                        <tr>
                            <td class="col-lg-1 col-md-1 col-sm-1" dir="auto">Add Images</td>
                        </tr>
                    </table>
                    <div class="row-fluid">

                    </div>
                </div>

                <!--Notes Ends----->

```

```

        </div>
    </div>
    <!-- End Description -->
    </div>
</div>
    </div>

<!-- JQUERY LIBS -->
<script type="text/javascript" src="js/jquery.min.js"></script>
<!-- BOOTSTRAP -->
<script type="text/javascript" src="js/bootstrap.min.js"></script>
<!-- OWL CAROUSEL -->
<script type="text/javascript" src="plugins/owl-carousel/owl.carousel.min.js"></script>
<!-- FLEXSLIDER -->
<script type="text/javascript" src="plugins/FlexSlider/jquery.flexslider-min.js"></script>
<!-- MAGNIFIC POPUP-->
<script type="text/javascript" src="js/jquery.magnific-popup.min.js"></script>
<!-- product-->
<script type="text/javascript" src="js/product.js"></script>
<!-- MAGNIFIC POPUP -->
<script type="text/javascript" src="js/magnific-popup.min.js"></script>
<!-- slick-->
<script type="text/javascript" src="js/slick.js"></script>
<!-- slick-slider-->
<script type="text/javascript" src="js/slick-slider.js"></script>
<!-- plugins-->
<script type="text/javascript" src="js/plugins.min.html"></script>
<!-- CUSTOM-->
<script type="text/javascript" src="js/custom.js"></script>
</div></section>
</asp:Content>
<%--<asp:Content ID="Content6" ContentPlaceHolderID="ContentFooter" Runat="Server">
</asp:Content>--%>

```

```

***** UserDetail.aspx.cs*****

```

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Text;
using DowBAL;
using DowBEntity;
using DowDAL;
using System.Data;
using System.Data.SqlClient;
using System.Configuration;
using System.Drawing;
using System.Web.Security;
using System.IO;

public partial class UserDetail : System.Web.UI.Page
{
    public string PageType;

```

```

string pageid;
int UserId;
int PrId = 0;
DDLogTracer objlog = new DDLogTracer();

protected void Page_Load(object sender, EventArgs e)
{
    CIsLoggedInUser CIsusr = (CIsLoggedInUser)Session["LoggedInUser_Specs"];
    if (CIsusr != null)
    {
        PageType = Page.Request.QueryString["PageType"];
        if (PageType == null)
        {
            PageType = "Edit";
        }
        Session["PageType"] = PageType;
        pageid = "4";// Request.QueryString["pageid"].ToString();
        //ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(CIsusr.UserId),
Convert.ToInt32(CIsusr.GroupId), PageType);

        //if (!this.IsPostBack)
        {
            PopDlIs ObjPD = new PopDlIs();
            // ObjPD.BindDropDowns(ddl_businessstype, (new
PopDlIs()).GetBusinessType());
            ObjPD.BindDropDowns(ddl_legalstr, (new PopDlIs()).GetLegalstr());
            ObjPD.BindDropDowns(ddl_currency, (new PopDlIs()).GetCurrency());
            ObjPD.BindDropDowns(ddl_uom, (new PopDlIs()).Gettbluom());
            ObjPD.BindDropDowns(ddl_users, (new PopDlIs()).GetUserCompany());
            ObjPD.BindDropDowns(ddl_addtype, (new PopDlIs()).GetAddType());
            ObjPD.BindDropDowns(ddl_country, (new PopDlIs()).GetCountries());
            ObjPD.BindDropDowns(ddl_country1, (new PopDlIs()).GetCountries());
            ObjPD.BindDropDowns(ddl_GroupId, (new
PopDlIs()).GetGroups(Convert.ToInt32(CIsusr.UserId), Convert.ToInt32(CIsusr.GroupId)));

            Addressdet.Visible = true;

            string ProductId = Page.Request.QueryString["Id"];
            UserId = Convert.ToInt16(CIsusr.UserId);
            if (UserId > 0)
            {
                //-----user infor
                EtetblUser lst = (new tblUsers()).getuserbyid(UserId);
                txt_username.Text = lst.UserId.ToString();
                txt_username.Text = lst.UserName;
                txt_FirstName.Text = lst.FirstName;
                txt_LastName.Text = lst.LastName;
                email_name.Text = lst.EmailID;
                txt_mobile.Text = lst.ContactNum;
                ddl_country.SelectedIndex = Convert.ToInt16(lst.Region);
                txt_mobile.Text = lst.ContactNum;
                ddl_GroupId.SelectedValue = lst.GroupId.ToString();
                txt_Password.Text = new BalCommon().EncryptPwd(lst.Password.ToString());
                txt_Password1.Text = new BalCommon().EncryptPwd(lst.Password.ToString());
            }
        }
    }
}

```

```

//-----user infor ends
ddl_users.SelectedValue = UserId.ToString();
ddl_users.Enabled = false;
Etetbluser_profile Prd = new tblUsers().getuserprofilebyid(UserId);
if (Prd != null && Prd.userid > 0)
{
    Details.Text = Prd.Company_Description;
    txt_BusinessType.Text = Prd.Business_Type;
    ddl_uom.SelectedIndex = (int)Prd.UOMId;
    ddl_legalstr.SelectedValue = Prd.Legal_Structure;
    ddl_currency.SelectedIndex = (int)Prd.CID;
    txt_yrest.Text = Prd.Year_Established;
    txt_noempl.Text = Prd.Number_of_Employees;
    txt_turnover.Text = Prd.Turnover_per_Year;
    ddl_currency.SelectedIndex = Prd.CID;
    txt_qa.Text = Prd.Quality_Assurance_Accreditation;
    txt_regno.Text = Prd.Registration_Number;
    txt_dunsno.Text = Prd.D_U_N_S_Number;
    txt_special.Text = Prd.Specializing_in;
    txt_prodcap.Text = Prd.Monthly_Production_Capacity;
    txt_website.Text = Prd.Company_Website_URL;
    if (Prd.Company_logo != null)
    {
        // string conn =
ConfigurationManager.ConnectionStrings["DowConectingString"].ConnectionString;
        // SqlConnection connection = new SqlConnection(conn);
        // string sql = "SELECT
name='~\\images\\FILE;' + convert(varchar(5),userid) + '.jpg', ContentType='IMAGE/JPEG',
Images=company_logo FROM tbluser_profile WHERE UserId = @PID";
        // SqlCommand cmd = new SqlCommand(sql, connection);
        // cmd.CommandType = CommandType.Text;
        // cmd.Parameters.AddWithValue("@PID", UserId);

        // connection.Open();
        // SqlDataReader sqlDataReader = cmd.ExecuteReader();
        // int sno = 0;
        // Image1.Visible = false;
        // List<ImageItem> il = new List<ImageItem>();
        // while (sqlDataReader.Read())
        // {
        //     sno++;
        //     byte[] bytes = (byte[])sqlDataReader["Images"].ToString();
        //     string base64String = Convert.ToBase64String(bytes, 0, bytes.Length);
        //     ImageItem wimg = new ImageItem();
        //     Image1.Visible = true;

        //     wimg.ImageUrl = "data:image/png;base64," + base64String;
        //     il.Add(new ImageItem { Key = sno.ToString(), ImageUrl =
"data:image/png;base64," + base64String, ToolTip = "test image" });

        //     Image1.Visible = true;
        //     Image1.ImageUrl = "data:image/png;base64," + base64String;

        // }
    }
}

```

```

        // connection.Close();
        // // sda.Dispose();
        // connection.Dispose();
        //}
    }
    //
}
StringBuilder sb = new StringBuilder();

List<Etetbluser_address> Lst = (new tblUsers()).getusersaddressbyuserid(UserId,
""");
sb.Append("<table class='table table-striped table-detail col-lg-12 col-md-12 col-
sm-12' runat='server'>\n");
sb.Append("    <thead class='mdb-color darken-3'> \n");
sb.Append("    <tr class='text-white'> \n");
sb.Append("    <th style='width:5%'>Default</th> \n");
//sb.Append("    <th scope='row'> \n");
sb.Append("    <th><i class='fa fa-home'></i>Type</th> \n");
sb.Append("    <th style='width:15%'>Name</th> \n");
sb.Append("    <th style='width:20%'>Addr1</th> \n");
sb.Append("    <th style='width:20%'>Addr2</th> \n");
sb.Append("    <th style='width:20%'>Addr3</th> \n");
sb.Append("    <th style='width:15%'>City</th> \n");
sb.Append("    <th style='width:10%'>Pincode</th> \n");
sb.Append("    <th style='width:10%'>Country</th> \n");
sb.Append("    <th style='width:15%'>Contact No</th> \n");
sb.Append("    <th style='width:5%'>Action</th> \n");
sb.Append("    </tr> \n");
sb.Append("    </thead> \n");
string chk;
foreach (var add in Lst)
{
    if (add.IsDefault) { chk = "checked"; } else { chk = ""; }
    sb.Append("    <tr> \n");
    sb.Append("    <th scope='row'> \n");
    sb.Append("    <input class='form-check-input' type='checkbox' id='checkbox"
+ add.address_type + "' +chk+"> \n");
    sb.Append("    <label for='checkbox' + add.address_type + "' class='label-
table'></label>\n");
    sb.Append("    </th> \n");
    sb.Append("    <td>" + add.address_typedesc + "</td> \n");
    sb.Append("    <td>" + Clsusr.UserName + "</td> \n");
    sb.Append("    <td>" + add.address1 + "</td> \n");
    sb.Append("    <td>" + add.address2 + "</td> \n");
    sb.Append("    <td>" + add.address3 + "</td> \n");
    sb.Append("    <td>" + add.city + "</td> \n");
    sb.Append("    <td>" + add.pincode + "</td>\n");
    sb.Append("    <td>" + add.countryname + "</td> \n");
    sb.Append("    <td>" + add.ContactNo + "</td> \n");
    sb.Append("    <td>X</td> \n");
}

```

```

        //sb.Append(" <td><a href='/postproducts.aspx?userid=" + add.userid +
"&pagetype=Edit' class='btn btn-outline-danger waves-effect btn-sm font-weight-bold'><i
class='fas fa-edit'></i> &nbsp; Edit</a></td>");
        sb.Append(" </tr></Table> \n");

    }
    sb.Append(" </tbody> \n");
    objlog.Trace_Log(1, "home", "1", "menu", "\n" + sb.ToString());

    addlist.InnerHtml = sb.ToString();
    DataTable dt = new tblUsers().GetUserImagesDTbyUid(UserId, "");
    //download(dt);
    List<EtetblUserImages> Pics = (new tblUsers()).getuserImagesbyuserid(UserId,
    "");
    foreach (var pic in Pics)
    {

        switch (pic.srno)
        {
            case 1:
                Image2.ImageUrl = pic.ImageUrl;
                Image2.Visible = true;

                break;
            case 2:
                Image3.ImageUrl = pic.ImageUrl;
                Image3.Visible = true;
                break;
            case 3:
                Image4.ImageUrl = pic.ImageUrl;
                Image4.Visible = true;
                break;
            case 4:
                Image5.ImageUrl = pic.ImageUrl;
                Image5.Visible = true;
                break;
            default:
                break;
        }
    }
}
}
}
}
private DataTable GetData(SqlCommand cmd)
{
    DataTable dt = new DataTable();
    String strConnString =
System.Configuration.ConfigurationManager.ConnectionStrings["DowConectingString"].Con
nectionString;
    SqlConnection con = new SqlConnection(strConnString);
    SqlDataAdapter sda = new SqlDataAdapter();
    cmd.CommandType = CommandType.Text;

```

```

cmd.Connection = con;
try
{
    con.Open();
    sda.SelectCommand = cmd;
    sda.Fill(dt);
    return dt;
}
catch
{
    return null;
}
finally
{
    con.Close();
    sda.Dispose();
    con.Dispose();
}
}
public byte[] GetPhtoStream()
{
    byte[] bufferPhoto = new byte[FileUpload1.PostedFile.ContentLength];
    Stream photoStream = FileUpload1.PostedFile.InputStream;
    photoStream.Read(bufferPhoto, 0, FileUpload1.PostedFile.ContentLength);

    return bufferPhoto;
}
protected void SaveAdd_Click(object sender, EventArgs e)
{
    Addressdet.Visible = true;
}
private void ContentSaveControlClicked()
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    UserId = Convert.ToInt32(Clsusr.UserId);
    try
    {
        if (Addressdet.Visible == true)
        {
            //Save Address first check data Existing
            if (ddl_addtype.SelectedIndex == 0)
            {
                Common.ShowMessageHtml("Address Type cannot be blank", 0, true,
dvMsgBox);
            }
            else if (txt_add1.Text == "")
            {
                Common.ShowMessageHtml("Address cannot be blank", 0, true, dvMsgBox);
            }
            else if (txt_city.Text == "")
            {

```



```

        Common.ShowMessageHtml("City cannot be blank", 0, true, dvMsgBox);
    }
    else if (txt_pincode.Text == "")
    {
        Common.ShowMessageHtml("Pincode cannot be blank", 0, true, dvMsgBox);
    }
    else if (txt_contactno.Text == "")
    {
        Common.ShowMessageHtml("Contact No cannot be blank", 0, true,
dvMsgBox);
    }
    else
    {
        Etetbluser_address et = new Etetbluser_address();
        et.address_type = ddl_addtype.SelectedIndex;
        et.address1 = txt_add1.Text;
        et.address3 = txt_add3.Text;
        et.address2 = txt_add2.Text;
        et.city = txt_city.Text;
        et.country = 1;
        et.countryname = "India";
        et.pincode = txt_pincode.Text;
        et.ContactNo = txt_contactno.Text;
        bool ret = new tblUsers().InsertAddress(et);
        if (ret)
        {
            Common.ShowMessageHtml("Address Updated", 0, true, dvMsgBox);
            List<Etetbluser_address> Lst = (new
tblUsers()).getusersaddressbyuserid(UserId, "");

            //if (Lst != null && Lst.Count > 0)
            //{
            //    WDGrid_UserAddress.DataSource = Lst;
            //    WDGrid_UserAddress.DataBind();
            //    WDGrid_UserAddress.Visible = true;
            //}
            //else
            //{
            //    WDGrid_UserAddress.Visible = false;
            //}

        }
    }
}
else if (Details.Text == "")
{
    Common.ShowMessageHtml("Company Description cannot be blank", 0, true,
dvMsgBox);
}
}

```

```

else
{
    Etetbluser_profile Prd = new Etetbluser_profile();
    Prd.userid = Convert.ToInt32(Request.QueryString["Id"]);

    bool Result = false;
    PageType = Session["PageType"].ToString();

    Prd.userid = Convert.ToInt32(Request.QueryString["ProductId"]);
    Prd.Company_Description = Details.Text;
    Prd.Business_Type = txt_BusinessType.Text;
    Prd.UOMId = ddl_uom.SelectedIndex;
    Prd.Legal_Structure = ddl_legalstr.SelectedValue;
    Prd.CID = ddl_currency.SelectedIndex;
    Prd.Year_Established = txt_yrest.Text;
    Prd.Number_of_Employees = txt_noempl.Text;
    Prd.Turnover_per_Year = txt_turnover.Text;
    Prd.CID = ddl_currency.SelectedIndex;
    Prd.Quality_Assurance_Accreditation = txt_qa.Text;
    Prd.Registration_Number = txt_regno.Text;
    Prd.D_U_N_S_Number = txt_dunsno.Text;
    Prd.Specializing_in = txt_special.Text;
    Prd.Monthly_Production_Capacity = txt_prodcap.Text;
    Prd.Company_Website_URL = txt_website.Text;
    Prd.UpdatedOn = DateTime.Now;
    Prd.CreatedOn = DateTime.Now;
    Prd.userid = Convert.ToInt16(Clsusr.UserId);
    //Prd.Company_logo = Photo;
    if (PageType == "Add")
    {
        Result = new tblUsers().InsertProfile(Prd);
    }
    else
    {
        Result = new tblUsers().UpdateProfile(Prd);
    }

    if (Result == true)
    {
        if (PageType == "Add")
        {
            dvMsgBox;
            Common.ShowMessageHtml("Adding of Profile Sucessful ", 1, true,
            dvMsgBox);
        }
        else
        {
            dvMsgBox;
            Common.ShowMessageHtml("Profile Edited Sucessfully ", 1, true,
            dvMsgBox);
        }
        Response.Redirect("~/Home.aspx?pageid=1&msg=Product Updated
        Sucessfully");
    }
}

```

```

    }
}

catch (Exception exc)
{
    objlog.Trace_Error(4, "DateOrWed", "Profile", "Load", "Exception:" + exc.ToString());
    Common.ShowMessageHtml(exc.ToString(), 0, true, dvMsgBox);
}

//objlog.Trace_Log(0, "DateOrWed", "MangeDealer", "SaveClick", "Start Here..");
}

protected void btn_SaveAddress_Click(object sender, System.EventArgs e)
{//save code
    Addressdet.Visible = true;
}

protected void Button2_Click(object sender, System.EventArgs e)
{
    Addressdet.Visible = true;
}

protected void btn_saveimage_Click(object sender, System.EventArgs e)
{
    string path = FileUpload1.FileName;
    byte[] Photo = GetPhotoStream();
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    UserId = Convert.ToInt16(Clsusr.UserId);
    EtetblUserImages ep = new EtetblUserImages();

    ep.UserId = UserId;

    ep.Images = Photo;
    bool ret = new tblUsers().InsertImages(ep);

    List<EtetblUserImages> Pics = (new tblUsers()).getuserImagesbyuserid(UserId, "");
    foreach (var pic in Pics)
    {
        switch (pic.srno)
        {
            case 1:
                Image2.ImageUrl = pic.ImageUrl;
                Image2.Visible = true;
                break;
            case 2:
                Image3.ImageUrl = pic.ImageUrl;
                Image3.Visible = true;
                break;
            case 3:
                Image4.ImageUrl = pic.ImageUrl;
                Image4.Visible = true;
                break;
            case 4:
                Image5.ImageUrl = pic.ImageUrl;

```

```

        Image5.Visible = true;
        break;
    default:
        break;
    }
}
//if (Pics != null && Pics.Count > 0)
//{
//    WebDataPictures.DataSource = Pics;
//    WebDataPictures.DataBind();
//    WebDataPictures.Visible = true;
//}
//else
//{
//    WebDataPictures.Visible = false;
//}
}
protected void upload_image()
{
    string path = FileUpload1.FileName;
    byte[] Photo = GetPhtoStream();
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    UserId = Convert.ToInt16(Clsusr.UserId);
    EtetblUserImages ep = new EtetblUserImages();

    ep.UserId = UserId;

    ep.Images = Photo;
    bool ret = new tblUsers().InsertImages(ep);

    List<EtetblUserImages> Pics = (new tblUsers()).getuserImagesbyuserid(UserId, "");
    foreach (var pic in Pics)
    {
        switch (pic.srno)
        {
            case 1:
                Image2.ImageUrl = pic.ImageUrl;
                Image2.Visible = true;
                break;
            case 2:
                Image3.ImageUrl = pic.ImageUrl;
                Image3.Visible = true;
                break;
            case 3:
                Image4.ImageUrl = pic.ImageUrl;
                Image4.Visible = true;
                break;
            case 4:
                Image5.ImageUrl = pic.ImageUrl;
                Image5.Visible = true;
                break;
            default:
                break;
        }
    }
}

```

```

    }
    //if (Pics != null && Pics.Count > 0)
    //{
    //    WebDataPictures.DataSource = Pics;
    //    WebDataPictures.DataBind();
    //    WebDataPictures.Visible = true;
    //}
    //else
    //{
    //    WebDataPictures.Visible = false;
    //}
}
private void download(DataTable dt)
{
    foreach (DataRow row in dt.Rows)
    {
        //    Byte[] bytes = (Byte[])dt.Rows[0]["Data"].ToString();
        Byte[] bytes = (Byte[])row["Images"];
        Response.Buffer = true;
        Response.Charset = "";
        Response.Cache.SetCacheability(HttpCacheability.NoCache);
        Response.ContentType = row["ContentType"].ToString();
        //    Response.ContentType = dt.Rows[0]["ContentType"].ToString();
        //    Response.AddHeader("content-disposition", "attachment;filename=" +
dt.Rows[0]["Name"]);
        //    var path = Server.MapPath(row["Name"]);
        string path = Server.MapPath("");
        path = path + @row["Name"].ToString();
        Response.AddHeader("content-disposition", "attachment;filename=" + path);

        Response.BinaryWrite(bytes);
        Response.TransmitFile(path);
        //Response.WriteFile(path);
        Response.Flush();
        Response.End();
    }
}

protected void btn_upload_Click(object sender, EventArgs e)
{
    string path = FileUpload1.FileName;
    byte[] Photo = GetPhtoStream();
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    UserId = Convert.ToInt16(Clsusr.UserId);
    EtetblUserImages ep = new EtetblUserImages();

    ep.UserId = UserId;

    ep.Images = Photo;
    bool ret = new tblUsers().InsertImages(ep);

    List<EtetblUserImages> Pics = (new tblUsers()).getuserImagesbyuserid(UserId, "");
}

```

```

foreach (var pic in Pics)
{
    switch (pic.srno)
    {
        case 1:
            Image2.ImageUrl = pic.ImageUrl;
            Image2.Visible = true;
            break;
        case 2:
            Image3.ImageUrl = pic.ImageUrl;
            Image3.Visible = true;
            break;
        case 3:
            Image4.ImageUrl = pic.ImageUrl;
            Image4.Visible = true;
            break;
        case 4:
            Image5.ImageUrl = pic.ImageUrl;
            Image5.Visible = true;
            break;
        default:
            break;
    }
}

protected void btn_addaddress_Click(object sender, EventArgs e)
{
    Addressdet.Visible =true;
}
}

```

***** UserList.aspx *****

```

<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="UserList.aspx.cs" Inherits="View_UserList" %>

```

```

<%@ Register Assembly="Infragistics4.Web.v14.2, Version=14.2.20142.2480,
Culture=neutral, PublicKeyToken=7dd5c3163f2cd0cb"
Namespace="Infragistics.Web.UI.GridControls" TagPrefix="ig" %>

```

```

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
</asp:Content>

```

```

<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
<div class="container-fluid">
<div class="row-fluid" style="padding-top: 20px;">
<div class="span2" id="dvlogo" runat="server" style="display: block;">
<%----%>
</div>
<div class="span10">

```

```

        <div class="row">
            <div id="dvMsgBox" class="span12 ordersave" runat="server" style="display:
none;">
                <asp:Label ID="Label1" runat="server" Text=""></asp:Label>
            </div>
        </div>
    </div>
</div>
</div>
</div>
</div>
</div>
</asp:Content>

<asp:Content ID="Content4" ContentPlaceHolderID="CPH_LeftSideBar" runat="Server">
    <div id="LeftSideBar" runat="server" class="span2">
        <asp:Table ID="Table1" runat="server">
            </asp:Table>
        </div>
    </asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="CPH_MainContent" runat="Server">

    <div class="span10" id="MainContent" runat="server">
        <div id="ShowPageHeader" class="row" style="display: block;" runat="server">
            <div id="PageName" class="span12 customer" runat="server">
                </div>
            </div>

            <div id="Div1" class="row" style="display: block;">
                <div id="Div2" class="span12 customertable">

                    <%--<asp:ScriptManager ID="ScriptManager1"
runat="server"></asp:ScriptManager>--%>

                    <ig:WebDataGrid ID="WDGrid_UserList" runat="server" CssClass="wdglist"
AutoGenerateColumns="False">
                        <Columns>

                            <ig:BoundDataField DataFieldName="UserName" Key="UserName" ">
                                <Header Text="User Name">
                                    </Header>
                            </ig:BoundDataField>
                            <ig:BoundDataField DataFieldName="FirstName" Key="FirstName">
                                <Header Text="First Name">
                                    </Header>
                            </ig:BoundDataField>
                            <ig:BoundDataField DataFieldName="EmailID" Key="EmailID">
                                <Header Text="Email Id">
                                    </Header>
                            </ig:BoundDataField>
                            <ig:BoundDataField DataFieldName="DealerName" Key="DealerName">
                                <Header Text="Dealer Name">
                                    </Header>
                            </ig:BoundDataField>
                            <ig:BoundDataField DataFieldName="ContactNum" Key="ContactNum">
                                <Header Text="Contact Number">
                                    </Header>
                        </Columns>
                    </ig:WebDataGrid>
                </div>
            </div>
        </div>
    </div>
</asp:Content>

```

```

        </ig:BoundDataField>
        <ig:BoundDataField DataFieldName="GroupName" Key="GroupName"
Width="50px">
            <Header Text="Group Name">
                </Header>
            </ig:BoundDataField>

            <ig:BoundDataField DataFieldName="CreatedOn" Key="CreatedOn"
DataFormatString="{0:dd/MM/yyyy}">
                <Header Text="Created On">
                    </Header>
                </ig:BoundDataField>
            <ig:BoundDataField DataFieldName="Status" Key="Status">
                <Header Text="Is Active">
                    </Header>
                </ig:BoundDataField>

            <ig:TemplateDataField Key="userid" Width="60px">
                <ItemTemplate>
                    <asp:HyperLink ID="userid" runat="server" CssClass="btn btn-small btn-
primary" Text="Open"
                        NavigateUrl='<% #Eval("UserName",
"~/ManageUser.aspx?pageid=163&PageType=Edit&username={0}")%>'>
                </asp:HyperLink>
                </ItemTemplate>
                <Header Text="Open" />
            </ig:TemplateDataField>

        </Columns>
        <Behaviors>
            <ig:Paging CurrentPageLinkCssClass="Current" PageLinkCssClass="Link"
PagerCssClass="Pager" PagerMode="NumericFirstLast" PageSize="15" QuickPages="3"
FirstPageText="First" LastPageText="Last" NextPageText="Next"
PreviousPageText="Previous" PagerAppearance="bottom">
                </ig:Paging>

            <ig:Filtering>
                </ig:Filtering>
            <ig:Sorting Enabled="true">
                </ig:Sorting>
        </Behaviors>
    </ig:WebDataGrid>
    <asp:SqlDataSource ID="SqlDataSource1" runat="server"
ConnectionString="<%%$ ConnectionStrings:DowConectingString %>"
SelectCommand="SELECT [UserId], [UserName], [FirstName], [LastName], [EmailID],
[ContactNum], [CreatedOn], [Status] FROM [tblUser]"></asp:SqlDataSource>
    </div>
</div>
</asp:Content>

```

*****UserList.aspx.cs *****

```

using System;
using System.Collections.Generic;
using System.Linq;

```



```

using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBAL;
using DowBEntity;
using System.Data;

public partial class View_UserList : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr != null)
        {

            string pageid = Request.QueryString["pageid"];
            ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
Convert.ToInt32(Clsusr.GroupId));
            string msg = Request.QueryString["msg"];
            if (msg != null)
            {
                Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
            }
            Bindetails();
        }
        else
        {
            Response.Redirect("MLB_Login.aspx");
        }
    }
}

private void ManageNavigations(int pageid, int userid, int groupid)
{
    bool IsLeftBar = false;
    IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
myBtnClickHandler);

    if (IsLeftBar == true)
    {
        LeftSideBar.Attributes["class"] = "span2";
        MainContent.Attributes["class"] = "span10";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;"); //style="display:
block;"
        PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
    else
    {
        LeftSideBar.Attributes["class"] = "";
        MainContent.Attributes["class"] = "span12";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
    }
}

```

```

        PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
}

// This is Event Handler
protected void myBtnClickHandler(object sender, EventArgs e)
{

    LinkButton lnk = (LinkButton)sender;
    string ID = lnk.ID;

    DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
    if (ds != null)
    {
        if (ds.Tables[0].Rows.Count > 0)
        {
            DataTable dtdll = ds.Tables[0];
            foreach (DataRow drnew in dtdll.Rows)
            {
                switch (drnew["lbtncode"].ToString())
                {
                    case "SAVE":
                        // Call SAVE Method
                        break;
                    case "URL":
                        Response.Redirect(drnew["pageurl"].ToString());
                        break;
                    case "AUDIT":
                        Response.Redirect(drnew["pageurl"].ToString() + "&frompage=" +
Request.QueryString["pageid"]);
                        break;
                    case "NEW":
                        Response.Redirect("~/ManageUser.aspx?pageid=163&pagetype=Add");
                        break;
                    case "CANCEL":
                        // Call Cancel Method
                        break;
                    default:
                        break;
                }
            }
        }
    }
}

private void Bindetails()
{

```

```

try
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    if (Clsusr != null)
    {
        List<EtetblUser> Lst;
        if (Convert.ToInt32(Clsusr.GroupId) == 1)
        {
            Lst = (new tblUsers()).getusers("ManageUser.aspx");
        }

        else
        {
            Lst = (new tblUsers()).getusersbyusers("ManageUser.aspx",
Convert.ToInt32(Clsusr.DealerId), Convert.ToInt32(Clsusr.GroupId));

        }
        if (Lst != null && Lst.Count > 0)
        {
            WDGrid_UserList.DataSource = Lst;
            WDGrid_UserList.DataBind();
            WDGrid_UserList.Visible = true;
        }
        else
        {
            WDGrid_UserList.Visible = false;
        }
    }
}

catch (Exception exc)
{
    throw exc;
}

}

```

```

}
***** VatCodeList.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="VatCodeList.aspx.cs" Inherits="VatCodeList" %>

```

```

<%@ Register Assembly="Infragistics4.WebUI.WebSpellChecker.v14.2,
Version=14.2.20142.1028, Culture=neutral, PublicKeyToken=7dd5c3163f2cd0cb"
Namespace="Infragistics.WebUI.WebDataInput" TagPrefix="igtxt" %>
<%@ Register Assembly="Infragistics4.Web.v14.2, Version=14.2.20142.1028,
Culture=neutral, PublicKeyToken=7dd5c3163f2cd0cb"
Namespace="Infragistics.Web.UI.GridControls" TagPrefix="ig" %>

```

```

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
</asp:Content>

<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
  <div class="container-fluid">
    <div class="row-fluid" style="padding-top: 20px;">
      <div class="span2" id="dvlogo" runat="server" style="display: block;">
        <!---->
      </div>
      <div class="span10">
        <div class="row">
          <div id="dvMsgBox" class="span12 ordersave" runat="server" style="display:
none;">
            <asp:Label ID="Label1" runat="server" Text=""></asp:Label>
          </div>
        </div>
      </div>
    </div>
  </div>
</asp:Content>

<asp:Content ID="Content4" ContentPlaceHolderID="CPH_LeftSideBar" runat="Server">
  <div id="LeftSideBar" runat="server" class="span2">
    <asp:Table ID="Table1" runat="server">
    </asp:Table>
  </div>
</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="CPH_MainContent" runat="Server">

  <div class="span10" id="MainContent" runat="server">
    <div id="ShowPageHeader" class="row" style="display: block;" runat="server">
      <div id="PageName" class="span12 customer" runat="server">
      </div>
    </div>

    <div id="Div1" class="row" style="display: block;">
      <div id="Div2" class="span12 customertable">

        <!-- Main Content Comes here-->

        <!--<asp:ScriptManager ID="ScriptManager1"
runat="server"></asp:ScriptManager>-->

        <ig:WebDataGrid ID="WDGrid_UserList" runat="server" CssClass="wdglist"
AutoGenerateColumns="False">
          <Columns>
            <ig:BoundDataField DataFieldName="Code" Key="Code" Width="100px">
              <Header Text=" Code">
              </Header>
            </ig:BoundDataField>
          </Columns>
        </ig:WebDataGrid>
      </div>
    </div>
  </div>

```

```

        <ig:BoundDataField DataFieldName="Description" Key="Description"
Width="250px">
            <Header Text=" Description">
                </Header>
            </ig:BoundDataField>
            <ig:BoundDataField DataFieldName="Rate" Key="Rate" Width="100px">
                <Header Text="Rate ">
                    </Header>
                </ig:BoundDataField>
            <ig:BoundDataField DataFieldName="IsDefault" Key="Id" Width="100px">
                <Header Text="Is Default">
                    </Header>
                </ig:BoundDataField>

        <ig:BoundDataField DataFieldName="CreatedOn" Key="CreatedOn"
Width="100px">
            <Header Text="Created On">
                </Header>
            </ig:BoundDataField>

        <ig:TemplateDataField Key="DealerId" Width="60px">
            <ItemTemplate>
                <asp:HyperLink ID="DealerId" runat="server" CssClass="btn btn-small
btn-primary" Text="Open"
                NavigateUrl='<% #Eval("Id",
"~/ManageVatCodes.aspx?pageid=187&PageType=Edit&Id={0}")%>'>
            </asp:HyperLink>
            </ItemTemplate>
            <Header Text="Open" />
        </ig:TemplateDataField>
    </Columns>
    <Behaviors>
        <ig:Paging CurrentPageLinkCssClass="Current" PageLinkCssClass="Link"
PagerCssClass="Pager" PagerMode="NumericFirstLast" PageSize="15" QuickPages="3"
FirstPageText="First" LastPageText="Last" NextPageText="Next"
PreviousPageText="Previous" PagerAppearance="bottom">
            </ig:Paging>

        <ig:Filtering>
            </ig:Filtering>
        <ig:Sorting Enabled="true">
            </ig:Sorting>
    </Behaviors>
</ig:WebDataGrid>
    <asp:SqlDataSource ID="SqlDataSource1" runat="server"
ConnectionString="<%= $ ConnectionStrings:DowConectingString %>"
SelectCommand="SELECT [UserId], [UserName], [FirstName], [LastName], [EmailID],
[ContactNum], [CreatedOn], [Status] FROM [tblUser]"></asp:SqlDataSource>

    </div>
</div>
</div>
</asp:Content>

```

***** VatCodeList.aspx.cs*****

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBEntity;
using DowBAL;
using System.Data;

public partial class VatCodeList : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        //if (!IsPostBack)
        //{
            ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
            if (Clsusr != null)
            {

                string pageid = Request.QueryString["pageid"];
                ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
Convert.ToInt32(Clsusr.GroupId));
                string msg = Request.QueryString["msg"];
                if (msg != null)
                {
                    Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
                }
                if (Clsusr.GroupId == "1")
                {
                    Bindetails();
                }
            }
            else
            {
                Response.Redirect("MLB_Login.aspx");
            }
        }

        private void ManageNavigations(int pageid, int userid, int groupid)
        {
            bool IsLeftBar = false;
            IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
myBtnClickHandler);

            if (IsLeftBar == true)
            {
                LeftSideBar.Attributes["class"] = "span2";
                MainContent.Attributes["class"] = "span10";
                ShowPageHeader.Attributes.CssStyle.Add("display", "block;"); //style="display:
block;"
                PageName.InnerText = new
tblPager().GettblPagerNameById(Convert.ToInt32(pageid));
            }
            else

```

