



DEVELOP SOFTWARE WITH C#

Train with Ravi [DateOrWed Application Development]

RAVI KUMAR PARAMKUSAM

Application Development with C#

Train with Ravi

Development of C# Application DateOrWed.com

Preface

I developed DateOrWed.com some two years back. The Idea struck me when I wanted to develop a matrimonial website. Then I thought why don't I develop a website to cater to the Needs of both Dating and Matrimonial searching.

During the development, I saw a lot of websites getting hacked and their customer data being stolen and released in the open. From that, I got the idea to encrypt the profile data of the users to shut down the possibility of data being hacked.

Finally, I started developing the website DateOrWed.com I had to wait for a few months for the availability of the website.

After two years of hard work, the website took shape to be brought live.

So, read on.

Chapter 1

The Idea:

To develop a web application for catering to the needs of those who want to Date and/or Wed.

Requirements of the Application:

The data is to be completely encrypted.

To be developed in N Tier.

Users can hide their profile data and show the required parts to other users.

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.

<https://www.dropbox.com/scl/fi/5e65q73krs4yqzrp9wrif/dowscript.sql?rlkey=q70keugo14zuxrwr651nftg8k&st=9yrefbyz&dl=0>

```
/****** Object: Table [dbo].[tblPages] Script Date: 04-09-2024 21:58:12 *****/
```

```
SET ANSI_NULLS ON
```

```
GO
```

```
SET QUOTED_IDENTIFIER ON
```

```
GO
```

```
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]
```

```
GO
```

```
/****** Object: Table [dbo].[tblPermissionGroup] Script Date: 04-09-2024 21:58:12 *****/
```

```
SET ANSI_NULLS ON
```

```
GO
```

```
SET QUOTED_IDENTIFIER ON
```

```
GO
```

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

```
    [GroupId] [int] NOT NULL,
```

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

```
    [Status] [bit] NULL,
```

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

```
    [RoleId] [int] NULL,
```

```
    [GroupPage] [varchar](50) 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]
GO
/***** Object: View [dbo].[tblGrpMapping]  Script Date: 04-09-2024 21:58:12 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO

CREATE VIEW [dbo].[tblGrpMapping] AS
SELECT    TP.Pageld + tPG.GroupId AS MapId, tPG.GroupId, TP.Pageld, 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
GO
/***** Object: View [dbo].[v_tblGrpMapping]  Script Date: 04-09-2024 21:58:12 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO

CREATE VIEW [dbo].[v_tblGrpMapping]
AS
SELECT    m.MapId, m.GroupId, g.GroupName, m.Pageld, 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.Pageld = p.Pageld
where p.Pageld=1

GO
/***** Object: Table [dbo].[tblLeftButtons]  Script Date: 04-09-2024 21:58:12 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[tblLeftButtons](
    [lbtnid] [int] NOT NULL,
    [lbtnName] [varchar](100) NULL,
    [lbtnImage] [varchar](max) NULL,
    [lbtnUrl] [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]
GO
/***** Object: Table [dbo].[tblPageButtonMapping]  Script Date: 04-09-2024 21:58:12 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
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]
GO
/***** Object: View [dbo].[v_tblPageButtonMapping]  Script Date: 04-09-2024 21:58:12 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE view [dbo].[v_tblPageButtonMapping] as
SELECT pagebtnid
        ,pageid
        ,p.lbtnid,l.lbtnName
        ,show
        ,priority
FROM tblPageButtonMapping p,dbo.tblLeftButtons l

```

```

    where p.lbtid=l.lbtid
GO
/***** Object: View [dbo].[tbluseraccepted]  Script Date: 04-09-2024 21:58:12 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE view [dbo].[tbluseraccepted] as
SELECT [UserId]
      ,[PSId]
      ,[CreatedOn]
      ,[UId]
      ,[Active]
      ,[SLType]
      ,[AcceptdOn]
FROM [dbo].[UserSendProfile] union all
SELECT [PSId] as UserId
      ,UserId as PSId
      ,[CreatedOn]
      ,[UId]
      ,[Active]
      ,[SLType]
      ,[AcceptdOn]
FROM [dbo].[UserSendProfile]
GO
/***** Object: View [dbo].[v_allcolumns]  Script Date: 04-09-2024 21:58:12 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
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'

GO
/***** Object: View [dbo].[v_dealerbrand]  Script Date: 04-09-2024 21:58:12 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
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

GO

```



```

/***** Object: Table [dbo].[AuditTrail]  Script Date: 04-09-2024 21:58:13 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
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]
GO
/***** Object: Table [dbo].[Castes]  Script Date: 04-09-2024 21:58:13 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Castes](
    [mc_id] [int] NOT NULL,
    [mc_desc] [varchar](50) NOT NULL,
    [uid] [int] NOT NULL,
    [CreatedOn] [datetime2](7) NOT NULL,
    PRIMARY KEY CLUSTERED
(
    [mc_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]
GO
/***** Object: Table [dbo].[city]  Script Date: 04-09-2024 21:58:13 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[city](
    [id] [int] NOT NULL,
    [shortcd] [varchar](25) NOT NULL,

```

```

        [cityname] [varchar](255) NOT NULL,
        [statecode] [int] NOT NULL,
        [IsBlocked] [bit] 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]
GO
/***** Object: Table [dbo].[CollegesList]  Script Date: 04-09-2024 21:58:13 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[CollegesList](
        [C_id] [int] NOT NULL,
        [CollegeName] [varchar](500) NOT NULL,
        [City] [int] NOT NULL,
        [Country] [int] NOT NULL,
        [UserId] [int] NOT NULL,
        [ZipCode] [varchar](10) NOT NULL,
        [CreatedON] [smalldatetime] NOT NULL,
PRIMARY KEY CLUSTERED
(
        [C_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]
GO
/***** Object: Table [dbo].[country]  Script Date: 04-09-2024 21:58:13 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
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]
GO

```

```

/***** Object: Table [dbo].[Currency]  Script Date: 04-09-2024 21:58:13 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
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]
GO

```

```

/***** Object: Table [dbo].[Department]  Script Date: 04-09-2024 21:58:13 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
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]
GO

```

```

/***** Object: Table [dbo].[Languages]  Script Date: 04-09-2024 21:58:13 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Languages](
    [Lan_id] [int] NOT NULL,
    [Language] [varchar](50) NOT NULL,

```

```

        [Country] [varchar](10) NULL,
        [UserId] [int] NOT NULL,
        [CreatedON] [smalldatetime] NOT NULL,
PRIMARY KEY CLUSTERED
(
        [Lan_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]
GO
/***** Object: Table [dbo].[LocationList]  Script Date: 04-09-2024 21:58:13 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[LocationList](
        [ll_id] [int] NOT NULL,
        [ll_desc] [varchar](50) NOT NULL,
        [CreatedOn] [smalldatetime] NOT NULL,
        [Uid] [int] NULL,
PRIMARY KEY CLUSTERED
(
        [ll_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]
GO
/***** Object: Table [dbo].[MartialStatus]  Script Date: 04-09-2024 21:58:13 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[MartialStatus](
        [ms_id] [int] NOT NULL,
        [ms_desc] [varchar](50) NULL,
        [uid] [int] NULL,
        [CreatedOn] [smalldatetime] NULL,
PRIMARY KEY CLUSTERED
(
        [ms_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]
GO
/***** Object: Table [dbo].[menulist]  Script Date: 04-09-2024 21:58:13 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON

```

```

GO
CREATE TABLE [dbo].[menulist](
    [pageid] [int] NOT NULL,
    [pagename] [varchar](50) NOT NULL,
    [pageurl] [varchar](100) NOT NULL,
    [parentid] [int] NOT NULL,
    [priority] [int] NOT NULL,
    [level] [int] NOT NULL,
    [ismenuitem] [int] NOT NULL,
    [active] [int] NOT NULL,
    [imagepath] [varchar](100) NOT NULL,
    [stype] [int] NOT NULL,
    [mlevel] [int] NOT NULL,
    [btnimg] [varchar](50) NOT NULL,
    [createdon] [smalldatetime] NOT NULL,
    [updatedon] [smalldatetime] NOT NULL,
    [userid] [int] NOT NULL,
    [coloroption] [bit] NOT NULL,
    [epid] [int] NOT NULL,
    CONSTRAINT [menulist_idx1] 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]
GO
/***** Object: Table [dbo].[NewsItem]  Script Date: 04-09-2024 21:58:13 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
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]
GO
/***** Object: Table [dbo].[PartnerChoice]  Script Date: 04-09-2024 21:58:14 *****/
SET ANSI_NULLS ON

```

```

GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[PartnerChoice](
    [pc_id] [varchar](1) NOT NULL,
    [pc_desc] [varchar](20) NOT NULL,
    [CreatedOn] [smalldatetime] NOT NULL,
    [uid] [int] NOT NULL,
PRIMARY KEY CLUSTERED
(
    [pc_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]
GO
/***** Object: Table [dbo].[RecentPlaces]  Script Date: 04-09-2024 21:58:14 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
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]
GO
/***** Object: Table [dbo].[Religions]  Script Date: 04-09-2024 21:58:14 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Religions](
    [ml_id] [int] NOT NULL,
    [ml_desc] [varchar](50) NOT NULL,
    [uid] [int] NOT NULL,
    [CreatedOn] [datetime2](7) NOT NULL,
PRIMARY KEY CLUSTERED
(
    [ml_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]
GO
/***** Object: Table [dbo].[Role]  Script Date: 04-09-2024 21:58:14 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
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]
GO
/***** Object: Table [dbo].[SchoolsList]  Script Date: 04-09-2024 21:58:14 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[SchoolsList](
    [S_id] [int] NOT NULL,
    [SchoolName] [varchar](500) NOT NULL,
    [City] [int] NOT NULL,
    [Country] [int] NOT NULL,
    [UserId] [int] NOT NULL,
    [ZipCode] [varchar](10) NOT NULL,
    [CreatedON] [smalldatetime] NOT NULL,
    PRIMARY KEY CLUSTERED
(
    [S_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]
GO
/***** Object: Table [dbo].[SexualOr]  Script Date: 04-09-2024 21:58:14 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[SexualOr](

```

```

        [sx_id] [int] NOT NULL,
        [sx_desc] [varchar](10) NULL,
        [CreatedOn] [smalldatetime] NULL,
        [uid] [int] NULL,
PRIMARY KEY CLUSTERED
(
        [sx_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]
GO
/***** Object: Table [dbo].[state]  Script Date: 04-09-2024 21:58:14 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[state](
        [id] [int] NOT NULL,
        [shortcd] [varchar](25) NOT NULL,
        [statename] [varchar](255) NOT NULL,
        [countrycode] [int] NOT NULL,
        [IsBlocked] [bit] 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]
GO
/***** Object: Table [dbo].[Tbl_ExceptionLoggingToDataBase]  Script Date: 04-09-2024 21:58:14
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
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]

```



```

GO
/***** Object: Table [dbo].[tbl_moptions]  Script Date: 04-09-2024 21:58:14 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[tbl_moptions](
    [mopt_id] [int] NOT NULL,
    [mopt_desc] [varchar](50) NOT NULL,
    [uid] [int] NOT NULL,
    [CreatedOn] [smalldatetime] NOT NULL,
PRIMARY KEY CLUSTERED
(
    [mopt_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]
GO
/***** Object: Table [dbo].[tblcoursalitems]  Script Date: 04-09-2024 21:58:14 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
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]
GO
/***** Object: Table [dbo].[tblLeftDDL]  Script Date: 04-09-2024 21:58:14 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[tblLeftDDL](
    [liddid] [int] NOT NULL,
    [liddvalue] [varchar](100) NULL,
    [liddpriority] [int] NULL,
    [lbtnid] [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]
GO
/***** Object: Table [dbo].[tblLogIn]  Script Date: 04-09-2024 21:58:14 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
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]
GO
/***** Object: Table [dbo].[tblRole]  Script Date: 04-09-2024 21:58:14 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
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]
GO
/***** Object: Table [dbo].[tblUser]  Script Date: 04-09-2024 21:58:14 *****/
SET ANSI_NULLS ON
GO

```

```

SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[tblUser](
    [UserId] [int] NOT NULL,
    [UserName] [varchar](100) NOT NULL,
    [Password] [varchar](400) NOT NULL,
    [FirstName] [varchar](100) NOT NULL,
    [LastName] [varchar](100) NOT NULL,
    [EmailID] [varchar](150) NOT NULL,
    [ContactNum] [varchar](50) NOT NULL,
    [CreatedOn] [datetime] NOT NULL,
    [Status] [bit] NOT NULL,
    [RoleID] [int] NOT NULL,
    [GroupID] [int] NOT NULL,
    [IsRemoved] [bit] NOT NULL,
    [Department] [varchar](100) NOT NULL,
    [Region] [varchar](100) NOT NULL,
    [UpdatedOn] [datetime] NOT NULL,
    [DealerId] [int] NOT NULL,
    [Gender] [varchar](10) NOT NULL,
    [Landline] [varchar](50) NULL,
    [MTongue] [int] NOT NULL,
    [DateOfBirth] [smalldatetime] NOT NULL,
    [TimeOfBirth] [smalldatetime] NOT NULL,
    [PlaceofBirth] [varchar](50) NOT NULL,
    [UpdatedUserId] [int] NOT NULL,
    [Matchingchk] [bit] NOT NULL,
    [adshowchk] [bit] NOT NULL,
    CONSTRAINT [PK_tblUser] 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]
) ON [PRIMARY]
GO
/***** Object: Table [dbo].[tbluser_add_type]  Script Date: 04-09-2024 21:58:14 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
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]
GO
/***** Object: Table [dbo].[tbluser_address]  Script Date: 04-09-2024 21:58:14 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
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,
    [IsDefault] [bit] NULL,
    [addname] [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]
GO
/***** Object: Table [dbo].[tblUser_Emails]  Script Date: 04-09-2024 21:58:14 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[tblUser_Emails](
    [EmailId] [int] NOT NULL,
    [EMSentUserId] [int] NOT NULL,
    [EMRcvdUserId] [int] NOT NULL,
    [EmSubject] [varchar](2250) NOT NULL,
    [EmBody] [varchar](max) NOT NULL,
    [EmSign] [varchar](2500) 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]
GO

```

```

/***** Object: Table [dbo].[tbluser_moptions]  Script Date: 04-09-2024 21:58:15 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[tbluser_moptions](
    [userid] [int] NOT NULL,
    [mopt_id] [int] NOT NULL,
    [mopt_value] [int] NOT NULL,
    [uid] [int] NOT NULL,
    [CreatedOn] [smalldatetime] NOT NULL,
PRIMARY KEY CLUSTERED
(
    [userid] ASC,
    [mopt_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]
GO
/***** Object: Table [dbo].[tbluser_options]  Script Date: 04-09-2024 21:58:15 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[tbluser_options](
    [userid] [int] NOT NULL,
    [Gender_Search] [varchar](1) NOT NULL,
    [Age_from] [int] NOT NULL,
    [Age_to] [int] NOT NULL,
    [Religion_search] [int] NOT NULL,
    [Caste_search] [int] NOT NULL,
    [Area_search] [int] NOT NULL,
    [MTongue_search] [int] NOT NULL,
    [GeoLoc_search] [varchar](100) NOT NULL,
    [CreatedOn] [smalldatetime] NOT 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]
) ON [PRIMARY]
GO
/***** Object: Table [dbo].[tbluser_privacy_chk]  Script Date: 04-09-2024 21:58:15 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[tbluser_privacy_chk](

```

```
[userid] [int] NOT NULL,  
[User_privacy] [int] NOT NULL,  
[Details_chk] [bit] NOT NULL,  
[Occupation_chk] [bit] NOT NULL,  
[Occupation_type_chk] [bit] NOT NULL,  
[Edu_qualification_chk] [bit] NOT NULL,  
[Marital_status_chk] [bit] NOT NULL,  
[Sex_ornt_chk] [bit] NOT NULL,  
[dob_chk] [bit] NOT NULL,  
[tob_chk] [bit] NOT NULL,  
[pob_chk] [bit] NOT NULL,  
[family_nos_chk] [bit] NOT NULL,  
[web_url_chk] [bit] NOT NULL,  
[adhaar_id_chk] [bit] NOT NULL,  
[pan_no_chk] [bit] NOT NULL,  
[weekly_days_chk] [bit] NOT NULL,  
[Income_Year_chk] [bit] NOT NULL,  
[currency_chk] [bit] NOT NULL,  
[Religion_chk] [bit] NOT NULL,  
[Caste_chk] [bit] NOT NULL,  
[Sub_Caste_chk] [bit] NULL,  
[Location_chk] [bit] NOT NULL,  
[GeoLocation_chk] [bit] NOT NULL,  
[Gender_Search_chk] [bit] NOT NULL,  
[Age_from_chk] [bit] NOT NULL,  
[Age_to_chk] [bit] NOT NULL,  
[Religion_search_chk] [bit] NOT NULL,  
[Caste_search_chk] [bit] NOT NULL,  
[Area_search_chk] [bit] NOT NULL,  
[MTongue_search_chk] [bit] NOT NULL,  
[GeoLoc_search_chk] [bit] NOT NULL,  
[Religions_Choice] [bit] NOT NULL,  
[Caste_Choice] [bit] NOT NULL,  
[Areas_Choice] [bit] NOT NULL,  
[Cities_Choice] [bit] NOT NULL,  
[Countries_Choice] [bit] NOT NULL,  
[Languages_Choice] [bit] NOT NULL,  
[uid] [int] NOT NULL,  
[CreatedOn] [smalldatetime] NOT NULL,  
[UpdatedOn] [smalldatetime] NOT NULL,  
[school_chk] [bit] NULL,  
[college_chk] [bit] NULL,  
[university_chk] [bit] NULL,  
[Photos_chk] [bit] NULL,  
[about_chk] [bit] NULL,  
PRIMARY KEY CLUSTERED  
(  
    [userid] ASC,
```

```

        [User_privacy] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object: Table [dbo].[tbluser_profile]  Script Date: 04-09-2024 21:58:15 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[tbluser_profile](
    [userid] [int] NOT NULL,
    [Details] [varchar](max) NOT NULL,
    [about] [varchar](max) NOT NULL,
    [Occupation] [varchar](500) NOT NULL,
    [Occupation_type] [varchar](500) NOT NULL,
    [Edu_qualification] [varchar](500) NOT NULL,
    [Marital_status] [int] NOT NULL,
    [Sex_ornt] [int] NOT NULL,
    [dob] [smalldatetime] NOT NULL,
    [tob] [varchar](10) NOT NULL,
    [pob] [varchar](200) NOT NULL,
    [family_nos] [int] NOT NULL,
    [web_url] [varchar](1000) NOT NULL,
    [adhaar_id] [varchar](200) NOT NULL,
    [pan_no] [varchar](100) NOT NULL,
    [weekly_days] [int] NOT NULL,
    [Income_Year] [numeric](10, 0) NOT NULL,
    [currency] [int] NOT NULL,
    [Religion] [int] NOT NULL,
    [Caste] [int] NOT NULL,
    [Sub_Caste] [varchar](500) NULL,
    [School] [int] NULL,
    [college] [int] NULL,
    [university] [int] NULL,
    [Location] [varchar](500) NOT NULL,
    [GeoLocation] [varchar](1000) NOT NULL,
    [Company_logo] [image] NULL,
    [CreatedOn] [smalldatetime] NOT NULL,
    [UpdatedOn] [smalldatetime] NOT NULL,
    [uid] [int] NOT NULL,
    [logo_path] [varchar](1500) NOT NULL,
    [age] [int] NOT NULL,
    [college_year] [int] NULL,
    [school_year] [int] NULL,
    [Univ_year] [int] 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]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
GO
/***** Object: Table [dbo].[tblUserImages]  Script Date: 04-09-2024 21:58:15 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[tblUserImages](
    [UserId] [int] NOT NULL,
    [srno] [int] NOT NULL,
    [Images] [image] NULL,
    [CreatedOn] [smalldatetime] NULL,
    [ImgPath] [varchar](50) 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]
GO
/***** Object: Table [dbo].[tblUserLockOutDetails]  Script Date: 04-09-2024 21:58:15 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[tblUserLockOutDetails](
    [UserLockId] [int] IDENTITY(1,1) NOT NULL,
    [Logonattempts] [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]
GO
/***** Object: Table [dbo].[tblUserLog]  Script Date: 04-09-2024 21:58:15 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[tblUserLog](
    [UserLogID] [int] IDENTITY(1,1) NOT NULL,

```



```

        [PerformAction] [varchar](max) NULL,
        [PerformActionTime] [datetime] NULL,
        [UserId] [int] NULL,
        [LoginId] [int] NULL,
        [LoggedCity] [varchar](50) NULL,
        [LoggedCountry] [varchar](50) NULL,
        [LoggedIPAddress] [varchar](50) 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]
GO
/***** Object: Table [dbo].[tbluserold]  Script Date: 04-09-2024 21:58:15 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[tbluserold](
        [UserId] [int] NOT NULL,
        [UserName] [varchar](100) NOT NULL,
        [Password] [varchar](400) NOT NULL,
        [FirstName] [varchar](100) NOT NULL,
        [LastName] [varchar](100) NOT NULL,
        [EmailID] [varchar](150) NOT NULL,
        [ContactNum] [varchar](50) NOT NULL,
        [CreatedOn] [datetime] NOT NULL,
        [Status] [bit] NOT NULL,
        [RoleID] [int] NOT NULL,
        [GroupId] [int] NOT NULL,
        [IsRemoved] [bit] NOT NULL,
        [Department] [varchar](100) NOT NULL,
        [Region] [varchar](100) NOT NULL,
        [UpdatedOn] [datetime] NOT NULL,
        [DealerId] [int] NOT NULL,
        [Gender] [char](1) NOT NULL,
        [Landline] [varchar](50) NULL,
        [MTongue] [int] NULL
) ON [PRIMARY]
GO
/***** Object: Table [dbo].[UniversityList]  Script Date: 04-09-2024 21:58:15 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[UniversityList](
        [U_id] [int] NOT NULL,

```

```

        [UniversityName] [varchar](500) NOT NULL,
        [City] [int] NOT NULL,
        [Country] [int] NOT NULL,
        [UserId] [int] NOT NULL,
        [ZipCode] [varchar](10) NOT NULL,
        [CreatedON] [smalldatetime] NOT NULL,
PRIMARY KEY CLUSTERED
(
        [U_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]
GO
/***** Object: Table [dbo].[UserBlockedProfile]  Script Date: 04-09-2024 21:58:15 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[UserBlockedProfile](
        [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]
GO
/***** Object: Table [dbo].[UserCart]  Script Date: 04-09-2024 21:58:15 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
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

```

```

(
    [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]
GO
/***** Object: Table [dbo].[UserPrivilege]  Script Date: 04-09-2024 21:58:15 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
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]
GO
/***** Object: Table [dbo].[UserRole]  Script Date: 04-09-2024 21:58:16 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
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]
GO
/***** Object: Table [dbo].[UserSendProfile]  Script Date: 04-09-2024 21:58:16 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[UserSendProfile](
    [UserId] [int] NOT NULL,

```

```

        [PSId] [int] NOT NULL,
        [CreatedOn] [smalldatetime] NOT NULL,
        [UId] [int] NOT NULL,
        [Active] [bit] NOT NULL,
        [SLType] [bit] NOT NULL,
        [AcceptdOn] [smalldatetime] 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]
GO
/***** Object: Table [dbo].[UserShortlist]  Script Date: 04-09-2024 21:58:16 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
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]
GO
/***** Object: Table [dbo].[WelcomeCallOutcomeEnum]  Script Date: 04-09-2024 21:58:16 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
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]
GO
/***** Object: Table [dbo].[words]  Script Date: 04-09-2024 21:58:16 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[words](
    [word] [nchar](10) NULL,
    [eword] [nchar](100) NULL
) ON [PRIMARY]
GO
ALTER TABLE [dbo].[Role] ADD CONSTRAINT [DF_Role_IsActive] DEFAULT ((1)) FOR [IsActive]
GO
ALTER TABLE [dbo].[tblLeftButtons] ADD CONSTRAINT [DF_tblLeftButtons_lbbtnType] DEFAULT ((0)) FOR
[lbbtnType]
GO
ALTER TABLE [dbo].[tblPageButtonMapping] ADD CONSTRAINT [DF_tblPageButtonMapping_show]
DEFAULT ((0)) FOR [show]
GO
ALTER TABLE [dbo].[tblPermissionGroup] ADD CONSTRAINT [DF_tblPermissionGroup_RoleId] DEFAULT
((3)) FOR [RoleId]
GO
ALTER TABLE [dbo].[tblUser] ADD DEFAULT (getdate()) FOR [CreatedOn]
GO
ALTER TABLE [dbo].[tblUser] ADD DEFAULT ((0)) FOR [IsRemoved]
GO
ALTER TABLE [dbo].[tblUser] ADD DEFAULT ((0)) FOR [DealerId]
GO
ALTER TABLE [dbo].[tblUser] ADD DEFAULT ((0)) FOR [MTongue]
GO
ALTER TABLE [dbo].[tblUser] ADD DEFAULT (getdate()) FOR [DateOfBirth]
GO
ALTER TABLE [dbo].[tblUser] ADD DEFAULT (getdate()) FOR [TimeOfBirth]
GO
ALTER TABLE [dbo].[tbluser_moptions] ADD CONSTRAINT [DF_tbluser_moptions_userid] DEFAULT ((0))
FOR [userid]
GO
ALTER TABLE [dbo].[tbluser_moptions] ADD CONSTRAINT [DF_tbluser_moptions_mopt_id] DEFAULT
((0)) FOR [mopt_id]
GO
ALTER TABLE [dbo].[tbluser_moptions] ADD CONSTRAINT [DF_tbluser_moptions_mopt_value]
DEFAULT ((0)) FOR [mopt_value]
GO
ALTER TABLE [dbo].[tbluser_moptions] ADD CONSTRAINT [DF_tbluser_moptions_uid] DEFAULT ((0))
FOR [uid]

```

```

GO
ALTER TABLE [dbo].[tbluser_privacy_chk] ADD CONSTRAINT [DF_tbluser_privacy_chk_userid ]
DEFAULT ((0)) FOR [userid]
GO
ALTER TABLE [dbo].[tbluser_privacy_chk] ADD CONSTRAINT [DF_tbluser_privacy_chk_ User_privacy ]
DEFAULT ((0)) FOR [User_privacy]
GO
ALTER TABLE [dbo].[tbluser_privacy_chk] ADD CONSTRAINT [DF_tbluser_privacy_chk_ Details_chk ]
DEFAULT ((0)) FOR [Details_chk]
GO
ALTER TABLE [dbo].[tbluser_privacy_chk] ADD CONSTRAINT [DF_tbluser_privacy_chk_ Occupation_chk
] DEFAULT ((0)) FOR [Occupation_chk]
GO
ALTER TABLE [dbo].[tbluser_privacy_chk] ADD CONSTRAINT [DF_tbluser_privacy_chk_
Occupation_type_chk ] DEFAULT ((0)) FOR [Occupation_type_chk]
GO
ALTER TABLE [dbo].[tbluser_privacy_chk] ADD CONSTRAINT [DF_tbluser_privacy_chk_
Edu_qualification_chk ] DEFAULT ((0)) FOR [Edu_qualification_chk]
GO
ALTER TABLE [dbo].[tbluser_privacy_chk] ADD CONSTRAINT [DF_tbluser_privacy_chk_
Marital_status_chk ] DEFAULT ((0)) FOR [Marital_status_chk]
GO
ALTER TABLE [dbo].[tbluser_privacy_chk] ADD CONSTRAINT [DF_tbluser_privacy_chk_ Sex_ornt_chk ]
DEFAULT ((0)) FOR [Sex_ornt_chk]
GO
ALTER TABLE [dbo].[tbluser_privacy_chk] ADD CONSTRAINT [DF_tbluser_privacy_chk_ dob_chk ]
DEFAULT ((0)) FOR [dob_chk]
GO
ALTER TABLE [dbo].[tbluser_privacy_chk] ADD CONSTRAINT [DF_tbluser_privacy_chk_ tob_chk ]
DEFAULT ((0)) FOR [tob_chk]
GO
ALTER TABLE [dbo].[tbluser_privacy_chk] ADD CONSTRAINT [DF_tbluser_privacy_chk_ pob_chk ]
DEFAULT ((0)) FOR [pob_chk]
GO
ALTER TABLE [dbo].[tbluser_privacy_chk] ADD CONSTRAINT [DF_tbluser_privacy_chk_ family_nos_chk
] DEFAULT ((0)) FOR [family_nos_chk]
GO
ALTER TABLE [dbo].[tbluser_privacy_chk] ADD CONSTRAINT [DF_tbluser_privacy_chk_ web_url_chk ]
DEFAULT ((0)) FOR [web_url_chk]
GO
ALTER TABLE [dbo].[tbluser_privacy_chk] ADD CONSTRAINT [DF_tbluser_privacy_chk_ adhaar_id_chk ]
DEFAULT ((0)) FOR [adhaar_id_chk]
GO
ALTER TABLE [dbo].[tbluser_privacy_chk] ADD CONSTRAINT [DF_tbluser_privacy_chk_ pan_no_chk ]
DEFAULT ((0)) FOR [pan_no_chk]
GO
ALTER TABLE [dbo].[tbluser_privacy_chk] ADD CONSTRAINT [DF_tbluser_privacy_chk_
weekly_days_chk ] DEFAULT ((0)) FOR [weekly_days_chk]

```

```

GO
ALTER TABLE [dbo].[tbluser_privacy_chk] ADD CONSTRAINT [DF_tbluser_privacy_chk_
Income_Year_chk ] DEFAULT ((0)) FOR [Income_Year_chk]
GO
ALTER TABLE [dbo].[tbluser_privacy_chk] ADD CONSTRAINT [DF_tbluser_privacy_chk_
currency_chk ]
DEFAULT ((0)) FOR [currency_chk]
GO
ALTER TABLE [dbo].[tbluser_privacy_chk] ADD CONSTRAINT [DF_tbluser_privacy_chk_
Religion_chk ]
DEFAULT ((0)) FOR [Religion_chk]
GO
ALTER TABLE [dbo].[tbluser_privacy_chk] ADD CONSTRAINT [DF_tbluser_privacy_chk_
Caste_chk ]
DEFAULT ((0)) FOR [Caste_chk]
GO
ALTER TABLE [dbo].[tbluser_privacy_chk] ADD CONSTRAINT [DF_tbluser_privacy_chk_
Sub_Caste_chk ]
DEFAULT ((0)) FOR [Sub_Caste_chk]
GO
ALTER TABLE [dbo].[tbluser_privacy_chk] ADD CONSTRAINT [DF_tbluser_privacy_chk_
Location_chk ]
DEFAULT ((0)) FOR [Location_chk]
GO
ALTER TABLE [dbo].[tbluser_privacy_chk] ADD CONSTRAINT [DF_tbluser_privacy_chk_
GeoLocation_chk ] DEFAULT ((0)) FOR [GeoLocation_chk]
GO
ALTER TABLE [dbo].[tbluser_privacy_chk] ADD CONSTRAINT [DF_tbluser_privacy_chk_
Gender_Search_chk ] DEFAULT ((0)) FOR [Gender_Search_chk]
GO
ALTER TABLE [dbo].[tbluser_privacy_chk] ADD CONSTRAINT [DF_tbluser_privacy_chk_
Age_from_chk ]
DEFAULT ((0)) FOR [Age_from_chk]
GO
ALTER TABLE [dbo].[tbluser_privacy_chk] ADD CONSTRAINT [DF_tbluser_privacy_chk_
Age_to_chk ]
DEFAULT ((0)) FOR [Age_to_chk]
GO
ALTER TABLE [dbo].[tbluser_privacy_chk] ADD CONSTRAINT [DF_tbluser_privacy_chk_
Religion_search_chk ] DEFAULT ((0)) FOR [Religion_search_chk]
GO
ALTER TABLE [dbo].[tbluser_privacy_chk] ADD CONSTRAINT [DF_tbluser_privacy_chk_
Caste_search_chk ] DEFAULT ((0)) FOR [Caste_search_chk]
GO
ALTER TABLE [dbo].[tbluser_privacy_chk] ADD CONSTRAINT [DF_tbluser_privacy_chk_
Area_search_chk ] DEFAULT ((0)) FOR [Area_search_chk]
GO
ALTER TABLE [dbo].[tbluser_privacy_chk] ADD CONSTRAINT [DF_tbluser_privacy_chk_
MTongue_search_chk ] DEFAULT ((0)) FOR [MTongue_search_chk]
GO
ALTER TABLE [dbo].[tbluser_privacy_chk] ADD CONSTRAINT [DF_tbluser_privacy_chk_
GeoLoc_search_chk ] DEFAULT ((0)) FOR [GeoLoc_search_chk]
GO
ALTER TABLE [dbo].[tbluser_privacy_chk] ADD CONSTRAINT [DF_tbluser_privacy_chk_
Religions_Choice ] DEFAULT ((0)) FOR [Religions_Choice]

```

```

GO
ALTER TABLE [dbo].[tbluser_privacy_chk] ADD CONSTRAINT [DF_tbluser_privacy_chk_ Caste_Choice ]
DEFAULT ((0)) FOR [Caste_Choice]
GO
ALTER TABLE [dbo].[tbluser_privacy_chk] ADD CONSTRAINT [DF_tbluser_privacy_chk_ Areas_Choice ]
DEFAULT ((0)) FOR [Areas_Choice]
GO
ALTER TABLE [dbo].[tbluser_privacy_chk] ADD CONSTRAINT [DF_tbluser_privacy_chk_ Cities_Choice ]
DEFAULT ((0)) FOR [Cities_Choice]
GO
ALTER TABLE [dbo].[tbluser_privacy_chk] ADD CONSTRAINT [DF_tbluser_privacy_chk_
Countries_Choice ] DEFAULT ((0)) FOR [Countries_Choice]
GO
ALTER TABLE [dbo].[tbluser_privacy_chk] ADD CONSTRAINT [DF_tbluser_privacy_chk_
Languages_Choice ] DEFAULT ((0)) FOR [Languages_Choice]
GO
ALTER TABLE [dbo].[tbluser_privacy_chk] ADD CONSTRAINT [DF_tbluser_privacy_chk_ uid ] DEFAULT
((0)) FOR [uid]
ALTER TABLE [dbo].[tblLeftDDL] WITH CHECK ADD CONSTRAINT [FK_tblLeftDDL_tblLeftButtons]
FOREIGN KEY([lbtnid])
REFERENCES [dbo].[tblLeftButtons] ([lbtnid])
GO
ALTER TABLE [dbo].[tblLeftDDL] CHECK CONSTRAINT [FK_tblLeftDDL_tblLeftButtons]
GO
ALTER TABLE [dbo].[tblUserLog] WITH CHECK ADD CONSTRAINT [FK_tblUserLog_tblLogIn] FOREIGN
KEY([LoginId])
REFERENCES [dbo].[tblLogIn] ([LoginId])
GO
ALTER TABLE [dbo].[tblUserLog] CHECK CONSTRAINT [FK_tblUserLog_tblLogIn]
GO
ALTER TABLE [dbo].[UserRole] WITH CHECK ADD CONSTRAINT [FK_UserRole_Role] FOREIGN
KEY([RoleId])
REFERENCES [dbo].[Role] ([Id])
GO
ALTER TABLE [dbo].[UserRole] CHECK CONSTRAINT [FK_UserRole_Role]
GO
/***** Object: StoredProcedure [dbo].[AddButtonMapping]  Script Date: 04-09-2024 21:58:16
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
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,@mapdescription=map
description from tblPageButtonMapping where pagebtid=@pagebtid
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 @pagebtidnew=isnull(MAX(pagebtid),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
(pagebtid,[pageid],[lbtnid],[show],[priority],[urlpageid],mapid,groupid,maptype
,mapdescription,isactive)
values(@pagebtidnew,@pageid,@lbtnid,@show,1
,@urlpageid,@mapid,@groupid,@maptype
,@mapdescription,1)
end
end
end

```

GO

/***** Object: StoredProcedure [dbo].[AddParent] Script Date: 04-09-2024 21:58:16 *****/

SET ANSI_NULLS ON

GO

SET QUOTED_IDENTIFIER ON

GO

CREATE PROC [dbo].[AddParent]

@PageId int,

@ParentId int,

@Level int

AS

BEGIN

insert into tblTempPages(PId,parentId,[level]) values(@PageId,@ParentId,0)

END

GO

/***** Object: StoredProcedure [dbo].[createtabs] Script Date: 04-09-2024 21:58:16 *****/

SET ANSI_NULLS ON

GO

SET QUOTED_IDENTIFIER ON

GO

CREATE procedure [dbo].[createtabs] as

declare @tabname nvarchar(50)

DECLARE @DynamicSQL nvarchar(1000);

declare @sno int=0

while @sno<100

begin

set @sno=@sno+1

set @tabname='productSeller'+right('0'+convert(varchar(2),@sno),2)

SET @DynamicSQL = N'

CREATE TABLE '+@tabname+'(

[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
)ON [PRIMARY])
'
EXEC (@DynamicSQL);
end
GO
/***** Object: StoredProcedure [dbo].[createview]  Script Date: 04-09-2024 21:58:16 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE procedure [dbo].[createview] as

declare @tablename nvarchar(50)
DECLARE @DynamicSQL nvarchar(max);
declare @sno int=0
set @DynamicSQL=N'create view v_productSeller as select * from productSeller01 '
while @sno<99
begin
set @sno=@sno+1
set @tablename='productSeller'+right('0'+convert(varchar(2),@sno),2)

SET @DynamicSQL = @DynamicSQL+' union all select * from '+@tablename
end
--select @DynamicSQL
EXEC (@DynamicSQL);

GO
/***** Object: StoredProcedure [dbo].[deletegroup]  Script Date: 04-09-2024 21:58:16 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO

CREATE PROCEDURE [dbo].[deletegroup]
@GrpId int
AS
BEGIN
update [group] set status =0 where groupid = @GrpId
END

GO
/***** Object: StoredProcedure [dbo].[DeleteRecentPlaces]  Script Date: 04-09-2024 21:58:16
*****/

```

```

SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE procedure [dbo].[DeleteRecentPlaces](@userId int) as
begin
    delete from RecentPlaces where rp_user=@userId
end
GO
/***** Object: StoredProcedure [dbo].[ExceptionLoggingToDataBase]  Script Date: 04-09-2024
21:58:16 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
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
GO
CREATE PROC [dbo].[GetAllPermissionGroups]
AS
BEGIN
SELECT * FROM tblPermissionGroup
END

GO
/***** Object: StoredProcedure [dbo].[GetAttachment]  Script Date: 04-09-2024 21:58:16 *****/

```

```

SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE procedure [dbo].[GetAttachment](@id int)
as
begin
    select * from Note where id=@id
end
GO
/***** Object: StoredProcedure [dbo].[GetAuditTrailbyPageId]  Script Date: 04-09-2024 21:58:16
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO

CREATE procedure [dbo].[GetAuditTrailbyPageId] (@Id int,@PageId int) as
begin
if @PageId=171 or @PageId=62
begin
select * from AuditTrail where (PageId=171 or PageId=62 ) and ObjectId=@Id order by auditdate desc
end
else if @PageId=131 or @PageId=172
begin
select * from AuditTrail where (PageId=131 or PageId=172 ) and ObjectId=@Id order by auditdate desc
end
else if @PageId=177 or @PageId=179 or @PageId=204 or @PageId=37
begin
select * from AuditTrail where (PageId=177 or PageId=179 or PageId=204 or PageId=37 ) and
ObjectId=@Id order by auditdate desc
end
else if @PageId=0
begin
select * from AuditTrail order by auditdate desc
end
else
begin
select * from AuditTrail where ObjectId=@Id order by auditdate desc,Id desc
end
end

GO
/***** Object: StoredProcedure [dbo].[GetLastPageinPageTree]  Script Date: 04-09-2024 21:58:16
*****/

```

```

SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
-- =====
-- Author:          <Author,,Name>
-- alter date: <alter 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 = @Pageld
    --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

GO
/***** Object: StoredProcedure [dbo].[GetLeftButtons]  Script Date: 04-09-2024 21:58:16 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
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.lbtnid,lb.lbtnname,lb.lbtnimage,lbtncode,lb.lbtnurl,urlpageid,pagename,
pageurl,lb.lbtnpriority,lb.lbtnid,pb.show,pb.priority,lb.lbtnntype,pb.groupid,pb.maptypes,pb.mapdescripti
on
from tblLeftButtons lb,tblPageButtonMapping pb,tblpages tp
where lb.lbtnid=pb.lbtnid 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',lbtnurl='#',urlp
ageid=1,pagename='Search',
----
pageurl='OrderSearchList.aspx?pageid=2&pid='+convert(varchar(10),pcatid),lbtnpriority=10,lbtnid=2,sh
ow=1,priority=1,lbtnntype=2,groupid=1,maptype=1,mapdescription='maptype1=LinkButton,maptype2=H
TML or Script'

--from [dbo].[ProductCat]
END
GO
/***** Object: StoredProcedure [dbo].[GetLeftButtonsbylbtnid]  Script Date: 04-09-2024 21:58:16
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO

```

```

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.lbtnpriority,lb.lbtn
id,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.lbtnpriority,lb.lbtn
id,pb.show,lb.lbtntype,pb.groupid
order by lbtnpriority
```

END

GO

```
/****** Object: StoredProcedure [dbo].[GetLeftButtonsbyPagebtnid]  Script Date: 04-09-2024
```

```
21:58:16 *****/
```

```
SET ANSI_NULLS ON
```

GO

```
SET QUOTED_IDENTIFIER ON
```

GO

```
CREATE PROCEDURE [dbo].[GetLeftButtonsbyPagebtnid](@pagebtnid int)
```

```
AS
```

```
BEGIN
```

```
/*
```

```
lbtntype
```

```
//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,pageurl,lb.lbtn
priority,lb.lbtnid,pb.show,pb.priority,lb.lbtntype,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

GO

```
/****** Object: StoredProcedure [dbo].[GetLeftButtonsPermission]  Script Date: 04-09-2024 21:58:16
```

```
*****/
```

```
SET ANSI_NULLS ON
```

GO


```

SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[GetLeftButtonsPermission](@pageid int,@groupid int)
AS
BEGIN

--select
lb.lbtnid,lb.lbtnname,lbtnimage,lbtnpriority,lbtnntype,pbm.pagebtnid,pbm.pageid,pbm.lbtnid,pbm.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,lbtnntype,pbm.pagebtnid,pbm.pageid,pbm.lbtnid,pbm.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
GO
/***** Object: StoredProcedure [dbo].[GetLeftButtonsProducts]  Script Date: 04-09-2024 21:58:16
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[GetLeftButtonsProducts](@pageid int,@groupid int)
AS
BEGIN

select
lbtnid=ProductId,lbtnname=ProductName,lbtnimage="",lbtnpriority=ProductId,lbtnntype=1,pagebtnid=Pr
oductGroupId,pageid=ProductGroupCategory,lbtnid=ProductId,show=1,priority=ProductId,groupid=Pro
ductGroupId
from [dbo].[product]
where ProductGroupId=@groupid or ProductGroupId=@pageid
END

GO

```

```

/***** Object: StoredProcedure [dbo].[GetLeftDDL]  Script Date: 04-09-2024 21:58:16 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO

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 ,lddpriority=0,@lbtnid
lbtnid from tblUser tu order by tu.username
end
else if (@Roleid=2)
begin
select tu.UserId ID,tu.UserName TEXT ,lddpriority=0,@lbtnid
lbtnid 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 ,lddpriority=0,@lbtnid
lbtnid from tblUser tu WHERE tu.UserId=@loginid and RoleID>2 order by tu.username
end
end
end

```

```

        end
    IF @lbtnid=43 --QUEUE
    begin
    SELECT ID,NAME TEXT,Iddpriority=0,@lbtnid FROM dbo.ActionQueue
    end

    if @lbtnid=44 --role
    begin
    SELECT ID,NAME TEXT,Iddpriority=0,@lbtnid FROM dbo.Role
    end
    if @lbtnid=27 --user
        SELECT userid ID,UserName TEXT,Iddpriority=0,@lbtnid FROM dbo.tblUser order by username

    If(@lbtnid=22)
    begin
    SELECT ID,NAME TEXT,Iddpriority=0,@lbtnid FROM dbo.CustomerOrderLineStatusEnum
    end
    If(@lbtnid=60)
    begin
    SELECT ID,TStatusName TEXT,Iddpriority=0,@lbtnid FROM dbo.TaskStatus
    end
    else if @lbtnid=28 --dealer
    BEGIN
    if(@Roleid=1)
    begin
    select dealer.DealerId ID,dealer.CompanyName TEXT ,Iddpriority=0,@lbtnid lbtnid from dealer
    end
    else if (@Roleid=2)
    begin
    select dealer.DealerId ID,dealer.CompanyName TEXT ,Iddpriority=0,@lbtnid lbtnid from dealer,tblUser
    WHERE tbluser.DealerId=Dealer.DealerId and tbluser.UserId=@loginid
    end
    else
    begin
    select dealer.DealerId ID,dealer.CompanyName TEXT ,Iddpriority=0,@lbtnid lbtnid from dealer,tblUser
    WHERE tbluser.DealerId=Dealer.DealerId and tbluser.UserId=@loginid
    end
    end
    end
    end

GO
/***** Object: StoredProcedure [dbo].[GetMyAcceptedList]  Script Date: 04-09-2024 21:58:16
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO

```

```
CREATE procedure [dbo].[GetMyAcceptedList]( @UserId int,@stype bit )
as
begin
select t.*,p.*,uc.Active,uc.SLType
from tblUser t,tbluser_profile p, tbluseraccepted uc
where t.UserId=p.userid and uc.PSId=t.UserId
and uc.UserId=@UserId
and uc.sltype=@stype
```

```
end
```

```
GO
```

```
/***** Object: StoredProcedure [dbo].[GetMyRcvdProfile] Script Date: 04-09-2024 21:58:16 *****/
```

```
SET ANSI_NULLS ON
```

```
GO
```

```
SET QUOTED_IDENTIFIER ON
```

```
GO
```

```
CREATE procedure [dbo].[GetMyRcvdProfile]( @UserId int,@stype bit )
```

```
as
```

```
begin
```

```
select t.*,p.*,uc.Active,uc.SLType,uc.AcceptdOn
from tblUser t,tbluser_profile p, UserSendprofile uc
where t.UserId=p.userid and uc.userid=t.UserId
and uc.psid=@UserId
--and uc.Active=@stype
```

```
end
```

```
GO
```

```
/***** Object: StoredProcedure [dbo].[GetMySendProfile] Script Date: 04-09-2024 21:58:16
```

```
*****/
```

```
SET ANSI_NULLS ON
```

```
GO
```

```
SET QUOTED_IDENTIFIER ON
```

```
GO
```

```
CREATE procedure [dbo].[GetMySendProfile]( @UserId int,@stype bit )
```

```
as
```

```
begin
```

```
select t.*,p.*,uc.Active,uc.SLType,uc.AcceptdOn
from tblUser t,tbluser_profile p, UserSendprofile uc
where t.UserId=p.userid and uc.PSId=t.UserId
and uc.UserId=@UserId
--and uc.Active=@stype
```

```
end
```

```
GO
```

```
/***** Object: StoredProcedure [dbo].[GetMyShortList] Script Date: 04-09-2024 21:58:16 *****/
```

```

SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE procedure [dbo].[GetMyShortList]( @Userid int,@stype bit )
as
begin
select t.*,p.*,uc.Active,uc.SLType
  from tblUser t,tbluser_profile p, UserShortList uc
 where t.UserId=p.userid and uc.PSId=t.UserId
 and uc.UserId=@Userid
 and uc.Active=@stype

end
GO
/***** Object: StoredProcedure [dbo].[GetMyUserCart]  Script Date: 04-09-2024 21:58:16 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE procedure [dbo].[GetMyUserCart]( @Userid int )
as
begin
select tu.*,p.*,uc.createdon,uc.UClId,uc.PayId,uc.PSId
  from tblUser tu, UserCart uc,tbluser_profile p
 where uc.PSId=tu.userid and uc.Active=1 and p.userid=uc.PSId
 and uc.UserId=@Userid
end
GO
/***** Object: StoredProcedure [dbo].[GetNextDealerId]  Script Date: 04-09-2024 21:58:16 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE procedure [dbo].[GetNextDealerId] AS
  select NextDealerId=isnull(MAX(DealerId),0)+1 from dealer

GO
/***** Object: StoredProcedure [dbo].[GetNextGroupId]  Script Date: 04-09-2024 21:58:16 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
  CREATE procedure [dbo].[GetNextGroupId] as
SELECT GroupId=isnull(max(GroupId),0)+1
  FROM tblpermissionGroup
GO

```

```

/***** Object: StoredProcedure [dbo].[GetNextHoldId]  Script Date: 04-09-2024 21:58:16 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE procedure [dbo].[GetNextHoldId] AS
    select NextHoldId=isnull(MAX(Id),0)+1 from HoldReason
GO
/***** Object: StoredProcedure [dbo].[GetNextHoldReasonId]  Script Date: 04-09-2024 21:58:16
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE procedure [dbo].[GetNextHoldReasonId] AS
    select NextHoldReasonId=isnull(MAX(Id),0)+1 from HoldReason
GO
/***** Object: StoredProcedure [dbo].[GetNextNotelId]  Script Date: 04-09-2024 21:58:16 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE procedure [dbo].[GetNextNotelId] AS
    select NextNotelId=isnull(MAX(Id),0)+1 from Note

GO
/***** Object: StoredProcedure [dbo].[GetNextOrderRefNumber]  Script Date: 04-09-2024 21:58:16
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE procedure [dbo].[GetNextOrderRefNumber] AS
    select NextOrderReferenceNumber=isnull(MAX(OrderReferenceNumber),0)+1 from CustomerOrder
GO
/***** Object: StoredProcedure [dbo].[GetNextOutstandingPrintRunExecuted]  Script Date: 04-09-
2024 21:58:16 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE procedure [dbo].[GetNextOutstandingPrintRunExecuted]
as
begin
    select isnull(max(osprrunid),0)+1 osprrunid from OutstandingPrintRunExecuted
end

```

```

GO
/***** Object: StoredProcedure [dbo].[GetNextUserId]  Script Date: 04-09-2024 21:58:16 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE procedure [dbo].[GetNextUserId] as
select NextUserId=MAX(UserId)+1 from [tblUser]

```

```

GO
/***** Object: StoredProcedure [dbo].[GetPageTree]  Script Date: 04-09-2024 21:58:16 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
-- =====
-- Author:          <Author,,Name>
-- alter date: <alter 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 = @Pageld
    --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

```

```

GO
/***** Object: StoredProcedure [dbo].[GetPageTreeForMenu]  Script Date: 04-09-2024 21:58:16 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON

```

```

GO
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 ingnor null mapid--20141127-->>
  where mapid is not null
-----By Syed to ingnor 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.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 )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

GO
/***** Object: StoredProcedure [dbo].[GetPageTreeForPage]  Script Date: 04-09-2024 21:58:16
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[GetPageTreeForPage]

@GroupId int
AS
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 tblpages
  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 tblpages 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, 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
```

```
GO
```

```
/***** Object: StoredProcedure [dbo].[GetPerSetsForGroup] Script Date: 04-09-2024 21:58:16
```

```
*****/
```

```
SET ANSI_NULLS ON
```

```
GO
```

```
SET QUOTED_IDENTIFIER ON
```

```
GO
```

```
CREATE PROCEDURE [dbo].[GetPerSetsForGroup]
```

```
    @GroupId int
```

```
AS
```

```
BEGIN
```

```
--SET @GroupId = 1 ;
```

```
WITH n(PageId, Pagename, [level], parentId,PageCode,PageUrl,concatenador) AS
```

```
(
```

```
    SELECT PageId, Pagename,[level],parentId,PageCode,PageUrl, ('+CONVERT(VARCHAR (MAX),  
PageId)+' - '+CONVERT(VARCHAR (MAX), [level]))+' as concatenador
```

```
    FROM tblpages
```

```
    WHERE --PageUrl Like '%'+@PageUrl+'%'
```

```

    parentId =0 and isMenuItem=1
--
    UNION ALL
    SELECT m.PageId, 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.PageId END)+' - '+CONVERT(VARCHAR (MAX), m.level)+')' as concatenador
    FROM tblpages as m, n
    WHERE n.PageId = m.parentId and m.isMenuItem=1
)
--SELECT distinct * FROM n ORDER BY concatenador asc

```

```

select distinct MapId,
    t1.PageId,
    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.pageId = t2.PageId
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

```

GO
/***** Object: StoredProcedure [dbo].[GetSearchFinalUsers]  Script Date: 04-09-2024 21:58:16
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO

```

```

CREATE procedure [dbo].[GetSearchFinalUsers](@SrchStrng varchar(300)) as

```

```

select distinct * from tbluser
where

```

```
(username like '%'+@SrchStrng+'%' or firstname like '%'+@SrchStrng+'%' or lastname like '%'+@SrchStrng+'%') )
```

```
GO
```

```
/****** Object: StoredProcedure [dbo].[GetSearchGroupResults] Script Date: 04-09-2024 21:58:16  
*****/
```

```
SET ANSI_NULLS ON
```

```
GO
```

```
SET QUOTED_IDENTIFIER ON
```

```
GO
```

```
CREATE procedure [dbo].[GetSearchGroupResults](  
    @groupid varchar(1),  
    @gender varchar(1)=0,  
    @age_from varchar(5)=0,  
    @age_to varchar(5)=0,  
    @religion varchar(5)=0',  
    @caste varchar(5)=0',  
    @area varchar(25)='',  
    @mtongue varchar(5)=0',  
    @geoloc varchar(100)=''  
    ) as
```

```
Declare @MainSql Varchar(2000)
```

```
declare @age int
```

```
begin
```

```
    update tbluser_profile set age= datediff(year,dob,getdate())
```

```
set @MainSql=' select * from tbluser u, [dbo].[tbluser_profile] p where u.userid=p.userid and  
u.groupid="'+ @groupid+''''
```

```
if @gender>'0'
```

```
begin
```

```
set @MainSql= @MainSql+' and u.gender ='''+ @gender+''''
```

```
end
```

```
if @age_from>0
```

```
begin
```

```
set @MainSql= @MainSql+' and age >='''+@age_from +''''
```

```
end
```

```
if @age_to>0
```

```
begin
```

```
set @MainSql= @MainSql+' and age <='''+@age_to +''''
```

```
end
```

```
if @religion>0
```

```
begin
```

```
set @MainSql= @MainSql+'and p.Religion ='+'@religion +'
```

```
end
```

```
if @caste>0
```

```
begin
```

```
set @MainSql= @MainSql+' and p.caste ='+'@caste +'
```

```
end
```

```

if @area>'0'
begin
set @MainSql= @MainSql+' and p.location ='''+@area +''''
end
if @mtongue>0
begin
set @MainSql= @MainSql+' and u.Mtongue ='+'@mtongue +'
end
if @geoloc<>''
begin
set @MainSql= @MainSql+' and u.geolocation ='+'@geoloc +'
end

        end
        --select * from tbluser u, [dbo].[tbluser_profile] p where u.userid=p.userid and
u.groupid='4' and u.gender ='M'

        exec(@MainSql)
--print @MainSql

--[GetSearchGroupResults] '4','M','0','0','0','0','','0','"
GO
/***** Object: StoredProcedure [dbo].[GetSearchResults]  Script Date: 04-09-2024 21:58:16 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO

        CREATE procedure [dbo].[GetSearchResults](
        @SrchStrng varchar(100)='',
        @gender varchar(1)='0',
        @age_from varchar(5)='0',
        @age_to varchar(5)='0',
        @religion varchar(5)='0',
        @caste varchar(5)='0',
        @area varchar(25)='',
        @mtongue varchar(5)='0',
        @geoloc varchar(100)='',
        @GROUPID VARCHAR(1)='0'
        ) as
Declare @MainSql Varchar(2000)
declare @partsql varchar(200)
declare @age int
declare @count int
declare @sex varchar(1)
begin
if @gender='0'

```

```

set @sex=""
else if @gender='1'
SET @sex='W'
else if @gender='2'
SET @sex='M'
else
set @sex=""
set @SrchStrng=RTRIM(ltrim(@SrchStrng))

update tbluser_profile set age= datediff(year,dob,getdate())
set @MainSql=' select * from tbluser u, [dbo].[tbluser_profile] p where u.userid=p.userid and
u.groupid>3 and ( u.username like "%'+ @SrchStrng +%" or firstname like "%'+@SrchStrng+%" or
lastname like "%'+@SrchStrng+%" ) '
set @partsql=' '
if @GROUPID>'0'
begin
set @partsql= @partsql+' and u.groupid="' + @groupid +'"
end
if @gender>'0'
begin
set @partsql= @partsql+' and u.gender ="' + @sex +'"
end
if @age_from>0
begin
set @partsql= @partsql+' and age >="' + @age_from +'"
end
if @age_to>0
begin
set @partsql= @partsql+' and age <="' + @age_to +'"
end
if @religion>0
begin
set @partsql= @partsql+'and p.Religion =' + @religion +'"
end
if @caste>0
begin
set @partsql= @partsql+' and p.caste =' + @caste +'"
end
if @area>'0'
begin
set @partsql= @partsql+' and p.location ="' + @area +'"
end
if @mtongue>0
begin
set @partsql= @partsql+' and u.Mtongue ="' + @mtongue +'"
end
if @geoloc<>'
begin

```

```

set @partsql= @partsql+' and p.geolocation ='''+@geoloc +''''
end
--select @count =count(*) from tbluser u, [dbo].[tbluser_profile] p where u.userid=p.userid and
u.groupid>3 and ( u.username like '''+ @SrchStrng +'''' or firstname like '''+@SrchStrng+'''' or
lastname like '''+@SrchStrng+'''' )

set @MainSql=@MainSql+@partsql

end

--select * from tbluser u, [dbo].[tbluser_profile] p where u.userid=p.userid --and
u.groupid=@GROUPID
--and ( u.username like '%4%' or firstname like '%4%' or lastname like '%4%' ) and u.gender
='M' and age >='1' and age <='1'and p.Religion =1 and p.caste =1 and p.location ='a' and u.Mtongue ='1'
and p.geolocation ='a'
Exec( @MainSql)
-- print @@MainSql

GO
/***** Object: StoredProcedure [dbo].[GetTotalGroupByResults] Script Date: 04-09-2024 21:58:16
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO

CREATE procedure [dbo].[GetTotalGroupByResults](
@GROUPID VARCHAR(1)='0',@userid int=0
) as
begin
DECLARE @searchstr varchar(100)
='&fromage=0&toage=25&Location=0&Religion=0&MotherTongue=0'
DECLARE @searchstr1 varchar(100)
='&fromage=25&toage=30&Location=0&Religion=0&MotherTongue=0'
DECLARE @searchstr2 varchar(100)
='&fromage=30&toage=70&Location=0&Religion=0&MotherTongue=0'

select searchstr = @searchstr , cnt=count(*),'Age Less than 25 ' as Descr from tbluser u,
[dbo].[tbluser_profile] p where u.userid=p.userid and u.groupid>3 and u.UserId<>@userid and
u.groupid=@GROUPID and p.age<25
union all
select searchstr = @searchstr1 ,cnt=count(*),'Age between 25 & 30 ' as Descr from
tbluser u, [dbo].[tbluser_profile] p where u.userid=p.userid and u.groupid>3 and u.UserId<>@userid
and u.groupid=@GROUPID and p.age>=25 and age<=30
union all

```



```
select searchstr = @searchstr2 , cnt=count(*),'Age Greater than 30 ' as Descr from tbluser u,
[dbo].[tbluser_profile] p where u.userid=p.userid and u.groupid>3 and u.UserId<>@userid and
u.groupid=@GROUPIP and p.age>30
union all
```

```
select searchstr =
'&fromage=0&toage=70&Location='+Location+'&Religion=0&MotherTongue=0'
,cnt=count(*),'Location '+Location as Descr from tbluser u, [dbo].[tbluser_profile] p where
u.userid=p.userid and u.groupid>3 and u.UserId<>@userid and u.groupid=@GROUPIP group by
Location
```

```
union
```

```
select searchstr =
```

```
'&fromage=0&toage=70&Religion='+convert(varchar(10),p.Religion)+'&Location=0&MotherTon
gue=0' , cnt=count(*),'Religion '+r.ml_desc as Descr from tbluser u, [dbo].[tbluser_profile] p,[Religions]
r where u.userid=p.userid and u.groupid>3 and u.UserId<>@userid and u.groupid=@GROUPIP and
r.ml_id=p.Religion group by r.ml_desc,Religion
```