```

    {
        LeftSideBar.Attributes["class"] = "";
        MainContent.Attributes["class"] = "span12";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
        PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
}

// This is Event Handler
protected void myBtnClickHandler(object sender, EventArgs e)
{
    LinkButton lnk = (LinkButton)sender;
    string ID = lnk.ID;

    DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
    if (ds != null)
    {
        if (ds.Tables[0].Rows.Count > 0)
        {
            DataTable dtdll = ds.Tables[0];
            foreach (DataRow drnew in dtdll.Rows)
            {
                switch (drnew["lbtncode"].ToString())
                {
                    case "SAVE":
                        // Call SAVE Method
                        break;
                    case "URL":
                        Response.Redirect(drnew["pageurl"].ToString());
                        break;
                    case "AUDIT":
                        Response.Redirect(drnew["pageurl"].ToString() + "&frompage=" +
Request.QueryString["pageid"]);
                        break;
                    case "CANCEL":
                        Response.Redirect("Home.aspx?pageid=1");
                        break;
                    case "NEW":
                        Response.Redirect("ManageVatCodes.aspx?pageid=4&pagetype=Add");
                        break;
                    default:
                        break;
                }
            }
        }
    }
}
}

```

```

private void Bindetails()
{
    //try
    //{
        List<EteVatCode> Lst = (new BalVatCodes()).GetVatCodes();

        if (Lst != null && Lst.Count > 0)
        {
            WDGrid_UserList.DataSource = Lst;
            WDGrid_UserList.DataBind();
            WDGrid_UserList.Visible = true;
        }
        else
        {
            WDGrid_UserList.Visible = false;
        }
    }
}
protected override void OnInit(EventArgs e)
{
    base.OnInit(e);

    if (this.Master is MainMaster)
    {
        ((MainMaster)this.Master).MasterControlClicked += new
        EventHandler(ContentControlClicked);
    }
}
private void ContentControlClicked(object sender, EventArgs e)
{
    string pageid = "pageid=" + Session["pageid"].ToString();

    Response.Redirect("~/ManageVatCodes.aspx?pageid=187&PageType=Add&dealname=0
    &" + 0);
}
}
*****VendorColorsAdd.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/ProductMaster.master"
AutoEventWireup="true" CodeFile="VendorColorsAdd.aspx.cs" Inherits="VendorColorsAdd"
%>
<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
Runat="Server">
</asp:Content>
<asp:Content ID="Content4" ContentPlaceHolderID="CPH_TopMessages" Runat="Server">
    <div class="col-lg-12 col-md-12 col-sm-12">
        <div class="row">
            <%--<h3>Ravikiran Techno Soft</h3>--%>
            <div id="dvMsgBox" class="span10 ordersave" runat="server" style="display:
block;">
                <%--
                    <asp:Label ID="Label2" runat="server" Text=""
Height="100px"></asp:Label>--%>
                    <asp:Label ID="Label3" runat="server" Text="" Height="60px"></asp:Label>
                </div>
            </div>
        </div>
    </div>

```



```

    </div>
    <div class="container-fluid">
        <div class="row-fluid" style="padding-top: 20px;">
            <div>
                <h5 class="my-4 dark-grey-text font-weight-bold" style="padding-left:20px">Manage
                Selected Colors
                    <asp:Button ID="btn_save" runat="server" CssClass="btn btn-secondary btn-md"
                    Text="Save" OnClick="btn_save_Click" /> <%-- btn-rounded--%>
                    <asp:Button ID="btn_clear" runat="server" CssClass ="btn btn-secondary btn-md btn
                    rounded" Text="Clear" OnClick="btn_clear_Click" />
                <%--
                    <a href="#" class="btn btn-secondary btn-md btn-rounded"> &nbsp;Edit</a> <%--<i
                    class="fas fa-edit"></i>
                    <a href="#" class="btn btn-secondary btn-md btn-rounded"> &nbsp;Delete</a><%--<i
                    class="fas fa-trash"></i>--%>
                </h5>
                <!-- Navbar links -->

                </div>
                <div class="span10">
                    <div class="row">
                        <div id="Div1" class="span12 ordersave" runat="server" style="display: none;">
                            <asp:Label ID="Label1" runat="server" Text=""></asp:Label>
                        </div>
                    </div>
                </div>
            </div>
        </div>

        <%-- <div id="LeftSideBar" runat="server" class="span2">
            <asp:Table ID="Table1" runat="server">
                </asp:Table>
            </div>--%>
        </div>
    </asp:Content>
    <asp:Content ID="Content2" ContentPlaceHolderID="CPH_MainContent" Runat="Server" >
        <div class="container-fluid" style="padding-left:100px; padding-right:50px">

            <!-- Navbar links -->
            <section class="mb-5">

                <!-- Card -->
                <div class="card card-cascade narrower">

                    </div>

                </section>

            <section class="mb-5">

                <!-- Card -->
                <div class="card card-cascade narrower">

                    <section>

                        <div class="card">

```

```

        <div id="prtable" class="card-body" runat="server">

            <asp:Table ID="Dtable" runat="server" border="1" CssClass="table table-striped
            table-detail col-lg-6 align-items-sm-center" >

</asp:Table>

<!-- Table -->
<!-- Table -->

<hr class="my-0"/>

    <!-- Bottom Table UI -->

</div>
</div>

</section>

    <!-- Card -->
</div>
</section>

</div>
</asp:Content>
<asp:Content ID="Content3" ContentPlaceHolderID="ContentPlaceHolder1"
Runat="Server">

</asp:Content>

*****VendorColorsAdd.aspx.cs *****
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBEntity;
using DowBAL;
using System.Xml;
using System.Xml.Serialization;
using System.IO;
using System.Text;
using System.Data.SqlClient;
using System.Configuration;
using System.Data;

```

```

using System.Web.Security;

public partial class VendorColorsAdd : System.Web.UI.Page
{
    string PageType;
    int Id;
    DDLogTracer objlog = new DDLogTracer();

    protected void Page_Load(object sender, EventArgs e)
    {
        Id = Convert.ToInt16(Session["Pgid"].ToString());
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr != null)
        {

            string msg = Request.QueryString["msg"];
            if (msg != null)
            {
                Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
            }

            if (Convert.ToInt16(Clsusr.GroupId) > 3)
            {
                Bindetails();
            }
            else
            {
                Response.Redirect("MLB_Login.aspx");
            }
        }
        private void Bindetails()
        {
            ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
            List<EteproductImages> Lst = new Products().GetProductExtColors(0); //user selected
            colors
            List<EteSellerColors> cnts = new
            Products().GetSellerColorOptions(Convert.ToInt16(Clsusr.UserId));
            CheckBox chk = new CheckBox();
            Dtable.Rows.Clear();
            //try
            // {
            if (Lst != null && Lst.Count > 0)
            {
                TableRow tr = new TableRow();
                TableCell td = new TableCell();
                TableHeaderRow thr = new TableHeaderRow();
                td = new TableCell();
                chk.ID = "CheckBoxAll";
                chk.Text = "Select All";
                td.Controls.Add(chk);
                thr.Cells.Add(td);
                td = new TableCell();
                td.Text = "Image";
            }
        }
    }
}

```

```

thr.Cells.Add(td);
td = new TableCell();
td = new TableCell();
td.Text = "Color Name";
thr.Cells.Add(td);
td = new TableCell();
td.Text = "Selected Date";
thr.Cells.Add(td);

Dtable.Rows.Add(thr);
tr = new TableRow();
td = new TableCell();

StringBuilder sb = new StringBuilder();

DateTime dts;

foreach (var dt in Lst)
{
    int pcount = (from a in cnts where a.ColorId == dt.ProductId select a).Count();
    if (pcount > 0)
    {
        dts = (from b in cnts where b.ColorId == dt.ProductId select
b.CreatedOn).FirstOrDefault();
    }
    else { dts = DateTime.Today.Date; }

    tr = new TableRow();
    td = new TableCell();
    chk = new CheckBox();
    chk.ID = "CheckBox" + dt.ProductId;
    chk.Checked = true;
    if (pcount > 0)
    {
        chk.Checked = true;
    }
    else
    {
        chk.Checked = false;
    }
    td.Controls.Add(chk);
    tr.Cells.Add(td);
    td = new TableCell();
    string img = "<img src=\"" + dt.ImagePath + "\" runat='server'
style='height:100px;width:100px' /> ";
    td.Text = img;
    tr.Cells.Add(td);
    td = new TableCell();
    td.Text = "<span class='badge badge-primary'" + dt.Descr + "</span>";
    tr.Cells.Add(td);
    td = new TableCell();
    td = new TableCell();
    td.Text = Convert.ToString(dts).Substring(0, 10);
    tr.Cells.Add(td);
    Dtable.Rows.Add(tr);

```

```
    }  
  }  
}
```

```
protected void btn_save_Click(object sender, EventArgs e)  
{  
    bool ret = false;  
    int grid = Convert.ToInt16(Session["Pgid"].ToString());  
    string user = "";  
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];  
    if (Clsusr == null)  
    {  
        HttpCookie myCookie = Request.Cookies["DateOrWedCookie"];  
        if (myCookie != null)  
        {  
            user = myCookie.Value.ToString();  
        }  
        else  
        {  
            user = "Guest";  
        }  
        guest_login(user);  
    }  
    //try  
    {  
        CheckBox cbox = new CheckBox();  
        cbox = (CheckBox)Dtable.FindControl("CheckBoxAll");  
        List<EteproductImages> Lst = new Products().GetProductExtColors(0); //user  
selected colors  
        if (cbox.Checked)  
        {  
            foreach (var list in Lst)  
            {  
                ret = update((int)list.ProductId, true);  
            }  
        }  
        else  
        {  
            foreach (var list in Lst)  
            {  
                cbox = (CheckBox)Dtable.FindControl("CheckBox" + list.ProductId);  
                if (cbox.Checked)  
                {  
                    ret = update((int)list.ProductId, true);  
                }  
                else  
                {  
                    ret = update((int)list.ProductId, false); }  
            }  
        }  
    }  
}
```

```

        Response.Redirect("~/VendorHome.aspx?pageid=1&msg=Product Updated
Sucessfully");
    }

    //catch (Exception exc)
    //{
    //    //objlog.Trace_Error(4, "DateOrWed", "MangeDealer", "Load", "Exception:" +
exc.ToString());
    //    Common.ShowMessageHtml(exc.ToString(), 0, true, dvMsgBox);
    //}
}
protected bool update(int prid,bool chk)
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedIn_Specs"];

    bool ret = new Products().UpdSellerColors(prid, Convert.ToInt16(Clsusr.UserId), chk);
    return ret;
}
protected void guest_login(string user)
{
    string City = "Hyderabad";// _City;
    string Country = "India";// _Country;

    BALCreateUser objlogin = new BALCreateUser();
    DataTable status_login = objlogin.LoginGuest(user, user,
Request.ServerVariables["REMOTE_ADDR"].ToString(), "", City, Country);
    string login_status = status_login.Rows[0]["UserStatus"].ToString();

    ClsLoggedInUser objLoggedInUser;
    if (login_status == "True")
    {
        objLoggedInUser = new ClsLoggedInUser();

        objLoggedInUser.UserId = status_login.Rows[0]["UserId"].ToString();
        objLoggedInUser.UserName = status_login.Rows[0]["UserName"].ToString();
        objLoggedInUser.GroupId = status_login.Rows[0]["GroupId"].ToString();
        objLoggedInUser.LoginStatus = "True";
        objLoggedInUser.LoginID = Convert.ToInt32(status_login.Rows[0]["LoginId"]);
        objLoggedInUser.DealerId = Convert.ToInt32(status_login.Rows[0]["DealerId"]);
        objLoggedInUser.RoleId = Convert.ToInt32(status_login.Rows[0]["RoleId"]);
        objLoggedInUser.BrandId = 1; //Convert.ToInt32(status_login.Rows[0]["BrandId"]);

        Session["LoggedIn_Specs"] = objLoggedInUser;

        FormsAuthentication.SetAuthCookie(objLoggedInUser.UserName, false);
        HttpCookie myCookie = new HttpCookie("DateOrWedCookie");
        DateTime now = DateTime.Now;
        myCookie.Value = objLoggedInUser.UserName;
        myCookie.Expires = now.AddYears(1);
        Response.Cookies.Add(myCookie);
    }
}

```

```

    }
}

protected void btn_retrieve_Click(object sender, EventArgs e)
{
    Bindetails();
}

protected void btn_clear_Click(object sender, EventArgs e)
{
}

}

}
}
}

*****VendorColorsAddUser.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/ProductMaster.master"
AutoEventWireup="true" CodeFile="VendorColorsAddUser.aspx.cs"
Inherits="VendorColorsAddUser" %>
<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
Runat="Server">
</asp:Content>
<asp:Content ID="Content4" ContentPlaceHolderID="CPH_TopMessages" Runat="Server">
    <div class="col-lg-12 col-md-12 col-sm-12">
        <div class="row">
            <%--<h3>Ravikiran Techno Soft</h3>--%>
            <div id="dvMsgBox" class="span10 ordersave" runat="server" style="display:
block;">
                <%--
                    <asp:Label ID="Label2" runat="server" Text=""
Height="100px"></asp:Label>--%>
                    <asp:Label ID="Label3" runat="server" Text="" Height="60px"></asp:Label>
                </div>
            </div>
        </div>
        <div class="container-fluid">
            <div class="row-fluid" style="padding-top: 20px;">
                <div>
                    <h5 class="my-4 dark-grey-text font-weight-bold" style="padding-left:20px">Manage
Selected Colors
                    <asp:Button ID="btn_save" runat="server" CssClass="btn btn-secondary btn-md"
Text="Save" OnClick="btn_save_Click" /> <%-- btn-rounded--%>
                    <asp:Button ID="btn_clear" runat="server" CssClass ="btn btn-secondary btn-md btn
rounded" Text="Clear" OnClick="btn_clear_Click" />
                    <%--
                        <a href="#" class="btn btn-secondary btn-md btn-rounded"> &nbsp;Edit</a> <%--
<i class="fas fa-edit"></i>
                        <a href="#" class="btn btn-secondary btn-md btn-rounded"> &nbsp;Delete</a><%--<i
class="fas fa-trash"></i>--%>
                    </h5>
                    <!-- Navbar links -->

                    </div>
                <div class="span10">
                    <div class="row">
                        <div id="Div1" class="span12 ordersave" runat="server" style="display: none;">

```



```
</div>
```

```
</div>
```

```
</section>
```

```
<!-- Card -->
```

```
</div>
```

```
</section>
```

```
</div>
```

```
</asp:Content>
```

```
<asp:Content ID="Content3" ContentPlaceHolderID="ContentPlaceHolder1"  
Runat="Server">
```

```
</asp:Content>
```

```
*****VendorColorsAddUser.aspx.cs *****
```

```
using System;  
using System.Collections.Generic;  
using System.Linq;  
using System.Web;  
using System.Web.UI;  
using System.Web.UI.WebControls;  
using DowBEntity;  
using DowBAL;  
using System.Xml;  
using System.Xml.Serialization;  
using System.IO;  
using System.Text;  
using System.Data.SqlClient;  
using System.Configuration;  
using System.Data;  
using System.Web.Security;
```

```
public partial class VendorColorsAddUser : System.Web.UI.Page
```

```
{
```

```
    string PageType;
```

```
    int Id;
```

```
    DDLogTracer objlog = new DDLogTracer();
```

```
protected void Page_Load(object sender, EventArgs e)
```

```
{
```

```
    Id = Convert.ToInt16(Session["Pgid"].ToString());
```

```
    CIsLoggedInUser CIsusr = (CIsLoggedInUser)Session["LoggedInUser_Specs"];
```

```
    if (CIsusr != null)
```

```
    {
```

```
        string msg = Request.QueryString["msg"];
```

```
        if (msg != null)
```

```
        {
```

```
            Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
```

```
        }
```

```

        if (Convert.ToInt16(Clsusr.GroupId) > 3)
        {
            Bindetails();
        }
    }
    else
    {
        Response.Redirect("MLB_Login.aspx");
    }
}
private void Bindetails()
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    List<EteproductImages> Lst = new Products().GetProductExtColors(0); //user selected
colors
    List<EteSellerColors> cnts = new
Products().GetSellerColorOptions(Convert.ToInt16(Session["VendorCode"]));
    CheckBox chk = new CheckBox();
    Dtable.Rows.Clear();
    //try
    // {
    if (Lst != null && Lst.Count > 0)
    {
        TableRow tr = new TableRow();
        TableCell td = new TableCell();
        TableHeaderRow thr = new TableHeaderRow();
        td = new TableCell();
        chk.ID = "CheckBoxAll";
        chk.Text = "Select All";
        td.Controls.Add(chk);
        thr.Cells.Add(td);
        td = new TableCell();
        td.Text = "Image";
        thr.Cells.Add(td);
        td = new TableCell();
        td = new TableCell();
        td.Text = "Color Name";
        thr.Cells.Add(td);
        td = new TableCell();
        td.Text = "Selected Date";
        thr.Cells.Add(td);

        Dtable.Rows.Add(thr);
        tr = new TableRow();
        td = new TableCell();

        StringBuilder sb = new StringBuilder();

        DateTime dts;

        foreach (var dt in Lst)
        {
            int pcount = (from a in cnts where a.ColorId == dt.ProductId select a).Count();
            if (pcount > 0)

```

```

        {
            dts = (from b in cnts where b.ColorId == dt.ProductId select
b.CreatedOn).FirstOrDefault();
        }
        else { dts = DateTime.Today.Date; }

        tr = new TableRow();
        td = new TableCell();
        chk = new CheckBox();
        chk.ID = "CheckBox" + dt.ProductId;
        chk.Checked = true;
        if (pcount > 0)
        {
            chk.Checked = true;
        }
        else
        {
            chk.Checked = false;
        }
        td.Controls.Add(chk);
        tr.Cells.Add(td);
        td = new TableCell();
        string img = "<img src=\"" + dt.ImagePath + "\" runat='server'
style='height:100px;width:100px' /> ";
        td.Text = img;
        tr.Cells.Add(td);
        td = new TableCell();
        td.Text = "<span class='badge badge-primary'" + dt.Descr + "</span>";
        tr.Cells.Add(td);
        td = new TableCell();
        td = new TableCell();
        td.Text = Convert.ToString(dts).Substring(0, 10);
        tr.Cells.Add(td);
        Dtable.Rows.Add(tr);
    }
}
}

```

```

protected void btn_save_Click(object sender, EventArgs e)
{
    bool ret = false;
    int grid = Convert.ToInt16(Session["Pgid"].ToString());
    string user = "";
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    if (Clsusr == null)
    {
        HttpCookie myCookie = Request.Cookies["DateOrWedCookie"];
        if (myCookie != null)
        {
            user = myCookie.Value.ToString();
        }
        else

```

```

        {
            user = "Guest";
        }
        guest_login(user);

    }
    //try
    {
        CheckBox cbox = new CheckBox();
        cbox = (CheckBox)Dtable.FindControl("CheckBoxAll");
        List<EteproductImages> Lst = new Products().GetProductExtColors(0); //user
selected colors
        if (cbox.Checked)
        {
            foreach (var list in Lst)
            {
                ret = update((int)list.ProductId, true);
            }
        }
        else
        {
            foreach (var list in Lst)
            {
                cbox = (CheckBox)Dtable.FindControl("CheckBox" + list.ProductId);
                if (cbox.Checked)
                {
                    ret = update((int)list.ProductId, true);
                }
                else
                {
                    ret = update((int)list.ProductId, false); }
            }
        }
        Response.Redirect("~/VendorHome.aspx?pageid=1&msg=Product Updated
Sucessfully");
    }

    //catch (Exception exc)
    //{
    //    //objlog.Trace_Error(4, "DateOrWed", "MangeDealer", "Load", "Exception:" +
exc.ToString());
    //    Common.ShowMessageHtml(exc.ToString(), 0, true, dvMsgBox);

    //}

}
protected bool update(int prid,bool chk)
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];

    bool ret = new Products().UpdSellerColors(prid, Convert.ToInt16(Clsusr.UserId), chk);
    return ret;
}
protected void guest_login(string user)

```



```

<div class="span10">
  <div class="row">
    <%--<h3>Ravikiran Techno Soft</h3>--%>
    <div id="dvMsgBox" class="span10 ordersave" runat="server" style="display:
block;">
      <asp:Label ID="Label2" runat="server" Text="" Height="10px"></asp:Label>
    </div>
  </div>
</div></asp:Content>
<asp:Content ID="Content4" ContentPlaceHolderID="CPH_TopMessages" Runat="Server">

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentMain" Runat="Server">
<!-- Main layout -->

<div class="container-fluid">

  <!-- Section: Intro -->
  <section class="mt-md-4 pt-md-2 mb-5 pb-4">

    <!-- Grid row -->
    <div class="row">

      <!-- Grid column -->
      <div class="col-xl-3 col-md-6 mb-xl-0 mb-4">

        <!-- Card -->
        <div class="card card-cascade cascading-admin-card">

          <!-- Card Data -->
          <div class="admin-up">
            <i class="far fa-money-bill-alt primary-color mr-3 z-depth-2"></i>
            <div class="data">
              <p class="text-uppercase">sales</p>
              <h4 class="font-weight-bold dark-grey-text">2000$</h4>
            </div>
          </div>

          <!-- Card content -->
          <div class="card-body card-body-cascade">
            <div class="progress mb-3">
              <div class="progress-bar bg-primary" role="progressbar" style="width: 25%" aria-
valuenow="25" aria-valuemin="0" aria-valuemax="100"></div>
            </div>
            <p class="card-text">Better than last week (25%)</p>
          </div>

        </div>
      </div>
    </div>
  </div>
</div>
<!-- Grid column -->

<!-- Grid column -->

```

```

<div class="col-xl-3 col-md-6 mb-xl-0 mb-4">

  <!-- Card -->
  <div class="card card-cascade cascading-admin-card">

    <!-- Card Data -->
    <div class="admin-up">
      <i class="fas fa-chart-line warning-color mr-3 z-depth-2"></i>
      <div class="data">
        <p class="text-uppercase">subscriptions</p>
        <h4 class="font-weight-bold dark-grey-text">200</h4>
      </div>
    </div>

    <!-- Card content -->
    <div class="card-body card-body-cascade">
      <div class="progress mb-3">
        <div class="progress-bar red accent-2" role="progressbar" style="width: 25%"
aria-valuenow="25" aria-valuemin="0" aria-valuemax="100"></div>
      </div>
      <p class="card-text">Worse than last week (25%)</p>
    </div>

  </div>
  <!-- Card -->

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-xl-3 col-md-6 mb-md-0 mb-4">

  <!-- Card -->
  <div class="card card-cascade cascading-admin-card">

    <!-- Card Data -->
    <div class="admin-up">
      <i class="fas fa-chart-pie light-blue lighten-1 mr-3 z-depth-2"></i>
      <div class="data">
        <p class="text-uppercase">traffic</p>
        <h4 class="font-weight-bold dark-grey-text">20000</h4>
      </div>
    </div>

    <!-- Card content -->
    <div class="card-body card-body-cascade">
      <div class="progress mb-3">
        <div class="progress-bar red accent-2" role="progressbar" style="width: 75%"
aria-valuenow="75" aria-valuemin="0" aria-valuemax="100"></div>
      </div>
      <p class="card-text">Worse than last week (75%)</p>
    </div>

  </div>
  <!-- Card -->

```

```

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-xl-3 col-md-6 mb-0">

  <!-- Card -->
  <div class="card card-cascade cascading-admin-card">

    <!-- Card Data -->
    <div class="admin-up">
      <i class="fas fa-chart-bar red accent-2 mr-3 z-depth-2"></i>
      <div class="data">
        <p class="text-uppercase">organic traffic</p>
        <h4 class="font-weight-bold dark-grey-text">2000</h4>
      </div>
    </div>

    <!-- Card content -->
    <div class="card-body card-body-cascade">
      <div class="progress mb-3">
        <div class="progress-bar bg-primary" role="progressbar" style="width: 25%" aria-
valuenow="25" aria-valuemin="0" aria-valuemax="100"></div>
      </div>
      <p class="card-text">Better than last week (25%)</p>
    </div>

  </div>
<!-- Card -->

</div>
<!-- Grid column -->

</div>
<!-- Grid row -->

</section>
<!-- Section: Intro -->

<!-- Section: Main panel -->
<section class="mb-5">

  <!-- Card -->
  <div class="card card-cascade narrower">

    <!-- Section: Chart -->
    <section>

      <!-- Grid row -->
      <div class="row">

        <!-- Grid column -->
        <div class="col-xl-5 col-lg-12 mr-0 pb-2">

```



```

<!-- Card image -->
<div class="view view-cascade gradient-card-header light-blue lighten-1">
  <h2 class="h2-responsive mb-0 font-weight-500">Purchased</h2>
</div>

<!-- Card content -->
<div class="card-body card-body-cascade pb-0">

  <!-- Panel data -->
  <div class="row py-3 pl-4">

    <!-- First column -->
    <div class="col-md-6">

      <!-- Date select -->
      <p class="lead"><span class="badge info-color p-2">Data range</span></p>
      <select class="mdb-select colorful-select dropdown-info md-form">
        <option value="" disabled selected>Choose time period</option>
        <option value="1">Today</option>
        <option value="2">Yesterday</option>
        <option value="3">Last 7 days</option>
        <option value="3">Last 30 days</option>
        <option value="3">Last week</option>
        <option value="3">Last month</option>
      </select>

      <!-- Date pickers -->
      <p class="lead pt-3 pb-4"><span class="badge info-color p-2">Custom
date</span></p>
      <div class="md-form">
        <input placeholder="Selected date" type="text" id="from" class="form-control
datepicker">
        <label for="date-picker-example">From</label>
      </div>
      <div class="md-form">
        <input placeholder="Selected date" type="text" id="to" class="form-control
datepicker">
        <label for="date-picker-example">To</label>
      </div>
    </div>

    <!-- First column -->

    <!-- Second column -->
    <div class="col-md-6 text-center pl-xl-2 my-md-0 my-3">

      <!-- Summary -->
      <p>Total sales: <strong>2000$</strong>
      <button type="button" class="btn btn-info btn-sm p-2" data-toggle="tooltip"
data-placement="top"
        title="Total sales in the given period"><i class="fas fa-
question"></i></button>
      </p>
      <p>Average sales: <strong>100$</strong>

```

```

        <button type="button" class="btn btn-info btn-sm p-2 mr-0" data-
toggle="tooltip" data-placement="top"
        title="Average daily sales in the given period"><i class="fas fa-
question"></i></button>
    </p>

    <!-- Change chart -->
    <span class="min-chart my-4" id="chart-sales" data-percent="76"><span
class="percent"></span></span>
    <h5>
        <span class="badge red accent-2 p-2">Change <i class="fas fa-arrow-circle-
up ml-1"></i></span>
        <button type="button" class="btn btn-info btn-sm p-2" data-toggle="tooltip"
data-placement="top"
        title="Percentage change compared to the same period in the past"><i
class="fas fa-question"></i>
        </button>
    </h5>

</div>
<!-- Second column -->

</div>
<!-- Panel data -->

</div>
<!-- Card content -->

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class="col-xl-7 col-lg-12 mb-4 pb-2">

    <!-- Chart -->
    <div class="view view-cascade gradient-card-header blue-gradient">

        <canvas id="lineChart" height="175"></canvas>

    </div>

</div>
<!-- Grid column -->

</div>
<!-- Grid row -->

</section>
<!-- Section: Chart -->

<!-- Section: Table -->
<section>

    <!-- Top Table UI -->
    <div class="table-ui p-2 mb-3 mx-4 mb-5">

```

```

<!-- Grid row -->
<div class="row">

  <!-- Grid column -->
  <div class="col-xl-3 col-md-6">

    <!-- Name -->
    <select class="mdb-select colorful-select dropdown-info mx-2 md-form mt-3 md-
dropdown">
      <option value="" disabled selected>Bulk actions</option>
      <option value="1">Delate</option>
      <option value="2">Export</option>
      <option value="3">Change segment</option>
    </select>

  </div>
  <!-- Grid column -->

  <!-- Grid column -->
  <div class="col-xl-3 col-md-6">

    <!-- Blue select -->
    <select class="mdb-select colorful-select dropdown-info mx-2 md-form mt-3 md-
dropdown">
      <option value="" disabled selected>Show only</option>
      <option value="1">All <span> (2000)</span></option>
      <option value="2">Never opened <span> (200)</span></option>
      <option value="3">Opened but unanswered <span> (1800)</span></option>
      <option value="4">Answered <span> (200)</span></option>
      <option value="5">Unsubscribed <span> (50)</span></option>
    </select>

  </div>
  <!-- Grid column -->

  <!-- Grid column -->
  <div class="col-xl-3 col-md-6">

    <!-- Blue select -->
    <select class="mdb-select colorful-select dropdown-info mx-2 md-form mt-3 md-
dropdown">
      <option value="" disabled selected>Filter segments</option>
      <option value="1">Contacts in no segments <span> (100)</span></option>
      <option value="1">Segment 1 <span> (2000)</span></option>
      <option value="2">Segment 2 <span> (1000)</span></option>
      <option value="3">Segment 3 <span> (4000)</span></option>
    </select>

  </div>
  <!-- Grid column -->

  <!-- Grid column -->
  <div class="col-xl-3 col-md-6">

```

```

        <div class="form-inline mt-2 ml-2">
            <div class="form-group md-form mt-2">
                <input class="form-control w-100" type="text" placeholder="Search">
            </div>
            <button class="btn btn-sm btn-primary ml-2 mr-0 mb-md-0 mb-4 px-2"><i
class="fas fa-search"></i></button>
        </div>

    </div>
    <!-- Grid column -->

</div>
<!-- Grid row -->

</div>
<!-- Top Table UI -->

</section>
<!--Section: Table-->

</div>
<!-- Card -->

</section>
<!-- Section: Main panel -->

</div>

<!-- Main layout --></asp:Content>
<asp:Content ID="Content3" ContentPlaceHolderID="ContentPlaceHolderfooter"
Runat="Server">
</asp:Content>

```

```

***** VendorHome.aspx.cs*****
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBEntity;
using DowBAL;

public partial class VendorHome : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr != null)
        {

            string msg = Request.QueryString["msg"];
            if (msg != null)
            {
                Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
            }
        }
    }
}

```

```

    }
}
else
{
    Response.Redirect("MLB_Login.aspx");
}
}
}
}
}
}
*****VendorProducts.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/VendorMasterPage.master"
AutoEventWireup="true" CodeFile="VendorProducts.aspx.cs" Inherits="VendorProducts" %>
<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
Runat="Server">
</asp:Content>
<asp:Content ID="Content4" ContentPlaceHolderID="CPH_TopMessages" Runat="Server">
    <div class="span10">
        <div class="row">
            <%--<h3>Ravikiran Techno Soft</h3>--%>
            <div id="Div1" class="span10 ordersave" runat="server" style="display: block;">
<%--
                <asp:Label ID="Label2" runat="server" Text=""
Height="100px"></asp:Label>--%>
                <asp:Label ID="Label3" runat="server" Text="" Height="60px"></asp:Label>
            </div>
        </div>
    </div>
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentMain" Runat="Server" >
    <div class="container-fluid" style="padding-left:100px; padding-right:50px">

        <div><%--VendorProductsAdd.aspx--%>
            <h5 class="my-3 dark-grey-text font-weight-bold">Products
                <a href="PostProducts.aspx" class="btn btn-outline-danger waves-effect btn-sm font-
weight-bold"><i class="far fa-plus-square"></i> &nbsp;Add New</a>
            <%--
                <a href="#" class="btn btn-outline-danger waves-effect btn-sm font-weight-
bold"><i class="fas fa-edit"></i> &nbsp;Edit</a> --%>
                <a href="#" class="btn btn-outline-danger waves-effect btn-sm font-weight-bold"><i
class="fas fa-edit"></i> &nbsp;Edit</a>
                <a href="#" class="btn btn-outline-danger waves-effect btn-sm font-weight-bold"><i
class="fas fa-trash-alt"></i> &nbsp;Delete</a>
            </h5>
        </div>
        <!-- Navbar links -->
        <section class="mb-5">

            <!-- Card -->
            <div class="card card-cascade narrower">

                <section>

                    <div class="card">
                        <div id="pstable" class=" col-md-12 col-sm-auto card-body" runat="server"> <%--card-
body--%>

```

```

<!-- Table -->
<table class="table table-striped" runat="server">

  <!-- Table head -->

  <thead class='mdb-color darken-3'>
  <tr class='text-white'>
  <th scope='row'>
  <input class='form-check-input' type='checkbox' id='checkboxall'>
  <label for='checkboxall' class='label-table'></label>
  </th>
  <th><i class='far fa-image'></i></th>
  <th style='width:20%'>Name</th>
  <th>Stock</th>
  <th>Price</th>
  <th style='width:20%'>Categories</th>
  <th>Date</th>
  <th style='width:10%'>Action</th>
  </tr>
  </thead>
  <!-- Table head -->

  <!-- Table body -->
  <tbody>
  <tr>
  <th scope="row">
  <input class="form-check-input" type="checkbox" id="checkbox2">
  <label for="checkbox2" class="label-table"></label>
  </th>
  <td></td>
  <td><a href="#">Best Bridal Dresses</a></td>
  <td class="text-success font-weight-bold">In stock</td>
  <td><s>₹8099</s><br>₹7999</td>
  <td><span class="badge badge-primary">Women Fashion</span> <span
class="badge badge-primary">Bridal</span></td>
  <td>Published<br>2018/11/16</td>
  </tr>
  </tbody>
  <!-- Table body -->
</table>
<!-- Table -->

<hr class="my-0">

  <!-- Bottom Table UI -->
  <div class="d-flex justify-content-between">

    <!-- Name -->
    <select class="mdb-select colorful-select dropdown-primary md-form hidden-md-
down">
      <option value="" disabled>Rows number</option>
      <option value="1" selected>6 rows</option>
      <option value="2">25 rows</option>
      <option value="3">50 rows</option>
    </select>
  </div>

```

```

    <option value="4">100 rows</option>
</select>

<!-- Pagination -->
<nav class="my-4">
  <ul class="pagination pagination-circle pg-blue mb-0">

    <!--First-->
    <li class="page-item disabled clearfix d-none d-md-block"><a class="page-
link">First</a></li>

    <!-- Arrow left -->
    <li class="page-item disabled">
      <a class="page-link" aria-label="Previous">
        <span aria-hidden="true">&laquo;</span>
        <span class="sr-only">Previous</span>
      </a>
    </li>

    <!-- Numbers -->
    <li class="page-item active"><a class="page-link">1</a></li>
    <li class="page-item"><a class="page-link">2</a></li>
    <li class="page-item"><a class="page-link">3</a></li>
    <li class="page-item"><a class="page-link">4</a></li>
    <li class="page-item"><a class="page-link">5</a></li>

    <!-- Arrow right -->
    <li class="page-item">
      <a class="page-link" aria-label="Next">
        <span aria-hidden="true">&raquo;</span>
        <span class="sr-only">Next</span>
      </a>
    </li>

    <!-- First -->
    <li class="page-item clearfix d-none d-md-block"><a class="page-
link">Last</a></li>

  </ul>
</nav>
<!-- Pagination -->

</div>
<!-- Bottom Table UI -->

</div>
</div>

</section>

</div>
<!-- Card -->

</section>

```

```
</div>
</asp:Content>
<asp:Content ID="Content3" ContentPlaceHolderID="ContentPlaceHolderfooter"
Runat="Server">
```

```
</asp:Content>
```

```
***** VendorProducts.aspx.cs *****
```

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBEntity;
using DowBAL;
using System.Text;
```

```
public partial class VendorProducts : System.Web.UI.Page
```

```
{
```

```
    DDLogTracer objlog = new DDLogTracer();
```

```
    protected void Page_Load(object sender, EventArgs e)
```

```
    {
```

```
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
```

```
        if (Clsusr != null)
```

```
        {
```

```
            string msg = Request.QueryString["msg"];
```

```
            if (msg != null)
```

```
            {
```

```
                //Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
```

```
            }
```

```
            if (!IsPostBack)
```

```
            {
```

```
                Bindetails(Convert.ToInt16(Clsusr.UserId));
```

```
            }
```

```
        }
```

```
    else
```

```
    {
```

```
        Response.Redirect("MLB_Login.aspx");
```

```
    }
```

```
}
```

```
private void Bindetails(int userid)
```

```
{
```

```
    //try
```

```
    //{
```

```
    StringBuilder sb = new StringBuilder();
```

```
    sb.Append("<table class='table' runat='server'>\n");
```

```
    sb.Append("    <thead class='mdb-color darken-3'> \n");
```

```
    sb.Append("    <tr class='text-white'>    \n");
```

```
    sb.Append("    <th scope='row'>    \n");
```

```
    sb.Append("    <input class='form-check-input' type='checkbox' id='checkboxall'> \n");
```

```
    sb.Append("    <label for='checkboxall' class='label-table'></label>\n");
```



```

sb.Append("        </th> \n");
sb.Append("        <th><i class='far fa-image'></i></th> \n");
sb.Append("        <th style='width:20%'>Name</th> \n");
sb.Append("        <th>Stock</th> \n");
sb.Append("        <th>Price</th> \n");
sb.Append("        <th style='width:20%'>Categories</th> \n");
sb.Append("        <th>Date</th> \n");
sb.Append("        <th style='width:10%'>Action</th> \n");
sb.Append("    </tr> \n");
sb.Append(" </thead> \n");

```

```

List<ProductProductSeller> Lst = (new
Products()).GetProductSellerbySellerId(userid,1);

```

```

if (Lst != null && Lst.Count > 0)
{
    sb.Append(" <tbody> \n");
    foreach (var dt in Lst)
    {
        sb.Append("        <tr> \n");
        sb.Append("        <th scope='row'> \n");
        sb.Append("        <input class='form-check-input' type='checkbox'
id='checkbox"+dt.PSId+"'> \n");
        sb.Append("        <label for='checkbox" + dt.PSId + "' class='label-
table'></label>\n");
        sb.Append("        </th> \n");
        sb.Append("        <td><img src='"+dt.ImagePath+ "' class='img-fluid'
style='height:100px;width:100px'></td> \n");
        sb.Append("        <td><a href='#'>"+dt.SellerDescription+"</a></td> \n");
        if (dt.TStock > 0) {
            sb.Append("        <td class='text-success font-weight-bold'>In stock(" +
dt.TStock + ")</td> \n");
        }
        else
        {
            sb.Append("        <td class='text-danger font-weight-bold'>Out Of stock</td>
\n");
        }
        //<td class="text-danger font-weight-bold">Out Of stock</td>
        sb.Append("        <td><s>₹" + dt.Price + "</s><br>₹" + dt.Price + "</td>
\n");
        sb.Append("        <td><span class='badge badge-primary'>" + dt.CatName +
"</span> <span class='badge badge-primary'>" + dt.GroupName + "</span></span> <span
class='badge badge-primary'>" + dt.ProductName + "</span></td>\n");
        sb.Append("        <td>Published<br>"+Convert.ToString(
dt.CreatedOn).Substring(0,10)+"</td> \n");
        sb.Append("        <td><a href='/postproducts.aspx?psid="+dt.PSId
+"&pagetype=Edit' class='btn btn-outline-danger waves-effect btn-sm font-weight-bold'><i
class='fas fa-edit'></i> &nbsp;Edit</a></td>");
        sb.Append("        </tr> \n");
    }
    sb.Append(" </tbody> \n");
}

```

```

    }
    prtble.InnerHtml = sb.ToString();
    objlog.Trace_Log(1, "home", "1", "menu", "\n" + sb.ToString());
    // WDGrid_UserList.DataSource = Lst;
    // WDGrid_UserList.DataBind();
    // WDGrid_UserList.Visible = true;
    //}
    //else
    //{
    // WDGrid_UserList.Visible = false;
    //}
}

}
*****VendorProductsAdd.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/ProductMaster.master"
AutoEventWireup="true" CodeFile="VendorProductsAdd.aspx.cs"
Inherits="VendorProductsAdd" %>
<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
Runat="Server">
</asp:Content>
<asp:Content ID="Content4" ContentPlaceHolderID="CPH_TopMessages" Runat="Server">
    <div class="col-lg-12 col-md-12 col-sm-12">
        <div class="row">
            <%--<h3>Ravikiran Techno Soft</h3>--%>
            <div id="dvMsgBox" class="span10 ordersave" runat="server" style="display:
block;">
<%--                <asp:Label ID="Label2" runat="server" Text=""
Height="100px"></asp:Label>--%>
                <asp:Label ID="Label3" runat="server" Text="" Height="60px"></asp:Label>
            </div>
        </div>
    </div>
    <div class="container-fluid">
        <div class="row-fluid" style="padding-top: 20px;">
            <div>
                <h5 class="my-4 dark-grey-text font-weight-bold" style="padding-left:20px">Manage
Selected Products
                <asp:Button ID="btn_save" runat="server" CssClass="btn btn-secondary btn-md"
Text="Save" OnClick="btn_save_Click" /> <%-- btn-rounded--%>
                <asp:Button ID="btn_clear" runat="server" CssClass="btn btn-secondary btn-md btn
rounded" Text="Clear" OnClick="btn_clear_Click" />
                <%--                <a href="#" class="btn btn-secondary btn-md btn-rounded"> &nbsp;Edit</a> <%--
<i class="fas fa-edit"></i>
                <a href="#" class="btn btn-secondary btn-md btn-rounded"> &nbsp;Delete</a><%--<i
class="fas fa-trash"></i>--%>
            </div>
        </div>
    </div>
    <div class="span10">
        <div class="row">
            <div id="Div1" class="span12 ordersave" runat="server" style="display: none;">
                <asp:Label ID="Label1" runat="server" Text=""></asp:Label>
            </div>

```

```

        </div>
    </div>
</div>

<%-- <div id="LeftSideBar" runat="server" class="span2">
    <asp:Table ID="Table1" runat="server">
        </asp:Table>
    </div--%>
</div>
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="CPH_MainContent" Runat="Server" >
    <div class="container-fluid" style="padding-left:100px; padding-right:50px">

        <!-- Navbar links -->
        <section class="mb-5">

            <!-- Card -->
            <div class="card card-cascade narrower">

                <div class=" col-lg-8 col-md-8 col-sm-6 category-num">
                    <table id="Table5" class="table table-striped table-detail col-lg-6"
runat="server">

                        <tbody>

                            <tr>
                                <td class="span3" dir="auto" style="width:170px">Product
Category : </td>

                                <td class="span3">
                                    <asp:DropDownList ID="ddl_productcat" runat="server"
OnSelectedIndexChanged="ddl_productcat_SelectedIndexChanged" Visible="true"
Width="200px">
                                        <asp:ListItem Value="0">Bridal</asp:ListItem>
                                        <asp:ListItem Value="1" >Womens</asp:ListItem>
                                    </asp:DropDownList>
                                    </td><td>
                                        <asp:Button ID="retgroup" runat="server" CssClass="btn btn-
secondary btn-md " OnClick="ddl_productcat_SelectedIndexChanged" Text="Retrieve Sub"
                                        />
                                    </td>
                                </tr><tr>
                                <td class="span3" dir="auto" style="width:200px">Product Sub-
Category : </td>

                                <td class="span3">
                                    <asp:DropDownList ID="ddl_productgroup" runat="server"
Visible="true" Enabled="true"
OnSelectedIndexChanged="ddl_productgroup_SelectedIndexChanged" Width="200px">
                                        </asp:DropDownList>
                                    </td>
                                </tr>
                            </tbody>
                    </table>
                </div>
            </div>
        </div>
    </div>

```

```

                <td>                <asp:Button ID="btn_retrieve" runat="server"
CssClass="=btn btn-secondary btn-md btn rounded" Text="Retrive Data"
OnClick="btn_retrieve_Click" />
</td>
            </tr>
        </tbody></table>
    </div>
</div>
</section>

<section class="mb-5">
    <!-- Card -->
    <div class="card card-cascade narrower">
        <section>
            <div class="card">
                <div id="prtable" class="card-body" runat="server">
                    <asp:Table ID="Dtable" runat="server" border="1" CssClass="table table-striped
table-detail col-lg-6 align-items-sm-center" >
</asp:Table>
                </div>
            </div>
        </section>
    </div>
    </div>
</section>
    <!-- Card -->
</div>
</section>
</div>
</asp:Content>
<asp:Content ID="Content3" ContentPlaceHolderID="ContentPlaceHolder1"
Runat="Server">

```

</asp:Content>

*****VendorProductsAdd.aspx.cs *****

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBEntity;
using DowBAL;
using System.Xml;
using System.Xml.Serialization;
using System.IO;
using System.Text;
using System.Data.SqlClient;
using System.Configuration;
using System.Data;
using System.Web.Security;
```

```
public partial class VendorProductsAdd : System.Web.UI.Page
```

```
{
    string PageType;
    int Id;
    DDLogTracer objlog = new DDLogTracer();
```

```
protected void Page_Load(object sender, EventArgs e)
```

```
{
    Id = Convert.ToInt16(Session["Pgid"].ToString());
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    if (Clsusr != null)
    {
```

```
        string msg = Request.QueryString["msg"];
        if (msg != null)
        {
            Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
        }
    }
```

```
    if (Convert.ToInt16(Clsusr.GroupId) > 3)
```

```
    {
        PopDtls ObjPD = new PopDtls();
        ObjPD.BindDropDowns(ddl_productcat, (new
PopDtls()).GetProductCatbyType(1));
        Binddetails(Id);
    }
```

```
    }
```

```
    else
```

```
    {
        Response.Redirect("MLB_Login.aspx");
    }
```

```
    }
private void Binddetails(int grid)
```

```

{
    Session["Pgid"] = grid.ToString();
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    List<Eteproduct> Lst = new List<Eteproduct>();
    List<Eteproduct> cnts = new Products().GetSellerProductsbygrid(grid,
Convert.ToInt16(Clsusr.UserId));
    Lst = new Products().GetProductbyGroup(grid, "");
    CheckBox chk = new CheckBox();
    Dtable.Rows.Clear();
    //try
    // {
    if (Lst != null && Lst.Count > 0)
    {
        TableRow tr = new TableRow();
        TableCell td = new TableCell();
        TableHeaderRow thr = new TableHeaderRow();
        td = new TableCell();
        chk.ID = "CheckBoxAll";
        td.Controls.Add(chk);
        thr.Cells.Add(td);
        td = new TableCell();
        td.Text = "Image";
        thr.Cells.Add(td);
        td = new TableCell();
        td = new TableCell();
        td.Text = "Category Name";
        thr.Cells.Add(td);
        td = new TableCell();
        td.Text = "Product Name";
        thr.Cells.Add(td);
        td = new TableCell();
        td.Text = "Selected Date";
        thr.Cells.Add(td);

        Dtable.Rows.Add(thr);
        tr = new TableRow();
        td = new TableCell();

        StringBuilder sb = new StringBuilder();

        DateTime dts;

        foreach (var dt in Lst)
        {
            int pcount = (from a in cnts where a.ProductId == dt.ProductId select a).Count();
            if (pcount > 0)
            { dts = (from b in cnts where b.ProductId == dt.ProductId select
b.CreatedOn).FirstOrDefault(); }
            else { dts = DateTime.Today.Date; }
            tr = new TableRow();
            td = new TableCell();
            chk = new CheckBox();
            chk.ID="CheckBox" + dt.ProductId ;
            chk.Checked = true;
            if (pcount > 0)

```

```

        {
            chk.Checked = true;
        }
        else
        {
            chk.Checked = false;
        }
        td.Controls.Add(chk);
        tr.Cells.Add(td);
        td = new TableCell();
        string img = "<img src=\"" + dt.imageurl + "\" runat='server'
style='height:100px;width:100px' /> ";
        td.Text = img;
        tr.Cells.Add(td);
        td = new TableCell();
        td.Text = "<span class='badge badge-primary'>" + dt.CatName + "</span> <span
class='badge badge-primary'>" + dt.GroupName + "</span>";
        tr.Cells.Add(td);
        td = new TableCell();
        td.Text = "<span class='badge badge-primary'>" + dt.ProductName + "</span>";
        tr.Cells.Add(td);
        td = new TableCell();
        td.Text = Convert.ToString(dts).Substring(0, 10);
        tr.Cells.Add(td);
        Dtable.Rows.Add(tr);
    }
}
}
private void Bindetails1(int grid)
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    List<Eteproduct> Lst =new List<Eteproduct>();
    List<Eteproduct> cnts=new Products().GetSellerProductsbygrid(grid,Convert.ToInt16(
Clsusr.UserId));
    Lst=new Products().GetProductbyGroup(grid,"");
    //try    //{
    StringBuilder sb = new StringBuilder();
    sb.Append("<table class='table' runat='server'>\n");
    sb.Append("    <thead class='mdb-color darken-3'> \n");
    sb.Append("    <tr class='text-white'>    \n");
    sb.Append("    <th scope='row'>    \n");
    sb.Append("    <asp:CheckBox ID='CheckBoxAll' runat='server' Checked='false' />
\n");
    sb.Append("    <label for='checkbox1' class='label-table'></label> \n");
    sb.Append("    </th> \n");
    sb.Append("    <th><i class='far fa-image'></i></th>    \n");
    sb.Append("    <th style='width:20%'>Name</th>    \n");
    //sb.Append("    <th>Stock</th>    \n");
    //sb.Append("    <th>Price</th>    \n");
    sb.Append("    <th style='width:30%'>Categories</th>    \n");
    sb.Append("    <th>Date</th>    \n");
    sb.Append("    </tr> \n");
    sb.Append("    </thead>    \n");
}
}
}

```

```

sb.Append("    <!-- Table head --> \n");
sb.Append("    <!-- Table body --> \n");
sb.Append("    <tbody> \n");

DateTime dts;
if (Lst != null && Lst.Count > 0)
{
    foreach (var dt in Lst)
    {
        int pcount=(from a in cnts where a.ProductId==dt.ProductId select a).Count();
        if (pcount > 0)
        { dts = (from b in cnts where b.ProductId == dt.ProductId select
b.CreatedOn).FirstOrDefault(); }
        else { dts = DateTime.Today.Date; }
        sb.Append("        <tr> \n");
        sb.Append("        <td scope='row'> \n");
        if(pcount>0)
        {
            sb.Append("        <asp:CheckBox ID='CheckBox"+dt.ProductId+"\" runat='server'
Checked='true' /> \n");
        }
        else
        {
            sb.Append("        <asp:CheckBox ID='CheckBox" + dt.ProductId + "\"
runat='server' Checked='false' /> \n");
        }

        //sb.Append("        <label for='checkbox2' class='label-table'></label> \n");
        sb.Append("        </td> \n");
        sb.Append("        <td><img src='"+dt.imageUrl+ "\" class='img-fluid'
style='height:100px;width:100px'/></td> \n");
        sb.Append("        <td><a href='#'" + dt.ProductName + "></a></td> \n");
        //sb.Append("        <td><s>₹" + dt.Price + "</s><br>₹" + dt.Price + "</td>
\n");
        sb.Append("        <td><span class='badge badge-primary'" + dt.CatName +
"</span> <span class='badge badge-primary'" + dt.GroupName + "</span> </td>
\n");
        sb.Append("        <td>Selected<br>" + Convert.ToString(
dts).Substring(0,10)+"</td> \n");
        sb.Append("        </tr> \n");

    }
    sb.Append("    </tbody></table> \n");

    prtable.InnerHtml = sb.ToString();
    objlog.Trace_Log(1, "home", "1", "menu", "\n" + sb.ToString());
}
}
protected void btn_save_Click(object sender, EventArgs e)
{
    bool ret = false;
    int grid = Convert.ToInt16(Session["Pgid"].ToString());
    string user = "";
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];

```



```

if (Clsusr == null)
{
    HttpCookie myCookie = Request.Cookies["DateOrWedCookie"];
    if (myCookie != null)
    {
        user = myCookie.Value.ToString();
    }
    else
    {
        user = "Guest";
    }
    guest_login(user);
}
//try
{
    CheckBox cbox = new CheckBox();
    cbox = (CheckBox)Dtable.FindControl("CheckBoxAll");
    List<Eteproduct> Lst = new Products().GetProductbyGroup(grid, "");
    if (cbox.Checked)
    {
        foreach (var list in Lst)
        {
            ret= update(list.ProductId,true);
        }
    }
    else
    {
        foreach (var list in Lst)
        {
            cbox = (CheckBox)Dtable.FindControl("CheckBox" + list.ProductId);
            if (cbox.Checked)
            {
                ret= update(list.ProductId,true);
            }
            else
            {
                ret = update(list.ProductId, false); }
        }
    }
    Response.Redirect("~/VendorHome.aspx?pageid=1&msg=Product Updated
Sucessfully");
}

//catch (Exception exc)
//{{
// //objlog.Trace_Error(4, "DateOrWed", "MangeDealer", "Load", "Exception:" +
exc.ToString());
// Common.ShowMessageHtml(exc.ToString(), 0, true, dvMsgBox);

//}}

}
protected bool update(decimal prid,bool chk)

```

```

{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];

    bool ret = new Products().UpdSellerProduct(prid, Convert.ToInt16( Clsusr.UserId), chk);
    return ret;
}
protected void guest_login(string user)
{

    string City = "Hyderabad";// _City;
    string Country = "India";// _Country;

    BALCreateUser objlogin = new BALCreateUser();
    DataTable status_login = objlogin.LoginGuest(user, user,
Request.ServerVariables["REMOTE_ADDR"].ToString(), "", City, Country);
    string login_status = status_login.Rows[0]["UserStatus"].ToString();

    ClsLoggedInUser objLoggedInUser;
    if (login_status == "True")
    {
        objLoggedInUser = new ClsLoggedInUser();

        objLoggedInUser.UserId = status_login.Rows[0]["UserId"].ToString();
        objLoggedInUser.UserName = status_login.Rows[0]["UserName"].ToString();
        objLoggedInUser.GroupId = status_login.Rows[0]["GroupId"].ToString();
        objLoggedInUser.LoginStatus = "True";
        objLoggedInUser.LoginID = Convert.ToInt32(status_login.Rows[0]["LoginId"]);
        objLoggedInUser.DealerId = Convert.ToInt32(status_login.Rows[0]["DealerId"]);
        objLoggedInUser.RoleId = Convert.ToInt32(status_login.Rows[0]["RoleId"]);
        objLoggedInUser.BrandId = 1; //Convert.ToInt32(status_login.Rows[0]["BrandId"]);

        Session["LoggedInUser_Specs"] = objLoggedInUser;

        FormsAuthentication.SetAuthCookie(objLoggedInUser.UserName, false);
        HttpCookie myCookie = new HttpCookie("DateOrWedCookie");
        DateTime now = DateTime.Now;
        myCookie.Value = objLoggedInUser.UserName;
        myCookie.Expires = now.AddYears(1);
        Response.Cookies.Add(myCookie);

    }
}

protected void btn_retrieve_Click(object sender, EventArgs e)
{
    Bindetails(Convert.ToInt16(ddl_productgroup.SelectedValue));
}

protected void btn_clear_Click(object sender, EventArgs e)
{
}
}

```

```

protected void ddl_productcat_SelectedIndexChanged(object sender, EventArgs e)
{
    int prcat =Convert.ToInt16( ddl_productcat.Selectedvalue);
    if (prcat == 0) { prcat = 2; }
    PopDlIs ObjPD = new PopDlIs();
    ObjPD.BindDropDowns(ddl_productgroup, (new
PopDlIs().GetproductGroupbyCat(prcat)));

    ddl_productgroup.Enabled = true;
}
protected void ddl_productgroup_SelectedIndexChanged(object sender, EventArgs e)
{
    Session["Pgid"] = ddl_productgroup.Selectedvalue;
    Bindetails(Convert.ToInt16(ddl_productgroup.Selectedvalue));
}
}
}
***** VendorProductsAddUser.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/ProductMaster.master"
AutoEventWireup="true" CodeFile="VendorProductsAddUser.aspx.cs"
Inherits="VendorProductsAddUser" %>
<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
Runat="Server">
</asp:Content>
<asp:Content ID="Content4" ContentPlaceHolderID="CPH_TopMessages" Runat="Server">
    <div class="col-lg-12 col-md-12 col-sm-12">
        <div class="row">
            <%--<h3>Ravikiran Techno Soft</h3>--%>
            <div id="dvMsgBox" class="span10 ordersave" runat="server" style="display:
block;">
                <%--
                    <asp:Label ID="Label2" runat="server" Text=""
Height="100px"></asp:Label>--%>
                    <asp:Label ID="Label3" runat="server" Text="" Height="60px"></asp:Label>
                </div>
            </div>
        </div>
        <div class="container-fluid">
            <div class="row-fluid" style="padding-top: 20px;">
                <div>
                    <h5 class="my-4 dark-grey-text font-weight-bold" style="padding-left:20px">Manage
Selected Products
                    <asp:Button ID="btn_save" runat="server" CssClass="btn btn-secondary btn-md"
Text="Save" OnClick="btn_save_Click" /> <%-- btn-rounded--%>
                    <asp:Button ID="btn_clear" runat="server" CssClass ="btn btn-secondary btn-md btn
rounded" Text="Clear" OnClick="btn_clear_Click" />
                <%--
                    <a href="#" class="btn btn-secondary btn-md btn-rounded"> &nbsp;Edit</a> <%--
                <i class="fas fa-edit"></i>
                    <a href="#" class="btn btn-secondary btn-md btn-rounded"> &nbsp;Delete</a><%--<i
class="fas fa-trash"></i>--%>
                </h5>
            <!-- Navbar links -->

                </div>
            <div class="span10">
                <div class="row">

```

```

        <div id="Div1" class="span12 ordersave" runat="server" style="display: none;">
            <asp:Label ID="Label1" runat="server" Text=""></asp:Label>
        </div>
    </div>
</div>
</div>
</div>

<%-- <div id="LeftSideBar" runat="server" class="span2">
    <asp:Table ID="Table1" runat="server">
        </asp:Table>
    </div>--%>
</div>
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="CPH_MainContent" Runat="Server" >
    <div class="container-fluid" style="padding-left:100px; padding-right:50px">

        <!-- Navbar links -->
        <section class="mb-5">

            <!-- Card -->
            <div class="card card-cascade narrower">

                <div class=" col-lg-8 col-md-8 col-sm-6 category-num">
                    <table id="Table5" class="table table-striped table-detail col-lg-8"
runat="server">

                        <tbody>

                            <tr>
                                <td class="span3" dir="auto" style="width:170px">Product
Category : </td>

                                <td class="span3">
                                    <asp:DropDownList ID="ddl_productcat" runat="server"
OnSelectedIndexChanged="ddl_productcat_SelectedIndexChanged" Visible="true"
Width="200px">
                                        <asp:ListItem Value="0">Bridal</asp:ListItem>
                                        <asp:ListItem Value="1" >Womens</asp:ListItem>
                                    </asp:DropDownList>
                                    </td><td>
                                        <asp:Button ID="retgroup" runat="server" CssClass="btn btn-
secondary btn-md " OnClick="ddl_productcat_SelectedIndexChanged" Text="Retrieve Sub"
/>
                                    </td>
                                </tr><tr>
                                <td class="span3" dir="auto" style="width:200px">Product Sub-
Category : </td>

                                <td class="span3">
                                    <asp:DropDownList ID="ddl_productgroup" runat="server"
Visible="true" Enabled="true"
OnSelectedIndexChanged="ddl_productgroup_SelectedIndexChanged" Width="200px">

```

```

                </asp:DropDownList>
            </td>
            <td>
                <asp:Button ID="btn_retrieve" runat="server"
                CssClass="=btn btn-secondary btn-md btn rounded" Text="Retrive Data"
                OnClick="btn_retrieve_Click" />
            </td>
        </tr>
    </tbody></table>
</div>
</section>

<section class="mb-5">
    <!-- Card -->
    <div class="card card-cascade narrower">
        <section>
            <div class="card">
                <div id="prtable" class="card-body" runat="server">
                    <asp:Table ID="Dtable" runat="server" border="1" CssClass="table table-striped
                    table-detail col-lg-6 align-items-sm-center" >
                </asp:Table>
            </div>
            <!-- Table -->
            <!-- Table -->
            <hr class="my-0"/>
            <!-- Bottom Table UI -->
        </div>
    </div>
</section>

    <!-- Card -->
</div>
</section>

</div>
</asp:Content>

```

```
<asp:Content ID="Content3" ContentPlaceHolderID="ContentPlaceHolder1"
Runat="Server">
```

```
</asp:Content>
```

```
***** VendorProductsAddUser.aspx.cs *****
```

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBEntity;
using DowBAL;
using System.Xml;
using System.Xml.Serialization;
using System.IO;
using System.Text;
using System.Data.SqlClient;
using System.Configuration;
using System.Data;
using System.Web.Security;
```

```
public partial class VendorProductsAddUser : System.Web.UI.Page
```

```
{
    string PageType;
    int Id;
    DDLogTracer objlog = new DDLogTracer();
```

```
protected void Page_Load(object sender, EventArgs e)
```

```
{
    Id = Convert.ToInt16(Session["Pgid"].ToString());
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    if (Clsusr != null)
    {
```

```
        string msg = Request.QueryString["msg"];
        if (msg != null)
        {
            Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
        }
    }
```

```
    if (Convert.ToInt16(Clsusr.GroupId) > 3)
    {
```

```
        PopDtls ObjPD = new PopDtls();
        ObjPD.BindDropDowns(ddl_productcat, (new
PopDtls()).GetProductCatbyType(1));
        Bindetails(Id);
    }
```

```
    }
    else
```

```
    {
        Response.Redirect("MLB_Login.aspx");
    }
}
```

```

}
private void Bindetails(int grid)
{
    Session["Pgid"] = grid.ToString();
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    List<Eteproduct> Lst = new List<Eteproduct>();
    List<Eteproduct> cnts = new Products().GetSellerProductsbygrid(grid,
Convert.ToInt16(Clsusr.UserId));
    Lst = new Products().GetProductbyGroup(grid, "");
    CheckBox chk = new CheckBox();
    Dtable.Rows.Clear();
    //try
    // {
    if (Lst != null && Lst.Count > 0)
    {
        TableRow tr = new TableRow();
        TableCell td = new TableCell();
        TableHeaderRow thr = new TableHeaderRow();
        td = new TableCell();
        chk.ID = "CheckBoxAll";
        td.Controls.Add(chk);
        thr.Cells.Add(td);
        td = new TableCell();
        td.Text = "Image";
        thr.Cells.Add(td);
        td = new TableCell();
        td = new TableCell();
        td.Text = "Category Name";
        thr.Cells.Add(td);
        td = new TableCell();
        td.Text = "Product Name";
        thr.Cells.Add(td);
        td = new TableCell();
        td.Text = "Selected Date";
        thr.Cells.Add(td);

        Dtable.Rows.Add(thr);
        tr = new TableRow();
        td = new TableCell();

        StringBuilder sb = new StringBuilder();

        DateTime dts;

        foreach (var dt in Lst)
        {
            int pcount = (from a in cnts where a.ProductId == dt.ProductId select a).Count();
            if (pcount > 0)
            { dts = (from b in cnts where b.ProductId == dt.ProductId select
b.CreatedOn).FirstOrDefault(); }
            else { dts = DateTime.Today.Date; }
            tr = new TableRow();
            td = new TableCell();
            chk = new CheckBox();
            chk.ID="CheckBox" + dt.ProductId ;

```

```

        chk.Checked = true;
        if (pcount > 0)
        {
            chk.Checked = true;
        }
        else
        {
            chk.Checked = false;
        }
        td.Controls.Add(chk);
        tr.Cells.Add(td);
        td = new TableCell();
        string img = "<img src=\"" + dt.imageurl + "\" runat='server'
style='height:100px;width:100px' /> ";
        td.Text = img;
        tr.Cells.Add(td);
        td = new TableCell();
        td.Text = "<span class='badge badge-primary'>" + dt.CatName + "</span> <span
class='badge badge-primary'>" + dt.GroupName + "</span>";
        tr.Cells.Add(td);
        td = new TableCell();
        td.Text = "<span class='badge badge-primary'>" + dt.ProductName + "</span>";
        tr.Cells.Add(td);
        td = new TableCell();
        td.Text = Convert.ToString(dts).Substring(0, 10);
        tr.Cells.Add(td);
        Dtable.Rows.Add(tr);
    }
}
}
private void Bindetails1(int grid)
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    List<Eteproduct> Lst = new List<Eteproduct>();
    List<Eteproduct> cnts = new Products().GetSellerProductsbygrid(grid, Convert.ToInt16(
Clsusr.UserId));
    Lst = new Products().GetProductbyGroup(grid, "");
    //try    //{
    StringBuilder sb = new StringBuilder();
    sb.Append("<table class='table' runat='server'>\n");
    sb.Append("    <thead class='mdb-color darken-3'> \n");
    sb.Append("        <tr class='text-white'>        \n");
    sb.Append("            <th scope='row'>        \n");
    sb.Append("            <asp:CheckBox ID='CheckBoxAll' runat='server' Checked='false' />
\n");
    sb.Append("            <label for='checkbox1' class='label-table'></label> \n");
    sb.Append("        </th> \n");
    sb.Append("        <th><i class='far fa-image'></i></th>        \n");
    sb.Append("        <th style='width:20%'>Name</th>        \n");
    //sb.Append("        <th>Stock</th>        \n");
    //sb.Append("        <th>Price</th>        \n");
    sb.Append("        <th style='width:30%'>Categories</th>        \n");
    sb.Append("        <th>Date</th>        \n");
}
}
}

```



```

sb.Append("    </tr> \n");
sb.Append("    </thead> \n");
sb.Append("    <!-- Table head --> \n");
sb.Append("    <!-- Table body --> \n");
sb.Append("    <tbody> \n");

DateTime dts;
if (Lst != null && Lst.Count > 0)
{
    foreach (var dt in Lst)
    {
        int pcount=(from a in cnts where a.ProductId==dt.ProductId select a).Count();
        if (pcount > 0)
        { dts = (from b in cnts where b.ProductId == dt.ProductId select
b.CreatedOn).FirstOrDefault(); }
        else { dts = DateTime.Today.Date; }
        sb.Append("    <tr> \n");
        sb.Append("    <td scope='row'> \n");
        if(pcount>0)
        {
            sb.Append("    <asp:CheckBox ID='CheckBox'+dt.ProductId+'\" runat='server'
Checked='true' /> \n");
        }
        else
        {
            sb.Append("    <asp:CheckBox ID='CheckBox\" + dt.ProductId + \"
runat='server' Checked='false' /> \n");
        }

        //sb.Append("    <label for='checkbox2' class='label-table'></label> \n");
        sb.Append("    </td> \n");
        sb.Append("    <td><img src=\""+dt.imageUrl+ "\" class='img-fluid'
style='height:100px;width:100px'/></td> \n");
        sb.Append("    <td><a href='#\"+dt.ProductName+\"</a></td> \n");
        //sb.Append("    <td><s>₹\" + dt.Price + \"</s><br>₹\" + dt.Price + \"</td>
\n");
        sb.Append("    <td><span class='badge badge-primary'>\" + dt.CatName +
\"</span> <span class='badge badge-primary'>\" + dt.GroupName + \"</span> </td>
\n");
        sb.Append("    <td>Selected<br>\"+Convert.ToString(
dts).Substring(0,10)+\"</td> \n");
        sb.Append("    </tr> \n");
    }
    sb.Append(" </tbody></table> \n");

    prttable.InnerHtml = sb.ToString();
    objlog.Trace_Log(1, "home", "1", "menu", "\n" + sb.ToString());
}
}
protected void btn_save_Click(object sender, EventArgs e)
{
    bool ret = false;
    int grid = Convert.ToInt16(Session["Pgid"].ToString());

```

```

string user = "";
ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
if (Clsusr == null)
{
    HttpCookie myCookie = Request.Cookies["DateOrWedCookie"];
    if (myCookie != null)
    {
        user = myCookie.Value.ToString();
    }
    else
    {
        user = "Guest";
    }
    guest_login(user);
}
//try
{
    CheckBox cbox = new CheckBox();
    cbox = (CheckBox)Dtable.FindControl("CheckBoxAll");
    List<Eteproduct> Lst = new Products().GetProductbyGroup(grid, "");
    if (cbox.Checked)
    {
        foreach (var list in Lst)
        {
            ret= update(list.ProductId,true);
        }
    }
    else
    {
        foreach (var list in Lst)
        {
            cbox = (CheckBox)Dtable.FindControl("CheckBox" + list.ProductId);
            if (cbox.Checked)
            {
                ret= update(list.ProductId,true);
            }
            else
            {
                ret = update(list.ProductId, false); }
        }
    }
    Response.Redirect("~/VendorHome.aspx?pageid=1&msg=Product Updated
Sucessfully");
}

//catch (Exception exc)
//{{
// //objlog.Trace_Error(4, "DateOrWed", "MangeDealer", "Load", "Exception:" +
exc.ToString());
// Common.ShowMessageHtml(exc.ToString(), 0, true, dvMsgBox);

//}}