```
union
```

```
select searchstr =
```

```
'&fromage=0&toage=70&MotherTongue=0'+convert(varchar(10),l.Lan_id)+'&Location=0&Religi
on=0' , cnt=count(*),'Mother Tongue '+L.Language as Descr from tbluser u, [dbo].[tbluser_profile]
p,Languages L where u.userid=p.userid and u.groupid>3 and u.UserId<>@userid and
u.groupid=@GROUPIP and l.Lan_id=u.MTongue group by L.Language,l.Lan_id
```

```
end
```

```
GO
```

```
/****** Object: StoredProcedure [dbo].[GetTotalResults] Script Date: 04-09-2024 21:58:16 *****/
```

```
SET ANSI_NULLS ON
```

```
GO
```

```
SET QUOTED_IDENTIFIER ON
```

```
GO
```

```
create procedure [dbo].[GetTotalResults](
```

```
@GROUPIP VARCHAR(1)='0'
```

```
) as
```

```
begin
```

```
select * from tbluser u, [dbo].[tbluser_profile] p where u.userid=p.userid and u.groupid>3 and
u.groupid=@GROUPIP
```

```
end
```

```
GO
```

```
/****** Object: StoredProcedure [dbo].[GetUserLandingPage] Script Date: 04-09-2024 21:58:16
```

```
*****/
```

```
SET ANSI_NULLS ON
```

```
GO
```

```

SET QUOTED_IDENTIFIER ON
GO
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

GO
/***** Object: StoredProcedure [dbo].[GetUserList] Script Date: 04-09-2024 21:58:16 *****/

```

```

SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[GetUserList](@username varchar(50))
AS
BEGIN
select * from tbluser
END

GO
/***** Object: StoredProcedure [dbo].[GetUserOtherServices]  Script Date: 04-09-2024 21:58:16
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE procedure [dbo].[GetUserOtherServices](@userid int,@psid decimal) as
begin
declare @tot int
select @tot=count(psid) from productseller where userid=@userid and psid<>@psid
if @tot>0
begin
select ps.psid,pss.productid,pss.PrdPrice,pss.PrdUnit from productseller ps,productSellerServices pss
where ps.PSId=pss.PSId
and ps.UserId=@userid and ps.PSId<>@psid

end
else
begin
select psid=@psid,ps.productid,prdprice=0,prdunit=1 from SellersProducts ps where
ps.UserId=@userid and ps.productid not in(select productid from productSellerServices where
PSId=@psid)

end
end
select ps.psid,pss.productid,pss.PrdPrice,pss.PrdUnit from productseller ps,productSellerServices pss
where ps.PSId=pss.PSId
GO
/***** Object: StoredProcedure [dbo].[GetUserPage]  Script Date: 04-09-2024 21:58:16 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[GetUserPage] AS
GO
/***** Object: StoredProcedure [dbo].[ManagePermissions]  Script Date: 04-09-2024 21:58:16
*****/

```

```

SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
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 ingnor null mapid--20141127-->>
      where mapid is not null
-----By Syed to ingnor null mapid--20141127--<<
      ORDER BY pageid asc
    end
  Else
    begin
WITH n(PageId, Pagename, [level], parentId,PageCode,PageUrl,concatenador) AS
  (
    SELECT PageId, Pagename,[level],parentId,PageCode,PageUrl,
(''+CONVERT(VARCHAR (MAX), PageId)+' - ''+CONVERT(VARCHAR (MAX), [level])+'') as concatenador
    FROM tblpages
    WHERE --PageUrl Like '''+@PageUrl+''
      parentId =0 and isMenuItem <= 1
    --
      UNION ALL
    SELECT m.PageId,
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.PageId END)+' - ''+CONVERT(VARCHAR (MAX), m.level)+'') as
concatenador
      FROM tblpages as m, n
      WHERE n.PageId = m.parentId and m.isMenuItem <= 1
  )
--SELECT distinct * FROM n ORDER BY concatenador asc

select distinct MapId,
      t1.PageId,
      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.pageId = t2.PageId
      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

```

```

GO
/***** Object: StoredProcedure [dbo].[spalterUser]  Script Date: 04-09-2024 21:58:16 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROC [dbo].[spalterUser]
@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),
--@alterdOn 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,
UpdatedOn,GroupId,Department,Region,EmailID,ContactNum)
VALUES(@UserName,@Password,@FirstName,getdate(),@GroupId,@deptment,@region,@EmailId,@C
ontactnum)

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 alterd'
    END
    ELSE
    BEGIN
        SELECT @rtn = 'Contact Number Already Exists'
    END
    END
    ELSE
    BEGIN
        SELECT @rtn = 'EmailId Already Exists'
    END
    END
    ELSE
    BEGIN
        SELECT @rtn = 'Username already exists'
    END
    END

END
GO
/***** Object: StoredProcedure [dbo].[spalterUserLockOut]  Script Date: 04-09-2024 21:58:16
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROC [dbo].[spalterUserLockOut]
@BloackingPeriod INT,
@Logonattempts 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_Logonattempts INT
DECLARE @Prv_BlockingPeriod INT
select @Prv_BlockingPeriod = BloackingPeriod,@Prv_Logonattempts=Logonattempts from
tblUserLockOutDetails
where UserLockId = 1

```

```

        if(not exists ( select * from tblUserLockOutDetails where logonattempts = @Logonattempts))
        begin
            INSERT INTO tblUserLog (PerformAction, PerformActionTime, UserId, LoginId)
            VALUES('Set Number of failed logon attempts from '+ CONVERT(varchar ,@Prv_Logonattempts) +'
to '+ CONVERT(varchar, @Logonattempts) +'', getdate(), @UserId, @LoginId)
        end

```

```

        if(not exists ( select * from tblUserLockOutDetails where BloackingPeriod = @BloackingPeriod))
        begin
            INSERT INTO tblUserLog (PerformAction, PerformActionTime, UserId, LoginId)
            VALUES('Set Logon Blocking Period (in Hrs) from '+ CONVERT(varchar ,@Prv_BlockingPeriod) +' to
'+ CONVERT(varchar, @BloackingPeriod) +'', getdate(), @UserId, @LoginId)
        end

```

```

UPDATE tblUserLockOutDetails SET
Logonattempts=@Logonattempts,BloackingPeriod=@BloackingPeriod,UpdatedOn=getdate() where
UserLockId=1
UPDAtE tblUserLockOutTimeDevice SET UserLockOutTime=@UserLockOutTime where UserLockOutId=1
END
ELSE
BEGIN
INSERT INTO tblUserLockOutDetails(Logonattempts,BloackingPeriod,UpdatedOn)
VALUES(@Logonattempts,@BloackingPeriod,getdate())
END
END

```



```

-- EXEC spalterUserLockOut '1',10
-- SELECT * FROM tblUserLockOutTimeDevice
-- select * from tblUserLockOutDetails

GO
/***** Object: StoredProcedure [dbo].[spCheckGuest]  Script Date: 04-09-2024 21:58:16 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROC [dbo].[spCheckGuest]
@UserName varchar(100),
@Password varchar(100)="",
@rtm 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
    declare @FirstName varchar(50)
    declare @LastName varchar(50)
    SET @status = 'false'
    set @UserName=dbo.EncryptData(@UserName)

    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 @Logonattempts INT
                    DECLARE @BloackPeriod INT
                    declare @minremaining int
                    SELECT @UserId = UserId FROM tblUser WHERE UserName =
@UserName
                    SELECT @Logonattempts=Logonattempts, @BloackPeriod =
BlockingPeriod FROM tblUserLockOutDetails
                    declare @LupdatedOn datetime

```

```

declare @Cmptime datetime
SELECT @GroupId = GroupId,@DealerId = DealerId, @RoleId =
RoleId,@LupdatedOn=UpdatedOn,@FirstName=FirstName,@LastName=LastName FROM tblUser
WHERE UserId = @UserId

Select @BrandId=0
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,[LoggedCity],[LoggedCountry],[LoggedIPAddress]) VALUES('Logged
On', getdate(), @UserId, @LoginId,@City,@Country,@IpAddress)
                SELECT @rtn as ResultStatus,@UserId as
UserId,@GroupId AS GroupId,dbo.EncryptData( @UserName) as UserName,@status as
UserStatus,@LoginId AS LoginId ,@DealerId as DealerId,@RoleId as RoleId ,@BrandId as BrandId
,@FirstName as FirstName,@LastName as LastName

        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>@Logonattempts)
                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 ,@FirstName as
FirstName,@LastName as LastName

```

```

                end
            else
                begin
                    INSERT INTO
tblLogin(LoginTime,UserId,LoginStatus,SourceIp,City,Country) VALUES(getdate(),
@UserId,'false',@IpAddress,@City,@Country)

                    SELECT @LoginId = @@IDENTITY
                    INSERT INTO tblUserLog

(PerformAction, PerformActionTime, UserId,
LoginId,[LoggedCity],[LoggedCountry],[LoggedIPAddress])VALUES('Failed Logon', getdate(), @UserId,
@LoginId,@City,@Country,@IpAddress)

                    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 ,@FirstName as
FirstName,@LastName as LastName

                end
            END
        end
    else
        begin
            SET @rtn = 'User Removed Or Password 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 ,@FirstName as FirstName,@LastName as LastName
        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 ,@FirstName as FirstName,@LastName as LastName
    end
end

GO
/***** Object: StoredProcedure [dbo].[spCheckLogin]  Script Date: 04-09-2024 21:58:16 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
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 =0
    DECLARE @UserId INT =0
    DECLARE @GroupId INT =0
    DECLARE @status bit =0
    declare @RoleId int =0
    declare @DealerId int =0
    declare @BrandId int =0
    SET @status = 'false'
    declare @FirstName varchar(50)
    declare @LastName varchar(50)
    set @Password=dbo.EncryptData(@Password)
    set @UserName=dbo.EncryptData(@UserName)

    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 @Logonattempts INT
                    DECLARE @BloackPeriod INT
                    declare @minremaining int
                    SELECT @UserId = UserId FROM tblUser WHERE UserName =
@UserName
                    SELECT @Logonattempts=Logonattempts, @BloackPeriod =
BloackingPeriod FROM tblUserLockOutDetails
                    declare @LupdatedOn datetime
                    declare @Cmptime datetime
                    SELECT @GroupId = GroupId,@DealerId = DealerId, @RoleId =
RoleId,@LupdatedOn=UpdatedOn,@FirstName=FirstName,@LastName=LastName FROM tblUser
WHERE UserId = @UserId
                    Select @BrandId=0
                    SELECT @Cmptime = DATEADD(mi,@BloackPeriod,@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,SourceIp,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,[LoggedCity],[LoggedCountry],[LoggedIPAddress]) VALUES('Logged
On', getdate(), @UserId, @LoginId,@City,@Country,@IpAddress)
SELECT @rtn as ResultStatus,@UserId as
UserId,@GroupId AS GroupId,dbo.EncryptData( @UserName) as UserName,@status as
UserStatus,@LoginId AS LoginId ,@DealerId as DealerId,@RoleId as RoleId ,@BrandId as BrandId
,@FirstName as FirstName,@LastName as LastName

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>@Logonattempts)
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 ,@FirstName as
FirstName,@LastName as LastName
end
else
begin
INSERT INTO
tblLogin(LogInTime,UserId,LoginStatus,SourceIp,City,Country) VALUES(getdate(),
@UserId,'false',@IpAddress,@City,@Country)
SELECT @LoginId = @@IDENTITY
INSERT INTO tblUserLog
(PerformAction, PerformActionTime, UserId,
LoginId,[LoggedCity],[LoggedCountry],[LoggedIPAddress])VALUES('Failed Logon', getdate(), @UserId,
@LoginId,@City,@Country,@IpAddress)
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 ,@FirstName as FirstName,@LastName
as LastName
```

```
end
```

```
END
```

```
end
```

```
else
```

```
begin
```

```
SET @rtn = 'User Removed Or Password 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 ,@Password ,@FirstName as FirstName,@LastName as LastName
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 ,@FirstName as FirstName,@LastName as LastName
```

```
end
```

```
end
```

```
GO
```

```
/****** Object: StoredProcedure [dbo].[spCheckLoginDet] Script Date: 04-09-2024 21:58:16 *****/
```

```
SET ANSI_NULLS ON
```

```
GO
```

```
SET QUOTED_IDENTIFIER ON
```

```
GO
```

```
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 @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 @Logonattempts INT
                DECLARE @BloackPeriod INT
                declare @minremaining int
                SELECT @UserId = UserId FROM tblUser WHERE UserName =
@UserName
                SELECT @Logonattempts=Logonattempts, @BloackPeriod =
BloackingPeriod 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>@Logonattempts)
                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,SourceIp,City,Country) VALUES(getdate()),
@UserId,'false',@IpAddress,@City,@Country)
                    SELECT @LoginId = @@IDENTITY
                    INSERT INTO tblUserLog
(PerformAction, PerformActionTime, UserId, LoginId)VALUES('Failed Logon', 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 Passwored 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

```



```

end
GO
/***** Object: StoredProcedure [dbo].[spCreateUser]  Script Date: 04-09-2024 21:58:16 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
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,@C
ontactnum)

            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
    ELSE
    BEGIN
        SELECT @rtn = 'Username already exists'
    END
    END

END
GO
/***** Object: StoredProcedure [dbo].[spCreateUserLockOut]  Script Date: 04-09-2024 21:58:16
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
-- SELECT * FROM tblUserLockOutDetails
CREATE PROC [dbo].[spCreateUserLockOut]
@BlockingPeriod INT,
@Logonattempts 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_Logonattempts INT
DECLARE @Prv_BlockingPeriod INT
select @Prv_BlockingPeriod = BloackingPeriod,@Prv_Logonattempts=Logonattempts from
tblUserLockOutDetails
where UserLockId = 1

```

```

        if(not exists ( select * from tblUserLockOutDetails where logonattempts = @Logonattempts))
        begin
            INSERT INTO tblUserLog (PerformAction, PerformActionTime, UserId, LoginId)
            VALUES('Set Number of failed logon attempts from '+ CONVERT(varchar ,@Prv_Logonattempts) +'
to '+ CONVERT(varchar, @Logonattempts) +'', getdate(), @UserId, @LoginId)
        end

```

```

        if(not exists ( select * from tblUserLockOutDetails where BloackingPeriod = @BloackingPeriod))
        begin
            INSERT INTO tblUserLog (PerformAction, PerformActionTime, UserId, LoginId)
            VALUES('Set Logon Blocking Period (in Hrs) from '+ CONVERT(varchar ,@Prv_BlockingPeriod) +' to
'+ CONVERT(varchar, @BloackingPeriod) +'', getdate(), @UserId, @LoginId)
        end

```

```

UPDATE tblUserLockOutDetails SET
Logonattempts=@Logonattempts,BloackingPeriod=@BloackingPeriod,UpdatedOn=getdate() where
UserLockId=1
UPDAtE tblUserLockOutTimeDevice SET UserLockOutTime=@UserLockOutTime where UserLockOutId=1
END
ELSE
BEGIN
INSERT INTO tblUserLockOutDetails(Logonattempts,BloackingPeriod,UpdatedOn)
VALUES(@Logonattempts,@BloackingPeriod,getdate())
END
END

```

```
-- EXEC spCreateUserLockOut '1',10
-- SELECT * FROM tblUserLockOutTimeDevice
-- select * from tblUserLockOutDetails
```

```
GO
```

```
/***** Object: StoredProcedure [dbo].[spDisableUser]  Script Date: 04-09-2024 21:58:16 *****/
```

```
SET ANSI_NULLS ON
```

```
GO
```

```
SET QUOTED_IDENTIFIER ON
```

```
GO
```

```
CREATE PROC [dbo].[spDisableUser]
```

```
@UserID int,
```

```
@Status bit
```

```
AS
```

```
BEGIN
```

```
UPDATE tblUser SET [Status] = @Status where UserId = @UserID
```

```
END
```

```
GO
```

```
/***** Object: StoredProcedure [dbo].[spGetAllUsers]  Script Date: 04-09-2024 21:58:16 *****/
```

```
SET ANSI_NULLS ON
```

```
GO
```

```
SET QUOTED_IDENTIFIER ON
```

```
GO
```

```
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),@Recnt = 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.alterdon,103)+ CONVERT(CHAR( 5),t1.alterdon,114) alterdon,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,@Recnt)+' 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.alterdon, 103) +' ' + RIGHT(CONVERT(VARCHAR, t1.alterdon, 100), 7)
alterdon,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,@Recnt)+' 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.alterdon, 103) +' ' + RIGHT(CONVERT(VARCHAR, t1.alterdon, 100), 7)
alterdon,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
```

```
GO
```

```
/****** Object: StoredProcedure [dbo].[spGetIDSValue] Script Date: 04-09-2024 21:58:16 *****/
```

```
SET ANSI_NULLS ON
```

```
GO
```

```

SET QUOTED_IDENTIFIER ON
GO
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

GO
/***** Object: StoredProcedure [dbo].[spGetPerSetsforGrp]  Script Date: 04-09-2024 21:58:16
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
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

```

```

GO
/***** Object: StoredProcedure [dbo].[spGetPerSetsforPage]  Script Date: 04-09-2024 21:58:16
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
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.PagId,t1.PagName,t1.PagCode,t1.PagUrl,
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.pagId = t2.PagId and t3.GroupId =t2.groupid
and t1.PagUrl 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,PagId, Pagename, [level], parentId,PageCode,PageUrl,concatenador) AS
(
SELECT isMenuItem, PagId, Pagename,[level],parentId,PageCode,PageUrl, ('+CONVERT(VARCHAR
(MAX), PagId)+' - '+CONVERT(VARCHAR (MAX), [level]))+')' as concatenador
FROM tblpages
WHERE PageUrl Like '%'+@PageUrl+'%'
UNION ALL
SELECT m.isMenuItem, m.PagId, m.Pagename,m.level,m.parentId,m.PageCode,m.PagUrl,
n.concatenador+' * ('+CONVERT (VARCHAR (MAX), case when ISNULL(m.parentId, 0) = 0 then 0 else
m.PagId END)+' - '+CONVERT(VARCHAR (MAX), m.level)+'')' as concatenador

```

```
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

GO

/****** Object: StoredProcedure [dbo].[spGetSourceIp] Script Date: 04-09-2024 21:58:16 *****/

SET ANSI_NULLS ON

GO

SET QUOTED_IDENTIFIER ON

GO

--sp_helptext 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.alterdon,103)+ CONVERT(CHAR( 5),t1.alterdon,114) alterdon,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.Sourcelp,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.Sourcelp,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
```

```
GO
```

```
/***** Object: StoredProcedure [dbo].[spGetUserInfo]  Script Date: 04-09-2024 21:58:16 *****/
```

```
SET ANSI_NULLS ON
```

```
GO
```

```
SET QUOTED_IDENTIFIER ON
```

```
GO
```

```
CREATE PROC [dbo].[spGetUserInfo]
```

```
@UserId int
```

```
AS
```

```
BEGIN
```

```
select
```

```
userId,UserName,[Password],FirstName,LastName,EmailId,GroupId,ContactNum,Department,Region
```

```
from tblUser where UserId = @UserId
```

```
END
```

```
GO
```

```
/***** Object: StoredProcedure [dbo].[spGetUserLoclOutDetails]  Script Date: 04-09-2024 21:58:16
```

```
*****/
```

```
SET ANSI_NULLS ON
```

```
GO
```

```
SET QUOTED_IDENTIFIER ON
```

```
GO
```

```
CREATE PROC [dbo].[spGetUserLoclOutDetails]
```

```
AS
```

```
BEGIN
```

```
SELECT Logonattempts, BloackingPeriod FROM tblUserLockOutDetails where UserLockId=1
```

```
END
```

```
GO
```

```
/***** Object: StoredProcedure [dbo].[spGetUserLoclOutTimeDevice]  Script Date: 04-09-2024
```

```
21:58:16 *****/
```

```
SET ANSI_NULLS ON
```

```
GO
```

```
SET QUOTED_IDENTIFIER ON
```

```

GO
CREATE PROC [dbo].[spGetUserLoclOutTimeDevice]

AS
BEGIN
    SELECT top 1 * from tblUserLockOutTimeDevice
END

GO
/***** Object: StoredProcedure [dbo].[spLogOut]  Script Date: 04-09-2024 21:58:16 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROC [dbo].[spLogOut]
@UserId INT=0,
@LogInId INT =0 ,
@city varchar(50)='',
@country varchar(50)='',
@ipadd varchar(50)='0'
AS
BEGIN
UPDATE tblLogin SET LogOutTime=getdate() WHERE UserId = @UserId and LogInId =@LogInId

INSERT INTO tblUserLog (PerformAction, PerformActionTime, UserId,
LoginId,LoggedCity,LoggedCountry,LoggedIPAddress)
VALUES('Logged Off', getdate(), @UserId, @LogInId,@city,@country,@ipadd)

END

GO
/***** Object: StoredProcedure [dbo].[spRemoveUser]  Script Date: 04-09-2024 21:58:16 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROC [dbo].[spRemoveUser]
@UserID int,
@Status bit
AS
BEGIN

UPDATE tblUser SET [IsRemoved] = @Status where UserId = @UserID
END

GO
/***** Object: StoredProcedure [dbo].[spUpdateUser]  Script Date: 04-09-2024 21:58:16 *****/
SET ANSI_NULLS ON

```

```

GO
SET QUOTED_IDENTIFIER ON
GO
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
-- SELECT * FROM tblUser
GO
/***** Object: StoredProcedure [dbo].[UnlockSchedule] Script Date: 04-09-2024 21:58:16 *****/
SET ANSI_NULLS ON
GO

```

```
SET QUOTED_IDENTIFIER ON
GO
```

Note: Database scripts for MySQL and PostgreSQL can be provided on request

Chapter 3

Entity classes:

These classes can be downloaded from Dropbox using this link

<https://www.dropbox.com/scl/fi/dn12dt6ze5ncbydqxgh03/DOWEntity.zip?rlkey=hgrsn970osm0212wybbs3rnrk&st=8icvw73c&dl=0>

or use the following code:

Name Space DOWEntity

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

```
namespace DOWEntity
{
    public class UserPermissions
    {
        private int _MapId;

        public int MapId
        {
            get { return _MapId; }
            set { _MapId = value; }
        }

        private int _PageId;

        public int PageId
        {
            get { return _PageId; }
            set { _PageId = 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 Microsoft.Practices.EnterpriseLibrary.Validation;
using Microsoft.Practices.EnterpriseLibrary.Validation.Validators;
using System;

namespace EzeTalk.BussinesEntitys
{
    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.";
    }
}
}

}
}
-----
namespace DOWEntity
{
    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 ErrorMessage { get; set; }  
    }  
}
```

```
namespace DOWEntity
```

```
{  
    public class PermissionsGroup  
    {  
        public string mpageid { get; set; }  
        public string status { get; set; }  
        public string buttonid { get; set; }  
    }  
}
```

```
namespace DOWEntity
```

```
{  
  
    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; }  
    }  
  
}
```

```
}  
namespace DOWEntity
```

```
{  
    public class qadata  
    {  
        public int Id { get; set; }  
        public string ddlval { get; set; }  
        public string txtval { get; set; }  
    }  
    public class ChatUsers  
    {  
        public string ConnectionId { get; set; }  
        public string UserName { get; set; }  
        public string UserImage { get; set; }  
        public string LoginTime { get; set; }  
        public string UName { get; set; }  
    }  
    public class ChatMessages  
    {
```

```
    public string UserName { get; set; }

    public string Message { get; set; }

    public string Time { get; set; }

    public string UserImage { get; set; }
    public string UName { get; set; }
}

}
-----
namespace DOWEntity
{

    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 DOWEntity
{
    public class Location
    {
        public string IPAddress { get; set; }
        public string CountryName { get; set; }
        public string CountryCode { get; set; }
        public string CityName { get; set; }
        public string RegionName { get; set; }
        public string ZipCode { get; set; }
        public string Latitude { get; set; }
        public string Longitude { get; set; }
        public string TimeZone { get; set; }
    }
}
```

```
}
-----
namespace DOWEntity
{
    public class EteUserPrivilege
    {
        public int UserId { get; set; }

        public int PrivilegeId { get; set; }

        public System.DateTime CreatedOn { get; set; }

        public int BMSUser { get; set; }
        public int PageId { 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; }
    }
    public class Etebluserlog
    {
        public int UserLogID{get;set;}

        public string PerformAction{get;set;}

        public System.DateTime PerformActionTime{get;set;}

        public int UserId{get;set;}

        public int LoginId{get;set;}

        public string LoggedCity{get;set;}

        public string LoggedCountry{get;set;}

        public string LoggedIPAddress{get;set;}
    }
}
```

```
}  
-----  
namespace DOWEntity  
{  
    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> PglD { get; set; }  
        public int UId { get; set; }  
  
    }  
}
```

```
-----  
using System;  
  
namespace DOWEntity  
{  
    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 DateTime CreatedOn { get; set; }  
  
        public bool Status { get; set; }  
  
    }  
}
```

```

public int RoleID { get; set; }

public int GroupId { get; set; }

public bool IsRemoved { get; set; }

public string Department { get; set; }

public string Region { get; set; }

public 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; }
public string Gender { get; set; }
public string Landline { get; set; }

public int MTongue { get; set; }
public System.DateTime DateOfBirth { get; set; }
public System.DateTime TimeOfBirth { get; set; }
public string PlaceOfBirth { get; set; }

public bool MatchingChk { get; set; }
public bool adshowchk { get; set; }
}
public class EteGetSearchResults
{
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.DateTime    CreatedOn { get; set; }
public  bool    Status { get; set; }
public  int    RoleID { get; set; }
public  int    GroupId { get; set; }
public  bool    IsRemoved { get; set; }
public  string  Department { get; set; }
public  string  Region { get; set; }
public  System.DateTime    UpdatedOn { get; set; }
public  int    DealerId { get; set; }
public  string  Gender { get; set; }
public  string  Landline { get; set; }
}

```

```
public int     MTongue { get; set; }
public int     userid1 { get; set; }
public string  Details { get; set; }
public string  Occupation { get; set; }
public string  Occupationtype { get; set; }
public string  Eduqualification { get; set; }
public int     Martialstatus { get; set; }
public string  Sexornt { get; set; }
public System.DateTime    dob { get; set; }
public string  tob { get; set; }
public string  pob { get; set; }
public int     familynos { get; set; }
public string  weburl { get; set; }
public string  adhaarid { get; set; }
public string  panno { get; set; }
public int     weeklydays { get; set; }
public decimalIncomeYear { get; set; }
public int     currency { get; set; }
public int     Religion { get; set; }
public int     Caste { get; set; }
public string  SubCaste { get; set; }
public string  Location { get; set; }
public string  GeoLocation { get; set; }
public System.Data.Linq.Binary    Companylogo { get; set; }
public System.DateTime    CreatedOn1 { get; set; }
public System.DateTime    UpdatedOn1 { get; set; }
public int     uid { get; set; }
public string  logopath { get; set; }
public int     age { get; set; }
public System.DateTime    DateOfBirth { get; set; }
public System.DateTime    TimeOfBirth { get; set; }
public string  PlaceOfBirth { get; set; }

public bool MatchingChk { get; set; }
public bool adshowchk { get; set; }
public System.Data.Linq.Binary ImageURL { get; set; }
public string About { get; set; }
public string MotherTongue { get; set; }
public string GroupName { get; set; }
public string Aname { get; set; }
public bool Active { get; set; }
public bool Sltype { get; set; }
public bool SentActive { get; set; }
public bool SentSltype { get; set; }
public bool RecvActive { get; set; }
public bool RecvSltype { get; set; }
public bool HeartListAdded { get; set; }
public bool ShortListAdded { get; set; }
```

```
public bool SentProfileAdded { get; set; }
public bool RcvdProfileAdded { get; set; }
public int School { get; set; }
public int College { get; set; }
public int University { get; set; }
public string SchoolName { get; set; }
public string CollegeName { get; set; }
public string UniversityName { get; set; }
```

```
}
public class EteHeartInfo
{
    public int UserId { get; set; }
    public bool Sltype { get; set; }
    public bool SentActive { get; set; }
    public bool SentSltype { get; set; }
    public bool RecvActive { get; set; }
    public bool RecvSltype { get; set; }
    public bool HeartListAdded { get; set; }
    public bool ShortListAdded { get; set; }
    public bool SentProfileAdded { get; set; }
    public bool RcvdProfileAdded { get; set; }
```

```
}
public class getcount
{
    public string Desc { get; set; }
    public int count { get; set; }
    public string SearchStr { get; set; }
}
```

```
}
.....
namespace DOWEntity
{
    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; }
    }
}
.....
```

```

namespace DOWEntity
{
    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;

```

```

namespace DOWEntity
{
    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 Etebluser_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 bool IsActive { get; set; }
    public string MSAttachment { 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; }
    public string ImgPath { get; set; }

}
}

```

```

using System;

```

```

namespace DOWEntity
{
    public class Etetbluser_profile
    {
        public int userid { get; set; }
        public string Details { get; set; }
        public string Occupation { get; set; }
        public string Occupationtype { get; set; }
        public string Eduqualification { get; set; }
        public int Martialstatus { get; set; }
        public int Sexornt { get; set; }
        public DateTime dob { get; set; }
        public string tob { get; set; }
        public string pob { get; set; }
        public int familynos { get; set; }
        public string weburl { get; set; }
        public string adhaarid { get; set; }
        public string panno { get; set; }
        public int weeklydays { get; set; }
        public System.Data.Linq.Binary Companylogo { get; set; }
        public DateTime CreatedOn { get; set; }
        public DateTime UpdatedOn { get; set; }
        public int uid { get; set; }
        public string logopath { get; set; }
        public decimal Income_year { get; set; }
        public int currency { get; set; }
        public string Location { get; set; }
    }
}

```

```
public string GeoLocation { get; set; }
public int Religion { get; set; }

public int Caste { get; set; }

public string Sub_Caste { get; set; }
public string StrReligion { get; set; }

public string StrCaste { get; set; }

public string StrMaritalstatus { get; set; }

public string StrSexornt { get; set; }
public string About { get; set; }
public int Age { get; set; }
public int School { get; set; }
public int College { get; set; }
public int University { get; set; }
public string SchoolName { get; set; }
public string CollegeName { get; set; }
public string UniversityName { get; set; }
public int School_year { get; set; }
public int College_year { get; set; }
public int Univ_year { get; set; }
public bool Privacy { get; set; }
```

```
    }
}
-----
namespace DOWEntity
{
    public class Etetbluser_privacy_chk
    {
        public int userid;

        public int User_privacy;

        public bool Details_chk;

        public bool Occupation_chk;

        public bool Occupation_type_chk;

        public bool Edu_qualification_chk;

        public bool Martial_status_chk;
```

public bool Sex_ornt_chk;
public bool dob_chk;
public bool tob_chk;
public bool pob_chk;
public bool family_nos_chk;
public bool web_url_chk;
public bool adhaar_id_chk;
public bool pan_no_chk;
public bool weekly_days_chk;
public bool Income_Year_chk;
public bool currency_chk;
public bool Religion_chk;
public bool Caste_chk;
public bool Sub_Caste_chk;
public bool Location_chk;
public bool GeoLocation_chk;
public bool Gender_Search_chk;
public bool Age_from_chk;
public bool Age_to_chk;
public bool Religion_search_chk;
public bool Caste_search_chk;
public bool Area_search_chk;
public bool MTongue_search_chk;
public bool GeoLoc_search_chk;

```

public bool Religions_Choice;

public bool Caste_Choice;

public bool Areas_Choice;

public bool Cities_Choice;

public bool Countries_Choice;

public bool Languages_Choice;

public int uid;

public System.DateTime CreatedOn;

public System.DateTime UpdatedOn;
public bool school_chk;
public bool college_chk;
public bool university_chk;
public bool about_chk;
public bool Photos_chk;
}
}
-----
namespace DOWEntity
{
    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; }
```

```
}
}
```

```
namespace DOWEntity
```

```
{
```

```
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 Uid { get; set; }
```

```
}
}
```

```
namespace DOWEntity
```

```
{
```

```
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> DecisionById { 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 UId { get; set; }
public string StatusName { get; set; }

}
}
using System;

namespace DOWEntity
{
    public class Etetbl_moptions
    {
        public int mopt_id { get; set; }

        public string mopt_desc { get; set; }

        public int uid { get; set; }

        public System.DateTime CreatedOn { get; set; }

    }
    public class Etetbluser_moptions
    {
        public int userid{ get; set; }

        public int mopt_id{ get; set; }

        public string mopt_desc { get; set; }

        public int mopt_value{ get; set; }
    }
}

```

```

    public int uid{ get; set; }

    public DateTime CreatedOn{ get; set; }
}
public class Etetbluser_moptionsVal
{
    public int userid{ get; set; }

    public int mopt_id{ get; set; }

    public int mopt_value{ get; set; }

    public int uid{ get; set; }

    public DateTime CreatedOn{ get; set; }
}

public class Etetbluser_options
{
    public int userid{ get; set; }

    public string Gender_Search{ get; set; }

    public int Age_from{ get; set; }

    public int Age_to{ get; set; }

    public int Religion_search{ get; set; }

    public int Caste_search{ get; set; }

    public int Area_search{ get; set; }

    public int MTongue_search{ get; set; }

    public string GeoLoc_search{ get; set; }

    public DateTime CreatedOn{ get; set; }
    public string Location_search{ get; set; }
}
}
}
-----
namespace DOWEntity
{
    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; }
```

```
}
}
```

```
namespace DOWEntity
```

```
{
    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; }
    }
}
```

```
}
}
```

```
namespace DOWEntity
```

```
{
    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 PageId { get; set; }
        public System.Nullable<System.DateTime> AuditDate { get; set; }
    }
}
```

```
public class EteLoginDetails
{
    public System.Nullable<int> UserId { get; set; }
    public System.Nullable<int> PageId { get; set; }
    public string PageName { set; get; }
}
}
-----
namespace DOWEntity
{
    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 UIId { 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 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 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; }

}
```

```

}
-----
namespace DOWEntity
{
    public class EteAllColumns
    {
        public string tabname { get; set; }

        public string colname { get; set; }

        public int columnid { get; set; }

        public bool isnullable { get; set; }
    }
}
-----
namespace DOWEntity
{
    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; }
    }
}
-----
namespace DOWEntity
{
    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.IO;
using System.Net;
using System.Net.Mail;

```

```

namespace DOWEntity
{
    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));
            }
            mm.IsBodyHtml = false;
            SmtpClient smtp = new SmtpClient();
            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);
        }
    }
}

```

```

namespace DOWEntity
{
    public class EtetblPageButtonMapping
    {
        public int pagebtid { get; set; }
        public System.Nullable<int> pageid { get; set; }
        public int lbtid { 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; }
}
}
-----
namespace DOWEntity
{
    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;

namespace DOWEntity
{
    public class BusinessException : Exception
    {
        public BusinessException()
        {

```

```
}

public BusinessException(string message)
    : base(message)
{
}

public BusinessException(string message, Exception inner)
    : base(message, inner)
{
}
}
}
```

using System;

```
namespace DOWEntity
{
    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.Configuration;
namespace DOWEntity
{
    sealed public class Configuration
    {
        private Configuration()
```

```

    {
    }
    internal static string ConnectionString
    {
        get
        {
            return ConfigurationManager.ConnectionStrings["DOWConectingString"].ConnectionString;
        }
    }
}

}
-----
namespace DOWEntity
{
    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; }
}
```

```
}  
-----  
namespace DOWEntity  
{  
    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;  
        public string FirstName;  
        public string LastName;  
        public int Choice;  
  
    }  
}
```

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;
using System.Data;
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);
    }
}
}
}

```

```

using System;

```

```

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 GetLeftButtonsPermission(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,@lbtnype
            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 DOWEntity;
using System;
using System.Configuration;
using System.Data;
using System.Data.SqlClient;

```

```

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 { }
        }
    }
}

```

```

    }
    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 UserId, int LoginId, string city, string country, string ipaddress)
    {
        SqlParameter[] parameters = new SqlParameter[]
        {
            new SqlParameter("@UserId", UserId),
            new SqlParameter("@LoginId", LoginId),
            new SqlParameter("@city", city),
            new SqlParameter("@country", country),
            new SqlParameter("@ipadd", ipaddress),

        };

        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 GrouptId, 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", GrouptId),
        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 GrouptId, 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", GrouptId),
        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

```

```

    {
        dsSourceIp = SqlHelper.ExecuteDataset(DOWConectingString, CommandType.StoredProcedure,
"spGetSourceIp", parameters);
    }
    catch (Exception ex)
    {
        throw ex;
    }

    return dsSourceIp;
}
#endregion GetSourceIp 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 DOWEntity.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;
    }
}
}

```

namespace DOWDal

```

{
    partial class DOWPortalDataContext
    {
    }
}
}
-----
//DOWConectingString is picked up from web.config
using System.Configuration;
namespace DOWDal
{
    public class ConnectionHelper
    {
        public static string DOWConectingString =
ConfigurationSettings.AppSettings["DOWConectingString"].ToString();
    }
}
}
-----
using System;

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 GetLeftButtonsPermission(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 DOWEntity;
using System;
using System.Collections.Generic;
using System.Configuration;
using System.Data;
using System.Data.SqlClient;
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.PagId = Convert.ToInt32(ds.Tables[0].Rows[0]["PagId"].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.PagId = Convert.ToInt32(ds.Tables[0].Rows[0]["PagId"].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.PageId = Convert.ToInt32(dr["PageId"].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 DOWEntity;
using System;
using System.Configuration;
using System.Data;
using System.Data.SqlClient;

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 UserId, int LoginId, string city, string country, string ipaddress)
{
    SqlParameter[] parameters = new SqlParameter[]
    {
        new SqlParameter("@UserId", UserId),
        new SqlParameter("@LoginId", LoginId),
        new SqlParameter("@city", city),
        new SqlParameter("@country", country),
        new SqlParameter("@ipadd", ipaddress),

    };

    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 GrouptId, 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", GrouptId),
        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 GrouptId, 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", GrouptId),
        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 GetSourceIp details
public DataSet GetSourceIp(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 dsSourceIp = null;
    try
    {
        dsSourceIp = SqlHelper.ExecuteDataset(DOWConectingString, CommandType.StoredProcedure,
"spGetSourceIp", parameters);
    }
    catch (Exception ex)
    {
        throw ex;
    }

    return dsSourceIp;
}
#endregion GetSourceIp details

#region User Lock Details
public bool CreateUserLockOut(int BlockingPeriod, 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 DOWEntity.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;
using System.Data;
using System.Data.SqlClient;
using System.Xml;
```

```
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 (derived 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 to be 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 =
        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 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 =
        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 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
    /// 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 result = ExecuteNonQuery(conn, trans, "PublishOrders", 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 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 =
SqlHelper.ParameterCache.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 =
        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 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 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 SqlParameters
            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 SqlParameters
        return ExecuteDataset(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.:
    /// DataSet ds = ExecuteDataset(trans, CommandType.StoredProcedure, "GetOrders", new
SqlCommandParameter("@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 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 SqlDataReader, 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 =
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 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 =
SqlConnectionHelper.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
SqlConnection.
    /// </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 SqlTransaction
    /// 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 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 SqlDataReader containing the resultset generated by the command</returns>
    public static SqlDataReader ExecuteReader(SqlTransaction 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
    /// 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.:

```

```

    /// 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(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))
        {
            SqlParameter[] commandParameters =
SqlHelper.ParameterCache.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, CommandType.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 =
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 SqlParameters
        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, (SqlTransaction)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 =
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 SqlParameters
            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 SqlParameterers
        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 SqlParameterers 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 SqlParameterers 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 SqlParameterers
        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 SqlParameterers 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 SqlParameterers 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 =
SqlConnectionHelper.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
SqlConnection.

```



```

/// </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
    /// 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(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 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 =
SqlHelper.ParameterCache.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
}

```

```

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());

    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
```

```
public static void CacheParameterSet(string connectionString, string commandText, params  
SqlParameter[] commandParameters)  
{  
    string hashKey = connectionString + ":" + commandText;  
  
    paramCache[hashKey] = commandParameters;  
}
```

```
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
```

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

```
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 Data Tier

Chapter 5

Develop Business Classes

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

<https://www.dropbox.com/scl/fi/tcdon1ha060m7azuy8uea/DOWBusinessLayer.zip?rlkey=vm9q182sxj9djjasub845o2hy&st=3974cvt1&dl=0>

The following classed are added to the business layer

```
using DOWDal;
using DOWEntity;
using System;
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.Configuration;
using System.IO;

```

```

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);
        WriteErrorFile(Error_Write_String, path);
        return;
    }
}

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 DOWDal;
using DOWEntity;
using System;
using System.Collections.Generic;
using System.Linq;
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()).GetPagelId(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,
        PageId = 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 DOWDal;
using DOWEntity;
using System;
using System.Collections.Generic;
using System.Data;
using System.IO;
using System.Linq;
//
using System.Text.RegularExpressions;
using System.Xml;
using System.Xml.Serialization;
//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 Access",
        }
        );
    }

}

}
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,
                        ObjectId = row.ObjectId,
                        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.ObjectId.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 = ClasstToXML(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 DOWDal;
using DOWEntity;
using System.Data;
using System.Security.Cryptography;
using System.Text;
using System;

namespace DOWBal
{
    public class BALCreateUser
    {
        readonly DALUser UserDb = new DALUser();
        public static string key = "cmee-2jh5-alrr45";

        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 Pgeno, int psize)
{
    DALUser Dusr = new DALUser();
    return Dusr.GetUserInfo(col, exp, Pgeno, 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)
{
    //uname = new BalCommon().encrypt(uname);
    //return UserDb.LoginGuest(uname, pwd, MachineName, IPAddress, City, Country);
    DataTable dt;
    dt = UserDb.LoginGuest(uname, pwd, MachineName, IPAddress, City, Country);
    dt.Rows[0]["FirstName"] = Decrypt(dt.Rows[0]["FirstName"].ToString());
    dt.Rows[0]["LastName"] = Decrypt(dt.Rows[0]["LastName"].ToString());

    return dt;
}

public DataTable Login(string uname, string pwd, string IPAddress, string MachineName, string City,
string Country)
{
    //pwd = new BalCommon().encrypt(pwd);
    DataTable dt;
    dt= UserDb.Login(uname, pwd, MachineName, IPAddress, City, Country);
    dt.Rows[0]["FirstName"] = Decrypt(dt.Rows[0]["FirstName"].ToString());
    dt.Rows[0]["LastName"] = Decrypt(dt.Rows[0]["LastName"].ToString());

    return dt;
}

public DataTable LoginDet(string uname, string pwd, string IPAddress, string MachineName, string
City, string Country)
{
    DataTable dt;
    //pwd = new BalCommon().encrypt(pwd);
    dt= UserDb.Login(uname, pwd, MachineName, IPAddress, City, Country);
    dt.Rows[0]["FirstName"] = Decrypt(dt.Rows[0]["FirstName"].ToString());
    dt.Rows[0]["LastName"] = Decrypt(dt.Rows[0]["LastName"].ToString());
    return dt;
}

public bool LogOut(int UserId, int LoginId, string city, string country, string ipaddress)
{

```

```

    return UserDb.LogOut(UserId, LoginId, city,country,ipaddress);
}
public bool RemoveUser(int UserId, bool status)
{
    return UserDb.RemoveUser(UserId, status);
}
public DataSet GetSourceIp(string col, string exp, int Pgeno, int psize)
{
    DALUser Dusr = new DALUser();
    return Dusr.GetSourceIp(col, exp, Pgeno, 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);
}
public static string Decrypt(string input)
{
    try
    {
        if (input != "" && input != null)
        {
            string output = new tblUsers().Decrypt(input);

            // byte[] inputArray = Convert.FromBase64String(input);
            // TripleDESCryptoServiceProvider tripleDES = new TripleDESCryptoServiceProvider();
            // tripleDES.Key = UTF8Encoding.UTF8.GetBytes(key);
            // tripleDES.Mode = CipherMode.ECB;
            // tripleDES.Padding = PaddingMode.PKCS7;
            // ICryptoTransform cTransform = tripleDES.CreateDecryptor();
            // byte[] resultArray = cTransform.TransformFinalBlock(inputArray, 0, inputArray.Length);
            // tripleDES.Clear();

            // return UTF8Encoding.UTF8.GetString(resultArray);
        }
        else { return ""; }
    }
    catch
    {
        return input;
    }
}

```

```

    }
}
public static string Encrypt(string input)
{
    string output = new tblUsers().Encrypt(input);
    return output;

    //if (input == null) { return null; }
    //byte[] inputArray = UTF8Encoding.UTF8.GetBytes(input);
    //TripleDESCryptoServiceProvider tripleDES = new TripleDESCryptoServiceProvider();
    //tripleDES.Key = UTF8Encoding.UTF8.GetBytes(key);
    //tripleDES.Mode = CipherMode.ECB;
    //tripleDES.Padding = PaddingMode.PKCS7;
    //ICryptoTransform cTransform = tripleDES.CreateEncryptor();
    //byte[] resultArray = cTransform.TransformFinalBlock(inputArray, 0, inputArray.Length);
    //tripleDES.Clear();
    //return Convert.ToBase64String(resultArray, 0, resultArray.Length);
}
}
}

```

```

}
}
-----

```

```

using DOWDal;
using System;
using System.IO;
using System.Security.Cryptography;
using System.Text;
using System.Xml;
//using Microsoft.Practices.EnterpriseLibrary.Validation;
//using Microsoft.Practices.EnterpriseLibrary.Validation.Validators;

```

```

namespace DOWBal
{
    public class BalCommon
    {
        public static string key = "cmee-2jh5-alrr45";//
        public static string salt = "c969bcbc-77e4-11eb-9439-0242ac130002";

        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 static string Encrypt(string input)
{
    byte[] inputArray = UTF8Encoding.UTF8.GetBytes(input);
    TripleDESCryptoServiceProvider tripleDES = new TripleDESCryptoServiceProvider();
    tripleDES.Key = UTF8Encoding.UTF8.GetBytes(key);
    tripleDES.Mode = CipherMode.ECB;
    tripleDES.Padding = PaddingMode.PKCS7;
    ICryptoTransform cTransform = tripleDES.CreateEncryptor();
    byte[] resultArray = cTransform.TransformFinalBlock(inputArray, 0, inputArray.Length);
    tripleDES.Clear();
    return Convert.ToBase64String(resultArray, 0, resultArray.Length);
}

public static string Decrypt(string input)
{
    try
    {
        if (input != "" && input != null)
        {
            byte[] inputArray = Convert.FromBase64String(input);
            TripleDESCryptoServiceProvider tripleDES = new TripleDESCryptoServiceProvider();
            tripleDES.Key = UTF8Encoding.UTF8.GetBytes(key);
            tripleDES.Mode = CipherMode.ECB;
            tripleDES.Padding = PaddingMode.PKCS7;
            ICryptoTransform cTransform = tripleDES.CreateDecryptor();
            byte[] resultArray = cTransform.TransformFinalBlock(inputArray, 0, inputArray.Length);
            tripleDES.Clear();

            return UTF8Encoding.UTF8.GetString(resultArray);
        }
        else { return ""; }
    }
    catch
    {
        return "";
    }
}
}

```

```

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 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 string EncryptPwd(string pwd)
{
    string opwd = "";
    using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
    {
        opwd = db.EncryptData(pwd);
    }

    return opwd;
}
public string GetEncryptData(string pwd)
{
    string opwd = "";
    using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
    {
        opwd = db.EncryptData(pwd);
    }

    return opwd;
}

```



```

//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 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 lbtid, int loginid, int lbtntype)
        {
            return clstreev.BindDynamic_DDL(lbtid, loginid, lbtntype);
        }

        public DataSet GetPageButtonDetails(int pagebtid)
        {
            return clstreev.GetPageButtonDetails(pagebtid);
        }
        public DataSet GetLeftButtonsPermission(int pageid, int groupid)
        {
            return clstreev.GetLeftButtonsPermission(pageid, groupid);
        }
        public DataSet GetLeftButtonsProducts(int pageid, int groupid)
        {

```

```

        return clstreev.GetLeftButtonsPermission(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 DOWDal;
using DOWEntity;
using System;
using System.Collections.Generic;
using System.Linq;
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> GetRecentPlaces(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 DOWDal;
using DOWEntity;
using System.Collections.Generic;
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 DOWDal;
using System;
using System.Collections;
using System.Collections.Generic;
using System.Data;
using System.IO;
using System.Linq;
using System.Xml;
using System.Xml.Serialization;

```



```

namespace DOWBal
{

    public class Common
    {
        public static int GetDupRecCount(string TableName, string IdColName, Hashtable colvalues, string ColName)
        {
            return DOWBal.Common.GetDupRecCount(TableName, IdColName, colvalues, ColName);
        }
        public static DataSet GetDDIData(string TableName, string ValueColumn, string TextColumn, Hashtable colvalues)
        {
            return DOWBal.Common.GetDDIData(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 DOWDal;
using DOWEntity;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web.UI.WebControls;

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> 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> GetLanguages()
    {
        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.Languages select new { c.Lan_id, c.Language1 }).ToList();
            result.Add(new PopDDL { Text = "-- Select a Language --", Value = "0" });

            foreach (var row in list)
            {
                result.Add(new PopDDL { Text = row.Language1, Value = row.Lan_id.ToString() });
            }
        }

        return result;
    }
    public List<PopDDL> GetSchoosList()
    {
        List<PopDDL> result = new List<PopDDL>();
        using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
        {
            result.Add(new PopDDL { Text = "-- Select School --", Value = "0" });

```

```

var list = (from c in db.SchoolsLists select new { c.S_id, c.SchoolName }).ToList();

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

return result;
}
public List<PopDDL> GetCollegesList()
{
    List<PopDDL> result = new List<PopDDL>();
    using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
    {
        result.Add(new PopDDL { Text = "-- Select College --", Value = "0" });
        var list = (from c in db.CollegesLists select new { c.C_id, c.CollegeName }).ToList();

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

    return result;
}
public List<PopDDL> GetUniverssityList()
{
    List<PopDDL> result = new List<PopDDL>();
    using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
    {
        result.Add(new PopDDL { Text = "-- Select University --", Value = "0" });
        var list = (from c in db.UniversityLists select new { c.U_id, c.UniversityName }).ToList();

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

```



```

    return result;
}
public List<PopDDL> GetPartnerChoice()
{
    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.PartnerChoices select new { c.pc_id, c.pc_desc }).ToList();
        result.Add(new PopDDL { Text = "-- Select a Partner Choice --", Value = "0" });

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

    return result;
}
public List<PopDDL> GetMaritalStatus()
{
    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.MaritalStatus select new { c.ms_id, c.ms_desc }).ToList();
        result.Add(new PopDDL { Text = "-- Select your Marital Status --", Value = "0" });

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

    return result;
}
public List<PopDDL> GetSexualOr()
{
    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.SexualOrs select new { c.sx_id, c.sx_desc }).ToList();
result.Add(new PopDDL { Text = "-- Select your Orientation --", Value = "0" });

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

return result;
}
public List<PopDDL> GetCastes()
{
    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.Castes select new { c.mc_id, c.mc_desc }).ToList();
result.Add(new PopDDL { Text = "-- Select a Caste --", Value = "0" });

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

return result;
}
public List<PopDDL> GetReligions()
{
    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.Religions select new { c.ml_id, c.ml_desc }).ToList();
result.Add(new PopDDL { Text = "-- Select a Religion --", Value = "0" });

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

```

```

    }
}

return result;
}
public List<PopDDL> GetStates()
{
    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.states where c.IsBlocked == false select new { c.id, c.statename
}).ToList();
        result.Add(new PopDDL { Text = "-- Select a State --", Value = "0" });

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

    return result;
}
public List<PopDDL> GetCities()
{
    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.cities where c.IsBlocked == false select new { c.id, c.cityname }).ToList();
        result.Add(new PopDDL { Text = "-- Select a City --", Value = "0" });

        foreach (var row in list)
        {
            result.Add(new PopDDL { Text = row.cityname, 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> 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> 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 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> GetAllGroups()
{
    List<PopDDL> result = new List<PopDDL>();
    using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
    {
        var list = (from c in db.tblPermissionGroups 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 DOWDal;
using DOWEntity;
using System;
using System.Collections.Generic;
using System.Data;
using System.Data.SqlClient;
using System.Linq;
using System.Net;
using System.Security.Cryptography;
using System.Text;
using System.Configuration;

```

```

namespace DOWBal

```

```

{

```

```

public class EdInstitutions

```

```

{

```

```

    public int InsertCollegesList(EteEdInst data)

```

```

    {

```

```

        bool result = false;

```

```

        int nextid = 0;

```

```

        //Eteproduct _user = new Eteproduct();

```

```

        using (var db = new DOWPortalDataContext(DOWBal.Configuration.ConnectionString))

```

```

        {

```

```

            db.Connection.Open();

```

```

            db.Transaction = db.Connection.BeginTransaction();

```

```

            try

```

```

    {
        var user = (from c in db.CollegesLists where c.C_id == data.id select c).SingleOrDefault();
        if (user == null)
        {
            nextid = (from c in db.CollegesLists orderby c.C_id descending select
c.C_id).FirstOrDefault();
            //if (nextid == 0) { nextid = 0; }
            nextid++;
            var college = (from c in db.CollegesLists where c.CollegeName == data.Name &&
c.City==data.City && c.Country==data.Country select c).ToList();
            if (college.Count == 0)
            {
                CollegesList _cs = new CollegesList
                {
                    C_id = nextid,
                    UserId = data.UserId,
                    CollegeName = data.Name,
                    City = data.City,
                    Country = data.Country,
                    ZipCode = data.ZipCode,
                    CreatedON = data.CreatedON,
                };
                db.CollegesLists.InsertOnSubmit(_cs);

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

return nextid;
}
public int InsertSchoolsList(EteEdInst data)
{
    bool result = false;
    //Eteproduct _user = new Eteproduct();
    int nextid = 0;
    using (var db = new DOWPortalDataContext(DOWBal.Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();
    }
}

```



```

try
{
    var user = (from c in db.SchoolsLists where c.S_id == data.id select c).SingleOrDefault();
    if (user == null)
    {
        nextid = (from c in db.SchoolsLists orderby c.S_id descending select c.S_id).FirstOrDefault();
        //if (nextid == null) { nextid = 0; }
        nextid++;
        var college = (from c in db.SchoolsLists where c.SchoolName == data.Name && c.City ==
data.City && c.Country == data.Country select c).ToList();
        if (college.Count == 0)
        {
            SchoolsList _cs = new SchoolsList
            {
                S_id = nextid,
                UserId = data.UserId,
                SchoolName = data.Name,
                City = data.City,
                Country = data.Country,
                ZipCode = data.ZipCode,
                CreatedON = data.CreatedON,
            };
            db.SchoolsLists.InsertOnSubmit(_cs);

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

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

return nextid;
}
public int InsertUniversityList(EteEdInst data)
{
    bool result = false;
    int nextid = 0;
    //Eteproduct _user = new Eteproduct();
    using (var db = new DOWPortalDataContext(DOWBal.Configuration.ConnectionString))
    {

```

```

db.Connection.Open();
db.Transaction = db.Connection.BeginTransaction();

try
{
    var user = (from c in db.UniversityLists where c.U_id == data.id select c).SingleOrDefault();
    if (user == null)
    {
        nextid = (from c in db.UniversityLists orderby c.U_id descending select
c.U_id).FirstOrDefault();
        //if (nextid == null) { nextid = 0; }
        nextid++;
        var college = (from c in db.UniversityLists where c.UniversityName == data.Name && c.City
== data.City && c.Country == data.Country select c).ToList();
        if (college.Count == 0)
        {
            UniversityList _cs = new UniversityList
            {
                U_id = nextid,
                UserId = data.UserId,
                UniversityName = data.Name,
                City = data.City,
                Country = data.Country,
                ZipCode = data.ZipCode,
                CreatedON = data.CreatedON,
            };
            db.UniversityLists.InsertOnSubmit(_cs);

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

return nextid;
}

public int InsertCity(Etecscity data)
{
    int nextid = 0;
    using (var db = new DOWPortalDataContext(DOWBal.Configuration.ConnectionString))

```

```

{
    db.Connection.Open();
    db.Transaction = db.Connection.BeginTransaction();

    try
    {
        var user = (from c in db.cities where c.id == data.Id select c).SingleOrDefault();
        if (user == null)
        {
            nextid = (from c in db.cities orderby c.id descending select c.id).FirstOrDefault();
            //if (nextid == null) { nextid = 0; }
            nextid++;
            var college = (from c in db.cities where c.cityname == data.Cname && c.statecode == data.Pid
select c).ToList();
            if (college.Count == 0)
            {
                city_cs = new city
                {
                    id=nextid,
                    cityname=data.Cname,
                    statecode=data.Pid,
                    shortcd=data.ShortCd,
                    IsBlocked=false,

                };
                db.cities.InsertOnSubmit(_cs);

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

return nextid;
}
public int InsertState(Etecscity data)
{
    int nextid = 0;
    using (var db = new DOWPortalDataContext(DOWBal.Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();
    }
}

```

```

try
{
    var user = (from c in db.states where c.id == data.Id select c).SingleOrDefault();
    if (user == null)
    {
        nextid = (from c in db.states orderby c.id descending select c.id).FirstOrDefault();
        //if (nextid == null) { nextid = 0; }
        nextid++;
        var college = (from c in db.states where c.statename == data.Cname && c.countrycode ==
data.Pid select c).ToList();
        if (college.Count == 0)
        {
            state _cs = new state
            {
                id = nextid,
                statename = data.Cname,
                countrycode = data.Pid,
                shortcd = data.ShortCd,
                IsBlocked = false,

            };
            db.states.InsertOnSubmit(_cs);

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

return nextid;
}
public int InsertCountry(Etecsity data)
{
    int nextid = 0;
    using (var db = new DOWPortalDataContext(DOWBal.Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {

```

```

var user = (from c in db.countries where c.id == data.Id select c).SingleOrDefault();
if (user == null)
{
    nextid = (from c in db.countries orderby c.id descending select c.id).FirstOrDefault();
    //if (nextid == null) { nextid = 0; }
    nextid++;
    var college = (from c in db.countries where c.countryname == data.Cname select c).ToList();
    if (college.Count == 0)
    {
        country_cs = new country
        {
            id = nextid,
            countryname = data.Cname,

            shortcd = data.ShortCd,
            IsBlocked = false,

        };
        db.countries.InsertOnSubmit(_cs);

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

return nextid;
}

```

```

}
}

```

```

using System;
using System.Configuration;
using System.IO;
using System.Net.Mail;

```

```

namespace DOWEntity
{
    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 Ravi >>>>> 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");

```



```

        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 DOWDal;
using DOWEntity;
using System;
using System.Collections.Generic;
using System.Linq;
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(EtetblGrpMapping data)
{
    bool result = false;
    EtetblGrpMapping _user = new EtetblGrpMapping();
    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<Etev_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.PageId),
                });
            }
        }
    }
}

```

```

        });
    }
}

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 DOWDal;
using DOWEntity;
using System;
using System.Collections.Generic;
using System.Linq;
namespace DOWBal
{
    public class tblPageButtonMapings
    {

```

```

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(EtblPageButtonMapping data)
{
    bool result = false;
    EtblPageButtonMapping _user = new EtblPageButtonMapping();
    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,
            maptype = data.maptype,
            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> GettblPageButtonMappingGroupid(int Groupid)
{
    List<EtetblPageButtonMapping> _user = new List<EtetblPageButtonMapping>();
    using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
    {

        db.Connection.Open();

        var users = (from c in db.tblPageButtonMapings where c.groupid == Groupid select c).ToList();
        foreach (var data in users)
        {
            _user.Add(new EtetblPageButtonMapping
            {
                pagebtnid = data.pagebtnid,
                mapid = data.mapid,
            });
        }

        return _user;
    }
}
}
}

```



```

        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)

```



```

using System.Linq;
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(EtblLeftButtons data)
        {
            bool result = false;
            EtblLeftButtons _user = new EtblLeftButtons();
            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(EtblLeftButtons 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 DOWDal;
using DOWEntity;
using System;
using System.Collections.Generic;
using System.Linq;
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.PagId == 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 GetPagId(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(EtctlPages 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.PageId == data.PageId).SingleOrDefault();