```

```

}
protected bool update(decimal prid,bool chk)
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];

    bool ret = new Products().UpdSellerProduct(prid,Convert.ToInt16( Clsusr.UserId), chk);
    return ret;
}
protected void guest_login(string user)
{

    string City = "Hyderabad";// _City;
    string Country = "India";// _Country;

    BALCreateUser objlogin = new BALCreateUser();
    DataTable status_login = objlogin.LoginGuest(user, user,
Request.ServerVariables["REMOTE_ADDR"].ToString(), "", City, Country);
    string login_status = status_login.Rows[0]["UserStatus"].ToString();

    ClsLoggedInUser objLoggedInUser;
    if (login_status == "True")
    {
        objLoggedInUser = new ClsLoggedInUser();

        objLoggedInUser.UserId = status_login.Rows[0]["UserId"].ToString();
        objLoggedInUser.UserName = status_login.Rows[0]["UserName"].ToString();
        objLoggedInUser.GroupId = status_login.Rows[0]["GroupId"].ToString();
        objLoggedInUser.LoginStatus = "True";
        objLoggedInUser.LoginID = Convert.ToInt32(status_login.Rows[0]["LoginId"]);
        objLoggedInUser.DealerId = Convert.ToInt32(status_login.Rows[0]["DealerId"]);
        objLoggedInUser.RoleId = Convert.ToInt32(status_login.Rows[0]["RoleId"]);
        objLoggedInUser.BrandId = 1; //Convert.ToInt32(status_login.Rows[0]["BrandId"]);

        Session["LoggedInUser_Specs"] = objLoggedInUser;

        FormsAuthentication.SetAuthCookie(objLoggedInUser.UserName, false);
        HttpCookie myCookie = new HttpCookie("DateOrWedCookie");
        DateTime now = DateTime.Now;
        myCookie.Value = objLoggedInUser.UserName;
        myCookie.Expires = now.AddYears(1);
        Response.Cookies.Add(myCookie);

    }
}

protected void btn_retrieve_Click(object sender, EventArgs e)
{
    Bindetails(Convert.ToInt16(ddl_productgroup.SelectedValue));
}

protected void btn_clear_Click(object sender, EventArgs e)
{

```



```
right">12</span></a></li>
    <li><a href="#">Lehengas <span class="pull-
right">11</span></a></li>
    <li><a href="#">Indo Westers <span class="pull-
right">10</span></a></li>
    <li><a href="#">Salwar Kameez <span class="pull-
right">14</span></a></li>
    <li><a href="#">Kurtees <span class="pull-
right">15</span></a></li>
```

```
</ul>
```

```
</div>
```

```
<!-- END CATEGORIES WIDGET -->
```

```
<!-- SIDE-BAR PRODUCT WIDGET -->
```

```
<div id="toprate" class="" runat="server">
```

```
<div class="widget mb30">
```

```
<h6 class="text-uppercase bottom-line">Top Rated</h6>
```

```
</div>
```

```
</div>
```

```
<!-- END SIDE-BAR PRODUCT WIDGET -->
```

```
<!-- SIDE-BAR PRODUCT WIDGET -->
```

```
<div id="bestseller" class="" runat="server">
```

```
<div class="widget mb30">
```

```
<h6 class="text-uppercase bottom-line">Best Sellers</h6>
```

```
</div>
```

```
<div class="small-thumb-proudct mb30">
```

```
<div class="thumb-product">
```

```

```

```
<div class="thumb-product-inner">
```

```
<h6><a href="shoppingdetail.aspx">Printed Su..</a></h6>
```

```
<span class="thumb-product-cat">Fashion</span>
```

```
</div>
```

```
<ul class="list-inline overflow-h">
```

```
<li class="thumb-product-price line-through">$95.00</li>
```

```
<li class="thumb-product-price">$75.00</li>
```

```
</ul>
```

```
</div>
```

```
</div>
```

```
<div class="small-thumb-proudct mb30">
```

```
<div class="thumb-product">
```

```

```

```
<div class="thumb-product-inner">
```

```
<h6><a href="shoppingdetail.aspx">Printed Su..</a></h6>
```

```
<span class="thumb-product-cat">Fashion</span>
```

```
</div>
```

```
<ul class="list-inline overflow-h">
```

```
<li class="thumb-product-price line-through">$70.00</li>
```

```
<li class="thumb-product-price">$60.00</li>
```

```
</ul>
```

```
</div>
```

```
</div>
```

```

<div class="small-thumb-proudct mb30">
  <div class="thumb-product">
    
    <div class="thumb-product-inner">
      <h6><a href="shoppingdetail.aspx">Printed Su..</a></h6>
      <span class="thumb-product-cat">Fashion</span>
    </div>
    <ul class="list-inline overflow-h">
      <li class="thumb-product-price">$65.00</li>
    </ul>
  </div>
</div>
</div>

```

<!-- END SIDE-BAR PRODUCT WIDGET -->

```

<!-- TAGS WIDGET -->
<div class="widget">
  <h6 class="text-uppercase bottom-line">Tags</h6>
  <div class="tags">
    <a href="#">Fashion</a>
    <a href="#">Electronics</a>
    <a href="#">Services</a>
    <a href="#">Businesses</a>
    <a href="#">Shops</a>
    <a href="#">Markets</a>
  </div>
</div>
<!-- END TAGS WIDGET -->
</div>
</aside>

```

```

<div class=" col-lg-12 col-md-12 col-sm-12 col-xs-12">

```

```

  <div class="row hidden-xs">
    <div class="toolbar">
      <div id="results" class="col-sm-4 category-num" runat="server">
        <h2>Designer</h2>
        <small class="category-num-coun">9 Results</small>
      </div>
    </div>
    <div class="col-sm-8 pull-right">
      <ul class="view-mode">
        <li class="grid-list-icons">
<%--
          <asp:Button ID="btn1" runat="server" OnClick="btn1_Click"
type="button" class="btn btn-floating color-green " Text="List" /><i class='fa fa-th-list'></i>
          <asp:Button ID="btn2" runat="server" OnClick="btn2_Click" type="button"
class="btn btn-floating color-green " Text="Grid" /><i class='fa fa-th'></i> wrn-btn--%>
          <asp:ImageButton ID="ImgBtn1" runat="server"
ImageUrl="images/small/list.png" OnClick="ImgBtn1_Click" />
          <asp:ImageButton ID="ImgBtn2" runat="server"
ImageUrl="images/small/grid.png" OnClick="ImgBtn2_Click" />

```

```

<%--
    <button id="Button1" type="button" class="btn btn-secondary wrn-
btn mt-0 ml-0" onclick="btn1_Click" ><i class='fa fa-th-list'></i>&nbsp;&nbsp; List</button>
    <button id="Button2" type="button" style="width:10px" class="btn btn-
product " onclick="btn2_Click" ><i class='fa fa-th'></i></button>--%>
    </li>
    <li class="sort-list-btn">
    <h6>Sort By :</h6>
    <div class="btn-group">
    <button type="button" class="btn btn-default dropdown-toggle"
data-toggle="dropdown">
        Relevent <span class="caret"></span>
    </button>
    <ul class="dropdown-menu" role="menu">
    <li><a href="#">Relevent</a></li>
    <li><a href="#">Price High to Low</a></li>
    <li><a href="#">Price Low to High</a></li>
    <li><a href="#">New Arrived</a></li>
    </ul>
    </div>
    </li>
    <li class="sort-list-btn">
    <h6>Show :</h6>
    <div class="btn-group">
    <button type="button" class="btn btn-default dropdown-toggle"
data-toggle="dropdown">
        30 <span class="caret"></span>
    </button>
    <ul class="dropdown-menu" role="menu">
    <li><a href="#">30</a></li>
    <li><a href="#">45</a></li>
    <li><a href="#">60</a></li>
    <li><a href="#">90</a></li>
    </ul>
    </div>
    </li>
    </ul>
</div>

</div>
<div class="col-md-12 col-sm-12 col-xs-12">
    <div class="row hidden-xs">
    <div class="toolbar">
    <div id="results" class="col-sm-8 category-num" runat="server">
    <h2>Designer</h2>
    <small class="category-num-coun">9 Results</small>
    </div>
    </div>
</div>
</div>
</div>
<%--dynamic code from here--%>
    <div id="appvendors" class="category-products" runat="server" style="display:none">
</div>

```

```
</div>
```

```
</div>
```

```
<!--dynamic code ends here-->
```

```
<div id="AppGrid" class="category-products col-grid-style col-lg-12" runat="server" style="display:block">  
<div>
```

```
</div>
```

```
</div>
```

```
<!--dynamic code ends here-->
```

```
<!-- Start Pagination-->
```

```
<div id="paging" class="cd-pagination" runat="server"> <!--href="#"-->  
<ul class="pagination">  
<li><a href="#"><i class="fa fa-angle-left" aria-hidden="true"></i></a></li>  
<li class="active"><a onclick="javascript:ShowPage('DivL1','DivG1')">1</a></li>  
<li><a onclick="javascript:ShowPage('DivL2','DivG2')">2</a></li>  
<li><a onclick="javascript:ShowPage('DivL3','DivG3')">3</a></li>  
<li><a onclick="javascript:ShowPage('DivL4','DivG4')">4</a></li>  
<li><a onclick="javascript:ShowPage('DivL5','DivG5')">5</a></li>  
<li><a onclick="javascript:ShowPage('DivL6','Div6')"><i class="fa fa-angle-right" aria-hidden="true"></i></a></li>  
</ul>  
</div>  
</div>
```

```
<script type="text/javascript">
```

```
function ShowPage(Div,Div1)
```

```
{
```

```
    alert("test");
```

```
    document.getElementById('DivL1').style.display = "none";
```

```
    document.getElementById('DivL2').style.display = "none";
```

```
    document.getElementById('DivL3').style.display = "none";
```

```
    document.getElementById('DivL4').style.display = "none";
```

```
    document.getElementById('DivL5').style.display = "none";
```

```
    document.getElementById('DivL6').style.display = "none";
```

```
    document.getElementById('DivG1').style.display = "none";
```

```
    document.getElementById('DivG2').style.display = "none";
```

```
    document.getElementById('DivG3').style.display = "none";
```

```
    document.getElementById('DivG4').style.display = "none";
```

```
    document.getElementById('DivG5').style.display = "none";
```

```
    document.getElementById('DivG6').style.display = "none";
```

```
    document.getElementById(Div).style.display = "block";
```

```
    document.getElementById(Div1).style.display = "block";
```

```
    document.getElementById(Div).classList.add("active");
```

```
    document.getElementById(Div1).classList.add("active");
```

```
}
```



```
        </script>
    <!-- End Pagination-->
</div>
    </div>

    <!-- END MAIN CONTAIN -->

    <!-- ./Content Bodey Here -->
```

```
</asp:Content>
<asp:Content ID="Content6" ContentPlaceHolderID="ContentFooter" Runat="Server">
</asp:Content>
```

```
*****VendorProductsShow.aspx.cs *****
```

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBAL;
using DowBEntity;
//using LogTracer;
using System.Text;
using System.Text.RegularExpressions;
```

```
public partial class VendorProductsShow : System.Web.UI.Page
{
```

```
    DDLogTracer objlog = new DDLogTracer();
    bool IsMobile = false;
    protected void Page_Load(object sender, EventArgs e)
    {
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr != null)
        {
        }
        else
        {
            Response.Redirect("~/MLB_Login.aspx");
        }
        if (Request.Browser.IsMobileDevice) { IsMobile = true; } else { IsMobile = false; }
        if (IsMobile)
        {
            sidebar.Attributes.CssStyle.Add("display", "none;");
        }
        //if (!IsPostBack)
        { Bindetails(); }
    }
}
```

```
private void Bindetails()
{
```

```
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
}
```

```

string pt = "cat",title="",vendor;
int pid = 0;
    ImgBtn1.Enabled = false;
ImgBtn2.Enabled = true;
appendvords.Attributes.CssStyle.Add("display", "none;");
AppGrid.Attributes.CssStyle.Add("display", "block;");
int srno = 1, perpage = 3, dvno = 1,sno=1;
string cat = Request.QueryString["cid"];
if (cat != "" || cat != null)
{
    pid = Convert.ToInt16(cat);
}
string PreHtml = "";
StringBuilder sb = new StringBuilder();
StringBuilder sbg = new StringBuilder();
vendor = Request.QueryString["vendor"];

sb.Append(" <div class='directory-list'>");
    if (pid == 0)
    {
        pid = Convert.ToInt16(Request.QueryString["pid"]);
        pt = "prod";
    }

    //try
    {
        IList<ProductProductSeller> Lst=new List<ProductProductSeller>();
        if (vendor != null)
        {
            Lst = (new Products()).GetProductSellerbySellerId(Convert.ToInt16(vendor), 1);
            pt = "vendor";
        }
        else if (pt == "prod")
        {
            Lst = (new Products()).GetProductSellerbyProductId(pid, "SearchList");
            if (Lst.Count() == 0 && pid!=0)
            {
                Eteproduct ep = new Products().GetProduct(pid, "");
                string pname = ep.ProductName;
                Lst = new Products().GetProductSellerbySearch(pname, 1);
            }
        }
        else
        {
            Lst = (new Products()).GetProductSellerbyCatId(pid, 1, "SearchList");
        }
        if (Lst.Count > 0)
        {
            ProductProductSeller pst = (from c in Lst select c).FirstOrDefault();

            if (pt == "prod")
            {
                title = pst.ProductName;
            }
        }
    }

```

```

else if (pt == "vendor")
{
    title = Lst[0].SellerName;
}
else
{
    title = pst.CatName;
}
sb.Append("<h2>" + title + "</h2>\n <small class='category-num-coun'>" +
Convert.ToString(Lst.Count) + " Results</small>\n");
results.InnerHtml = sb.ToString();
sb = new StringBuilder();
string path = "images/NoImage.jpg";
string path1 = "images/NoImage.jpg";
if (Lst != null && Lst.Count > 0)
{
    sbg.Append("\n <div class='row clearfix'>\n");
    sb.Append("\n <div class='row clearfix'>\n");

    sb.Append("<div id='DivL" + dvno + "' class='products-list products-list-
description-brd mb30' style='display:block'>\n");

    sbg.Append("<div id='DivG" + dvno + "' class='Row clearfix'
style='display:block'> \n");
    sbg.Append(" <div class='row clearfix'>\n");
    foreach (var dt in Lst)
    {
        string href = "";
        href = "ShoppingDetail.aspx?id=" + dt.PSId;
        var list = new Products().GetProductSellerImage((int)dt.PSId, "").ToList();
        int cnt = list.Count();
        if (cnt > 1)
        {
            path = list[0].ImagePath;
            path1 = list[1].ImagePath; //"http://rk.bakhala.com/"+
        }
        else
        if (list.Count() > 0)
        {
            path = list[0].ImagePath;
        }

        if (path == null || path == "")
        {
            path = "images/NoImage.jpg";
        }
        if (path1 == null || path1 == "") { path1 = path; }
        List<EteproductSellerStocks> stk = new
Products().GetSellerStocksbyPsId((int)dt.PSId);
        var col = (from c in stk where c.Stock > 0 orderby c.colId select
c.colId).Distinct();
        var siz = (from d in stk where d.Stock > 0 orderby d.EpOptId select
d.EpOptId).Distinct();

```

```

//list code
    sb.Append(" <div class='products-list products-list-description-brd mb30'>
\n");
    sb.Append(" <div class='row'> \n");
    sb.Append(" <div class='col-sm-4'> \n");
    sb.Append(" <div class='product-grid-img'> \n");
    sb.Append(" <ul class='product-labels'> \n");
    sb.Append(" <li>Sale</li> \n");
    sb.Append(" </ul> \n");
    sb.Append(" <a href='shoppingdetail.aspx?pid=" + dt.PSId + "'
class='product-grid-img-thumbnail'> \n");
    sb.Append("  \n");
    sb.Append("  \n");
    sb.Append(" </a> \n");
    sb.Append(" <div class='product-grid-atb text-center'><i class='fa fa-
shopping-cart'></i><a class=" href='shoppingdetail.aspx?pid=" + dt.PSId + "'>Add to
Cart</a></div> \n");
    sb.Append(" </div> \n");
    sb.Append(" </div> \n");
    sb.Append(" <div class='col-sm-8 products-list-description'> \n");
    sb.Append(" <div class='overflow-h'> \n");
    sb.Append(" <ul class='list-inline overflow-h mb15'> \n");
    sb.Append(" <li><h4 class='title-price'><a href='shoppingdetail.aspx?pid="
+ dt.PSId + "'> + dt.ProductDescription + "</a></h4></li> \n");
    sb.Append(" <li class='pull-right'> \n");
    sb.Append(" <ul class='list-inline product-ratings'> \n");
    sb.Append(" <li><i class='rating-selected fa fa-star'></i></li> \n");
    sb.Append(" <li><i class='rating-selected fa fa-star'></i></li> \n");
    sb.Append(" <li><i class='rating-selected fa fa-star'></i></li> \n");
    sb.Append(" <li><i class='rating fa fa-star'></i></li> \n");
    sb.Append(" <li><i class='rating fa fa-star'></i></li> \n");
    //sb.Append("<li><a href='ServicesList.aspx?vendor=" + dt.SellerId + "'> +
"sold by:" + dt.SellerName + "</a> \n");
    sb.Append(" </ul> \n");
    sb.Append(" </li> \n");
    sb.Append(" </ul> \n");
    sb.Append(" <div class='mb10'> \n");
    sb.Append(" <span class='title-price'>" + dt.Price.ToString() + "</span>
\n");
    sb.Append(" <span class='title-price line-through'>" + dt.Price.ToString() +
"</span> \n");
    sb.Append(" </div> \n");
    sb.Append(" <p class='mb15'>" + dt.SellerDescription + "</p> \n");

//colors and sizes
if(siz.Count()>1)
{
\n");
    sb.Append(" <h3 class='shop-product-title'>Sizes Available</h3>
\n");
    sb.Append(" <ul class='list-inline product-size mb30'> \n");
    string chk = "checked";

    foreach (var sz in siz)

```

```

        {
            string size = new Products().GetSizeByOptlId(sz);
            sb.Append("    <li> \n");
            sb.Append("    <input type='checkbox' id='size-" + sz + "' name='size'
" + chk + "> \n");
            sb.Append("    <label for='size-" + dt.PSId + sz + "'> + size +
"</label> \n");

            sb.Append("    </li> \n");
            chk = "";
        }
        sb.Append("    </ul> \n");
    }
    if (col.Count() > 1)
    {
        sb.Append("    <h3 class='shop-product-title'>Colors Available</h3>
\n");
        sb.Append("    <ul class='list-inline product-color mb30'> \n");
        sb.Append("    <li> \n");
        foreach (var colr in col)
        {
            Eteproduct cl = new Products().GetColorByColId(colr);
            sb.Append("    <li> \n");
            sb.Append("    <input type='radio' id='color-" + cl.ProductId + "' name='color'>
\n");
            sb.Append("    <img id='img' + cl.ProductId + "' src='" + cl.imageurl + "' /></li>");
        }
        sb.Append("    </ul> \n");
    }

    //colors and sizes ends
    sb.Append("<div><a href='ShoppingList.aspx?vendor=" + dt.SellerId +
">" + "Sold by:" + dt.SellerName + "</div> \n");
    sb.Append("    <ul class='list-inline add-to-wishlist mb15'> \n");
    sb.Append("    <li class='product-wishlist'> \n");
    sb.Append("    <i class='fa fa-heart'></i> \n");
    sb.Append("    <a href='MyWishList.aspx?psid=" + dt.PSId + "'>Add to
Wishlist</a> \n");
    sb.Append("    </li> \n");
    sb.Append("    <li class='product-wishlist'> \n");
    sb.Append("    <i class='fa fa-exchange'></i> \n");
    //sb.Append("    <a href='shoppingdetail.aspx?pid=" + dt.PSId + "'>Add
to Compare</a> \n");
    sb.Append("    </li> \n");
    //sb.Append("    </ul> \n"); sb.Append("    <button type='button'
class='btn btn-product'>Add to Cart</button> \n");
    sb.Append("    <a href='MyCart.aspx?pid=" + dt.PSId + "' class='btn
btn-product'>Add to Cart</a> \n");

    sb.Append(" </div> \n");
    sb.Append(" </div> \n");
    sb.Append(" </div> \n");
    sb.Append(" </div> \n");
    sb.Append(" \n");

```

//grid code

```

        sbg.Append("        <div class='col-md-4 col-sm-6 col-xs-12'> \n");
        sbg.Append("        <div class='products grid-product mb30'> \n");
        sbg.Append("        <div class='product-grid-inner'> \n");
        sbg.Append("        <div class='product-grid-img'> \n");
        sbg.Append("        <ul class='product-labels'> \n");
        sbg.Append("        <li>Sale</li> \n");
        sbg.Append("        </ul> \n");
        sbg.Append("        <a href='shoppingdetail.aspx?pid=" + dt.PSId + "'
class='product-grid-img-thumbnail'> \n");
        sbg.Append("         \n");
        sbg.Append("         \n");
        sbg.Append("        </a> \n");
        sbg.Append("        <div class='product-grid-atb text-center'><a class="
href='shoppingdetail.aspx?pid=" + dt.PSId + "'>Quick View</a></div> \n");
        sbg.Append("        </div> \n");
        sbg.Append("        <div class='product-grid-caption text-center'>
\n");
        sbg.Append("        <h6><a class='p-grid-title'
href='shoppingdetail.aspx?pid=" + dt.PSId + "'>" + dt.ProductDescription + "</a></h6>
\n");
        sbg.Append("        <p>" + dt.SellerDescription + "</p> \n");
        //colors and sizes
        if (siz.Count() > 1)
        {
            sbg.Append("        <h3 class='shop-product-title'>Sizes Available</h3>
\n");
            sbg.Append("        <ul class='list-inline product-size mb30'> \n");
            string chk = "checked";

            foreach (var sz in siz)
            {
                string size = new Products().GetSizeByOptId(sz);
                sbg.Append("        <li> \n");
                sbg.Append("        <input type='radio' id='size-" + sz + "' name='size' " +
chk + "> \n");
                sbg.Append("        <label for='size-" + sz + "'>" + size + "</label>
\n");

                sbg.Append("        </li> \n");
                chk = "";
            }
            sbg.Append("        </ul> \n");
        }
        if (col.Count() > 1)
        {
            sbg.Append("        <h3 class='shop-product-title'>Colors Available</h3>
\n");
            sbg.Append("        <ul class='list-inline product-color mb30'> \n");
            sbg.Append("        <li> \n");
            foreach (var colr in col)
            {
                Eteproduct cl = new Products().GetColorByColId(colr);
                sbg.Append("        <li> \n");

```

```

        sbg.Append(" <input type='radio' id='color-" + cl.ProductId + ""
name='color'> \n");
        sbg.Append(" <img id='img" + cl.ProductId + "" src="" + cl.imageurl + ""
/></li>");
    }
    sbg.Append(" </ul> \n");
}
//colors and sizes ends
    sbg.Append("<li><a href='ShoppingList.aspx?vendor=" + dt.SellerId + ">"
+"sold by:" + dt.SellerName + "</a> \n");
    sbg.Append(" <span class='price clearfix'><span class='old-price'>" +
dt.Price.ToString() + "</span> $ <span class='special-price'>" + dt.Price.ToString() +
"</span></span> \n");
    sbg.Append(" <a class='btn btn-product cd-cart-btn'
href='MyCart.aspx?psid=" + dt.PSId + "><i class='fa fa-shopping-cart'></i> Add to Cart</a>
\n");
    sbg.Append(" </div> \n");
    sbg.Append(" </div> \n");
    sbg.Append(" </div> \n");
    sbg.Append(" </div> \n");
    srno++;
    sno++;
    if (sno > 3)
    {
        sbg.Append(" </div> \n");
        sbg.Append(" <div class='row clearfix'>\n");
        sno = 1;
    }
    if (srno > perpage)
    {
        dvno++;
        srno = 1;
        sb.Append(" </div> \n");
        sbg.Append(" </div> \n");
        sb.Append("<div id='DivL" + dvno + "" class='products-list products-list-
description-brd mb30' style='display:none'>\n");

        sbg.Append("<div id='DivG" + dvno + "" class='Row clearfix'
style='display:none'> \n");
    }

}

}

sb.Append(" </div> \n");
sb.Append(" </div> \n");
sbg.Append(" </div> \n");
sbg.Append(" </div> \n");
sbg.Append(" </div> \n");

PreHtml = sb.ToString();
if (PreHtml.Length > 0)
{

```

```

        //objlog.Trace_Error(4, "ShoppingList", "1", "appvendors", PreHtml,
Session["Logpath"].ToString());
        appvendors.InnerHtml = PreHtml;

    }
    PreHtml = sbg.ToString();
    if (PreHtml.Length > 0)
    {
        //objlog.Trace_Log(4, "ShoppingList", "1", "AppGrid", PreHtml,
Session["Logpath"].ToString());
        AppGrid.InnerHtml = PreHtml;

    }
    int i = 1;
    sb = new StringBuilder();
    sb.Append("    <ul class='pagination'>    \n");
    sb.Append("    <li><a href='#'><i class='fa fa-angle-left' aria-
hidden='true'></i></a></li>    \n");
    sb.Append("    <li class='active'><a onclick='javascript:ShowPage('DivL" + i +
", 'DivG" + i + "')>" + i + "</a></li>\n ");
    for ( i = 2; i <= dvno; i++)
    {
        sb.Append("    <li class='"><a onclick='javascript:ShowPage('DivL" + i +
", 'DivG" + i + "')>" + i + "</a></li>\n ");
    }
    sb.Append("    <li><a href='#'><i class='fa fa-angle-right' aria-
hidden='true'></i></a></li>    \n");
    sb.Append("</ul> \n ");
    paging.InnerHtml = sb.ToString();
    objlog.Trace_Log(4, "ShoppingList", "1", "paging", sb.ToString(),
Session["Logpath"].ToString());

    }

    //sb = new StringBuilder();
    //int grp = pst.ProductGroupId;
    //List<Eteproduct> prd = new Products().GetProductbyGroup(grp, "");
    //if (prd.Count > 0)
    //{
    //    // sb.Append("<h6 class='text-uppercase bottom-line'>Show Results
for</h6>\n");
    //    // sb.Append("<ul class='icons-list'>\n");
    //    // foreach (var data in prd)
    //    // {
    //    //     if (Convert.ToInt16( data.Type) > 0)
    //    //     {
    //    //         sb.Append("<li><a href='ShoppingList.aspx?pgid=" + data.ProductId +
">" + data.ProductName + " <span class='pull-right'>" + data.Type + "</span></a></li>\n");
    //    //     }
    //    // }
    //    // sb.Append("</ul>\n");
    //    // Filter.InnerHtml = sb.ToString();
    //}
    //sb = new StringBuilder();
    //sb.Append("    <div class='widget mb30'>    ");

```



```

        //sb.Append("    <h6 class='text-uppercase bottom-line'>Top Rated</h6> ");
        //sb.Append("    </div> ");
        //List<ProductProductSeller> top1 = new
Products().GetProductSellerbySearch("all",1);
        //var top = (from c in top1 orderby c.CurPrice descending select c).Take(10);
        //foreach (var tr in top)
        //{
        //    sb.Append("\n <div class='row clearfix'>\n");

        //    sb.Append("    <div class='small-thumb-proudct mb30'> ");
        //    sb.Append("    <div class='thumb-product'> ");
        //    sb.Append("     ");
        //    //sb.Append("    <div class='thumb-product-inner'> ");
        //    sb.Append("    <span class='thumb-product-cat'>" + tr.CatName + "</span>
");
        //    sb.Append("    <h6><a href='shoppingdetail.aspx?pid=" + tr.PSId + ">" +
tr.ProductDescription + "</a></h6> ");
        //    sb.Append("    </div> ");
        //    sb.Append("    <ul class='list-inline overflow-h'> ");
        //    sb.Append("    <li class='thumb-product-price line-
through'>" + tr.CurPrice + "</li> ");
        //    sb.Append("    <li class='thumb-product-price'>" + tr.CurPrice + "</li> ");
        //    sb.Append("    </ul> ");
        //    sb.Append("    </div> ");
        //    sb.Append("    </div> ");
        //}
        //toprate.InnerHtml = sb.ToString();
        //sb = new StringBuilder();
        //sb.Append("    <div class='widget mb30'> ");
        //sb.Append("    <h6 class='text-uppercase bottom-line'>Best Sellers</h6> ");
        //sb.Append("    </div> ");
        // top = (from c in top1 orderby c.CurPrice ascending select c).Take(10);

        ////List<ProductProductSeller> best = new
Products().GetProductSellerbySearch("");
        //foreach (var tr in top)
        //{
        //    sb.Append("\n <div class='row clearfix'>\n");

        //    sb.Append("    <div class='small-thumb-proudct mb30'> ");
        //    sb.Append("    <div class='thumb-product'> ");
        //    sb.Append("     ");
        //    //sb.Append("    <div class='thumb-product-inner'> ");
        //    sb.Append("    <span class='thumb-product-cat'>" + tr.CatName + "</span>
");
        //    sb.Append("    <h6><a href='shoppingdetail.aspx?pid=" + tr.PSId + ">" +
tr.ProductDescription + "</a></h6> ");
        //    sb.Append("    </div> ");
        //    sb.Append("    <ul class='list-inline overflow-h'> ");
        //    sb.Append("    <li class='thumb-product-price line-through'>" + tr.CurPrice +
"</li> ");
        //    sb.Append("    <li class='thumb-product-price'>" + tr.CurPrice + "</li> ");
        //    sb.Append("    </ul> ");

```

```

        // sb.Append(" </div> ");
        // sb.Append(" </div> ");

        //}
        // bestseller.InnerHtml=sb.ToString();
        //sb = new StringBuilder();
    }
}
// catch (Exception ex)
//{

// Common.ShowMessageHtml(ex.Message, 1, true, this.dvMsgBox);
// ExceptionLogging.SendExcepToDB(ex);

//}
}

protected void btn1_Click(object sender, EventArgs e)
{
    appvendors.Attributes.CssStyle.Add("display", "block;");
    AppGrid.Attributes.CssStyle.Add("display", "none;");
}

protected void btn2_Click(object sender, EventArgs e)
{
    appvendors.Attributes.CssStyle.Add("display", "none;");
    AppGrid.Attributes.CssStyle.Add("display", "block;");
}

protected void ImgBtn1_Click(object sender, ImageClickEventArgs e)
{
    appvendors.Attributes.CssStyle.Add("display", "block;");
    AppGrid.Attributes.CssStyle.Add("display", "none;");
    ImgBtn1.Enabled = false;
    ImgBtn2.Enabled = true;
}

protected void ImgBtn2_Click(object sender, ImageClickEventArgs e)
{
    appvendors.Attributes.CssStyle.Add("display", "none;");
    AppGrid.Attributes.CssStyle.Add("display", "block;");
    ImgBtn1.Enabled = true;
    ImgBtn2.Enabled = false;
}
}
***** VendorProductsUser.aspx*****
<%@ Page Title="" Language="C#" MasterPageFile="~/VendorMasterPage.master"
AutoEventWireup="true" CodeFile="VendorProductsUser.aspx.cs"
Inherits="VendorProductsUser" %>
<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
Runat="Server">
</asp:Content>
<asp:Content ID="Content4" ContentPlaceHolderID="CPH_TopMessages" Runat="Server">
    <div class="span10">
        <div class="row">

```

```

                <%--<h3>Ravikiran Techno Soft</h3>--%>
                <div id="Div1" class="span10 ordersave" runat="server" style="display: block;">
<%--
                <asp:Label ID="Label2" runat="server" Text=""
Height="100px"></asp:Label>--%>
                <asp:Label ID="Label3" runat="server" Text="" Height="60px"></asp:Label>
                </div>
            </div>
        </div>
    </asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentMain" Runat="Server" >
    <div class="container-fluid" style="padding-left:100px; padding-right:50px">

        <div><%--VendorProductsAdd.aspx--%>
            <h5 class="my-3 dark-grey-text font-weight-bold">Products
                <a href="PostProducts.aspx" class="btn btn-outline-danger waves-effect btn-sm font-
weight-bold"><i class="far fa-plus-square"></i> &nbsp;Add New</a>
<%--
                <a href="#" class="btn btn-outline-danger waves-effect btn-sm font-weight-
bold"><i class="fas fa-edit"></i> &nbsp;Edit</a> --%>
                <a href="#" class="btn btn-outline-danger waves-effect btn-sm font-weight-bold"><i
class="fas fa-edit"></i> &nbsp;Edit</a>
                <a href="#" class="btn btn-outline-danger waves-effect btn-sm font-weight-bold"><i
class="fas fa-trash-alt"></i> &nbsp;Delete</a>
            </h5>
        </div>
        <!-- Navbar links -->
        <section class="mb-5">

            <!-- Card -->
            <div class="card card-cascade narrower">

                <section>

                    <div class="card">
                        <div id="pstable" class=" col-md-12 col-sm-auto card-body" runat="server"> <%--card-
body--%>

                            <!-- Table -->
                            <table class="table table-striped" runat="server">

                                <!-- Table head -->

                                    <thead class='mdb-color darken-3'>
                                        <tr class='text-white'>
                                            <th scope='row'>
                                                <input class='form-check-input' type='checkbox' id='checkboxall'>
<label for='checkboxall' class='label-table'></label>
                                            </th>
                                            <th><i class='far fa-image'></i></th>
                                            <th style='width:20%' >Name</th>
                                            <th>Stock</th>
                                            <th>Price</th>
                                            <th style='width:20%'>Categories</th>
                                            <th>Date</th>
                                            <th style='width:10%'>Action</th>

```