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

            _user.PageId = data.PageId;
            _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.PagelId == 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.PagelId == id select c).ToList();
        foreach (var data in users)
        {
            result = new EtetblPages
            {
                PagelId = data.PagelId,
                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 DOWDal;  
using DOWEntity;  
using System;  
using System.Collections.Generic;  
using System.Data;  
using System.Data.SqlClient;  
using System.Linq;  
using System.Net;  
using System.Security.Cryptography;  
using System.Text;  
using System.IO;  
namespace DOWBal  
{  
    public class tblUsers  
    {  
        public static string key = "cmee-2jh5-alrr45";  
        public static byte[] salt= new byte[] { 0x49, 0x76, 0x61, 0x6e, 0x20, 0x4d, 0x65, 0x64, 0x76, 0x65,  
0x64, 0x65, 0x76 };  
        // "cner-2jh4-alxr45";  
        public EteHeartInfo GetHeartInfo(int userid, int psid)  
        {  
            bool SentActive = false, SentSType = false, RecvActive = false, RecvStype = false;  
            bool HeartListAdded = false, ShortListAdded = false, SentProfileAdded = false, RcvdProfileAdded =  
false;  
            EteHeartInfo info = new EteHeartInfo();  
            using (var db = new DOWPortalDataContext(Configuration.ConnectionString))  
            {  
                var sentprof = (from c in db.UserSendProfiles where c.UserId == userid && c.PSID == psid select  
c).FirstOrDefault();  
                if (sentprof != null) { SentActive = sentprof.Active; SentSType = sentprof.SLType;  
SentProfileAdded = true; }  
                var Recvprof = (from c in db.UserSendProfiles where c.UserId == psid && c.PSID == userid select  
c).FirstOrDefault();  
                if (Recvprof != null) { RecvActive = Recvprof.Active; RecvStype = Recvprof.SLType;  
RcvdProfileAdded = true; }  
                int heartcnt = (from c in db.UserCarts where c.UserId == userid && c.PSID == psid select  
c).Count();  
                if (heartcnt > 0) { HeartListAdded = true; }  
                int shortcnt = (from c in db.UserShortlists where c.UserId == userid && c.PSID == psid select  
c).Count();  
                if (shortcnt > 0) { ShortListAdded = true; }  
            }  
        }  
    }  
}
```

```

info = new EteHeartInfo
{
    UserId = userid,
    SentActive = SentActive,
    SentSType = SentSType,
    RecvActive = RecvActive,
    RecvSType = RecvSType,
    HeartListAdded = HeartListAdded,
    ShortListAdded = ShortListAdded,
    SentProfileAdded = SentProfileAdded,
    RcvdProfileAdded = RcvdProfileAdded,
};
}
return info;
}
public List<EteGetSearchResults> GetCartbyUserId(int userid)
{
    List<EteGetSearchResults> usercart = new List<EteGetSearchResults>();
    //try
    {
        using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
        {
            List<GetMyUserCartResult> user = db.GetMyUserCart(userid).ToList();
            foreach (var row in user)
            {
                string imageUrl = "";
                var imagePath = (from d in db.tblUserImages where d.UserId == row.UserId select
d.Images).FirstOrDefault();
                var path = (from d in db.tblUserImages where d.UserId == row.UserId select
d.ImgPath).FirstOrDefault();
                if (Path == null) { Path = ""; }
                if (ImagePath != null)
                {
                    byte[] bytes = (byte[])ImagePath.ToArray();
                    string base64String = Convert.ToBase64String(bytes, 0, bytes.Length);
                    imageUrl = "data:image/png;base64," + base64String;
                }
                string dept = "";
                string gender = row.Gender.ToString();
                if (gender == "W") { dept = "WOMEN PROFILES"; }
                else if (gender == "M") { dept = "MEN PROFILES"; }
                else if (gender == "B") { dept = "BI-SEXUAL PROFILES"; } else { dept = "OTHERS PROFILES"; }
                string mothertongue = (from c in db.Languages where c.Lan_id == row.MTongue select
c.Language1).SingleOrDefault();
                string Sexornt = "";
                if (row.Sex_ornt == 1) { Sexornt = "Straight"; } else if (row.Sex_ornt == 2) { Sexornt = "Gay";
} else { Sexornt = "Bi-Sexual"; }

```

```

        string groupname = (from c in db.tblPermissionGroups where c.GroupId == row.GroupId
select c.GroupName).FirstOrDefault();
        string Schoolname = (from c in db.SchoolsLists where c.S_id == row.School select
c.SchoolName).SingleOrDefault();
        string UnivName = (from c in db.UniversityLists where c.U_id == row.School select
c.UniversityName).SingleOrDefault();
        string CollegeName = (from c in db.CollegesLists where c.C_id == row.School select
c.CollegeName).SingleOrDefault();
        bool SentActive = false, SentSType = false, RecvActive = false, RecvSType = false;
        bool HeartListAdded = false, ShortListAdded = false, SentProfileAdded = false,
RcvdProfileAdded = false;
        var sentprof = (from c in db.UserSendProfiles where c.UserId == userid && c.PSId ==
row.UserId select c).FirstOrDefault();
        if (sentprof != null) { SentActive = sentprof.Active; SentSType = sentprof.SLType;
SentProfileAdded = true; }
        var Recvprof = (from c in db.UserSendProfiles where c.UserId == row.UserId && c.PSId ==
userid select c).FirstOrDefault();
        if (Recvprof != null) { RecvActive = Recvprof.Active; RecvSType = Recvprof.SLType;
RcvdProfileAdded = true; }
        HeartListAdded = true;
        int shortcnt = (from c in db.UserShortlists where c.UserId == userid && c.PSId ==
row.UserId select c).Count();
        if (shortcnt > 0) { ShortListAdded = true; }

        usercart.Add(new EteGetSearchResults
        {

            UserId = row.UserId,
            UserName = Decrypt(db.EncryptData(row.UserName)),
            FirstName = Decrypt(row.FirstName),
            LastName = Decrypt(row.LastName),
            EmailID = Decrypt(row.EmailID),
            ContactNum = Decrypt(row.ContactNum),
            CreatedOn = row.CreatedOn,
            Status = row.Status,
            RoleID = row.RoleID,
            GroupId = row.GroupId,
            IsRemoved = row.IsRemoved,
            Department = dept,
            Region = row.Region,
            UpdatedOn = row.UpdatedOn,
            DealerId = row.DealerId,
            Gender = row.Gender,
            Landline = Decrypt(row.Landline),
            MTongue = (int)row.MTongue,
            DateOfBirth = (DateTime)row.DateOfBirth,
            TimeOfBirth = (DateTime)row.TimeOfBirth,
            PlaceOfBirth = Decrypt(row.PlaceofBirth),

```

adshowchk = (bool)row.adshowchk,
MatchingChk = (bool)row.Matchingchk,

userid1 = row.userid1,
Details = Decrypt(row.Details),
Occupation = Decrypt(row.Occupation),
Occupationtype = Decrypt(row.Occupation_type),
Eduqualification = Decrypt(row.Edu_qualification),
Maritalstatus = row.Martial_status,
Sexornt = Sexornt,
dob = row.dob,
tob = row.tob,
pob = Decrypt(row.pob),
familynos = row.family_nos,
weburl = Decrypt(row.web_url),
adhaarid = Decrypt(row.adhaar_id),
panno = Decrypt(row.pan_no),
weeklydays = row.weekly_days,
IncomeYear = row.Income_Year,
currency = row.currency,
Religion = row.Religion,
Caste = row.Caste,
SubCaste = Decrypt(row.Sub_Caste),
Location = row.Location,
GeoLocation = Decrypt(row.GeoLocation),
Companylogo = row.Company_logo,
CreatedOn1 = row.CreatedOn1,
UpdatedOn1 = row.UpdatedOn1,
uid = row.uid,
logopath = row.logo_path,
age = row.age,
About = row.about,
MotherTongue = mothertongue,
GroupName = groupname,

Aname = Path,
ImageURL = ImagePath,
SentActive = SentActive,
SentSltype = SentSType,
RecvActive = RecvActive,
RecvSltype = RecvSType,
HeartListAdded = HeartListAdded,
ShortListAdded = ShortListAdded,
SentProfileAdded = SentProfileAdded,
RcvdProfileAdded = RcvdProfileAdded,
School = (int)row.School,
College = (int)row.college,
University = (int)row.university,


```

        SchoolName = Schoolname,
        CollegeName = CollegeName,
        UniversityName = UnivName,

    });

    }

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

//    throw exc;
//}
var usercart1= GetAllUserPrivacyDetails( userid,usercart);
return usercart1;
}

public List<EteGetSearchResults> GetShortListbyUserId(int userid)
{
    List<EteGetSearchResults> usercart = new List<EteGetSearchResults>();
    //try
    {
        using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
        {
            List<GetMyShortListResult> user = db.GetMyShortList(userid, true).ToList();
            foreach (var row in user)
            {
                var ImagePath = (from d in db.tblUserImages where d.UserId == row.UserId select
d.Images).FirstOrDefault();
                byte[] bytes = (byte[])ImagePath.ToArray();
                string base64String = Convert.ToBase64String(bytes, 0, bytes.Length);
                string imageUrl = "data:image/png;base64," + base64String;
                string dept = "";
                string gender = row.Gender.ToString();
                if (gender == "W") { dept = "WOMEN PROFILES"; }
                else if (gender == "M") { dept = "MEN PROFILES"; }
                else if (gender == "B") { dept = "BI-SEXUAL PROFILES"; } else { dept = "OTHERS PROFILES"; }
                string mothertongue = (from c in db.Languages where c.Lan_id == row.MTongue select
c.Language1).SingleOrDefault();
                string Schoolname = (from c in db.SchoolsLists where c.S_id == row.School select
c.SchoolName).SingleOrDefault();
                string UnivName = (from c in db.UniversityLists where c.U_id == row.School select
c.UniversityName).SingleOrDefault();
                string CollegeName = (from c in db.CollegesLists where c.C_id == row.School select
c.CollegeName).SingleOrDefault();

```

```

        string Sexornt = "";
        if (row.Sex_ornt == 1) { Sexornt = "Straight"; } else if (row.Sex_ornt == 2) { Sexornt = "Gay";
    } else { Sexornt = "Bi-Sexual"; }
        string groupname = (from c in db.tblPermissionGroups where c.GroupId == row.GroupId
select c.GroupName).FirstOrDefault();
        bool SentActive = false, SentSType = false, RecvActive = false, RecvStype = false;
        bool HeartListAdded = false, ShortListAdded = false, SentProfileAdded = false,
RcvdProfileAdded = false;
        var sentprof = (from c in db.UserSendProfiles where c.UserId == userid && c.PSId ==
row.UserId select c).FirstOrDefault();
        if (sentprof != null) { SentActive = sentprof.Active; SentSType = sentprof.SLType;
SentProfileAdded = true; }
        var Recvprof = (from c in db.UserSendProfiles where c.UserId == row.UserId && c.PSId ==
userid select c).FirstOrDefault();
        if (Recvprof != null) { RecvActive = Recvprof.Active; RecvStype = Recvprof.SLType;
RcvdProfileAdded = true; }
        int heartcnt = (from c in db.UserCarts where c.UserId == userid && c.PSId == row.UserId
select c).Count();
        if (heartcnt > 0) { HeartListAdded = true; }
        ShortListAdded = true;

        usercart.Add(new EteGetSearchResults
        {

            UserId = row.UserId,
            UserName = Decrypt(db.EncryptData(row.UserName)),
            FirstName = Decrypt(row.FirstName),
            LastName = Decrypt(row.LastName),
            EmailID = Decrypt(row.EmailID),
            ContactNum = Decrypt(row.ContactNum),
            CreatedOn = row.CreatedOn,
            Status = row.Status,
            RoleID = row.RoleID,
            GroupId = row.GroupId,
            IsRemoved = row.IsRemoved,
            Department = dept,
            Region = row.Region,
            UpdatedOn = row.UpdatedOn,
            DealerId = row.DealerId,
            Gender = row.Gender,
            Landline = Decrypt(row.Landline),
            MTongue = (int)row.MTongue,
            DateOfBirth = (DateTime)row.DateOfBirth,
            TimeOfBirth = (DateTime)row.TimeOfBirth,
            PlaceOfBirth = Decrypt(row.PlaceofBirth),
            adshowchk = (bool)row.adshowchk,
            MatchingChk = (bool)row.Matchingchk,

```

userid1 = row.userid1,
Details = Decrypt(row.Details),
Occupation = Decrypt(row.Occupation),
Occupationtype = Decrypt(row.Occupation_type),
Eduqualification = Decrypt(row.Edu_qualification),
Maritalstatus = row.Marital_status,
Sexornt = Sexornt,
dob = row.dob,
tob = row.tob,
pob = Decrypt(row.pob),
familynos = row.family_nos,
weburl = Decrypt(row.web_url),
adhaarid = Decrypt(row.adhaar_id),
panno = Decrypt(row.pan_no),
weeklydays = row.weekly_days,
IncomeYear = row.Income_Year,
currency = row.currency,
Religion = row.Religion,
Caste = row.Caste,
SubCaste = Decrypt(row.Sub_Caste),
Location = row.Location,
GeoLocation = Decrypt(row.GeoLocation),
Companylogo = row.Company_logo,
CreatedOn1 = row.CreatedOn1,
UpdatedOn1 = row.UpdatedOn1,
uid = row.uid,
logopath = row.logo_path,
age = row.age,
About = row.about,
MotherTongue = mothertongue,
GroupName = groupname,

Aname = ImageUrl,
ImageURL = ImagePath,
Active=row.Active,
Sltype=row.SLType,
SentActive=SentActive,
SentSltype=SentSType,
RecvActive=RecvActive,
RecvSltype=RecvStype,
HeartListAdded = HeartListAdded,
ShortListAdded = ShortListAdded,
SentProfileAdded = SentProfileAdded,
RcvdProfileAdded = RcvdProfileAdded,
School = (int)row.School,
College = (int)row.college,
University = (int)row.university,
SchoolName = Schoolname,

```

        CollegeName = CollegeName,
        UniversityName = UnivName,

    });

    }

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

//    throw exc;
//}
var usercart1 = GetAllUserPrivacyDetails(userid, usercart);
return usercart1;
}
public List<EteGetSearchResults> GetSendProfilebyUserId(int userid)
{
    List<EteGetSearchResults> usercart = new List<EteGetSearchResults>();
    //try
    {
        using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
        {
            List<GetMySendProfileResult> user = db.GetMySendProfile(userid, true).ToList();
            foreach (var row in user)
            {
                //var ImagePath = (from d in db.tblUserImages where d.UserId == row.UserId select
d.Images).FirstOrDefault();
                //byte[] bytes = (byte[])ImagePath.ToArray();
                //string base64String = Convert.ToBase64String(bytes, 0, bytes.Length);
                //string imageUrl = "data:image/png;base64," + base64String;
                string dept = "";
                string gender = row.Gender.ToString();
                if (gender == "W") { dept = "WOMEN PROFILES"; }
                else if (gender == "M") { dept = "MEN PROFILES"; }
                else if (gender == "B") { dept = "BI-SEXUAL PROFILES"; } else { dept = "OTHERS PROFILES"; }
                string Sexornt = "";
                if (row.Sex_ornt == 1) { Sexornt = "Straight"; } else if (row.Sex_ornt == 2) { Sexornt = "Gay";
} else { Sexornt = "Bi-Sexual"; }
                string mothertongue = (from c in db.Languages where c.Lan_id == row.MTongue select
c.Language1).SingleOrDefault();
                string Schoolname = (from c in db.SchoolsLists where c.S_id == row.School select
c.SchoolName).SingleOrDefault();
                string UnivName = (from c in db.UniversityLists where c.U_id == row.School select
c.UniversityName).SingleOrDefault();

```

```

        string CollegeName = (from c in db.CollegesLists where c.C_id == row.School select
c.CollegeName).SingleOrDefault();
        string groupname = (from c in db.tblPermissionGroups where c.GroupId == row.GroupId
select c.GroupName).FirstOrDefault();
        bool RecvActive = false, RecvStype = false;
        bool HeartListAdded = false, ShortListAdded = false, SentProfileAdded = false,
RcvdProfileAdded = false;

        var Recvprof = (from c in db.UserSendProfiles where c.UserId == row.UserId && c.PSId ==
userid select c).FirstOrDefault();
        if (Recvprof != null) { RecvActive = Recvprof.Active; RecvStype = Recvprof.SLType; }
        int heartcnt = (from c in db.UserCarts where c.UserId == userid && c.PSId == row.UserId
select c).Count();
        int shortcnt = (from c in db.UserShortlists where c.UserId == userid && c.PSId ==
row.UserId select c).Count();
        if (heartcnt > 0) { HeartListAdded = true; }
        if (shortcnt > 0) { ShortListAdded = true; }

        usercart.Add(new EteGetSearchResults
        {
            UserId = row.UserId,
            UserName = Decrypt(db.EncryptData(row.UserName)),
            FirstName = Decrypt(row.FirstName),
            LastName = Decrypt(row.LastName),
            EmailID = Decrypt(row.EmailID),
            ContactNum = Decrypt(row.ContactNum),
            CreatedOn = row.CreatedOn,
            Status = row.Status,
            RoleID = row.RoleID,
            GroupId = row.GroupId,
            IsRemoved = row.IsRemoved,
            Department = dept,
            Region = row.Region,
            UpdatedOn = row.UpdatedOn,
            DealerId = row.DealerId,
            Gender = row.Gender,
            Landline = Decrypt(row.Landline),
            MTongue = (int)row.MTongue,
            DateOfBirth = (DateTime)row.DateOfBirth,
            TimeOfBirth = (DateTime)row.TimeOfBirth,
            PlaceOfBirth = Decrypt(row.PlaceofBirth),
            adshowchk = (bool)row.adshowchk,
            MatchingChk = (bool)row.Matchingchk,

            userid1 = row.userid1,
            Details = Decrypt(row.Details),
            Occupation = Decrypt(row.Occupation),
            Occupationtype = Decrypt(row.Occupation_type),
        }
    );
}

```

```
Eduqualification = Decrypt(row.Edu_qualification),
Maritalstatus = row.Marital_status,
Sexornt = Sexornt,
dob = row.dob,
tob = row.tob,
pob = Decrypt(row.pob),
familynos = row.family_nos,
weburl = Decrypt(row.web_url),
adhaarid = Decrypt(row.adhaar_id),
panno = Decrypt(row.pan_no),
weeklydays = row.weekly_days,
IncomeYear = row.Income_Year,
currency = row.currency,
Religion = row.Religion,
Caste = row.Caste,
SubCaste = Decrypt(row.Sub_Caste),
Location = row.Location,
GeoLocation = Decrypt(row.GeoLocation),
Companylogo = row.Company_logo,
CreatedOn1 = row.CreatedOn1,
UpdatedOn1 = row.UpdatedOn1,
uid = row.uid,
logopath = row.logo_path,
age = row.age,
About = row.about,
MotherTongue = mothertongue,
GroupName = groupname,
Active=row.Active,
Sltype=row.SLType,
SentActive = row.Active,
SentSltype = row.SLType,
RecvActive = RecvActive,
RecvSltype = RecvStype,
HeartListAdded = HeartListAdded,
ShortListAdded = ShortListAdded,
SentProfileAdded = SentProfileAdded,
RcvdProfileAdded = RcvdProfileAdded,
School = (int)row.School,
College = (int)row.college,
University = (int)row.university,
SchoolName = Schoolname,
CollegeName = CollegeName,
UniversityName = UnivName,
```

```
});
```

```
}
```

```

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

//  throw exc;
//}
var usercart1 = GetAllUserPrivacyDetails(userid, usercart);
return usercart1;
}
public List<EteGetSearchResults> GetRcvdProfilebyUserId(int userid)
{
    List<EteGetSearchResults> usercart = new List<EteGetSearchResults>();
    //try
    {
        using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
        {
            List<GetMyRcvdProfileResult> user = db.GetMyRcvdProfile(userid, true).ToList();
            foreach (var row in user)
            {
                //var ImagePath = (from d in db.tblUserImages where d.UserId == row.UserId select
d.Images).FirstOrDefault();
                //byte[] bytes = (byte[])ImagePath.ToArray();
                //string base64String = Convert.ToBase64String(bytes, 0, bytes.Length);
                //string imageUrl = "data:image/png;base64," + base64String;
                string dept = "";
                string gender = row.Gender.ToString();
                if (gender == "W") { dept = "WOMEN PROFILES"; }
                else if (gender == "M") { dept = "MEN PROFILES"; }
                else if (gender == "B") { dept = "BI-SEXUAL PROFILES"; } else { dept = "OTHERS PROFILES"; }
                string Sexornt = "";
                if (row.Sex_ornt == 1) { Sexornt = "Straight"; } else if (row.Sex_ornt == 2) { Sexornt = "Gay";
} else { Sexornt = "Bi-Sexual"; }
                string groupname = (from c in db.tblPermissionGroups where c.GroupId == row.GroupId
select c.GroupName).FirstOrDefault();
                string mothertongue = (from c in db.Languages where c.Lan_id == row.MTongue select
c.Language1).SingleOrDefault();
                string Schoolname = (from c in db.SchoolsLists where c.S_id == row.School select
c.SchoolName).SingleOrDefault();
                string UnivName = (from c in db.UniversityLists where c.U_id == row.School select
c.UniversityName).SingleOrDefault();
                string CollegeName = (from c in db.CollegesLists where c.C_id == row.School select
c.CollegeName).SingleOrDefault();
                bool SentActive = false, SentSType = false;
                bool HeartListAdded = false, ShortListAdded = false, SentProfileAdded = false,
RcvdProfileAdded = false;

```

```

        var sentprof = (from c in db.UserSendProfiles where c.UserId == userid && c.PSId ==
row.UserId select c).FirstOrDefault();
        if (sentprof != null) { SentActive = sentprof.Active; SentSType = sentprof.SLType;
SentProfileAdded = true; }
        RcvdProfileAdded = true;
        int heartcnt = (from c in db.UserCarts where c.UserId == userid && c.PSId == row.UserId
select c).Count();
        int shortcnt = (from c in db.UserShortlists where c.UserId == userid && c.PSId ==
row.UserId select c).Count();
        if (heartcnt > 0) { HeartListAdded = true; }
        if (shortcnt > 0) { ShortListAdded = true; }

        usercart.Add(new EteGetSearchResults
        {

            UserId = row.UserId,
            UserName = Decrypt(db.EncryptData(row.UserName)),
            FirstName = Decrypt(row.FirstName),
            LastName = Decrypt(row.LastName),
            EmailID = Decrypt(row.EmailID),
            ContactNum = Decrypt(row.ContactNum),
            CreatedOn = row.CreatedOn,
            Status = row.Status,
            RoleID = row.RoleID,
            GroupId = row.GroupId,
            IsRemoved = row.IsRemoved,
            Department = dept,
            Region = row.Region,
            UpdatedOn = row.UpdatedOn,
            DealerId = row.DealerId,
            Gender = row.Gender,
            Landline = Decrypt(row.Landline),
            MTongue = (int)row.MTongue,
            DateOfBirth = (DateTime)row.DateOfBirth,
            TimeOfBirth = (DateTime)row.TimeOfBirth,
            PlaceOfBirth = Decrypt(row.PlaceofBirth),
            adshowchk = (bool)row.adshowchk,
            MatchingChk = (bool)row.Matchingchk,

            userid1 = row.userid1,
            Details = Decrypt(row.Details),
            Occupation = Decrypt(row.Occupation),
            Occupationtype = Decrypt(row.Occupation_type),
            Eduqualification = Decrypt(row.Edu_qualification),
            Martialstatus = row.Martial_status,
            Sexornt = Sexornt,
            dob = row.dob,
            tob = row.tob,

```



```
pob = Decrypt(row.pob),
familynos = row.family_nos,
weburl = Decrypt(row.web_url),
adhaarid = Decrypt(row.adhaar_id),
panno = Decrypt(row.pan_no),
weeklydays = row.weekly_days,
IncomeYear = row.Income_Year,
currency = row.currency,
Religion = row.Religion,
Caste = row.Caste,
SubCaste = Decrypt(row.Sub_Caste),
Location = row.Location,
GeoLocation = Decrypt(row.GeoLocation),
Companylogo = row.Company_logo,
CreatedOn1 = row.CreatedOn1,
UpdatedOn1 = row.UpdatedOn1,
uid = row.uid,
logopath = row.logo_path,
age = row.age,
About = row.about,
MotherTongue = mothertongue,
GroupName = groupname,
Active=row.Active,
Sltype=row.SLType,
SentActive = SentActive,
SentSltype = SentSType,
RecvActive = row.Active,
RecvSltype = row.SLType,
HeartListAdded = HeartListAdded,
ShortListAdded = ShortListAdded,
SentProfileAdded = SentProfileAdded,
RcvdProfileAdded = RcvdProfileAdded,
School=(int)row.School,
College = (int)row.college,
University = (int)row.university,
SchoolName=Schoolname,
CollegeName=CollegeName,
UniversityName=UnivName,

});

}

}
}
//catch (Exception exc)
//{
```

```

// objlog.Trace_Error(5, "DOW", "Bal.Products", "GetShortListbyUserId", exc.Message);

// throw exc;
//}
var usercart1 = GetAllUserPrivacyDetails(userid, usercart);

return usercart1;
}
public List<EteGetSearchResults> GetAcceptedProfilebyUserId(int userid)
{
List<EteGetSearchResults> usercart = new List<EteGetSearchResults>();
//try
{
using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
{
List<GetMyAcceptedListResult> user = db.GetMyAcceptedList(userid, true).ToList();
foreach (var row in user)
{
//var ImagePath = (from d in db.tblUserImages where d.UserId == row.UserId select
d.Images).FirstOrDefault();
//byte[] bytes = (byte[])ImagePath.ToArray();
//string base64String = Convert.ToBase64String(bytes, 0, bytes.Length);
//string ImageUrl = "data:image/png;base64," + base64String;
var ImagePath = (from d in db.tblUserImages where d.UserId == row.UserId select
d.Images).FirstOrDefault();
var Path = (from d in db.tblUserImages where d.UserId == row.UserId select
d.ImgPath).FirstOrDefault();
string dept = "";
string gender = row.Gender.ToString();
if (gender == "W") { dept = "WOMEN PROFILES"; }
else if (gender == "M") { dept = "MEN PROFILES"; }
else if (gender == "B") { dept = "BI-SEXUAL PROFILES"; } else { dept = "OTHERS PROFILES"; }
string Sexornt = "";
if (row.Sex_ornt == 1) { Sexornt = "Straight"; } else if (row.Sex_ornt == 2) { Sexornt = "Gay";
} else { Sexornt = "Bi-Sexual"; }
string groupname = (from c in db.tblPermissionGroups where c.GroupId == row.GroupId
select c.GroupName).FirstOrDefault();
string mothertongue = (from c in db.Languages where c.Lan_id == row.MTongue select
c.Language1).SingleOrDefault();
string Schoolname = (from c in db.SchoolsLists where c.S_id == row.School select
c.SchoolName).SingleOrDefault();
string UnivName = (from c in db.UniversityLists where c.U_id == row.School select
c.UniversityName).SingleOrDefault();
string CollegeName = (from c in db.CollegesLists where c.C_id == row.School select
c.CollegeName).SingleOrDefault();
bool SentActive = false, SentSType = false;
bool HeartListAdded = false, ShortListAdded = false, SentProfileAdded = false,
RcvdProfileAdded = false;

```

```

        var sentprof = (from c in db.UserSendProfiles where c.UserId == userid && c.PSId ==
row.UserId select c).FirstOrDefault();
        if (sentprof != null) { SentActive = sentprof.Active; SentSType = sentprof.SLType;
SentProfileAdded = true; }
        RcvdProfileAdded = true;
        int heartcnt = (from c in db.UserCarts where c.UserId == userid && c.PSId == row.UserId
select c).Count();
        int shortcnt = (from c in db.UserShortlists where c.UserId == userid && c.PSId ==
row.UserId select c).Count();
        if (heartcnt > 0) { HeartListAdded = true; }
        if (shortcnt > 0) { ShortListAdded = true; }

        usercart.Add(new EteGetSearchResults
        {

            UserId = row.UserId,
            UserName = Decrypt(db.EncryptData(row.UserName)),
            FirstName = Decrypt(row.FirstName),
            LastName = Decrypt(row.LastName),
            EmailID = Decrypt(row.EmailID),
            ContactNum = Decrypt(row.ContactNum),
            CreatedOn = row.CreatedOn,
            Status = row.Status,
            RoleID = row.RoleID,
            GroupId = row.GroupId,
            IsRemoved = row.IsRemoved,
            Department = dept,
            Region = row.Region,
            UpdatedOn = row.UpdatedOn,
            DealerId = row.DealerId,
            Gender = row.Gender,
            Landline = Decrypt(row.Landline),
            MTongue = (int)row.MTongue,
            DateOfBirth = (DateTime)row.DateOfBirth,
            TimeOfBirth = (DateTime)row.TimeOfBirth,
            PlaceOfBirth = Decrypt(row.PlaceofBirth),
            adshowchk = (bool)row.adshowchk,
            MatchingChk = (bool)row.Matchingchk,

            userid1 = row.userid1,
            Details = Decrypt(row.Details),
            Occupation = Decrypt(row.Occupation),
            Occupationtype = Decrypt(row.Occupation_type),
            Eduqualification = Decrypt(row.Edu_qualification),
            Martialstatus = row.Martial_status,
            Sexornt = Sexornt,
            dob = row.dob,
            tob = row.tob,

```

```
pob = Decrypt(row.pob),
familynos = row.family_nos,
weburl = Decrypt(row.web_url),
adhaarid = Decrypt(row.adhaar_id),
panno = Decrypt(row.pan_no),
weeklydays = row.weekly_days,
IncomeYear = row.Income_Year,
currency = row.currency,
Religion = row.Religion,
Caste = row.Caste,
SubCaste = Decrypt(row.Sub_Caste),
Location = row.Location,
GeoLocation = Decrypt(row.GeoLocation),
Companylogo = row.Company_logo,
CreatedOn1 = row.CreatedOn1,
UpdatedOn1 = row.UpdatedOn1,
uid = row.uid,
logopath = row.logo_path,
age = row.age,
About = row.about,
MotherTongue = mothertongue,
GroupName = groupname,
Active = row.Active,
Sltype = row.SLType,
SentActive = SentActive,
SentSltype = SentSType,
RecvActive = row.Active,
RecvSltype = row.SLType,
HeartListAdded = HeartListAdded,
ShortListAdded = ShortListAdded,
SentProfileAdded = SentProfileAdded,
RcvdProfileAdded = RcvdProfileAdded,
School = (int)row.School,
College = (int)row.college,
University = (int)row.university,
SchoolName = Schoolname,
CollegeName = CollegeName,
UniversityName = UnivName,
Aname=Path,

});

}

}
}
//catch (Exception exc)
//{
```

```

// objlog.Trace_Error(5, "DOW", "Bal.Products", "GetShortListbyUserId", exc.Message);

// throw exc;
//}
var usercart1 = GetAllUserPrivacyDetails(userid, usercart);
return usercart1;
}
public bool UserNameExists(string name)
{
    bool exists = false;
    using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
    {

        db.Connection.Open();
        name = db.EncryptData((name));

        var _cust = (from c in db.tblUsers where c.UserName == name select c).SingleOrDefault();

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

    }

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

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

        db.Connection.Open();

        var _cust = (from c in db.tblUsers where c.EmailID ==
name | | c.UserName==db.EncryptData(name) select c).Count();

        if (_cust >0)
        {
            exists = true;
        }

    }
}

```

```

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

    using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();

        var _cust = (from c in db.tblUsers where c.ContactNum == name select c).ToList();

        if (_cust.Count>0)
        {
            exists = true;
        }
    }

    return exists;
}
public int getuserid(string Username)
{
    int userid = 0;
    using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        Username = db.EncryptData(Encrypt(Username));

        var users = (from c in db.tblUsers where c.UserName == Username select
c.UserId).SingleOrDefault();
        if (users != null) { userid =Convert.ToInt16( users); }
    }

    return userid;
}
public string getgroupname(int groupid)
{
    string userid = "";
    using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        string groupname = (from c in db.tblPermissionGroups where c.GroupId == groupid select
c.GroupName).FirstOrDefault();

```

```

        if (groupname != null) {
            userid = groupname;
        }

    }

    return userid;

}

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; }
        UserName = Decrypt(db.EncryptData(UserName));
    }

    return UserName;
}

public string getreligion(int rid)
{
    string UserName = "";
    using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        UserName = (from c in db.Religions where c.ml_id == rid select c.ml_desc).SingleOrDefault();
        if (UserName == null) { UserName = ""; }
    }

    return UserName;
}

public string getcaste(int cid)
{
    string UserName = "";
    using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        UserName = (from c in db.Castes where c.mc_id == cid select c.mc_desc).SingleOrDefault();
        if (UserName == null) { UserName = ""; }
    }
}

```

```

    return UserName;
}
public string getmatstatus(int mid)
{
    string UserName = "";
    if (mid == 1) { UserName = "Single/Unmarried"; }
    if (mid == 2) { UserName = "Single/Divorced(no children)"; }
    if (mid == 3) { UserName = "Single/Divorced(with children)"; }
    if (mid == 4) { UserName = "Single/Widowed(no children)"; }
    if (mid == 5) { UserName = "Single/Widowed(with children)"; }
    if (mid == 6) { UserName = "Single/Seperated(no children)"; }
    if (mid == 7) { UserName = "Single/Seperated(with children)"; }
    if (mid == 8) { UserName = "Its Complicated"; }
    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 EtetblUser CheckLogin(string userNameee, string password)
{
    EtetblUser result = null;

    string pword = password;
    result = (new EtetblUser { UserName = "" });

    try
    {
        using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
        {
            userNameee = db.EncryptData(Encrypt(userNameee));
            password = db.EncryptData(Encrypt(password));
            result = (from x in db.tblUsers where x.UserName == userNameee && x.Password ==
password select new EtetblUser { UserName = x.UserName }).SingleOrDefault();

        }

    }
    catch (Exception exc)
    {
        throw exc;
    }
    return result;
}
public bool InsertUserPrivacyCheck(Etetbluser_privacy_chk 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_privacy_chk _cs = new tbluser_privacy_chk
            {

```

```
userid = data.userid,
User_privacy = data.User_privacy,
Details_chk = data.Details_chk,
Occupation_chk = data.Occupation_chk,
Occupation_type_chk = data.Occupation_type_chk,
Edu_qualification_chk = data.Edu_qualification_chk,
Marital_status_chk = data.Marital_status_chk,
Sex_ornt_chk = data.Sex_ornt_chk,
dob_chk = data.dob_chk,
tob_chk = data.tob_chk,
pob_chk = data.pob_chk,
family_nos_chk = data.family_nos_chk,
web_url_chk = data.web_url_chk,
adhaar_id_chk = data.adhaar_id_chk,
pan_no_chk = data.pan_no_chk,
weekly_days_chk = data.weekly_days_chk,
Income_Year_chk = data.Income_Year_chk,
currency_chk = data.currency_chk,
Religion_chk = data.Religion_chk,
Caste_chk = data.Caste_chk,
Sub_Caste_chk = data.Sub_Caste_chk,
Location_chk = data.Location_chk,
GeoLocation_chk = data.GeoLocation_chk,
Gender_Search_chk = data.Gender_Search_chk,
Age_from_chk = data.Age_from_chk,
Age_to_chk = data.Age_to_chk,
Religion_search_chk = data.Religion_search_chk,
Caste_search_chk = data.Caste_search_chk,
Area_search_chk = data.Area_search_chk,
MTongue_search_chk = data.MTongue_search_chk,
GeoLoc_search_chk = data.GeoLoc_search_chk,
Religions_Choice = data.Religions_Choice,
Caste_Choice = data.Caste_Choice,
Areas_Choice = data.Areas_Choice,
Cities_Choice = data.Cities_Choice,
Countries_Choice = data.Countries_Choice,
Languages_Choice = data.Languages_Choice,
uid = data.uid,
CreatedOn = data.CreatedOn,
UpdatedOn = data.UpdatedOn,
school_chk=data.school_chk,
college_chk=data.college_chk,
university_chk=data.university_chk,
about_chk=data.about_chk,
Photos_chk=data.Photos_chk,

};
db.tbluser_privacy_chks.InsertOnSubmit(_cs);
```

```

        db.SubmitChanges();
        db.Transaction.Commit();
        var COid = (from a in db.tbluser_privacy_chks where a.userid == data.userid &&
a.User_privacy == data.User_privacy 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.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 UpdatePass()
{
    bool result = false;

    using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();
        var _user = (from c in db.tblUsers select c).ToList();

        foreach (var row in _user)
        {
            //string pass = Encrypt(row.Password);

            //row.Landline = db.EncryptData(Encrypt(row.UserName));
            //row.Password = db.EncryptData(pass);

```

```

        //row.FirstName = Encrypt(row.FirstName);
        //row.LastName = Encrypt(row.LastName);
        //row.Landline = Encrypt(row.Landline);
        //row.EmailID = Encrypt(row.EmailID);
        //row.PlaceofBirth = Encrypt(row.PlaceofBirth);
        //row.ContactNum = Encrypt(row.ContactNum);
    }
    db.SubmitChanges();
    db.Transaction.Commit();
    //db.Transaction = db.Connection.BeginTransaction();

    //var _userp = (from c in db.tbluser_profiles select c).ToList();
    //foreach (var row in _userp)
    //{
    //    row.about = Encrypt(row.about);
    //    row.Details = Encrypt(row.Details);
    //    row.Occupation = Encrypt(row.Occupation);
    //    row.Occupation_type = Encrypt(row.Occupation_type);
    //    row.pan_no = Encrypt(row.pan_no);
    //    row.adhaar_id = Encrypt(row.adhaar_id);
    //    row.Edu_qualification = Encrypt(row.Edu_qualification);
    //    row.pob = Encrypt(row.pob);
    //    row.web_url = Encrypt(row.web_url);
    //    row.Sub_Caste = Encrypt(row.Sub_Caste);

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

}
return result;
}
public bool Insert(EtetblUser data, Etetbluserlog Log)
{
    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();
        data.UserName = db.EncryptData((data.UserName));
        try
        {
            {
                tblUser_cs = new tblUser
                {
                    UserId = userid,

```

```

    UserName =data.UserName,
    Password =db.EncryptData( data.Password),
    FirstName =Encrypt( data.FirstName),
    LastName = Encrypt(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,
    Gender = data.Gender,
    Landline = Encrypt(data.Landline),
    MTongue = data.MTongue,
    DateOfBirth=data.DateOfBirth,
    TimeOfBirth=data.TimeOfBirth,
    PlaceOfBirth=Encrypt(data.PlaceOfBirth),
    adshowchk=(bool)data.adshowchk,
    Matchingchk = (bool)data.MatchingChk,
};
db.tblUsers.InsertOnSubmit(_cs);

db.SubmitChanges();
db.Transaction.Commit();
//db.Transaction = db.Connection.BeginTransaction();
//int Ids =(int) (from c in db.tblUserLogs orderby c.LoginId descending select
c.LoginId).FirstOrDefault();
//Ids++;
//tblUserLog _log = new tblUserLog
//{
//  LoginId = Ids,
//  UserId = userid,
//  LoggedCity = Log.LoggedCity,
//  LoggedCountry = Log.LoggedCountry,
//  PerformAction = Log.PerformAction,
//  PerformActionTime = Log.PerformActionTime,
//  LoggedIPAddress = Log.LoggedIPAddress,

//};
//db.tblUserLogs.InsertOnSubmit(_log);

//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 InserttbluserLog(EteTbluserlog data)
{
    bool result = false;
    using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        db.Transaction = db.Connection.BeginTransaction();

        try
        {
            tblUserLog _cs = new tblUserLog
            {
                LoginId = data.LoginId,
                UserId = data.UserId,
                LoggedCity = data.LoggedCity,
                LoggedCountry = data.LoggedCountry,
                PerformAction = data.PerformAction,
                PerformActionTime = data.PerformActionTime,
            }

```

```

        LoggedIPAddress = data.LoggedIPAddress,

};
db.tblUserLogs.InsertOnSubmit(_cs);

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

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

return result;
}
public bool DeleteCart(EteUserCart Kart)
{

{
    bool result = false;

using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
{
    db.Connection.Open();
    try
    {
        var _ord = db.UserCarts.Where(e => e.UserId == Kart.UserId && e.PSId==Kart.PSId);

        if (_ord != null)
        {
            db.Transaction = db.Connection.BeginTransaction();
            db.UserCarts.DeleteAllOnSubmit(_ord);
            db.SubmitChanges();
            db.Transaction.Commit();
            result = true;
        }
    }
}

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

```

```

    }
}

return result;
}
}
public bool DeleteAddress(int Userid, string Addtype)
{

{
    bool result = false;

using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
{
    db.Connection.Open();
    try
    {
        var _ord = db.tbluser_addresses.Where(e => e.userid == Userid);

        if (_ord != null)
        {
            db.Transaction = db.Connection.BeginTransaction();
            db.tbluser_addresses.DeleteAllOnSubmit(_ord);
            db.SubmitChanges();
            db.Transaction.Commit();
            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,
        pincode=data.pancode,

    };
    db.tbluser_addresses.InsertOnSubmit(_cs);

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

return result;
}
public bool UpdateUser_options(Etetbluser_options data)
{
    Etetbluser_options odata = new Etetbluser_options();
    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_options.Where(e => e.userid == data.userid).SingleOrDefault();

```

```

if (_user == null)
{

tbluser_option user = new tbluser_option
{
    userid = data.userid,
    Gender_Search = data.Gender_Search,
    Age_from = data.Age_from,
    Age_to = data.Age_to,
    Religion_search = data.Religion_search,
    Caste_search = data.Caste_search,
    Area_search = data.Area_search,
    MTongue_search = data.MTongue_search,
    CreatedOn = DateTime.Now,
    Geoloc_search = "",

};
db.tbluser_options.InsertOnSubmit(user);

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

}
else
{

odata = new Etetbluser_options
{
    userid = _user.userid,
    Gender_Search = _user.Gender_Search,
    Age_from = _user.Age_from,
    Age_to = _user.Age_to,
    Religion_search = _user.Religion_search,
    Caste_search = _user.Caste_search,
    Area_search = _user.Area_search,
    MTongue_search = _user.MTongue_search,

};
if (odata.CreatedOn == null) { odata.CreatedOn = DateTime.Today; }
if (data.CreatedOn == null) { data.CreatedOn = odata.CreatedOn; }
//_user.UserId = data.UserId;
_user.Gender_Search = data.Gender_Search;
_user.Age_from = data.Age_from;
_user.Age_to = data.Age_to;
_user.Religion_search = data.Religion_search;

```

```

    _user.Caste_search = data.Caste_search;
    _user.Area_search = data.Area_search;
    _user.MTongue_search = data.MTongue_search;

    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 = 1,
    };
    bool res = new BalAuditTrail().Insert(at);
}
}
//catch (Exception exc)
//{

//    throw exc;
//}

return result;
}
public bool UpdatePrivacyCheck(Etetbluser_privacy_chk data)
{
    Etetbluser_privacy_chk odata = new Etetbluser_privacy_chk();

```

```

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_privacy_chks.Where(e => e.userid == data.userid && e.User_privacy
== data.User_privacy).SingleOrDefault();

        if (_user == null)
        {
            throw (new ApplicationException("Unable to find the User record"));
        }
        odata = new Etetbluser_privacy_chk
        {
            userid = data.userid,
            User_privacy = data.User_privacy,
            Details_chk = data.Details_chk,
            Occupation_chk = data.Occupation_chk,
            Occupation_type_chk = data.Occupation_type_chk,
            Edu_qualification_chk = data.Edu_qualification_chk,
            Martial_status_chk = data.Martial_status_chk,
            Sex_ornt_chk = data.Sex_ornt_chk,
            dob_chk = data.dob_chk,
            tob_chk = data.tob_chk,
            pob_chk = data.pob_chk,
            family_nos_chk = data.family_nos_chk,
            web_url_chk = data.web_url_chk,
            adhaar_id_chk = data.adhaar_id_chk,
            pan_no_chk = data.pan_no_chk,
            weekly_days_chk = data.weekly_days_chk,
            Income_Year_chk = data.Income_Year_chk,
            currency_chk = data.currency_chk,
            Religion_chk = data.Religion_chk,
            Caste_chk = data.Caste_chk,
            Sub_Caste_chk = data.Sub_Caste_chk,
            Location_chk = data.Location_chk,
            GeoLocation_chk = data.GeoLocation_chk,
            Gender_Search_chk = data.Gender_Search_chk,
            Age_from_chk = data.Age_from_chk,
            Age_to_chk = data.Age_to_chk,
            Religion_search_chk = data.Religion_search_chk,
            Caste_search_chk = data.Caste_search_chk,
            Area_search_chk = data.Area_search_chk,

```

```
MTongue_search_chk = data.MTongue_search_chk,
GeoLoc_search_chk = data.GeoLoc_search_chk,
Religions_Choice = data.Religions_Choice,
Caste_Choice = data.Caste_Choice,
Areas_Choice = data.Areas_Choice,
Cities_Choice = data.Cities_Choice,
Countries_Choice = data.Countries_Choice,
Languages_Choice = data.Languages_Choice,
uid = data.uid,
CreatedOn = data.CreatedOn,
UpdatedOn = data.UpdatedOn,

};
if (odata.CreatedOn == null) { odata.CreatedOn = DateTime.Today; }
if (data.CreatedOn == null) { data.CreatedOn = odata.CreatedOn; }
_user.userid = data.userid;
_user.User_privacy = data.User_privacy;
_user.Details_chk = data.Details_chk;
_user.Occupation_chk = data.Occupation_chk;
_user.Occupation_type_chk = data.Occupation_type_chk;
_user.Edu_qualification_chk = data.Edu_qualification_chk;
_user.Martial_status_chk = data.Martial_status_chk;
_user.Sex_ornt_chk = data.Sex_ornt_chk;
_user.dob_chk = data.dob_chk;
_user.tob_chk = data.tob_chk;
_user.pob_chk = data.pob_chk;
_user.family_nos_chk = data.family_nos_chk;
_user.web_url_chk = data.web_url_chk;
_user.adhaar_id_chk = data.adhaar_id_chk;
_user.pan_no_chk = data.pan_no_chk;
_user.weekly_days_chk = data.weekly_days_chk;
_user.Income_Year_chk = data.Income_Year_chk;
_user.currency_chk = data.currency_chk;
_user.Religion_chk = data.Religion_chk;
_user.Caste_chk = data.Caste_chk;
_user.Sub_Caste_chk = data.Sub_Caste_chk;
_user.Location_chk = data.Location_chk;
_user.GeoLocation_chk = data.GeoLocation_chk;
_user.Gender_Search_chk = data.Gender_Search_chk;
_user.Age_from_chk = data.Age_from_chk;
_user.Age_to_chk = data.Age_to_chk;
_user.Religion_search_chk = data.Religion_search_chk;
_user.Caste_search_chk = data.Caste_search_chk;
_user.Area_search_chk = data.Area_search_chk;
_user.MTongue_search_chk = data.MTongue_search_chk;
_user.GeoLoc_search_chk = data.GeoLoc_search_chk;
_user.Religions_Choice = data.Religions_Choice;
_user.Caste_Choice = data.Caste_Choice;
```

```

    _user.Areas_Choice = data.Areas_Choice;
    _user.Cities_Choice = data.Cities_Choice;
    _user.Countries_Choice = data.Countries_Choice;
    _user.Languages_Choice = data.Languages_Choice;
    _user.uid = data.uid;
    //_user.CreatedOn = data.CreatedOn;
    _user.UpdatedOn = data.UpdatedOn;
    _user.Photos_chk = data.Photos_chk;
    _user.school_chk = data.school_chk;
    _user.college_chk = data.college_chk;
    _user.university_chk = data.university_chk;
    _user.about_chk = data.about_chk;

    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 = 1, // data.PageId,
    };
    bool res = new BalAuditTrail().Insert(at);
}
//catch (Exception exc)
//{{

//  throw exc;
//}}

```

```

        return result;
    }
    public bool GetPrivacyCount(int userid, int UserPrivacy)
    {
bool ret=false;
using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
{
    db.Connection.Open();
    db.Transaction = db.Connection.BeginTransaction();

    //try
    {

        int data = db.tbluser_privacy_chks.Where(e => e.userid == userid && e.User_privacy ==
UserPrivacy).Count();

        if (data > 0)
        {
            ret = true;
        }
    }
}
return ret;
    }

    public Etetbluser_privacy_chk GetPrivacyCheck( int userid,int UserPrivacy)
    {
        Etetbluser_privacy_chk odata = new Etetbluser_privacy_chk();
        using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
        {
            db.Connection.Open();
            db.Transaction = db.Connection.BeginTransaction();

            //try
            {

                var data = db.tbluser_privacy_chks.Where(e => e.userid == userid && e.User_privacy ==
UserPrivacy).SingleOrDefault();

                if (data == null)
                {
                    odata = new Etetbluser_privacy_chk();
                    return odata;
                }
                odata = new Etetbluser_privacy_chk
                {

```

```
userid = data.userid,
User_privacy = data.User_privacy,
Details_chk = data.Details_chk,
Occupation_chk = data.Occupation_chk,
Occupation_type_chk = data.Occupation_type_chk,
Edu_qualification_chk = data.Edu_qualification_chk,
Marital_status_chk = data.Marital_status_chk,
Sex_ornt_chk = data.Sex_ornt_chk,
dob_chk = data.dob_chk,
tob_chk = data.tob_chk,
pob_chk = data.pob_chk,
family_nos_chk = data.family_nos_chk,
web_url_chk = data.web_url_chk,
adhaar_id_chk = data.adhaar_id_chk,
pan_no_chk = data.pan_no_chk,
weekly_days_chk = data.weekly_days_chk,
Income_Year_chk = data.Income_Year_chk,
currency_chk = data.currency_chk,
Religion_chk = data.Religion_chk,
Caste_chk = data.Caste_chk,
Sub_Caste_chk =(bool) data.Sub_Caste_chk,
Location_chk = data.Location_chk,
GeoLocation_chk = data.GeoLocation_chk,
Gender_Search_chk = data.Gender_Search_chk,
Age_from_chk = data.Age_from_chk,
Age_to_chk = data.Age_to_chk,
Religion_search_chk = data.Religion_search_chk,
Caste_search_chk = data.Caste_search_chk,
Area_search_chk = data.Area_search_chk,
MTongue_search_chk = data.MTongue_search_chk,
GeoLoc_search_chk = data.GeoLoc_search_chk,
Religions_Choice = data.Religions_Choice,
Caste_Choice = data.Caste_Choice,
Areas_Choice = data.Areas_Choice,
Cities_Choice = data.Cities_Choice,
Countries_Choice = data.Countries_Choice,
Languages_Choice = data.Languages_Choice,
uid = data.uid,
CreatedOn = data.CreatedOn,
UpdatedOn = data.UpdatedOn,
about_chk =Convert.ToBoolean( data.about_chk),
school_chk =(bool) data.school_chk,
college_chk = (bool)data.college_chk,
university_chk = (bool)data.university_chk,
Photos_chk = (bool)data.Photos_chk,
```

```
};
```

```
}
```



```

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

        //  throw exc;
        //}
    }

    return odata;
}
public bool Updatebyid(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
        {
            data.UserName = db.EncryptData(Encrypt(data.UserName));
            data.Password = db.EncryptData(Encrypt(data.Password));

            var _user = db.tblUsers.Where(e => e.UserId == data.UserId).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,
// Gender = _user.Gender,
// Landline = _user.Landline,
// MTongue = (int)_user.MTongue,
// UId = _user.UserId,
// DateOfBirth = (DateTime)_user.DateOfBirth,
// TimeOfBirth = (DateTime)_user.TimeOfBirth,
// PlaceOfBirth = _user.PlaceofBirth,
// adshowchk = (bool)_user.adshowchk,
// MatchingChk = (bool)_user.Matchingchk,

//};
//if (odata.CreatedOn == null) { odata.CreatedOn = DateTime.Today; }
//if (data.CreatedOn == null) { data.CreatedOn = odata.CreatedOn; }
////_user.UserId = data.UserId;

_user.UserName = data.UserName;
//if (data.Password != "" && data.Password != null)
{
    _user.Password = data.Password;
}

//_user.FirstName = Encrypt(data.FirstName);

//_user.LastName = Encrypt(data.LastName);

//_user.EmailID = Encrypt(data.EmailID);

//_user.ContactNum = Encrypt(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.Gender = data.Gender;
//_user.Landline = Encrypt(data.Landline);
//_user.MTongue = (int)data.MTongue;

```

```

    //_user.DateOfBirth = data.DateOfBirth;
    //_user.TimeOfBirth = data.TimeOfBirth;
    //_user.PlaceOfBirth = Encrypt(data.PlaceOfBirth);
    //_user.adshowchk = (bool)data.adshowchk;
    //_user.Matchingchk = (bool)data.MatchingChk;

    //_user.UpdatedOn = DateTime.Now;
    //_user.DealerId = data.DealerId;
    //_user.Gender = data.Gender;

    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 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
        {
            data.UserName = db.EncryptData(Encrypt(data.UserName));
            data.Password = db.EncryptData(Encrypt(data.Password));
            var _user = db.tblUsers.Where(e => e.UserId == data.UserId).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,
                Gender = _user.Gender,
                Landline = _user.Landline,
                MTongue = (int)_user.MTongue,
                UId = _user.UserId,
                DateOfBirth = (DateTime) _user.DateOfBirth,
                TimeOfBirth = (DateTime)_user.TimeOfBirth,
                PlaceOfBirth = _user.PlaceOfBirth,
                adshowchk = (bool)_user.adshowchk,
            }
        }
    }
}

```

```

    MatchingChk = (bool)_user.Matchingchk,

};
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 name saving is not required
// _user.Password = data.Password; //password save is disabled

_user.FirstName = Encrypt(data.FirstName);

_user.LastName = Encrypt(data.LastName);

_user.EmailID = Encrypt(data.EmailID);

_user.ContactNum = Encrypt(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.Gender = data.Gender;
if (Encrypt(data.Landline) == null) { _user.Landline = "0"; } else { _user.Landline =
Encrypt(data.Landline); }
_user.MTongue = (int)data.MTongue;
//if (data.DateOfBirth > Convert.ToDateTime("01/01/1901"))
{ //date of birth is shifted to profile data
    //_user.DateOfBirth = data.DateOfBirth;
    //_user.TimeOfBirth = data.TimeOfBirth;
}
//_user.PlaceOfBirth = Encrypt(data.PlaceOfBirth);
_user.adshowchk = (bool)data.adshowchk;
_user.Matchingchk = (bool)data.MatchingChk;

_user.UpdatedOn = DateTime.Now;
_user.DealerId = data.DealerId;
_user.Gender = data.Gender;

```

```

        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 bool Updateuserbak(EtetblUser data)
{
    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.UserId == data.UserId).SingleOrDefault();

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

    _user.UserName = data.UserName;
    _user.Password = data.Password;

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

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

    throw exc;
}
}

return result;
}
public Etetbluser_options getuserchoies(int userid)
{
    Etetbluser_options result = new Etetbluser_options();
    using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var users = (from c in db.tbluser_options where c.userid == userid select c).ToList();
        foreach (var row in users)
        {
            result = new Etetbluser_options
            {
                userid = row.userid,
                Gender_Search = row.Gender_Search,
                Age_from = row.Age_from,
                Age_to = row.Age_to,
            }
        }
    }
}

```

```

        Religion_search = row.Religion_search,
        Caste_search = row.Caste_search,
        Area_search = row.Area_search,
        MTongue_search = row.MTongue_search,
        GeoLoc_search = row.GeoLoc_search,
        CreatedOn = row.CreatedOn,

        };
        break;
    }
}

return result;
}
public EtetblUser getuser(string UserName)
{
    EtetblUser result = new EtetblUser();
    using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        UserName = db.EncryptData(Encrypt(UserName)); ;
        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 = Decrypt(db.EncryptData(row.UserName)),
                Password = Decrypt(db.EncryptData(row.Password)),
                FirstName = Decrypt(row.FirstName),
                LastName = Decrypt(row.LastName),
                EmailID = Decrypt(row.EmailID),
                ContactNum = Decrypt(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,
                Gender = row.Gender,
            };
            break;
        }
    }
}

```



```

    return result;
}
public EtetblUser getuserbyemail(string email)
{
    EtetblUser result = new EtetblUser();
    using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        email = Encrypt(email);
        //UserName = db.EncryptData(Encrypt(UserName)); ;
        var users = (from c in db.tblUsers where c.EmailID == email select c).ToList();
        foreach (var row in users)
        {
            result = new EtetblUser
            {
                UserId = row.UserId,
                UserName = Decrypt(db.EncryptData(row.UserName)),
                Password = Decrypt(db.EncryptData(row.Password)),
                FirstName = Decrypt(row.FirstName),
                LastName = Decrypt(row.LastName),
                EmailID = Decrypt(row.EmailID),
                ContactNum = Decrypt(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,
                Gender = row.Gender,
            };
            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 = Decrypt(db.EncryptData(row.UserName)),
            //Password = row.Password,
            FirstName = Decrypt( row.FirstName),
            LastName = Decrypt(row.LastName),
            EmailID = Decrypt(row.EmailID),
            ContactNum = Decrypt(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,
            Gender = row.Gender,
            Landline = Decrypt(row.Landline),
            MTongue = (int)row.MTongue,
            DateOfBirth =(DateTime) row.DateOfBirth,
            TimeOfBirth = (DateTime)row.TimeOfBirth,
            PlaceOfBirth =Decrypt( row.PlaceofBirth),
            adshowchk = (bool)row.adshowchk,
            MatchingChk = (bool)row.Matchingchk,
        };
        break;
    }
}

return result;
}

public Etetbluser_options Gettbluser_options(int UserId)
{
    Etetbluser_options eopt=new Etetbluser_options();
    using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var opt = (from c in db.tbluser_options where c.userid == UserId select c).SingleOrDefault();
        if (opt != null)
        {
            eopt.Gender_Search = opt.Gender_Search;
            eopt.GeoLoc_search = opt.GeoLoc_search;
            eopt.MTongue_search = opt.MTongue_search;
            eopt.Religion_search = opt.Religion_search;
            eopt.Age_from = opt.Age_from;
        }
    }
}

```

```

        eopt.Age_to = opt.Age_to;
        eopt.Area_search = opt.Area_search;
        eopt.Caste_search = opt.Caste_search;

        ;
    }
}
return eopt;
}
public GetSearchResultsResult GetUserPrivacyDetails(int userid, int psid, GetSearchResultsResult
userdet)
{
    Etetbluser_privacy_chk pchk = new Etetbluser_privacy_chk();
    pchk = GetPrivacyCheck(userid, psid);
    if (pchk.User_privacy == 0)
    {
        pchk = GetPrivacyCheck(userid, 0);
    }
    int i = 0;

    if (pchk.about_chk) { userdet.about = "Data with held on user Request"; i++; }
    if (pchk.Caste_chk) { userdet.Caste = 0; i++; }
    if (pchk.college_chk) { userdet.college = 0; i++; }
    if (pchk.Details_chk) { userdet.Details = "Data with held on user Request"; i++; }
    if (pchk.dob_chk) { userdet.dob = Convert.ToDateTime("1900/01/01"); i++; }
    if (pchk.Edu_qualification_chk) { userdet.Edu_qualification = "Data with held on user Request"; i++; }
    if (pchk.family_nos_chk) { userdet.family_nos = 0; i++; }
    if (pchk.GeoLocation_chk) { userdet.GeoLocation = "Data with held on user Request"; i++; }
    if (pchk.Income_Year_chk) { userdet.Income_Year = 0; i++; }
    if (pchk.Location_chk) { userdet.Location = "Data with held on user Request"; }
    if (pchk.Martial_status_chk) { userdet.Martial_status = 0; i++; }
    if (pchk.Occupation_chk) { userdet.Occupation = "Data with held on user Request"; i++; }
    if (pchk.Occupation_type_chk) { userdet.Occupation_type = "Data with held on user Request"; i++; }
    if (pchk.pan_no_chk) { userdet.pan_no = "Data with held on user Request"; i++; }
    if (pchk.pob_chk) { userdet.pob = "Data with held on user Request"; i++; }
    if (pchk.Religion_chk) { userdet.Religion = 0; i++; }
    if (pchk.school_chk) { userdet.School = 0; i++; }
    if (pchk.Sex_ornt_chk) { userdet.Sex_ornt = 0; i++; }
    if (pchk.Sub_Caste_chk) { userdet.Sub_Caste = "Data with held on user Request"; i++; }
    if (pchk.tob_chk) { userdet.tob = "Data with held on user Request"; i++; }
    if (pchk.university_chk) { userdet.university = 0; i++; }
    if (pchk.web_url_chk) { userdet.web_url = "Data with held on user Request"; i++; }
    if (pchk.weekly_days_chk) { userdet.weekly_days = 0; i++; }
    userdet.RoleID = i;
    return userdet;
}
public Etetbluser_profile GetUserProfilePrivacyDetails(int userid, int psid, Etetbluser_profile userdet)

```