```

</tr>
</thead>
<!-- Table head -->

<!-- Table body -->
<tbody>
<tr>
<th scope="row">
  <input class="form-check-input" type="checkbox" id="checkbox2">
  <label for="checkbox2" class="label-table"></label>
</th>
<td></td>
<td><a href="#">Best Bridal Dresses</a></td>
<td class="text-success font-weight-bold">In stock</td>
<td><s>₹8099</s><br>₹7999</td>
<td><span class="badge badge-primary">Women Fashion</span> <span
class="badge badge-primary">Bridal</span></td>
<td>Published<br>2018/11/16</td>
</tr>
</tbody>
<!-- Table body -->
</table>
<!-- Table -->

<hr class="my-0">

<!-- Bottom Table UI -->
<div class="d-flex justify-content-between">

  <!-- Name -->
  <select class="mdb-select colorful-select dropdown-primary md-form hidden-md-
down">
    <option value="" disabled>Rows number</option>
    <option value="1" selected>6 rows</option>
    <option value="2">25 rows</option>
    <option value="3">50 rows</option>
    <option value="4">100 rows</option>
  </select>

  <!-- Pagination -->
  <nav class="my-4">
    <ul class="pagination pagination-circle pg-blue mb-0">

      <!--First-->
      <li class="page-item disabled clearfix d-none d-md-block"><a class="page-
link">First</a></li>

      <!-- Arrow left -->
      <li class="page-item disabled">
        <a class="page-link" aria-label="Previous">
          <span aria-hidden="true">&laquo;</span>
          <span class="sr-only">Previous</span>
        </a>
      </li>

```

```

        <!-- Numbers -->
        <li class="page-item active"><a class="page-link">1</a></li>
        <li class="page-item"><a class="page-link">2</a></li>
        <li class="page-item"><a class="page-link">3</a></li>
        <li class="page-item"><a class="page-link">4</a></li>
        <li class="page-item"><a class="page-link">5</a></li>

        <!-- Arrow right -->
        <li class="page-item">
            <a class="page-link" aria-label="Next">
                <span aria-hidden="true">&raquo;</span>
                <span class="sr-only">Next</span>
            </a>
        </li>

        <!-- First -->
        <li class="page-item clearfix d-none d-md-block"><a class="page-
link">Last</a></li>

        </ul>
    </nav>
    <!-- Pagination -->

</div>
<!-- Bottom Table UI -->

</div>
</div>

</section>

</div>
<!-- Card -->

</section>

</div>
</asp:Content>
<asp:Content ID="Content3" ContentPlaceHolderID="ContentPlaceHolderfooter"
Runat="Server">

</asp:Content>

***** VendorProductsUser.aspx.cs*****
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBEntity;
using DowBAL;
using System.Text;

public partial class VendorProductsUser : System.Web.UI.Page

```

```

{
    DDLogTracer objlog = new DDLogTracer();

    protected void Page_Load(object sender, EventArgs e)
    {
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr != null)
        {

            string msg = Request.QueryString["msg"];
            if (msg != null)
            {
                //Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
            }
            if (!IsPostBack)
            {
                Bindetails(Convert.ToInt16(Session["VendorCode"]));
            }
        }
        else
        {
            Response.Redirect("MLB_Login.aspx");
        }
    }
}

private void Bindetails(int userid)
{
    //try
    //{
    StringBuilder sb = new StringBuilder();
    sb.Append("<table class='table' runat='server'>\n");
    sb.Append("    <thead class='mdb-color darken-3'> \n");
    sb.Append("    <tr class='text-white'>      \n");
    sb.Append("    <th scope='row'>      \n");
    sb.Append("    <input class='form-check-input' type='checkbox' id='checkboxall'> \n");
    sb.Append("    <label for='checkboxall' class='label-table'></label>\n");
    sb.Append("    </th> \n");
    sb.Append("    <th><i class='far fa-image'></i></th>      \n");
    sb.Append("    <th style='width:20%'>Name</th>      \n");
    sb.Append("    <th>Stock</th>      \n");
    sb.Append("    <th>Price</th>      \n");
    sb.Append("    <th style='width:20%'>Categories</th>      \n");
    sb.Append("    <th>Date</th>      \n");
    sb.Append("    <th style='width:10%'>Action</th>      \n");
    sb.Append("    </tr> \n");
    sb.Append("    </thead>      \n");

    List<ProductProductSeller> Lst = (new
Products()).GetProductSellerbySellerId(userid,1);

    if (Lst != null && Lst.Count > 0)
    {
        sb.Append("    <tbody>      \n");
        foreach (var dt in Lst)

```

```

        {
            sb.Append("        <tr> \n");
            sb.Append("        <th scope='row'> \n");
            sb.Append("        <input class='form-check-input' type='checkbox'
id='checkbox"+dt.PSId+"> \n");
            sb.Append("        <label for='checkbox" + dt.PSId + "' class='label-
table'></label>\n");
            sb.Append("        </th> \n");
            sb.Append("        <td><img src='"+dt.ImagePath+ "' class='img-fluid'
style='height:100px;width:100px'></td> \n");
            sb.Append("        <td><a href='#'+dt.SellerDescription+</a></td> \n");
            if (dt.TStock > 0) {
                sb.Append("        <td class='text-success font-weight-bold'>In stock(" +
dt.TStock + ")</td> \n");
            }
            else
            {
                sb.Append("        <td class='text-danger font-weight-bold'>Out Of stock</td>
\n");
            }
            //<td class="text-danger font-weight-bold">Out Of stock</td>
            sb.Append("        <td><s>₹" + dt.Price + "</s><br>₹" + dt.Price + "</td>
\n");
            sb.Append("        <td><span class='badge badge-primary'>" + dt.CatName +
"</span> <span class='badge badge-primary'>" + dt.GroupName + "</span></span> <span
class='badge badge-primary'>" + dt.ProductName + "</span></td>\n");
            sb.Append("        <td>Published<br>" + Convert.ToString(
dt.CreatedOn).Substring(0,10)+"</td> \n");
            sb.Append("        <td><a href='/postproducts.aspx?psid="+dt.PSId
+"&pagetype=Edit'class='btn btn-outline-danger waves-effect btn-sm font-weight-bold'><i
class='fas fa-edit'></i> &nbsp;&nbsp;Edit</a></td>");
            sb.Append("        </tr> \n");
        }
        sb.Append("    </tbody> \n");
    }
    prtable.InnerHtml = sb.ToString();
    objlog.Trace_Log(1, "home", "1", "menu", "\n" + sb.ToString());
    // WDGrid_UserList.DataSource = Lst;
    // WDGrid_UserList.DataBind();
    // WDGrid_UserList.Visible = true;
    //}
    //else
    //{
    // WDGrid_UserList.Visible = false;
    //}
}
}
***** VendorServicesAdd.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/ProductMaster.master"
AutoEventWireup="true" CodeFile="VendorServicesAdd.aspx.cs"
Inherits="VendorServicesAdd" %>

```

```

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
Runat="Server">
</asp:Content>
<asp:Content ID="Content4" ContentPlaceHolderID="CPH_TopMessages" Runat="Server">
  <div class="col-lg-12 col-md-12 col-sm-12">
    <div class="row">
      <%--<h3>Ravikiran Techno Soft</h3>--%>
      <div id="dvMsgBox" class="span10 ordersave" runat="server" style="display:
block;">
<%--          <asp:Label ID="Label2" runat="server" Text=""
Height="100px"></asp:Label>--%>
          <asp:Label ID="Label3" runat="server" Text="" Height="60px"></asp:Label>
        </div>
      </div>
    </div>
    <div class="container-fluid">
      <div class="row-fluid" style="padding-top: 20px;">
        <div>
          <h5 class="my-4 dark-grey-text font-weight-bold" style="padding-left:20px">Manage
Selected Services
          <asp:Button ID="btn_save" runat="server" CssClass="btn btn-secondary btn-md"
Text="Save" OnClick="btn_save_Click" /> <%-- btn-rounded--%>
          <asp:Button ID="btn_clear" runat="server" CssClass ="btn btn-secondary btn-md btn
rounded" Text="Clear" OnClick="btn_clear_Click" />
<%--      <a href="#" class="btn btn-secondary btn-md btn-rounded"> &nbsp;Edit</a> <%--
<i class="fas fa-edit"></i>
          <a href="#" class="btn btn-secondary btn-md btn-rounded"> &nbsp;Delete</a><%--<i
class="fas fa-trash"></i>--%>
        </h5>
        <!-- Navbar links -->

          </div>
        <div class="span10">
          <div class="row">
            <div id="Div1" class="span12 ordersave" runat="server" style="display: none;">
              <asp:Label ID="Label1" runat="server" Text=""></asp:Label>
            </div>
          </div>
        </div>
      </div>
    </div>

    <%-- <div id="LeftSideBar" runat="server" class="span2">
      <asp:Table ID="Table1" runat="server">
        </asp:Table>
      </div>--%>
    </div>
  </asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="CPH_MainContent" Runat="Server" >
  <div class="container-fluid" style="padding-left:100px; padding-right:50px">

    <!-- Navbar links -->
    <%-- <section class="mb-5">

      <!-- Card -->
      <div class="card card-cascade narrower">

```



```

        <div class=" col-lg-10 col-md-10 col-sm-10 category-num">
        <table id="Table5" class="table table-striped table-detail col-lg-6"
runat="server">
        <tbody>
        <tr>
        <td class="span3" dir="auto" style="width:170px">Service Category
: </td>
        <td class="span3">
        <asp:DropDownList ID="ddl_productcat" runat="server"
OnSelectedIndexChanged="ddl_productcat_SelectedIndexChanged" Visible="true"
Width="200px">
        <asp:ListItem Value="0">Venues</asp:ListItem>
        <asp:ListItem Value="1" >Catering</asp:ListItem>
        </asp:DropDownList>
        <asp:Button ID="retgroup" runat="server" CssClass="btn btn-
secondary btn-md " OnClick="ddl_productcat_SelectedIndexChanged" Text="Retrieve Data"
/>
        </td>
        </tr>
        <tr>
        <td class="span3" dir="auto" style="width:200px">Product Sub-
Category : </td>
        <td class="span3">
        <asp:DropDownList ID="ddl_productgroup" runat="server"
Visible="true" Enabled="true"
OnSelectedIndexChanged="ddl_productgroup_SelectedIndexChanged" Width="200px">
        </asp:DropDownList>
        <asp:Button ID="btn_retrieve" runat="server"
CssClass="=btn btn-secondary btn-md btn rounded" Text="Retrive Data"
OnClick="btn_retrieve_Click" />
        </td>
        </tr>
        </tbody></table>
        </div>
        </section>--%>
<section class="mb-5">
        <!-- Card -->
        <div class="card card-cascade narrower">
        <section>
        <div class="card">

```

```

<div id="prtable" class="card-body" runat="server">

    <asp:Table ID="Dtable" runat="server" border="1" CssClass="table table-striped
table-detail col-lg-6 align-items-sm-center" >

</asp:Table>

<!-- Table -->
<!-- Table -->

<hr class="my-0"/>

    <!-- Bottom Table UI -->

</div>
</div>

</section>

    <!-- Card -->
</div>
</section>

</div>
</asp:Content>
<asp:Content ID="Content3" ContentPlaceHolderID="ContentPlaceHolder1"
Runat="Server">

</asp:Content>

*****VendorServicesAdd.aspx.cs *****
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBEntity;
using DowBAL;
using System.Xml;
using System.Xml.Serialization;
using System.IO;
using System.Text;
using System.Data.SqlClient;
using System.Configuration;
using System.Data;

```

```

using System.Web.Security;

public partial class VendorServicesAdd : System.Web.UI.Page
{
    string PageType;
    int Id;
    DDLogTracer objlog = new DDLogTracer();

    protected void Page_Load(object sender, EventArgs e)
    {
        Id = Convert.ToInt16(Session["Pgid"].ToString());
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr != null)
        {

            string msg = Request.QueryString["msg"];
            if (msg != null)
            {
                Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
            }
            Bindetails();

            //if (Convert.ToInt16(Clsusr.GroupId) > 3)
            //{
                PopDlls ObjPD = new PopDlls();
                // ObjPD.BindDropDowns(ddl_productcat, (new
                PopDlls()).GetProductCatbyType(2));
                // ObjPD.BindDropDowns(ddl_productgroup, (new
                PopDlls()).GetproductGroupbyServices(2));
                //}
            }
            else
            {
                Response.Redirect("MLB_Login.aspx");
            }
        }
        private void Bindetails()
        {

            //Session["Pgid"] = grid.ToString();
            ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
            List<Eteproduct> Lst = new List<Eteproduct>();
            List<Eteproduct> cnts = new Products().GetSellerProductServices(
            Convert.ToInt16(Clsusr.UserId));
            Lst = new Products().GetProductServices( "");
            CheckBox chk = new CheckBox();
            Dtable.Rows.Clear();
            //try
            // {
            if (Lst != null && Lst.Count > 0)
            {
                TableRow tr = new TableRow();
                TableCell td = new TableCell();
                TableHeaderRow thr = new TableHeaderRow();

```

```

td = new TableCell();
chk.ID = "CheckBoxAll";
td.Controls.Add(chk);
thr.Cells.Add(td);
td = new TableCell();
td.Text = "Image";
thr.Cells.Add(td);
td = new TableCell();
td = new TableCell();
td.Text = "Category Name";
thr.Cells.Add(td);
td = new TableCell();
td.Text = "Product Name";
thr.Cells.Add(td);
td = new TableCell();
td.Text = "Selected Date";
thr.Cells.Add(td);

Dtable.Rows.Add(thr);
tr = new TableRow();
td = new TableCell();

StringBuilder sb = new StringBuilder();

DateTime dts;

foreach (var dt in Lst)
{
    int pcount = (from a in cnts where a.ProductId == dt.ProductId select a).Count();
    if (pcount > 0)
    { dts = (from b in cnts where b.ProductId == dt.ProductId select
b.CreatedOn).FirstOrDefault(); }
    else { dts = DateTime.Today.Date; }
    tr = new TableRow();
    td = new TableCell();
    chk = new CheckBox();
    chk.ID="CheckBox" + dt.ProductId ;
    chk.Checked = true;
    if (pcount > 0)
    {
        chk.Checked = true;
    }
    else
    {
        chk.Checked = false;
    }
    td.Controls.Add(chk);
    tr.Cells.Add(td);
    td = new TableCell();
    string img = "<img src=\"" + dt.imageurl + "\" runat='server'
style='height:100px;width:100px' /> ";
    td.Text = img;
    tr.Cells.Add(td);
    td = new TableCell();

```

```

        td.Text = "<span class='badge badge-primary'>" + dt.CatName + "</span>";
    }; //<span class='badge badge-primary'>" + dt.GroupName + "</span>";
    tr.Cells.Add(td);
    td = new TableCell();
    td.Text = "<span class='badge badge-primary'>" + dt.ProductName + "</span>";
    tr.Cells.Add(td);
    td = new TableCell();
    td.Text = Convert.ToString(dts).Substring(0, 10);
    tr.Cells.Add(td);
    Dtable.Rows.Add(tr);
}
}
}
private void Bindetails1(int grid)
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    List<Eteproduct> Lst = new List<Eteproduct>();
    List<Eteproduct> cnts = new Products().GetSellerProductsbygrid(grid, Convert.ToInt16(
    Clsusr.UserId));
    Lst = new Products().GetProductbyGroup(grid, "");
    //try    //{
    StringBuilder sb = new StringBuilder();
    sb.Append("<table class='table' runat='server'>\n");
    sb.Append("    <thead class='mdb-color darken-3'>\n");
    sb.Append("        <tr class='text-white'>        \n");
    sb.Append("            <th scope='row'>        \n");
    sb.Append("            <asp:CheckBox ID='CheckBoxAll' runat='server' Checked='false' />
        \n");
    sb.Append("            <label for='checkbox1' class='label-table'></label> \n");
    sb.Append("        </th> \n");
    sb.Append("        <th><i class='far fa-image'></i></th>        \n");
    sb.Append("        <th style='width:20%'>Name</th>        \n");
    //sb.Append("        <th>Stock</th>        \n");
    //sb.Append("        <th>Price</th>        \n");
    sb.Append("        <th style='width:30%'>Categories</th>        \n");
    sb.Append("        <th>Date</th>        \n");
    sb.Append("    </tr> \n");
    sb.Append("    </thead> \n");
    sb.Append("    <tbody>        \n");

    DateTime dts;
    if (Lst != null && Lst.Count > 0)
    {
        foreach (var dt in Lst)
        {
            int pcount = (from a in cnts where a.ProductId == dt.ProductId select a).Count();
            if (pcount > 0)
            {
                dts = (from b in cnts where b.ProductId == dt.ProductId select
                b.CreatedOn).FirstOrDefault();
            }
            else { dts = DateTime.Today.Date; }
            sb.Append("        <tr> \n");
            sb.Append("            <td scope='row'>        \n");
            if (pcount > 0)

```

```

        {
            sb.Append("      <asp:CheckBox ID='CheckBox'+dt.ProductId+'\" runat='server'
Checked='true' />      \n");
        }
        else
        {
            sb.Append("      <asp:CheckBox ID='CheckBox\" + dt.ProductId + \"
runat='server' Checked='false' />      \n");
        }

        //sb.Append("      <label for='checkbox2' class='label-table'></label> \n");
        sb.Append("      </td> \n");
        sb.Append("      <td><img src=\""+dt.imageUrl+ "\" class='img-fluid'
style='height:100px;width:100px'/></td>      \n");
        sb.Append("      <td><a href='#\">"+dt.ProductName+"</a></td>      \n");
        //sb.Append("      <td><s>₹\" + dt.Price + "</s><br>₹\" + dt.Price + "</td>
\n");
        sb.Append("      <td><span class='badge badge-primary'>\" + dt.CatName +
\"</span> <span class='badge badge-primary'>\" + dt.GroupName + "</span> </td>
\n");
        sb.Append("      <td>Selected<br>\"+Convert.ToString(
dts).Substring(0,10)+\"</td> \n");
        sb.Append("      </tr> \n");
    }
    sb.Append(" </tbody></table>      \n");

    prtable.InnerHtml = sb.ToString();
    objlog.Trace_Log(1, "home", "1", "menu", "\n" + sb.ToString());
}
}
protected void btn_save_Click(object sender, EventArgs e)
{
    bool ret = false;
    int grid = Convert.ToInt16(Session["Pgid"].ToString());
    string user = "";
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    if (Clsusr == null)
    {
        HttpCookie myCookie = Request.Cookies["DateOrWedCookie"];
        if (myCookie != null)
        {
            user = myCookie.Value.ToString();
        }
        else
        {
            user = "Guest";
        }
        guest_login(user);
    }
}
//try
{
    CheckBox cbox = new CheckBox();

```

```

cbox = (CheckBox)Dtable.FindControl("CheckBoxAll");
List<Eteproduct> Lst = new Products().GetProductServices( "");
if (cbox.Checked)
{
    foreach (var list in Lst)
    {
        ret= update(list.ProductId,true);
    }
}
else
{
    foreach (var list in Lst)
    {
        cbox = (CheckBox)Dtable.FindControl("CheckBox" + list.ProductId);
        if (cbox.Checked)
        {
            ret= update(list.ProductId,true);
        }
        else
        {
            ret = update(list.ProductId, false); }
    }
}
Response.Redirect("~/VendorHome.aspx?pageid=1&msg=Product Updated
Sucessfully");
}

//catch (Exception exc)
//{{
// //objlog.Trace_Error(4, "DateOrWed", "MangeDealer", "Load", "Exception:" +
exc.ToString());
// Common.ShowMessageHtml(exc.ToString(), 0, true, dvMsgBox);

//}}

}
protected bool update(decimal prid,bool chk)
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedIn_Specs"];

    bool ret = new Products().UpdSellerProduct(prid,Convert.ToInt16( Clsusr.UserId), chk);
    return ret;
}
protected void guest_login(string user)
{

    string City = "Hyderabad";// _City;
    string Country = "India";// _Country;

    BALCreateUser objlogin = new BALCreateUser();
    DataTable status_login = objlogin.LoginGuest(user, user,
Request.ServerVariables["REMOTE_ADDR"].ToString(), "", City, Country);
    string login_status = status_login.Rows[0]["UserStatus"].ToString();

```

```

ClsLoggedInUser objLoggedInUser;
if (login_status == "True")
{
    objLoggedInUser = new ClsLoggedInUser();

    objLoggedInUser.UserId = status_login.Rows[0]["UserId"].ToString();
    objLoggedInUser.UserName = status_login.Rows[0]["UserName"].ToString();
    objLoggedInUser.GroupId = status_login.Rows[0]["GroupId"].ToString();
    objLoggedInUser.LoginStatus = "True";
    objLoggedInUser.LoginID = Convert.ToInt32(status_login.Rows[0]["LoginId"]);
    objLoggedInUser.DealerId = Convert.ToInt32(status_login.Rows[0]["DealerId"]);
    objLoggedInUser.RoleId = Convert.ToInt32(status_login.Rows[0]["RoleId"]);
    objLoggedInUser.BrandId = 1; //Convert.ToInt32(status_login.Rows[0]["BrandId"]);

    Session["LoggedInUser_Specs"] = objLoggedInUser;

    FormsAuthentication.SetAuthCookie(objLoggedInUser.UserName, false);
    HttpCookie myCookie = new HttpCookie("DateOrWedCookie");
    DateTime now = DateTime.Now;
    myCookie.Value = objLoggedInUser.UserName;
    myCookie.Expires = now.AddYears(1);
    Response.Cookies.Add(myCookie);

}
}

protected void btn_retrieve_Click(object sender, EventArgs e)
{
    Binddetails();
}

protected void btn_clear_Click(object sender, EventArgs e)
{
}

protected void ddl_productcat_SelectedIndexChanged(object sender, EventArgs e)
{
    //int prcat =Convert.ToInt16( ddl_productcat.SelectedIndex);
    //if (prcat == 0) { prcat = 1; }
    //PopDlIs ObjPD = new PopDlIs();
    //ObjPD.BindDropDowns(ddl_productgroup, (new
PopDlIs()).GetproductGroupbyServices(2));

    //ddl_productgroup.Enabled = true;
}

protected void ddl_productgroup_SelectedIndexChanged(object sender, EventArgs e)
{
    //Session["Pgid"] = ddl_productgroup.SelectedValue;
    Binddetails();
}
}
}

```



```

***** VendorServices.aspx.cs*****
<%@ Page Title="" Language="C#" MasterPageFile="~/VendorMasterPage.master"
AutoEventWireup="true" CodeFile="VendorServices.aspx.cs" Inherits="VendorServices" %>
<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
Runat="Server">
    <div>
        <asp:Label ID="Label13" runat="server" Text="" Height="20px"></asp:Label>

    </div>
</asp:Content>
<asp:Content ID="Content4" ContentPlaceHolderID="CPH_TopMessages" Runat="Server">
    <div class="span10">
        <div class="row">
            <%--<h3>Ravikiran Techno Soft</h3>--%>
            <div id="dvMsgBox" class="span10 ordersave" runat="server" style="display:
block;">
                <asp:Label ID="Label2" runat="server" Text="" Height="50px"></asp:Label>
                <asp:Label ID="Label1" runat="server" Text="" Height="60px"></asp:Label>
            </div>
        </div>
    </div>
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentMain" Runat="Server" >
    <div class="container-fluid" style="padding-left:100px; padding-right:50px">

        <div><%--VendorProductsAdd.aspx--%>
            <h5 class="my-4 dark-grey-text font-weight-bold">Products
                <a href="PostServices.aspx" class="btn btn-outline-danger waves-effect btn-sm font-
weight-bold"><i class="far fa-plus-square"></i> &nbsp;Add New</a>
            <%--
                <a href="#" class="btn btn-outline-danger waves-effect btn-sm font-weight-
bold"><i class="fas fa-edit"></i> &nbsp;Edit</a> --%>
                <a href="#" class="btn btn-outline-danger waves-effect btn-sm font-weight-bold"><i
class="fas fa-edit"></i> &nbsp;Edit</a>
                <a href="#" class="btn btn-outline-danger waves-effect btn-sm font-weight-bold"><i
class="fas fa-trash-alt"></i> &nbsp;Delete</a>
            </h5>
        </div>
        <!-- Navbar links -->
        <section class="mb-5">

            <!-- Card -->
            <div class="card card-cascade narrower">

                <section>

                    <div class="card">
                        <div id="prtable" class="col-md-12 col-sm-auto card-body" runat="server">

                            <!-- Table -->
                            <table class="table table-striped" runat="server">

                                <!-- Table head -->
                                <thead class="mdb-color darken-3">

```

```

<tr class="text-white">
  <th scope="row">
    <input class="form-check-input" type="checkbox" id="checkbox1">
    <label for="checkbox1" class="label-table"></label>
  </th>
  <th><i class="far fa-image"></i></th>
  <th style="width:20%">Name</th>
  <th>Stock</th>
  <th>Price</th>
  <th style="width:20%">Categories</th>
  <th>Date</th>
</tr>
</thead>
<!-- Table head -->

<!-- Table body -->
<tbody>
<tr>
  <th scope="row">
    <input class="form-check-input" type="checkbox" id="checkbox2">
    <label for="checkbox2" class="label-table"></label>
  </th>
  <td></td>
  <td><a href="#">Best Bridal Dresses</a></td>
  <td class="text-success font-weight-bold">In stock</td>
  <td><s>₹8099</s><br>₹7999</td>
  <td><span class="badge badge-primary">Women Fashion</span> <span
class="badge badge-primary">Bridal</span></td>
  <td>Published<br>2018/11/16</td>
</tr>
<tr>
  <th scope="row">
    <input class="form-check-input" type="checkbox" id="checkbox3">
    <label for="checkbox3" class="label-table"></label>
  </th>
  <td></td>
  <td><a href="#">Best Groom Dresses</a></td>
  <td class="text-success font-weight-bold">In stock</td>
  <td><s>₹9099</s><br>₹8999</td>
  <td><span class="badge badge-primary">Men Fashion</span> <span
class="badge badge-primary">Groom</span></td>
  <td>Published<br>2018/11/16</td>
</tr>
<tr>
  <th scope="row">
    <input class="form-check-input" type="checkbox" id="checkbox4">
    <label for="checkbox4" class="label-table"></label>
  </th>
  <td></td>
  <td><a href="#">Best Bridegroom Dresses</a></td>
  <td class="text-danger font-weight-bold">Out Of stock</td>
  <td><s>₹10199</s><br>₹9999</td>

```

```

        <td><span class="badge badge-primary">Women Fashion</span> <span
class="badge badge-primary">Men Fashion</span> <span class="badge badge-
primary">Bridegroom</span></td>
        <td>Published<br>2018/11/16</td>
</tr>
<tr>
<th scope="row">
    <input class="form-check-input" type="checkbox" id="checkbox5">
    <label for="checkbox5" class="label-table"></label>
</th>
<td></td>
<td><a href="#">Best Bridal Dresses</a></td>
<td class="text-success font-weight-bold">In stock</td>
<td><s>₹ 8099</s><br>₹7999</td>
<td><span class="badge badge-primary">Women Fashion</span> <span
class="badge badge-primary">Bridal</span></td>
<td>Published<br>2018/11/16</td>
</tr>
<tr>
<th scope="row">
    <input class="form-check-input" type="checkbox" id="checkbox6">
    <label for="checkbox6" class="label-table"></label>
</th>
<td></td>
<td><a href="#">Best Groom Dresses</a></td>
<td class="text-success font-weight-bold">In stock</td>
<td><s>₹9099</s><br>₹8999</td>
<td><span class="badge badge-primary">Men Fashion</span> <span
class="badge badge-primary">Groom</span></td>
<td>Published<br>2018/11/16</td>
</tr>
<tr>
<th scope="row">
    <input class="form-check-input" type="checkbox" id="checkbox7">
    <label for="checkbox7" class="label-table"></label>
</th>
<td></td>
<td><a href="#">Best Bridegroom Dresses</a></td>
<td class="text-danger font-weight-bold">Out Of stock</td>
<td><s>₹10199</s><br>₹9999</td>
<td><span class="badge badge-primary">Women Fashion</span> <span
class="badge badge-primary">Men Fashion</span> <span class="badge badge-
primary">Bridegroom</span></td>
<td>Published<br>2018/11/16</td>
</tr>
</tbody>
<!-- Table body -->
</table>
<!-- Table -->

<hr class="my-0">

<!-- Bottom Table UI -->
<div class="d-flex justify-content-between">

```

```

<!-- Name -->
down">
<select class="mdb-select colorful-select dropdown-primary md-form hidden-md-
  <option value="" disabled>Rows number</option>
  <option value="1" selected>6 rows</option>
  <option value="2">25 rows</option>
  <option value="3">50 rows</option>
  <option value="4">100 rows</option>
</select>

<!-- Pagination -->
<nav class="my-4">
  <ul class="pagination pagination-circle pg-blue mb-0">

    <!--First-->
    <li class="page-item disabled clearfix d-none d-md-block"><a class="page-
link">First</a></li>

    <!-- Arrow left -->
    <li class="page-item disabled">
      <a class="page-link" aria-label="Previous">
        <span aria-hidden="true">&laquo;</span>
        <span class="sr-only">Previous</span>
      </a>
    </li>

    <!-- Numbers -->
    <li class="page-item active"><a class="page-link">1</a></li>
    <li class="page-item"><a class="page-link">2</a></li>
    <li class="page-item"><a class="page-link">3</a></li>
    <li class="page-item"><a class="page-link">4</a></li>
    <li class="page-item"><a class="page-link">5</a></li>

    <!-- Arrow right -->
    <li class="page-item">
      <a class="page-link" aria-label="Next">
        <span aria-hidden="true">&raquo;</span>
        <span class="sr-only">Next</span>
      </a>
    </li>

    <!-- First -->
    <li class="page-item clearfix d-none d-md-block"><a class="page-
link">Last</a></li>

  </ul>
</nav>
<!-- Pagination -->

</div>
<!-- Bottom Table UI -->

</div>
</div>

```

```

</section>

</div>
<!-- Card -->

</section>

</div>
</asp:Content>
<asp:Content ID="Content3" ContentPlaceHolderID="ContentPlaceHolderfooter"
Runat="Server">

</asp:Content>

*****VendorServices.aspx.cs *****
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBEntity;
using DowBAL;
using System.Text;

public partial class VendorServices : System.Web.UI.Page
{
    DDLogTracer objlog = new DDLogTracer();

    protected void Page_Load(object sender, EventArgs e)
    {
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr != null)
        {
            string pageid = "6";// Request.QueryString["pageid"];
            //ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
Convert.ToInt32(Clsusr.GroupId));
            string msg = Request.QueryString["msg"];
            if (msg != null)
            {
                //Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
            }
            if (Convert.ToInt16(Clsusr.GroupId) > 3)
            {
                Bindetails(Convert.ToInt16(Clsusr.UserId));
            }
        }
        else
        {
            Response.Redirect("MLB_Login.aspx");
        }
    }
    private void Bindetails(int userid)
    {

```

```

//try
//{
StringBuilder sb = new StringBuilder();
prtable.InnerHtml = "";

sb.Append("<table class='table' runat='server'>\n");
sb.Append("    <thead class='mdb-color darken-3'> \n");
sb.Append("    <tr class='text-white'> \n");
sb.Append("    <th scope='row'> \n");
sb.Append("    <input class='form-check-input' type='checkbox' id='checkboxall'>
    \n");
sb.Append("    <label for='checkboxall' class='label-table'></label>\n");
sb.Append("    </th> \n");
sb.Append("    <th><i class='far fa-image'></i></th> \n");
sb.Append("    <th style='width:20%'>Name</th> \n");
sb.Append("    <th>Availability</th> \n");
sb.Append("    <th>Price</th> \n");
sb.Append("    <th style='width:30%'>Categories</th> \n");
sb.Append("    <th>Date</th> \n");
sb.Append("    <th style='width:10%'>Action</th> \n");
sb.Append("    </tr> \n");
sb.Append("    </thead> \n");
sb.Append("    <tbody> \n");

```

```

List<ProductProductSeller> Lst = (new
Products()).GetProductSellerbySellerId(userid,2);

```

```

if (Lst != null && Lst.Count > 0)
{
    foreach (var dt in Lst)
    {
        sb.Append("    <tr> \n");
        sb.Append("    <th scope='row'> \n");
        sb.Append("    <input class='form-check-input' type='checkbox'
id='checkbox'+dt.PSId+'> \n");
        sb.Append("    <label for='checkbox' + dt.PSId + "" class='label-
table'></label>\n");
        sb.Append("    </th> \n");
        sb.Append("    <td><img src='"+dt.ImagePath+ "" class='img-fluid'
style='height:100px;width:100px'></td> \n");
        sb.Append("    <td><a href='#'+dt.SellerDescription+'</a></td> \n");
        if (dt.TStock > 0) {
            sb.Append("    <td class='text-success font-weight-bold'>Check Availability("
+ dt.TStock + "</td> \n");
        }
        else
        {
            sb.Append("    <td class='text-danger font-weight-bold'>Check
Availability</td> \n");
        }
        //<td class="text-danger font-weight-bold">Out Of stock</td>
        sb.Append("    <td><s>₹" + dt.Price + "</s><br>₹" + dt.Price + "</td>
\n");
    }
}

```

```

        sb.Append("        <td><span class='badge badge-primary'>" + dt.GroupName +
"/span></span> <span class='badge badge-primary'>" + dt.ProductName + "</span></td>
\n");
        sb.Append("        <td>Published<br>" + Convert.ToString(
dt.CreatedOn).Substring(0,10)+"</td> \n");
        sb.Append("        <td><a href='/postservices.aspx?psid=" + dt.PSId +
"&pagetype=Edit' class='btn btn-outline-danger waves-effect btn-sm font-weight-bold'><i
class='fas fa-edit'></i> &nbsp;Edit</a></td>");
        sb.Append("        </tr> \n");

    }
    sb.Append(" </tbody> \n");
    prtable.InnerHtml = sb.ToString();
    objlog.Trace_Log(1, "home", "1", "menu", "\n" + sb.ToString());