```

{
    Etetbluser_privacy_chk pchk = new Etetbluser_privacy_chk();
    pchk = GetPrivacyCheck( psid,userid);
    int i = 0;
    if (pchk.User_privacy == 0)
    {
        pchk = GetPrivacyCheck(psid, 0);
    }
    if (pchk.about_chk) { userdet.About = "Data with held on user Request"; i++; }
    if (pchk.Caste_chk) { userdet.Caste = 0; i++; }
    if (pchk.college_chk) { userdet.CollegeName = "Data with held on user Request"; i++; }
    if (pchk.Details_chk) { userdet.Details = "Data with held on user Request"; i++; }
    if (pchk.dob_chk) { userdet.dob = Convert.ToDateTime("1900/01/01"); i++; }
    if (pchk.Edu_qualification_chk) { userdet.Eduqualification = "Data with held on user Request"; i++; }
    if (pchk.family_nos_chk) { userdet.familynos = 0; i++; }
    if (pchk.GeoLocation_chk) { userdet.GeoLocation = "Data with held on user Request"; i++; }
    if (pchk.Income_Year_chk) { userdet.Income_year = 0; i++; }
    if (pchk.Location_chk) { userdet.Location = "Data with held on user Request"; i++; }
    if (pchk.Martial_status_chk) { userdet.Martialstatus = 0; i++; }
    if (pchk.Occupation_chk) { userdet.Occupation = "Data with held on user Request"; i++; }
    if (pchk.Occupation_type_chk) { userdet.Occupationtype = "Data with held on user Request"; i++; }
    if (pchk.pan_no_chk) { userdet.panno = "Data with held on user Request"; i++; }
    if (pchk.pob_chk) { userdet.pob = "Data with held on user Request"; i++; }
    if (pchk.Religion_chk) { userdet.Religion = 0; i++; }
    if (pchk.school_chk) { userdet.SchoolName = "Data with held on user Request"; i++; }
    if (pchk.Sex_ornt_chk) { userdet.StrSexornt = "Data with held on user Request"; i++; }
    if (pchk.Sub_Caste_chk) { userdet.Sub_Caste = "Data with held on user Request"; i++; }
    if (pchk.tob_chk) { userdet.tob = "Data with held on user Request"; i++; }
    if (pchk.university_chk) { userdet.UniversityName = "Data with held on user Request"; i++; }
    if (pchk.web_url_chk) { userdet.weburl = "Data with held on user Request"; i++; }
    if (pchk.weekly_days_chk) { userdet.weeklydays = 0; i++; }

    if (i > 0) { userdet.Privacy = true; }

    return userdet;
}
public List<EteGetSearchResults> GetAllUserPrivacyDetails(int userid, List<EteGetSearchResults>
userdets)
{
    foreach (var userdet in userdets)
    {
        int psid = userdet.UserId;
        int i = 0;
        Etetbluser_privacy_chk pchk = new Etetbluser_privacy_chk();
        pchk = GetPrivacyCheck(psid,userid );
        if (pchk.User_privacy == 0)
        {
            pchk = GetPrivacyCheck(psid,0);

```

```

    }
    if (pchk.about_chk) { userdet.About = "Data with held on user Request"; i++; }
    if (pchk.Caste_chk) { userdet.Caste = 0; i++; }
    if (pchk.college_chk) { userdet.College = 0; userdet.CollegeName = "Data with held on user
Request"; i++; }
    if (pchk.Details_chk) { userdet.Details = "Data with held on user Request"; i++; }
    if (pchk.dob_chk) { userdet.dob = Convert.ToDateTime("1900/01/01"); i++; }
    if (pchk.Edu_qualification_chk) { userdet.Eduqualification = "Data with held on user Request"; i++; }
    if (pchk.family_nos_chk) { userdet.familynos = 0; i++; }
    if (pchk.GeoLocation_chk) { userdet.GeoLocation = "Data with held on user Request"; i++; }
    if (pchk.Income_Year_chk) { userdet.IncomeYear = 0; i++; }
    if (pchk.Location_chk) { userdet.Location = "Data with held on user Request"; i++; }
    if (pchk.Martial_status_chk) { userdet.Martialstatus = 0; i++; }
    if (pchk.Occupation_chk) { userdet.Occupation = "Data with held on user Request"; i++; }
    if (pchk.Occupation_type_chk) { userdet.Occupationtype = "Data with held on user Request"; i++; }
    if (pchk.pan_no_chk) { userdet.panno = "Data with held on user Request"; i++; }
    if (pchk.pob_chk) { userdet.pob = "Data with held on user Request"; i++; }
    if (pchk.Religion_chk) { userdet.Religion = 0; i++; }
    if (pchk.school_chk) { userdet.School = 0; userdet.SchoolName = "Data with held on user Request";
i++; }
    if (pchk.Sex_ornt_chk) { userdet.Sexornt = "Data with held on user Request"; i++; }
    if (pchk.Sub_Caste_chk) { userdet.SubCaste = "Data with held on user Request"; i++; }
    if (pchk.tob_chk) { userdet.tob = "Data with held on user Request"; i++; }
    if (pchk.university_chk) { userdet.University = 0; userdet.UniversityName = "Data with held on user
Request"; i++; }
    if (pchk.web_url_chk) { userdet.weburl = "Data with held on user Request"; i++; }
    if (pchk.weekly_days_chk) { userdet.weeklydays = 0; i++; }
    userdet.RoleID= i;

}
//int i = userdets.Count();
//if (i > 0)
//{
// while (i < 5)
// {
// userdets.Add(new EteGetSearchResults
// {
// userid1 = 0,
// UserName = "",
// });
// i++;
// }
// }
return userdets;
}

```

```

public List<EteGetSearchResults> getsearchprofile(int userid, string searchstr, string Groupid,
Etetbluser_options opt)
{
    string gender = "W";
    string age_from = "20";
    string age_to = "30";
    string religion = "0";
    string caste = "0";
    string area = "";
    string mtongue = "0";
    string geoloc = "";
    string dept = "";
    List<EteGetSearchResults> result = new List<EteGetSearchResults>();
    using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        if (opt != null)
        {
            gender = opt.Gender_Search;
            age_from = opt.Age_from.ToString();
            age_to = opt.Age_to.ToString();
            religion = opt.Religion_search.ToString();
            caste = opt.Caste_search.ToString();
            area = opt.Location_search.ToString();
            mtongue = opt.MTongue_search.ToString();
            geoloc = opt.GeoLoc_search;
            if (geoloc == null) { geoloc = ""; }
            dept = "";
        }
        List<GetSearchResultsResult> users;
        if (gender == "W") { dept = "WOMEN PROFILES"; } else if (gender == "M") { dept = "MEN
PROFILES"; }
        else if (gender == "B") { dept = "BI-SEXUAL PROFILES"; } else { dept = "PROFILES"; }
        users= db.GetSearchResults(searchstr, gender, age_from, age_to, religion, caste, area,
mtongue, geoloc,Groupid).ToList();
        if (users.Count == 0) {
            users = db.GetSearchResults(searchstr, gender, "0", "0", "0", "0", "", "0", "",
Groupid).ToList();
        }

        foreach (var row in users)
        {
            //System.Data.Linq.Binary Image = (from d in db.tblUserImages where d.UserId ==
row.UserId select d.Images).FirstOrDefault();
            //if (Image != null)
            //{
            //    byte[] bytes = (byte[])Image.ToArray();
            //    string base64String = Convert.ToBase64String(bytes, 0, bytes.Length);

```

```

// string imageUrl = "data:image/png;base64," + base64String;
//}

string groupname = (from c in db.tblPermissionGroups where c.GroupId == row.GroupId
select c.GroupName).FirstOrDefault() ;
if (row.UserId != userid)
{ //change the value of row to row1 here for privacy check
string Sexornt = "";
if (row.Sex_ornt == 1) { Sexornt = "Straight"; } else if (row.Sex_ornt == 2) { Sexornt =
"Gay"; } else { Sexornt = "Bi-Sexual"; }
string mothertongue = (from c in db.Languages where c.Lan_id == row.MTongue select
c.Language1).SingleOrDefault();
bool SentActive = false, SentSType = false, RecvActive = false, RecvStype = false;
bool HeartListAdded = false, ShortListAdded = false, SentProfileAdded = false,
RcvdProfileAdded = false;
var sentprof = (from c in db.UserSendProfiles where c.UserId == userid && c.PSId ==
row.UserId select c).FirstOrDefault();
if (sentprof != null) { SentActive = sentprof.Active; SentSType = sentprof.SLType;
SentProfileAdded = true; }
var Recvprof = (from c in db.UserSendProfiles where c.UserId == row.UserId && c.PSId
== userid select c).FirstOrDefault();
if (Recvprof != null) { RecvActive = Recvprof.Active; RecvStype = Recvprof.SLType;
RcvdProfileAdded = true; }
int heartcnt = (from c in db.UserCarts where c.UserId == userid && c.PSId == row.UserId
select c).Count();
int shortcnt = (from c in db.UserShortlists where c.UserId == userid && c.PSId ==
row.UserId select c).Count();
if (heartcnt > 0) { HeartListAdded = true; }
if (shortcnt > 0) { ShortListAdded = true; }
string Schoolname = (from c in db.SchoolsLists where c.S_id == row.School select
c.SchoolName).SingleOrDefault();
string UnivName = (from c in db.UniversityLists where c.U_id == row.School select
c.UniversityName).SingleOrDefault();
string CollegeName = (from c in db.CollegesLists where c.C_id == row.School select
c.CollegeName).SingleOrDefault();
result.Add(new EteGetSearchResults
{
    UserId = row.UserId,
    UserName = Decrypt(db.EncryptData(row.UserName)),
    FirstName = Decrypt( row.FirstName),
    LastName = Decrypt(row.LastName),
    EmailID = Decrypt(row.EmailID),
    ContactNum = Decrypt(row.ContactNum),
    CreatedOn = row.CreatedOn,
    Status = row.Status,
    RoleID = row.RoleID,
    GroupId = row.GroupId,
    IsRemoved = row.IsRemoved,

```

Department = dept,
Region = row.Region,
UpdatedOn = row.UpdatedOn,
DealerId = row.DealerId,
Gender = row.Gender,
Landline = Decrypt(row.Landline),
MTongue = (int)row.MTongue,
DateOfBirth = (DateTime)row.DateOfBirth,
TimeOfBirth = (DateTime)row.TimeOfBirth,
PlaceOfBirth = Decrypt(row.PlaceofBirth),
adshowchk = (bool)row.adshowchk,
MatchingChk = (bool)row.Matchingchk,

userid1 = row.userid1,
Details = Decrypt(row.Details),
Occupation = Decrypt(row.Occupation),
Occupationtype = Decrypt(row.Occupation_type),
Eduqualification = Decrypt(row.Edu_qualification),
Martialstatus = row.Martial_status,
Sexornt = Sexornt,
dob = row.dob,
tob = row.tob,
pob = Decrypt(row.pob),
familynos = row.family_nos,
weburl = Decrypt(row.web_url),
adhaarid = Decrypt(row.adhaar_id),
panno = Decrypt(row.pan_no),
weeklydays = row.weekly_days,
IncomeYear = row.Income_Year,
currency = row.currency,
Religion = row.Religion,
Caste = row.Caste,
SubCaste = Decrypt(row.Sub_Caste),
Location = row.Location,
GeoLocation = Decrypt(row.GeoLocation),
Companylogo = row.Company_logo,
CreatedOn1 = row.CreatedOn1,
UpdatedOn1 = row.UpdatedOn1,
uid = row.uid,
logopath = row.logo_path,
age = row.age,
About=row.about,
MotherTongue = mothertongue,
GroupName = groupname,

SentActive = SentActive,
SentSItpe = SentSItpe,
RecvActive = RecvActive,


```

        RecvSltype = RecvStype,
        HeartListAdded = HeartListAdded,
        ShortListAdded = ShortListAdded,
        SentProfileAdded = SentProfileAdded,
        RcvdProfileAdded = RcvdProfileAdded,
        School = (int)row.School,
        College = (int)row.college,
        University = (int)row.university,
        SchoolName = Schoolname,
        CollegeName = CollegeName,
        UniversityName = UnivName,
    });
    //break;

    }
}

}
var usercart1 = GetAllUserPrivacyDetails(userid, result);
return usercart1;
}
public List<EteGetSearchResults> getsearchgroupprofile(int userid, string groupid)
{
    string gender = "W";
    string age_from = "0";
    string age_to = "0";
    string religion = "0";
    string caste = "0";
    string area = "";
    string mtongue = "0";
    string geoloc = "";
    List<EteGetSearchResults> result = new List<EteGetSearchResults>();
    using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var opt = (from c in db.tbluser_options where c.userid == userid select c).SingleOrDefault();
        if (opt != null)
        {
            gender = opt.Gender_Search;
            age_from = opt.Age_from.ToString();
            age_to = opt.Age_to.ToString();
            religion = opt.Religion_search.ToString();
            caste = opt.Caste_search.ToString();
            area = opt.Area_search.ToString();
            mtongue = opt.MTongue_search.ToString();
            geoloc = opt.GeoLoc_search.ToString();
        }
    }
}

```

```

string dept = "";
if (gender == "W") { dept = "WOMEN PROFILES"; }
else if (gender == "M") { dept = "MEN PROFILES"; }
else if (gender == "B") { dept = "BI-SEXUAL PROFILES"; } else { dept = "OTHERS PROFILES"; }
List<GetSearchResultsResult> users = db.GetSearchResults("", "0", "0", "0", "0", "0", "", "0", "",
"0").ToList();

//users = db.GetSearchGroupResults(groupid, gender, age_from, age_to, religion, caste, area,
mtongue, geoloc).ToList();

foreach (var row in users)
{
//System.Data.Linq.Binary Image = (from d in db.tblUserImages where d.UserId ==
row.UserId select d.Images).FirstOrDefault();
//if (Image != null)
//{
// byte[] bytes = (byte[])Image.ToArray();
// string base64String = Convert.ToBase64String(bytes, 0, bytes.Length);
// string imageUrl = "data:image/png;base64," + base64String;
//}
string groupname = (from c in db.tblPermissionGroups where c.GroupId == row.GroupId
select c.GroupName).FirstOrDefault();
if (row.UserId != userid)
{
string Sexornt = "";
if (row.Sex_ornt == 1) { Sexornt = "Straight"; } else if (row.Sex_ornt == 2) { Sexornt = "Gay";
} else { Sexornt = "Bi-Sexual"; }
string mothertongue = (from c in db.Languages where c.Lan_id == row.MTongue select
c.Language1).SingleOrDefault();
result.Add(new EteGetSearchResults
{
UserId = row.UserId,
UserName = Decrypt(db.EncryptData(row.UserName)),
FirstName = Decrypt(row.FirstName),
LastName = Decrypt(row.LastName),
EmailID = Decrypt(row.EmailID),
ContactNum = Decrypt(row.ContactNum),
CreatedOn = row.CreatedOn,
Status = row.Status,
RoleID = row.RoleID,
GroupId = row.GroupId,
IsRemoved = row.IsRemoved,
Department = dept,
Region = row.Region,
UpdatedOn = row.UpdatedOn,
DealerId = row.DealerId,
Gender = row.Gender,
Landline = Decrypt(row.Landline),

```

```

MTongue = (int)row.MTongue,
DateOfBirth = (DateTime)row.DateOfBirth,
TimeOfBirth = (DateTime)row.TimeOfBirth,
PlaceOfBirth = Decrypt(row.PlaceofBirth),
adshowchk = (bool)row.adshowchk,
MatchingChk = (bool)row.Matchingchk,

userid1 = row.userid1,
Details = Decrypt(row.Details),
Occupation = Decrypt(row.Occupation),
Occupationtype = Decrypt(row.Occupation_type),
Eduqualification = Decrypt(row.Edu_qualification),
Martialstatus = row.Martial_status,
Sexornt = Sexornt,
dob = row.dob,
tob = row.tob,
pob = Decrypt(row.pob),
familynos = row.family_nos,
weburl = Decrypt(row.web_url),
adhaarid = Decrypt(row.adhaar_id),
panno = Decrypt(row.pan_no),
weeklydays = row.weekly_days,
IncomeYear = row.Income_Year,
currency = row.currency,
Religion = row.Religion,
Caste = row.Caste,
SubCaste = Decrypt(row.Sub_Caste),
Location = row.Location,
GeoLocation = Decrypt(row.GeoLocation),
Companylogo = row.Company_logo,
CreatedOn1 = row.CreatedOn1,
UpdatedOn1 = row.UpdatedOn1,
uid = row.uid,
logopath = row.logo_path,
age = row.age,
About = row.about,
MotherTongue = mothertongue,
GroupName = groupname,

});
//break;

}
}

}
var usercart1 = GetAllUserPrivacyDetails(userid, result);
return usercart1;

```

```

}
public List<EteGetSearchResults> gettotalgroupprofile(int userid, string groupid)
{
    List<EteGetSearchResults> result = new List<EteGetSearchResults>();
    using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();

        List<GetTotalResultsResult> users = db.GetTotalResults(groupid).ToList();

        foreach (var row in users)
        {
            //System.Data.Linq.Binary Image = (from d in db.tblUserImages where d.UserId ==
row.UserId select d.Images).FirstOrDefault();
            //if (Image != null)
            //{
            //    byte[] bytes = (byte[])Image.ToArray();
            //    string base64String = Convert.ToBase64String(bytes, 0, bytes.Length);
            //    string imageUrl = "data:image/png;base64," + base64String;
            //}
            if (row.UserId != userid)
            {
                string Sexornt = "", gender = "";
                if (row.Sex_ornt == 1) { Sexornt = "Straight"; } else if (row.Sex_ornt == 2) { Sexornt = "Gay";
} else { Sexornt = "Bi-Sexual"; }
                string dept = "";
                gender = row.Gender.ToString();
                if (gender == "W") { dept = "WOMEN PROFILES"; }
                else if (gender == "M") { dept = "MEN PROFILES"; }
                else if (gender == "B") { dept = "BI-SEXUAL PROFILES"; } else { dept = "OTHERS PROFILES"; }
                result.Add(new EteGetSearchResults
                {
                    UserId = row.UserId,
                    UserName = Decrypt(db.EncryptData(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 = dept,
                    Region = row.Region,
                    UpdatedOn = row.UpdatedOn,
                });
            }
        }
    }
}

```

```

DealerId = row.DealerId,
Gender = row.Gender.ToString(),
Landline = row.Landline,
MTongue = row.MTongue,
userid1 = row.userid1,
Details = row.Details,
Occupation = row.Occupation,
Occupationtype = row.Occupation_type,
Eduqualification = row.Edu_qualification,
Maritalstatus = row.Marital_status,
Sexornt = Sexornt,
dob = row.dob,
tob = row.tob,
pob = row.pob,
familynos = row.family_nos,
weblink = row.web_url,
adhaarid = row.adhaar_id,
panno = row.pan_no,
weeklydays = row.weekly_days,
IncomeYear = row.Income_Year,
currency = row.currency,
Religion = row.Religion,
Caste = row.Caste,
SubCaste = row.Sub_Caste,
Location = row.Location,
GeoLocation = row.GeoLocation,
Companylogo = row.Company_logo,
CreatedOn1 = row.CreatedOn1,
UpdatedOn1 = row.UpdatedOn1,
uid = row.uid,
logopath = row.logo_path,
age = row.age,

});
//break;
}

}

}
var usercart1 = GetAllUserPrivacyDetails(userid, result);
return usercart1;

}
public List<getcount> gettotalgroupcount(int userid, string groupid)
{
List<getcount> result = new List<getcount>();

```

```

using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
{
    db.Connection.Open();

    List<GetTotalGroupByResultsResult> lst =
db.GetTotalGroupByResults(groupid,userid).ToList();
    var list = (from c in lst orderby c.Descr select c).ToList();
    foreach (var row in list)
    {
        result.Add(new getcount
        {
            Desc = row.Descr,
            count =(int) row.cnt,
            SearchStr=row.searchstr,

        });
    }
}
return result;

}
public List<EtetblUser> getsearchuser(string UserName)
{
    List<EtetblUser> result = new List<EtetblUser>();
    using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        UserName = db.EncryptData(Encrypt(UserName));
        //          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.tblUserImages where d.UserId == row.UserId
select d.Images).FirstOrDefault();
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 = Decrypt(db.EncryptData(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,
    MTongue = (int)row.MTongue,
    DateOfBirth = (DateTime)row.DateOfBirth,
    TimeOfBirth = (DateTime)row.TimeOfBirth,
    PlaceOfBirth = row.PlaceofBirth,
    adshowchk = (bool)row.adshowchk,
    MatchingChk = (bool)row.Matchingchk,

});
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()).GetPagelD(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);
            if (GettblPermissionGroup != null)
            {
                GroupName = GettblPermissionGroup.GroupName;
            }

            _user.Add(new EtetblUser
            {
                UserId = row.UserId,
                UserName = Decrypt(db.EncryptData(row.UserName)),
                Password = row.Password,
                FirstName = Decrypt(row.FirstName),
                LastName = Decrypt(row.LastName),
                EmailID = Decrypt(row.EmailID),
                ContactNum = Decrypt(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,
                PagelD = pgid,
                GroupName = GroupName,
            });
        }
    }
}

```



```

        Gender = row.Gender,
        MTongue = (int)row.MTongue,
        DateOfBirth = (DateTime)row.DateOfBirth,
        TimeOfBirth = (DateTime)row.TimeOfBirth,
        PlaceOfBirth = row.PlaceofBirth,
        adshowchk = (bool)row.adshowchk,
        MatchingChk = (bool)row.Matchingchk,
    });
}

return _user;
}
}
public List<EtetblUser> getusersbak(string Curl)
{
    List<EtetblUser> _user = new List<EtetblUser>();
    using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
    {

        db.Connection.Open();
        int pgid = (new tblPages()).GetPagelD(Curl);
        var users = (from c in db.tblUsers orderby c.UserId select c).ToList();

        foreach (var row in users)
        {

            _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,
                Landline=row.Landline,

                IsRemoved = row.IsRemoved,
                Department = row.Department,
                Region = row.Region,
                UpdatedOn = row.UpdatedOn,
                DealerId = row.DealerId,
                DealerName = "",
            });
        }
    }
}

```

```

        Pageld = pgid,
        GroupName = "",
        Gender = row.Gender,
        MTongue = (int)row.MTongue,
        DateOfBirth = (DateTime)row.DateOfBirth,
        TimeOfBirth = (DateTime)row.TimeOfBirth,
        PlaceOfBirth = row.PlaceofBirth,
        adshowchk = (bool)row.adshowchk,
        MatchingChk = (bool)row.Matchingchk,
    });
}

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()).GetPageld(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);
                if (GettblPermissionGroup != null)
                {
                    GroupName = GettblPermissionGroup.GroupName;
                }
                EteDealer getdealer = (new Dealers()).getdealer(row.DealerId);
                if (getdealer != null)
                {
                    DealerName = getdealer.CompanyName;
                }
                _user.Add(new EtetblUser
                {
                    UserId = row.UserId,
                    UserName = Decrypt(db.EncryptData(row.UserName)),
                    Password = Decrypt(db.EncryptData(row.Password)),
                    FirstName = Decrypt(row.FirstName),
                    LastName = Decrypt(row.LastName),
                });
            }
        }
    }
}

```

```

        EmailID = Decrypt(row.EmailID),
        ContactNum = Decrypt(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,
        MTongue = (int)row.MTongue,
        DateOfBirth = (DateTime)row.DateOfBirth,
        TimeOfBirth = (DateTime)row.TimeOfBirth,
        PlaceOfBirth = row.PlaceOfBirth,
        adshowchk = (bool)row.adshowchk,
        MatchingChk = (bool)row.Matchingchk,
    });
}
}

return _user;
}
}
public List<Etetbluser_address> getusersaddressbyuserid(int UserId, string Curl)
{
    List<Etetbluser_address> _user = new List<Etetbluser_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 tblPagesess()).GetPageld(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 = Decrypt( EMSentUserName),
            });
        }
    }
}

```

```

        EMRcvdUserName = Decrypt( EMRcvdUserName),
        MSAttachment=data.EMAttachment,
        IsActive = (bool)data.IsActive,

    });
}
}

return _user;
}
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,
                EMAttachment = data.MSAttachment, //attachment
                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;
}
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()).GetPagelD(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,
ImgPath=users.ImgPath,
});
}
}
// 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()).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);
        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
        {
            var _userprofile = db.tbluser_profiles.Where(e => e.userid == data.userid).SingleOrDefault();

            if (_userprofile != null)
            {
                result = UpdateProfile(data);
            }
            else
            {
                tbluser_profile _cs = new tbluser_profile
                {
                    userid = data.userid,
                    Details = Encrypt(data.Details),
                    Occupation = Encrypt(data.Occupation),

                    Occupation_type = Encrypt(data.Occupationtype),
                    Edu_qualification = Encrypt(data.Eduqualification),
                    Martial_status = data.Martialstatus,
                    Sex_ornt = data.Sexornt,
                    dob = data.dob,
                    tob = data.tob,
                    pob = Encrypt(data.pob),
                    family_nos = data.familynos,
                    web_url = Encrypt(data.weburl),
                    adhaar_id = Encrypt(data.adhaarid),
                    pan_no = Encrypt(data.panno),
                    weekly_days = data.weeklydays,
                }
            }
        }
        catch { }
    }
}

```

```

Company_logo = data.Companylogo,
CreatedOn = data.CreatedOn,
UpdatedOn = data.UpdatedOn,
uid = data.uid,
logo_path = data.logopath,
Religion = data.Religion,
Caste = data.Caste,
Sub_Caste = Encrypt(data.Sub_Caste),

Income_Year = data.Income_year,
currency = data.currency,
Location = data.Location,
GeoLocation = Encrypt(data.GeoLocation),
about=Encrypt(data.About),
age=data.Age,
School=data.School,
college=data.College,
university=data.University,
school_year = data.School_year,
college_year = data.College_year,
Univ_year = data.Univ_year,
};
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;
Etebluser_profile odata = new Etebluser_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,
                    Details = data.Details,
                    Occupation = data.Occupation,
                    Occupationtype = data.Occupationtype,
                    Eduqualification = data.Eduqualification,
                    Martialstatus = data.Martialstatus,
                    Sexornt = data.Sexornt,
                    dob = data.dob,
                    tob = data.tob,
                    pob = data.pob,
                    familynos = data.familynos,
                    weblink = data.weblink,
                    adhaarid = data.adhaarid,
                    panno = data.panno,
                    weeklydays = data.weeklydays,
                    Companylogo = data.Companylogo,
                    CreatedOn = data.CreatedOn,
                }
            }
        }
    }
}

```

```

UpdatedOn = data.UpdatedOn,
uid = data.uid,
logopath = data.logopath,
Religion=data.Religion,
Caste=data.Caste,
Sub_Caste=data.Sub_Caste,
About=data.About,
School = data.School,
College = data.College,
University = data.University,
School_year = data.School_year,
College_year = data.College_year,
Univ_year = data.Univ_year,

};
if (odata.CreatedOn == null) { odata.CreatedOn = DateTime.Today; }
if (data.CreatedOn == null) { data.CreatedOn = odata.CreatedOn; }

_user.userid = data.userid;
_user.Details = Encrypt(data.Details);
_user.Occupation = Encrypt(data.Occupation);
_user.Occupation_type = Encrypt(data.Occupationtype);
_user.Edu_qualification = Encrypt(data.Eduqualification);
_user.Martial_status = data.Martialstatus;
_user.Sex_ornt = data.Sexornt;
_user.dob = data.dob;
_user.tob = data.tob;
_user.pob = Encrypt(data.pob);
_user.family_nos = data.familynos;
_user.web_url = Encrypt(data.weburl);
_user.adhaar_id = Encrypt(data.adhaarid);
_user.pan_no = Encrypt(data.panno);
_user.weekly_days = data.weeklydays;
_user.Company_logo = data.Companylogo;
_user.CreatedOn = data.CreatedOn;
_user.UpdatedOn = data.UpdatedOn;
_user.uid = data.uid;
_user.logo_path = data.logopath;
_user.Income_Year = data.Income_year;
_user.currency = data.currency;
_user.Location = data.Location;
_user.GeoLocation = Encrypt(data.GeoLocation);
_user.Religion = data.Religion;
_user.Caste = data.Caste;
_user.Sub_Caste = Encrypt(data.Sub_Caste);
_user.about = Encrypt(data.About);
_user.university = data.University;
_user.college = data.College;

```

```

        _user.School = data.School;
        _user.Univ_year = data.Univ_year;
        _user.college_year = data.College_year;
        _user.school_year = data.School_year;
        db.SubmitChanges();
        db.Transaction.Commit();

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

//    throw exc;
//}
}
//try
{
    mess = BalAuditTrail.GetAuditXml("tblUser_profile", 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 = 1,
    };
    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)
    {
        //string Sexornt = "";
        //if (data.Sex_ornt == 1) { Sexornt = "Straight"; } else if (data.Sex_ornt == 2) { Sexornt = "Gay";
    } else { Sexornt = "Bi-Sexual"; }

        string sxor = (from d in db.SexualOrs where d.sx_id == data.Sex_ornt select
d.sx_desc).SingleOrDefault();
        string matst = (from e in db.MartialStatus where e.ms_id == data.Martial_status select
e.ms_desc).SingleOrDefault();
        string relg = (from f in db.Religions where f.ml_id == data.Sex_ornt select
f.ml_desc).SingleOrDefault();
        string castee = (from g in db.Castes where g.mc_id == data.Sex_ornt select
g.mc_desc).SingleOrDefault();
        string Schoolname = (from c in db.SchoolsLists where c.S_id == data.School select
c.SchoolName).SingleOrDefault();
        string UnivName = (from c in db.UniversityLists where c.U_id == data.university select
c.UniversityName).SingleOrDefault();
        string CollegeName = (from c in db.CollegesLists where c.C_id == data.college select
c.CollegeName).SingleOrDefault();

        result = new Etetbluser_profile
        {
            userid = data.userid,
            Details =Decrypt( data.Details),
            Occupation = Decrypt(data.Occupation),
            Occupationtype = Decrypt(data.Occupation_type),
            Eduqualification = Decrypt(data.Edu_qualification),
            Martialstatus = data.Martial_status,
            Sexornt = data.Sex_ornt,
            dob = data.dob,
            tob = data.tob,
            pob = Decrypt(data.pob),
            familynos = data.family_nos,
            weburl = Decrypt(data.web_url),
            adhaarid = Decrypt(data.adhaar_id),
            panno = Decrypt(data.pan_no),
            weeklydays = data.weekly_days,
            Companylogo = data.Company_logo,
            CreatedOn = data.CreatedOn,
            UpdatedOn = data.UpdatedOn,
            uid = data.uid,
            logopath = data.logo_path,
            Income_year = data.Income_Year,
            Location = data.Location,

```

```

        currency = data.currency,
        GeoLocation = Decrypt( data.GeoLocation),
        Religion = data.Religion,
        Caste = data.Caste,
        Sub_Caste = Decrypt(data.Sub_Caste),
        StrSexornt = sxor,
        StrMartialstatus = matst,
        StrReligion=relg,
        StrCaste=castee,
        About= Decrypt( data.about),
        School = (int)data.School,
        College = (int)data.college,
        University = (int)data.university,
        SchoolName = Schoolname,
        CollegeName = CollegeName,
        UniversityName = UnivName,
        School_year = (int)data.school_year,
        College_year = (int)data.college_year,
        Univ_year = (int)data.Univ_year,

};

}
}

return result;
}
public List<Etetbluser_profile> getuserprofiles()
{
    List<Etetbluser_profile> output = new List<Etetbluser_profile>();
    Etetbluser_profile result = new Etetbluser_profile();
    using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
    {
        db.Connection.Open();
        var data1 = (from c in db.tbluser_profiles select c).ToList();
        if (data1 != null)
        {
            //string Sexornt = "";
            //if (data.Sex_ornt == 1) { Sexornt = "Straight"; } else if (data.Sex_ornt == 2) { Sexornt = "Gay";
} else { Sexornt = "Bi-Sexual"; }
            foreach (var data in data1)
            {
                string sxor = (from d in db.SexualOrs where d.sx_id == data.Sex_ornt select
d.sx_desc).SingleOrDefault();
                string matst = (from e in db.MartialStatus where e.ms_id == data.Martial_status select
e.ms_desc).SingleOrDefault();
                string relg = (from f in db.Religions where f.ml_id == data.Sex_ornt select
f.ml_desc).SingleOrDefault();

```

```

        string castee = (from g in db.Castes where g.mc_id == data.Sex_ornt select
g.mc_desc).SingleOrDefault();
        string Schoolname = (from c in db.SchoolsLists where c.S_id == data.School select
c.SchoolName).SingleOrDefault();
        string UnivName = (from c in db.UniversityLists where c.U_id == data.university select
c.UniversityName).SingleOrDefault();
        string CollegeName = (from c in db.CollegesLists where c.C_id == data.college select
c.CollegeName).SingleOrDefault();

```

```

output.Add(new Etetbluser_profile
{

```

```

    userid = data.userid,
    Details = Decrypt(data.Details),
    Occupation = Decrypt(data.Occupation),
    Occupationtype = Decrypt(data.Occupation_type),
    Eduqualification = Decrypt(data.Edu_qualification),
    Martialstatus = data.Martial_status,
    Sexornt = data.Sex_ornt,
    dob = data.dob,
    tob = data.tob,
    pob = Decrypt(data.pob),
    familynos = data.family_nos,
    webservurl = Decrypt(data.web_url),
    adhaarid = Decrypt(data.adhaar_id),
    panno = Decrypt(data.pan_no),
    weeklydays = data.weekly_days,
    Companylogo = data.Company_logo,
    CreatedOn = data.CreatedOn,
    UpdatedOn = data.UpdatedOn,
    uid = data.uid,
    logopath = data.logo_path,
    Income_year = data.Income_Year,
    Location = data.Location,
    currency = data.currency,
    GeoLocation = Decrypt(data.GeoLocation),
    Religion = data.Religion,
    Caste = data.Caste,
    Sub_Caste = Decrypt(data.Sub_Caste),
    StrSexornt = sxor,
    StrMartialstatus = matst,
    StrReligion = relg,
    StrCaste = castee,
    About = Decrypt(data.about),
    School = (int)data.School,
    College = (int)data.college,
    University = (int)data.university,
    SchoolName = Schoolname,

```



```

        CollegeName = CollegeName,
        UniversityName = UnivName,
        School_year = (int)data.school_year,
        College_year = (int)data.college_year,
        Univ_year = (int)data.Univ_year,

    });
}
}
}

return output;
}
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.Companylogo;

        }
        else
        {
            tbluser_profile_cs = new tbluser_profile
            {
                userid = data.userid,
                CreatedOn = DateTime.Now,
                Company_logo = data.Companylogo,
                UpdatedOn = DateTime.Now,

            };
            db.tbluser_profiles.InsertOnSubmit(_cs);
        }
        db.SubmitChanges();
        db.Transaction.Commit();

        result = true;
    }

    return result;
}

```

```

}
public string InsertImages(EtetblUserImages data)
{
    bool result = false;
    string path = "";
    //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++;
        path = "/images/Profiles/img-"+data.UserId.ToString()+"-"+srno.ToString()+".jpg";
        tblUserImage _cs = new tblUserImage
        {
            UserId = data.UserId,
            srno = srno,
            CreatedOn = DateTime.Now,
            Images = data.Images,
            ImgPath=path,
        };
        db.tblUserImages.InsertOnSubmit(_cs);

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

        result = true;
    }

    return path;
}
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 InsertSendProfile(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.UserSendProfiles where c.UserId == data.UserId && c.PSId ==
data.PSId select c).SingleOrDefault();
            if (user == null)
            {
                UserSendProfile _cs = new UserSendProfile
                {
                    UserId = data.UserId,
                    PSId = data.PSId,
                    UId = data.UId,
                    CreatedOn = DateTime.Now,
                    Active = data.Active,
                    SLType = stype,
                }
            }
        }
    }
}

```

```

        };
        db.UserSendProfiles.InsertOnSubmit(_cs);

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

return result;
}
public bool UpdateSendProfile(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.UserSendProfiles where c.UserId == data.UserId && c.PSId ==
data.PSId select c).SingleOrDefault();
                if (user != null)
                {
                    user.UserId = data.UserId;
                    user.PSId = data.PSId;
                    user.UId = data.UId;
                    user.CreatedOn = DateTime.Now;
                    user.Active = data.Active;
                    user.SLType = stype;
                    if (stype) { user.AcceptdOn = DateTime.Now; }

                    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.UserId == data.UserId && c.PSId==data.PSId 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 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 string Encrypt(string input)
{
    if (input == null) { return null; }
    byte[] inputArray = UTF8Encoding.UTF8.GetBytes(input);
    TripleDESCryptoServiceProvider tripleDES = new TripleDESCryptoServiceProvider();
    tripleDES.Key = UTF8Encoding.UTF8.GetBytes(key);
    tripleDES.Mode = CipherMode.ECB;
    tripleDES.Padding = PaddingMode.PKCS7;
    ICryptoTransform cTransform = tripleDES.CreateEncryptor();
    byte[] resultArray = cTransform.TransformFinalBlock(inputArray, 0, inputArray.Length);
    tripleDES.Clear();
    return Convert.ToBase64String(resultArray, 0, resultArray.Length);
}

public string Decrypt(string input)
{
    try
    {
        if (input != "" && input != null)
        {
            byte[] inputArray = Convert.FromBase64String(input);
            TripleDESCryptoServiceProvider tripleDES = new TripleDESCryptoServiceProvider();
            tripleDES.Key = UTF8Encoding.UTF8.GetBytes(key);
            tripleDES.Mode = CipherMode.ECB;
            tripleDES.Padding = PaddingMode.PKCS7;
            ICryptoTransform cTransform = tripleDES.CreateDecryptor();
            byte[] resultArray = cTransform.TransformFinalBlock(inputArray, 0, inputArray.Length);
            tripleDES.Clear();

            return UTF8Encoding.UTF8.GetString(resultArray);
        }
    }
    else { return ""; }
}

```

```

    }
    catch
    {
        return input;
    }
}
public string Encrypt2(string inText)
{
    byte[] bytesBuff = Encoding.Unicode.GetBytes(inText);
    using (Aes aes = Aes.Create())
    {
        Rfc2898DeriveBytes crypto = new Rfc2898DeriveBytes(key,salt) ;
        aes.Key = crypto.GetBytes(32);
        aes.IV = crypto.GetBytes(16);
        using (MemoryStream mStream = new MemoryStream())
        {
            using (CryptoStream cStream = new CryptoStream(mStream, aes.CreateEncryptor(),
CryptoStreamMode.Write))
            {
                cStream.Write(bytesBuff, 0, bytesBuff.Length);
                cStream.Close();
            }
            inText = Convert.ToBase64String(mStream.ToArray());
        }
    }
    return inText;
}

//Decrypting a string
public string Decrypt2(string cryptTxt)
{
    //cryptTxt = cryptTxt.Replace(" ", "+");
    byte[] bytesBuff = Convert.FromBase64String(cryptTxt);
    using (Aes aes = Aes.Create())
    {
        Rfc2898DeriveBytes crypto = new Rfc2898DeriveBytes(key, salt);
        aes.Key = crypto.GetBytes(32);
        aes.IV = crypto.GetBytes(16);
        using (MemoryStream mStream = new MemoryStream())
        {
            using (CryptoStream cStream = new CryptoStream(mStream, aes.CreateDecryptor(),
CryptoStreamMode.Write))
            {
                cStream.Write(bytesBuff, 0, bytesBuff.Length);
                cStream.Close();
            }
            cryptTxt = Encoding.Unicode.GetString(mStream.ToArray());
        }
    }
}

```

```

        return cryptTxt;
    }
}
}
}
-----
using DOWDal;
using DOWEntity;
using System;
using System.Collections.Generic;
using System.Linq;
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(EtetblPermissionGroup data)
{
    int result = 0;
    EtetblPermissionGroup _user = new EtetblPermissionGroup();
    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(Et tblPermissionGroup 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 DOWDal;
using DOWEntity;
using System;
using System.Collections.Generic;
using System.Linq;
namespace DOWBal
{
    public class UserPrivileges
    {
        public List<Etetbl_moptions> GetUserOptionDetbyUser(int UserId, int Mopt)
        {
            List<Etetbl_moptions> lst1 = new List<Etetbl_moptions>();
            using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
            {
                db.Connection.Open();

                var lst = (from c in db.tbl_moptions select c).ToList();
                foreach (var _cust in lst)
                {
                    if (_cust != null)
                    {
                        lst1.Add(new Etetbl_moptions
                        {
                            mopt_id = _cust.mopt_id,
                            mopt_desc = _cust.mopt_desc,
                        });
                    }
                }
                return lst1;
            }
        }
        public Etetbl_moptions GetOptions()
    }
}

```

```

{
    Etetbl_moptions lst = new Etetbl_moptions();
    using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
    {

        db.Connection.Open();

        var _cust = (from c in db.tbl_moptions select c).SingleOrDefault();

        if (_cust != null)
        {
            lst = new Etetbl_moptions
            {
                mopt_id = _cust.mopt_id,
                mopt_desc = _cust.mopt_desc,

            };
        }

        return lst;
    }
}

public Etetbluser_options GetUserOptionsByUser(int userid)
{
    Etetbluser_options lst = new Etetbluser_options();
    using (var db = new DOWPortalDataContext(Configuration.ConnectionString))
    {

        db.Connection.Open();

        var _cust = (from c in db.tbluser_options where c.userid == userid select c).SingleOrDefault();

        if (_cust != null)
        {
            lst = new Etetbluser_options
            {
                userid = _cust.userid,
                Gender_Search = _cust.Gender_Search,
                Age_from = _cust.Age_from,
                Age_to = _cust.Age_to,
                Religion_search = _cust.Religion_search,
                Caste_search = _cust.Caste_search,
                Area_search = _cust.Area_search,
                MTongue_search = _cust.MTongue_search,
                GeoLoc_search = _cust.GeoLoc_search,
                CreatedOn = _cust.CreatedOn,
            };
        }
    }
}

```

```

        };
    }

    return lst;
}
}
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.PrivilegeId == Id select c).SingleOrDefault();

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

    }

    return exists;
}

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
            {
                PrivilegeId = data.PrivilegeId,
                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.PrivilegeId == data.PrivilegeId).SingleOrDefault();

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

            _user.PrivilegeId = data.PrivilegeId;
            _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.PrivilegeId == name select c).ToList();
        foreach (var data in users)
        {
            result = new EteUserPrivilege
            {
                PrivilegeId = data.PrivilegeId,
                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
            {
                PrivilegeId = data.PrivilegeId,
                UserId = data.UserId,

```



```

        //IsActive = data.IsActive,
        CreatedOn = data.CreatedOn
    });
}

return _user;
}
}

```

```

}
}

```

```

using DOWDal;
using DOWEntity;
using System;
using System.Collections.Generic;
using System.Linq;
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? GetNextRoleId()
        //{

```

```

// 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.Configuration;

```

```
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;
            }
        }
    }
}
}
```

This concludes Business Layer coding

Chapter 6

Develop Master Windows to inherit during development

Each window will have two files, one .aspx and its corresponding .cs file

***** BListingMasterPage .aspx*****

```
<%@ 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>DateOrWed Dashboard</title>
```

```
  <link rel="icon" href="img/favicon.png" type="image/gif">
```

```
  <!-- Font Awesome -->
```

```
  <link rel="icon" type="image/gif" href="images/DateOrWed.png">
```

```
    <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">
```

```
  <!-- 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">
```

```
<style>
```

```
</style>
```

```
  <!-- BOOTSTRAP -->
```

```
<!-- 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>--%>
```

```
  <link rel="stylesheet" href="dowcss/css/bootstrap.min.css"/>
```

```

<link rel="stylesheet" href="dowcss/css/font-awesome.min.css"/>
<link rel="stylesheet" href="font/fontawesome/css/all.css"/>
<%-- <link rel="stylesheet" href="dowcss/css/sstyle.css"/>--%>
<link rel="stylesheet" href="dowcss/css/animate.css"/>
<link rel="stylesheet" href="dowcss/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="dowcss/css/slick.css"/>
<link rel="stylesheet" type="text/css" href="dowcss/css/slick-theme.css"/>
<link rel="stylesheet" href="css/sstyle.css"/>
<link rel="stylesheet" href="css/maps.css"/>

<!-- WEB FONTS -->
<link href='https://fonts.googleapis.com/css?family=Open+Sans:300,400,600,700' rel='stylesheet'
type='text/css'>
<link href="https://fonts.googleapis.com/css?family=Raleway:500,800" rel="stylesheet"
type="text/css" media="all" />
<script type="text/javascript" src="dowcss/js/jquery.min.js"></script>
<script type="text/javascript" src="dowcss/js/bootstrap.min.js"></script>
<script type="text/javascript" src="plugins/owl-carousel/owl.carousel.min.js"></script>
<script type="text/javascript" src="plugins/FlexSlider/jquery.flexslider-min.js"></script>
<script type="text/javascript" src="dowcss/js/jquery.magnific-popup.min.js"></script>
<script type="text/javascript" src="dowcss/js/magnific-popup.min.js"></script>
<script type="text/javascript" src="dowcss/js/slick.js"></script>
<script type="text/javascript" src="dowcss/js/slick-slider.js"></script>
<script type="text/javascript" src="dowcss/js/custom.js"></script>
<script type="text/javascript" src="js/maps.js"></script>
</head>

<body>
<form id="form1" runat="server" >
<asp:ScriptManager ID="ScriptManager1" runat="server">
</asp:ScriptManager>
<!-- Start your project here-->
<asp:ContentPlaceHolder ID="ContentPlaceHolder1" runat="server">

<!--Navbar -->

<nav class="mb-1 navbar fixed-top navbar-expand-lg navbar-light bg-light" hidden="hidden">
<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>

        </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 = 'ViewSearchResult.aspx?pageid=198' + '&SrchStrng=' +
      encodeURIComponent(SearchString) + '&loc=' + encodeURIComponent(loc);
      //$(location).attr('href', url);
      window.location.href = url;
      event.preventDefault();
    }
  </script>

  </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>
</asp:ContentPlaceHolder>
</asp:ContentPlaceHolder>

<asp:ContentPlaceHolder ID="CPH_MainMaster_Header" runat="server">

<section class="buynsell pt-5 pb-3" >
<div class="container-fluid">
<div class="row">
<div class="col-md-2" id="sidebar" >
<div class="m-list" id="mlist" runat="server"><h5><i class="fas fa-list"></i> DashBoard</h5></div>
<div class="list-group panel">
    <a href="Dashboard.aspx" class="list-group-item collapsed" data-parent="#sidebar"> <span
class="hidden-sm-down"><i class="fas fa-circle"></i> Dashboard</span></a>
    <a href="Home.aspx" class="list-group-item collapsed" data-parent="#sidebar"> <span
class="hidden-sm-down"><i class="fas fa-circle"></i> Home</span></a>

    <a href="#menu3" class="list-group-item collapsed" data-toggle="collapse" data-
parent="#sidebar" aria-expanded="false"> <span class="hidden-sm-down"><i class="fas fa-circle"></i>
Profile <i class="fas fa-chevron-right" style="font-size: 12px;"></i></span></a>

    <div class="collapse" id="menu3">

        <a href="LoginInfo.aspx" class="list-group-item" data-parent="#menu3"><i class="fas fa-
circle"></i> Login Info </a>
        <a href="ProfileInfo.aspx" class="list-group-item" data-parent="#menu3"><i class="fas fa-
circle"></i> Profile Details </a>
        <a href="ProfileSettings.aspx" class="list-group-item" data-parent="#menu3"><i class="fas
fa-circle"></i> Settings </a>

```

```
        <a href="ProfileAddress.aspx" class="list-group-item" data-parent="#menu3"><i class="fas fa-circle"></i> Addresses </a>
        <a href="ProfileImages.aspx" class="list-group-item" data-parent="#menu3"><i class="fas fa-circle"></i> Images </a>
        <a href="UserChoices.aspx" class="list-group-item" data-parent="#menu3"><i class="fas fa-circle"></i> Your Choices </a>
        <a href="ManageCommunications.aspx" class="list-group-item" data-parent="#menu3"><i class="fas fa-circle"></i> Communications </a>
    </div>
```

```
<%--        <a href="#" class="list-group-item collapsed" data-parent="#sidebar"> <span class="hidden-sm-down"><i class="fas fa-circle"></i> TV & Appliances</span></a>
        <a href="#" class="list-group-item collapsed" data-parent="#sidebar"> <span class="hidden-sm-down"><i class="fas fa-circle"></i> Electronics</span></a>
        <a href="#" class="list-group-item collapsed" data-parent="#sidebar"> <span class="hidden-sm-down"><i class="fas fa-circle"></i> Furniture & Furnishing</span></a>
        <a href="#" class="list-group-item collapsed" data-parent="#sidebar"> <span class="hidden-sm-down"><i class="fas fa-circle"></i> Home & Living</span></a>
        <a href="#" class="list-group-item collapsed" data-parent="#sidebar"> <span class="hidden-sm-down"><i class="fas fa-circle"></i> Metal Appliances</span></a>
        <a href="#" class="list-group-item collapsed" data-parent="#sidebar"> <span class="hidden-sm-down"><i class="fas fa-circle"></i> Wedding Material</span></a>--%>
```

```
    </div>
```

```
</div>
```

```
<!-- Grid row -->
```

```
<%--    <div class="row mt-4" style="display:none" id="TopList" runat="server">
```

```
        <!-- Grid column -->
```

```
        <div class="col-lg-3 col-md-3 col-xs-10 col-xl-3 mb-xl-0 mb-4 ml-0">
```

```
            <!-- Card -->
```

```
            <div class="card card-cascade cascading-admin-card">
```

```
                <!-- Card Data -->
```

```
                <div class="admin-up">
```

```
                    <i class="far fa-heart primary-color mr-3 z-depth-2"></i>
```

```
                    <div class="data">
```

```
                        <p class="text-uppercase">Dating Connections</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">Dating Statistics (25%)</p>
  </div>

</div>
<!-- Card -->

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class=" col-lg-4 col-md-4 col-xs-12 col-xl-3 mb-xl-0 mb-4 ml-0">

  <!-- Card -->
  <div class="card card-cascade cascading-admin-card">

    <!-- Card Data -->
    <div class="admin-up">
      <i class="fas fa-church warning-color mr-3 z-depth-2"></i>
      <div class="data">
        <p class="text-uppercase"> Wedding Requests</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">Wedding Prospects (25%)</p>
    </div>

  </div>
<!-- Card -->

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class=" col-lg-4 col-md-4 col-xs-12 col-xl-3 mb-xl-0 mb-4 ml-0">

  <!-- Card -->
  <div class="card card-cascade cascading-admin-card">

    <!-- Card Data -->
    <div class="admin-up">

```

```

        <i class="fas fa-handshake light-blue lighten-1 mr-3 z-depth-2"></i>
        <div class="data">
          <p class="text-uppercase"> Friends Connections</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">Connections with Friends</p>
      </div>

    </div>
    <!-- Card -->

  </div>
  <!-- Grid column -->

<!-- Grid column -->
<div class=" col-lg-4 col-md-4 col-xs-12 col-xl-3 mb-xl-0 mb-4 ml-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"> Requests on your Account</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 -->

    <div class="col-md-9" hidden="hidden">
        <!--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="mt-md-6 "> <!--pt-md-2 mb-2 pb-4 ml-5 hidden="hidden"--%>

  </section>
</asp:ContentPlaceholder>

<asp:ContentPlaceholder ID="ContentMainPage" 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> info@yoursite.com</p>
<p>
  <i class="fas fa-phone mr-3"></i> +91 - 9876543210</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">© 2022 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>
<asp:ContentPlaceHolder ID="ContentFooter" 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.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.Configuration;
using DOWBal;
using DOWEntity;
using System.Data;
using System.Web.Security;
using System.Net;
using System.Text;
using System.Web.Script.Serialization;
```

```

public partial class BListingMasterPage : System.Web.UI.MasterPage
{
    DDLogTracer objlog = new DDLogTracer();
    bool IsMobile = false;
    protected void Page_Load(object sender, EventArgs e)
    {
        if (Request.Browser.IsMobileDevice)
        { IsMobile = true; }
        else
        { IsMobile = false; }
        string IPAddress = GetIPAddress();
        Session["ipaddress"] = IPAddress;
        StringBuilder sb = new StringBuilder();
        Session["Page"] = "1";

        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr == null)
        {
            bool ret = new Common().guest_login();
            if (ret == false)
            {
                Response.Redirect("~/Login.aspx");
            }
        }

    }
    protected void guest_login(string user)
    {

        List<Location> locations = new List<Location>();
        string City="";
        string Country="";
        string ipAddress = GetIPAddress();
        try
        {
            string APIKey = "744627112f51a720c93d8e11741a02c6f50ad135ea28e733d10f52dc0596b193";
            string url = string.Format("http://api.ipinfodb.com/v3/ip-city/?key={0}&ip={1}&format=json",
APIKey, ipAddress);
            using (WebClient client = new WebClient())
            {
                string json = client.DownloadString(url);
                Location location = new JavaScriptSerializer().Deserialize<Location>(json);
                locations.Add(location);
                City = locations[0].CityName;
                Country = locations[0].CountryName;
            }
        }
    }
}

```

```

    }
    catch { }
    BALCreateUser objlogin = new BALCreateUser();
    DataTable status_login = objlogin.LoginGuest(Common.Encrypt(user), user, ipAddress, "", 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 = new
tblUsers().getusername(Convert.ToInt16(objLoggedInUser.UserId));
        //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"]);
        objLoggedInUser.FirstName = status_login.Rows[0]["FirstName"].ToString();
        objLoggedInUser.LastName = status_login.Rows[0]["LastName"].ToString();
        Session["LoggedUser_Specs"] = objLoggedInUser;

        FormsAuthentication.SetAuthCookie(objLoggedInUser.UserName, false);
        HttpCookie myCookie = new HttpCookie("DOWCookie");
        DateTime now = DateTime.Now;
        myCookie.Value = objLoggedInUser.UserName;
        myCookie.Expires = now.AddYears(1);

        Response.Cookies.Add(myCookie);

    }
}
public string GetIPAddress()
{
    String address = "";
    try
    {
        WebRequest request = WebRequest.Create("http://checkip.dyndns.org/");
        using (WebResponse response = request.GetResponse())
        using (StreamReader stream = new StreamReader(response.GetResponseStream()))
        {
            address = stream.ReadToEnd();
        }
    }
}

```

```

        int first = address.IndexOf("Address: ") + 9;
        int last = address.LastIndexOf("</body>");
        address = address.Substring(first, last - first);
    }
    catch { }
    return address;

}

}

***** DashboardMasterPage.master *****
<%@ Master Language="C#" AutoEventWireup="true" CodeFile="DashboardMasterPage.master.cs"
Inherits="DashboardMasterPage" %>
<!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 | Dashboard</title>
    <!-- Favicon -->
    <link rel="icon" href="images/DateOrWed.png" type="image/gif">
    <!-- 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>
    <link rel="stylesheet" type="text/css" href="dowcss/css/slick.css"/>
    <link rel="stylesheet" type="text/css" href="dowcss/css/slick-theme.css"/>
<%--    <script type="text/javascript" src="dowcss/js/custom.js"></script>--%>
<%--    <script type="text/javascript" src="js/maps.js"></script>--%>
    <!-- BOOTSTRAP -->
    <link rel="stylesheet" href="dowcss/css/bootstrap.min.css"/>
    <link rel="stylesheet" href="dowcss/css/font-awesome.min.css"/>
    <link rel="stylesheet" href="font/fontawesome/css/all.css"/>
    <link rel="stylesheet" href="dowcss/css/sstyle.css"/>
    <link rel="stylesheet" href="dowcss/css/animate.css"/>
    <link rel="stylesheet" href="dowcss/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="dowcss/css/slick.css"/>
<link rel="stylesheet" type="text/css" href="dowcss/css/slick-theme.css"/>
<%-- <link rel="stylesheet" href="css/sstyle.css"/>--%>
<link rel="stylesheet" href="css/maps.css"/>

<!-- WEB FONTS -->
<link href='https://fonts.googleapis.com/css?family=Open+Sans:300,400,600,700' rel='stylesheet'
type='text/css'>
<link href="https://fonts.googleapis.com/css?family=Raleway:500,800" rel="stylesheet"
type="text/css" media="all" />
<script type="text/javascript" src="dowcss/js/jquery.min.js"></script>
<script type="text/javascript" src="dowcss/js/bootstrap.min.js"></script>
<script type="text/javascript" src="plugins/owl-carousel/owl.carousel.min.js"></script>
<script type="text/javascript" src="plugins/FlexSlider/jquery.flexslider-min.js"></script>
<script type="text/javascript" src="dowcss/js/jquery.magnific-popup.min.js"></script>
<script type="text/javascript" src="dowcss/js/magnific-popup.min.js"></script>
<script type="text/javascript" src="dowcss/js/slick.js"></script>
<script type="text/javascript" src="dowcss/js/slick-slider.js"></script>
<script type="text/javascript" src="dowcss/js/custom.js"></script>
<script type="text/javascript" src="js/maps.js"></script>
</script>

// Material Select Initialization
$(document).ready(function () {
    $('mdb-select').materialSelect();
});
</script>

</head>

<body class="fixed-sn white-skin">
<form id="form1" runat="server" >
    <asp:ScriptManager ID="ScriptManager1" runat="server">
    </asp:ScriptManager>

<!-- 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=""></a> <%--
dateorwed5.jpg--%>

```



```
</div>
</li>
```

```
<!-- Search Form -->
```

```
<li>
  <div class="md-form mt-0 waves-light" hidden="hidden">
    <input type="text" class="form-control py-2" placeholder="Search">
  </div>
```

```
</li>
```

```
<!-- Side navigation links -->
```

```
<li>
  <ul class="collapsible collapsible-accordion">
    <li class="active">
      <a href="Home.aspx" class="collapsible-header waves-effect"><i class="w-fa fas fa-home"></i>Home</a>
    </li>
    <li>
      <a href="DashBoard.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-user-cog"></i>Profile<i class="fas fa-angle-down rotate-icon"></i>
      </a>
      <div class="collapsible-body">
        <ul>
          <li>
            <a href="LoginInfo.aspx" class="waves-effect">Login Info </a>
          <li>
            <a href="ProfileInfo.aspx" class="waves-effect">Profile Details </a>
          </li>
          <li>
            <a href="ProfileSettings.aspx" class="waves-effect">Settings </a>
          </li>
          <li>
            <a href="ProfileAddress.aspx" class="waves-effect">Addresses </a>
          </li>
          <li>
            <a href="ProfileImages.aspx" class="waves-effect">Images </a>
          </li>
          <li>
            <a href="ProfileMapLocation.aspx" class="waves-effect">Map Location </a>
          </li>
          <li>
            <a href="UserChoices.aspx" class="waves-effect">Your Choices </a>
          </li>
```

```

        <li>
<%--      <a href="ManageLogistics.aspx" class="waves-effect">Logistics </a>
        </li>
        <li>
          <a href="ManageOrderRet.aspx" class="waves-effect">Order returned </a>
        </li>--%>
        <li>
          <a href="DOWChat.aspx" class="waves-effect">Communications </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 Profile 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 Profile 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" href="DowChat.aspx"><i class="fas fa-comments"></i> <span
class="clearfix d-none d-sm-inline-block">Chat</span></a>
</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="#">My account</a>
    <a class="dropdown-item" href="MyHeartList.aspx">My HeartList</a>
    <a class="dropdown-item" href="MyShortList.aspx">My ShortList</a>
    <a class="dropdown-item" href="SendProfile.aspx">My Sent Profiles</a>
    <a class="dropdown-item" href="RcvdProfile.aspx">My Received Profiles</a>
    <a class="dropdown-item" href="Logout.aspx">Log Out</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;"
hidden="hidden">

  <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 -->

<!-- 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="ContentMainPage" 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">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>

// Main chart
$(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');
        $(' .product-grid-caption a').toggleClass('navy-blue-bg-a text-white').toggleClass('list-group-
border');
        $(' .product-grid-caption p').toggleClass('navy-blue-bg-a text-white').toggleClass('list-group-
border');

    });
});

</script>
</form>

</body>
</html>
***** DashboardMasterPage.master.cs *****
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DOWEntity;

```

```

using DOWBal;
using System.Text;
using System.Data;

using System.Web.Security;

public partial class DashBoardMasterPage : System.Web.UI.MasterPage
{
    string UserName;
    protected void Page_Load(object sender, EventArgs e)
    {
        Session["Pid"] = "1";
        Session["Pgid"] = "0";
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr == null)
        {
            bool ret = new Common().guest_login();
            if (ret == false)
            {
                Response.Redirect("~/Login.aspx");
            }
        }
    }
}

}

***** DOWMasterPage.master *****
<%@ Master Language="C#" AutoEventWireup="true" CodeFile="DOWMasterPage.master.cs"
Inherits="DOWMasterPage" %>
<!DOCTYPE html>
<!--[if !IE]><!--> <html lang="en"> <!--![endif]-->
<head>
    <title>DateorWed - Your Choice</title>
    <!-- META -->
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
    <meta name="description" content="Dow Template">
    <meta name="author" content="ravikiran">