}
// WDGrid_UserList.DataSource = Lst;
// WDGrid_UserList.DataBind();
// WDGrid_UserList.Visible = true;
//}
else
{
    prtable.InnerHtml = "No Data found";
    Common.ShowMessageHtml("No Data found", 1, true, dvMsgBox);
}
}
}
}
}
}
***** VendorServicesAddUser.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/ProductMaster.master"
AutoEventWireup="true" CodeFile="VendorServicesAddUser.aspx.cs"
Inherits="VendorServicesAddUser" %>
<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
Runat="Server">
</asp:Content>
<asp:Content ID="Content4" ContentPlaceHolderID="CPH_TopMessages" Runat="Server">
<div class="col-lg-12 col-md-12 col-sm-12">
<div class="row">
<%--<h3>Ravikiran Techno Soft</h3>--%>
<div id="dvMsgBox" class="span10 ordersave" runat="server" style="display:
block;">
<%--
<asp:Label ID="Label2" runat="server" Text=""
Height="100px"></asp:Label>--%>
<asp:Label ID="Label3" runat="server" Text="" Height="60px"></asp:Label>
</div>
</div>
</div>
<div class="container-fluid">
<div class="row-fluid" style="padding-top: 20px;">
<div>
<h5 class="my-4 dark-grey-text font-weight-bold" style="padding-left:20px">Manage
Selected Services
<asp:Button ID="btn_save" runat="server" CssClass="btn btn-secondary btn-md"
Text="Save" OnClick="btn_save_Click" /> <%-- btn-rounded--%>

```



```
                <asp:Button ID="retgroup" runat="server" CssClass="btn btn-
secondary btn-md " OnClick="ddl_productcat_SelectedIndexChanged" Text="Retrieve Data"
/>
```

```
            </td>
```

```
        </tr>
```

```
        <tr>
```

```
            <td class="span3" dir="auto" style="width:200px">Product Sub-
Category : </td>
```

```
            <td class="span3">
```

```
                <asp:DropDownList ID="ddl_productgroup" runat="server"
Visible="true" Enabled="true"
OnSelectedIndexChanged="ddl_productgroup_SelectedIndexChanged" Width="200px">
```

```
                </asp:DropDownList>
```

```
            </td>
```

```
            <td>                <asp:Button ID="btn_retrieve" runat="server"
CssClass="=btn btn-secondary btn-md btn rounded" Text="Retrive Data"
OnClick="btn_retrieve_Click" />
```

```
        </td>
```

```
    </tr>
```

```
    </tbody></table>
```

```
    </div>
```

```
</div>
```

```
</section>--%>
```

```
<section class="mb-5">
```

```
    <!-- Card -->
```

```
    <div class="card card-cascade narrower">
```

```
        <section>
```

```
            <div class="card">
```

```
                <div id="prtable" class="card-body" runat="server">
```

```
                <asp:Table ID="Dtable" runat="server" border="1" CssClass="table table-striped
table-detail col-lg-6 align-items-sm-center" >
```

```
    </asp:Table>
```

```
<!-- Table -->
```

```
<!-- Table -->
```

```
<hr class="my-0"/>
```

```
<!-- Bottom Table UI -->
```

```
</div>
```

```
</div>
```

```
</section>
```

```
<!-- Card -->
```

```
</div>
```

```
</section>
```

```
</div>
```

```
</asp:Content>
```

```
<asp:Content ID="Content3" ContentPlaceHolderID="ContentPlaceHolder1"  
Runat="Server">
```

```
</asp:Content>
```

```
*****VendorServicesAddUser.aspx.cs *****
```

```
using System;  
using System.Collections.Generic;  
using System.Linq;  
using System.Web;  
using System.Web.UI;  
using System.Web.UI.WebControls;  
using DowBEntity;  
using DowBAL;  
using System.Xml;  
using System.Xml.Serialization;  
using System.IO;  
using System.Text;  
using System.Data.SqlClient;  
using System.Configuration;  
using System.Data;  
using System.Web.Security;
```

```
public partial class VendorServicesAddUser : System.Web.UI.Page
```

```
{
```

```
    string PageType;
```

```
    int Id;
```

```
    DDLogTracer objlog = new DDLogTracer();
```

```
protected void Page_Load(object sender, EventArgs e)
```

```
{
```

```
    Id = Convert.ToInt16(Session["Pgid"].ToString());
```

```
    CIsLoggedInUser CIsusr = (CIsLoggedInUser)Session["LoggedInUser_Specs"];  
    if (CIsusr != null)
```

```
    {
```

```
    {
```

```
        string msg = Request.QueryString["msg"];
```

```
        if (msg != null)
```

```
        {
```

```
            Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
```

```
        }
```

```
    }
```

```
}
```

```

Bindetails();

//if (Convert.ToInt16(Clsusr.GroupId) > 3)
//{
//    PopDtls ObjPD = new PopDtls();
//    ObjPD.BindDropDowns(ddl_productcat, (new
PopDtls()).GetProductCatbyType(2));
//    ObjPD.BindDropDowns(ddl_productgroup, (new
PopDtls()).GetproductGroupbyServices(2));
//}
}
else
{
    Response.Redirect("MLB_Login.aspx");
}
}
private void Bindetails()
{

//Session["Pgid"] = grid.ToString();
ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
List<Eteproduct> Lst = new List<Eteproduct>();
List<Eteproduct> cnts = new Products().GetSellerProductServices(
Convert.ToInt16(Clsusr.UserId));
Lst = new Products().GetProductServices( "");
CheckBox chk = new CheckBox();
Dtable.Rows.Clear();
//try
// {
if (Lst != null && Lst.Count > 0)
{
    TableRow tr = new TableRow();
    TableCell td = new TableCell();
    TableHeaderRow thr = new TableHeaderRow();
    td = new TableCell();
    chk.ID = "CheckBoxAll";
    td.Controls.Add(chk);
    thr.Cells.Add(td);
    td = new TableCell();
    td.Text = "Image";
    thr.Cells.Add(td);
    td = new TableCell();
    td = new TableCell();
    td.Text = "Category Name";
    thr.Cells.Add(td);
    td = new TableCell();
    td.Text = "Product Name";
    thr.Cells.Add(td);
    td = new TableCell();
    td.Text = "Selected Date";
    thr.Cells.Add(td);

    Dtable.Rows.Add(thr);
    tr = new TableRow();
    td = new TableCell();

```

```

StringBuilder sb = new StringBuilder();

DateTime dts;

    foreach (var dt in Lst)
    {
        int pcount = (from a in cnts where a.ProductId == dt.ProductId select a).Count();
        if (pcount > 0)
            { dts = (from b in cnts where b.ProductId == dt.ProductId select
b.CreatedOn).FirstOrDefault(); }
        else { dts = DateTime.Today.Date; }
        tr = new TableRow();
        td = new TableCell();
        chk = new CheckBox();
        chk.ID="CheckBox" + dt.ProductId ;
        chk.Checked = true;
        if (pcount > 0)
            {
                chk.Checked = true;
            }
        else
            {
                chk.Checked = false;
            }
        td.Controls.Add(chk);
        tr.Cells.Add(td);
        td = new TableCell();
        string img = "<img src=\"" + dt.imageurl + "\" runat='server'
style='height:100px;width:100px' /> ";
        td.Text = img;
        tr.Cells.Add(td);
        td = new TableCell();
        td.Text = "<span class='badge badge-primary'>" + dt.CatName + "</span>
";//<span class='badge badge-primary'>" + dt.GroupName + "</span>
        tr.Cells.Add(td);
        td = new TableCell();
        td.Text = "<span class='badge badge-primary'>" + dt.ProductName + "</span>";
        tr.Cells.Add(td);
        td = new TableCell();
        td.Text = Convert.ToString(dts).Substring(0, 10);
        tr.Cells.Add(td);
        Dtable.Rows.Add(tr);

    }
}

private void Bindetails1(int grid)
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    List<Eteproduct> Lst =new List<Eteproduct>();
    List<Eteproduct> cnts=new Products().GetSellerProductsbygrid(grid,Convert.ToInt16(
Clsusr.UserId));
    Lst=new Products().GetProductbyGroup(grid,"");
}

```

```

//try    //{
    StringBuilder sb = new StringBuilder();
    sb.Append("<table class='table' runat='server'>\n");
    sb.Append("    <thead class='mdb-color darken-3'> \n");
    sb.Append("    <tr class='text-white'> \n");
    sb.Append("    <th scope='row'> \n");
    sb.Append("    <asp:CheckBox ID='CheckBoxAll' runat='server' Checked='false' /> \n");
    sb.Append("    <label for='checkbox1' class='label-table'></label> \n");
    sb.Append("    </th> \n");
    sb.Append("    <th><i class='far fa-image'></i></th> \n");
    sb.Append("    <th style='width:20%'>Name</th> \n");
    //sb.Append("    <th>Stock</th> \n");
    //sb.Append("    <th>Price</th> \n");
    sb.Append("    <th style='width:30%'>Categories</th> \n");
    sb.Append("    <th>Date</th> \n");
    sb.Append("    </tr> \n");
    sb.Append("    </thead> \n");
    sb.Append("    <tbody> \n");

    DateTime dts;
    if (Lst != null && Lst.Count > 0)
    {
        foreach (var dt in Lst)
        {
            int pcount=(from a in cnts where a.ProductId==dt.ProductId select a).Count();
            if (pcount > 0)
            { dts = (from b in cnts where b.ProductId == dt.ProductId select
b.CreatedOn).FirstOrDefault(); }
            else { dts = DateTime.Today.Date; }
            sb.Append("    <tr> \n");
            sb.Append("    <td scope='row'> \n");
            if(pcount>0)
            {
                sb.Append("    <asp:CheckBox ID='CheckBox"+dt.ProductId+"' runat='server'
Checked='true' /> \n");
            }
            else
            {
                sb.Append("    <asp:CheckBox ID='CheckBox" + dt.ProductId + "'
runat='server' Checked='false' /> \n");
            }

            //sb.Append("    <label for='checkbox2' class='label-table'></label> \n");
            sb.Append("    </td> \n");
            sb.Append("    <td><img src='"+dt.imageUrl+ "' class='img-fluid'
style='height:100px;width:100px'/></td> \n");
            sb.Append("    <td><a href='#'>" + dt.ProductName + "</a></td> \n");
            //sb.Append("    <td><s>₹" + dt.Price + "</s><br>₹" + dt.Price + "</td> \n");
            sb.Append("    <td><span class='badge badge-primary'>" + dt.CatName +
"</span> <span class='badge badge-primary'>" + dt.GroupName + "</span> </td> \n");

```

```

        sb.Append("        <td>Selected<br>" + Convert.ToString(
dts).Substring(0,10)+"</td> \n");
        sb.Append("        </tr> \n");

    }
    sb.Append("    </tbody></table> \n");

    prttable.InnerHtml = sb.ToString();
    objlog.Trace_Log(1, "home", "1", "menu", "\n" + sb.ToString());
}
}
protected void btn_save_Click(object sender, EventArgs e)
{
    bool ret = false;
    int grid = Convert.ToInt16(Session["Pgid"].ToString());
    string user = "";
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    if (Clsusr == null)
    {
        HttpCookie myCookie = Request.Cookies["DateOrWedCookie"];
        if (myCookie != null)
        {
            user = myCookie.Value.ToString();
        }
        else
        {
            user = "Guest";
        }
        guest_login(user);
    }
}
//try
{
    CheckBox cbox = new CheckBox();
    cbox = (CheckBox)Dtable.FindControl("CheckBoxAll");
    List<Eteproduct> Lst = new Products().GetProductServices("");
    if (cbox.Checked)
    {
        foreach (var list in Lst)
        {
            ret= update(list.ProductId,true);
        }
    }
    else
    {
        foreach (var list in Lst)
        {
            cbox = (CheckBox)Dtable.FindControl("CheckBox" + list.ProductId);
            if (cbox.Checked)
            {
                ret= update(list.ProductId,true);
            }
            else
            {
                ret = update(list.ProductId, false); }
        }
    }
}
}

```

```

    }
    }
    Response.Redirect("~/VendorHome.aspx?pageid=1&msg=Product Updated
Sucessfully");
}

//catch (Exception exc)
//{{
// //objlog.Trace_Error(4, "DateOrWed", "MangeDealer", "Load", "Exception:" +
exc.ToString());
// Common.ShowMessageHtml(exc.ToString(), 0, true, dvMsgBox);

//}}

}
protected bool update(decimal prid,bool chk)
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];

    bool ret = new Products().UpdSellerProduct(prid,
Convert.ToInt16(Session["VendorCode"]), chk);
    return ret;
}
protected void guest_login(string user)
{
    string City = "Hyderabad";// _City;
    string Country = "India";// _Country;

    BALCreateUser objlogin = new BALCreateUser();
    DataTable status_login = objlogin.LoginGuest(user, user,
Request.ServerVariables["REMOTE_ADDR"].ToString(), "", City, Country);
    string login_status = status_login.Rows[0]["UserStatus"].ToString();

    ClsLoggedInUser objLogedInUser;
    if (login_status == "True")
    {
        objLogedInUser = new ClsLoggedInUser();

        objLogedInUser.UserId = status_login.Rows[0]["UserId"].ToString();
        objLogedInUser.UserName = status_login.Rows[0]["UserName"].ToString();
        objLogedInUser.GroupId = status_login.Rows[0]["GroupId"].ToString();
        objLogedInUser.LoginStatus = "True";
        objLogedInUser.LoginID = Convert.ToInt32(status_login.Rows[0]["LoginId"]);
        objLogedInUser.DealerId = Convert.ToInt32(status_login.Rows[0]["DealerId"]);
        objLogedInUser.RoleId = Convert.ToInt32(status_login.Rows[0]["RoleId"]);
        objLogedInUser.BrandId = 1; //Convert.ToInt32(status_login.Rows[0]["BrandId"]);

        Session["LoggedInUser_Specs"] = objLogedInUser;

        FormsAuthentication.SetAuthCookie(objLogedInUser.UserName, false);
        HttpCookie myCookie = new HttpCookie("DateOrWedCookie");
        DateTime now = DateTime.Now;

```

```

myCookie.Value = objLoggedInUser.UserName;
myCookie.Expires = now.AddYears(1);
Response.Cookies.Add(myCookie);

}
}

protected void btn_retrieve_Click(object sender, EventArgs e)
{
    Bindetails();
}

protected void btn_clear_Click(object sender, EventArgs e)
{
}

protected void ddl_productcat_SelectedIndexChanged(object sender, EventArgs e)
{
    //int prcat =Convert.ToInt16( ddl_productcat.SelectedIndex);
    //if (prcat == 0) { prcat = 1; }
    //PopDlIs ObjPD = new PopDlIs();
    //ObjPD.BindDropDowns(ddl_productgroup, (new
PopDlIs()).GetproductGroupbyServices(2));

    //ddl_productgroup.Enabled = true;
}
protected void ddl_productgroup_SelectedIndexChanged(object sender, EventArgs e)
{
    //Session["Pgid"] = ddl_productgroup.SelectedValue;
    Bindetails();
}
}
}
*****VendorServicesShow.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/ServicesMasterPage.master"
AutoEventWireup="true" CodeFile="VendorServicesShow.aspx.cs"
Inherits="VendorServicesShow" %>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
Runat="Server">
</asp:Content>
<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" Runat="Server">
    <div class="col-md-9 col-sm-12 col-xs-12">
        <div class="row">
            <div id="dvMsgBox" class="span10 ordersave" runat="server"
style="display: block; height:100px">
                <asp:Label ID="Label1" runat="server" Text=""></asp:Label>
            </div>
        </div></div>

</asp:Content>

<asp:Content ID="Content5" ContentPlaceHolderID="CPH_MainContent" Runat="Server">

```



```

<!-- Content Bodey Here -->

    <!-- START MAIN CONTAIN -->
    <div class="main-contain shop-product bg-white">
        <div class="container">
            <div class="row">
                <aside>
                    <div id="sidebar" class="col-md-3 col-sm-12 col-xs-12 sidebar" runat="server">
                        <!-- CATEGORIES WIDGET -->
                        <div id="Filter" class="widget mb30" runat="server">
                            <h6 class="text-uppercase bottom-line">Show Results for</h6>
                            <ul class="icons-list">
                                <li><a href="#">Sarees <span class="pull-right">12</span></a></li>
                                <li><a href="#">Lehengas <span class="pull-right">11</span></a></li>
                                <li><a href="#">Indo Westers <span class="pull-right">10</span></a></li>
                                <li><a href="#">Salwar Kameez <span class="pull-right">14</span></a></li>
                                <li><a href="#">Kurtees <span class="pull-right">15</span></a></li><!--%>
                            </ul>
                        </div>
                        <!-- END CATEGORIES WIDGET -->

                        <!-- SIDE-BAR PRODUCT WIDGET -->
                        <div id="toprate" class="" runat="server">
                            <div class="widget mb30">
                                <h6 class="text-uppercase bottom-line">Top Rated</h6>
                                </div>
                                <div class="small-thumb-proudct mb30">
                                    <div class="thumb-product">
                                        
                                        <div class="thumb-product-inner">
                                            <h6><a href="shoppingdetail.aspx">Printed Su..</a></h6>
                                            <span class="thumb-product-cat">Fashion</span>
                                        </div>
                                        <ul class="list-inline overflow-h">
                                            <li class="thumb-product-price line-through">$60.00</li>
                                            <li class="thumb-product-price">$30.00</li>
                                        </ul>
                                    </div>
                                </div>
                                <div class="small-thumb-proudct mb30">
                                    <div class="thumb-product">
                                        
                                        <div class="thumb-product-inner">
                                            <h6><a href="shoppingdetail.aspx">Printed Su..</a></h6>
                                            <span class="thumb-product-cat">Fashion</span>
                                        </div>
                                        <ul class="list-inline overflow-h">
                                            <li class="thumb-product-price">$65.00</li>
                                        </ul>
                                    </div>
                                </div>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>

```

```

<div class="thumb-product">
  
  <div class="thumb-product-inner">
    <h6><a href="shoppingdetail.aspx">Printed Su..</a></h6>
    <span class="thumb-product-cat">Fashion</span>
  </div>
  <ul class="list-inline overflow-h">
    <li class="thumb-product-price line-through">$75.00</li>
    <li class="thumb-product-price">$65.00</li>
  </ul>
</div>
</div>--%>
</div>
<!-- END SIDE-BAR PRODUCT WIDGET -->

<!-- SIDE-BAR PRODUCT WIDGET -->
<div id="bestseller" class="" runat="server">
  <div class="widget mb30">
    <h6 class="text-uppercase bottom-line">Best Sellers</h6>
    </div>
<%--
      <div class="small-thumb-proudct mb30">
        <div class="thumb-product">
          
          <div class="thumb-product-inner">
            <h6><a href="shoppingdetail.aspx">Printed Su..</a></h6>
            <span class="thumb-product-cat">Fashion</span>
          </div>
          <ul class="list-inline overflow-h">
            <li class="thumb-product-price line-through">$95.00</li>
            <li class="thumb-product-price">$75.00</li>
          </ul>
        </div>
      </div>
      <div class="small-thumb-proudct mb30">
        <div class="thumb-product">
          
          <div class="thumb-product-inner">
            <h6><a href="shoppingdetail.aspx">Printed Su..</a></h6>
            <span class="thumb-product-cat">Fashion</span>
          </div>
          <ul class="list-inline overflow-h">
            <li class="thumb-product-price line-through">$70.00</li>
            <li class="thumb-product-price">$60.00</li>
          </ul>
        </div>
      </div>
      <div class="small-thumb-proudct mb30">
        <div class="thumb-product">
          
          <div class="thumb-product-inner">
            <h6><a href="shoppingdetail.aspx">Printed Su..</a></h6>

```

```

        <span class="thumb-product-cat">Fashion</span>
    </div>
    <ul class="list-inline overflow-h">
        <li class="thumb-product-price">$65.00</li>
    </ul>
</div>
</div>
--%>
    </div>

```

<!-- END SIDE-BAR PRODUCT WIDGET -->

```

<!-- TAGS WIDGET -->
<div class="widget">
    <h6 class="text-uppercase bottom-line">Tags</h6>
    <div class="tags">
        <a href="#">Fashion</a>
        <a href="#">Electronics</a>
        <a href="#">Services</a>
        <a href="#">Businesses</a>
        <a href="#">Shops</a>
        <a href="#">Markets</a>
    </div>
</div>
<!-- END TAGS WIDGET -->
</div>
</aside>

```

```

<div class="col-md-9 col-sm-12 col-xs-12">

```

```

        <div class="row hidden-xs">
            <div class="toolbar">
                <div id="results" class="col-sm-4 category-num" runat="server">
                    <h2>Designer</h2>
                    <small class="category-num-coun">9 Results</small>
                </div>
            </div>--%>
            <div class="col-sm-8 pull-right">
                <ul class="view-mode">
                    <li class="grid-list-icons">
                        <asp:Button ID="btn1" runat="server" OnClick="btn1_Click"
                        type="button" class="btn btn-floating color-green " Text="List" /><i class='fa fa-th-list'></i>
                        <asp:Button ID="btn2" runat="server" OnClick="btn2_Click" type="button"
                        class="btn btn-floating color-green " Text="Grid" /><i class='fa fa-th'></i> wrn-btn--%>
                            <asp:ImageButton ID="ImgBtn1" runat="server"
                            ImageUrl="images/small/list.png" OnClick="ImgBtn1_Click" />
                            <asp:ImageButton ID="ImgBtn2" runat="server"
                            ImageUrl="images/small/grid.png" OnClick="ImgBtn2_Click" />
                        <!--
                            <button id="Button1" type="button" class="btn btn-secondary wrn-
                            btn mt-0 ml-0" onclick="btn1_Click" ><i class='fa fa-th-list'></i>&nbsp;&nbsp;&nbsp;List</button>
                            <button id="Button2" type="button" style="width:10px" class="btn btn-
                            product " onclick="btn2_Click" ><i class='fa fa-th'></i></button>--%>
                    </li>
                    <li class="sort-list-btn">

```

```

        <h6>Sort By :</h6>
        <div class="btn-group">
            <button type="button" class="btn btn-default dropdown-toggle"
data-toggle="dropdown">
                Relevent <span class="caret"></span>
            </button>
            <ul class="dropdown-menu" role="menu">
                <li><a href="#">Relevent</a></li>
                <li><a href="#">Price High to Low</a></li>
                <li><a href="#">Price Low to High</a></li>
                <li><a href="#">New Arrived</a></li>
            </ul>
        </div>
    </li>
    <li class="sort-list-btn">
        <h6>Show :</h6>
        <div class="btn-group">
            <button type="button" class="btn btn-default dropdown-toggle"
data-toggle="dropdown">
                30 <span class="caret"></span>
            </button>
            <ul class="dropdown-menu" role="menu">
                <li><a href="#">30</a></li>
                <li><a href="#">15</a></li>
                <li><a href="#">9</a></li>
                <li><a href="#">6</a></li>
            </ul>
        </div>
    </li>
</ul>
</div>
</div>
<div class="col-md-9 col-sm-12 col-xs-12">
    <div class="row hidden-xs">
        <div class="toolbar">
            <div id="results" class="col-sm-8 category-num" runat="server">
                <h2>Venues</h2>
                <small class="category-num-coun">9 Results</small>
            </div>
        </div>
    </div>
</div>
</div>
<!--dynamic code from here-->
<div id="appvendors" class="category-products" runat="server" style="display:block">

    </div>
<!--dynamic code ends here-->

    <div id="AppGrid" class="category-products col-grid-style" runat="server"
style="display:none">

```

```

</div>
        <%--dynamic code ends here--%>

<!-- Start Pagination-->
    <div class="cd-pagination">
    <ul class="pagination">
    <li><a href="#"><i class="fa fa-angle-left" aria-hidden="true"></i></a></li>
    <li class="active"><a href="#">1</a></li>
    <li><a href="#">2</a></li>
    <li><a href="#">3</a></li>
    <li><a href="#">4</a></li>
    <li><a href="#">5</a></li>
    <li><a href="#"><i class="fa fa-angle-right" aria-hidden="true"></i></a></li>
    </ul>
    </div>
    <!-- End Pagination-->
    </div>
    </div>
    </div>
        <!-- END MAIN CONTAIN -->

    <!-- ./Content Bodey Here -->

</asp:Content>
<asp:Content ID="Content6" ContentPlaceHolderID="ContentFooter" Runat="Server">
</asp:Content>

***** VendorServicesShow.aspx.cs*****
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBAL;
using DowBEntity;
//using LogTracer;
using System.Text;
using System.Text.RegularExpressions;

public partial class VendorServicesShow : System.Web.UI.Page
{
    DDLogTracer objlog = new DDLogTracer();
    bool IsMobile = false;
    protected void Page_Load(object sender, EventArgs e)
    {
        CIsLoggedInUser CIsusr = (CIsLoggedInUser)Session["LoggedInUser_Specs"];
        if (CIsusr != null)
        {

        }

        if (Request.Browser.IsMobileDevice) { IsMobile = true; } else { IsMobile = false; }
    }
}

```

```

        if (IsMobile)
        {
            sidebar.Attributes.CssStyle.Add("display", "none;");
        }
        if (!IsPostBack) { Bindetails(); }
    }
    private void Bindetails()
    {
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        string pt = "cat",title="";
        ImgBtn1.Enabled = false;
        ImgBtn2.Enabled = true;
        appvendors.Attributes.CssStyle.Add("display", "block;");
        AppGrid.Attributes.CssStyle.Add("display", "none;");
        string srch = "";

        int pid = Convert.ToInt16(Request.QueryString["cid"]);
        string PreHtml = "";
        StringBuilder sb = new StringBuilder();
        StringBuilder sbg = new StringBuilder();
        string vendor = Request.QueryString["vendor"];

        sb.Append(" <div class='directory-list'>");
        if (pid == 0)
        {
            pid = Convert.ToInt16(Request.QueryString["sid"]);
            pt = "prod";
        }
        if (pid == 0)
        {
            srch = Request.QueryString["SrchStrng"];
            if (srch != null && srch != "") { pt = "search"; }
        }

        try
        {
            IList<ProductProductSeller> Lst=new List<ProductProductSeller>();
            if (vendor != null)
            {
                Lst = (new Products()).GetProductSellerbySellerId(Convert.ToInt16(vendor), 2);
                pt = "vendor";
            }

            else if (pt == "prod")
            {
                Lst = (new Products()).GetProductSellerbyProductId(pid, "SearchList");
            }
            else if ( pt=="cat")
            {
                Lst = (new Products()).GetProductSellerbyCatId(pid, 2, "SearchList");
            }
        }
    }

```

```

}
else if(pt == "search")
{
    Lst = (new Products()).GetProductSellerbySearch(srch, 2);
    if (Lst.Count() == 0)
    {
        Lst = (new Products()).GetProductSellerbySearch("", 2);
    }
}

if(Lst.Count()==0)
{
    if (pid > 0)
    {
        Eteproduct ep = new Products().GetProduct(pid, "");
        string pname = ep.ProductName;
        Lst = (new Products()).GetProductSellerbySearch(pname, 2);
    }
}
if (Lst.Count > 0) {
    ProductProductSeller pst = (from c in Lst select c).FirstOrDefault();

    if (pt == "prod")
    {
        title = pst.ProductName;
    }
    else
    {
        title = pst.CatName;
    }
    sb.Append("<h2>" + title + "</h2>\n <small class='category-num-coun'" +
Convert.ToString(Lst.Count) + " Results</small>\n");
    results.InnerHtml = sb.ToString();
    sb = new StringBuilder();
    string path = "images/NoImage.jpg";
    string path1 = "images/NoImage.jpg";
    if (Lst != null && Lst.Count > 0)
    {
        sbg.Append("\n <div class='row clearfix'>");
        sbg.Append(" <div class='row clearfix'>");

        foreach (var dt in Lst)
        {
            string href = "";
            href = "ShoppingDetail.aspx?id=" + dt.PSId;
            var list = new Products().GetProductSellerImage((int)dt.PSId, "").ToList();
            int cnt = list.Count();
            if (cnt > 1)
            {
                path = list[0].ImagePath;
                path1 = list[1].ImagePath;
            }
            else

```

```

        if (list.Count() > 0)
        {
            path = list[0].ImagePath;
        }

        if (path == null || path == "")
        {
            path = "images/NoImage.jpg";
        }
        if (path1 == null || path1 == "") { path1 = path; }
        List<Eteproduct> stk = new Products().GetSellerServicesbypsid((int)dt.PSId);
        List<Eteproduct> oth = new
Products().GetUserOtherServices(dt.UserId,dt.PSId);
        //list code
        sb.Append(" <div class='products-list products-list-description-brd mb30'>
\n");
        sb.Append(" <div class='row'> \n");
        sb.Append(" <div class='col-sm-4'> \n");
        sb.Append(" <div class='product-grid-img'> \n");
        sb.Append(" <ul class='product-labels'> \n");
        sb.Append(" <li>Sale</li> \n");
        sb.Append(" </ul> \n");
        sb.Append(" <a href='shoppingdetail.aspx?pid=" + dt.PSId + "'
class='product-grid-img-thumbnail'> \n");
        sb.Append("  \n");
        sb.Append("  \n");
        sb.Append(" </a> \n");
        sb.Append(" <div class='product-grid-atb text-center'><i class='fa fa-
shopping-cart'></i><a class=" href='shoppingdetail.aspx?pid=" + dt.PSId + "'>Add to
Cart</a></div> \n");
        sb.Append(" </div> \n");
        sb.Append(" </div> \n");
        sb.Append(" <div class='col-sm-8 products-list-description'> \n");
        sb.Append(" <div class='overflow-h'> \n");
        sb.Append(" <ul class='list-inline overflow-h mb15'> \n");
        sb.Append(" <li><h4 class='title-price'><a href='shoppingdetail.aspx?pid="
+ dt.PSId + "'> + dt.ProductDescription + "</a></h4></li> \n");
        sb.Append(" <li class='pull-right'> \n");
        sb.Append(" <ul class='list-inline product-ratings'> \n");
        sb.Append(" <li><i class='rating-selected fa fa-star'></i></li> \n");
        sb.Append(" <li><i class='rating-selected fa fa-star'></i></li> \n");
        sb.Append(" <li><i class='rating-selected fa fa-star'></i></li> \n");
        sb.Append(" <li><i class='rating fa fa-star'></i></li> \n");
        sb.Append(" <li><i class='rating fa fa-star'></i></li> \n");
        sb.Append(" </ul> \n");
        sb.Append(" </li> \n");
        sb.Append(" </ul> \n");
        sb.Append(" <div class='mb10'> \n");
        sb.Append(" <span class='title-price'>" + dt.Price.ToString() +
"</span><span> For " + dt.PriceUnits + " " + dt.QtyUOM + "</span>\n");
        //sb.Append(" <span class='title-price line-through'>" +
dt.Price.ToString() + "</span> \n");
        sb.Append(" </div> \n");

```



```

sb.Append(" <p class='mb15'>" + dt.SellerDescription + "</p> \n");

//services included
sb.Append(" <h3 class='shop-product-title'>Services Included</h3>
\n");
sb.Append(" <ul class='list-inline product-size mb30'> \n");
foreach (var sz in stk)
{
sb.Append(" <li><h5 class='title-price'><a
href='servicedetail.aspx?pid=" + dt.PSId + ">" + sz.ProductName + "</a></h5></li>
\n");
}
sb.Append(" </ul> \n");
sb.Append(" <h3 class='shop-product-title bottom-line'>Services
Available</h3>
\n");
sb.Append(" <ul class='list-inline product-color mb30'> \n");
sb.Append(" <li> \n");
foreach (var ser in oth)
{
sb.Append(" <li><h5 class='title-price'><a
href='serviceslist.aspx?pid=" + ser.ProductId + ">" + ser.ProductName + "</a></h5></li>
\n");
}

sb.Append(" </ul> \n");

//servicesends
sb.Append(" <ul class='list-inline add-to-wishlist mb15'> \n");
sb.Append(" <li class='product-wishlist'> \n");
sb.Append(" <i class='fa fa-heart'></i> \n");
sb.Append(" <a href='MyWishList.aspx?psid=" + dt.PSId + ">Add to
Wishlist</a> \n");
sb.Append(" </li> \n");
sb.Append(" <li class='product-wishlist'> \n");
sb.Append(" <i class='fas fa-exchange'></i> \n");
sb.Append("<li><a href='ServicesList.aspx?vendor=" + dt.SellerId + ">" +
"Services Provided by" + dt.SellerName + "</a> \n");
//sb.Append(" <a href='shoppingdetail.aspx?pid=" + dt.PSId + ">Add
to Compare</a>
\n");
sb.Append(" </li> \n");
sb.Append(" </ul> \n");
sb.Append(" <a href='MyCart.aspx?pid=" + dt.PSId + "' class='btn
btn-product'>Add to Cart</a> \n");
//sb.Append(" <button type='button' class='btn btn-product'>Add to
Cart</button> \n");
sb.Append(" </div> \n");
sb.Append(" </div> \n");
sb.Append(" </div> \n");
sb.Append(" </div> \n");
sb.Append(" \n");

//grid code
sbg.Append(" <div class='col-md-4 col-sm-6 col-xs-12'> \n");
sbg.Append(" <div class='products grid-product mb30'> \n");

```

```

        sbg.Append("        <div class='product-grid-inner'>        \n");
        sbg.Append("        <div class='product-grid-img'>        \n");
        sbg.Append("        <ul class='product-labels'>        \n");
        sbg.Append("        <li>Sale</li>        \n");
        sbg.Append("        </ul>        \n");
        sbg.Append("        <a href='shoppingdetail.aspx?pid=' + dt.PSId + "
class='product-grid-img-thumbnail'> \n");
        sbg.Append("        <img src='\" + path + \"' alt='img' class='img-responsive
primary-img' />        \n");
        sbg.Append("        <img src='\" + path1 + \"' alt='img' class='img-responsive
socendary-img' />        \n");
        sbg.Append("        </a>        \n");
        sbg.Append("        <div class='product-grid-atb text-center'><a class="
href='shoppingdetail.aspx?pid=' + dt.PSId + ">Quick View</a></div>        \n");
        sbg.Append("        </div>        \n");
        sbg.Append("        <div class='product-grid-caption text-center'>
\n");
        sbg.Append("        <h6><a class='p-grid-title'
href='shoppingdetail.aspx?pid=' + dt.PSId + ">" + dt.ProductDescription + "</a></h6>
\n");
        sbg.Append("        <p>" + dt.SellerDescription + "</p>        \n");
        sbg.Append("        <span class='price clearfix'><span class='old-price'>" +
dt.Price.ToString() + "</span> $ <span class='special-price'>" + dt.Price.ToString() +
"</span></span>        \n");
        sbg.Append("        <a class='btn btn-product cd-cart-btn'
href='MyCart.aspx?psid=' + dt.PSId + "><i class='fa fa-shopping-cart'></i> Add to Cart</a>
\n");
        sbg.Append("<li><a href='ServicesList.aspx?vendor=' + dt.SellerId + ">" +
"Services Provided by" + dt.SellerName + "</a>        \n");
        sbg.Append("        <h3 class='shop-product-title bottom-line'>Services
Included</h3> \n");
        sbg.Append("        <ul class='list-inline product-size mb30'>        \n");
        foreach (var sz in stk)
        {
            sbg.Append("        <li><h5 class='title-price'><a
href='ServicesList.aspx?sid=' + dt.ProductId + ">" + sz.ProductName + "</a></h5></li>
\n");
        }
        sbg.Append("        </ul>        \n");
        sbg.Append("        </div>        \n");
        sbg.Append("        </div>        \n");
        sbg.Append("        </div>        \n");
        sbg.Append("        </div>        \n");
    }

    sbg.Append("    </div> \n");
    sbg.Append("    </div> \n");