    <!-- FAVICON -->
    <link rel="shortcut icon" href="images/DateOrWed.png"/>

    <!-- BOOTSTRAP -->
    <link rel="stylesheet" href="dowcss/css/bootstrap.min.css"/>
    <link rel="stylesheet" href="font/fontawesome6/pro/css/all.css"/>
    <link rel="stylesheet" href="dowcss/css/sstyle.css"/>
    <link rel="stylesheet" href="dowcss/css/animate.css"/>

```

```

<link rel="stylesheet" href="dowcss/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" href="css/sstyle.css"/>

<%-- Slick css for images product slider with '<>' as previous and next--%>
  <link rel="stylesheet" type="text/css" href="dowcss/css/slick.css"/>
  <link rel="stylesheet" type="text/css" href="dowcss/css/slick-theme.css"/>
  <link rel="stylesheet" href="css/style.css"/>
<%-- <link rel="stylesheet" href="css/maps.css"/>--%>

  <!-- WEB FONTS -->
  <link href='https://fonts.googleapis.com/css?family=Open+Sans:300,400,600,700' rel='stylesheet'
type='text/css'>
  <link href="https://fonts.googleapis.com/css?family=Raleway:500,800" rel="stylesheet"
type="text/css" media="all" />
  <script type="text/javascript" src="dowcss/js/jquery.min.js"></script>

  <script type="text/javascript" src="dowcss/js/bootstrap.min.js"></script>
  <script type="text/javascript" src="plugins/owl-carousel/owl.carousel.min.js"></script>
  <script type="text/javascript" src="plugins/FlexSlider/jquery.flexslider-min.js"></script>
  <script type="text/javascript" src="dowcss/js/jquery.magnific-popup.min.js"></script>
  <script type="text/javascript" src="dowcss/js/slick.js"></script>
  <script type="text/javascript" src="dowcss/js/slick-slider.js"></script>
  <script type="text/javascript" src="dowcss/js/custom.js"></script>
  <script type="text/javascript" src="js/maps.js"></script>
  <script type="text/javascript" src="DOWCss/js/jquery.min.js"></script>

<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>
<%--nav bar transition -----

  <%--nav bar transition ends -----%>
  <div class="wrapper">

    <!-- Header -->
    <div class="header header-static">
      <!-- Topbar -->
      <div class="topbar">

        <div class="container"> <%----%>

```



```
<div class="row ">
  <div class="col-lg-4 col-sm-4">
    <!-- Topbar Navigation -->
    <ul class="topbar-left">

      <li>
        <a href="#"><i class="fas fa-mail-bulk"></i> info@dateorwed.com </a>
      </li>
      <li>
        <a><i class="fa fa-phone"></i>+91-9291355392</a>
      </li>

    </ul><!--topbar-left-->

  </div>
</div>
```

```
<div class="d-none d-sm-block d-md-none ">
  <div class="navbar-toggle col-sm-1 pull-left" >
    <a href="Home.aspx">
   </a>
  </div>
```

```
<button class="navbar-toggle mb1 " type="button" data-toggle="collapse" data-target="#account"
  aria-controls="account" aria-expanded="false" aria-label="Toggle navigation" style="font-size:
medium">
  <span class=" fa fa-bars fa-5x" style="font-size: larger"></span>
</button>
</div>
```

```
<div id="account" class="collapse navbar-collapse col-sm-8 pull-right overflow-h"
runat="server">
  <ul class="navbar-nav ml-auto list-inline topbar-right pull-right">

    <li><a href="#">Welcome Guest</a></li>
    <%--<li hidden="hidden"><a href="MyShortList.aspx"><i class="fas fa-heart"></i>
Shortlist (0)</a></li>--%>
    <li><a href="Login.aspx">Login</a> | <a href="Register.aspx">Register</a>| <a
href="Home.aspx">Home</a></li>

  </ul>
</div>
```

```
</div>
```

```

<!-- <div class="collapse navbar-collapse" id="navbarSupportedContent-4">
--%>
    <div class="topbar row clearfix ">
    <div class=" search-head text-center">

    <div class="row">
    <div class="col-sm-12"> <!--col-lg-12 col-md-12--%>
    <div class="row d-sm-none d-md-block ">

    <div class="collapse navbar-collapse" id="navbarSupportedContent-5"> <!-- text-black-50--%>
    <div class="col-lg-2 col-md-2 " >
    <a 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 Option ---</asp:ListItem>
    <asp:ListItem Value="4"> Dating Search </asp:ListItem>
    <asp:ListItem Value="5"> Wedding Search </asp:ListItem>
    <asp:ListItem Value="6"> Friends Search </asp:ListItem>
    <asp:ListItem Value="7"> Dating and Wedding </asp:ListItem>
    </asp:DropDownList>
    </div>
    <div class="col-lg-4 col-md-5 col-sm-12 p-0">
    <asp:TextBox ID="txt_SearchString" runat="server" CssClass="form-control search-slt"
Width="560px" Height="37px" MaxLength="50"
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-primary wrn-btn mt-0
ml-0" onclick="JavaScript:GetURLSring1()" ><i class='fa fa-search'></i>&nbsp;&nbsp;&nbsp;Search</button>--
%>
    <asp:ImageButton ID="ImageButton1" runat="server" ImageUrl="images/Search.2pg.jpg"
OnClick="ImageButton1_Click" Height="50px" />
    </div>
<!-- Shopping Cart -->
    <div id="shopcart" class="shop-badge badge-icons pull-right" runat="server">
    <a href="AcceptProfile.aspx?psid=0"><i class="fas fa-heart" style="font-size: x-
large"></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">
    <ul class="list-unstyled mCustomScrollbar" data-mcs-theme="minimal-dark">
    <li>

```

```

        
        <button type="button" class="close">x</button>
        <div class="overflow-h">
            <span>ABC XYZ</span>
            <small>Date or Wed</small>
        </div>
    </li>
    <li>
        
        <button type="button" class="close">x</button>
        <div class="overflow-h">
            <span>xyz abc</span>
            <small>Date and Wed</small>
        </div>
    </li>
</ul>

</div>
</div>
<!-- End Shopping Cart -->
</div>
</div>
</div>
</div>
</div>
<script type="text/javascript">
    function FindById(id, type) {

        try {

            // $("body").css("cursor", "wait");
            var ajaxUrl = "Ajax_Call.aspx/FindAddressByID"
            $.ajax({
                type: "POST",          //GET or POST or PUT or DELETE verb
                url: ajaxUrl,          // Location of the service
                data: '{id: "' + id + '" }',          //Data sent to server
                contentType: "application/json; charset=utf-8",
                dataType: "json",      //Expected data format from server
                processData: true,     //True or False
                success: function (json) { //On Successful service call
                    ProcessAddressLine(json.d, type);
                    // $("#dvAjax").html(result);
                },
                error: function (XMLHttpRequest, textStatus, errorThrown) {
                    alert("Status: " + textStatus); alert("Error: " + errorThrown);
                } // When Service call fails
            });
        }
    }

```

```

    finally {
        // $("body").css("cursor", "default");
    }
}

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 ajaxUrl = "Ajax_Call.aspx/EncryptString"

    var url = 'SearchList.aspx?SrchStrng=' + encodeURIComponent(SearchString) + '&cat=' +
    encodeURIComponent(loc);
    //$(location).attr('href', url);
    window.location.href = url;
    event.preventDefault();
}
</script>
</div>
</div>

</div><!--container-->
</div>

</div>
<!-- End Navbar -->
</div>
<!-- End Header-->

<!-- 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="sm-12" >
            <div class="row">
                <div id="dvMsgBox" class="span10 ordersave" runat="server" style="display: block;">
                    <asp:Label ID="Label1" runat="server" Text=""></asp:Label>
                </div>
            </div>
        </div>
    </div>
</asp:ContentPlaceHolder>

```

```

        </div>
    </div>--%>
</div>
</div>
</asp:ContentPlaceHolder>
<asp:ContentPlaceHolder ID="ContentMainPage" runat="server">
    <div class="row-fluid">

        </div>
</asp:ContentPlaceHolder>
<!-- Top Logo/Messages Div/Section End -->
<asp:ContentPlaceHolder ID="ContentPlaceHolder1" runat="server">
    <div class="row-fluid">

        </div>
</asp:ContentPlaceHolder>

    <asp:ContentPlaceHolder ID="CPH_MainContent" runat="server">

        <div class="container-fluid">
            <div class="row-fluid">

                </div>

            </div>

        </asp:ContentPlaceHolder>

<asp:ContentPlaceHolder ID="ContentFooter" runat="server">

<!-- START FOOTER -->
    <footer class="footer" id="footer" runat="server">
        <div class="container">
            <div class="row">
                <section class="home-main-contant-style3 bg-white">
                    <div class="parallax-block parallaxBg">
                        <div class="container">
                            <div class="parallax-block-text">
                                <p>DateorWed.com - YourChoice <span class="color-green">We are a social networking
site.</span> <br> Thank you! </p>
                            </div>
                            <div class="pre-btn-group">
                                <a class="btn btn-product mr15" href="#">Search Now</a>
                                <a class="btn btn-product mr15" href="#">Post a Profile</a>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </section>

```

```
<!-- End Parallax -->
```

```
<!-- START NEW -->
```

```
<section class="home-main-contant-style3 bg-white" style="display:none">
```

```
  <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>
```

```
</section>
```

```
<!-- END NEW -->
```

```
<!-- Start Suvscribe -->
```

```

<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="fab 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="#"><i class="fab fa-facebook"></i></a></li>
            <li><a href="#"><i class="fab fa-twitter"></i></a></li>
            <li><a href="#"><i class="fab fa-tumblr"></i></a></li>
            <li><a href="#"><i class="fab fa-instagram"></i></a></li>
            <li><a href="#"><i class="fab fa-linkedin"></i></a></li>
          </ul>
        </div>
      </div>
    </div>
  </div>
</div>
<!--End Suvscribe -->
<!-- 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) 9291355392</p>
      <p>E-mail: <a href="mailto:admin@dateorwed.com">admin@dateorwed.com</a></p>
    </address>
  </div>
</div>

```

```
</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 an Agent 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="#">dating</a></li>
```

```
        <li><a href="#">Dating & Wedding</a></li>
```

```
        <li><a href="#">Wedding</a></li>
```

```
        <li><a href="#">Option</a></li>
```

```
        <li><a href="#">Dating or Wedding</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="#">Dating <span class="pull-right">12</span></a></li>
```

```
        <li><a href="#">Wedding <span class="pull-right">11</span></a></li>
```

```
        <li><a href="#">Date or Wed <span class="pull-right">10</span></a></li>
```

```
        <li><a href="#">Date and Wed <span class="pull-right">14</span></a></li>
```

```
        <li><a href="#">Others <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 Profiles</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 Profiles</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="#">Looking for</a></h6>
          <div class="widget-posts-meta">Wed</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="#">Searching for</a></h6>
          <div class="widget-posts-meta">-Date-</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="#">Date or Wed</a></h6>
          <div class="widget-posts-meta">-Wed-</div>
        </div>
      </li>
    </ul>
  </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>
```

```
    </form>
```

```
    </body>
```

```
    </html>
```

```
***** DOWMasterPage.master.cs *****
```

```
using System;
```

```
using System.Collections.Generic;
```

```
using System.Linq;
```

```
using System.Web;
```

```
using System.Web.UI;
```

```
using System.Web.UI.WebControls;
```

```
using DOWEntity;
```

```
using DOWBal;
```

```
using System.Text;
```

```
//using DOWDal;
```

```
using System.Data;
```

```
using System.Web.Security;
```

```
public partial class DOWMasterPage : System.Web.UI.MasterPage
```

```
{
```

```
    string UserName;
```

```
    DDLogTracer objlog = new DDLogTracer();
```

```
    bool IsMobile = false;
```

```
    protected void Page_Load(object sender, EventArgs e)
```

```
{
```

```
    string logpath = Server.MapPath("~/log/" + "Log" + Convert.ToString(DateTime.Today.Day) + ".txt");
```

```
    Session["Logpath"] = logpath;
```

```
    if (Request.Browser.IsMobileDevice) { IsMobile = true; } else { IsMobile = false; }
```

```
    StringBuilder sb = new StringBuilder();
```

```
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedIn_Specs"];
```

```
    if (Clsusr == null)
```

```
{
```

```
    bool ret = new Common().guest_login();
```

```
    if (ret == false)
```

```
{
```

```

        Response.Redirect("~/Login.aspx");
    }
}

if (Clsusr != null)
{
    if (Clsusr.FirstName == "" || Clsusr.FirstName == null)
    {
        UserName = Clsusr.UserName;
    }
    else { UserName = Clsusr.FirstName; }
    if (Clsusr.UserName != "Guest")
    {
        var cnt = new tblUsers().GetShortListbyUserId(Convert.ToInt16(Clsusr.UserId));
        int dftcnt = cnt.Count();
        var sentlist = new tblUsers().GetSendProfilebyUserId(Convert.ToInt16(Clsusr.UserId));
        var rcvlist = new tblUsers().GetRcvdProfilebyUserId(Convert.ToInt16(Clsusr.UserId));
        sb.Append("<ul class='navbar-nav ml-auto list-inline topbar-right pull-right'>\n");
        //sb.Append("<li><a href='Dashboard.aspx'><i class='fas fa-users-cog'></i> Welcome " +
        UserName + "</a></li>\n");

        if (Convert.ToInt16(Clsusr.GroupId) > 3)
            {/ / 

            sb.Append("<li><a href='DashBoard.aspx'><i class='fas fa-tachometer'></i>"+
            Clsusr.FirstName + "'s Dashboard </a></li>\n");//(" + Clsusr.FirstName + "
            sb.Append("<li><a href='MyShortList.aspx'><i class='fas fa-user-circle'></i>Shortlist (" +
            Convert.ToString(dftcnt) + ")</a></li>\n");
            sb.Append("<li><a href='SendProfile.aspx?psid=0'><i class='fas fa-heart-circle'></i>Sent (" +
            Convert.ToString(sentlist.Count) + ")</a></li>\n");
            sb.Append("<li><a href='RcvdProfile.aspx?psid=0'><i class='fas fa-heart-square'></i>Received
            (" + Convert.ToString(rcvlist.Count) + ")</a></li>\n");
            sb.Append("<li> | <a href='UserDetail.aspx'><i class='fas fa-user'></i> Profile </a> |
            </li>\n");
            sb.Append("<li> <a href='logout.aspx'><i class='fas fa-lock'></i> Logout </a> | <a
            href='Home.aspx'><i class='fas fa-home'></i> Home </a><span><Span></li>\n");
            }
            else
            {
                sb.Append("<li> | <a href='logout.aspx'><i class='fas fa-lock'></i> Logout </a>|</li>\n");
                sb.Append("<li> <a href='Home.aspx'><i class='fas fa-home'></i> Home </a></li>\n");
            }
        }
        account.InnerHtml = sb.ToString();
    }
}

```

```

        PopDlIs ObjPD = new PopDlIs();
        //ObjPD.BindDropDowns(ddl_location, (new PopDlIs()).GetStores());
        sb = new StringBuilder();
        //if (!IsPostBack)
        if (IsMobile)
        {
//      logoheader1.Visible = false;
        }
        if (Clsusr != null)
        {
            //menu.InnerHtml = Common.Generate_newmenu();
            List<EteGetSearchResults> Lst = (new
tblUsers()).GetAcceptedProfilebyUserId(Convert.ToInt16(Clsusr.UserId));
            if (Lst.Count == 0)
            {
                shopcart.InnerHtml = "<i class='fab fa-opencart'></i>";
            }
            else
            {
                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.Aname + '' alt=''> \n");
                    sb.Append(" <div class='overflow-h'> \n");
                    sb.Append(" <a href='ProfileDetail.aspx?id=' + Common.Encrypt(list.UserName.ToString()) +
""><h4> + list.FirstName + " " + list.LastName + "</h4></a> \n");
                    sb.Append(" <small>(" + list.GroupName + ")</small><br> \n");
                    sb.Append(" </div> \n");
                    sb.Append(" <a href='ProfileDetail.aspx?id=' + Common.Encrypt(list.UserId.ToString()) + ""
style='text-decoration: underline' class=' title-price '>Profile</a><br> \n");//cd-cart-btn pull-right
                    //sb.Append(" <a href='MyHeartList.aspx?pid=' +
Common.Encrypt(list.UserId.ToString()) + "&dtype=1' style='text-decoration: underline' class='
'>Delete</a> \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> ");
    }
}
protected void imgbtn_Click(object sender, ImageClickEventArgs e)
{
    string srctext = ""; // TextBox1.Text;
    Response.Redirect("SearchList.aspx?SrchStrng=" + Common.Encrypt(srctext) + "&cat=0");
}

protected void ImageButton1_Click(object sender, ImageClickEventArgs e)
{
    //ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"]; ;

    //string cat = ddl_location.Selected.Value.ToString(); //
    Common.Encrypt(ddl_location.Selected.Value.ToString());
    //string srch = Common.Encrypt(txt_SearchString.Text);
    //Response.Redirect("~/SearchList.aspx?pageid=198&SrchStrng=" + srch + "&cat=" + cat);

}
}
}
***** 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>Date Or Wed | DASHBOARD</title>
    <!-- Favicon -->
    <link rel="icon" type="image/gif" href="images/DateOrWed.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">

```

```

<form id="form1" runat="server" >
    <asp:ScriptManager ID="ScriptManager1" runat="server">
    </asp:ScriptManager>

<!-- 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="Home.aspx" class="collapsible-header waves-effect"><i class="w-fa fas fa-home"></i>Home</a>
                </li>
                <li>
                    <a href="DashBoard.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-user-cog"></i>Profile<i class="fas fa-angle-down rotate-icon"></i>
                    </a>
                    <div class="collapsible-body">
                        <ul>
                            <li>
                                <a href="UserList.aspx" class="waves-effect">Users Info </a>
                            <li>
                                <a href="ProfileInfo.aspx" class="waves-effect">Profile Details </a>
                            </li>
                        </ul>
                    </div>
                </li>
            </ul>
        </li>
    </ul>

```

```
</li>
<li>
  <a href="ProfileSettings.aspx" class="waves-effect">Settings </a>
</li>
<li>
  <a href="ProfileAddress.aspx" class="waves-effect">Addresses </a>
</li>
<li>
  <a href="ProfileImages.aspx" class="waves-effect">Images </a>
</li>
<li>
  <a href="ProfileMapLocation.aspx" class="waves-effect">Map Location </a>
</li>
<li>
  <a href="UserChoices.aspx" class="waves-effect">Your Choices </a>
</li>
<li>
  <a href="ManageLogistics.aspx" class="waves-effect">Logistics </a>
</li>
<li>
  <a href="ManageOrderRet.aspx" class="waves-effect">Order returned </a>
</li>--%>
<li>
  <a href="ManageCommunications.aspx" class="waves-effect">Communications </a>
</li>
</ul>
</div>
</li>
<li hidden="hidden">
  <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>
    </ul>
  </div>
</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>
        <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 hidden="hidden">
    <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>
<!-- 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="ContentMainPage" 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">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: {
        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>
</form>
</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 DOWEntity;
using DOWBal;
using System.Text;
using System.Data;
using System.Web.Security;

public partial class VendorMasterPage : System.Web.UI.MasterPage
{
    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"];
        if (Clsusr == null)
        {
            bool ret = new Common().guest_login();
            if (ret == false)
            {
                Response.Redirect("~/Login.aspx");
            }
        }
    }
    protected void guest_login(string user)
    {
        string City = "Hyderabad";// _City;
        string Country = "India";// _Country;

        BALCreateUser objlogin = new BALCreateUser();
        DataTable status_login = objlogin.LoginGuest(Common.Encrypt(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 = new
tblUsers().getusername(Convert.ToInt16(objLoggedInUser.UserId));
            //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 = objLoggedInUser.UserName;
myCookie.Expires = now.AddYears(1);
Response.Cookies.Add(myCookie);
```

```
}
```

```
}
```

```
}
```

This concludes creation master pages

Chapter 7

Create Application windows inheriting the master windows
First, these are a few files to be added to the app_code folder

```
***** Common.cs *****
```

```
using DOWBal;
using DOWDal;
using DOWEntity;
using System;
using System.Collections;
using System.Collections.Generic;
using System.Data;
using System.IO;
using System.Linq;
using System.Net;
using System.Security.Cryptography;
using System.Text;
using System.Web;
using System.Web.UI.HtmlControls;
using System.Web.UI.WebControls;
using System.Web.Security;
using System.Web.Script.Serialization;
//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 key = "cmee-2jh5-alrr45";
    public static string salt = "xxxxxxxxxxxxxxxxxxxxxx"; //add your own salt
    string City = "Mumbai"; // _City;
    string Country = "India"; // _Country;

    //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 Encrypt(string input)
{
    string output = new tblUsers().Encrypt(input);
    return output;
}
```

```
public static string Decrypt(string input)
{
    try
    {
        if (input != "" && input != null)
        {
            string output = new tblUsers().Decrypt(input);
            return output;
        }
        else { return ""; }
    }
    catch
    {
        return "";
    }
}
```

```
public static string EncryptQueryString(string inputText) //, string key, string salt)
{
    string output = "";

    return output;
}
```

```
public static string DecryptQueryString(string inputText)//, string key, string salt)
{
    string output = "";
    return output;
}
```

```
public bool guest_login()
{
    bool ret = false;
    try
    {
        string user = "";
    }
}
```

```

HttpCookie myCookie = HttpContext.Current.Request.Cookies["DOWCookie"];
if (myCookie != null)
{
    user = myCookie.Value.ToString();
}
else
{
    user = HttpContext.Current.User.Identity.Name;
}
if (user == "" || user == null)
{
    user = Encrypt("Guest");
}
string City = "Hyderabad";// _City;
string Country = "India";// _Country;
string ipaddress = GetIPAddress();
string machinename = "";
GetLocation(ipaddress);

BALCreateUser objlogin = new BALCreateUser();
DataTable status_login = objlogin.LoginGuest(user, ipaddress, machinename, 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;
    objLoggedInUser.FirstName = status_login.Rows[0]["FirstName"].ToString();
    objLoggedInUser.LastName = status_login.Rows[0]["LastName"].ToString();
    HttpContext.Current.Session["LoggedUser_Specs"] = objLoggedInUser;

    FormsAuthentication.SetAuthCookie(objLoggedInUser.UserName, false);
    myCookie = new HttpCookie("DOWCookie");
    DateTime now = DateTime.Now;
    myCookie.Value = objLoggedInUser.UserName;
    myCookie.Expires = now.AddYears(1);

    myCookie.Secure = true;
}

```

```

        HttpContext.Current.Response.Cookies.Add(myCookie);
        ret = true;
    }
}
catch (Exception exc)
{
    objlog.Trace_Error(4, "DateOrWed", "Common", "LoginGuest", "Exception:" + exc.ToString());
    return ret;
}

return ret;
}
public List<Location> GetLocation(string ipAddress)
{
    List<Location> locations = new List<Location>();
    //string ipAddress = Request.ServerVariables["HTTP_X_FORWARDED_FOR"];
    //if (string.IsNullOrEmpty(ipAddress))
    //{
    //    ipAddress = Request.ServerVariables["REMOTE_ADDR"];
    //}

    string APIKey = "744627112f51a720c93d8e11741a02c6f50ad135ea28e733d10f52dc0596b193";
    string url = string.Format("http://api.ipinfodb.com/v3/ip-city/?key={0}&ip={1}&format=json",
APIKey, ipAddress);
    using (WebClient client = new WebClient())
    {
        string json = client.DownloadString(url);
        Location location = new JavaScriptSerializer().Deserialize<Location>(json);
        locations.Add(location);
        City = locations[0].CityName;
        Country = locations[0].CountryName;
    }
    return locations;
}
public string GetIPAddress()
{
    String address = "";
    try
    {
        WebRequest request = WebRequest.Create("http://checkip.dyndns.org/");
        using (WebResponse response = request.GetResponse())
        using (StreamReader stream = new StreamReader(response.GetResponseStream()))
        {
            address = stream.ReadToEnd();
        }

        int first = address.IndexOf("Address: ") + 9;

```

```

        int last = address.LastIndexOf("</body>");
        address = address.Substring(first, last - first);
    }
    catch { }
    return address;
}

public static string BuildRecentPlaces(int userid)
{
    StringBuilder sbuilder = new StringBuilder();

    BalRecentPlacess ObjRcntPlaces = new BalRecentPlacess();
    string ClsHLighter = string.Empty;

    //List<UserPermissions> UserPerSets = BalPerGrp.GetPersets_usr(userName);
    List<EteRecentPlaces> Ist_RcntPlaces = ObjRcntPlaces.GetRecentPlacess(userid);
    if (Ist_RcntPlaces.Count > 0)
    {
        string sType = "";
        sbuilder.Append("<div class='text'> <a href='#'><i class='fa fa-tasks'></i></a>");
        foreach (EteRecentPlaces row in Ist_RcntPlaces)
        {
            if (sType != row.rp_type.ToString().Replace(" ", ""))
            {
                sType = row.rp_type.ToString();
                sbuilder.Append("&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;" + row.rp_type.ToString() + ": <a
href=" + row.rp_url + ">" + row.rp_heading + "</a>");
            }
        }
        sbuilder.Append("</div>");
    }

    return sbuilder.ToString();
}

public static string BuildMenu(string userName, string ReqFileName)
{
    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.Pageld ascending
                select Pset;
            var PgLanding = Childs.Take(1);

            foreach (UserPermissions pglnd 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='" + pgland.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 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;
}
public string GetIp()
{
    String address = "";
    WebRequest request = WebRequest.Create("http://checkip.dyndns.org/");
    using (WebResponse response = request.GetResponse())
    using (StreamReader stream = new StreamReader(response.GetResponseStream()))
    {
        address = stream.ReadToEnd();
    }

    int first = address.IndexOf("Address: ") + 9;
    int last = address.LastIndexOf("</body>");
    address = address.Substring(first, last - first);

    return address;
}

```

```

#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"].ToString() == "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"].ToString() == "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 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)
        {
            //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"].T
oString();
                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["lbtntype"].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

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 = "<span><span><i class='fa fa-check-circle'></i> " + message;
```

```
            div.Attributes["class"] = "alert alert-success";
```

```
        }
```

```
        else
```

```
        {
```

```
            Msg = "<span><span><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"].ToString() != null || drnew["lbtnimage"].ToString() != "")
{
    imgUrl = string.Format("<i class='{0}'></i>", drnew["lbtnimage"].ToString());
}
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"].ToString() != null || drnew["lbtnimage"].ToString() != "")
                {
                    imgUrl = string.Format("<i class='{0}'></i>", drnew["lbtnimage"].ToString());
                }
                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
}

```

```

***** DDLogTracer.cs *****

```

```

using System;
using System.Configuration;
using System.IO;

```

```

//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="Log";

```

```

    //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\\\\" + "\\\"));
}
else
{
    filepath = (Logpath); // + (DateTime.Now.ToString("yyyy\\\\" + "\\\"));
}
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;
}
}
}
//}
***** ExceptionLogging.cs *****
using System;
using System.Configuration;
using System.Data;
using System.Data.SqlClient;
using context = System.Web.HttpContext;

/// <summary>
/// Summary description for ExceptionLogging
/// article by Vithal Wadje

/// </summary>
public static class ExceptionLogging
{

    private static String exepurl;
    static SqlConnection con;
    private static void connection()
    {
        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();
    }
}
***** 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.WriteLine("Inner Exception Type: ");
            sw.WriteLine(exc.InnerException.GetType().ToString());
            sw.WriteLine("Inner Exception: ");
            sw.WriteLine(exc.InnerException.Message);
            sw.WriteLine("Inner Source: ");
            sw.WriteLine(exc.InnerException.Source);
            if (exc.InnerException.StackTrace != null)
            {
                sw.WriteLine("Inner Stack Trace: ");
                sw.WriteLine(exc.InnerException.StackTrace);
            }
        }
        sw.WriteLine("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
}
}
***** FieldValidator.cs *****
using System;
using System.Web.UI.WebControls;

/// <summary>
/// Summary description for FieldValidator
/// </summary>
public class FieldValidator
{
    public FieldValidator()
    {
        //
        // TODO: Add constructor logic here
        //
    }
    public bool IsValidEmail(string email)
    {
        try
        {
            var addr = new System.Net.Mail.MailAddress(email);
            return addr.Address == email;
        }
        catch
        {
            return false;
        }
    }
    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 DOWBal;
using DOWEntity;
using System;
using System.IO;
using System.Web.UI.WebControls;
using System.Windows.Forms;
/// <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; //objRow["AttachmentFilename"].ToString();
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";
}
}

```

```

//Set the contenttype based on File Extension
//switch (ext)
//{
// case ".doc":
//     contenttype = "application/vnd.ms-word";
//     break;
// case ".docx":
//     contenttype = "application/vnd.ms-word";

```

```
// break;
// case ".xls":
//     contenttype = "application/vnd.ms-excel";
//     break;
// case ".xlsx":
//     contenttype = "application/vnd.ms-excel";
//     break;
// case ".ppt":
//     contenttype = "application/ms-powerpoint";
//     break;
// case ".pptx":
//     contenttype = "application/ms-powerpoint";
//     break;
// case ".rtf":
//     contenttype = "application/rtf";
//     break;
// case ".zip":
//     contenttype = "application/zip";
//     break;
// case ".mp3":
//     contenttype = "audio/mpeg";
//     break;
// case ".tiff":
//     contenttype = "image/tiff";
//     break;
// case ".tif":
//     contenttype = "image/tiff";
//     break;
// case ".txt":
//     contenttype = "text/plain";
//     break;
// case ".jpg":
//     contenttype = "image/jpg";
//     break;
// case ".png":
//     contenttype = "image/png";
//     break;
// case ".gif":
//     contenttype = "image/gif";
//     break;
// case ".pdf":
//     contenttype = "application/pdf";
//     break;
// default:
//     contenttype = "application/octet-stream";
//     break;
//}
```

```

    return sMime;
}

//public static void btnUpload(FileUpload FUpload)
//{
//    // Read the file and convert it to Byte Array
//    string filePath = FUpload.PostedFile.FileName;
//    string filename = Path.GetFileName(filePath);
//    string ext = Path.GetExtension(filename);
//    string contenttype = String.Empty;

//    //Set the contenttype based on File Extension
//    switch (ext)
//    {
//        case ".doc":
//            contenttype = "application/vnd.ms-word";
//            break;
//        case ".docx":
//            contenttype = "application/vnd.ms-word";
//            break;
//        case ".xls":
//            contenttype = "application/vnd.ms-excel";
//            break;
//        case ".xlsx":
//            contenttype = "application/vnd.ms-excel";
//            break;
//        case ".jpg":
//            contenttype = "image/jpg";
//            break;
//        case ".png":
//            contenttype = "image/png";
//            break;
//        case ".gif":
//            contenttype = "image/gif";
//            break;
//        case ".pdf":
//            contenttype = "application/pdf";
//            break;
//    }

//    if (contenttype != String.Empty)
//    {
//        Stream fs = FUpload.PostedFile.InputStream;
//        BinaryReader br = new BinaryReader(fs);
//        Byte[] bytes = br.ReadBytes((Int32)fs.Length);

```

```

// }
// else
// {

// }
//}

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;
}
}

```

This concludes app folder files

Chapter 8

Now let us drive into developing Windows in the application

```
***** web.config *****
```

```
<?xml version="1.0"?>
<configuration>
  <configSections>
    <!--<section name="infragistics.web" type="System.Configuration.SingleTagSectionHandler,System,
Version=1.0.3300.0, Culture=neutral, PublicKeyToken=b77a5c561934e089" />-->
    <sectionGroup name="dotNetOpenAuth"
type="DotNetOpenAuth.Configuration.DotNetOpenAuthSection, DotNetOpenAuth.Core">
      <section name="oauth" type="DotNetOpenAuth.Configuration.OAuthElement,
DotNetOpenAuth.OAuth" requirePermission="false" allowLocation="true"/>
      <section name="openid" type="DotNetOpenAuth.Configuration.OpenIdElement,
DotNetOpenAuth.OpenId" requirePermission="false" allowLocation="true"/>
      <section name="messaging" type="DotNetOpenAuth.Configuration.MessagingElement,
DotNetOpenAuth.Core" requirePermission="false" allowLocation="true"/>
      <section name="reporting" type="DotNetOpenAuth.Configuration.ReportingElement,
DotNetOpenAuth.Core" requirePermission="false" allowLocation="true"/>
    </sectionGroup>
    <!-- For more information on Entity Framework configuration, visit
http://go.microsoft.com/fwlink/?LinkID=237468 -->
    <section name="entityFramework"
type="System.Data.Entity.Internal.ConfigFile.EntityFrameworkSection, EntityFramework,
Version=6.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089" requirePermission="false"/>
  </configSections>
  <!--<infragistics.web styleSetName="Default" styleSetPath="~/ig_res" enableAppStyling="true" />-->
  <system.webServer>
    <validation validateIntegratedModeConfiguration="false"/>
    <defaultDocument>
      <files>
        <add value="Home.aspx"/>
      </files>
    </defaultDocument>
    <security>
      <requestFiltering>
        <requestLimits maxAllowedContentLength="1073741824"/>
      </requestFiltering>
    </security>
    <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>
```

<!--

For a description of web.config changes see <http://go.microsoft.com/fwlink/?LinkId=235367>.

The following attributes can be set on the <httpRuntime> tag.

```
<system.Web>
  <httpRuntime targetFramework="4.5" />
</system.Web>
-->
<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.WebRequest, 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" maxRequestLength="1048576"/>
  <authentication mode="None">
    <!--<forms loginUrl="login.aspx" name="DateorWed.com" timeout="696000"/>-->
  </authentication>
  <sessionState cookieless="AutoDetect" customProvider="DefaultSessionProvider" mode="InProc"
timeout="49600">
    <providers>
      <add name="DefaultSessionProvider" type="System.Web.Providers.DefaultSessionStateProvider,
System.Web.Providers, Version=2.0.0.0, Culture=neutral, PublicKeyToken=31bf3856ad364e35"
connectionStringName="DefaultConnection"/>
    </providers>
  </sessionState>
  <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"/>
<profile defaultProvider="DefaultProfileProvider">
    <providers>
        <add name="DefaultProfileProvider" type="System.Web.Providers.DefaultProfileProvider,
System.Web.Providers, Version=2.0.0.0, Culture=neutral, PublicKeyToken=31bf3856ad364e35"
connectionStringName="DefaultConnection" applicationName="/" />
    </providers>
</profile>
<membership defaultProvider="DefaultMembershipProvider">
    <providers>
        <add name="DefaultMembershipProvider"
type="System.Web.Providers.DefaultMembershipProvider, System.Web.Providers, Version=2.0.0.0,
Culture=neutral, PublicKeyToken=31bf3856ad364e35" connectionStringName="DefaultConnection"
enablePasswordRetrieval="false" enablePasswordReset="true" requiresQuestionAndAnswer="false"
requiresUniqueEmail="false" maxInvalidPasswordAttempts="5" minRequiredPasswordLength="6"
minRequiredNonalphanumericCharacters="0" passwordAttemptWindow="10" applicationName="/" />
    </providers>
</membership>
<roleManager defaultProvider="DefaultRoleProvider">
    <providers>
        <add name="DefaultRoleProvider" type="System.Web.Providers.DefaultRoleProvider,
System.Web.Providers, Version=2.0.0.0, Culture=neutral, PublicKeyToken=31bf3856ad364e35"
connectionStringName="DefaultConnection" applicationName="/" />
    </providers>
</roleManager>
<pages>
    <controls>
        <add assembly="Microsoft.AspNet.Web.Optimization.WebForms"
namespace="Microsoft.AspNet.Web.Optimization.WebForms" tagPrefix="webopt"/>
        <add tagPrefix="ajaxToolkit" assembly="AjaxControlToolkit" namespace="AjaxControlToolkit"/>
    </controls>
</pages>
</system.web>
<!--<appSettings></appSettings-->
<appSettings>
    <add key="owin:AutomaticAppStartup" value="true"/>
    <add key="owin:appStartup" value="SignalRChat.Startup"/>
    <!--<add key="DOWConectingString" value="Data Source=SQL5042.site4now.net;Initial
Catalog=DB_9FAA34_DOWDB;Persist Security Info=True;User
ID=DB_9FAA34_DOWDB_admin;Password=myDOW123;Timeout=2200" />-->
    <add key="DOWConectingString" value="Data Source=raviw10;Initial Catalog=DateOrWedDb;Persist
Security Info=True;User ID=SA;Password=raks1204;Timeout=2200"/>
    <!--*****logconfig parameters*****-->
    <add key="GoogleMapsKey" value="AlzaSyBSxyfN6Qe4VZNXau0NRbSIW4dgJWtYOLU"/>
    <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="SSRSPPrimaryURL" 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="remote.eze-talk.com"/>
<!--mail.eze-talk.com-->
<add key="mailFrom" value="Provisioning@eze-talk.com"/>
<!--provisioning@eze-talk.com-->
<add key="SmtpPort" value="465"/>
<!--25-->
<add key="SmtpServerUser" value="Provisioning"/>
<!--ezetalk\Synverse-->
<add key="SmtpServerPassword" value="Dedodade321"/>
<!--aK4Hhi4M8W9V-->
<!--Email Server Configuration Ends-->
</appSettings>
<connectionStrings>
  <add name="DOWConectingString" connectionString="Data Source=raviw10;Initial
Catalog=DateOrWedDb;Persist Security Info=True;User ID=sa;Password=raks1204"
providerName="System.Data.SqlClient"/>
  <!--<add name="DOWConectingString" connectionString="Data Source=SQL5042.site4now.net;Initial
Catalog=DB_9FAA34_DOWDB;Persist Security Info=True;User
ID=DB_9FAA34_DOWDB_admin;Password=myDOW123" providerName="System.Data.SqlClient" />-->
  <add name="DefaultConnection" providerName="System.Data.SqlClient" connectionString="Data
Source=(LocalDb)\v11.0;Initial Catalog=aspnet-DateOrWed-20200120210557;Integrated
Security=SSPI;AttachDBFilename=|DataDirectory|\aspnet-DateOrWed-20200120210557.mdf"/>
</connectionStrings>
<runtime>
  <assemblyBinding xmlns="urn:schemas-microsoft-com:asm.v1">
    <dependentAssembly>
      <assemblyIdentity name="Newtonsoft.Json" publicKeyToken="30ad4fe6b2a6aeed"
culture="neutral"/>

```

```

    <bindingRedirect oldVersion="0.0.0.0-13.0.0.0" newVersion="13.0.0.0"/>
  </dependentAssembly>
  <dependentAssembly>
    <assemblyIdentity name="DotNetOpenAuth.AspNet" publicKeyToken="2780ccd10d57b246"
culture="neutral"/>
    <bindingRedirect oldVersion="0.0.0.0-4.3.0.0" newVersion="4.3.0.0"/>
  </dependentAssembly>
  <dependentAssembly>
    <assemblyIdentity name="DotNetOpenAuth.Core" publicKeyToken="2780ccd10d57b246"
culture="neutral"/>
    <bindingRedirect oldVersion="0.0.0.0-4.3.0.0" newVersion="4.3.0.0"/>
  </dependentAssembly>
  <dependentAssembly>
    <assemblyIdentity name="WebGrease" publicKeyToken="31bf3856ad364e35" culture="neutral"/>
    <bindingRedirect oldVersion="0.0.0.0-1.6.5135.21930" newVersion="1.6.5135.21930"/>
  </dependentAssembly>
  <dependentAssembly>
    <assemblyIdentity name="Antlr3.Runtime" publicKeyToken="eb42632606e9261f"
culture="neutral"/>
    <bindingRedirect oldVersion="0.0.0.0-3.5.0.2" newVersion="3.5.0.2"/>
  </dependentAssembly>
  <dependentAssembly>
    <assemblyIdentity name="Microsoft.Owin" publicKeyToken="31bf3856ad364e35"
culture="neutral"/>
    <bindingRedirect oldVersion="0.0.0.0-4.2.0.0" newVersion="4.2.0.0"/>
  </dependentAssembly>
  <dependentAssembly>
    <assemblyIdentity name="AjaxControlToolkit" publicKeyToken="28f01b0e84b6d53e"
culture="neutral"/>
    <bindingRedirect oldVersion="0.0.0.0-20.1.0.0" newVersion="20.1.0.0"/>
  </dependentAssembly>
</assemblyBinding>
<!-- When targeting ASP.NET MVC 3, this assemblyBinding makes MVC 1 and 2 references relink
to MVC 3 so libraries such as DotNetOpenAuth that compile against MVC 1 will work with it.
<assemblyBinding xmlns="urn:schemas-microsoft-com:asm.v1">
  <dependentAssembly>
    <assemblyIdentity name="System.Web.Mvc" publicKeyToken="31bf3856ad364e35" />
    <bindingRedirect oldVersion="1.0.0.0-3.0.0.0" newVersion="3.0.0.0" />
  </dependentAssembly>
</assemblyBinding>
-->
<!-- This prevents the Windows Event Log from frequently logging that HMAC1 is being used (when
the other party needs it). -->
  <legacyHMACWarning enabled="0"/>
</runtime>
<dotNetOpenAuth>
  <!-- This is an optional configuration section where aspects of dotnetopenauth can be customized. -->

```

```

<!-- For a complete set of configuration options see
http://www.dotnetopenauth.net/developers/code-snippets/configuration-options/ -->
<openid>
  <relyingParty>
    <security requireSsl="false">
      <!-- Uncomment the trustedProviders tag if your relying party should only accept positive
assertions from a closed set of OpenID Providers. -->
      <!--<trustedProviders rejectAssertionsFromUntrustedProviders="true">
        <add endpoint="https://www.google.com/accounts/o8/ud" />
      </trustedProviders-->
    </security>
    <behaviors>
      <!-- The following OPTIONAL behavior allows RPs to use SREG only, but be compatible
with OPs that use Attribute Exchange (in various formats). -->
      <add type="DotNetOpenAuth.OpenId.RelyingParty.Behaviors.AXFetchAsSregTransform,
DotNetOpenAuth.OpenId.RelyingParty"/>
    </behaviors>
  </relyingParty>
</openid>
<messaging>
  <untrustedWebRequest>
    <whitelistHosts>
      <!-- Uncomment to enable communication with localhost (should generally not activate in
production!) -->
      <!--<add name="localhost" />-->
    </whitelistHosts>
  </untrustedWebRequest>
</messaging>
<!-- Allow DotNetOpenAuth to publish usage statistics to library authors to improve the library. -->
<reporting enabled="true"/>
</dotNetOpenAuth>
<uri>
  <!-- The uri section is necessary to turn on .NET 3.5 support for IDN (international domain names),
which is necessary for OpenID urls with unicode characters in the domain/host name.
It is also required to put the Uri class into RFC 3986 escaping mode, which OpenID and OAuth
require. -->
  <idn enabled="All"/>
  <iriParsing enabled="true"/>
</uri>
<system.net>
  <defaultProxy enabled="true"/>
  <settings>
    <!-- This setting causes .NET to check certificate revocation lists (CRL)
before trusting HTTPS certificates. But this setting tends to not
be allowed in shared hosting environments. -->
    <!--<servicePointManager checkCertificateRevocationList="true"/>-->
  </settings>
</system.net>

```

```

<entityFramework>
  <providers>
    <provider invariantName="System.Data.SqlClient"
type="System.Data.Entity.SqlServer.SqlProviderServices, EntityFramework.SqlServer"/>
  </providers>
</entityFramework>
</configuration>

```

```

***** AcceptProfile.aspx *****

```

```

<%@ Page Title="" Language="C#" MasterPageFile="~/DOWMasterPage.master"
AutoEventWireup="true" CodeFile="AcceptProfile.aspx.cs" Inherits="AcceptProfile" %>

```

```

<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">
      <div class="col-lg-12 col-md-12 col=col-sm-12">
        <div class="row">
          <div id="dvMsgBox" class="span12 ordersave" runat="server" style="display: block;">
            <asp:Label ID="Label1" runat="server" Text="" Height="1px"></asp:Label>
          </div>
        </div>
      </div>
    </div>
  </div>

```

```

</asp:Content>

```

```

<asp:Content ID="Content4" ContentPlaceHolderID="CPH_MainContent" Runat="Server">
  <div class="content_holder">

```

```

    <!-- 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">Profiles Sent</h6>
        </div>
        <div class="row">
          <div id="NewProducts" class="new-product slider" runat="server">

          </div>
        </div>
      </div>
    </div>

```

```

</section>

<!-- 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>
<script type="text/javascript" src="plugins/FlexSlider/jquery.flexslider-min.js"></script>
<script type="text/javascript" src="js/jquery.magnific-popup.min.js"></script>
<script type="text/javascript" src="js/magnific-popup.min.js"></script>
<script type="text/javascript" src="js/slick.js"></script>
<script type="text/javascript" src="js/slick-slider.js"></script>
<script type="text/javascript" src="js/custom.js"></script>

</div>

</asp:Content>
<asp:Content ID="Content5" ContentPlaceHolderID="ContentPlaceHolder1" Runat="Server">
</asp:Content>
<asp:Content ID="Content6" ContentPlaceHolderID="ContentFooter" Runat="Server">
<!-- START FOOTER -->
<footer class="footer" id="footer" runat="server">
<div class="container">
<div class="row">
<section class="home-main-contant-style3 bg-white">
<div class="parallax-block parallaxBg">
<div class="container">
<div class="parallax-block-text">
<p>DateorWed.com - YourChoice <span class="color-green">We are a social networking
site.</span> <br> Thank you! </p>
</div>
<div class="pre-btn-group">
<a class="btn btn-product mr15" href="#">Search Now</a>
<a class="btn btn-product mr15" href="#">Post a Profile</a>
</div>
</div>
</div>
</div>
</section>
<!-- End Parallax -->

<!-- START NEW -->
<section class="home-main-contant-style3 bg-white" style="display:none">
<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>
</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="fab 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="#"><i class="fab fa-facebook"></i></a></li>
      <li><a href="#"><i class="fab fa-twitter"></i></a></li>
      <li><a href="#"><i class="fab fa-tumblr"></i></a></li>
      <li><a href="#"><i class="fab fa-instagram"></i></a></li>
      <li><a href="#"><i class="fab fa-linkedin"></i></a></li>
    </ul>
  </div>
</div>
</div>
</div>
</div>
</div>
<!--End Suvscribe -->
<!-- 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:2-76 Shanthinagar, Hyderabad, Telangana, India, Pin:500070</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 an Agent 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="#">dating</a></li>
        <li><a href="#">Dating & Wedding</a></li>
        <li><a href="#">Wedding</a></li>
        <li><a href="#">Option</a></li>
        <li><a href="#">Dating or Wedding</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="#">Dating <span class="pull-right">12</span></a></li>
            <li><a href="#">Wedding <span class="pull-right">11</span></a></li>
            <li><a href="#">Date or Wed <span class="pull-right">10</span></a></li>
            <li><a href="#">Date and Wed <span class="pull-right">14</span></a></li>
            <li><a href="#">Others <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 Profiles</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 Profiles</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="#">Looking for</a></h6>
  <div class="widget-posts-meta">Wed</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="#">Searching for</a></h6>
    <div class="widget-posts-meta">-Date-</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="#">Date or Wed</a></h6>
    <div class="widget-posts-meta">-Wed-</div>
  </div>
</li>
</ul>
</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>
```

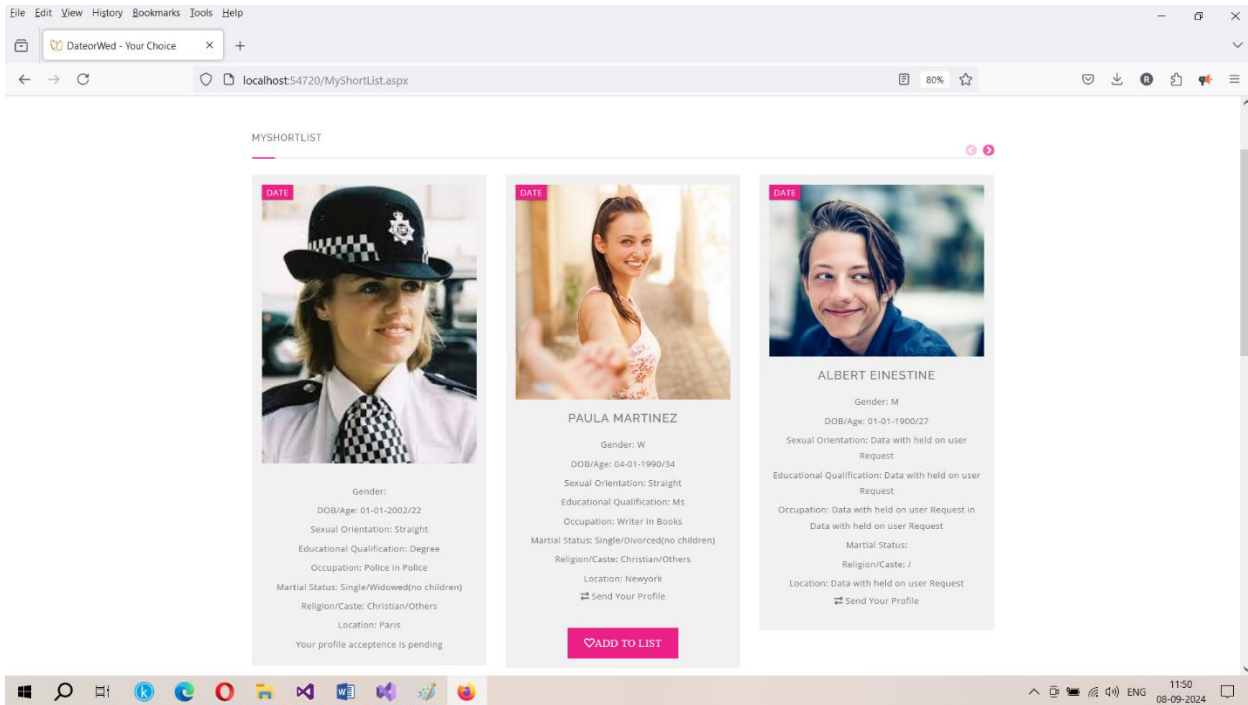
```
</div>
```

```
</div>
```

```
</footer>
```

```
<!-- END FOOTER-->
```

```
</asp:Content>
```



***** AcceptProfile.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 DOWEntity;
using System.IO;
public partial class AcceptProfile : System.Web.UI.Page
{
    DDLogTracer objlog = new DDLogTracer();
    bool IsMobile = false;
    protected void Page_Load(object sender, EventArgs e)
    {
        // string SrchStrng = "";
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr == null)
        {
            bool ret = new Common().guest_login();
            if (ret == false)
            {
                Response.Redirect("~/Login.aspx");
            }
        }
        if (Clsusr != null)
    
```

```

{
    string psid = (Request.QueryString["psid"]);
    if (psid != null)
    {
        psid = Common.Decrypt(psid);
//        int pid = new tblUsers().getuserid(psid);
        EteUserShortList lst = new EteUserShortList();

        lst.PSID = Convert.ToInt16(Clsusr.UserId);
        lst.UserId = Convert.ToInt16(psid);
        lst.UId = Convert.ToInt16(Clsusr.UserId);

        lst.CreatedOn = DateTime.Today;
        lst.Active = true;
        bool result = new tblUsers().UpdateSendProfile(lst, true);

    }    { Bindetails(psid); }

}
}
private void Bindetails(string SrchStrng)
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"]; ;
    int sno = 0;
    string cat = "";
    string PreHtml = "";
    StringBuilder sb = new StringBuilder();

    cat = Common.Decrypt(Request.QueryString["cat"]).Trim();
//    int len = cat.Length;
//    cat = cat.Substring(0, len - 15);
//    cat = cat.TrimEnd('\0');

//    sb.Append("<div class='directory-list'>");
    Etebluser_options opt = new Etebluser_options();

//try
{
    List<EteGetSearchResults> Lst = (new
tblUsers()).GetAcceptedProfilebyUserId(Convert.ToInt16(Clsusr.UserId));
    if (Lst.Count > 0)
    {
        var pst = (from c in Lst orderby c.UserId descending select c).FirstOrDefault();
        Lst=(from c in Lst orderby c.UserId descending select c).ToList();
        if (Lst != null && Lst.Count > 0)
        {

```

```

//IList<EteGetSearchResults> Lst1 = new List<EteGetSearchResults>();

foreach (var dt in Lst)
{
    string path = "images/NoPhoto.png";
    string path1 = "images/No Photo.png";
    string href = "";
    string OutId = Common.Encrypt(dt.UserName.ToString());
    href = "ProfileDetail.aspx?id=" + OutId;
    var list = new tblUsers().getuserImagesbyuserid((int)dt.UserId, "").ToList();
    int cnt = list.Count();
    if (cnt > 1)
    {
        path = list[0].ImgPath;
        path1 = list[1].ImgPath;//"http://dateorwed.com/" +
    }
    else
        if (list.Count() > 0)
        {
            path = list[0].ImageUrl;
        }

    if (path == null || path == "")
    {
        path = "images/Nocolor.png";
    }
    if (File.Exists(path))
    {
        path = list[0].ImageUrl;
        path1 = list[1].ImageUrl;
    }
    if (path1 == null || path1 == "") { path1 = path; }

    string martsttus = new tblUsers().getmatstatus(dt.Martialstatus);
    string relgn = new tblUsers().getreligion(dt.Religion);
    string caste = new tblUsers().getcaste(dt.Caste);
    //slick code
    sb.Append("<div class='col-lg-3 col-md-4 item'> \n");
    sb.Append("<div class='products grid-product'> \n");
    sb.Append("<div class='product-grid-inner'>\n");
    sb.Append("<div class='product-grid-img'> \n");
    sb.Append("<ul class='product-labels'> \n");
    sb.Append("<li>Date</li> \n");
    sb.Append("</ul> \n");
    sb.Append("<a href='ProfileDetail.aspx?id=" + OutId + "' class='product-grid-img-
thumbnail'> \n");

```

```

        sb.Append("      <img src='" + path + "' alt='img' class='img-responsive primary-img' />
\n");
    sb.Append("      <img src='" + path1 + "' alt='img' class='img-responsive socendary-img'
/> \n");
    sb.Append("    </a> \n");
    sb.Append("    <div class='product-grid-atb text-center'><a class="
href='ProfileDetail.aspx?id=" + OutId + "'>Detailed View</a></div> \n");
    sb.Append("  </div> \n");
    sb.Append("  <div class='product-grid-caption text-center'> \n");
    sb.Append("  <h4><a class='p-grid-title' href='ProfileDetail.aspx?id=" + OutId + "'> +
dt.FirstName + " " + dt.LastName + "</a></h4> \n");
    //new code for details
    sb.Append("    <p class='mb5'>" + "Gender: " + dt.Gender + "</p> \n");
    sb.Append("<p class='mb5'> DOB/Age: " + dt.dob.ToShortDateString() + "/" + dt.age + "<p>
\n");
    sb.Append("    <p class='mb5'>" + "Sexual Orientation: " + dt.Sexornt + "</p> \n");
    sb.Append("    <p class='mb5'>" + "Educational Qualification: " + dt.Eduqualification +
"</p> \n");
    sb.Append("    <p class='mb5'>" + "Occupation: " + dt.Occupation + " in " +
dt.Occupationtype + "</p> \n");
    sb.Append("    <p class='mb5'>" + "Martial Status: " + martsttus + "</p> \n");
    sb.Append("    <p class='mb5'>" + "Religion/Caste: " + relgn + "/" + caste + "</p>
\n");
    sb.Append("    <p class='mb5'>" + "Location: " + dt.Location + "</p> \n");
    if (dt.Active)
    {
        if (dt.SItype)
        {
            sb.Append("    <p class='mb5 fa fa-heart'>You have accepted this Profile</p> \n");
        }
        else
        {
            sb.Append("    <p class='mb5 fa fa-heart-half'>You have not yet accepted this Profile</p>
\n");
        }
    }
    else
    {
        sb.Append("    <p class='mb5 fa fa-heart-broken'>You have Rejected this Profile</p>
\n");
    }
}
if (Convert.ToInt16(Clsusr.GroupId) > 3)
{
    //sb.Append("    <i class='fa fa-exchange-alt'></i> \n");

```

```

        //sb.Append(" <a href='SendProfile.aspx?psid=" + OutId + "'>Send Your
Profile</a><br><br> \n");
        sb.Append("<br> <i class='fa fa-heart'></i> \n");
        sb.Append(" <a href='MyShortList.aspx?psid=" + OutId + "'>Add to Shortlist</a><br>
\n");
        sb.Append("<br> <a href='MyHeartList.aspx?pid=" + OutId + "' class='btn fa fa-
heart'>Add to List</a> \n"); //<i class='fa fa-heart'>

    }

    //new code for details ends
    sb.Append(" </div> \n");
    sb.Append(" </div> \n");
    sb.Append(" </div> \n");
    sb.Append(" </div> \n");
    //grid code
    sno++;
    if (sno > 3)
    {
        sno = 1;
    }

} //for each list

PreHtml = sb.ToString();
if (PreHtml.Length > 0)
{
    objlog.Trace_Log(4, "Dashboard", "1", "NewProducts", PreHtml, "Log.ToString());
    NewProducts.InnerHtml = PreHtml;
}

}
}
else
{
    //data not found
    //Common.ShowMessageHtml("Search Options or Data not found", 1, true, this.dvMsgBox);
}
}
}

```



```

// catch (Exception ex)
//{

//Common.ShowMessageHtml(ex.Message, 1, true, this.dvMsgBox);
// ExceptionLogging.SendExcepToDB(ex);

//}
}

}
***** Dashboard.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/DashboardMasterPage.master"
AutoEventWireup="true" CodeFile="Dashboard.aspx.cs" Inherits="Dashboard" %>

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header" Runat="Server">

</asp:Content>
<asp:Content ID="Content4" ContentPlaceHolderID="CPH_TopMessages" Runat="Server">
<%-- <div class="container col-md-9 col-sm-12 col-xs-12">
    <div class="row">

        <div id="dvMsgBox" class="span12 ordersave" runat="server" style="display: block;">
            <asp:Label ID="Label2" runat="server" Text="Testing" Height="10px"></asp:Label>
        </div>
    </div>
</div> --%>
</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentMainPage" Runat="Server">
<!-- Main layout -->

    <div class="container pl-lg-5 pl-md-5">
        <div class="col-md-9 col-sm-12 col-xs-12 ml-5"> <%--ml-5--%>
            <div class="row">
                <div id="dvMsgBox" class=" col-lg-10 ordersave" runat="server" style="display: none;
height:100px">
                    <asp:Label ID="Label1" runat="server" Text="Message or Error
Messages"></asp:Label>
                    <br >
                </div>
            </div>

        </div>
    </div>
    <div class="container pl-lg-5 pl-xl-5 pl-md-0">

    <!-- Section: Intro -->

```

```
<section class="mt-md-8 pt-md-2 mb-2 pb-4 pl-xl-5 pl-lg-5 pl-md-0 pl-sm-0 hidden-xs"> <!--ml-5 --%>
```

```
<!-- Grid row -->
```

```
<div class="row pl-xl-5 pl-lg-5 pl-md-5 pl-sm-0"> <!--ml-5 new-product slider --%>
```

```
<div class=" d-sm-none d-md-block">
```

```
<!-- Grid column -->
```

```
<div class=" col-lg-3 col-md-3 col-xs-6 col-sm-6 mb-xl-0 mb-4 ml-0">
```

```
<!-- Card -->
```

```
<div class="card card-cascade cascading-admin-card">
```

```
<!-- Card Data -->
```

```
<div class="admin-up">
```

```
<i class="far fa-heart primary-color mr-3 z-depth-2"></i>
```

```
<div class="data">
```

```
<p class="text-uppercase">Dating Connections</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">Dating Statistics (25%)</p>
```

```
</div>
```

```
</div>
```

```
<!-- Card -->
```

```
</div>
```

```
<!-- Grid column -->
```

```
<!-- Grid column -->
```

```
<div class=" col-lg-3 col-md-3 col-xs-6 col-sm-6 mb-xl-0 mb-4 ml-0">
```

```
<!-- Card -->
```

```
<div class="card card-cascade cascading-admin-card">
```

```
<!-- Card Data -->
```

```
<div class="admin-up">
```

```
<i class="fas fa-church warning-color mr-3 z-depth-2"></i>
```

```
<div class="data">
```

```
<p class="text-uppercase"> Total Wedding Requests</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">Wedding Prospects (25%)</p>
  </div>

</div>
<!-- Card -->

</div>
<!-- Grid column -->

<!-- Grid column -->
<div class=" col-lg-3 col-md-3 col-xs-6 col-sm-6 mb-xl-0 mb-4 ml-0">

  <!-- Card -->
  <div class="card card-cascade cascading-admin-card">

    <!-- Card Data -->
    <div class="admin-up">
      <i class="fas fa-handshake light-blue lighten-1 mr-3 z-depth-2"></i>
      <div class="data">
        <p class="text-uppercase">Friends Connections</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">Connections with Friends</p>
    </div>

  </div>
  <!-- Card -->

</div>
<!-- Grid column -->