    PreHtml = sb.ToString();
    if (PreHtml.Length > 0)
    {

```

```

        //objlog.Trace_Log(4, "ServicesList", "1", "appvendors", PreHtml,
Session["Logpath"].ToString());
        appvendors.InnerHtml = PreHtml;
    }
    PreHtml = sbg.ToString();
    if (PreHtml.Length > 0)
    {
        objlog.Trace_Log(4, "ServicesList", "1", "AppGrid", PreHtml,
Session["Logpath"].ToString());
        AppGrid.InnerHtml = PreHtml;
    }
}
sb = new StringBuilder();
int grp = pst.ProductGroupId;
List<Eteproduct> prd = new Products().GetProductbyGroup(grp, "");
if (prd.Count > 0)
{
    sb.Append("<h6 class='text-uppercase bottom-line'>Show Results
for</h6>\n");
    sb.Append("<ul class='icons-list'>\n");
    foreach (var data in prd)
    {
        if (Convert.ToInt16( data.Type) > 0)
        {
            sb.Append("<li><a href='ServicesList.aspx?sid=" + data.ProductId + ">"
+ data.ProductName + " <span class='pull-right'>" + data.Type + "</span></a></li>\n");
        }
    }
    sb.Append("</ul>\n");
    Filter.InnerHtml = sb.ToString();
}
sb = new StringBuilder();
sb.Append("    <div class='widget mb30'> ");
sb.Append("    <h6 class='text-uppercase bottom-line'>Top Rated</h6> ");
sb.Append("    </div> ");
List<ProductProductSeller> top = new
Products().GetProductSellerbySearch("",2);
foreach (var tr in top)
{
    sb.Append("    <div class='small-thumb-proudct mb30'> ");
    sb.Append("    <div class='thumb-product'> ");
    sb.Append("     ");
    //sb.Append("    <div class='thumb-product-inner'> ");
    sb.Append("    <span class='thumb-product-cat'>" + tr.CatName + "</span>
");
    sb.Append("    <h6><a href='shoppingdetail.aspx?pid=" + tr.PSId + ">" +
tr.ProductDescription + "</a></h6> ");
    sb.Append("    </div> ");
    sb.Append("    <ul class='list-inline overflow-h'> ");
    sb.Append("    <li class='thumb-product-price line-
through'>" +tr.CurPrice+"</li> ");

```

```

        sb.Append("    <li class='thumb-product-price'>" + tr.CurPrice + "</li>    ");
        sb.Append("    </ul> ");
        //sb.Append(" </div> ");
        sb.Append(" </div> ");
    }
    toprate.InnerHtml = sb.ToString();
    sb = new StringBuilder();
    sb.Append("    <div class='widget mb30'>    ");
    sb.Append("        <h6 class='text-uppercase bottom-line'>Best Sellers</h6> ");
    sb.Append("    </div> ");

    //List<ProductProductSeller> best = new
Products().GetProductSellerbySearch("");
    foreach (var tr in top)
    {
        sb.Append("    <div class='small-thumb-proudct mb30'>    ");
        sb.Append("    <div class='thumb-product'> ");
        sb.Append("    <img class='thumb-product-img img-responsive' src=\"" +
tr.ImagePath + "\" alt=\"">    ");
        //sb.Append("    <div class='thumb-product-inner'>    ");
        sb.Append("    <span class='thumb-product-cat'>" + tr.CatName + "</span>
");
        sb.Append("    <h6><a href='shoppingdetail.aspx?pid=\"" + tr.PSId + "\"> +
tr.ProductDescription + "</a></h6> ");
        sb.Append("    </div> ");
        sb.Append("    <ul class='list-inline overflow-h'>    ");
        sb.Append("    <li class='thumb-product-price line-through'>" + tr.CurPrice +
"</li>    ");
        sb.Append("    <li class='thumb-product-price'>" + tr.CurPrice + "</li>    ");
        sb.Append("    </ul> ");
        //sb.Append(" </div> ");
        sb.Append(" </div> ");
    }
    bestseller.InnerHtml=sb.ToString();
    sb = new StringBuilder();
}
}
catch (Exception ex)
{
    Common.ShowMessageHtml(ex.Message, 1, true, this.dvMsgBox);
    ExceptionLogging.SendExcepToDB(ex);
}
}

protected void btn1_Click(object sender, EventArgs e)
{
    appvendors.Attributes.CssStyle.Add("display", "block;");
    AppGrid.Attributes.CssStyle.Add("display", "none;");
}
protected void btn2_Click(object sender, EventArgs e)
{

```

```

        appvendors.Attributes.CssStyle.Add("display", "none;");
        AppGrid.Attributes.CssStyle.Add("display", "block;");
    }
    protected void ImgBtn1_Click(object sender, ImageClickEventArgs e)
    {
        appvendors.Attributes.CssStyle.Add("display", "block;");
        AppGrid.Attributes.CssStyle.Add("display", "none;");
        ImgBtn1.Enabled = false;
        ImgBtn2.Enabled = true;
    }
    protected void ImgBtn2_Click(object sender, ImageClickEventArgs e)
    {
        appvendors.Attributes.CssStyle.Add("display", "none;");
        AppGrid.Attributes.CssStyle.Add("display", "block;");
        ImgBtn1.Enabled = true;
        ImgBtn2.Enabled = false;
    }
}
}
*****VendorServicesUser.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/VendorMasterPage.master"
AutoEventWireup="true" CodeFile="VendorServicesUser.aspx.cs"
Inherits="VendorServicesUser" %>
<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
Runat="Server">
    <div>
        <asp:Label ID="Label13" runat="server" Text="" Height="20px"></asp:Label>
    </div>
</asp:Content>
<asp:Content ID="Content4" ContentPlaceHolderID="CPH_TopMessages" Runat="Server">
    <div class="span10">
        <div class="row">
            <%--<h3>Ravikiran Techno Soft</h3>--%>
            <div id="dvMsgBox" class="span10 ordersave" runat="server" style="display:
block;">
                <asp:Label ID="Label2" runat="server" Text="" Height="50px"></asp:Label>
                <asp:Label ID="Label1" runat="server" Text="" Height="60px"></asp:Label>
            </div>
        </div>
    </div>
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentMain" Runat="Server" >
    <div class="container-fluid" style="padding-left:100px; padding-right:50px">
        <div><%--VendorProductsAdd.aspx--%>
            <h5 class="my-4 dark-grey-text font-weight-bold">Products
            <a href="PostServices.aspx" class="btn btn-outline-danger waves-effect btn-sm font-
weight-bold"><i class="far fa-plus-square"></i> &nbsp;Add New</a>
            <%--
            <a href="#" class="btn btn-outline-danger waves-effect btn-sm font-weight-
bold"><i class="fas fa-edit"></i> &nbsp;Edit</a> --%>
            <a href="#" class="btn btn-outline-danger waves-effect btn-sm font-weight-bold"><i
class="fas fa-edit"></i> &nbsp;Edit</a>

```

```

        <a href="#" class="btn btn-outline-danger waves-effect btn-sm font-weight-bold"><i
class="fas fa-trash-alt"></i> &nbsp;Delete</a>
    </h5>
</div>
<!-- Navbar links -->
<section class="mb-5">

<!-- Card -->
<div class="card card-cascade narrower">

    <section>

<div class="card">
    <div id="prtable" class="col-md-12 col-sm-auto card-body" runat="server">

<!-- Table -->
<table class="table table-striped" runat="server">

    <!-- Table head -->
    <thead class="mdb-color darken-3">
    <tr class="text-white">
        <th scope="row">
            <input class="form-check-input" type="checkbox" id="checkbox1">
            <label for="checkbox1" class="label-table"></label>
        </th>
        <th><i class="far fa-image"></i></th>
        <th style="width:20%">Name</th>
        <th>Stock</th>
        <th>Price</th>
        <th style="width:20%">Categories</th>
        <th>Date</th>
    </tr>
    </thead>
    <!-- Table head -->

    <!-- Table body -->
    <tbody>
    <tr>
        <th scope="row">
            <input class="form-check-input" type="checkbox" id="checkbox2">
            <label for="checkbox2" class="label-table"></label>
        </th>
        <td></td>
        <td><a href="#">Best Bridal Dresses</a></td>
        <td class="text-success font-weight-bold">In stock</td>
        <td><s>₹8099</s><br>₹7999</td>
        <td><span class="badge badge-primary">Women Fashion</span> <span
class="badge badge-primary">Bridal</span></td>
        <td>Published<br>2018/11/16</td>
    </tr>
    <tr>
        <th scope="row">
            <input class="form-check-input" type="checkbox" id="checkbox3">
            <label for="checkbox3" class="label-table"></label>

```

```

        </th>
        <td></td>
        <td><a href="#">Best Groom Dresses</a></td>
        <td class="text-success font-weight-bold">In stock</td>
        <td><s>₹9099</s><br>₹8999</td>
        <td><span class="badge badge-primary">Men Fashion</span> <span
class="badge badge-primary">Groom</span></td>
        <td>Published<br>2018/11/16</td>
    </tr>
    <tr>
    <th scope="row">
        <input class="form-check-input" type="checkbox" id="checkbox4">
        <label for="checkbox4" class="label-table"></label>
    </th>
    <td></td>
    <td><a href="#">Best Bridegroom Dresses</a></td>
    <td class="text-danger font-weight-bold">Out Of stock</td>
    <td><s>₹10199</s><br>₹9999</td>
    <td><span class="badge badge-primary">Women Fashion</span> <span
class="badge badge-primary">Men Fashion</span> <span class="badge badge-
primary">Bridegroom</span></td>
    <td>Published<br>2018/11/16</td>
    </tr>
    <tr>
    <th scope="row">
        <input class="form-check-input" type="checkbox" id="checkbox5">
        <label for="checkbox5" class="label-table"></label>
    </th>
    <td></td>
    <td><a href="#">Best Bridal Dresses</a></td>
    <td class="text-success font-weight-bold">In stock</td>
    <td><s>₹ 8099</s><br>₹7999</td>
    <td><span class="badge badge-primary">Women Fashion</span> <span
class="badge badge-primary">Bridal</span></td>
    <td>Published<br>2018/11/16</td>
    </tr>
    <tr>
    <th scope="row">
        <input class="form-check-input" type="checkbox" id="checkbox6">
        <label for="checkbox6" class="label-table"></label>
    </th>
    <td></td>
    <td><a href="#">Best Groom Dresses</a></td>
    <td class="text-success font-weight-bold">In stock</td>
    <td><s>₹9099</s><br>₹8999</td>
    <td><span class="badge badge-primary">Men Fashion</span> <span
class="badge badge-primary">Groom</span></td>
    <td>Published<br>2018/11/16</td>
    </tr>
    <tr>
    <th scope="row">
        <input class="form-check-input" type="checkbox" id="checkbox7">
        <label for="checkbox7" class="label-table"></label>
    </th>

```

```

        <td></td>
        <td><a href="#">Best Bridegroom Dresses</a></td>
        <td class="text-danger font-weight-bold">Out Of stock</td>
        <td><s>₹10199</s><br>₹9999</td>
        <td><span class="badge badge-primary">Women Fashion</span> <span
class="badge badge-primary">Men Fashion</span> <span class="badge badge-
primary">Bridegroom</span></td>
        <td>Published<br>2018/11/16</td>
    </tr>
</tbody>
<!-- Table body -->
</table>
<!-- Table -->

<hr class="my-0">

<!-- Bottom Table UI -->
<div class="d-flex justify-content-between">

    <!-- Name -->
    <select class="mdb-select colorful-select dropdown-primary md-form hidden-md-
down">
        <option value="" disabled>Rows number</option>
        <option value="1" selected>6 rows</option>
        <option value="2">25 rows</option>
        <option value="3">50 rows</option>
        <option value="4">100 rows</option>
    </select>

    <!-- Pagination -->
    <nav class="my-4">
        <ul class="pagination pagination-circle pg-blue mb-0">

            <!--First-->
            <li class="page-item disabled clearfix d-none d-md-block"><a class="page-
link">First</a></li>

            <!-- Arrow left -->
            <li class="page-item disabled">
                <a class="page-link" aria-label="Previous">
                    <span aria-hidden="true">&laquo;</span>
                    <span class="sr-only">Previous</span>
                </a>
            </li>

            <!-- Numbers -->
            <li class="page-item active"><a class="page-link">1</a></li>
            <li class="page-item"><a class="page-link">2</a></li>
            <li class="page-item"><a class="page-link">3</a></li>
            <li class="page-item"><a class="page-link">4</a></li>
            <li class="page-item"><a class="page-link">5</a></li>

            <!-- Arrow right -->
            <li class="page-item">
                <a class="page-link" aria-label="Next">

```



```

        <span aria-hidden="true">&raquo;</span>
        <span class="sr-only">Next</span>
    </a>
</li>

    <!-- First -->
    <li class="page-item clearfix d-none d-md-block"><a class="page-
link">Last</a></li>

</ul>
</nav>
<!-- Pagination -->

</div>
<!-- Bottom Table UI -->

</div>
</div>

</section>

</div>
<!-- Card -->

</section>

</div>
</asp:Content>
<asp:Content ID="Content3" ContentPlaceHolderID="ContentPlaceHolderfooter"
Runat="Server">

</asp:Content>

*****VendorServicesUser.aspx.cs *****
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBEntity;
using DowBAL;
using System.Text;

public partial class VendorServicesUser : System.Web.UI.Page
{
    DDLogTracer objlog = new DDLogTracer();

    protected void Page_Load(object sender, EventArgs e)
    {
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr != null)
        {
            string pageid = "6";// Request.QueryString["pageid"];

```

```

        //ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
Convert.ToInt32(Clsusr.GroupId));
        string msg = Request.QueryString["msg"];
        if (msg != null)
        {
            //Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
        }
        if (Convert.ToInt16(Clsusr.GroupId) > 3)
        {
            Bindetails(Convert.ToInt16(Session["VendorCode"]));
        }
    }
    else
    {
        Response.Redirect("MLB_Login.aspx");
    }
}
private void Bindetails(int userid)
{
    //try
    //{
        StringBuilder sb = new StringBuilder();
        prtable.InnerHtml = "";

        sb.Append("<table class='table' runat='server'>\n");
        sb.Append("    <thead class='mdb-color darken-3'>\n");
        sb.Append("        <tr class='text-white'> \n");
        sb.Append("            <th scope='row'> \n");
        sb.Append("                <input class='form-check-input' type='checkbox' id='checkboxall'>
\n");
        sb.Append("                    <label for='checkboxall' class='label-table'></label>\n");
        sb.Append("                </th> \n");
        sb.Append("                <th><i class='far fa-image'></i></th> \n");
        sb.Append("                <th style='width:20%'>Name</th> \n");
        sb.Append("                <th>Availability</th> \n");
        sb.Append("                <th>Price</th> \n");
        sb.Append("                <th style='width:30%'>Categories</th> \n");
        sb.Append("                <th>Date</th> \n");
        sb.Append("                <th style='width:10%'>Action</th> \n");
        sb.Append("            </tr> \n");
        sb.Append("        </thead> \n");
        sb.Append("        <tbody> \n");

        List<ProductProductSeller> Lst = (new
Products()).GetProductSellerbySellerId(userid,2);

        if (Lst != null && Lst.Count > 0)
        {
            foreach (var dt in Lst)
            {
                sb.Append("                <tr> \n");
                sb.Append("                    <th scope='row'> \n");
                sb.Append("                        <input class='form-check-input' type='checkbox'
id='checkbox"+dt.PSId+"'> \n");

```

```

        sb.Append("        <label for='checkbox' + dt.PSId + "" class='label-
table'></label>\n");
        sb.Append("        </th> \n");
        sb.Append("        <td><img src="" + dt.ImagePath+ "" class='img-fluid'
style='height:100px;width:100px'></td> \n");
        sb.Append("        <td><a href='#">"+dt.SellerDescription+"</a></td> \n");
        if (dt.TStock > 0) {
            sb.Append("        <td class='text-success font-weight-bold'>Check Availability("
+ dt.TStock + ")</td> \n");
        }
        else
        {
            sb.Append("        <td class='text-danger font-weight-bold'>Check
Availability</td> \n");
        }
        //<td class="text-danger font-weight-bold">Out Of stock</td>
        sb.Append("        <td><s>₹" + dt.Price + "</s><br>₹" + dt.Price + "</td>
\n");
        sb.Append("        <td><span class='badge badge-primary'>" + dt.GroupName +
"</span></span> <span class='badge badge-primary'>" + dt.ProductName + "</span></td>
\n");
        sb.Append("        <td>Published<br>" + Convert.ToString(
dt.CreatedOn).Substring(0,10)+"</td> \n");
        sb.Append("        <td><a href='/postservices.aspx?psid=" + dt.PSId +
"&pagetype=Edit' class='btn btn-outline-danger waves-effect btn-sm font-weight-bold'><i
class='fas fa-edit'></i> &nbsp;Edit</a></td>");
        sb.Append("        </tr> \n");

    }
    sb.Append(" </tbody> \n");
    prtable.InnerHtml = sb.ToString();
    objlog.Trace_Log(1, "home", "1", "menu", "\n" + sb.ToString());

}
// WDGrid_UserList.DataSource = Lst;
// WDGrid_UserList.DataBind();
// WDGrid_UserList.Visible = true;
//}
else
{
    prtable.InnerHtml = "No Data found";
    Common.ShowMessageHtml("No Data found", 1, true, dvMsgBox);

}
}
}

}
*****VendorSizesAdd.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/ProductMaster.master"
AutoEventWireup="true" CodeFile="VendorSizesAdd.aspx.cs" Inherits="VendorSizesAdd"
%>
<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
Runat="Server">
</asp:Content>
<asp:Content ID="Content4" ContentPlaceHolderID="CPH_TopMessages" Runat="Server">

```

```

        <div class="col-lg-12 col-md-12 col-sm-12">
            <div class="row">
                <%--<h3>Ravikiran Techno Soft</h3>--%>
                <div id="dvMsgBox" class="span10 ordersave" runat="server" style="display:
block;";>
<%--
                <asp:Label ID="Label2" runat="server" Text=""
Height="100px"></asp:Label>--%>
                <asp:Label ID="Label3" runat="server" Text="" Height="60px"></asp:Label>
                </div>
            </div>
        </div>
        <div class="container-fluid">
            <div class="row-fluid" style="padding-top: 20px;";>
                <div>
                    <h5 class="my-4 dark-grey-text font-weight-bold" style="padding-left:20px">Manage
Selected Colors
                    <asp:Button ID="btn_save" runat="server" CssClass="btn btn-secondary btn-md"
Text="Save" OnClick="btn_save_Click" /> <%-- btn-rounded--%>
                    <asp:Button ID="btn_clear" runat="server" CssClass ="btn btn-secondary btn-md btn
rounded" Text="Clear" OnClick="btn_clear_Click" />
<%--
                    <a href="#" class="btn btn-secondary btn-md btn-rounded"> &nbsp;Edit</a> <%--
<i class="fas fa-edit"></i>
                    <a href="#" class="btn btn-secondary btn-md btn-rounded"> &nbsp;Delete</a><%--<i
class="fas fa-trash"></i>--%>
                    </h5>
                <!-- Navbar links -->

                </div>
            <div class="span10">
                <div class="row">
                    <div id="Div1" class="span12 ordersave" runat="server" style="display: none;";>
                        <asp:Label ID="Label1" runat="server" Text=""></asp:Label>
                    </div>
                </div>
            </div>
        </div>

<%-- <div id="LeftSideBar" runat="server" class="span2">
    <asp:Table ID="Table1" runat="server">
        </asp:Table>
    </div>--%>
</div>
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="CPH_MainContent" Runat="Server" >
    <div class="container-fluid" style="padding-left:100px; padding-right:50px">

        <!-- Navbar links -->
        <section class="mb-5">

            <!-- Card -->
            <div class="card card-cascade narrower">
                <div class=" col-lg-8 col-md-8 col-sm-8 category-num">
                    <table id="Table5" class="table table-striped table-detail col-lg-8"
runat="server">

```

```

        <tbody>

        <tr>
        <td class="span3" dir="auto" style="width:270px">Product Sizes :
    </td>

        <td class="span3">
        <asp:DropDownList ID="ddl_productcat" runat="server"
    OnSelectedIndexChanged="ddl_productcat_SelectedIndexChanged" Visible="true"
    Width="200px">
        </asp:DropDownList>
        </td>
        <td>
        <asp:Button ID="Retrieve" runat="server" CssClass="btn btn-
    secondary btn-md " OnClick="btn_retrieve_Click" Text="Retrieve Data" />
        </td>
        </tr>

        </tbody></table>
    </div>

```

```
</div>
```

```
</section>
```

```
<section class="mb-5">
```

```
<!-- Card -->
```

```
<div class="card card-cascade narrower">
```

```
<section>
```

```
<div class="card">
```

```
<div id="prtable" class="card-body" runat="server">
```

```
<asp:Table ID="Dtable" runat="server" border="1" CssClass="table table-striped
table-detail col-lg-6 align-items-sm-center" >
```

```
</asp:Table>
```

```
<!-- Table -->
```

```
<!-- Table -->
```

```
<hr class="my-0"/>
```

```
<!-- Bottom Table UI -->
```

```
</div>
```

```
</div>
```

```
</section>
```

```
<!-- Card -->
```

```
</div>
```

```
</section>
```

```
</div>
```

```
</asp:Content>
```

```
<asp:Content ID="Content3" ContentPlaceHolderID="ContentPlaceHolder1"  
Runat="Server">
```

```
</asp:Content>
```

```
***** VendorSizesAdd.aspx.cs*****
```

```
using System;  
using System.Collections.Generic;  
using System.Linq;  
using System.Web;  
using System.Web.UI;  
using System.Web.UI.WebControls;  
using DowBEntity;  
using DowBAL;  
using System.Xml;  
using System.Xml.Serialization;  
using System.IO;  
using System.Text;  
using System.Data.SqlClient;  
using System.Configuration;  
using System.Data;  
using System.Web.Security;
```

```
public partial class VendorSizesAdd : System.Web.UI.Page  
{
```

```
    string PageType;  
    int Id;  
    DDLogTracer objlog = new DDLogTracer();
```

```
protected void Page_Load(object sender, EventArgs e)
```

```
{  
    Id = Convert.ToInt16(Session["Pgid"].ToString());  
    CIsLoggedInUser CIsusr = (CIsLoggedInUser)Session["LoggedInUser_Specs"];  
    if (CIsusr != null)  
    {  
  
        string msg = Request.QueryString["msg"];  
        if (msg != null)  
        {  
            Common.ShowMessageHtml(msg, 1, true, dvMsgBox);  
        }  
    }  
}
```

```

        if (Id == 0)
        {
            PopDlIs ObjPD = new PopDlIs();
            ObjPD.BindDropDowns(ddl_productcat, (new PopDlIs()).GetExtProp());
            Bindetails(Id);
        }
        Session["Pgid"] = "1";

    }
    else
    {
        Response.Redirect("MLB_Login.aspx");
    }
}
private void Bindetails(int pid)
{
    pid = Convert.ToInt16(Session["Pgid"].ToString());

    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    List<EteProductGroup> Lst = new Products().GetProductExtPropts(pid); //user selected
colors
    List<EteSellerSizes> cnts = new
Products().GetSellerSizes(Convert.ToInt16(Clsusr.UserId));
    CheckBox chk = new CheckBox();
    Dtable.Rows.Clear();
    //try
    // {
    if (Lst != null && Lst.Count > 0)
    {
        TableRow tr = new TableRow();
        TableCell td = new TableCell();
        TableHeaderRow thr = new TableHeaderRow();
        td = new TableCell();
        chk.ID = "CheckBoxAll";
        chk.Text = "Select All";
        td.Controls.Add(chk);
        thr.Cells.Add(td);
        td = new TableCell();
        td = new TableCell();
        td.Text = "Size";
        thr.Cells.Add(td);
        thr.Cells.Add(td);
        td = new TableCell();
        td.Text = "Selected Date";
        thr.Cells.Add(td);

        Dtable.Rows.Add(thr);
        tr = new TableRow();
        td = new TableCell();

        StringBuilder sb = new StringBuilder();

        DateTime dts;

```

```

        foreach (var dt in Lst)
        {
            int pcount = (from a in cnts where a.EpOptId == dt.ProductId select a).Count();
            if (pcount > 0)
            {
                dts = (from b in cnts where b.EpOptId == dt.ProductId select
b.CreatedOn).FirstOrDefault(); }
                else { dts = DateTime.Today.Date; }
                tr = new TableRow();
                td = new TableCell();
                chk = new CheckBox();
                chk.ID = "CheckBox" + dt.ProductId;
                chk.Checked = true;
                if (pcount > 0)
                {
                    chk.Checked = true;
                }
                else
                {
                    chk.Checked = false;
                }
                td.Controls.Add(chk);
                tr.Cells.Add(td);
                td = new TableCell();
                td.Text = "<span class='badge badge-primary'>" + dt.ProductName + "</span>";
                tr.Cells.Add(td);
                td = new TableCell();
                td = new TableCell();
                td.Text = Convert.ToString(dts).Substring(0, 10);
                tr.Cells.Add(td);
                Dtable.Rows.Add(tr);
            }
        }
    }
}

```

```

protected void btn_save_Click(object sender, EventArgs e)
{
    bool ret = false;
    int grid = Convert.ToInt16(Session["Pgid"].ToString());
    string user = "";
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    if (Clsusr == null)
    {
        HttpCookie myCookie = Request.Cookies["DateOrWedCookie"];
        if (myCookie != null)
        {
            user = myCookie.Value.ToString();
        }
        else
        {
            user = "Guest";
        }
    }
}

```



```

        guest_login(user);
    }
    //try
    {
        Id = Convert.ToInt16(Session["Pgid"].ToString());

        CheckBox cbox = new CheckBox();
        cbox = (CheckBox)Dtable.FindControl("CheckBoxAll");
        List<EteProductGroup> Lst = new Products().GetProductExtPropts(Id); //user
selected colors
        if (cbox.Checked)
        {
            foreach (var list in Lst)
            {
                ret= update(list.pgid,true);
            }
        }
        else
        {
            foreach (var list in Lst)
            {
                cbox = (CheckBox)Dtable.FindControl("CheckBox" + list.ProductId);
                if (cbox.Checked)
                {
                    ret= update((int)list.ProductId,true);
                }
                else
                {
                    ret = update((int)list.ProductId, false);
                }
            }
        }
        Response.Redirect("~/Home.aspx?pageid=1&msg=Product Updated Sucessfully");
    }

    //catch (Exception exc)
    //{
    //    //objlog.Trace_Error(4, "DateOrWed", "MangeDealer", "Load", "Exception:" +
exc.ToString());
    //    Common.ShowMessageHtml(exc.ToString(), 0, true, dvMsgBox);
    //}

}
protected bool update(int prid,bool chk)
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];

    bool ret = new Products().UpdSellerSizes(prid, Convert.ToInt16(Clsusr.UserId), chk);
    return ret;
}
protected void guest_login(string user)
{

```

```

string City = "Hyderabad";// _City;
string Country = "India";// _Country;

BALCreateUser objlogin = new BALCreateUser();
DataTable status_login = objlogin.LoginGuest(user, user,
Request.ServerVariables["REMOTE_ADDR"].ToString(), "", City, Country);
string login_status = status_login.Rows[0]["UserStatus"].ToString();

ClsLoggedInUser objLogedInUser;
if (login_status == "True")
{
    objLogedInUser = new ClsLoggedInUser();

    objLogedInUser.UserId = status_login.Rows[0]["UserId"].ToString();
    objLogedInUser.UserName = status_login.Rows[0]["UserName"].ToString();
    objLogedInUser.GroupId = status_login.Rows[0]["GroupId"].ToString();
    objLogedInUser.LoginStatus = "True";
    objLogedInUser.LoginID = Convert.ToInt32(status_login.Rows[0]["LoginId"]);
    objLogedInUser.DealerId = Convert.ToInt32(status_login.Rows[0]["DealerId"]);
    objLogedInUser.RoleId = Convert.ToInt32(status_login.Rows[0]["RoleId"]);
    objLogedInUser.BrandId = 1; //Convert.ToInt32(status_login.Rows[0]["BrandId"]);

    Session["LoggedUser_Specs"] = objLogedInUser;

    FormsAuthentication.SetAuthCookie(objLogedInUser.UserName, false);
    HttpCookie myCookie = new HttpCookie("DateOrWedCookie");
    DateTime now = DateTime.Now;
    myCookie.Value = objLogedInUser.UserName;
    myCookie.Expires = now.AddYears(1);
    Response.Cookies.Add(myCookie);

}
}

protected void btn_retrieve_Click(object sender, EventArgs e)
{Session["Pgid"]=ddl_productcat.SelectedIndex;
    Bindetails(ddl_productcat.SelectedIndex);

}
protected void btn_clear_Click(object sender, EventArgs e)
{

}
protected void ddl_productcat_SelectedIndexChanged(object sender, EventArgs e)
{
    Session["Pgid"] = ddl_productcat.SelectedIndex;
    Bindetails(Convert.ToInt16(ddl_productcat.SelectedValue));

}
}
***** VendorSizesAddUser.aspx *****

```

```

<%@ Page Title="" Language="C#" MasterPageFile="~/ProductMaster.master"
AutoEventWireup="true" CodeFile="VendorSizesAddUser.aspx.cs"
Inherits="VendorSizesAddUser" %>
<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
Runat="Server">
</asp:Content>
<asp:Content ID="Content4" ContentPlaceHolderID="CPH_TopMessages" Runat="Server">
  <div class="col-lg-12 col-md-12 col-sm-12">
    <div class="row">
      <%--<h3>Ravikiran Techno Soft</h3>--%>
      <div id="dvMsgBox" class="span10 ordersave" runat="server" style="display:
block;">
<%--      <asp:Label ID="Label2" runat="server" Text=""
Height="100px"></asp:Label>--%>
      <asp:Label ID="Label3" runat="server" Text="" Height="60px"></asp:Label>
      </div>
    </div>
  </div>
  <div class="container-fluid">
    <div class="row-fluid" style="padding-top: 20px;">
      <div>
        <h5 class="my-4 dark-grey-text font-weight-bold" style="padding-left:20px">Manage
Selected Colors
        <asp:Button ID="btn_save" runat="server" CssClass="btn btn-secondary btn-md"
Text="Save" OnClick="btn_save_Click" /> <%-- btn-rounded--%>
        <asp:Button ID="btn_clear" runat="server" CssClass ="btn btn-secondary btn-md btn
rounded" Text="Clear" OnClick="btn_clear_Click" />
<%--      <a href="#" class="btn btn-secondary btn-md btn-rounded"> &nbsp;&nbsp;Edit</a> <%--
<i class="fas fa-edit"></i>
        <a href="#" class="btn btn-secondary btn-md btn-rounded"> &nbsp;&nbsp;Delete</a><%--<i
class="fas fa-trash"></i>--%>
        </h5>
      <!-- Navbar links -->

      </div>
    <div class="span10">
      <div class="row">
        <div id="Div1" class="span12 ordersave" runat="server" style="display: none;">
          <asp:Label ID="Label1" runat="server" Text=""></asp:Label>
        </div>
      </div>
    </div>
  </div>

  <%-- <div id="LeftSideBar" runat="server" class="span2">
    <asp:Table ID="Table1" runat="server">
      </asp:Table>
    </div>--%>
  </div>
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="CPH_MainContent" Runat="Server" >
  <div class="container-fluid" style="padding-left:100px; padding-right:50px">

  <!-- Navbar links -->
<section class="mb-5">

```

```

<!-- Card -->
<div class="card card-cascade narrower">
    <div class=" col-lg-8 col-md-8 col-sm-8 category-num">
        <table id="Table5" class="table table-striped table-detail col-lg-8"
runat="server">
            <tbody>
                <tr>
                    <td class="span3" dir="auto" style="width:270px">Product Sizes :
                </td>
                <td class="span3">
                    <asp:DropDownList ID="ddl_productcat" runat="server"
OnSelectedIndexChanged="ddl_productcat_SelectedIndexChanged" Visible="true"
Width="200px">
                        </asp:DropDownList>
                    </td>
                <td>
                    <asp:Button ID="Retrieve" runat="server" CssClass="btn btn-
secondary btn-md " OnClick="btn_retrieve_Click" Text="Retrieve Data" />
                </td>
                </tr>
            </tbody></table>
        </div>
    </div>
</section>

```

```

<section class="mb-5">
    <!-- Card -->
    <div class="card card-cascade narrower">
        <section>
            <div class="card">
                <div id="prtable" class="card-body" runat="server">
                    <asp:Table ID="Dtable" runat="server" border="1" CssClass="table table-striped
table-detail col-lg-6 align-items-sm-center" >
                </div>
            </div>
        </section>
    </div>
</asp:Table>

```

```

<!-- Table -->
<!-- Table -->

<hr class="my-0"/>

    <!-- Bottom Table UI -->

</div>
</div>

</section>

    <!-- Card -->
</div>
</section>

</div>
</asp:Content>
<asp:Content ID="Content3" ContentPlaceHolderID="ContentPlaceHolder1"
Runat="Server">

</asp:Content>

```

```

***** VendorSizesAddUser.aspx.cs*****

```

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBEntity;
using DowBAL;
using System.Xml;
using System.Xml.Serialization;
using System.IO;
using System.Text;
using System.Data.SqlClient;
using System.Configuration;
using System.Data;
using System.Web.Security;

```

```

public partial class VendorSizesAddUser : System.Web.UI.Page

```

```

{
    string PageType;
    int Id;
    DDLogTracer objlog = new DDLogTracer();

```

```

    protected void Page_Load(object sender, EventArgs e)

```

```

    {
        Id = Convert.ToInt16(Session["Pgid"].ToString());
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr != null)

```

```

{
    string msg = Request.QueryString["msg"];
    if (msg != null)
    {
        Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
    }

    if (Id == 0)
    {
        PopDtls ObjPD = new PopDtls();
        ObjPD.BindDropDowns(ddl_productcat, (new PopDtls()).GetExtProp());
        Bindetails(Id);
    }
    Session["Pgid"] = "1";

}
else
{
    Response.Redirect("MLB_Login.aspx");
}
}
private void Bindetails(int pid)
{
    pid = Convert.ToInt16(Session["Pgid"].ToString());

    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    List<EteProductGroup> Lst = new Products().GetProductExtPropts(pid); //user selected
colors
    List<EteSellerSizes> cnts = new
Products().GetSellerSizes(Convert.ToInt16(Session["VendorCode"]));
    CheckBox chk = new CheckBox();
    Dtable.Rows.Clear();
    //try
    // {
    if (Lst != null && Lst.Count > 0)
    {
        TableRow tr = new TableRow();
        TableCell td = new TableCell();
        TableHeaderRow thr = new TableHeaderRow();
        td = new TableCell();
        chk.ID = "CheckBoxAll";
        chk.Text = "Select All";
        td.Controls.Add(chk);
        thr.Cells.Add(td);
        td = new TableCell();
        td = new TableCell();
        td.Text = "Size";
        thr.Cells.Add(td);
        thr.Cells.Add(td);
        td = new TableCell();
        td.Text = "Selected Date";
        thr.Cells.Add(td);

        Dtable.Rows.Add(thr);
    }
}

```

```

tr = new TableRow();
td = new TableCell();

StringBuilder sb = new StringBuilder();

DateTime dts;

foreach (var dt in Lst)
{
    int pcount = (from a in cnts where a.EpOptId == dt.ProductId select a).Count();
    if (pcount > 0)
        { dts = (from b in cnts where b.EpOptId == dt.ProductId select
b.CreatedOn).FirstOrDefault(); }
    else { dts = DateTime.Today.Date; }
    tr = new TableRow();
    td = new TableCell();
    chk = new CheckBox();
    chk.ID = "CheckBox" + dt.ProductId;
    chk.Checked = true;
    if (pcount > 0)
    {
        chk.Checked = true;
    }
    else
    {
        chk.Checked = false;
    }
    td.Controls.Add(chk);
    tr.Cells.Add(td);
    td = new TableCell();
    td.Text = "<span class='badge badge-primary'>" + dt.ProductName + "</span>";
    tr.Cells.Add(td);
    td = new TableCell();
    td = new TableCell();
    td.Text = Convert.ToString(dts).Substring(0, 10);
    tr.Cells.Add(td);
    Dtable.Rows.Add(tr);
}
}
}

```

```

protected void btn_save_Click(object sender, EventArgs e)
{
    bool ret = false;
    int grid = Convert.ToInt16(Session["Pgid"].ToString());
    string user = "";
    CIsLoggedInUser CIsusr = (CIsLoggedInUser)Session["LoggedInUser_Specs"];
    if (CIsusr == null)
    {
        HttpCookie myCookie = Request.Cookies["DateOrWedCookie"];
        if (myCookie != null)

```

```

    {
        user = myCookie.Value.ToString();
    }
    else
    {
        user = "Guest";
    }
    guest_login(user);
}
//try
{
    Id = Convert.ToInt16(Session["Pgid"].ToString());

    CheckBox cbox = new CheckBox();
    cbox = (CheckBox)Dtable.FindControl("CheckBoxAll");
    List<EteProductGroup> Lst = new Products().GetProductExtPropts(Id); //user
selected colors
    if (cbox.Checked)
    {
        foreach (var list in Lst)
        {
            ret= update(list.pgid,true);
        }
    }
    else
    {
        foreach (var list in Lst)
        {
            cbox = (CheckBox)Dtable.FindControl("CheckBox" + list.ProductId);
            if (cbox.Checked)
            {
                ret= update((int)list.ProductId,true);
            }
            else
            {
                ret = update((int)list.ProductId, false);
            }
        }
    }
    Response.Redirect("~/Home.aspx?pageid=1&msg=Product Updated Sucessfully");
}

//catch (Exception exc)
//{{
// //objlog.Trace_Error(4, "DateOrWed", "MangeDealer", "Load", "Exception:" +
exc.ToString());
// Common.ShowMessageHtml(exc.ToString(), 0, true, dvMsgBox);

//}}

}
protected bool update(int prid,bool chk)
{

```



```

ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];

bool ret = new Products().UpdSellerSizes(prid,
Convert.ToInt16(Session["VendorCode"]), chk);
return ret;
}
protected void guest_login(string user)
{

string City = "Hyderabad";// _City;
string Country = "India";// _Country;

BALCreateUser objlogin = new BALCreateUser();
DataTable status_login = objlogin.LoginGuest(user, user,
Request.ServerVariables["REMOTE_ADDR"].ToString(), "", City, Country);
string login_status = status_login.Rows[0]["UserStatus"].ToString();

ClsLoggedInUser objLoggedInUser;
if (login_status == "True")
{
objLoggedInUser = new ClsLoggedInUser();

objLoggedInUser.UserId = status_login.Rows[0]["UserId"].ToString();
objLoggedInUser.UserName = status_login.Rows[0]["UserName"].ToString();
objLoggedInUser.GroupId = status_login.Rows[0]["GroupId"].ToString();
objLoggedInUser.LoginStatus = "True";
objLoggedInUser.LoginID = Convert.ToInt32(status_login.Rows[0]["LoginId"]);
objLoggedInUser.DealerId = Convert.ToInt32(status_login.Rows[0]["DealerId"]);
objLoggedInUser.RoleId = Convert.ToInt32(status_login.Rows[0]["RoleId"]);
objLoggedInUser.BrandId = 1; //Convert.ToInt32(status_login.Rows[0]["BrandId"]);

Session["LoggedInUser_Specs"] = objLoggedInUser;

FormsAuthentication.SetAuthCookie(objLoggedInUser.UserName, false);
HttpCookie myCookie = new HttpCookie("DateOrWedCookie");
DateTime now = DateTime.Now;
myCookie.Value = objLoggedInUser.UserName;
myCookie.Expires = now.AddYears(1);
Response.Cookies.Add(myCookie);

}
}

protected void btn_retrieve_Click(object sender, EventArgs e)
{Session["Pgid"]=ddl_productcat.SelectedIndex;
Bindetails(ddl_productcat.SelectedIndex);

}
protected void btn_clear_Click(object sender, EventArgs e)
{

}
}

```

```

protected void ddl_productcat_SelectedIndexChanged(object sender, EventArgs e)
{
    Session["Pgid"] = ddl_productcat.SelectedIndex;
    Bindetails(Convert.ToInt16(ddl_productcat.SelectedValue));
}
}
}
*****ViewSearchResult.aspx*****
<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="ViewSearchResult.aspx.cs"
Inherits="ViewSearchResult" %>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">
    <script src="js/Custom.js" type="text/javascript"></script>

<script type="text/javascript">

    $(window).load(function () {
        activeTab = getCookie('seltab');
        //alert("." + activeTab);
        $(". " + activeTab).trigger('click');
        setCookie('seltab', "", 2);
    })

    function saveCookie(ckname) {
        setCookie('seltab', ckname, 2);
        //alert(ckname);
    }

    function setCookie(cname, cvalue, exdays) {
        var d = new Date();
        d.setTime(d.getTime() + (exdays * 24 * 60 * 60 * 1000));
        var expires = "expires=" + d.toUTCString();
        document.cookie = cname + "=" + cvalue + ";" + expires;
    }

    function getCookie(cname) {
        var name = cname + "=";
        var ca = document.cookie.split(';');
        for (var i = 0; i < ca.length; i++) {
            var c = ca[i];
            while (c.charAt(0) == ' ') c = c.substring(1);
            if (c.indexOf(name) == 0) return c.substring(name.length, c.length);
        }
        return "";
    }

</script>
</asp:Content>

<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
    <div class="container-fluid">
        <div class="row-fluid" style="padding-top: 20px;">

```



```
</div>
</div>
</asp:Content>
```

```
***** ViewSearchResult.aspx.cs*****
```

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBEntity;
using DowBAL;
using System.Data;
using DowDAL;
using System.Configuration;
using System.Data.SqlClient;
```

```
public partial class ViewSearchResult : System.Web.UI.Page
{
    string Msg = string.Empty;
    protected void Page_Load(object sender, EventArgs e)
    {
        //if (!IsPostBack)
        //{
            //ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
            //if (Clsusr == null)
            //{
                // Response.Redirect("MLB_Login.aspx");
            //}

            string pageid = Request.QueryString["pageid"];
            string SrchStrng =
HttpUtility.UrlDecode(Request.QueryString["SrchStrng"]);//HttpUtility.UrlDecode(Request.Qu
eryString["id"]
            string refrulr = HttpUtility.UrlDecode(Request.QueryString["fullURL"]);
            string loc = Request.QueryString["loc"];
            string cat = Request.QueryString["cat"];

            if (cat != null && cat != "")
            {
                Response.Redirect("~/ShoppingList.aspx?cid=" + cat + "&SrchStrng=" + SrchStrng);
            }
            SrchStrng = SrchStrng.TrimStart().TrimEnd();
            if (SrchStrng == "" || SrchStrng==null) { SrchStrng = "all"; }
            if (SrchStrng == "" || SrchStrng.Length < 0)
            {
                if (refrulr.Replace(" ", "") == "" || refrulr.Replace(" ", "").Length < 0)
                {
                    Response.Redirect("Home.aspx?pageid=1");
                }
            }
            else
            {

```

```

        Response.Redirect(refrurl);

    }
}
if (loc!=null)
{
    Response.Redirect("~/ShoppingList.aspx?SrchStrng=" + SrchStrng+"&loc="+loc);
}
{
    Response.Redirect("~/ShoppingList.aspx?SrchStrng=" + SrchStrng);
    //ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
Convert.ToInt32(Clsusr.GroupId));
    //BindProductList(SrchStrng, Convert.ToInt32(Clsusr.DealerId));
    //BindBuyLeadsList(SrchStrng, Convert.ToInt32(Clsusr.DealerId));
    //BindUserList(SrchStrng);
}
//Master txt_SearchString.text = "";

if (Msg.Length>0)
{ Common.ShowMessageHtml(Msg, 1, true, dvMsgBox); }
//}
}

private void BindProductList(string SearchString, int DealerID)
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    IList<DowDAL.GetSearchFinalProductsResult> Lst = (new
Products()).GetSearchFinalProducts(SearchString);
    foreach (var lst in Lst)
    {

    }
    if (Lst != null && Lst.Count > 0)
    {

        // if (WdGrid_ProductList.HasDataSource == true) {
WdGrid_ProductList.ClearDataSource(); }
        // WdGrid_ProductList.DataSource = Lst;
        // WdGrid_ProductList.DataBind();
        // WdGrid_ProductList.Visible = true;
        // Msg += "(" + Lst.Count.ToString() + ") Products Matched. ";
    }
    else
    {
        //Common.ShowMessageHtml("No records found", 0, true, dvMsgBox);
        Msg += "(0) PRODUCTS MATCHED. ";
        //WdGrid_ProductList.Visible = false;
    }
}
}
public class ImageList : List<CustomImage>
{
    public static ImageList GetList()
    {
        ImageList il = new ImageList();
        for (int i = 1; i < 5; i++)
    }
}

```



```

}

// This is Event Handler
protected void myBtnClickHandler(object sender, EventArgs e)
{
    LinkButton lnk = (LinkButton)sender;
    string ID = lnk.ID;

    DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));
    if (ds != null)
    {
        if (ds.Tables[0].Rows.Count > 0)
        {
            DataTable dtdll = ds.Tables[0];
            foreach (DataRow drnew in dtdll.Rows)
            {
                switch (drnew["lbtncode"].ToString())
                {
                    case "SAVE":
                        // Call SAVE Method
                        break;
                    case "URL":
                        Response.Redirect(drnew["pageurl"].ToString());
                        break;
                    case "AUDIT":
                        Response.Redirect(drnew["pageurl"].ToString() + "&frompage=" +
Request.QueryString["pageid"]);
                        break;

                    case "CANCEL":
                        // Call Cancel Method
                        break;
                    default:
                        break;
                }
            }
        }
    }
}

private void BindUserList(string SearchString)
{
    try
    {
        List<EtetblUser> Lst = (new tblUsers()).getsearchuser(SearchString);

        if (Lst != null && Lst.Count > 0)
        {
            //WDGrid_OderList.DataSource = Lst;
        }
    }
}

```

```

        //WDGrid_OderList.DataBind();
        //WDGrid_OderList.Visible = true;
        //Msg += "Customer found for given input, ";
        Msg += "(" + Lst.Count.ToString() + ") CUSTOMER MATCHED. ";
    }
    else
    {
        //Msg += " There is no customer found for given inputs, ";
        Msg += "(0) CUSTOMER MATCHED. ";
        //WDGrid_OderList.Visible = false;
    }
}
catch {}
}
protected override void OnInit(EventArgs e)
{
    base.OnInit(e);

    if (this.Master is MainMaster)
    {
        ((MainMaster)this.Master).MasterControlClicked += new
EventHandler(ContentControlClicked);
    }
}
private void ContentControlClicked(object sender, EventArgs e)
{
    string pageid = "pageid=" + Session["pageid"].ToString();
    Response.Redirect("~/Managedealer.aspx?PageType=Add&dealername=0&" +
pageid);
}

protected void Lnkbtn_ManageOrder_Click(object sender, EventArgs e)
{
    LinkButton lnkbutton = (LinkButton)sender;
    string strcustid = lnkbutton.CommandName;

    if (strcustid.Length > 0)
    {
        string[] stringSeparators = new string[] { "&", "=" };
        string[] arr1;
        arr1 = strcustid.Split(stringSeparators, StringSplitOptions.RemoveEmptyEntries);
        bool bchkstatus = false;
        Response.Redirect("ManageUser.aspx?pageid=163&PageType=View&" + strcustid);
        //foreach (var word in arr1)
        //{
        //    if (word == "status")
        //    {
        //        bchkstatus = true;
        //    }

        //    if (bchkstatus == true && word != "status")
        //    {
        //        StatusWiseRedirect(Convert.ToInt32(word), strcustid);
        //    }
        //}
    }
}

```



```

        //}
    }
}

private void StatusWiseRedirect(int statusid, string param)
{
    switch (statusid.ToString())
    {
        case "1":
            Response.Redirect("ManageOrder.aspx?pageid=177&PageType=Edit&" +
param);
            break;
        case "2":
            Response.Redirect("ManageOrder.aspx?pageid=177&PageType=Edit&" +
param);
            break;
        case "3":
            Response.Redirect("ManageOrder.aspx?pageid=177&PageType=Edit&" +
param);
            break;
        case "17":
            Response.Redirect("ManageOrder.aspx?pageid=179&PageType=Edit&" +
param);
            break;
        case "9":
            Response.Redirect("ManageOrder.aspx?pageid=179&PageType=Edit&" +
param);
            break;
        case "11":
            Response.Redirect("ManageOrder.aspx?pageid=204&PageType=Edit&" +
param);
            break;

        default:
            Response.Redirect("ManageOrder.aspx?pageid=179&PageType=Edit&" +
param);
            break;
    }
}
}
}

```

*****ViewServiceResult.aspx *****

```

<%@ Page Title="" Language="C#" MasterPageFile="~/MainMaster.master"
AutoEventWireup="true" CodeFile="ViewServiceResult.aspx.cs"
Inherits="ViewServiceResult" %>

```

```

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header"
runat="Server">

```

```

    <script src="js/Custom.js" type="text/javascript"></script>

```

```

<script type="text/javascript">

```

```

$(window).load(function () {
    activeTab = getCookie('seltab');
    //alert("." + activeTab);
    $(". " + activeTab).trigger('click');
    setCookie('seltab', " , 2);
})

function saveCookie(ckname) {
    setCookie('seltab', ckname, 2);
    //alert(ckname);
}

function setCookie(cname, cvalue, exdays) {
    var d = new Date();
    d.setTime(d.getTime() + (exdays * 24 * 60 * 60 * 1000));
    var expires = "expires=" + d.toUTCString();
    document.cookie = cname + "=" + cvalue + "; " + expires;
}

function getCookie(cname) {
    var name = cname + "=";
    var ca = document.cookie.split(';');
    for (var i = 0; i < ca.length; i++) {
        var c = ca[i];
        while (c.charAt(0) == ' ') c = c.substring(1);
        if (c.indexOf(name) == 0) return c.substring(name.length, c.length);
    }
    return "";
}

```

</script>

</asp:Content>

<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" runat="Server">

<div class="container-fluid">

<div class="row-fluid" style="padding-top: 20px;">

<div class="span2" id="dvlogo" runat="server" style="display: none;">

<%----%>

</div>

<div class="span10">

<div class="row">

<div id="dvMsgBox" class="span12 ordersave" runat="server" style="display: none;">

<asp:Label ID="Label1" runat="server" Text="Label"></asp:Label>

</div>

</div>

</div>

</div>

</div>

</asp:Content>

<asp:Content ID="Content4" ContentPlaceHolderID="CPH_LeftSideBar" runat="Server">

<div id="LeftSideBar" runat="server" class="span2">

<asp:Table ID="Table1" runat="server">

```

        </asp:Table>
    </div>
</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="CPH_MainContent" runat="Server">

<div class="span10" id="MainContent" runat="server">
    <div id="ShowPageHeader" class="row" style="display: block;" runat="server">
        <div id="PageName" class="span12 customer" runat="server">
            </div>
        </div>

        <div id="Div1" class="row" style="display: block;">
            <div id="Div2" class="span12 customertable">

                <div class="tabs_holder">
                    <ul id="tabs">

                        <li id="tab1" class="tab_selected span2"><a href="#your-tab-id-1"
onclick="saveCookie('tb1')" class="tb1"><i class="fa fa-th-large"></i>&nbsp;&nbsp;&nbsp;PRODUCT
LIST</a></li>
                        <li id="tab2"><a href="#your-tab-id-2" onclick="saveCookie('tb2')"
class="tb2"><i class="fa fa-bank"></i>&nbsp;&nbsp;&nbsp;BUY OFFERS LIST</a></li>
                        <li id="tab3"><a href="#your-tab-id-3" onclick="saveCookie('tb3')"
class="tb3"><i class="fa fa-bars"></i>&nbsp;&nbsp;&nbsp;BUYERS/SELLERS LIST</a></li>

                    </ul>
                    <div class="content_holder">
                        <div id="your-tab-id-1">

                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>
</asp:Content>

```

```

*****ViewServiceResult.aspx.cs *****
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DowBEntity;
using DowBAL;
using System.Data;
using DowDAL;
using System.Configuration;
using System.Data.SqlClient;

```

```

public partial class ViewServiceResult : System.Web.UI.Page
{
    string Msg = string.Empty;
    protected void Page_Load(object sender, EventArgs e)
    {
        //if (!IsPostBack)
        //{
            //ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
            //if (Clsusr == null)
            //{
                // Response.Redirect("MLB_Login.aspx");
            //}

            string pageid = Request.QueryString["pageid"];
            string SrchStrng = HttpUtility.UrlDecode(Request.QueryString["SrchStrng"]);
            string refrulr = HttpUtility.UrlDecode(Request.QueryString["fullURL"]);
            string loc = Request.QueryString["loc"];
            SrchStrng = SrchStrng.TrimStart().TrimEnd();
            if (SrchStrng == "" || SrchStrng.Length < 0)
            {
                if (refrulr.Replace(" ", "") == "" || refrulr.Replace(" ", "").Length < 0)
                {
                    Response.Redirect("Services.aspx?pageid=1");
                }
                else
                {
                    Response.Redirect(refrulr);
                }
            }
        }
        if (loc == "1")
        {
            Response.Redirect("~/ServicesList.aspx?SrchStrng=" + SrchStrng);
        }
        else
        {
            Response.Redirect("~/ServicesSearch.aspx?SrchStrng=" + SrchStrng);
            //ManageNavigations(Convert.ToInt32(pageid), Convert.ToInt32(Clsusr.UserId),
            Convert.ToInt32(Clsusr.GroupId));
            //BindProductList(SrchStrng, Convert.ToInt32(Clsusr.DealerId));
            //BindBuyLeadsList(SrchStrng, Convert.ToInt32(Clsusr.DealerId));
            //BindUserList(SrchStrng);
        }
        //Master txt_SearchString.text = "";

        if (Msg.Length>0)
        {
            Common.ShowMessageHtml(Msg, 1, true, dvMsgBox);
        }
    }

    private void BindProductList(string SearchString, int DealerID)
    {
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        IList<DowDAL.GetSearchFinalProductsResult> Lst = (new
        Products()).GetSearchFinalProducts(SearchString);
    }
}

```

```

        foreach (var lst in Lst)
        {

        }
        if (Lst != null && Lst.Count > 0)
        {

            // if (WdGrid_ProductList.HasDataSource == true) {
WdGrid_ProductList.ClearDataSource(); }
            // WdGrid_ProductList.DataSource = Lst;
            // WdGrid_ProductList.DataBind();
            // WdGrid_ProductList.Visible = true;
            // Msg += "(" + Lst.Count.ToString() + ") Products Matched. ";
        }
        else
        {
            //Common.ShowMessageHtml("No records found", 0, true, dvMsgBox);
            Msg += "(0) PRODUCTS MATCHED. ";
            //WdGrid_ProductList.Visible = false;
        }
    }
}
public class ImageList : List<CustomImage>
{
    public static ImageList GetList()
    {
        ImageList il = new ImageList();
        for (int i = 1; i < 5; i++)
        {
            il.Add(new CustomImage { Id = i, iUrl = "Image.ashx?id=" + i, Title = "test image"
});
        }
        return il;
    }
}
private void BindBuyLeadsList(string SearchString, int DealerID)
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];

    //try
    //{
    //    IList<EteCustomerOrder> Lst = (new
CustomerOrders()).GetSearchCustomerOrders("NewOrder", 1, Clsusr.RoleId,
Clsusr.DealerId, Convert.ToInt32(Clsusr.UserId), SearchString);
    //    if (Lst != null && Lst.Count > 0)
    //    {
    //        WdGrid_BuyLeadsList.DataSource = Lst;
    //        WdGrid_BuyLeadsList.DataBind();
    //        WdGrid_BuyLeadsList.Visible = true;
    //        //Msg += "Draft Order found for given input, ";
    //        Msg += "(" + Lst.Count.ToString() + ") DRAFT MATCHED. ";
    //    }
    //    else
    //    {
    //        //Common.ShowMessageHtml("No Orders for this user ", 0, true, dvMsgBox);
    //        //Msg += "There is no Draft Order found for given input, ";

```

```

//     Msg += "(0) DRAFT MATCHED. ";
//     WDGrid_BuyLeadsList.Visible = false;
// }
}
private void ManageNavigations(int pageid, int userid, int groupid)
{
    bool IsLeftBar = false;
    IsLeftBar = Navigations.Generate_LeftNaveBar(pageid, userid, groupid, Table1,
myBtnClickHandler);

    if (IsLeftBar == true)
    {
        LeftSideBar.Attributes["class"] = "span2";
        MainContent.Attributes["class"] = "span10";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;"); //style="display:
block;"
        PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
    else
    {
        LeftSideBar.Attributes["class"] = "";
        MainContent.Attributes["class"] = "span12";
        ShowPageHeader.Attributes.CssStyle.Add("display", "block;");
        PageName.InnerText = new
tblPagess().GettblPageNamebyId(Convert.ToInt32(pageid));
    }
}

```

// This is Event Handler

```

protected void myBtnClickHandler(object sender, EventArgs e)
{

```

```

    LinkButton Ink = (LinkButton)sender;
    string ID = Ink.ID;

```

```

    DataSet ds = Navigations.GetPageButtonDetails(Convert.ToInt32(ID));

```

```

    if (ds != null)

```

```

    {

```

```

        if (ds.Tables[0].Rows.Count > 0)

```

```

        {

```

```

            DataTable dtdll = ds.Tables[0];

```

```

            foreach (DataRow drnew in dtdll.Rows)

```

```

            {

```

```

                switch (drnew["lbtncode"].ToString())

```

```

                {

```

```

                    case "SAVE":

```

```

                        // Call SAVE Method

```

```

                        break;

```

```

                    case "URL":

```

```

                        Response.Redirect(drnew["pageurl"].ToString());

```

```

                        break;

```

```

                    case "AUDIT":

```

```
        Response.Redirect(drnew["pageurl"].ToString() + "&frompage=" +  
Request.QueryString["pageid"]);  
        break;
```

```
        case "CANCEL":  
            // Call Cancel Method  
            break;  
        default:
```

```
            break;
```

```
    }
```

```
    }
```

```
    }
```

```
    }
```

```
}
```

```
private void BindUserList(string SearchString)
```

```
{
```

```
    try
```

```
    {
```

```
        List<EtetblUser> Lst = (new tblUsers()).getsearchuser(SearchString);
```

```
        if (Lst != null && Lst.Count > 0)
```

```
        {
```

```
            //WDGrid_OderList.DataSource = Lst;
```

```
            //WDGrid_OderList.DataBind();
```

```
            //WDGrid_OderList.Visible = true;
```

```
            //Msg += "Customer found for given input, ";
```

```
            Msg += "(" + Lst.Count.ToString() + ") CUSTOMER MATCHED. ";
```

```
        }
```

```
        else
```

```
        {
```

```
            //Msg += " There is no customer found for given inputs, ";
```

```
            Msg += "(0) CUSTOMER MATCHED. ";
```

```
            //WDGrid_OderList.Visible = false;
```

```
        }
```

```
    }
```

```
    catch { }
```

```
}
```

```
protected override void OnInit(EventArgs e)
```

```
{
```

```
    base.OnInit(e);
```

```
    if (this.Master is MainMaster)
```

```
    {
```

```
        ((MainMaster)this.Master).MasterControlClicked += new
```

```
EventHandler(ContentControlClicked);
```

```
    }
```

```
}
```

```
private void ContentControlClicked(object sender, EventArgs e)
```

```
{
```

```

        string pageid = "pageid=" + Session["pageid"].ToString();
        Response.Redirect("~/Managedealer.aspx?PageType=Add&dealername=0&" +
pageid);
    }

protected void Lnkbtn_ManageOrder_Click(object sender, EventArgs e)
{
    LinkButton lnkbutton = (LinkButton)sender;
    string strcustid = lnkbutton.CommandName;

    if (strcustid.Length > 0)
    {
        string[] stringSeparators = new string[] { "&", "=" };
        string[] arr1;
        arr1 = strcustid.Split(stringSeparators, StringSplitOptions.RemoveEmptyEntries);
        bool bchkstatus = false;
        Response.Redirect("ManageUser.aspx?pageid=163&PageType=View&" + strcustid);
        //foreach (var word in arr1)
        //{
        //    if (word == "status")
        //    {
        //        bchkstatus = true;
        //    }

        //    if (bchkstatus == true && word != "status")
        //    {
        //        StatusWiseRedirect(Convert.ToInt32(word), strcustid);
        //    }

        //}
    }
}

private void StatusWiseRedirect(int statusid, string param)
{
    switch (statusid.ToString())
    {
        case "1":
            Response.Redirect("ManageOrder.aspx?pageid=177&PageType=Edit&" +
param);
            break;
        case "2":
            Response.Redirect("ManageOrder.aspx?pageid=177&PageType=Edit&" +
param);
            break;
        case "3":
            Response.Redirect("ManageOrder.aspx?pageid=177&PageType=Edit&" +
param);
            break;
        case "17":
            Response.Redirect("ManageOrder.aspx?pageid=179&PageType=Edit&" +
param);
            break;
    }
}

```



```

        case "9":
            Response.Redirect("ManageOrder.aspx?pageid=179&PageType=Edit&" +
param);
            break;
        case "11":
            Response.Redirect("ManageOrder.aspx?pageid=204&PageType=Edit&" +
param);
            break;

        default:
            Response.Redirect("ManageOrder.aspx?pageid=179&PageType=Edit&" +
param);
            break;
    }
}
}
}

```

*****web.config*****

```

<?xml version="1.0"?>
<!-- MLBdb : Currently working Database.-->
<configuration>
  <configSections>
    <section name="infragistics.web"
type="System.Configuration.SingleTagSectionHandler,System, Version=1.0.3300.0,
Culture=neutral, PublicKeyToken=b77a5c561934e089"/>
  </configSections>
  <infragistics.web styleSetName="Default" styleSetPath="~/ig_res"
enableAppStyling="true"/>
  <system.webServer>
    <validation validateIntegratedModeConfiguration="false"/>
    <defaultDocument>
      <files>
        <add value="Home.aspx"/>
      </files>
    </defaultDocument>
    <handlers>
      <remove name="ExtensionlessUrlHandler-Integrated-4.0"/>
      <remove name="OPTIONSVerbHandler"/>
      <remove name="TRACEVerbHandler"/>
      <add name="ExtensionlessUrlHandler-Integrated-4.0" path="*" verb="*"
type="System.Web.Handlers.TransferRequestHandler"
preCondition="integratedMode,runtimeVersionv4.0"/>
    </handlers>
  </system.webServer>
  <system.web>
    <customErrors mode="Off" defaultRedirect="DefaultRedirectErrorPage.aspx">
      <error statusCode="404" redirect="Http404ErrorPage.aspx"/>
    </customErrors>
    <compilation debug="true" targetFramework="4.5">
      <assemblies>
        <add assembly="System.Windows.Forms, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=B77A5C561934E089"/>
        <add assembly="System.Design, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=B03F5F7F11D50A3A"/>
        <add assembly="System.Data.Linq, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=B77A5C561934E089"/>

```

```

    <add assembly="Microsoft.Build.Framework, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=B03F5F7F11D50A3A"/>
    <add assembly="System.Net.Http, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=B03F5F7F11D50A3A"/>
  </assemblies>
</compilation>
<httpRuntime targetFramework="4.5"/>
<authentication mode="Forms">
  <forms loginUrl="MLB_Login.aspx" name="DateOrWed.com" timeout="9600"/>
</authentication>
<sessionState timeout="49600"/>
<webServices>
  <protocols>
    <add name="HttpGet"/>
    <add name="HttpPost"/>
  </protocols>
</webServices>
<httpHandlers>
  <add path="*.asmx" verb="*" type="System.Web.Script.Services.ScriptHandlerFactory"
validate="false"/>
</httpHandlers>
<identity impersonate="true"/>
</system.web>
<!--<appSettings></appSettings-->
<appSettings>
  <!--<add key="DowConectingString" value="Data Source=SQL5042.site4now.net;Initial
Catalog=DB_9FAA34_mlbDB;Persist Security Info=True;User
ID=DB_9FAA34_mlbDB_admin;Password=mymlb123;Timeout=2200" />-->
  <add key="DOWConectingString" value="Data
Source=tcp:panindiasqlserver.database.windows.net,1433;Initial
Catalog=DOWShopping;Persist Security Info=false;User
ID=dowadmin;Password=ARaks@1204;Timeout=2200"/>
  <!--*****logconfig parameters*****-->
  <add key="filesize" value="2"/>
  <add key="FileSizeFormat" value="mb"/>
  <add key="strLogFileName" value="log"/>
  <add key="strlogpath" value="\Log"/>
  <add key="strErrorFileName" value="error"/>
  <add key="HtmlTemplatePath" value="~\EmailTemplates"/>
  <add key="WkHtmlToPdfRunner" value="~\WkHtmlToPdf"/>
  <add key="WkHtmlToPdfPath" value="~\ThirdParty\WkHtmlToPdf\wkhtmltopdf.exe"/>
  <add key="pdfOutPut" value="~\Output"/>
  <add key="ImageOutPut" value="~\Images"/>
  <!--INFO = 0,  DEBUG = 1,  DB = 2,  DETAILED=3,  PANIC = 4,  EXC = 5-->
  <add key="level0" value="1"/>
  <add key="level1" value="1"/>
  <add key="level2" value="1"/>
  <add key="level3" value="1"/>
  <add key="level4" value="1"/>
  <add key="level5" value="1"/>
  <!--*****logconfig parameters*****-->
  <add key="SSRSPrimaryURL"
value="http://remote.DateOrWed.com:61500/ReportServer"/>
  <add key="CoolingOffAfter" value="5"/>
  <!--Adds No. of days in current data for Cooling Off-->

```

```

<add key="SmtpClientServer" value="smtp.secureserver.net"/>
<!--mail.eze-talk.com-->
<add key="mailFrom" value="admin@dateorwed.com"/>
<!--provisioning@eze-talk.com-->
<add key="SmtpPort" value="465"/>
<!--25-->
<add key="SmtpServerUser" value="Admin"/>
<!--ezetalk\\Synverse-->
<add key="SmtpServerPassword" value="DRaks@1204"/>
<!--aK4Hhi4M8W9V-->
<!--Email Server Configuration Ends-->
</appSettings>
<connectionStrings>
  <add name="DOWConectingString" connectionString="Data
Source=tc:panindiasqlserver.database.windows.net,1433;Initial
Catalog=dowshopping;Persist Security Info=false;User
ID=dowadmin;Password=ARaks@1204" providerName="System.Data.SqlClient"/>
  <!--<add name="DowConectingString" connectionString="Data
Source=SQL5042.site4now.net;Initial Catalog=DB_9FAA34_mlbDB;Persist Security
Info=True;User ID=DB_9FAA34_mlbDB_admin;Password=mymlb123"
providerName="System.Data.SqlClient" />-->
</connectionStrings>
<runtime>
  <assemblyBinding xmlns="urn:schemas-microsoft-com:asm.v1">
    <dependentAssembly>
      <assemblyIdentity name="Newtonsoft.Json" publicKeyToken="30ad4fe6b2a6aeed"
culture="neutral"/>
      <bindingRedirect oldVersion="0.0.0.0-4.5.0.0" newVersion="4.5.0.0"/>
    </dependentAssembly>
  </assemblyBinding>
</runtime>
<!--GroupId GroupName GroupPage
1 Admin AdminHome.aspx
2 Guest Home.aspx
3 Staff StaffHome.aspx
4 Vendor VendorHome.aspx
5 Buyer BuyerHome.aspx
6 ShopVendor ShoppingHome.aspx
7 ServVendor ServiceHome.aspx
8 BusVendor BussinessHome.aspx-->
</configuration>

```

***** Conclusion *****

This concludes the application development process.
This is the link to css files used in the application, copy to application folder
<https://www.dropbox.com/scl/fi/md3q5vsxxiosbwsyvhp5w/DowShopping.zip?rlkey=7bhauvijyc0uyqtuwg4wtk4d&st=yybnqi5y&dl=0>