```

```

<!-- Grid column -->
<div class="col-lg-3 col-md-3 col-xs-6 col-sm-6 mb-xl-0 mb-4 ml-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"> Request on your Account</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>
</div>
<!-- Grid row -->

</section>
<!-- Section: Intro -->

<!-- Section: Main panel -->
<section class=" col-sm-12 mb-5 pl-xl-0 pl-lg-5 pl-md-5 pl-sm-0">

  <!-- Card -->
  <div class="card card-cascade narrower pl-lg-5 pl-md-5 "> <!--card card-cascade narrower--%>

    <!-- Section: Chart -->
    <section class="home-main-contant-style2 pl-xl-0 pl-lg-5 pl-md-5 pl-sm-0 " > <!-- dip-bg-style
dip-style home-main-contant-style2 ml-5--%>
      <div class="container">
        <div class="widget">

```

```
    <h3 class=" text-capitalize center-content ml-5">Profiles HeartListed</h3>
  </div>
```

```
</div>
```

```
<div id="HeartListed" class="new-product slider mr-4" runat="server" style="display:block"> <!--
category-products col-grid-style--%>
```

```
    <h5 class=" color-green center-content ml-5">No Profiles HeartListed</h5>
```

```
</div>
```

```
<div class="container">
  <div class="widget "><!------%>
    <h3 class=" text-capitalize center-content ml-5">Profiles Shortlisted</h3>
  </div>
```

```
</div>
```

```
<div id="AppGrid" class="new-product slider mr-4" runat="server" style="display:block"> <!--ategy-
products col-grid-style--%>
```

```
    <h5 class=" color-green center-content ml-5">No Profiles Shortlisted</h5>
```

```
</div>
```

```
<div class="container">
  <div class="widget">
    <h3 class=" text-capitalize center-content ml-5">Profiles Sent</h3>
  </div>
```

```
</div>
```

```
<div id="appsent" class="new-product slider mr-4" runat="server" style="display:block">
```

```
    <h5 class=" color-green center-content ml-5">No Profiles Sent</h5>
```

```
</div>
```

```
<div class="container">
  <div class="widget">
    <h3 class=" text-capitalize center-content ml-5">Profiles Received</h3>
  </div>
```

```
</div>
```

```
<div id="appreceived" class="new-product slider mr-4 " runat="server" style="display:block">
```

```
    <h5 class=" color-green center-content ml-5">No Profiles Received</h5>
```

</div>

</section>

<!--Section: Table-->

<script type="text/javascript" src="plugins/owl-carousel/owl.carousel.min.js"></script>

<script type="text/javascript" src="plugins/FlexSlider/jquery.flexslider-min.js"></script>

<script type="text/javascript" src="js/jquery.magnific-popup.min.js"></script>

<%-- <script type="text/javascript" src="js/magnific-popup.min.js"></script>--%>

<script type="text/javascript" src="js/slick.js"></script>

<script type="text/javascript" src="js/slick-slider.js"></script>

<script type="text/javascript" src="js/custom.js"></script>

</div>

<!-- Card -->

</section>

<!-- Section: Main panel -->

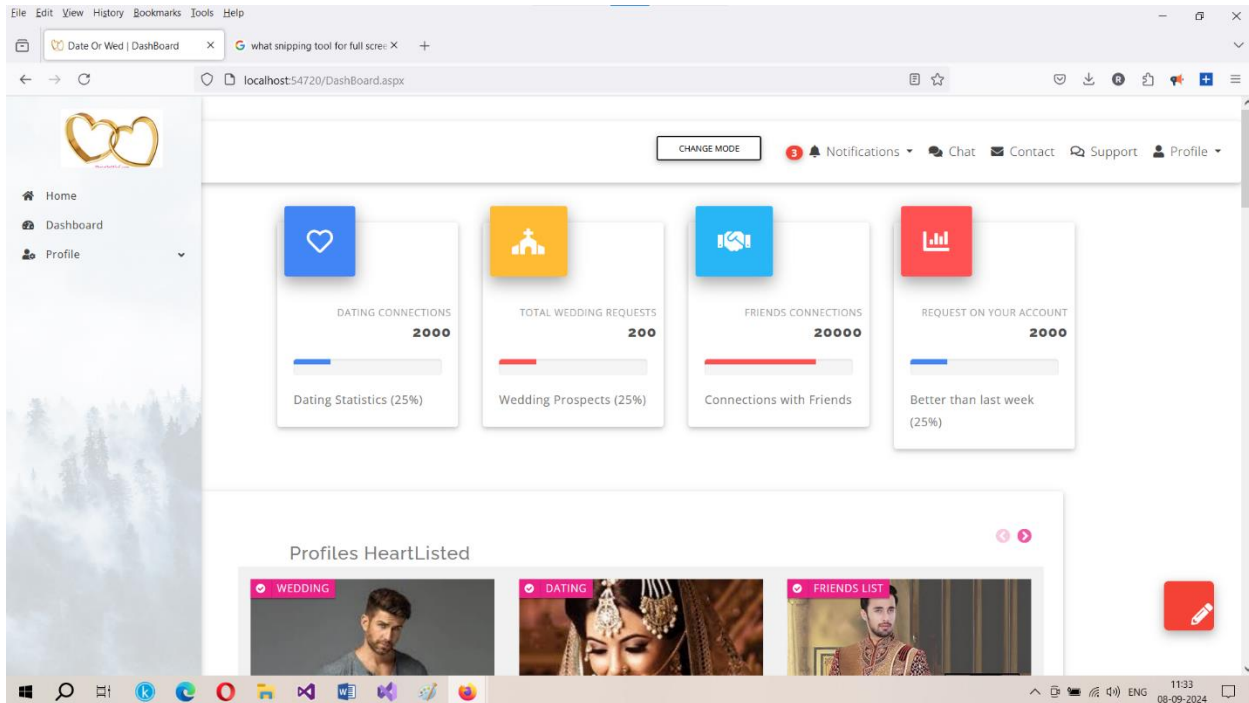
</div>

<!-- Main layout -->

</asp:Content>

<asp:Content ID="x" runat="server" ContentPlaceHolderID="ContentPlaceHolderfooter">

</asp:Content>



***** Dashboard.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 DOWEntity;
//using LogTracer;
using System.Text;
using System.Text.RegularExpressions;

public partial class DashBoard : System.Web.UI.Page
{
    DDLogTracer objlog = new DDLogTracer();
    //BalCommon blc = new BalCommon();
    int ccnt=0;
    protected void Page_Load(object sender, EventArgs e)
    {
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr == null)
        {
            bool ret = new Common().guest_login();
            if (ret == false)
            {
                Response.Redirect("~/Login.aspx");
            }
        }
    }
}

```

```

}

if (Clsusr != null)
{

    string msg = Request.QueryString["msg"];
    if (msg != null)
    {
        Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
    }
    BindetailsHeartList();
    Bindetails();
    BindetailsSent("");
    BindetailsReceived("");
    if (ccnt == 0)
    {
        //data not found
        Common.ShowMessageHtml("Data not found, there are no Shortlists or sent profiles", 1, true,
this.dvMsgBox);

    }
}
else
{
//    Response.Redirect("login.aspx");
}
}
private void Bindetails()
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"]; ;
    //string title = "";
    //title = "Profiles";
    //int sno = 0;
    string cat = "";
    string PreHtml = "";
    StringBuilder sb = new StringBuilder();
    StringBuilder sbg = new StringBuilder();

    //cat = Common.Decrypt(Request.QueryString["cat"]).Trim();
    //    int len = cat.Length;
    //    cat = cat.Substring(0, len - 15);
    //    cat = cat.TrimEnd('\0');

//    sb.Append(" <div class='directory-list'>");
    Etetbluser_options opt = new Etetbluser_options();

```



```

//try
{
    List<EteGetSearchResults> Lst = (new
tblUsers()).GetShortListbyUserId(Convert.ToInt16(Clsusr.UserId));
    ccnt += Lst.Count;
    if (Lst.Count > 0)
    {
        var pst = (from c in Lst select c).FirstOrDefault();

        //title = "ShortListed";
//      sbg.Append("<h2>" + title + "</h2>\n <small class='category-num-coun'>" +
Convert.ToString(Lst.Count) + " Profiles </small>\n");
        if (Lst != null && Lst.Count > 0)
        {

            //IList<EteGetSearchResults> Lst1 = new List<EteGetSearchResults>();

            foreach (var dt in Lst)
            {
                string path = "images/NoPhoto.png";
                string path1 = "images/No Photo.png";
                string href = "";
                string OutId = Common.Encrypt(dt.UserName.ToString());
                href = "ProfileDetail.aspx?id=" + OutId;
                var list = new tblUsers().getuserImagesbyuserid((int)dt.UserId, "").ToList();
                int cnt = list.Count();
                if (cnt > 1)
                {
                    path = list[0].ImgPath;
                    path1 = list[1].ImgPath;//"http://dateorwed.com/"+
                }
                else
                if (list.Count() > 0)
                {
                    path = list[0].ImgPath;
                }

                if (path == null || path == "")
                {
                    path = "images/Nocolor.png";
                }
                if (path1 == null || path1 == "") { path1 = path; }

                string martsttus = new tblUsers().getmatstatus(dt.Martialstatus);
                string relgn = new tblUsers().getreligion(dt.Religion);
                string caste = new tblUsers().getcaste(dt.Caste);
                //slick code
                sb.Append("      <div class='item'>          \n");//col-md-4

```

```

sb.Append("        <div class='products grid-product'>        \n");
sb.Append("        <div class='product-grid-inner'>\n");
sb.Append("        <div class='product-grid-img'> \n");
sb.Append("        <ul class='product-labels'>        \n");
sb.Append("        <li>Date</li> \n");
sb.Append("        </ul> \n");
sb.Append("        <a href='#' class='product-grid-img-thumbnail'>\n");
sb.Append("        <img src="" + path + "" alt='img' class='img-responsive primary-img' />
\n");
sb.Append("        <img src="" + path1 + "" alt='img' class='img-responsive socendary-img'
/> \n");
sb.Append("        </a> \n");
sb.Append("        <!-- <div class='product-grid-atb text-center'><i class='fa fa-shopping-
cart'></i><a class="" href='#'>Add to Cart</a></div --> \n");
sb.Append("        </div> \n");
sb.Append("        <div class='product-grid-caption text-center'> \n");
sb.Append("        <h4><a class='p-grid-title' href='#'>" + dt.FirstName + " " + dt.LastName
+ "</a></h4> \n");
sb.Append("        </div> \n");
sb.Append("        </div> \n");
sb.Append("        </div> \n");
sb.Append("        </div> \n");

//grid code
sbg.Append("    <div class=""> \n"); //col-lg-3 col-md-4 col-sm-6 col-xs-12
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>" + dt.GroupName + "</li> \n");
if (dt.Status)
{
    sbg.Append("    <li><i class='rating-selected fa fa-check-circle'></i></li> \n");
}
else
{
    sbg.Append("    <li><i class='rating-selected fa fa-times-circle'></i></li> \n");
}
sbg.Append("    </ul> \n");
sbg.Append("    <a href='ProfileDetail.aspx?id="" + OutId + "" 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='ProfileDetail.aspx?id=" + OutId + "'>Detailed View</a></div>        \n");
        sbg.Append("        </div> \n");
        sbg.Append("        <div class='product-grid-caption text-center'> \n");
        sbg.Append("        <ul class='list-inline overflow-h'> \n");
        sbg.Append("        <li style='text-decoration: underline' ><h4 class='title-price'><a
href='ProfileDetail.aspx?id=" + OutId + "'>" + dt.FirstName + " " + dt.LastName + "</a></h4></li>\n");
        sbg.Append("        </ul> \n");
        //sbg.Append("        <li class='pull-center'> \n");
        sbg.Append("        <p class='mb5'>" + "Gender: " + dt.Gender + "</p>        \n");
        sbg.Append("<p class='mb5'> DOB/Age: " + dt.dob.ToShortDateString() + "/" + dt.age +
"<p> \n");
        sbg.Append("        <p class='mb5'>" + "Sexual Orientation: " + dt.Sexornt + "</p> \n");
        sbg.Append("        <p class='mb5'>" + "Educational Qualification: " + dt.Eduqualification +
"</p> \n");
        sbg.Append("        <p class='mb5'>" + "Occupation: " + dt.Occupation + " in " +
dt.Occupationtype + "</p> \n");
        sbg.Append("        <p class='mb5'>" + "Marital Status: " + martsttus + "</p>        \n");
        sbg.Append("        <p class='mb5'>" + "Religion/Caste: " + relgn + "/" + caste + "</p>
\n");
        sbg.Append("        <p class='mb5'>" + "Location: " + dt.Location + "</p>        \n");
        if (Convert.ToInt16(Clsusr.GroupId) > 3)
        {
            sbg.Append("        <i class='fa fa-exchange-alt'></i>        \n");
            sbg.Append("        <a style='text-decoration: underline' href='SendProfile.aspx?psid=" +
OutId + "'>Send Your Profile</a><br>        \n");
            sbg.Append("<br>        <a style='text-decoration: underline;font-size:large;font-
style:italic' class='color-green' href='MyHeartList.aspx?pid=" + OutId + "' class=' fa fa-heart'>Add to Heart
List</a>\n"); //<i class='fa fa-heart'>
        }
        sbg.Append("        </div> \n");
        sbg.Append("        </div> \n");
        sbg.Append("        </div> \n");
        sbg.Append("        </div> \n");

        //sno++;

    } //for each list

    //sbg.Append("        </div> \n");
    //sbg.Append("        </div> \n");

```

```
PreHtml = sbg.ToString();
```

```

        if (PreHtml.Length > 0)
        {
            objlog.Trace_Log(4, "Dashboard", "1", "ShortList", PreHtml, "Log".ToString());
            AppGrid.InnerHtml = PreHtml;

        }
        else
        {
            AppGrid.InnerHtml = "";
        }

    }
}
else
{
    //data not found
    //Common.ShowMessageHtml("Search Options or Data not found", 1, true, this.dvMsgBox);

}

}
// catch (Exception ex)
//{

//Common.ShowMessageHtml(ex.Message, 1, true, this.dvMsgBox);
// ExceptionLogging.SendExcepToDB(ex);

//}
}
private void BindetailsSent(string SrchStrng)
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"]; ;
    //string title = "";
    //title = "Profiles";
    //int sno = 0;
    string cat = "";
    string PreHtml = "";
    StringBuilder sb = new StringBuilder();
    StringBuilder sbg = new StringBuilder();

    //cat = Common.Decrypt(Request.QueryString["cat"]).Trim();
    // int len = cat.Length;
    // cat = cat.Substring(0, len - 15);
    // cat = cat.TrimEnd('\0');

    sb.Append(" <div class='directory-list'>");

```

```

Etetbluser_options opt = new Etetbluser_options();

//try
{
    List<EteGetSearchResults> Lst = (new
tblUsers()).GetSendProfilebyUserId(Convert.ToInt16(Clsusr.UserId));
    ccnt = ccnt + Lst.Count;
    if (Lst.Count > 0)
    {
        var pst = (from c in Lst select c).FirstOrDefault();

        //title = "Sent Profiles";
        //      sbg.Append("<h2>" + title + "</h2>\n <small class='category-num-coun'>" +
Convert.ToString(Lst.Count) + " Profiles </small>\n");
        if (Lst != null && Lst.Count > 0)
        {
            //!List<EteGetSearchResults> Lst1 = new List<EteGetSearchResults>();

            foreach (var dt in Lst)
            {
                string path = "images/NoPhoto.png";
                string path1 = "images/No Photo.png";
                string href = "";
                string OutId = Common.Encrypt(dt.UserName.ToString());
                href = "ProfileDetail.aspx?id=" + OutId;
                var list = new tblUsers().getuserImagesbyuserid((int)dt.UserId, "").ToList();
                int cnt = list.Count();
                if (cnt > 1)
                {
                    path = list[0].ImgPath;
                    path1 = list[1].ImgPath;//"http://dateorwed.com/"+
                }
                else
                {
                    if (list.Count() > 0)
                    {
                        path = list[0].ImgPath;
                    }
                }

                if (path == null || path == "")
                {
                    path = "images/Nocolor.png";
                }
                if (path1 == null || path1 == "") { path1 = path; }

                string martsttus = new tblUsers().getmatstatus(dt.Martialstatus);
                string relgn = new tblUsers().getreligion(dt.Religion);
                string caste = new tblUsers().getcaste(dt.Caste);
                //slick code
            }
        }
    }
}

```

```

sb.Append("
sb.Append("
sb.Append("
sb.Append("
sb.Append("
sb.Append("
sb.Append("
sb.Append("
\n");
sb.Append("
/> \n");
sb.Append("
sb.Append("
sb.Append("
sb.Append("
+ "</a></h4> \n");
sb.Append("
sb.Append("
sb.Append("
sb.Append("

//grid code
sbg.Append("
sbg.Append("
sbg.Append("
sbg.Append("
sbg.Append("
sbg.Append("
if (dt.Status)
{
    sbg.Append("
}
else
{
    sbg.Append("
}
sbg.Append("
sbg.Append("
thumbnail> \n");
sbg.Append("
\n");
sbg.Append("
/> \n");
<div class=' item'> \n");//col-md-4
<div class='products grid-product'> \n");
<div class='product-grid-inner'>\n");
<div class='product-grid-img'> \n");
<ul class='product-labels'> \n");
<li>Date</li> \n");
</ul> \n");
<a href='#' class='product-grid-img-thumbnail'>\n");
<img src="" + path + "" alt='img' class='img-responsive primary-img' />
<img src="" + path1 + "" alt='img' class='img-responsive socendary-img'
</a> \n");
<!-- <div class='product-grid-atb text-center'><i class='fa fa-shopping-
cart'></i><a class="" href='#'>Add to Cart</a></div> --> \n");
</div> \n");
<div class='product-grid-caption text-center'> \n");
<h4><a class='p-grid-title' href='#'>" + dt.FirstName + " " + dt.LastName
+ "</a></h4> \n");
</div> \n");
</div> \n");
</div> \n");
</div> \n");
</div> \n");

<div class=""> \n");//col-lg-3 col-md-4 col-sm-6 col-xs-12
<div class='products grid-product mb30'> \n");
<div class='product-grid-inner'>\n");
<div class='product-grid-img'> \n");
<ul class='product-labels'> \n");
<li>" + dt.GroupName + "</li> \n");
<li><i class='rating-selected fa fa-check-circle'></i></li> \n");
}
<li><i class='rating-selected fa fa-times-circle'></i></li> \n");
}
</ul> \n");
<a href='ProfileDetail.aspx?id="" + OutId + "" class='product-grid-img-
thumbnail'> \n");
<img src="" + path + "" alt='img' class='img-responsive primary-img' />
<img src="" + path1 + "" alt='img' class='img-responsive socendary-img'

```

```

        sbg.Append("        </a> \n");
        sbg.Append("        <div class='product-grid-atb text-center'><a class='
href='ProfileDetail.aspx?id='" + OutId + "'>Detailed View</a></div>        \n");
        sbg.Append("        </div> \n");
        sbg.Append("        <div class='product-grid-caption text-center'> \n");
        sbg.Append("        <ul class='list-inline overflow-h'> \n");
        sbg.Append("        <li style='text-decoration: underline' ><h4 class='title-price'><a
href='ProfileDetail.aspx?id='" + OutId + "'>" + dt.FirstName + " " + dt.LastName + "</a></h4></li>\n");
        sbg.Append("        </ul> \n");
        //sbg.Append("        <li class='pull-center'> \n");
        sbg.Append("        <p class='mb5'>" + "Gender: " + dt.Gender + "</p> \n");
        sbg.Append("        <p class='mb5'> DOB/Age: " + dt.dob.ToShortDateString() + "/" + dt.age +
"</p> \n");
        sbg.Append("        <p class='mb5'>" + "Sexual Orientation: " + dt.Sexornt + "</p> \n");
        sbg.Append("        <p class='mb5'>" + "Educational Qualification: " + dt.Eduqualification +
"</p> \n");
        sbg.Append("        <p class='mb5'>" + "Occupation: " + dt.Occupation + " in " +
dt.Occupationtype + "</p> \n");
        sbg.Append("        <p class='mb5'>" + "Marital Status: " + martsttus + "</p> \n");
        sbg.Append("        <p class='mb5'>" + "Religion/Caste: " + relgn + "/" + caste + "</p>
\n");
        sbg.Append("        <p class='mb5'>" + "Location: " + dt.Location + "</p> \n");
        if (Convert.ToInt16(Clsusr.GroupId) > 3)
        {
            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='MyShortList.aspx?psid='" + OutId + "'>Add to Shortlist</a>
\n");
            sb.Append("        </li></ul> \n");
            sbg.Append("        <br>        <a style='text-decoration: underline;font-size:large;font-
style:italic' class='color-green' href='MyHeartList.aspx?pid='" + OutId + "' class=' fa fa-heart'>Add to Heart
List</a>\n"); //<i class='fa fa-heart'>
        }
        sbg.Append("        </div> \n");
        sbg.Append("        </div> \n");
        sbg.Append("        </div> \n");
        sbg.Append("        </div> \n");

    } //for each list

//sbg.Append("        </div> \n");
//sbg.Append("        </div> \n");

```

```

        PreHtml = sbg.ToString();
        if (PreHtml.Length > 0)
        {
            objlog.Trace_Log(4, "ShoppingList", "1", "SentList", PreHtml, "Log".ToString());
            appsent.InnerHtml = PreHtml;
        }
        else {
            appsent.InnerHtml = "";
        }

    }
}
else
{
    //data not found
    //Common.ShowMessageHtml("Search Options or Data not found", 1, true, this.dvMsgBox);

}

}
// catch (Exception ex)
//{

//Common.ShowMessageHtml(ex.Message, 1, true, this.dvMsgBox);
// ExceptionLogging.SendExcepToDB(ex);

//}
}
private void BindetailsReceived(string SrchStrng)
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"]; ;
    //string title = "";
    //title = "Profiles";
    //int sno = 0;
    string cat = "";
    string PreHtml = "";
    StringBuilder sb = new StringBuilder();
    StringBuilder sbg = new StringBuilder();

    //cat = Common.Decrypt(Request.QueryString["cat"]).Trim();
    // int len = cat.Length;
    // cat = cat.Substring(0, len - 15);
    // cat = cat.TrimEnd('\0');

    //sb.Append("<div class='col-lg-4 col-md-4 col-sm-12'>");

```



```

sb.Append("<div class='directory-list'>");
Etebluser_options opt = new Etebluser_options();

//try
{
    List<EteGetSearchResults> Lst = (new
tblUsers()).GetRcvdProfilebyUserId(Convert.ToInt16(Clsusr.UserId));
    ccnt = ccnt + Lst.Count;
    if (Lst.Count > 0)
    {
        var pst = (from c in Lst select c).FirstOrDefault();

        //title = "ShortListed";
        //      sbg.Append("<h2>" + title + "</h2>\n <small class='category-num-coun'>" +
Convert.ToString(Lst.Count) + " Profiles </small>\n");
        if (Lst != null && Lst.Count > 0)
        {
//          sbg.Append("\n <div class='new-product slider'>\n");

//IList<EteGetSearchResults> Lst1 = new List<EteGetSearchResults>();

foreach (var dt in Lst)
{string OutId = "";
    string path = "images/NoPhoto.png";
    string path1 = "images/No Photo.png";
    string href = "";
    if (dt.UserName != "") {
        OutId = Common.Encrypt(dt.UserName.ToString());
    }

    href = "ProfileDetail.aspx?id=" + OutId;
    var list = new tblUsers().getuserImagesbyuserid((int)dt.UserId, "").ToList();
    int cnt = list.Count();
    if (cnt > 1)
    {
        path = list[0].ImgPath;
        path1 = list[1].ImgPath;//"http://dateorwed.com/"+
    }
    else
        if (list.Count() > 0)
        {
            path = list[0].ImgPath;
        }

    if (path == null || path == "")
    {
        path = "images/Nocolor.png";
    }
}
}
}

```

```

}
if (path1 == null || path1 == "") { path1 = path; }

string martsttus = new tblUsers().getmatstatus(dt.Maritalstatus);
string relgn = new tblUsers().getreligion(dt.Religion);
string caste = new tblUsers().getcaste(dt.Caste);
//slick code
sb.Append("      <div class=' item'>      \n");//col-md-4
sb.Append("      <div class='products grid-product'>      \n");
sb.Append("      <div class='product-grid-inner'>\n");
sb.Append("      <div class='product-grid-img'> \n");
sb.Append("      <ul class='product-labels'>      \n");
sb.Append("      <li>Date</li> \n");
sb.Append("      </ul> \n");
sb.Append("      <a href='#' class='product-grid-img-thumbnail'>\n");
sb.Append("      <img src="" + path + "" alt='img' class='img-responsive primary-img' />
\n");
sb.Append("      <img src="" + path1 + "" alt='img' class='img-responsive socendary-img'
/> \n");
sb.Append("      </a> \n");
sb.Append("      <!-- <div class='product-grid-atb text-center'><i class='fa fa-shopping-
cart'></i><a class="" href='#'>Add to Cart</a></div --> \n");
sb.Append("      </div> \n");
sb.Append("      <div class='product-grid-caption text-center'> \n");
sb.Append("      <h4><a class='p-grid-title' href='#'>" + dt.FirstName + " " + dt.LastName
+ "</a></h4> \n");
sb.Append("      </div> \n");
sb.Append("      </div> \n");
sb.Append("      </div> \n");
sb.Append("      </div> \n");
sb.Append("      </div> \n");

//grid code
sbg.Append("      <div class=""> \n");//col-md-4 col-sm-6 col-xs-12
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>" + dt.GroupName + "</li> \n");
if (dt.Status)
{
    sbg.Append("      <li><i class='rating-selected fa fa-check-circle'></i></li> \n");
}
else
{
    sbg.Append("      <li><i class='rating-selected fa fa-times-circle'></i></li> \n");
}
}

```

```

        sbg.Append("        </ul> \n");
        sbg.Append("        <a href='ProfileDetail.aspx?id=" + OutId + "' 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='ProfileDetail.aspx?id=" + OutId + "'>Detailed View</a></div> \n");
        sbg.Append("        </div> \n");
        sbg.Append("        <div class='product-grid-caption text-center'> \n");
        sbg.Append("        <ul class='list-inline overflow-h'> \n");
        sbg.Append("        <li style='text-decoration: underline' ><h4 class='title-price'><a
href='ProfileDetail.aspx?id=" + OutId + "'>" + dt.FirstName + " " + dt.LastName + "</a></h4></li>\n");
        sbg.Append("        </ul> \n");
        //sbg.Append("        <li class='pull-center'> \n");
        sbg.Append("        <p class='mb5'>" + "Gender: " + dt.Gender + "</p> \n");
        sbg.Append("        <p class='mb5'> DOB/Age: " + dt.dob.ToShortDateString() + "/" + dt.age +
"</p> \n");
        sbg.Append("        <p class='mb5'>" + "Sexual Orientation: " + dt.Sexornt + "</p> \n");
        sbg.Append("        <p class='mb5'>" + "Educational Qualification: " + dt.Eduqualification +
"</p> \n");
        sbg.Append("        <p class='mb5'>" + "Occupation: " + dt.Occupation + " in " +
dt.Occupationtype + "</p> \n");
        sbg.Append("        <p class='mb5'>" + "Martial Status: " + martsttus + "</p> \n");
        sbg.Append("        <p class='mb5'>" + "Religion/Caste: " + relgn + "/" + caste + "</p>
\n");
        sbg.Append("        <p class='mb5'>" + "Location: " + dt.Location + "</p> \n");
        if (Convert.ToInt16(Clsusr.GroupId) > 3)
        {
            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='MyShortList.aspx?psid=" + OutId + "'>Add to Shortlist</a>
\n");
            sb.Append("        </li></ul> \n");
            sbg.Append("        <br>        <a style='text-decoration: underline;font-size:large;font-
style:italic' class='color-green' href='MyHeartList.aspx?pid=" + OutId + "' class=' fa fa-heart'>Add to Heart
List</a>\n"); //<i class='fa fa-heart'>
        }
        sbg.Append("        </div> \n");
        sbg.Append("        </div> \n");
        sbg.Append("        </div> \n");
        sbg.Append("        </div> \n");

```

```

    }//for each list

    sbg.Append("        </div> \n");
    //sbg.Append("        </div> \n");

    PreHtml = sbg.ToString();
    if (PreHtml.Length > 0)
    {
        objlog.Trace_Log(4, "DashBoard", "1", "ProfilesReceived", PreHtml, "Log".ToString());
        appreceived.InnerHtml = PreHtml;

    }
    else
    {
        appreceived.InnerHtml = "";
    }

}
}
else
{
    //data not found
    //Common.ShowMessageHtml("Search Options or Data not found", 1, true, this.dvMsgBox);

}

}
// catch (Exception ex)
//{

//Common.ShowMessageHtml(ex.Message, 1, true, this.dvMsgBox);
// ExceptionLogging.SendExcepToDB(ex);

//}
}
private void BindetailsHeartList()
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"]; ;
// title = "Profiles";
string cat = "";
string PreHtml = "";
StringBuilder sb = new StringBuilder();
StringBuilder sbg = new StringBuilder();

```

```

//cat = Common.Decrypt(Request.QueryString["cat"]).Trim();
// int len = cat.Length;
// cat = cat.Substring(0, len - 15);
// cat = cat.TrimEnd('\0');

sb.Append(" <div class='directory-list row'>");
Etetbluser_options opt = new Etetbluser_options();

//try
{
    List<EteGetSearchResults> Lst = (new
tblUsers()).GetCartbyUserId(Convert.ToInt16(Clsusr.UserId));
    ccnt += Lst.Count;
    if (Lst.Count > 0)
    {
        //var pst = (from c in Lst select c).FirstOrDefault();

        //title = "ShortListed";
        // sbg.Append("<h2>" + title + "</h2>\n <small class='category-num-coun'>" +
Convert.ToString(Lst.Count) + " Profiles </small>\n");
        if (Lst != null && Lst.Count > 0)
        {
            //!List<EteGetSearchResults> Lst1 = new List<EteGetSearchResults>();

            foreach (var dt in Lst)
            {
                string path = "images/NoPhoto.png";
                string path1 = "images/No Photo.png";
                string href = "";
                string OutId = Common.Encrypt(dt.UserName.ToString());
                href = "ProfileDetail.aspx?id=" + OutId;
                var list = new tblUsers().getuserImagesbyuserid((int)dt.UserId, "").ToList();
                int cnt = list.Count();
                if (cnt > 1)
                {
                    path = list[0].ImgPath;
                    path1 = list[1].ImgPath;//"http://dateorwed.com/"+
                }
                else
                if (list.Count() > 0)
                {
                    path = list[0].ImgPath;
                }

                if (path == null || path == "")
                {

```

```

        path = "images/Nocolor.png";
    }
    if (path1 == null || path1 == "") { path1 = path; }

    string martsttus = new tblUsers().getmatstatus(dt.Martialstatus);
    string relgn = new tblUsers().getreligion(dt.Religion);
    string caste = new tblUsers().getcaste(dt.Caste);
    //slick code
    sb.Append("        <div class=' item'>        \n");//col-md-4 col-sm-12
    sb.Append("        <div class='products grid-product'>        \n");
    sb.Append("        <div class='product-grid-inner'>\n");
    sb.Append("        <div class='product-grid-img'> \n");
    sb.Append("        <ul class='product-labels'>        \n");
    sb.Append("        <li>Date</li> \n");
    sb.Append("        </ul> \n");
    sb.Append("        <a href='#' class='product-grid-img-thumbnail'>\n");
    sb.Append("        <img src='" + path + "' alt='img' class='img-responsive primary-img' />
\n");
    sb.Append("        <img src='" + path1 + "' alt='img' class='img-responsive socendary-img'
/> \n");
    sb.Append("        </a> \n");
    sb.Append("        <!-- <div class='product-grid-atb text-center'><i class='fa fa-shopping-
cart'></i><a class='' href='#'>Add to Cart</a></div --> \n");
    sb.Append("        </div> \n");
    sb.Append("        <div class='product-grid-caption text-center'> \n");
    sb.Append("        <h4><a class='p-grid-title' href='#'>" + dt.FirstName + " " + dt.LastName
+ "</a></h4> \n");
    sb.Append("        </div> \n");
    sb.Append("        </div> \n");
    sb.Append("        </div> \n");
    sb.Append("        </div> \n");
    sb.Append("        </div> \n");

    //grid code
    sbg.Append("        <div class=''> \n");//col-md-4 col-sm-6 col-xs-12
    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>" + dt.GroupName + "</li> \n");
    if (dt.Status)
    {
        sbg.Append("        <li><i class='rating-selected fa fa-check-circle'></i></li> \n");
    }
    else
    {
        sbg.Append("        <li><i class='rating-selected fa fa-times-circle'></i></li> \n");
    }

```

```

    }
    sbg.Append("    </ul> \n");
    sbg.Append("    <a href='ProfileDetail.aspx?id=" + OutId + "' 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='ProfileDetail.aspx?id=" + OutId + "'>Detailed View</a></div> \n");
    sbg.Append("    </div> \n");
    sbg.Append("    <div class='product-grid-caption text-center'> \n");
    sbg.Append("        <ul class='list-inline overflow-h'> \n");
    sbg.Append("        <li style='text-decoration: underline' ><h4 class='title-price'><a
href='ProfileDetail.aspx?id=" + OutId + "'>" + dt.FirstName + " " + dt.LastName + "</a></h4></li>\n");
    sbg.Append("        </ul> \n");
    //sbg.Append("        <li class='pull-center'> \n");
    sbg.Append("        <p class='mb5'>" + "Gender: " + dt.Gender + "</p> \n");
    sbg.Append("        <p class='mb5'> DOB/Age: " + dt.dob.ToShortDateString() + "/" + dt.age +
"<p> \n");
    sbg.Append("        <p class='mb5'>" + "Sexual Orientation: " + dt.Sexornt + "</p> \n");
    sbg.Append("        <p class='mb5'>" + "Educational Qualification: " + dt.Eduqualification +
"</p> \n");
    sbg.Append("        <p class='mb5'>" + "Occupation: " + dt.Occupation + " in " +
dt.Occupationtype + "</p> \n");
    sbg.Append("        <p class='mb5'>" + "Marital Status: " + martsttus + "</p> \n");
    sbg.Append("        <p class='mb5'>" + "Religion/Caste: " + relgn + "/" + caste + "</p>
\n");
    sbg.Append("        <p class='mb5'>" + "Location: " + dt.Location + "</p> \n");
    if (Convert.ToInt16(Clsusr.GroupId) > 3)
    {
        sbg.Append("        <i class='fa fa-exchange-alt'></i> \n");
        sbg.Append("        <a style='text-decoration: underline' href='SendProfile.aspx?psid=" +
OutId + "'>Send Your Profile</a><br> \n");
        //sbg.Append("<br>        <a style='text-decoration: underline;font-size:large;font-
style:italic' class='color-green' href='MyHeartList.aspx?pid=" + OutId + "' class=' fa fa-heart'>Add to Heart
List</a>\n"); //<i class='fa fa-heart'>
    }
    sbg.Append("    </div> \n");
    sbg.Append("    </div> \n");
    sbg.Append("    </div> \n");
    sbg.Append("    </div> \n");
}
} //for each list

```

```

//sbg.Append("    </div> \n");
//sbg.Append("    </div> \n");

PreHtml = sbg.ToString();
if (PreHtml.Length > 0)
{
    objlog.Trace_Log(4, "Dashboard", "1", "Heartlist", sbg.ToString(), "Log".ToString());
    HeartListed.InnerHtml = PreHtml;

}
else
{
    HeartListed.InnerHtml = "";
}

}
}
else
{
    //data not found
    //Common.ShowMessageHtml("Search Options or Data not found", 1, true, this.dvMsgBox);

}

}
// catch (Exception ex)
//{

//Common.ShowMessageHtml(ex.Message, 1, true, this.dvMsgBox);
// ExceptionLogging.SendExcepToDB(ex);

//}
}

}
***** Default.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/DOWMasterPage.master"
AutoEventWireup="true" CodeFile="Default.aspx.cs" Inherits="_Default" %>

<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">
      <div class="col-lg-12 col-md-12 col-sm-12">
        <div class="row">
          <div id="dvMsgBox" class="span12 ordersave" runat="server" style="display: block;">
            <asp:Label ID="Label1" runat="server" Text="" Height="1px"></asp:Label>
          </div>
        </div>
      </div>
    </div>
  </div>
</asp:Content>
```

```
</asp:Content>
<asp:Content ID="Content4" ContentPlaceHolderID="CPH_MainContent" Runat="Server">
  <div class="content_holder">
    <section class="clearfix hidden-xs">
      <div id="megaslider" class="carousel slide col-lg-12 col-md-12 col-sm-12" data-ride="carousel"
runat="server">
        <ol class="carousel-indicators">
          <li data-target="#corousal" data-slide-to="0" class="active"></li>
          <li data-target="#corousal" data-slide-to="1"></li>
          <li data-target="#corousal" data-slide-to="2"></li>
          <li data-target="#corousalr" data-slide-to="3"></li>
          <li data-target="#corousal" data-slide-to="4"></li>
        </ol>
        <div class="carousel-inner" role="listbox" id="corousal" runat="server">
          <div class="item active beactive">
            
            <div class="carousel-caption">
              <h2>Welcome to DateOrWed.com</h2>
              <h2>Choice is yours</h2>
            </div>
          </div>
          <div class="item">
            
            <div class="carousel-caption">
              <h2>Your Profile your Choice</h2>
              <h2>Share the way you want</h2>
            </div>
          </div>
          <div class="item">
            
            <div class="carousel-caption">
              <h2>Search the way you Like</h2>
              <h2>Dating and Wedding Portal</h2>
            </div>
          </div>
        </div>
      </div>
    </section>
  </div>
</asp:Content>
```

```

        </div>
    </div>
    <div class="item">
        
        <div class="carousel-caption">
            <h2>Share Deatils You only Want Share</h2>
            <h2>Dating and Wedding Portal</h2>
        </div>
    </div>
    <div class="item">
        
        <div class="carousel-caption">
            <h3>Share Deatils You only Want Share</h3>
            <h2>Dating and Wedding Portal</h2>
        </div>
    </div>

    </div>
    <!-- Controls -->
<%--
    <a class="left carousel-control" href="#corousal" role="button" data-slide="prev">
    </a>
    <a class="right carousel-control" href="#corousal" role="button" data-slide="next">
    </a>--%>
</div>

</section>

<!-- 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 Profiles</h6>
        </div>
        <div class="row">
            <div id="NewProducts" class="new-product slider" runat="server">

                </div>
            </div>
        </div>
    </div>

```

</section>

```
<!-- START BANNER STYLE -->
<section class="home-main-contant-style banner-style3 bg-white">
  <div class="container">
    <div class="row clearfix new-product slider">
      <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 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>
</section>
```

```

    </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 Profiles</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>date</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="#">Looking for Dating</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>Wed</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="#">Looking for Wedding</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 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>date</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="#">Looking for Wedding</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>wed</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="#">Looking for Dating or Wedding</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 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>wed</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="#">Looking for Date and Wed</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>date</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="#">Looking for Wedding</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>wed</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="#">Looking for Dating</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>date</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="#">Looking for Wedding</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>
</div>
</div>
</section>
<!-- END Featured -->
```

```
<!-- 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>
<script type="text/javascript" src="plugins/FlexSlider/jquery.flexslider-min.js"></script>
<script type="text/javascript" src="js/jquery.magnific-popup.min.js"></script>
```

```
<script type="text/javascript" src="js/magnific-popup.min.js"></script>
<script type="text/javascript" src="js/slick.js"></script>
<script type="text/javascript" src="js/slick-slider.js"></script>
<script type="text/javascript" src="js/custom.js"></script>
```

```
</div>
```

```
</asp:Content>
```

```
<asp:Content ID="Content5" ContentPlaceHolderID="ContentPlaceHolder1" Runat="Server">
```

```
</asp:Content>
```

```
<asp:Content ID="Content6" ContentPlaceHolderID="ContentFooter" Runat="Server">
```

```
<!-- START FOOTER -->
```

```
<footer class="footer" id="footer" runat="server">
```

```
<div class="container">
```

```
<div class="row">
```

```
<section class="home-main-contant-style3 bg-white">
```

```
<div class="parallax-block parallaxBg">
```

```
<div class="container">
```

```
<div class="parallax-block-text">
```

```
<p>DateorWed.com - YourChoice <span class="color-green">We are a social networking  
site.</span> <br> Thank you! </p>
```

```
</div>
```

```
<div class="pre-btn-group">
```

```
<a class="btn btn-product mr15" href="#">Search Now</a>
```

```
<a class="btn btn-product mr15" href="#">Post a Profile</a>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</section>
```

```
<!-- End Parallax -->
```

```
<!-- START NEW -->
```

```
<section class="home-main-contant-style3 bg-white" style="display:none">
```

```
<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>
</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="fab 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="#"><i class="fab fa-facebook"></i></a></li>

```

```

        <li><a href="#"><i class="fab fa-twitter"></i></a></li>
        <li><a href="#"><i class="fab fa-tumblr"></i></a></li>
        <li><a href="#"><i class="fab fa-instagram"></i></a></li>
        <li><a href="#"><i class="fab fa-linkedin"></i></a></li>
    </ul>
</div>
</div>
</div>
</div>
</div>
</div>
<!--End Suvsubscribe -->
<!-- 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 an Agent 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="#">dating</a></li>
        <li><a href="#">Dating & Wedding</a></li>
        <li><a href="#">Wedding</a></li>
        <li><a href="#">Option</a></li>
        <li><a href="#">Dating or Wedding</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="#">Dating <span class="pull-right">12</span></a></li>
```

```
      <li><a href="#">Wedding <span class="pull-right">11</span></a></li>
```

```
      <li><a href="#">Date or Wed <span class="pull-right">10</span></a></li>
```

```
      <li><a href="#">Date and Wed <span class="pull-right">14</span></a></li>
```

```
      <li><a href="#">Others <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 Profiles</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 Profiles</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="#">Looking for</a></h6>
```

```
          <div class="widget-posts-meta">Wed</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="#">Searching for</a></h6>
    <div class="widget-posts-meta">-Date-</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="#">Date or Wed</a></h6>
    <div class="widget-posts-meta">-Wed-</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:Content>
```

```
***** Default.aspx.cs *****
```

```
<%@ Page Title="" Language="C#" MasterPageFile="~/DOWMasterPage.master"
AutoEventWireup="true" CodeFile="Default.aspx.cs" Inherits="_Default" %>
```

```
<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">
      <div class=" col-lg-12 col-md-12 col-sm-12">
        <div class="row">
          <div id="dvMsgBox" class="span12 ordersave" runat="server" style="display: block;">
            <asp:Label ID="Label1" runat="server" Text="" Height="1px"></asp:Label>
          </div>
        </div>
      </div>
    </div>
  </div>
</asp:Content>
```

```
</asp:Content>
<asp:Content ID="Content4" ContentPlaceHolderID="CPH_MainContent" Runat="Server">
  <div class="content_holder">
    <section class="clearfix hidden-xs">
      <div id="megaslider" class="carousel slide col-lg-12 col-md-12 col-sm-12" data-ride="carousel"
runat="server">
        <ol class="carousel-indicators">
          <li data-target="#corousal" data-slide-to="0" class="active"></li>
          <li data-target="#corousal" data-slide-to="1"></li>
          <li data-target="#corousal" data-slide-to="2"></li>
          <li data-target="#corousalr" data-slide-to="3"></li>
          <li data-target="#corousal" data-slide-to="4"></li>
        </ol>
        <div class="carousel-inner" role="listbox" id="corousal" runat="server">
          <div class="item active beactive">
            
            <div class="carousel-caption">
              <h2>Welcome to DateOrWed.com</h2>
              <h2>Choice is yours</h2>
            </div>
          </div>
          <div class="item">
            
            <div class="carousel-caption">
              <h2>Your Profile your Choice</h2>
              <h2>Share the way you want</h2>
            </div>
          </div>
          <div class="item">
            
            <div class="carousel-caption">
              <h2>Search the way you Like</h2>
              <h2>Dating and Wedding Portal</h2>
            </div>
          </div>
        </div>
      </div>
    </section>
  </div>
</asp:Content>
```



```

        </div>
    </div>
    <div class="item">
        
        <div class="carousel-caption">
            <h2>Share Deatils You only Want Share</h2>
            <h2>Dating and Wedding Portal</h2>
        </div>
    </div>
    <div class="item">
        
        <div class="carousel-caption">
            <h3>Share Deatils You only Want Share</h3>
            <h2>Dating and Wedding Portal</h2>
        </div>
    </div>

    </div>
    <!-- Controls -->
<%--
    <a class="left carousel-control" href="#corousal" role="button" data-slide="prev">
    </a>
    <a class="right carousel-control" href="#corousal" role="button" data-slide="next">
    </a>--%>
</div>

</section>

<!-- 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 Profiles</h6>
        </div>
        <div class="row">
            <div id="NewProducts" class="new-product slider" runat="server">

                </div>
            </div>
        </div>
    </div>

```

</section>

```
<!-- START BANNER STYLE -->
<section class="home-main-contant-style banner-style3 bg-white">
  <div class="container">
    <div class="row clearfix new-product slider">
      <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 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>
</section>
```

```

        </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>
</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 Profiles</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>date</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="#">Looking for Dating</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>Wed</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="#">Looking for Wedding</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 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>date</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="#">Looking for Wedding</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>wed</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="#">Looking for Dating or Wedding</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 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>wed</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="#">Looking for Date and Wed</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>date</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="#">Looking for Wedding</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>wed</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="#">Looking for Dating</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>date</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="#">Looking for Wedding</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>
</section>
<!-- END Featured -->

```

```

<!-- 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>
<script type="text/javascript" src="plugins/FlexSlider/jquery.flexslider-min.js"></script>
<script type="text/javascript" src="js/jquery.magnific-popup.min.js"></script>

```



```
<script type="text/javascript" src="js/magnific-popup.min.js"></script>
<script type="text/javascript" src="js/slick.js"></script>
<script type="text/javascript" src="js/slick-slider.js"></script>
<script type="text/javascript" src="js/custom.js"></script>
```

```
</div>
```

```
</asp:Content>
```

```
<asp:Content ID="Content5" ContentPlaceHolderID="ContentPlaceHolder1" Runat="Server">
```

```
</asp:Content>
```

```
<asp:Content ID="Content6" ContentPlaceHolderID="ContentFooter" Runat="Server">
```

```
<!-- START FOOTER -->
```

```
<footer class="footer" id="footer" runat="server">
```

```
<div class="container">
```

```
<div class="row">
```

```
<section class="home-main-contant-style3 bg-white">
```

```
<div class="parallax-block parallaxBg">
```

```
<div class="container">
```

```
<div class="parallax-block-text">
```

```
<p>DateorWed.com - YourChoice <span class="color-green">We are a social networking  
site.</span> <br> Thank you! </p>
```

```
</div>
```

```
<div class="pre-btn-group">
```

```
<a class="btn btn-product mr15" href="#">Search Now</a>
```

```
<a class="btn btn-product mr15" href="#">Post a Profile</a>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</section>
```

```
<!-- End Parallax -->
```

```
<!-- START NEW -->
```

```
<section class="home-main-contant-style3 bg-white" style="display:none">
```

```
<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>
</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="fab 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="#"><i class="fab fa-facebook"></i></a></li>

```

```

        <li><a href="#"><i class="fab fa-twitter"></i></a></li>
        <li><a href="#"><i class="fab fa-tumblr"></i></a></li>
        <li><a href="#"><i class="fab fa-instagram"></i></a></li>
        <li><a href="#"><i class="fab fa-linkedin"></i></a></li>
    </ul>
</div>
</div>
</div>
</div>
</div>
</div>
<!--End Suvbscribe -->
<!-- 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 an Agent 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="#">dating</a></li>
        <li><a href="#">Dating & Wedding</a></li>
        <li><a href="#">Wedding</a></li>
        <li><a href="#">Option</a></li>
        <li><a href="#">Dating or Wedding</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="#">Dating <span class="pull-right">12</span></a></li>
```

```
      <li><a href="#">Wedding <span class="pull-right">11</span></a></li>
```

```
      <li><a href="#">Date or Wed <span class="pull-right">10</span></a></li>
```

```
      <li><a href="#">Date and Wed <span class="pull-right">14</span></a></li>
```

```
      <li><a href="#">Others <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 Profiles</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 Profiles</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="#">Looking for</a></h6>
```

```
          <div class="widget-posts-meta">Wed</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="#">Searching for</a></h6>
    <div class="widget-posts-meta">-Date-</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="#">Date or Wed</a></h6>
    <div class="widget-posts-meta">-Wed-</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:Content>
```

```
***** DefaultRedirectErrorPage.aspx *****
```

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="DefaultRedirectErrorPage.aspx.cs"
Inherits="DefaultRedirectErrorPage" %>
```

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
```

```
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

```
<html>
```

```
<head id="Head1" runat="server">
```

```

<title>DefaultRedirect Error Page</title>
</head>
<body>
  <form id="form1" runat="server">
    <div>
      <h2>
        DefaultRedirect Error Page</h2>
        Standard error message suitable for all unhandled errors.
        The original exception object is not available.<br />
      <br />
      Return to the <a href='Default.aspx'> Default Page</a>
    </div>
  </form>
</body>
</html>
***** DefaultRedirectErrorPage.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 DefaultRedirectErrorPage : System.Web.UI.Page
{
    protected HttpException ex = null;

    protected void Page_Load(object sender, EventArgs e)
    {
        // Log the exception and notify system operators
        ex = new HttpException("defaultRedirect");
        ExceptionUtility.LogException(ex, "Caught in DefaultRedirectErrorPage");
        ExceptionUtility.NotifySystemOps(ex);
    }
}
***** DowChat.aspx *****
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="DowChat.aspx.cs"
Inherits="SignalRChat.DowChat" %>

<%@ Register Assembly="AjaxControlToolkit" Namespace="AjaxControlToolkit" TagPrefix="ajaxToolkit"
%>
<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
  <title>DateOrWed Chat : Chat Page</title>

  <link href="ChatCss/bootstrap.css" rel="stylesheet" />

```

```
<link href="ChatCss/style.css" rel="stylesheet" />
<link href="ChatCss/font-awesome.css" rel="stylesheet" />
<link rel="stylesheet" href="ProfDet/css/magnific-popup.css"/>
<script type="text/javascript" src="ProfDet/js/magnific-popup.min.js"></script>
```

```
<script src="Scripts/jquery-3.2.1.min.js"></script>
<script src="Scripts/jquery.signalR-2.2.2.min.js"></script>
<script src="Scripts/date.format.js"></script>
```

```
<!--EmojiOneArea -->
<link href="ChatCss/emojionearea.min.css" rel="stylesheet" />
<script src="Scripts/emojionearea.js"></script>
```

```
<!--Reference the autogenerated SignalR hub script. -->
<script src="signalr/hubs"></script>
<!--Jquery slimScroll -->
<script type="text/javascript" src="Scripts/jquery.slimscroll.min.js"></script>
```

```
<script type="text/javascript">
```

```
var IntervalVal;
$(function () {

    // Declare a proxy to reference the hub.

    var chatHub = $.connection.chatHub;
    registerClientMethods(chatHub);
    // Start Hub
    $.connection.hub.start().done(function () {

        registerEvents(chatHub)

    });

    // Reset Message Counter on Hover
    $("#divChatWindow").mouseover(function () {

        $("#MsgCountMain").html('0');
        $("#MsgCountMain").attr("title", '0 New Messages');
    });

    // Display Image Preview on File Upload
    $(document).on('change', '#<%# FileUpload1.ClientID%>', function (e) {

        var tmpPath = URL.createObjectURL(e.target.files[0]);
        $("#ImgDisp").attr('src', tmpPath);
    });
});
```



```

});

// Stop Title Alert
window.onfocus = function (event) {
    if (event.explicitOriginalTarget === window) {

        clearInterval(IntervalVal);
        document.title = 'DateOrWed Chat Room';
    }
}

});

// Show Title Alert
function ShowTitleAlert(newMessageTitle, pageTitle) {
    if (document.title == pageTitle) {
        document.title = newMessageTitle;
    }
    else {
        document.title = pageTitle;
    }
}

function registerEvents(chatHub) {

    var name = '<%=# this.UserName %>';

    if (name.length > 0) {
        chatHub.server.connect(name);
    }

    // Clear Chat
    $('#btnClearChat').click(function () {

        var msg = $('#divChatWindow').html();

        if (msg.length > 0) {
            chatHub.server.clearTimeout();
            $('#divChatWindow').html("");
            $('#txtMessage').val("");
        }
    }
}

```

```

});

// Send Button Click Event
$('#btnSendMsg').click(function () {

    var msg = $('#txtMessage').val();

    if (msg.length > 0) {

        var userName = $('#hdUserName').val();

        var date = GetCurrentDateTime(new Date());

        chatHub.server.sendMessageToAll(userName, msg, date);
        $('#txtMessage').val("");
        $('#txtMessage')[0].value="";
        $('#txtMessage').html( "");
    }

});

// Send Message on Enter Button
$('#txtMessage').keypress(function (e) {
    if (e.which == 13) {
        $('#btnSendMsg').click();
    }
});

}

function registerClientMethods(chatHub) {

    // Calls when user successfully logged in
    chatHub.client.onConnected = function (id, userName, allUsers, messages, times, UName) {

        $('#hdId').val(id);
        $('#hdUserName').val(userName);//userName
        $('#spanUser').html(UName);//userName

        // Add All Users
        for (i = 0; i < allUsers.length; i++) {

            AddUser(chatHub, allUsers[i].ConnectionId, allUsers[i].UserName, allUsers[i].UserImage,
allUsers[i].LoginTime,allUsers[i].UName);
        }

        // Add Existing Messages

```

```

        for (i = 0; i < messages.length; i++) {
            AddMessage(messages[i].UserName, messages[i].Message, messages[i].Time,
messages[i].UserImage,messages[i].UName);

        }
    }

    // On New User Connected
    chatHub.client.onNewUserConnected = function (id, name, UserImage, loginDate,UName) {
        AddUser(chatHub, id, name, UserImage, loginDate, UName);
    }

    // On User Disconnected
    chatHub.client.onUserDisconnected = function (id, userName, UName) {

        $('#Div' + id).remove();

        var ctrlId = 'private_' + id;
        $('#' + ctrlId).remove();

        var disc = $('<div class="disconnect">' + userName + ' logged off.</div>');//userName

        $(disc).hide();
        $('#divusers').prepend(disc);
        $(disc).fadeIn(200).delay(2000).fadeOut(200);

    }

    chatHub.client.messageReceived = function (userName, message, time, userimg,UName) {
        //alert(UName);
        AddMessage(userName, message, time, userimg, UName);

        // Display Message Count and Notification
        var CurrUser1 = $('#hdUserName').val();
        if (CurrUser1 != userName) {

            var msgcount = $('#MsgCountMain').html();
            msgcount++;
            $('#MsgCountMain').html(msgcount);
            $('#MsgCountMain').attr("title", msgcount + ' New Messages');
            var Notification = 'New Message From ' + userName;
            IntervalVal = setInterval("ShowTitleAlert('DateOrWed Chat Room', '" + Notification + '"",
800);

        }
    }
}

```

```

chatHub.client.sendPrivateMessage = function (windowId, fromUserName, message, userimg,
CurrentDateTime,UName) {

    var ctrlId = 'private_' + windowId;
    if ($('#' + ctrlId).length == 0) {

        OpenPrivateChatBox(chatHub, windowId, ctrlId, fromUserName, userimg,UName);

    }

    var CurrUser = $('#hdUserName').val();
    var Side = 'right';
    var TimeSide = 'left';

    if (CurrUser == fromUserName) {
        Side = 'left';
        TimeSide = 'right';
    }
    else {
        var Notification = 'New Message From ' + fromUserName;
        IntervalVal = setInterval("ShowTitleAlert('DateOrWed Chat Room', '" + Notification + '"",
800);

        var msgcount = $('#' + ctrlId).find('#MsgCountP').html();
        msgcount++;
        $('#' + ctrlId).find('#MsgCountP').html(msgcount);
        $('#' + ctrlId).find('#MsgCountP').attr("title", msgcount + ' New Messages');
    }

    var divChatP = '<div class="direct-chat-msg ' + Side + '">' +
        '<div class="direct-chat-info clearfix">' +
        '<span class="direct-chat-name pull-' + Side + '">' + UName+ '</span>' +//fromUserName
        '<span class="direct-chat-timestamp pull-' + TimeSide + '">' + CurrentDateTime + '</span>' +
        '</div>' +

        '' +
        '<div class="direct-chat-text" >' + message + '</div> </div>';

    $('#' + ctrlId).find('#divMessage').append(divChatP);

    // Apply Slim Scroll Bar in Private Chat Box
    var ScrollHeight = $('#' + ctrlId).find('#divMessage')[0].scrollHeight;
    $('#' + ctrlId).find('#divMessage').slimScroll({
        height: ScrollHeight
    });
}

```

```

}

function GetCurrentDateTime(now) {

    var localdate = dateFormat(now, "dddd, mmmm dS, yyyy, h:MM:ss TT");

    return localdate;
}

function AddUser(chatHub, id, name, UserImage, date, UName) {

    var userId = $('#hdId').val();

    var code, Clist;
    if (userId == id) {

        code = $('<div class="box-comment">' +
            '<img class="img-circle img-sm" src="" + UserImage + "" alt="User Image" />' +
            '<div class="comment-text">' +
            '<span class="username">' + UName + '<span class="text-muted pull-right">' + date +
            '</span> </span></div></div>');
        //name

        Clist = $('
            <li style="background:#494949;">' +
            '<a href="#">' +
            '<img class="contacts-list-img" src="" + UserImage + "" alt="User Image" />' +

            '<div class="contacts-list-info">' +
            '<span class="contacts-list-name" id="" + id + "">' + UName + '<small class="contacts-list-date
            pull-right">' + date + '</small> </span>' //name
            '<span class="contacts-list-msg">How have you been? I was...</span></div></a > </li >');

    }
    else {

        code = $('<div class="box-comment" id="Div' + id + "">' +
            '<img class="img-circle img-sm" src="" + UserImage + "" alt="User Image" />' +
            '<div class="comment-text">' +
            '<span class="username">' + '<a id="" + id + "" class="user" >' + UName + '<a>' + '<span
            class="text-muted pull-right">' + date + '</span> </span></div></div>');
        //name

        Clist = $('
            <li>' +
            '<a href="#">' +
            '<img class="contacts-list-img" src="" + UserImage + "" alt="User Image" />' +

```

```

    ' <div class="contacts-list-info">' +
    '<span class="contacts-list-name" id="" + id + "">' + UName+ ' <small class="contacts-list-date
pull-right">' + date + '</small> </span>' +//name
    ' <span class="contacts-list-msg">How have you been? I was...</span></div></a > </li >');

```

```

var UserLink = $('<a id="" + id + "" class="user" >' + name + '<a>');
$(code).click(function () {

```

```

    var id = $(UserLink).attr('id');

```

```

    if (userId != id) {

```

```

        var ctrlId = 'private_' + id;

```

```

        OpenPrivateChatBox(chatHub, id, ctrlId, name, UserImage, UName);

```

```

    }

```

```

});

```

```

var link = $('<span class="contacts-list-name" id="" + id + "">');

```

```

$(Clist).click(function () {

```

```

    var id = $(link).attr('id');

```

```

    if (userId != id) {

```

```

        var ctrlId = 'private_' + id;

```

```

        OpenPrivateChatBox(chatHub, id, ctrlId, name, UserImage, UName);

```

```

    }

```

```

});

```

```

}

```

```

$("#divusers").append(code);

```

```

$("#ContactList").append(Clist);

```

```

}

```

```

function AddMessage(userName, message, time, userimg, UName) {

```

```

    var CurrUser = $('#hdUserName').val();

```

```

    var Side = 'right';

```

```

    var TimeSide = 'left';

```

```

    if (CurrUser == userName) {

```

```

    Side = 'left';
    TimeSide = 'right';

}

var divChat = '<div class="direct-chat-msg ' + Side + "">' +
  '<div class="direct-chat-info clearfix">' +
  '<span class="direct-chat-name pull-' + Side + "">' + UName + '</span>' + //userName
  '<span class="direct-chat-timestamp pull-' + TimeSide + "">' + time + '</span>' +
  '</div>' +

  ' <img class="direct-chat-img" src="" + userimg + "" alt="Message User Image">' +
  '<div class="direct-chat-text" >' + message + '</div> </div>';

$('#divChatWindow').append(divChat);

var height = $('#divChatWindow')[0].scrollHeight;

// Apply Slim Scroll Bar in Group Chat Box
$('#divChatWindow').slimScroll({
  height: height
});

ParseEmoji($('#divChatWindow'));

}

function OpenPrivateChatBox(chatHub, userId, ctrlId, userName, UserImage,UName) {
  var PWClass = $('#PWCount').val();

  if ($('#PWCount').val() == 'info')
    PWClass = 'danger';
  else if ($('#PWCount').val() == 'danger')
    PWClass = 'warning';
  else
    PWClass = 'info';

  $('#PWCount').val(PWClass);
  var div1 = ' <div class="col-md-4"> <div id="" + ctrlId + "" class="box box-solid box-' + PWClass + ' ' +
  'direct-chat direct-chat-' + PWClass + "">' +
  ' <div class="box-header with-border">' +
  ' <h3 class="box-title">' + UName + '</h3>' +
  //userName
  ' <div class="box-tools pull-right">' +
  ' <span data-toggle="tooltip" id="MsgCountP" title="0 New Messages" class="badge bg-' +
  PWClass + "">0</span>' +
  ' <button type="button" class="btn btn-box-tool" data-widget="collapse">' +
  ' <i class="fa fa-minus"></i>' +

```

```

        ' </button>' +
        ' <button id="imgDelete" type="button" class="btn btn-box-tool" data-widget="remove"><i
class="fa fa-times"></i></button></div></div>' +

        ' <div class="box-body">' +
        ' <div id="divMessage" class="direct-chat-messages">' +

        ' </div>' +

        ' </div>' +
        ' <div class="box-footer">' +

        ' <input type="text" id="txtPrivateMessage" name="message" placeholder="Type Message
..." class="form-control" />' +

        ' <div class="input-group">' +
        ' <input type="text" name="message" placeholder="Type Message ..." class="form-control"
style="visibility:hidden;" />' +
        ' <span class="input-group-btn">' +
        ' <input type="button" id="btnSendMessage" class="btn btn-' + PWClass + ' btn-flat"
value="send" />' +
        ' </span>' +

        ' </div>' +

        ' </div>' +
        ' </div></div>';

```

```
var $div = $(div1);
```

```
// Closing Private Chat Box
```

```
$div.find('#imgDelete').click(function () {
    $('#' + ctrlId).remove();
});
```

```
// Send Button event in Private Chat
```

```
$div.find("#btnSendMessage").click(function () {

    $textBox = $div.find("#txtPrivateMessage");
```

```
var msg = $textBox.val();
```

```
if (msg.length > 0) {
```

```
    chatHub.server.sendPrivateMessage(userId, msg);
```

```
    $textBox.val("");
```



```

    }
});

// Text Box event on Enter Button
$div.find("#txtPrivateMessage").keypress(function (e) {
    if (e.which == 13) {
        $div.find("#btnSendMessage").click();
    }
});

// Clear Message Count on Mouse over
$div.find("#divMessage").mouseover(function () {

    $("#MsgCountP").html('0');
    $("#MsgCountP").attr("title", '0 New Messages');
});

// Append private chat div inside the main div
$('#PriChatDiv').append($div);
var msgTextbox = $div.find("#txtPrivateMessage");
$(msgTextbox).emojioneArea();
}

function ParseEmoji(div) {
    var input = $(div).html();

    var output = emojione.unicodeToImage(input);

    $(div).html(output);
}

function uploadComplete(sender, args) {
    var imgDisplay = $get("imgDisplay");
    imgDisplay.src = "images/loading.gif";
    imgDisplay.style.cssText = "";
    var img = new Image();
    img.onload = function () {
        imgDisplay.style.cssText = "Display:none;";
        imgDisplay.src = img.src;
    };

    imgDisplay.src = "<%= ResolveUrl(UploadFolderPath) %>" + args.get_fileName();
    var chatHub = $.connection.chatHub;
    var userName = $('#hdUserName').val();
    var date = GetCurrentDateTime(new Date());
    var sizeKB = (args.get_length() / 1024).toFixed(2);

    var msg1;

```

```

if (IsValidateFile(args.get_fileName())) {
    if (IsImageFile(args.get_fileName())) {
        msg1 =
            '<div class="box-body">' +
            '<div class="attachment-block clearfix">' +
            '<a><img id="imgC" style="width:100px;" class="attachment-img" src="" + imgDisplay.src +
            "" alt="Attachment Image"></a>' +
            '<div class="attachment-pushed">' +
            '<h4 class="attachment-heading"><i class="fa fa-image">' + args.get_fileName() + '
</i></h4> <br />' +
            '<div id="at" class="attachment-text"> Type: ' + args.get_contentType() +
            '// Dimensions : ' + imgDisplay.height + 'x' + imgDisplay.width + ',
            '</div>' +
            '</div>' +
            '</div>' +
            '<a id="btnDownload" href="" + imgDisplay.src + "" class="btn btn-default btn-xs"
            download="" + args.get_fileName() + ""><i class="fa fa fa-download"></i> Download</a>' +
            '<button type="button" id="ShowModellmg" value="" + imgDisplay.src + "" class="btn btn-
            default btn-xs"><i class="fa fa-camera"></i> View</button>' +
            '<span class="pull-right text-muted">File Size : ' + sizeKB + ' Kb</span>' +
            '</div>';
    }
    else {
        msg1 =
            '<div class="box-body">' +
            '<div class="attachment-block clearfix">' +
            '<a></a>' +
            '<div class="attachment-pushed">' +
            '<h4 class="attachment-heading"><i class="fa fa-file-o">' + args.get_fileName() + '
            </i></h4> <br />' +
            '<div id="at" class="attachment-text"> Type: ' + args.get_contentType() +
            '</div>' +
            '</div>' +
            '</div>' +
            '<a id="btnDownload" href="" + imgDisplay.src + "" class="btn btn-default btn-xs"
            download="" + args.get_fileName() + ""><i class="fa fa fa-download"></i> Download</a>' +
            '<a href="" + imgDisplay.src + "" target="_blank" class="btn btn-default btn-xs"><i class="fa
            fa-camera"></i> View</a>' +
            '<span class="pull-right text-muted">File Size : ' + sizeKB + ' Kb</span>' +
            '</div>';
    }
    chatHub.server.sendMessageToAll(userName, msg1, date);
}

```

```

    }

    imgDisplay.src = "";
}

function uploadStarted() {
    $get("imgDisplay").style.display = "none";
}

$(document).on('click', '#ShowModellmg', function () {
    $get("ImgModal").src = this.value;
    $('#ShowPictureModal').modal('show');
});

function IsValidateFile(fileF) {
    var allowedFiles = [".doc", ".docx", ".pdf", ".txt", ".xlsx", ".xls", ".png", ".jpg",
".gif", ".mp4", ".avi", ".mov"];
    var regex = new RegExp("[a-zA-Z0-9\\s_\\.\\-:~]+(" + allowedFiles.join('|') + ")$");
    if (!regex.test(fileF.toLowerCase())) {
        alert("Please upload files having extensions: " + allowedFiles.join(', ') + " only.");
        return false;
    }
    return true;
}

function IsImageFile(fileF) {
    var ImageFiles = [".png", ".jpg", ".gif"];
    var regex = new RegExp("(" + ImageFiles.join('|') + ")$");
    if (!regex.test(fileF.toLowerCase())) {
        return false;
    }
    return true;
}

</script>

</head>
<body>

<form id="form1" runat="server">

    <asp:ScriptManager ID="ScriptManager1" runat="server"></asp:ScriptManager>
    <div class="content-wrapper">

        <header class="main-header" style="background: #bedaeb">
            <!-- Logo -->
            <a href="#" class="logo">

```

```

        <!-- logo for regular state and mobile devices -->
        <span class="logo-lg"><b>DateOrWed</b> Chat Room</span>
    </a>
    <!-- Header Navbar: style can be found in header.less -->
    <nav class="navbar navbar-static-top" role="navigation">
        <!-- Sidebar toggle button-->

        <div class="navbar-custom-menu">
            <ul class="nav navbar-nav">
                <!-- Messages: style can be found in dropdown.less-->

                <!-- User Account: style can be found in dropdown.less -->
                <li class="dropdown user user-menu">
                    <a href="#" class="dropdown-toggle" data-toggle="dropdown">
                        
                        <span class="hidden-xs"><%= this.UName %></span>
                    </a>
                    <ul class="dropdown-menu">
                        <!-- User image -->
                        <li class="user-header">
                            
                            <p style="color: #000000;">
                                <%= UName %>
                            </p>
                        </li>
                        <!-- Menu Footer-->
                        <li class="user-footer">
                            <div class="pull-left">
                                <a class="btn btn-default btn-flat" data-toggle="modal"
href="#ChangePic">Change Picture</a>
                            </div>
                            <div class="pull-right">
                                <asp:Button ID="btnSignOut" runat="server" CssClass="btn btn-default btn-flat"
Text="Sign Out" OnClick="btnSignOut_Click" />
                            </div>
                        </li>
                    </ul>
                </li>
                <!-- Control Sidebar Toggle Button -->

            </ul>
        </div>
    </nav>
</header>

<div class="row">

    <div class="col-md-8">

```

```

<!-- DIRECT CHAT PRIMARY -->
<div class="box box-primary direct-chat direct-chat-primary">
  <div class="box-header with-border">
    <h3 class="box-title" style="color: dimgrey;">Welcome to Chat Room <span
id='spanUser'></span></h3>
    <div class="box-tools pull-right">
      <button type="button" class="btn btn-box-tool" id="btnClearChat" data-
toggle="tooltip" title="Clear Chat">
        <i class="fa fa-trash-o"></i>
      </button>

      <span data-toggle="tooltip" id="MsgCountMain" title="0 New Messages"
class="badge bg-gray">0</span>
    </div>
  </div>
  <!-- /.box-header -->
  <div class="box-body">
    <!-- Conversations are loaded here -->
    <div class="box-body" id="chat-box">
      <!-- Conversations are loaded here -->

      <div id="divChatWindow" class="direct-chat-messages" style="height: 275px;">
      </div>

      <div class="direct-chat-contacts">
        <ul class="contacts-list" id="ContactList">

          <!-- End Contact Item -->
        </ul>
        <!-- /.contatcts-list -->
      </div>
      <!-- /.direct-chat-pane -->

    </div>

  </div>
  <!-- /.box-body -->
  <div class="box-footer">

    <textarea id="txtMessage"></textarea>

    <div class="input-group" style="float: right;">
      <input class="form-control" style="visibility: hidden;" />
      <span class="input-group-btn">
        <input type="button" class="btn btn-primary btn-flat" id="btnSendMsg"
value="send" />

      </span>

```

```

<asp:UpdatePanel ID="UpdatePanel2" runat="server">
    <%-- <Triggers>
    <asp:PostBackTrigger ControlID="btnFile" />
    </Triggers>--%>
    <ContentTemplate>
        <span class="upload-btn-wrapper">
            <button id="btnFile" class="btn btn-default btn-flat"><i class="glyphicon
glyphicon-paperclip"></i></button>
            <ajaxToolkit:AsyncFileUpload OnClientUploadComplete="uploadComplete"
runat="server" ID="AsyncFileUpload1"
            ThrobberID="imgLoader" OnUploadedComplete="FileUploadComplete"
OnClientUploadStarted="uploadStarted" />
        </span>
    </ContentTemplate>
</asp:UpdatePanel>
</div>
<img id="imgDisplay" src="" class="user-image" style="height: 10px;" />
</div>
<!-- /.box-footer-->
</div>
<!-- /.direct-chat -->
</div>
<!-- /.col -->
<div class="col-md-4">

    <div class="box box-solid box-primary">

        <div class="box-header with-border">
            <h3 class="box-title">Online Users <span id='UserCount'></span></h3>
        </div>
<h5 class="box-title center-block"> Click on a user to sent Private Message </h5>

        <div class="box-footer box-comments" id="divusers">
        </div>

    </div>

</div>

<div class="row">
    <div class="col-md-12">
        <div class="row" id="PriChatDiv">
        </div>
    </div>
</div>

```

```
<textarea class="form-control" style="visibility: hidden;"></textarea>
```

```
    <!--/.private-chat -->  
</div>  
</div>
```

```
<!-- /.col -->
```

```
<!-- /.col -->
```

```
    <!-- /.col -->  
</div>  
<!-- /.row -->  
</div>
```

```
<span id="time"></span>  
<input id="hdId" type="hidden" />  
<input id="PWCount" type="hidden" value="info" />  
<input id="hdUserName" type="hidden" />
```

```
<div class="modal fade" id="ChangePic" role="dialog">  
  <div class="modal-dialog" style="width: 700px">  
    <div class="modal-content">  
      <div class="modal-header bg-light-blue-gradient with-border">  
        <button type="button" class="close" data-dismiss="modal">&times;</button>  
        <h4 class="modal-title">Change Profile Picture</h4>  
      </div>  
      <div class="modal-body">  
        <div class="container">
```

```
          <asp:UpdatePanel ID="UpdatePanel1" runat="server">
```

```
            <Triggers>  
              <asp:PostBackTrigger ControlID="btnChangePicModel" />  
            </Triggers>  
            <ContentTemplate>
```

```
              <div class="row">  
                <div class="col-md-12">  
                  <table class="table table-bordered table-striped table-hover table-responsive"  
style="width: 600px">
```

```
                    <tr>
```

```
<div class="col-md-12">
  <td class="text-primary col-md-4" style="font-weight: bold;">
    <img id="ImgDisp" src="" class="user-image" style="height: 100px;" />
  </td>
  <td class="text-primary col-md-4" style="font-weight: bold;">
    <asp:FileUpload ID="FileUpload1" runat="server" class="btn btn-
default" />
  </td>
</td class="col-md-4">
  <asp:Button ID="btnChangePicModel" runat="server" Text="Update
Picture" CssClass="btn btn-flat btn-success" OnClick="btnChangePicModel_Click" />

```

```
</td>
</div>

```

```
</tr>

```

```
<tr>
  <div class="col-md-12">
    <td class="col-md-12" colspan="3"></td>
  </div>

```

```
</tr>

```

```
</table>
</div>
</div>

```

```
</div>
  </ContentTemplate>
</asp:UpdatePanel>
</div>
</div>
</div>
</div>
</div>

```

```
<div class="modal fade" id="ShowPictureModal" role="dialog" runat="server">
  <div class=" mfp-auto-cursor" >
    <div class="mfp-bg mfp-fade mfp-ready">

```



```

        <div class=" mfp-wrap mfp-gallery mfp-close-btn-in mfp-fade mfp-ready" style="overflow:
hidden auto;">
            <button type="button" class="close" data-dismiss="modal">&times;</button>

            <div class=" mfp-container mfp-s-ready mfp-image-holder">
                <div class=" mfp-content">
                    <button title="Close (Esc)" type="button" class="mfp-close" data-
dismiss="modal">x</button>

                    <figure> 
                    <figcaption><div class="mfp-bottom-bar"><div class="mfp-title"></div><div class="mfp-
counter"></div></div></figcaption></figure></div>
                </div>
            </div>
        </div>
</div>
<style>
.upload-btn-wrapper {
    position: relative;
    overflow: hidden;
    display: inline-block;
    margin-top: 5px;
}

.upload-btn-wrapper input[type=file] {
    font-size: 100px;
    position: absolute;
    left: 0;
    top: 0;
    opacity: 0;
}

.direct-chat-text img {
    width: 20px;
}
</style>

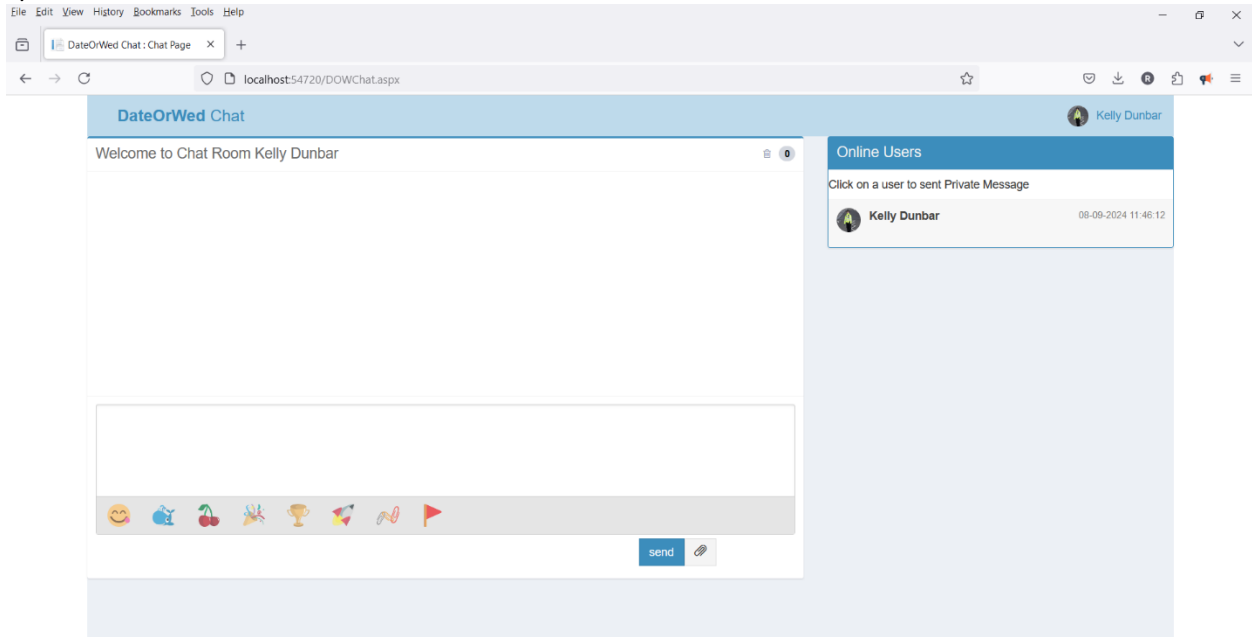
<script src="Scripts/bootstrap.min.js"></script>
<script>
    $(function () {
        $("#txtMessage").emojioneArea();

    });
</script>

</form>

```

```
</body>
</html>
```



```
***** DowChat.aspx.cs *****
```

```
using System;
using System.Collections.Generic;
using System.Drawing;
using System.Drawing.Drawing2D;
using System.IO;
using System.Linq;
using System.Web;
using System.Web.Services;
using System.Web.UI;
using System.Web.UI.WebControls;
using SD = System.Drawing;
using System.Web.UI.HtmlControls;
using DOWBal;
using DOWEntity;

namespace SignalRChat
{
    public partial class DowChat : System.Web.UI.Page
    {
        public string UserName = "";
        public string Name = "admin";
        public int userid = 0;
        public string UserImage = "/images/dummy.png";
        public string UName = "";
    }
}
```

```

protected string UploadFolderPath = "~/Uploads/";
protected void Page_Load(object sender, EventArgs e)
{
    if (Session["UserName"] != null)
    {
        UserName = Session["UserName"].ToString();
        GetUserImage(UserName);
        Name = Session["UserName"].ToString();
        userid = Convert.ToInt16(Session["userid"].ToString());

        EtetblUser user = new tblUsers().getuser(UserName);
        if (user.FirstName == "" && user.LastName == "")
        {
            UName = UserName;
        }
        else if (user.FirstName == user.LastName)
        {
            UName = user.FirstName;
        }
        else
        {
            UName = user.FirstName + " " + user.LastName;
        }

    }

    else
    {
        Response.Redirect("Home.aspx");
    }
    this.Header.DataBind();
}

protected void btnSignOut_Click(object sender, EventArgs e)
{
    //Session.Clear();
    //Session.Abandon();
    //Response.Redirect("Login.aspx");
    Response.Redirect("DashBoard.aspx");
}

public void GetUserImage(string Username)
{
    {
        userid = new tblUsers().getuserid(Username);
        List<EtetblUserImages> igs = new tblUsers().getuserImagesbyuserid(userid, "Profile");
    }
}

```

```

int cnt = igs.Count;
string path = "";
if (cnt > 0)
{
    path = igs[0].ImgPath;
}

if (path != "")
    UserImage = path;
}

}

protected void btnChangePicModel_Click(object sender, EventArgs e)
{
    string serverPath = HttpContext.Current.Server.MapPath("~/");
    string filepath = @"images\Profiles\Img-" + userid + "-1.jpg";
    if (FileUpload1.HasFile)
    {
        string FileWithPat = serverPath + filepath;
        //path = serverPath + path;

        FileUpload1.SaveAs(FileWithPat);
        SD.Image img = SD.Image.FromFile(FileWithPat);
        SD.Image img1 = RezielImage(img, 151, 150);
        img1.Save(FileWithPat);
        if (File.Exists(FileWithPat))
        {
            FileInfo fi = new FileInfo(FileWithPat);
            string ImageName = fi.Name;
            //string query = "update tbluserimagess set imgpath=" + filepath + " where userid=" +
userid + """;
            //if (ConnC.ExecuteQuery(query))
            //    UserImage = filepath;

        }
    }
}

#region Resize Image With Best Qaulity

private SD.Image RezielImage(SD.Image img, int maxWidth, int maxHeight)
{

```

```

if (img.Height < maxHeight && img.Width < maxWidth) return img;
using (img)
{
    Double xRatio = (double)img.Width / maxWidth;
    Double yRatio = (double)img.Height / maxHeight;
    Double ratio = Math.Max(xRatio, yRatio);
    int nnx = (int)Math.Floor(img.Width / ratio);
    int nny = (int)Math.Floor(img.Height / ratio);
    Bitmap cpy = new Bitmap(nnx, nny, SD.Imaging.PixelFormat.Format32bppArgb);
    using (Graphics gr = Graphics.FromImage(cpy))
    {
        gr.Clear(Color.Transparent);

        // This is said to give best quality when resizing images
        gr.InterpolationMode = InterpolationMode.HighQualityBicubic;

        gr.DrawImage(img,
            new Rectangle(0, 0, nnx, nny),
            new Rectangle(0, 0, img.Width, img.Height),
            GraphicsUnit.Pixel);
    }
    return cpy;
}

}

private MemoryStream ByteArrayToStream(byte[] arr)
{
    return new MemoryStream(arr, 0, arr.Length);
}

#endregion

protected void FileUploadComplete(object sender, EventArgs e)
{
    string filename = System.IO.Path.GetFileName(AsyncFileUpload1.FileName);
    AsyncFileUpload1.SaveAs(Server.MapPath(this.UploadFolderPath) + filename);
}

}

}
***** GenericErrorPage.aspx *****
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="GenericErrorPage.aspx.cs"
Inherits="GenericErrorPage" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

```

```

<html>
<head id="Head1" runat="server">
  <title>Generic Error Page</title>
</head>
<body>
  <form id="form1" runat="server">
  <div>
    <h2>
      Generic Error Page</h2>
    <asp:Panel ID="InnerErrorPanel" runat="server" Visible="false">
      <p>
        Inner Error Message:<br />
        <asp:Label ID="innerMessage" runat="server" Font-Bold="true"
          Font-Size="Large" /><br />
      </p>
      <pre>
        <asp:Label ID="innerTrace" runat="server" />
      </pre>
    </asp:Panel>
    <p>
      Error Message:<br />
      <asp:Label ID="exMessage" runat="server" Font-Bold="true"
        Font-Size="Large" />
    </p>
    <pre>
      <asp:Label ID="exTrace" runat="server" Visible="false" />
    </pre>
    <br />
    Return to the <a href='Default.aspx'> Default Page</a>
  </div>
  </form>
</body>
</html>

```

```

***** GenericErrorPage.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 GenericErrorPage : System.Web.UI.Page
{
    protected Exception ex = null;

    protected void Page_Load(object sender, EventArgs e)
    {

```

```

// Get the last error from the server
Exception ex = Server.GetLastError();

// Create a safe message
string safeMsg = "A problem has occurred in the web site. ";

// Show Inner Exception fields for local access
if (ex.InnerException != null)
{
    innerTrace.Text = ex.InnerException.StackTrace;
    InnerErrorPanel.Visible = Request.IsLocal;
    innerMessage.Text = ex.InnerException.Message;
}
// Show Trace for local access
if (Request.IsLocal)
    exTrace.Visible = true;
else
    ex = new Exception(safeMsg, ex);

// Fill the page fields
exMessage.Text = ex.Message;
exTrace.Text = ex.StackTrace;

// Log the exception and notify system operators
ExceptionUtility.LogException(ex, "Generic Error Page");
ExceptionUtility.NotifySystemOps(ex);

// Clear the error from the server
Server.ClearError();
}
}
***** 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 - Your Choice</title>
<link rel="icon" href="images/DateOrWed.png" type="image/gif"/>
<link rel="stylesheet" href="font/fontawesome6/pro/css/all.css">
<link href="css/bootstrap.min.css" rel="stylesheet">
<link href="css/mdb.min.css" rel="stylesheet">
<link href="css/style.css" rel="stylesheet">
<script type="text/javascript" src="js/jquery-3.4.1.min.js"></script>
<script type="text/javascript" src="js/popper.min.js"></script>

```

```

<script type="text/javascript" src="js/bootstrap.min.js"></script>
<script type="text/javascript" src="js/mdb.min.js"></script>
<link href="chatcss/style.css" rel="stylesheet">

</head>

<body>
  <form id="form1" runat="server">
    <!-- Start your project here-->
    <!--Navbar -->
<nav class=" navbar fixed-top navbar-expand-lg navbar-light bg-light pb-lg-4 pb-sm-5"> <!--pb-lg-1 pb-
sm-5--%>
  <div class="col-lg-2 col-md-2 col-sm-6 d-none d-lg-block d-md-block d-xl-none">
    <a class="navbar-brand" href="Default.aspx">
      

</a>

  </div>
    <div id="TopSearchBar" class="row clearfix " style="float: right; width: auto; margin-top: 5px;
margin-left: 3px;">
      <asp:TextBox ID="TextBox1" runat="server" Width="169px"
onkeydown="javascript:TextBox1_OnKeyDown(event)"></asp:TextBox>
      <asp:ImageButton ID="imgbtn" runat="server" ImageUrl="images/search3.jpg"
OnClick="imgbtn_Click" Height="30px" />

    </div>

<!--#080A52--#eb2188--%>

  <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 nav-flex-icons product-price"> <!--topbar-right--%>
      <a class="nav-link" href="DashBoard.aspx" id="dash" runat="server" >
        <!--<i class="fa fa-tachometer " > </i--%>
      

    <asp:Label ID="Label_wc" runat="server" Text="Welcome"></asp:Label>    </a>
    </li>
    <li class="nav-item" id="Li4" visible="false" runat="server">
      <a class="nav-link" href="UserList.aspx">

```



```

        <i class="fas fa-heart"></i> Staff-Users</a>
    </li>

    <li class="nav-item" id="Li3" visible="false" runat="server">
        <a class="nav-link" href="MyHeartList.aspx">
            <i class="fas fa-heart"></i> My HeartList</a>
        </li>
    <li class="nav-item" id="shortlist" visible="false" runat="server">
        <a class="nav-link" href="MyShortList.aspx">
            <i class="fas fa-heart"></i> My Shortlist</a>
        </li>
    <li class="nav-item" id="Li2" visible="false" runat="server">
        <a class="nav-link" href="SendProfile.aspx?psid=0">
            <i class="fas fa-paper-plane"></i>Sent Profiles</a>
        </li>
    <li class="nav-item" id="Li1" visible="false" runat="server">
        <a class="nav-link" href="RcvdProfile.aspx?psid=0">
            <i class="fas fa-file-text"></i>Received Profiles</a>
        </li>
    <li class="nav-item" id="profile" visible="false" runat="server" >
        <a class="nav-link" href="UserDetail.aspx">
            <i class="fas fa-user"></i> Profile</a>
        </li>
    <li id="reg" class="nav-item" runat="server">
        <a class="nav-link" href="Register.aspx">
            <i class="fas fa-registered"></i> Register</a>
        </li>
    <li class="nav-item">
        <a class="nav-link" href="Login.aspx">
            <i class="fa fa-user-alt-slash"></i> <asp:Label ID="Label_login" runat="server"
Text="Login"></asp:Label> </a>

        </li>
    </ul>

</div>

</nav>
<!--/.Navbar -->

<!--Carousel Wrapper-->
<div id="carousel-example-2" class="carousel slide carousel-fade pt-5 pt-sm-5 " data-ride="carousel">

    <!--Slides-->
    <div class="carousel-inner pt-sm-5" 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>
  </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>
  </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>

<!--/.Carouse</section>| Wrapper-->

<!-- Search -->

<section class="search-sec ml-auto col-lg-12 col-md-12 d-none d-lg-block d-md-block d-xl-none "
style="display:block" id="corsearch" runat="server" >
  <div class="container">
    <div class="row">
      <div class="col-lg-12">
        <div class="row">
          <div class="col-lg-12">
            <div class="row">

<div class="col-lg-2 col-md-2 col-sm-2 col-xs-auto p-0">
  <asp:DropDownList ID="ddl_location" runat="server" Height="50px">

  <asp:ListItem Value="0">--- Select an Option ---</asp:ListItem>
  <asp:ListItem Value="4">Dating</asp:ListItem>
  <asp:ListItem Value="5">Wedding</asp:ListItem>
  <asp:ListItem Value="6">Friends List</asp:ListItem>
  <asp:ListItem Value="7">Date or Wed</asp:ListItem>
  <asp:ListItem Value="8">Date and Wed</asp:ListItem>
</asp:DropDownList>
</div>
    <div class="col-lg-8 col-md-8 col-sm-6 col-xs-auto 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-2 col-xs-auto 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>--%>
      <asp:ImageButton ID="ImageButton1" runat="server" ImageUrl="images/searchnew.png"
OnClick="ImageButton1_Click" Height="50px" />

      <%--<button id="Button1" type="button" class="btn btn-secondary wrn-btn mt-0 ml-
0" onclick=""><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 = 'SearchList.aspx?pageid=198' + '&SrchStrng=' + encodeURIComponent(SearchString) +
'&cat=' + encodeURIComponent(loc);
var url = encodeURI('SearchList.aspx?pageid=198&SrchStrng=' + SearchString + '&cat=' + loc);
//$(location).attr('href', url);
window.location.href = url;
event.preventDefault();
}
</script>
</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">Dating Profiles</h4>
<p class="card-text">
<a class="badge badge-primary"
href="SearchList.aspx?cat=4&SrchStrng=&fromage=21&toage=25">Ages 21-25</a>
<a class="badge badge-primary"
href="SearchList.aspx?cat=4&SrchStrng=&fromage=26&toage=30">Ages 26-30</a>
<a class="badge badge-primary" href="SearchList.aspx?cat=4&SrchStrng=&psort=R">Religion
wise</a>
<a class="badge badge-primary" href="SearchList.aspx?cat=4&SrchStrng=&psort=c">Caste
wise</a>
<a class="badge badge-primary" href="SearchList.aspx?cat=4&SrchStrng=&psort=A">Location
wise</a>
<a class="badge badge-primary" href="SearchList.aspx?cat=4&SrchStrng=&psort=T">City
wise</a>
<a class="badge badge-primary" href="SearchList.aspx?cat=4&SrchStrng=&psort=G">Gender
wise</a>
<a class="badge badge-primary" href="SearchList.aspx?cat=4&SrchStrng=&psort=Y">Country
wise</a>

```

```
        <a class="badge badge-primary"
href="SearchList.aspx?cat=4&SrchStrng=&psort=M">Mothertongue wise</a>
        <a class="badge badge-primary" href="SearchList.aspx?cat=4&SrchStrng=&psort=S">Language
wise</a>
    </p>
    <a class="btn btn-secondary btn-md btn-rounded" href="SearchList.aspx?cat=4&SrchStrng="
>View Now</a>
</div>
</div>
</div>
```

```
<div class="col-md-4">
<div class="card mb-2">
    
    <div id="services" class="card-body text-center" runat="server">
    <h4 class="card-title font-weight-bold">Wedding Profiles</h4>
    <p class="card-text">
<%--    <span class="badge badge-primary">Ages 21-25</span>
        <span class="badge badge-primary">Ages 26-30</span>--%>
        <a class="badge badge-primary"
href="SearchList.aspx?cat=5&SrchStrng=&fromage=21&toage=25">Ages 21-25</a>
        <a class="badge badge-primary"
href="SearchList.aspx?cat=5&SrchStrng=&fromage=26&toage=30">Ages 26-30</a>
        <a class="badge badge-primary" href="SearchList.aspx?cat=5&SrchStrng=&psort=R">Religion
wise</a>
        <a class="badge badge-primary" href="SearchList.aspx?cat=5&SrchStrng=&psort=C">Caste
wise</a>
        <a class="badge badge-primary" href="SearchList.aspx?cat=5&SrchStrng=&psort=L">Location
wise</a>
        <a class="badge badge-primary" href="SearchList.aspx?cat=5&SrchStrng=&psort=T">City
wise</a>
        <a class="badge badge-primary" href="SearchList.aspx?cat=5&SrchStrng=&psort=G">Gender
wise</a>
        <a class="badge badge-primary" href="SearchList.aspx?cat=5&SrchStrng=&psort=Y">Country
wise</a>
        <a class="badge badge-primary"
href="SearchList.aspx?cat=5&SrchStrng=&psort=M">Mothertongue wise</a>
        <a class="badge badge-primary" href="SearchList.aspx?cat=5&SrchStrng=&psort=S">Language
wise</a>
    </p>
    <a class="btn btn-secondary btn-md btn-rounded"
href="SearchList.aspx?cat=5&SrchStrng=">View Now</a>
    </div>
</div>
</div>
```

```
<div class="col-md-4">
```

```
<div class="card mb-2">
  
  <div class="card-body text-center">
    <h4 class="card-title font-weight-bold">Friends Profiles</h4>
    <p class="card-text">
<!--      <span class="badge badge-primary">Ages 21-25</span>
      <span class="badge badge-primary">Ages 26-30</span>--%>
      <a class="badge badge-primary"
href="SearchList.aspx?cat=6&SrchStrng=&fromage=21&toage=25">Ages 21-25</a>
      <a class="badge badge-primary"
href="SearchList.aspx?cat=6&SrchStrng=&fromage=26&toage=30">Ages 26-30</a>
      <a class="badge badge-primary" href="SearchList.aspx?cat=6&SrchStrng=&psort=H">School
wise</a>
      <a class="badge badge-primary" href="SearchList.aspx?cat=6&SrchStrng=&psort=D">College
wise</a>
      <a class="badge badge-primary" href="SearchList.aspx?cat=6&SrchStrng=&psort=L">Location
wise</a>
      <a class="badge badge-primary" href="SearchList.aspx?cat=6&SrchStrng=&psort=T">City
wise</a>
      <a class="badge badge-primary" href="SearchList.aspx?cat=6&SrchStrng=&psort=G">Gender
wise</a>
      <a class="badge badge-primary" href="SearchList.aspx?cat=6&SrchStrng=&psort=Y">Country
wise</a>
      <a class="badge badge-primary"
href="SearchList.aspx?cat=6&SrchStrng=&psort=M">Mothertongue wise</a>
      <a class="badge badge-primary" href="SearchList.aspx?cat=6&SrchStrng=&psort=S">Language
wise</a>
    </p>
    <a class="btn btn-secondary btn-md btn-rounded"
href="SearchList.aspx?cat=6&SrchStrng=">View Now</a>
  </div>
</div>
</div>
</div>
</div>
</div>
<!-- ./Services -->

<!-- Footer -->
<footer class="page-footer font-small unique-color-dark">

<!-- Copyright -->
<div class="footer-copyright text-center py-3">© 2020 Copyright
  <a target="_blank" href="Home.aspx"> DateOrWed.com</a> All Rights Reserved
</div>
<!-- Copyright -->

</footer>
```

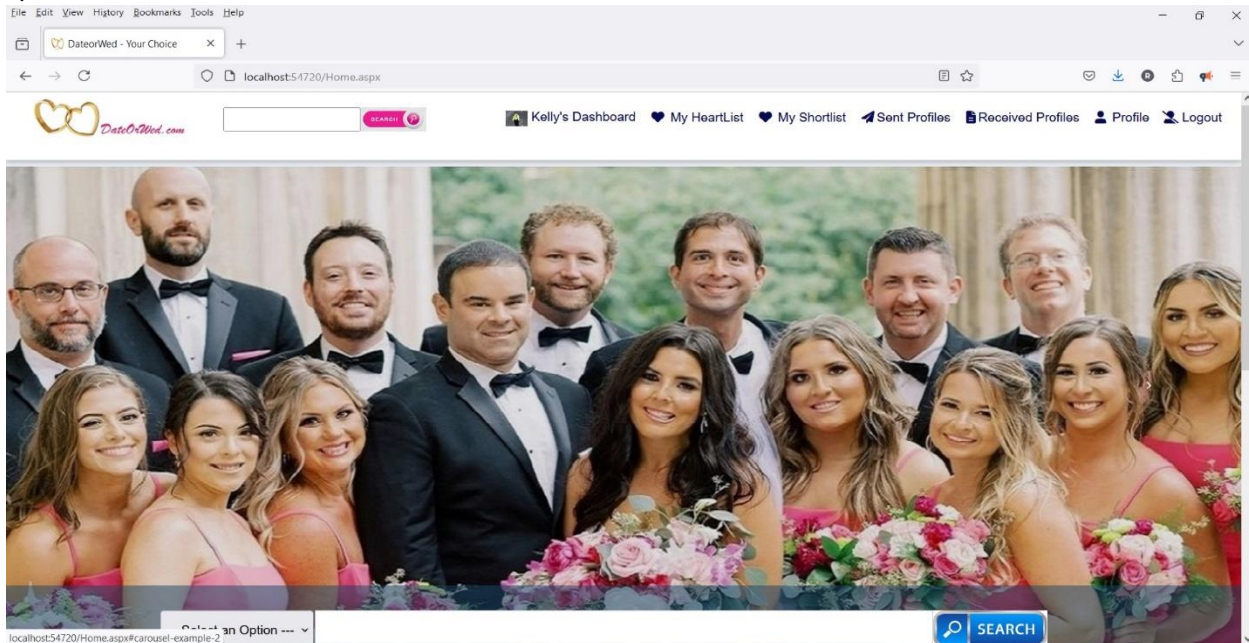
```
<!-- Footer -->
```

```
<!-- END your project here-->
```

```
</form>
```

```
</body>
```

```
</html>
```



```
***** Home.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.Configuration;
using DOWBal;
using DOWEntity;
using System.Data;
using System.Web.Security;
using System.Net;
using System.Text;
using System.Web.Script.Serialization;

public partial class Home : System.Web.UI.Page
{
    DDLogTracer objlog = new DDLogTracer();
    bool IsMobile = false;
```

```

protected void Page_Load(object sender, EventArgs e)
{
    if (Request.Browser.IsMobileDevice) { IsMobile = true; } else { IsMobile = false; }
    string IPAddress = GetIPAddress();
    Session["ipaddress"] = IPAddress;
    StringBuilder sb = new StringBuilder();
    Session["Page"] = "1";
    string user;
    user = Common.Encrypt("Guest");
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    if (Clsusr == null)
    {
        HttpCookie myCookie = Request.Cookies["DOWCookie"];
        if (myCookie != null)
        {
            Session["UserName"] = myCookie.Value.ToString();
        }
        else
        {
            Session["UserName"] = Common.Encrypt("Guest");
        }
        //user = "899999999";
        user = Session["UserName"].ToString();
        if (user != null)
        {
            //user = Common.Encrypt(user);
            string user1 = Common.Decrypt(user);
        }
        else
        {
            //
            HttpCookie myCookie = Request.Cookies["DOWCookie"];
            if (myCookie != null)
            {
                user = myCookie.Value.ToString();
                string user1 = Common.Decrypt(user);
            }
            else
            {
                user = Common.Encrypt("Guest");
                profile.Visible = false;
            }
        }
        guest_login(user);
        Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    }
    if (IsMobile)

```



```

{
    corsearch.Visible = false;
}
if (Clsusr != null)
{
    Session["UserName"] = Common.Decrypt(Clsusr.UserName);
    Session["userid"] = Clsusr.UserId;
    List<EtetblUserImages> igs = new
tblUsers().getuserImagesbyuserid(Convert.ToInt16(Clsusr.UserId), "Profile");
    int cnt = igs.Count;
    if (cnt > 0)
    {

        userimage.Source = igs[0].ImagePath;
    }

    if (Clsusr.UserName != "Guest")
    {
        Label_login.Text = "Logout";
        reg.Visible = false;
        profile.Visible = true;
        if (IsMobile)
        {
            //img1.Source = "images/cor3.jpg";
            //img2.Source = "images/cor7.jpg";
            //img3.Source = "images/cor1.jpg";
            //img4.Source = "images/cor8.jpg";
            //img5.Source = "images/cor5.jpg";
            //int dftcnt = 0; // new Products().GetSLItems(Convert.ToInt16(Clsusr.UserId), true);
            //sb.Append("<ul class='navbar-nav ml-auto list-inline topbar-right pull-right'>\n");
            //sb.Append("<li><a href='UserDetail.aspx'><i class='fas fa-users-cog'></i> Welcome " +
Clsusr.FirstName + "</a></li>\n");
            //sb.Append("<li><a href='MyShortList.aspx'><i class='fas fa-hand-holding-
heart'></i>Shortlist (" + Convert.ToString(dftcnt) + ")</a>\n");
            //sb.Append(" | <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
{
    Label_login.Text = "Logout";
    reg.Visible = false;
    if (Convert.ToInt16(Clsusr.GroupId) > 3)
    {
        profile.Visible = true;
        Li3.Visible = true;
        shortlist.Visible = true;
        Li1.Visible = true;
    }
}

```

```

        Li2.Visible = true;
    }
    else
    {
        if (Convert.ToInt16(Clsusr.GroupId) == 3 || Convert.ToInt16(Clsusr.GroupId)==1)
        {
            Li4.Visible = true;

            }
            shortlist.Visible = false;
            profile.Visible =false;
            Li3.Visible = false;
            Li1.Visible = false;
            Li2.Visible = false;
        }
        //account.Visible = false;
    }
}
if (Clsusr.UserName == "Guest") { dash.HRef = ""; }
else if (Clsusr.FirstName == "" | Clsusr.FirstName == null)
{
    //string gname = new tblUsers().getgroupname(Convert.ToInt16(Clsusr.GroupId));

    Label_wc.Text = "Your Dashboard"; //"Welcome " +

}
else { Label_wc.Text = Clsusr.FirstName + "'s Dashboard"; } //"Welcome " +
}

    if (IsMobile)
    {

        //shorts.Visible = false;
    }

}
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.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"]);
    objLoggedInUser.FirstName = status_login.Rows[0]["FirstName"].ToString();
    objLoggedInUser.LastName = status_login.Rows[0]["LastName"].ToString();
    Session["LoggedUser_Specs"] = objLoggedInUser;

    FormsAuthentication.SetAuthCookie(objLoggedInUser.UserName, false);
    HttpCookie myCookie = new HttpCookie("DOWCookie");
    DateTime now = DateTime.Now;
    myCookie.Value = objLoggedInUser.UserName;
    myCookie.Expires = now.AddYears(1);

    Response.Cookies.Add(myCookie);

}
}

public string GetIPAddress()
{
    String address = "";
    try
    {
        WebRequest request = WebRequest.Create("http://checkip.dyndns.org/");
        using (WebResponse response = request.GetResponse())
        using (StreamReader stream = new StreamReader(response.GetResponseStream()))
        {
            address = stream.ReadToEnd();
        }

        int first = address.IndexOf("Address: ") + 9;
        int last = address.LastIndexOf("</body>");
        address = address.Substring(first, last - first);
    }
    catch { }
    return address;
}

```

```

}
protected void Button1_Click(object sender, EventArgs e)
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"]; ;

    string cat = Common.Encrypt(ddl_location.SelectedValue.ToString());
    string srch = Common.Encrypt(txt_SearchString.Text);
    Response.Redirect("~/SearchList.aspx?pageid=198&SrchStrng=" + srch + "&cat="+cat) ;

}
protected void imgbtn_Click(object sender, ImageClickEventArgs e)
{
    string srctext = TextBox1.Text;
    Response.Redirect("SearchList.aspx?SrchStrng=" +Common.Encrypt( srctext)+"&cat=0");
}
protected void ImageButton1_Click(object sender, ImageClickEventArgs e)
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"]; ;

    string cat = ddl_location.SelectedValue.ToString();//
Common.Encrypt(ddl_location.SelectedValue.ToString());
    string srch = Common.Encrypt(txt_SearchString.Text);
    Response.Redirect("~/SearchList.aspx?pageid=198&SrchStrng=" + srch + "&cat=" + cat);

}
}

```

***** Http404ErrorPage.aspx *****

```

<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Http404ErrorPage.aspx.cs"
Inherits="Http404ErrorPage" %>

```

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"

```

```

"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

```

```

<html>

```

```

<head id="Head1" runat="server">

```

```

<title>HTTP 404 Error Page</title>

```

```

</head>

```

```

<body>

```

```

<form id="form1" runat="server">

```

```

<div>

```

```

<h2>

```

```

    HTTP 404 Error Page</h2>

```

```

    Standard error message suitable for file not found errors.

```

```

    The original exception object is not available,

```

```

    but the original requested URL is in the query string.<br />

```

```

<br />

```

```

    Return to the <a href='Default.aspx'> Default Page</a>
</div>
</form>
</body>
</html>
***** Http404ErrorPage.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 Http404ErrorPage : System.Web.UI.Page
{
    protected HttpException ex = null;

    protected void Page_Load(object sender, EventArgs e)
    {
        // Log the exception and notify system operators
        ex = new HttpException("HTTP 404");
        ExceptionUtility.LogException(ex, "Caught in Http404ErrorPage");
        ExceptionUtility.NotifySystemOps(ex);
    }
}
***** HttpErrorPage.aspx *****
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="HttpErrorPage.aspx.cs"
Inherits="HttpErrorPage" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html>
<head id="Head1" runat="server">
    <title>Http Error Page</title>
</head>
<body>
    <form id="form1" runat="server">
    <div>
        <h2>
            Http Error Page</h2>
        <asp:Panel ID="InnerErrorPanel" runat="server" Visible="false">
            <asp:Label ID="innerMessage" runat="server" Font-Bold="true"
                Font-Size="Large" /><br />
            <pre>
                <asp:Label ID="innerTrace" runat="server" />
            </pre>
        </asp:Panel>
        Error Message:<br />

```

```

<asp:Label ID="exMessage" runat="server" Font-Bold="true"
  Font-Size="Large" />
<pre>
  <asp:Label ID="exTrace" runat="server" Visible="false" />
</pre>
<br />
  Return to the <a href='Home.aspx'>Default Page</a>
</div>
</form>
</body>
</html>

```

***** HttpErrorPage.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;

public partial class HttpErrorPage : System.Web.UI.Page
{
    protected HttpException ex = null;

    protected void Page_Load(object sender, EventArgs e)
    {
        ex = (HttpException)Server.GetLastError();
        int httpCode = ex.GetHttpCode();

        // Filter for Error Codes and set text
        if (httpCode >= 400 && httpCode < 500)
            ex = new HttpException
                (httpCode, "Safe message for 4xx HTTP codes.", ex);
        else if (httpCode > 499)
            ex = new HttpException
                (ex.ErrorCode, "Safe message for 5xx HTTP codes.", ex);
        else
            ex = new HttpException
                (httpCode, "Safe message for unexpected HTTP codes.", ex);

        // Log the exception and notify system operators
        ExceptionUtility.LogException(ex, "HttpErrorPage");
        ExceptionUtility.NotifySystemOps(ex);

        // Fill the page fields
        exMessage.Text = ex.Message;
        exTrace.Text = ex.StackTrace;
    }
}

```

```

// Show Inner Exception fields for local access
if (ex.InnerException != null)
{
    innerTrace.Text = ex.InnerException.StackTrace;
    InnerErrorPanel.Visible = Request.IsLocal;
    innerMessage.Text = string.Format("HTTP {0}: {1}",
        httpCode, ex.InnerException.Message);
}
// Show Trace for local access
exTrace.Visible = Request.IsLocal;
string strErrorFileName = "~\\log\\LogFile.txt" ;
StreamWriter sw = null;

if ((!File.Exists(strErrorFileName)))
{
    sw = File.CreateText(strErrorFileName);
}
else
{
    sw = File.AppendText(strErrorFileName);
}
sw.WriteLine(ex.Message+"-"+ex.StackTrace);
sw.Close();

// Clear the error from the server
Server.ClearError();
}
}
***** Login.aspx *****
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Login.aspx.cs" Inherits="Login" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head>
    <title>DateOrWed - Your Choice</title>
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <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"/>

</head>

<body>
  <form id="form1" class="box login" runat="server">
    <div>
      <div>

        <div class="container" style="text-align:center; padding:20px 0px 20px 0px;">

          <!-- -->
          
          <h2 class="btn btn-rounded align-content-center" style="color: orangered;align-
content:flex-start">Login</h2>
          <div style="text-align: left;width:30em;margin: 2em auto 3em auto;">

            <fieldset class="boxBody border-dark">
              <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"></asp:TextBox>
                    <!-- <i class='fa fa-user'> --%><asp:Button ID="Button2" runat="server" CssClass="btn btn-
                    secondary btn-md btn-rounded pull-right" Text=" Login " OnClick="Button2_Click"/>

```



```

<%--      <asp:ImageButton ID="ImageButton2" CssClass="pull-left" runat="server"
style="margin-top:20px;" ImageUrl="~/images/register.jpg" OnClick="ImageButton2_Click" />

      <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>
  <tr> <td>
    <asp:Button ID="Button1" runat="server" CssClass="btn btn-primary btn-md btn-rounded
pull-left" Text="Register" OnClick="Button1_Click" />

    </td>
  <%-- <td>
<asp:Button ID="btnLoginToGoogle" Runat="server" Text="Google Login"
OnCommand="OpenLogin_Click" CommandArgument="https://www.google.com/accounts/o8/id"
Font-Bold="True" Font-Italic="True" Width="162px" />
    </td>--%>
  </tr>
</tbody></table>

  <div class="validation-summary-valid" id="validationSummary"><ul><li
style="display:none"></li>
</ul></div>
  </fieldset>

  <script type="text/javascript">

    //
    if (!window.mvcClientValidationMetadata) { window.mvcClientValidationMetadata = []; }
    window.mvcClientValidationMetadata.push({ "Fields": [], "FormId": "form0",
"ReplaceValidationSummary": true, "ValidationSummaryId": "validationSummary" });
    //]]&gt;
  &lt;/script&gt;

  &lt;/div&gt;
&lt;/div&gt;

  &lt;script type="text/javascript"&gt;
    document.getElementById("userName").focus();
  &lt;/script&gt;
</pre>
</div>
```

</div>

</div>

</form>

<div id="footer" style="text-align:center; position:relative; top:750px; margin-left:15px;">
<!-- Dow Version 1.39.10. Heartbeat. Copyright © 2018
Dateorwed.com.--%>
</div>

</body> <!-- #030548#eb2188--%>

</html>



DateOrWed.com

Login

User Name:

9966765997

Password:

.....

Login

Register

```
***** 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 DOWEntity;  
using System.Web.Security;  
using System.Net;  
using System.IO;  
using System.Web.Script.Serialization;  
using DotNetOpenAuth.OpenId;
```

```

using DotNetOpenAuth.OpenId.RelyingParty;

public partial class Login : System.Web.UI.Page
{
    readonly DDLogTracer objlog = new DDLogTracer();

    public static string _Country;
    public static string _City;
    public static string ip;
    protected void Page_Load(object sender, EventArgs e)
    {
        //List<EtetblUser> users = new tblUsers().getusersbak("");
        //foreach (EtetblUser usr in users)
        //{
        //    usr.UserName = (usr.UserName);
        //    string username = Common.Decrypt(Common.Encrypt(usr.UserName));
        //    usr.Password = (usr.Password);
        //    bool ret = new tblUsers().Update(usr);
        //}
        //List<Etetbluser_profile> users = new tblUsers().getuserprofiles();
        //foreach (Etetbluser_profile usr in users)
        //{
        //    bool ret = new tblUsers().InsertProfile(usr);
        //}
        try
        {
            ClsLoggedInUser Clsusr = (ClsLoggedInUser)HttpContext.Current.Session["LoggedInUser_Specs"];
            BALCreateUser objbal = new BALCreateUser();
            string ipadd = GetIp();
            GetLocation(ipadd);
            if (Clsusr != null)
            {
                objbal.LogOut(Convert.ToInt32(Clsusr.UserId), Convert.ToInt32(Clsusr.LoginID), _Country,
                _Country, ipadd);
            }
        }
        catch (Exception exc)
        {
            objlog.Trace_Error(4, "DateOrWed", "Profile", "Load", "Exception:" + exc.ToString());
        }
    }
    try
    {
        HttpContext.Current.Session.Abandon();
        Session.Clear();
        Session.RemoveAll();
        Session.Abandon();
        HttpCookie myCookie = new HttpCookie("DOWCookie");
        DateTime now = DateTime.Now;
    }
}

```

```

myCookie.Value = "";
myCookie.Expires = now.AddYears(-1);
Response.Cookies.Clear();
Response.Cookies.Add(myCookie);
Request.Cookies.Clear();
Session["PageType"] = "";

}
catch (Exception exc)
{
    objlog.Trace_Error(4, "DateOrWed", "Profile", "Load", "Exception:" + exc.ToString());

}
OpenIdRelyingParty OIDRP = new OpenIdRelyingParty();
var str = OIDRP.GetResponse();
if (str != null)
{
    switch (str.Status)
    {
        case AuthenticationStatus.Authenticated:
            lblStatus.Visible = false;
            Session["GoogleIdentifier"] = str.ClaimedIdentifier.ToString();
            Response.Redirect("Home.aspx");
            break;
        case AuthenticationStatus.Canceled:
            lblStatus.Text = "Cancelled.";
            lblStatus.Visible = true;
            break;
        case AuthenticationStatus.Failed:
            lblStatus.Text = "Login Failed.";
            lblStatus.Visible = true;
            break;
    }
}
}
protected void Button2_Click(object sender, EventArgs e)
{
    string password = "";
    //try
    {
        string ip = GetIp();
        if (ip != null)
        {
            GetLocation(ip);
        }
        BALCreateUser objlogin = new BALCreateUser();

        password = Common.Encrypt(txtpassword.Text.Trim());
    }
}

```

```

string username = Common.Encrypt(txtuname.Text.Trim());
DataTable status_login = objlogin.Login(Common.Encrypt(txtuname.Text.Trim()), password, ip, "",
_City, _Country);
string login_status = status_login.Rows[0]["UserStatus"].ToString();

ClsLoggedInUser objLoggedInUser;
// if (Convert.ToInt16( status_login.Rows[0]["UserId"].ToString() )>1)
if (Convert.ToBoolean(status_login.Rows[0]["UserStatus"]))
{
    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"].ToString());
    objLoggedInUser.DealerId = Convert.ToInt32(status_login.Rows[0]["DealerId"].ToString());
    objLoggedInUser.RoleId = Convert.ToInt32(status_login.Rows[0]["RoleId"].ToString());
    objLoggedInUser.FirstName = status_login.Rows[0]["FirstName"].ToString();
    objLoggedInUser.LastName = status_login.Rows[0]["LastName"].ToString();
    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 = objLoggedInUser.UserName;
    myCookie.Expires = now.AddYears(1);
    Session["UserName"] =Common.Decrypt( objLoggedInUser.UserName);
    username = Session["UserName"].ToString();
    Response.Cookies.Add(myCookie);
    Response.Redirect("~/Home.aspx");

    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"]));
        string page = new tblUsers().getuserpage(objLoggedInUser.UserName);
        Response.Redirect("~/Home.aspx");
    }
}

```

```

        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();
}
}
//catch (Exception exc)
//{
//    objlog.Trace_Error(4, "DateOrWed", "Profile", "Load", "Exception:" + exc.ToString());
//}

}
public List<Location> GetLocation(string ipAddress)
{
    List<Location> locations = new List<Location>();
    //string ipAddress = Request.ServerVariables["HTTP_X_FORWARDED_FOR"];
    //if (string.IsNullOrEmpty(ipAddress))
    //{
    //    ipAddress = Request.ServerVariables["REMOTE_ADDR"];
    //}

    string APIKey = "744627112f51a720c93d8e11741a02c6f50ad135ea28e733d10f52dc0596b193";
    string url = string.Format("http://api.ipinfodb.com/v3/ip-city/?key={0}&ip={1}&format=json",
APIKey, ipAddress);
    using (WebClient client = new WebClient())
    {
        string json = client.DownloadString(url);
        Location location = new JavaScriptSerializer().Deserialize<Location>(json);
        locations.Add(location);
        _City = locations[0].CityName;
        _Country = locations[0].CountryName;
    }
}

```

```

    }
    return locations;
}

[System.Web.Services.WebMethod]
public static string GetCountrynCity(string City, string Country)
{
    _City = City;
    _Country = Country;
    return "";
}
protected void OpenLogin_Click(object src, CommandEventArgs e)
{
    string str = e.CommandArgument.ToString();
    OpenIdRelyingParty openid = new OpenIdRelyingParty();
    var b = new UriBuilder(Request.Url) { Query = "" };
    var req = openid.CreateRequest(str, b.Uri, b.Uri);
    req.RedirectToProvider();
}
//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"], "", 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.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 = Convert.ToInt32(status_login.Rows[0]["BrandId"]);

//        Session["LoggedUser_Specs"] = objLogedInUser;

//        FormsAuthentication.SetAuthCookie(objLogedInUser.UserName, false);
//        ClsPermissioGroup clsperGrp = new ClsPermissioGroup();
//        UserPermissions Entuserper = clsperGrp.GetUserLandingPage(objLogedInUser.UserName);

```



```

// if (Entuserper.Pageld != 0)
// {
//     BalRecentPlacess objRemoveRcntPlc = new BalRecentPlacess();
//     bool rst =
objRemoveRcntPlc.DeleteRecentPlaces(Convert.ToInt32(status_login.Rows[0]["UserId"]));
//     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 = "London";// _City;
// string Country = "U.K.");// _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"], "", 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("DOWCookie");
//  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"]));
//      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("~/Register.aspx");
}
protected string GetIp()
{
    String address = "";
    try {
        WebRequest request = WebRequest.Create("http://checkip.dyndns.org/");
        using (WebResponse response = request.GetResponse())
        using (StreamReader stream = new StreamReader(response.GetResponseStream()))
        {
            address = stream.ReadToEnd();
        }

        int first = address.IndexOf("Address: ") + 9;
        int last = address.LastIndexOf("</body>");
        address = address.Substring(first, last - first);
    }
    catch (Exception exc)
    {
        objlog.Trace_Error(4, "DateOrWed", "Profile", "Load", "Exception:" + exc.ToString());
    }
    return address;
}

}
***** LoginInfo.aspx *****
<%@ Page Title="DateorWed - Your Profile" Language="C#"
MasterPageFile="~/DashBoardMasterPage.master" AutoEventWireup="true"
CodeFile="LoginInfo.aspx.cs" Inherits="LoginInfo" %>

<%@ 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">

```

```
<%-- <link rel="stylesheet" href="css/style1.css"/>--%>
<%-- <link rel="stylesheet" href="css/style2.css"/>--%>
```

```
</asp:Content>
<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
  <div class="container">
    <div class="row-fluid" style="padding-top: 1px; padding-left:50px">
      <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="Content5" ContentPlaceHolderID="ContentMainPage" Runat="Server">
  <!-- Content Bodey Here -->

  <section class=" row dip-style"><%--main-contain --%>
    <div class="container bg-gray ">
      <div class="row "><%--login-area--%>

      <div class="tab-content active ml-5">
        <!-- Description -->
        <div class="row-fluid">
          <div class="col-lg-6 col-md-6 col-sm-12" style="text-align: left;">
            <h3>LOGIN INFORMATION</h3></div>
            <asp:Button ID="btn3" runat="server" CssClass="btn btn-primary" Text="Save
Login Info" OnClick="btn3_Click"/>
          </div>
          <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-12 col-md-12 col-sm-12 detailview" style="background-color:
#ffffff; padding: 15px 5px;"> <%--border-top: none;border: 0px solid #eefef;--%>

              <!--<form class="form-horizontal">-->
              <div class="control-group">
                <table class="table table-striped col-lg-12 col-md-12 col-sm-12 bg-secondary" >
                  <tbody>
```

```

        <tr>
            <td class=" col-lg-4 col-md-4 col-sm-2"><strong>Email/Mobile/User Id
:*</strong></td>
            <td class="auto-style1 col-lg-8 col-md-8 col-sm-8">
                <asp:TextBox ID="txt_username" runat="server" MaxLength="50"
TextMode="Email" ></asp:TextBox>
            </td>
        </tr>
        <tr>
            <td ><strong>First Name :*</strong></td>
            <td >
                <asp:TextBox ID="txt_FirstName" runat="server" MaxLength="100"
onkeypress="return validatealpha(event)" CssClass="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 align-right"><strong>Last Name
:*</strong></td>
            <td class="align-left">
                <asp:TextBox ID="txt_LastName" runat="server" MaxLength="40"
onkeypress="return validatealpha(event)" CssClass="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">Your Gender :*</td>
            <td>
                <asp:DropDownList ID="ddl_gender" runat="server" CssClass="show">
<asp:ListItem Value="0">-- Select Your Gender --</asp:ListItem>
<asp:ListItem Value="1">Male</asp:ListItem>
<asp:ListItem Value="2">Women</asp:ListItem>
<asp:ListItem Value="3">Trans</asp:ListItem>
<asp:ListItem Value="4">Other</asp:ListItem>
                </asp:DropDownList>
            </td>
        </tr>
    </tr>

```

```

                <td align="right" class=" col-lg-3 col-md-3 col-sm-3 align-
right"><strong>Country:</strong></td>
                <td align="left" class="align-right">
                <asp:DropDownList ID="ddl_country" runat="server"
CssClass="show"></asp:DropDownList>

                </td>

            </tr>
            <tr>
                <td align="right" class=" col-lg-3 col-md-3 col-sm-3"><strong>Email id
:*</strong></td>
                <td align="left">
                <asp:TextBox ID="email_name" runat="server" MaxLength="50" TextMode="Email"
Visible="true" Enabled="true"></asp:TextBox>
                </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>Choice
:*</strong></td>
                <td width="25%" align="left">
                <asp:DropDownList ID="ddl_GroupId" runat="server"
Enabled="true" CssClass="show"></asp:DropDownList>
                <asp:SqlDataSource ID="SqlDataSource2" runat="server" ConnectionString="<%"$
ConnectionStrings:DOWConectingString %>" SelectCommand="SELECT GroupId,groupname FROM
[tblPermissionGroup]"></asp:SqlDataSource>

                </td>
            </tr>
        </tr>

```

```

        <tr>
            <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Monther Tongue :</td>
            <td>
                <asp:DropDownList ID="ddl_mtongue" runat="server"
Enabled="true" CssClass="show"></asp:DropDownList>
                <asp:SqlDataSource ID="SqlDataSource3" runat="server" ConnectionString="<%$
ConnectionStrings:DOWConectingString %>" SelectCommand="SELECT lan_id,language FROM
[Languages]"></asp:SqlDataSource>
            </td>
        </tr>
        <tr>
            <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Universal Number :</td>
            <td>
                <asp:TextBox ID="txt_adhaar_id" runat="server" MaxLength="20"
></asp:TextBox>
            </td><td hidden="hidden">
                <asp:CheckBox ID="adhaar_id_chk"
runat="server" Checked="true" />
            </td>
        </tr>
        <tr>
            <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Pan Number :</td>
            <td>
                <asp:TextBox ID="txt_pan_no" runat="server" MaxLength="10"
></asp:TextBox>
            </td><td hidden="hidden">
                <asp:CheckBox ID="pan_no_chk"
runat="server" Checked="true"/>
            </td>
        </tr>
        <tr hidden="hidden">
            <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>
            </td>
        </tr>
        <tr><td class="pull-right">
            <div class="form-check" id="d_currency_chk" runat="server">
                <input
class="form-check-input"
type="checkbox"
value="" checked=""
id="currency_chk" runat="server"
/>

```

```

<label class="form-check-label" for="currency_chk">

</label>
</div>

<!-- <input id="currency_chk" type="checkbox" class="show" value="" checked />--%>
<!-- <asp:CheckBox ID="currency_chk" runat="server" Text=" Click to Authorise" />--%>
  </td>
  <td ><i class="fa fa-check"></i> I authorise to use Date of Birth, Date of Time and Place of birth to be
used in Matching Matches using various methods like Numerology, Astrology, etc. Send an email for
your requests</td>

</tr>
<tr><td class="pull-right">
  <div class="form-check" id="d_ad_chk" runat="server">
  <input class="form-check-input" type="checkbox" runat="server" value="" id="ad_chk"
checked="" />
  <label class="form-check-label" for="ad_chk">

</label>
</div>

<!-- <input id="ad_chk" type="checkbox" class="show fa fa-check-circle" value="" checked />--%>
<!-- <asp:CheckBox ID="ad_chk" runat="server" Text=" Click to Allow" />--%>
  </td>
  <td><i class="fa fa-check"></i> I authorise to use Public / authorised Photos and Information to use
for Advertisement and to Post on the site.(Please send an email to confirm the same)</td>

</tr>
      <tr hidden="hidden">
          <td width="17%" 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>
              <!-- <asp:RegularExpressionValidator ID="RegularExpressionValidator1" runat="server"
ErrorMessage="RegularExpressionValidator"
ValidationExpression="^\.(?=.*{8,})(?=.*\d)(?=.*[a-z])(?=.*[A-Z])(?=.*[!@#\$%^&+=]).*$"
ControlToValidate="txt_Password">
              </asp:RegularExpressionValidator>--%>
          </td>
      </tr>
  </tbody>
</table>

</div>

</div>

```



```

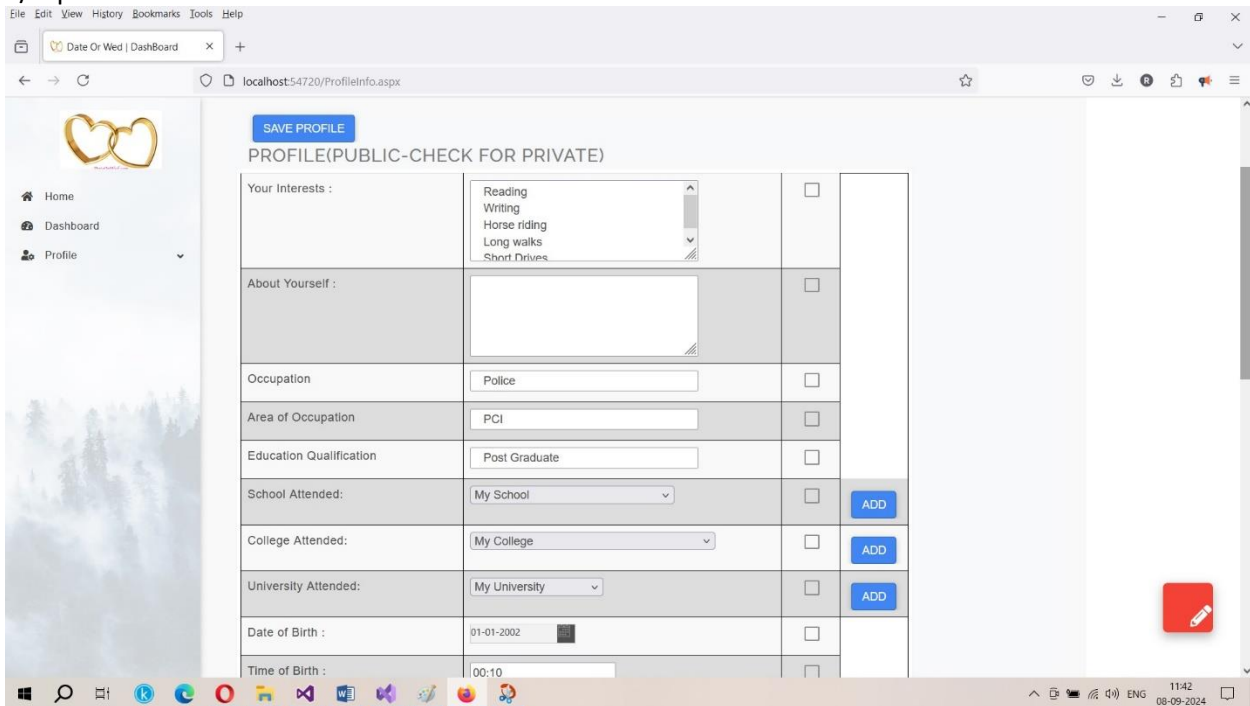
        </div>
    </div>

    </div>
<!-- End Description -->

</div>
</div>

</div></section>
<script>
    // Material Select Initialization
    $(document).ready(function () {
        $('mdb-select').materialSelect();
    });
</script>
</asp:Content>
<%--<asp:Content ID="Content6" ContentPlaceHolderID="ContentPlaceHolderfooter" Runat="Server">
</asp:Content>--%>

```



***** LoginInfo.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 DOWEntity;
using DOWDal;
using System.Data;
using System.Data.SqlClient;
using System.Configuration;
using System.Drawing;
using System.Web.Security;
using System.IO;
using System.Net;

public partial class LoginInfo : System.Web.UI.Page
{
    public string PageType;
    string pageid;
    int UserId;
    int Prid = 0;
    string user;
    DDLogTracer objlog = new DDLogTracer();
    public string UserName = "";

    protected void Page_Load(object sender, EventArgs e)
    {
        StringBuilder sb = new StringBuilder();
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr == null)
        {
            bool ret = new Common().guest_login();
            if (ret == false)
            {
                Response.Redirect("~/Login.aspx");
            }
        }

        if (Clsusr != null)
        {
            PageType = Page.Request.QueryString["PageType"];
            if (PageType == null)
            {
                PageType = "Edit";
            }

            Session["PageType"] = PageType;
            pageid = "4";
        }
    }
}

```

```

UserName=Common.Decrypt( Clsusr.UserName);
if (Convert.ToInt16(Clsusr.GroupId) < 4)
{
    Common.ShowMessageHtml("PI Login to View your Profile-" + Clsusr.UserName + "-" +
Clsusr.UserId + "-" + Clsusr.GroupId, 0, true, dvMsgBox);
    return;
}
PopDlIs ObjPD = new PopDlIs();
ObjPD.BindDropDowns(ddl_country, (new PopDlIs()).GetCountries());

ObjPD.BindDropDowns(ddl_GroupId, (new PopDlIs()).GetAllGroups());
ObjPD.BindDropDowns(ddl_mtongue, (new PopDlIs()).GetLanguages());

string ProductId = Page.Request.QueryString["Id"];
if (!this.IsPostBack)
{
    UserId = Convert.ToInt16(Clsusr.UserId);
    if (UserId > 0)
    {
        //-----user infor
        EteTblUser lst = (new tblUsers()).getuserbyid(UserId);
        //txt_username.Text = lst.UserId.ToString();
        txt_username.Text = lst.UserName;
        txt_username.Enabled = false;
        txt_FirstName.Text = lst.FirstName;
        txt_LastName.Text = lst.LastName;

        email_name.Text = lst.EmailID;
        if (email_name.Text == "" || email_name.Text == null)
        { email_name.Enabled = true; }
        txt_mobile.Text = lst.ContactNum;
        ddl_country.SelectedValue = (lst.Region);
        txt_mobile.Text = lst.ContactNum;
        ddl_GroupId.SelectedIndex = lst.GroupId;
        ddl_GroupId.Enabled = false;
        //ddl_GroupId.SelectedValue = lst.GroupId.ToString();
        //txt_Password.Text = new BalCommon().EncryptPwd(lst.Password.ToString());
        //txt_Password1.Text = new BalCommon().EncryptPwd(lst.Password.ToString());
        ddl_gender.SelectedValue = lst.Gender.ToString();
        ddl_mtongue.SelectedValue = lst.MTongue.ToString();
        txt_landline.Text = lst.Landline;
        sb = new StringBuilder();
        string chkval = "";
        if (lst.MatchingChk) { chkval = "checked"; };
        sb.Append("<input class='form-check-input' type='checkbox' value=" + id+'ad_chk'
runat='server'" + chkval + " />");
        sb.Append("<label class='form-check-label' for='ad_chk'>");
        d_ad_chk.InnerHtml = sb.ToString();
    }
}

```

```

        sb = new StringBuilder();
        chkval = "";
        if (lst.adshowchk) { chkval = "checked"; };
        sb.Append("<input class='form-check-input' type='checkbox' value="
id='currency_chk' runat='server'" + chkval + " />");
        sb.Append("<label class='form-check-label' for='currency_chk'>");
        d_currency_chk.InnerHtml = sb.ToString();

        //ad_chk.Value = lst.adshowchk.ToString();
        //currency_chk.Value = lst.MatchingChk.ToString();
        //-----user infor ends
        //ddl_users.SelectedValue = UserId.ToString();
        //ddl_users.Enabled = false;
    }
}

}
}

```

```

protected void btn3_Click(object sender, EventArgs e)
{
    //update tbluser
    objlog.Trace_Log(0, "DateOrWed", "MangeUser", "ContentSaveControlClicked", "Start Here..");
    ClsLoggedInUser ClsUser = (ClsLoggedInUser)Session["LoggedInUser_Specs"];

    //try
    {
        //if (BalCommon.IsValidEmail(txt_username.Text) == true)
        //{
            // 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 Match.", 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 &&
txt_landline.Text.Trim()!="")
{
    Common.ShowMessageHtml("Landline No is Numeric only", 0, true, dvMsgBox);
}
else if (new FieldValidator().ValidateAlpha(txt_FirstName) == false)
{
    Common.ShowMessageHtml("Name Alphabetics only", 0, true, dvMsgBox);
}

```

```

else if (new FieldValidator().ValidateAlpha(txt_LastName) == false)
{
    Common.ShowMessageHtml("Company Name Alphabetics only", 0, true, dvMsgBox);
}
//else if (new FieldValidator().ValidateAlpha(txt_region) == false)
//{

// Common.ShowMessageHtml("Region Alphabetics only", 0, true, dvMsgBox);
//}
else
{
    EtetblUser _tbluser = new EtetblUser();
    _tbluser.UserName = txt_username.Text;
    _tbluser.FirstName = txt_FirstName.Text;
    _tbluser.LastName = txt_LastName.Text;
    _tbluser.Department = ddl_GroupId.SelectedValue;
    _tbluser.Region = ddl_country.SelectedValue;
    _tbluser.GroupId = Convert.ToInt32(ddl_GroupId.SelectedValue);
    _tbluser.DealerId = 1;// Convert.ToInt32(ddl_DealerId.SelectedValue);
    _tbluser.EmailID = email_name.Text;
    _tbluser.ContactNum = txt_mobile.Text;
    _tbluser.Status = true;// chk_IsLoginAllowed.Checked;
    _tbluser.RoleID = 1;// Convert.ToInt32(ddl_RoleID.SelectedValue);
    _tbluser.Gender = ddl_gender.SelectedValue.ToString().Substring(0,1);
    _tbluser.adshowchk = Convert.ToBoolean(ad_chk.Checked);
    _tbluser.MatchingChk = Convert.ToBoolean(currency_chk.Checked);
    _tbluser.Landline = txt_landline.Text;
    _tbluser.MTongue =Convert.ToInt16( ddl_mtongue.SelectedValue);

    //if (txt_Password.Text!="")
    //{
    // _tbluser.Password = new BalCommon().EncryptPwd(txt_Password.Text);
    //}
    _tbluser.PagelId = Convert.ToInt32(Request.QueryString["pageid"]);
    _tbluser.UId = Convert.ToInt32(ClsUser.UserId);
    if (PageType == "Add")
    {

    }
    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 updated successfully.", 1, true, dvMsgBox);
    //Response.Redirect("~/Home.aspx?pageid=153&msg=User has been edited
successfully.");

}
else
{
    //ShowMessage("Update Un-Successful");
    Common.ShowMessageHtml("Update user failed.", 0, true, dvMsgBox);
}
}
}
}
//catch (Exception exc)
//{
//    objlog.Trace_Error(4, "DateOrWed", "MangeUser", "ContentSaveControlClicked", "Exception:"
+ exc.ToString());
//    Common.ShowMessageHtml(exc.ToString(), 0, true, dvMsgBox);

//}
objlog.Trace_Log(0, "DateOrWed", "MangeUser", "ContentSaveControlClicked", "Start Here..");
}

}
***** 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 DOWBal;
using System.Data;
//using DOWDal;
using DOWEntity;
using System.Web.Security;
using System.Net;
using System.IO;
using System.Web.Script.Serialization;

public partial class logout : System.Web.UI.Page
{
    DDLogTracer objlog = new DDLogTracer();
    public static string _Country;
    public static string _City;
    public static string ip;
    protected void Page_Load(object sender, EventArgs e)
    {
        try {
            if (HttpContext.Current.Session["LoggedUser_Specs"] != null)
            {
                ClsLoggedInUser Clsusr = (ClsLoggedInUser)HttpContext.Current.Session["LoggedUser_Specs"];
                string ip = GetIp();
                GetLocation(ip);
                string City = _City;
                string Country = _Country;
                BALCreateUser objbal = new BALCreateUser();
                objbal.LogOut(Convert.ToInt32(Clsusr.UserId), Convert.ToInt32(Clsusr.LoginID), City, Country, ip);
            }
        }
    }
}
```



```

HttpContext.Current.Session.Abandon();
Session.Clear();
Session.RemoveAll();
Session.Abandon();

HttpContextCookie myCookie = new HttpContextCookie("DOWCookie");
DateTime now = DateTime.Now;
myCookie.Value = "";
myCookie.Expires = now.AddYears(-1);
Response.Cookies.Clear();
Response.Cookies.Add(myCookie);

Response.Redirect("~/Login.aspx");
FormsAuthentication.SignOut();
}
else
{
    HttpContext.Current.Session.Abandon();
    FormsAuthentication.SignOut();
}

}
catch (Exception exc)
{
    objlog.Trace_Error(4, "DateOrWed", "Profile", "Load", "Exception:" + exc.ToString());
}

Response.Redirect("~/Login.aspx");
}

public List<Location> GetLocation(string ipAddress)
{
    List<Location> locations = new List<Location>();
    //string ipAddress = Request.ServerVariables["HTTP_X_FORWARDED_FOR"];
    //if (string.IsNullOrEmpty(ipAddress))
    //{
    //    ipAddress = Request.ServerVariables["REMOTE_ADDR"];
    //}

    string APIKey = "744627112f51a720c93d8e11741a02c6f50ad135ea28e733d10f52dc0596b193";
    string url = string.Format("http://api.ipinfodb.com/v3/ip-city/?key={0}&ip={1}&format=json",
APIKey, ipAddress);
    using (WebClient client = new WebClient())
    {
        string json = client.DownloadString(url);
        Location location = new JavaScriptSerializer().Deserialize<Location>(json);
        locations.Add(location);
    }
}

```

```

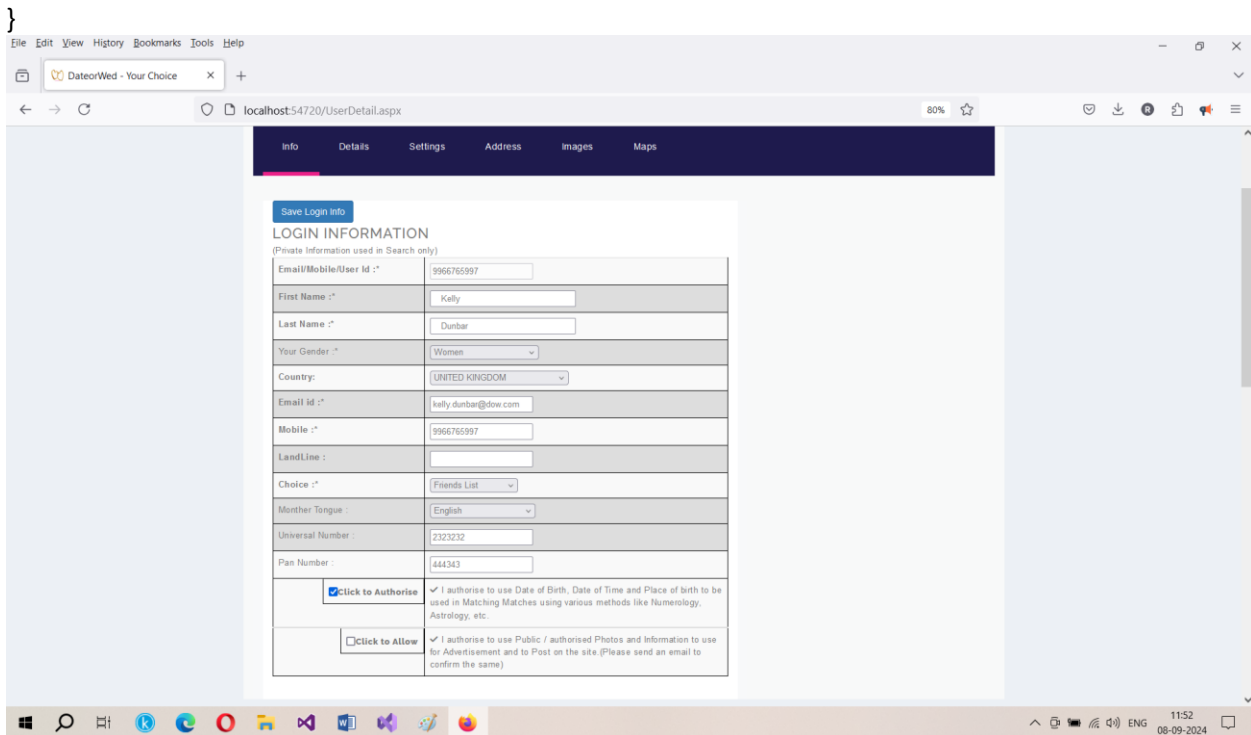
        _City = locations[0].CityName;
        _Country = locations[0].CountryName;

    }
    return locations;
}
protected string GetIp()
{
    String address = "";
    WebRequest request = WebRequest.Create("http://checkip.dyndns.org/");
    using (WebResponse response = request.GetResponse())
    using (StreamReader stream = new StreamReader(response.GetResponseStream()))
    {
        address = stream.ReadToEnd();
    }

    int first = address.IndexOf("Address: ") + 9;
    int last = address.LastIndexOf("</body>");
    address = address.Substring(first, last - first);

    return address;
}
}

```



***** ManageUser.aspx *****

```

<%@ Page Title="" Language="C#" MasterPageFile="~/DOWMasterPage.master"
AutoEventWireup="true" CodeFile="ManageUser.aspx.cs" Inherits="View_ManageUser" %>

```

```
<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 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">
                <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="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=" container-fluid row" style="display: block;" >
            <div id="Div2" class="col-lg-10 customertable">

                <div class="tabs_holder tabs_wrapper skin7 top" style="display: block; overflow: hidden;
position: relative;">

```



```

        <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">

```



```

using System.Web.UI.WebControls;
using DOWEntity;
using DOWBal;
//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["LoggedInUser_Specs"];
        if (ClsUser == null)
        {
            Response.Redirect("MLB_Login.aspx");
        }
        PageType = Page.Request.QueryString["PageType"];
        if (PageType == null)
        {
            PageType = "Add";
        }
        Session["PageType"] = PageType;
        EtetblUser lst = 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, "MyLaadBazar", "MangeUser", "Load", "Start Here..");
            try
            {
                username = Page.Request.QueryString["username"];
                int userid = Convert.ToInt16( Page.Request.QueryString["UserId"]);
                PopDlls ObjPD = new PopDlls();
                //ObjPD.BindDropDowns(ddl_RoleID, (new PopDlls()).GetRoles(ClsUser.RoleId));
                ObjPD.BindDropDowns(ddl_country, (new PopDlls()).GetCountries());
                ObjPD.BindDropDowns(ddl_GroupId, (new
                PopDlls()).GetGroups(Convert.ToInt32(ClsUser.UserId), Convert.ToInt32(ClsUser.GroupId)));
                if (PageType == "Edit" || PageType == "View")
                {
                    if (username != null)
                    {
                        lst = (new tblUsers()).getuser(username);
                    }
                }
            }
            catch { }
        }
    }
}

```



```

    }
    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 = 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, "MyLaadBazar", "OrderSearchList", "Bindetails_Filter", ex.Message);

}
    objlog.Trace_Log(0, "MyLaadBazar", "MangeUser", "Load", "Ends Here..");
}
}

//private void ManageNavigations(int pageid, int userid, int groupid,string PageType)
//{
//    objlog.Trace_Log(0, "MyLaadBazar", "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, "MyLaadBazar", "OrderSearchList", "Bindetails_Filter", ex.Message);

//  }
//  objlog.Trace_Log(0, "MyLaadBazar", "MangeUser", "ManageNavigations", "Ends Here..");
//}

// This is Event Handler
//protected void myBtnClickHandler(object sender, EventArgs e)
//{
//  objlog.Trace_Log(0, "MyLaadBazar", "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":
//          ContentSaveControlClicked();
//          break;
//      case "AUDIT":
//          Response.Redirect(drnew["pageurl"].ToString() + "&frompage=" +
Request.QueryString["pageid"] + "&ld=" + txt_username.Text);
//          break;

//      case "URL":
//          Response.Redirect(drnew["pageurl"].ToString());
//          break;
//      case "CANCEL":
//          ContentCancelControlClicked();
//          break;
//      default:

//          break;
//      }

//      }
//      }
//      }
//      catch (Exception exc)
//      {
//          objlog.Trace_Error(4, "MyLaadBazar", "MangeUser", "myBtnClickHandler", "Exception:" +
exc.ToString());
//          Common.ShowMessageHtml(exc.ToString(), 0, true, dvMsgBox);

//      }
//      objlog.Trace_Log(0, "MyLaadBazar", "MangeUser", "myBtnClickHandler", "Ends Here..");

//}

```

```

//private void ContentControlClicked()
//{
//    ShowMessage( "Other control clicked");
//}
private void ContentCancelControlClicked()
{
    objlog.Trace_Log(0, "MyLaadBazar", "MangeUser", "ContentCancelControlClicked",
"Starts Here..");
    try
    {

```

```

Common.ShowMessageHtml("", 1, false, dvMsgBox);
ClsLoggedInUser ClsUser = (ClsLoggedInUser)Session["LoggedInUser_Specs"];

PageType = Session["PageType"].ToString();
if (PageType == "Edit")
{
    //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, "MyLaadBazar", "MangeUser",
"ContentCancelControlClicked", "Exception:" + exc.ToString());
    Common.ShowMessageHtml(exc.ToString(), 0, true, dvMsgBox);
}
objlog.Trace_Log(0, "MyLaadBazar", "MangeUser", "ContentCancelControlClicked",
"Ends Here..");
}
private void ContentSaveControlClicked()
{
    objlog.Trace_Log(0, "MyLaadBazar", "MangeUser", "ContentSaveControlClicked", "Start
Here..");
    ClsLoggedInUser ClsUser = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    Etetbluserlog log=new Etetbluserlog();
    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("Admnin 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)
{
    Common.ShowMessageHtml("Landline No is Numeric only", 0, true, dvMsgBox);
}
else if (new FieldValidator().ValidateAlpha(txt_FirstName) == false)
{
    Common.ShowMessageHtml("Name Alphabetics only", 0, true, dvMsgBox);
}
else if (new FieldValidator().ValidateAlpha(txt_LastName) == false)
{
    Common.ShowMessageHtml("Company Name Alphabetics only", 0, true, dvMsgBox);
}

//else if (new FieldValidator().ValidateAlpha(txt_region) == false)
//{
// Common.ShowMessageHtml("Region Alphabetics only", 0, true, dvMsgBox);
//}
else
{
    EtetblUser _tbluser = new EtetblUser();
    _tbluser.UserName = txt_username.Text;
    _tbluser.FirstName = txt_FirstName.Text;
    _tbluser.LastName = txt_LastName.Text;
    _tbluser.Department = ddl_GroupId.SelectedValue;
    _tbluser.Region = ddl_country.SelectedValue;
    _tbluser.GroupId = Convert.ToInt32(ddl_GroupId.SelectedValue);
    _tbluser.DealerId = 1;// Convert.ToInt32(ddl_DealerId.SelectedValue);
    _tbluser.EmailID = txt_username.Text;
    _tbluser.ContactNum = txt_mobile.Text;
}

```

```

_tbluser.Status = true;// chk_IsLoginAllowed.Checked;
_tbluser.RoleID = 1;// Convert.ToInt32(ddl_RoleID.SelectedValue);
_tbluser.Password = new BalCommon().EncryptPwd(txt_Password.Text);
    _tbluser.PagelD=Convert.ToInt32( Request.QueryString["pageid"]);
    _tbluser.UId = Convert.ToInt32(ClsUser.UserId);
    if (PageType == "Add")
    {
        bool exist;
        exist =new tblUsers().UserNameExists(_tbluser.UserName);
        if (exist == false)
        {
            _tbluser.CreatedOn = DateTime.Now;
            sresult = (new tblUsers()).Insert(_tbluser,log);
            if (sresult == true)
            {
                Common.ShowMessageHtml("User has been
Created successfully.", 1, true, dvMsgBox);

                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
        {
            result = 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.");

    }
    else
    {
        //ShowMessage("Update Un-Sucessful");
        Common.ShowMessageHtml("Update user failed.", 0,
true, dvMsgBox);
    }
}
}
catch (Exception exc)
{
    objlog.Trace_Error(4, "MyLaadBazar", "MangeUser",
"ContentSaveControlClicked", "Exception:" + exc.ToString());
    Common.ShowMessageHtml(exc.ToString(), 0, true, dvMsgBox);
}
objlog.Trace_Log(0, "MyLaadBazar", "MangeUser", "ContentSaveControlClicked", "Start
Here..");
}

//protected void CV_username_ServerValidate(object source, ServerValidateEventArgs args)
//{
// //args.IsValid = FieldValidator.ValidateNumeric(txt_username);
//}
}
***** MyHeartList.aspx *****

```

```

<%@ Page Title="" Language="C#" MasterPageFile="~/DOWMasterPage.master"
AutoEventWireup="true" CodeFile="MyHeartList.aspx.cs" Inherits="MyHeartList" %>

<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">
            <div class="col-lg-12 col-md-12 col-sm-12">
                <div class="row">
                    <div id="dvMsgBox" class="span12 ordersave" runat="server" style="display: block;">
                        <asp:Label ID="Label1" runat="server" Text="" Height="1px"></asp:Label>
                    </div>
                </div>
            </div>
        </div>
    </div>

</asp:Content>
<asp:Content ID="Content4" ContentPlaceHolderID="CPH_MainContent" Runat="Server">
    <div class="content_holder">

        <!-- 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">MyHeartlist</h6>
                </div>
                <div class="row">
                    <div id="NewProducts" class="new-product slider" runat="server">

                        </div>
                    </div>
                </div>
            </div>
        </section>

```

```

    <!-- 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>
    <script type="text/javascript" src="plugins/FlexSlider/jquery.flexslider-min.js"></script>
    <script type="text/javascript" src="js/jquery.magnific-popup.min.js"></script>
    <script type="text/javascript" src="js/magnific-popup.min.js"></script>
    <script type="text/javascript" src="js/slick.js"></script>
    <script type="text/javascript" src="js/slick-slider.js"></script>
    <script type="text/javascript" src="js/custom.js"></script>

</div>

</asp:Content>
<asp:Content ID="Content5" ContentPlaceHolderID="ContentPlaceholder1" Runat="Server">
</asp:Content>
<asp:Content ID="Content6" ContentPlaceHolderID="ContentFooter" Runat="Server">
    <!-- START FOOTER -->
    <footer class="footer" id="footer" runat="server">
        <div class="container">
            <div class="row">
                <section class="home-main-contant-style3 bg-white">
                    <div class="parallax-block parallaxBg">
                        <div class="container">
                            <div class="parallax-block-text">
                                <p>DateorWed.com - YourChoice <span class="color-green">We are a social networking
site.</span> <br> Thank you! </p>
                            </div>
                            <div class="pre-btn-group">
                                <a class="btn btn-product mr15" href="#">Search Now</a>
                                <a class="btn btn-product mr15" href="#">Post a Profile</a>
                            </div>
                        </div>
                    </div>
                </section>
            </div>
        </div>
    <!-- End Parallax -->

```

```

<!-- START NEW -->
<section class="home-main-contant-style3 bg-white" style="display:none">
  <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>
</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="fab 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="#"><i class="fab fa-facebook"></i></a></li>
      <li><a href="#"><i class="fab fa-twitter"></i></a></li>
      <li><a href="#"><i class="fab fa-tumblr"></i></a></li>
      <li><a href="#"><i class="fab fa-instagram"></i></a></li>
      <li><a href="#"><i class="fab fa-linkedin"></i></a></li>
    </ul>
  </div>
</div>
</div>
</div>
</div>
</div>
</div>
<!--End Suvbscribe -->
<!-- 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 an Agent 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="#">dating</a></li>
        <li><a href="#">Dating & Wedding</a></li>
        <li><a href="#">Wedding</a></li>
        <li><a href="#">Option</a></li>
        <li><a href="#">Dating or Wedding</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="#">Dating <span class="pull-right">12</span></a></li>
      <li><a href="#">Wedding <span class="pull-right">11</span></a></li>
      <li><a href="#">Date or Wed <span class="pull-right">10</span></a></li>
      <li><a href="#">Date and Wed <span class="pull-right">14</span></a></li>
      <li><a href="#">Others <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 Profiles</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 Profiles</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="#">Looking for</a></h6>
          <div class="widget-posts-meta">Wed</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="#">Searching for</a></h6>
          <div class="widget-posts-meta">-Date-</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="#">Date or Wed</a></h6>
          <div class="widget-posts-meta">-Wed-</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>
```

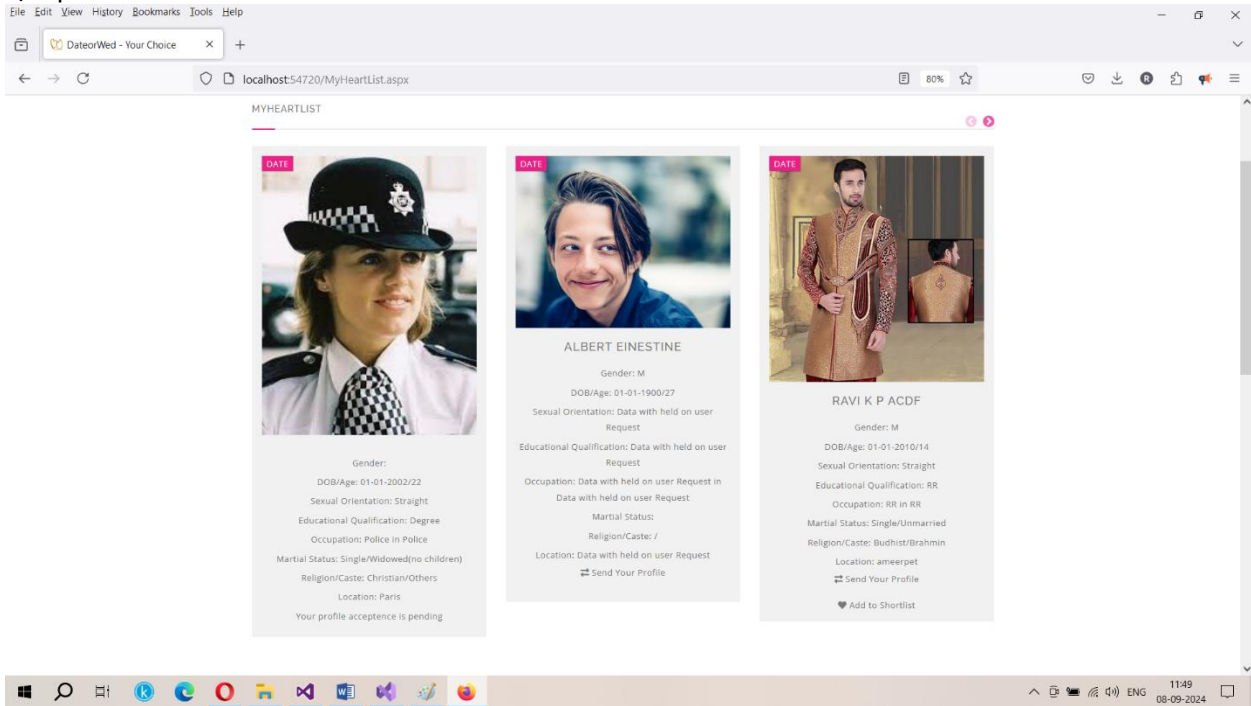

</div>

</div>

</footer>

<!-- END FOOTER-->

</asp:Content>



***** MyHeartList.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 DOWEntity;
using System.IO;
public partial class MyHeartList : System.Web.UI.Page
{
    DDLogTracer objlog = new DDLogTracer();
    bool IsMobile = false;
    protected void Page_Load(object sender, EventArgs e)
    {
        // string SrchStrng = "";
        int psid = 0;
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr == null)
```

```

{
    bool ret = new Common().guest_login();
    if (ret == false)
    {
        Response.Redirect("~/Login.aspx");
    }
}
if (Clsusr != null)
{
    string spsid = Common.Decrypt(Request.QueryString["pid"]);
    string dtype = Request.QueryString["dtype"];
    if (dtype == null) { dtype = "0"; }
    if (Clsusr != null && spsid != "")
    {
        psid = Convert.ToInt16(new tblUsers().getuserid(spsid));
    }

    if (psid > 0)
    {
        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;
        if (dtype == "0")
        {
            bool ret = new tblUsers().InsertCartList(cart);
        }
        else
        {
            bool ret = new tblUsers().DeleteCart(cart);
        }
    }

    { Bindetails(spsid); }
}
}
private void Bindetails(string SrchStrng)
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedIn_Specs"];
    if (Clsusr == null)
    {
        bool ret = new Common().guest_login();
        if (ret == false)
        {

```

```

        Response.Redirect("~/Login.aspx");
    }
}
int sno = 0;
string cat = "";
string PreHtml = "";
StringBuilder sb = new StringBuilder();

cat = Common.Decrypt(Request.QueryString["cat"]).Trim();
// int len = cat.Length;
// cat = cat.Substring(0, len - 15);
// cat = cat.TrimEnd('\0');

// sb.Append(" <div class='directory-list'>");
Etetbluser_options opt = new Etetbluser_options();

//try
{
    List<EteGetSearchResults> Lst = (new
tblUsers()).GetCartbyUserId(Convert.ToInt16(Clsusr.UserId));
    if (Lst.Count > 0)
    {
        var pst = (from c in Lst orderby c.UserId descending select c).FirstOrDefault();
        Lst=(from c in Lst orderby c.UserId descending select c).ToList();
        if (Lst != null && Lst.Count > 0)
        {

            //IList<EteGetSearchResults> Lst1 = new List<EteGetSearchResults>();

            foreach (var dt in Lst)
            {
                string path = "images/NoPhoto.png";
                string path1 = "images/No Photo.png";
                string href = "";
                string OutId = Common.Encrypt(dt.UserName.ToString());
                href = "ProfileDetail.aspx?id=" + OutId;
                var list = new tblUsers().getuserImagesbyuserid((int)dt.UserId, "").ToList();
                int cnt = list.Count();
                if (cnt > 1)
                {
                    path = list[0].ImgPath;
                    path1 = list[1].ImgPath;//"http://dateorwed.com/"+
                }
                else
                if (list.Count() > 0)
                {

```

```

        path = list[0].ImageUrl;
    }

    if (path == null || path == "")
    {
        path = "images/Nocolor.png";
    }
    if (File.Exists(path))
    {
        path = list[0].ImageUrl;
        path1 = list[1].ImageUrl;
    }
    if (path1 == null || path1 == "") { path1 = path; }

    string martsttus = new tblUsers().getmatstatus(dt.Martialstatus);
    string relgn = new tblUsers().getreligion(dt.Religion);
    string caste = new tblUsers().getcaste(dt.Caste);
    //slick code
    sb.Append("        <div class='col-lg-3 col-md-4 item'>        \n");
    sb.Append("        <div class='products grid-product'>        \n");
    sb.Append("        <div class='product-grid-inner'>\n");
    sb.Append("        <div class='product-grid-img'> \n");
    sb.Append("        <ul class='product-labels'>        \n");
    sb.Append("        <li>Date</li> \n");
    sb.Append("        </ul> \n");
    sb.Append("        <a href='ProfileDetail.aspx?id=" + OutId + "' 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'><a class="
href='ProfileDetail.aspx?id=" + OutId + "'>Detailed View</a></div>        \n");
    sb.Append("        </div> \n");
    sb.Append("        <div class='product-grid-caption text-center'> \n");
    sb.Append("        <h4><a class='p-grid-title' href='ProfileDetail.aspx?id=" + OutId + "'> +
dt.FirstName + " " + dt.LastName + "</a></h4> \n");
    //new code for details
    sb.Append("        <p class='mb5'>" + "Gender: " + dt.Gender + "</p>        \n");
    sb.Append("<p class='mb5'> DOB/Age: " + dt.dob.ToShortDateString() + "/" + dt.age + "<p>
\n");
    sb.Append("        <p class='mb5'>" + "Sexual Orientation: " + dt.Sexornt + "</p> \n");
    sb.Append("        <p class='mb5'>" + "Educational Qualification: " + dt.Eduqualification +
"</p> \n");
    sb.Append("        <p class='mb5'>" + "Occupation: " + dt.Occupation + " in " +
dt.Occupationtype + "</p> \n");
    sb.Append("        <p class='mb5'>" + "Martial Status: " + martsttus + "</p>        \n");

```

```

        sb.Append("      <p class='mb5'>" + "Religion/Caste: " + relgn + "/" + caste + "</p>
\n");
        sb.Append("      <p class='mb5'>" + "Location: " + dt.Location + "</p>  \n");
        if (Convert.ToInt16(Clsusr.GroupId) > 3)
        {
            if (!dt.SentProfileAdded)
            {
                sb.Append("      <i class='fa fa-exchange-alt'></i>      \n");

                sb.Append("      <a href='SendProfile.aspx?psid=" + OutId + "'>Send Your
Profile</a><br><br>  \n");
            }
            else
            {
                if (dt.SentSltype)
                {
                    sb.Append("      <p class='mb5'>Your profile has been accepted by them</p>
\n");
                }
                else
                {
                    if (dt.SentActive)
                    {
                        sb.Append("      <p class='mb5'>Your profile acceptance is pending</p> \n");
                    }
                    else
                    {
                        sb.Append("      <p class='mb5'>Your profile has been rejected</p>      \n");
                    }
                }
            }
        }

        if (!dt.ShortListAdded) {
            sb.Append("      <i class='fa fa-heart'></i>      \n");
            sb.Append("      <a href='MyShortList.aspx?psid=" + OutId + "'>Add to Shortlist</a><br>
\n");
        }

//          sb.Append("<br>      <a href='MyHeartList.aspx?pid=" + OutId + "' class='btn btn-
product fa fa-heart'>Add to List</a>      \n"); //<i class='fa fa-heart'>

    }

//new code for details ends
sb.Append("      </div> \n");
sb.Append("      </div> \n");
sb.Append("      </div> \n");

```

```

        sb.Append("        </div> \n");
        //grid code
        sno++;
        if (sno > 3)
        {
            sno = 1;
        }

    }//for each list

    PreHtml = sb.ToString();
    if (PreHtml.Length > 0)
    {
        objlog.Trace_Log(4, "Dashboard", "1", "NewProducts", PreHtml, "Log".ToString());
        NewProducts.InnerHtml = PreHtml;
    }

}
}
else
{
    //data not found
    //Common.ShowMessageHtml("Search Options or Data not found", 1, true, this.dvMsgBox);
}

}
// catch (Exception ex)
//{

//Common.ShowMessageHtml(ex.Message, 1, true, this.dvMsgBox);
// ExceptionLogging.SendExcepToDB(ex);

//}
}

}
***** MyShortList.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/DOWMasterPage.master"
AutoEventWireup="true" CodeFile="MyShortList.aspx.cs" Inherits="MyShortList" %>

```

```

<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">
      <div class=" col-lg-12 col-md-12 col=col-sm-12">
        <div class="row">
          <div id="dvMsgBox" class="span12 ordersave" runat="server" style="display: block;">
            <asp:Label ID="Label1" runat="server" Text="" Height="1px"></asp:Label>
          </div>
        </div>
      </div>
    </div>
  </div>
</asp:Content>
<asp:Content ID="Content4" ContentPlaceHolderID="CPH_MainContent" Runat="Server">
  <div class="content_holder">

    <!-- 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">MyShortlist</h6>
        </div>
        <div class="row">
          <div id="NewProducts" class="new-product slider" runat="server">

          </div>
        </div>
      </div>
    </section>

```

```

    <!-- 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>
    <script type="text/javascript" src="plugins/FlexSlider/jquery.flexslider-min.js"></script>
    <script type="text/javascript" src="js/jquery.magnific-popup.min.js"></script>
    <script type="text/javascript" src="js/magnific-popup.min.js"></script>
    <script type="text/javascript" src="js/slick.js"></script>
    <script type="text/javascript" src="js/slick-slider.js"></script>
    <script type="text/javascript" src="js/custom.js"></script>

</div>

</asp:Content>
<asp:Content ID="Content5" ContentPlaceHolderID="ContentPlaceHolder1" Runat="Server">
</asp:Content>
<asp:Content ID="Content6" ContentPlaceHolderID="ContentFooter" Runat="Server">
    <!-- START FOOTER -->
    <footer class="footer" id="footer" runat="server">
        <div class="container">
            <div class="row">
<section class="home-main-contant-style3 bg-white">
    <div class="parallax-block parallaxBg">
        <div class="container">
            <div class="parallax-block-text">
                <p>DateorWed.com - YourChoice <span class="color-green">We are a social networking
site.</span> <br> Thank you! </p>
            </div>
            <div class="pre-btn-group">
                <a class="btn btn-product mr15" href="#">Search Now</a>
                <a class="btn btn-product mr15" href="#">Post a Profile</a>
            </div>
        </div>
    </div>
</section>
    <!-- End Parallax -->

    <!-- START NEW -->
    <section class="home-main-contant-style3 bg-white" style="display:none">
        <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>
</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>
  </div>
</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="fab 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="#"><i class="fab fa-facebook"></i></a></li>
      <li><a href="#"><i class="fab fa-twitter"></i></a></li>
      <li><a href="#"><i class="fab fa-tumblr"></i></a></li>
      <li><a href="#"><i class="fab fa-instagram"></i></a></li>
      <li><a href="#"><i class="fab fa-linkedin"></i></a></li>
    </ul>
  </div>
</div>
</div>
</div>
</div>
<!--End Suvbscribe -->
<!-- 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 an Agent 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="#">dating</a></li>
      <li><a href="#">Dating & Wedding</a></li>
      <li><a href="#">Wedding</a></li>
      <li><a href="#">Option</a></li>
      <li><a href="#">Dating or Wedding</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="#">Dating <span class="pull-right">12</span></a></li>
      <li><a href="#">Wedding <span class="pull-right">11</span></a></li>
      <li><a href="#">Date or Wed <span class="pull-right">10</span></a></li>
      <li><a href="#">Date and Wed <span class="pull-right">14</span></a></li>
      <li><a href="#">Others <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 Profiles</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 Profiles</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="#">Looking for</a></h6>
        <div class="widget-posts-meta">Wed</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="#">Searching for</a></h6>
        <div class="widget-posts-meta">-Date-</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="#">Date or Wed</a></h6>
        <div class="widget-posts-meta">-Wed-</div>
      </div>
    </li>
  </ul>
</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>
```

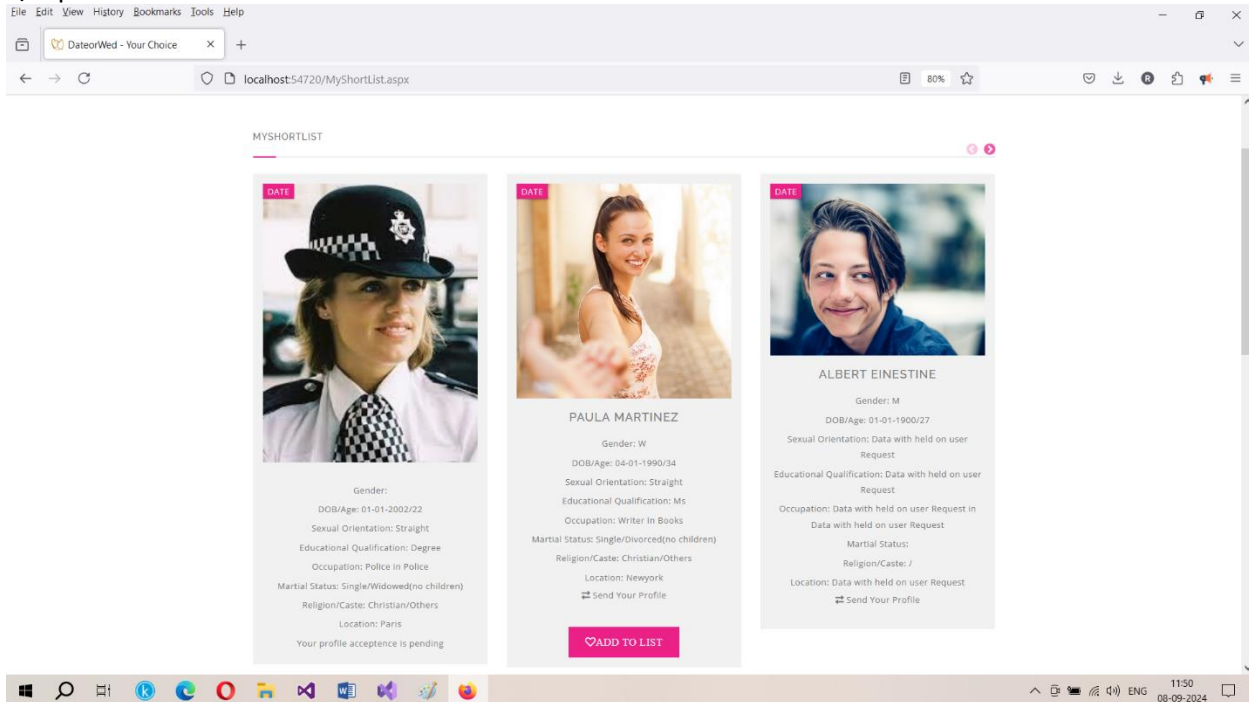
```
  </div>
```

```
</div>
```

```
</div>
```

```
</footer>
<!-- END FOOTER-->
```

```
</asp:Content>
```



```
***** MyShortList.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 DOWEntity;
using System.IO;
public partial class MyShortList : 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)
        {
            bool ret = new Common().guest_login();
            if (ret == false)
            {
                Response.Redirect("~/Login.aspx");
            }
        }
    }
}
```

```

    }
}
if (Clsusr != null)
{
    string psid = (Request.QueryString["psid"]);
    if (psid != null)
    {
        EteUserShortList lst = new EteUserShortList();
        lst.UserId = Convert.ToInt16(Clsusr.UserId);
        lst.PSId = new tblUsers().getuserid(Common.Decrypt(psid));
        lst.UId = Convert.ToInt16(Clsusr.UserId);

        lst.CreatedOn = DateTime.Today;
        lst.Active = true;
        bool result = new tblUsers().InsertShortList(lst, true);

    }    { Bindetails(psid); }

}
}
private void Bindetails(string SrchStrng)
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    if (Clsusr == null)
    {
        bool ret = new Common().guest_login();
        if (ret == false)
        {
            Response.Redirect("~/Login.aspx");
        }
    }
    int sno = 0;
    string cat = "";
    string PreHtml = "";
    StringBuilder sb = new StringBuilder();

    cat = Common.Decrypt(Request.QueryString["cat"]).Trim();
    // int len = cat.Length;
    // cat = cat.Substring(0, len - 15);
    // cat = cat.TrimEnd('\0');

    // sb.Append("<div class='directory-list'>");
    Etebluser_options opt = new Etebluser_options();

    //try
    {

```

```

List<EteGetSearchResults> Lst = (new
tblUsers()).GetShortListbyUserId(Convert.ToInt16(Clsusr.UserId));
if (Lst.Count > 0)
{
var pst = (from c in Lst orderby c.UserId descending select c).FirstOrDefault();
Lst=(from c in Lst orderby c.UserId descending select c).ToList();
if (Lst != null && Lst.Count > 0)
{

//IList<EteGetSearchResults> Lst1 = new List<EteGetSearchResults>();

foreach (var dt in Lst)
{
string path = "images/NoPhoto.png";
string path1 = "images/No Photo.png";
string href = "";
string OutId = Common.Encrypt(dt.UserName.ToString());
href = "ProfileDetail.aspx?id=" + OutId;
var list = new tblUsers().getuserimagesbyuserid((int)dt.UserId, "").ToList();
int cnt = list.Count();
if (cnt > 1)
{
path = list[0].ImgPath;
path1 = list[1].ImgPath;//"http://dateorwed.com/"+
}
else
if (list.Count() > 0)
{
path = list[0].ImageUrl;
}

if (path == null || path == "")
{
path = "images/Nocolor.png";
}
if (File.Exists(path))
{
path = list[0].ImageUrl;
path1 = list[1].ImageUrl;
}
if (path1 == null || path1 == "") { path1 = path; }

string martsttus = new tblUsers().getmatstatus(dt.Martialstatus);
string relgn = new tblUsers().getreligion(dt.Religion);
string caste = new tblUsers().getcaste(dt.Caste);
//slick code
sb.Append("        <div class='col-lg-3 col-md-4 item'>        \n");
sb.Append("        <div class='products grid-product'>        \n");

```



```

        sb.Append("        <div class='product-grid-inner'>\n");
        sb.Append("        <div class='product-grid-img'> \n");
        sb.Append("        <ul class='product-labels'> \n");
        sb.Append("        <li>Date</li> \n");
        sb.Append("        </ul> \n");
        sb.Append("        <a href='ProfileDetail.aspx?id=" + OutId + "' class='product-grid-img-
thumbnail'> \n");
        sb.Append("        <img src="" + path + "" alt='img' class='img-responsive primary-img' />
\n");
        sb.Append("        <img src="" + path1 + "" alt='img' class='img-responsive socendary-img'
/> \n");
        sb.Append("        </a> \n");
        sb.Append("        <div class='product-grid-atb text-center'><a class="
href='ProfileDetail.aspx?id=" + OutId + "'>Detailed View</a></div> \n");
        sb.Append("        </div> \n");
        sb.Append("        <div class='product-grid-caption text-center'> \n");
        sb.Append("        <h4><a class='p-grid-title' href='ProfileDetail.aspx?id=" + OutId + "'>" +
dt.FirstName + " " + dt.LastName + "</a></h4> \n");
        //new code for details
        sb.Append("        <p class='mb5'>" + "Gender: " + dt.Gender + "</p> \n");
        sb.Append("<p class='mb5'> DOB/Age: " + dt.dob.ToShortDateString() + "/" + dt.age + "<p>
\n");
        sb.Append("        <p class='mb5'>" + "Sexual Orientation: " + dt.Sexornt + "</p> \n");
        sb.Append("        <p class='mb5'>" + "Educational Qualification: " + dt.Eduqualification +
"</p> \n");
        sb.Append("        <p class='mb5'>" + "Occupation: " + dt.Occupation + " in " +
dt.Occupationtype + "</p> \n");
        sb.Append("        <p class='mb5'>" + "Martial Status: " + martsttus + "</p> \n");
        sb.Append("        <p class='mb5'>" + "Religion/Caste: " + relgn + "/" + caste + "</p>
\n");
        sb.Append("        <p class='mb5'>" + "Location: " + dt.Location + "</p> \n");
        if (Convert.ToInt16(Clsusr.GroupId) > 3)
        {
            if(!dt.SentProfileAdded)
            {
                sb.Append("        <i class='fa fa-exchange-alt'></i> \n");
                sb.Append("        <a href='SendProfile.aspx?psid=" + OutId + "'>Send Your
Profile</a><br><br> \n");
            }
            else
            {
                if (dt.SentSltype)
                {
                    sb.Append("        <p class='mb5'>Your profile has been accepted by them</p> \n");
                }
                else
                {
                    if (dt.SentActive)

```

```

        {
            sb.Append("        <p class='mb5'>Your profile acceptance is pending</p> \n");
        }
        else
        {
            sb.Append("        <p class='mb5'>Your profile has been rejected</p>        \n");
        }
    }
}

}
//sb.Append("    <i class='fa fa-heart'></i>        \n");
//sb.Append("    <a href='MyShortList.aspx?psid=" + OutId + "'>Add to Shortlist</a><br>
\n");
if (!dt.HeartListAdded)
{
    sb.Append("<br>        <a href='MyHeartList.aspx?pid=" + OutId + "' class='btn btn-
product fa fa-heart'>Add to List</a>        \n"); //<i class='fa fa-heart'>
}
}

//new code for details ends

sb.Append("        </div> \n");
sb.Append("        </div> \n");
sb.Append("        </div> \n");
sb.Append("        </div> \n");

//grid code
sno++;
if (sno > 3)
{
    sno = 1;
}

} //for each list

PreHtml = sb.ToString();
if (PreHtml.Length > 0)
{
    objlog.Trace_Log(4, "Dashboard", "1", "NewProducts", PreHtml, "Log".ToString());
}

```

```

        NewProducts.InnerHtml = PreHtml;

    }

}
else
{
    //data not found
    //Common.ShowMessageHtml("Search Options or Data not found", 1, true, this.dvMsgBox);

}

}
// catch (Exception ex)
//{

//Common.ShowMessageHtml(ex.Message, 1, true, this.dvMsgBox);
// ExceptionLogging.SendExcepToDB(ex);

//}
}

}
***** PersonalHome.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/DOWMasterPage.master"
AutoEventWireup="true" CodeFile="PersonalHome.aspx.cs" Inherits="PersonalHome" %>

<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>

</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>
  <!-- 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>

```

```

***** PersonalHome.aspx.cs *****

```

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using DOWEntity;

```

```

public partial class PersonalHome : System.Web.UI.Page
{

```

```

protected void Page_Load(object sender, EventArgs e)
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    if (Clsusr == null)
    {
        bool ret = new Common().guest_login();
        if (ret == false)
        {
            Response.Redirect("~/Login.aspx");
        }
    }
}
}
}
***** ProfileAddress.aspx *****
<%@ Page Title="DateorWed - Your Profile" Language="C#"
MasterPageFile="~/DashBoardMasterPage.master" AutoEventWireup="true"
CodeFile="ProfileAddress.aspx.cs" Inherits="ProfileAddress" %>

```

```

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header" Runat="Server">
    <script type="text/javascript" src="js/maps.js"></script>
    <script src="https://js.api.here.com/v3/3.1/mapsjs-core.js"
type="text/javascript" charset="utf-8"></script>
<script src="https://js.api.here.com/v3/3.1/mapsjs-service.js"
type="text/javascript" charset="utf-8"></script>
<%-- var platform = new H.service.Platform({ 'apikey':
'sRtQ7lqu4sVI4RaHYe089ukcoH5i9R1AdOeQWB7yXto'}));
var defaultLayers = platform.createDefaultLayers();
var map = new H.Map(
document.getElementById('mapContainer'),
defaultLayers.vector.normal.map,
{
zoom: 10,
center: { lat: 17.3850, lng: 78.4867 }
});--%>
<script>
// Initialize and add the map
function initMap() {
// The location of Uluru
var uluru = { lat: 17.3850, lng: 78.4867 };
// The map, centered at Uluru
var map = new google.maps.Map(
document.getElementById('map'), { zoom: 4, center: uluru });
// The marker, positioned at Uluru
var marker = new google.maps.Marker({ position: uluru, map: map });
}

```

```

</script>

</asp:Content>
<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
  <div class="container">
    <div class="row-fluid" style="padding-top: 20px; padding-left:50px">
      <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="Content5" ContentPlaceHolderID="ContentMainPage" Runat="Server">
  <!-- Content Bodey Here -->

  <section class=" dip-style ml-5"> <!--main-contain-->
    <div class="container bg-gray">
      <div class="login-area mb30">

        <div class="tab-content ml-5">
          <!-- Description -->

          <!-- Address -->
          <div class="tab-pane media-block active" id="Address">
            <div class="row-fluid">
              <div id="addlist" runat="server" class=" col-lg-12 col-md-12 col-sm-12 " span3=""
style=""><h3>Addresses(Private)</h3></div>
              <asp:Button ID="btn_SaveAddress" runat="server" CssClass="btn btn-primary btn-md btn-
rounded" Text="Save Address" Enabled="false" 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-12 col-md-12 col-sm-12" style="background-color: #ffffff; border:
1px solid #eefef; border-top: none; padding: 15px 5px;">
                <!--
                <h3>Add Addresses</h3>-->
                <table id="Addressdet" class="table table-striped table-detail col-lg-6 col-md-6
col-sm-12" runat="server" visible="false">
                  <tbody>

```

```

        <tr>
            <td class="col-lg-4 col-md-4 col-sm-3" dir="auto">Address Type :</td>
            <td>
                <asp:DropDownList ID="ddl_addtype" CssClass="show" runat="server">
<asp:ListItem Value="0">-- Select Your Address Type --</asp:ListItem>
<asp:ListItem Value="1">Home Address</asp:ListItem>
<asp:ListItem Value="2">Office Address</asp:ListItem>
<asp:ListItem Value="3">Factory Address</asp:ListItem>
<asp:ListItem Value="4">Billing Address</asp:ListItem>

                </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" CssClass="show"
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>

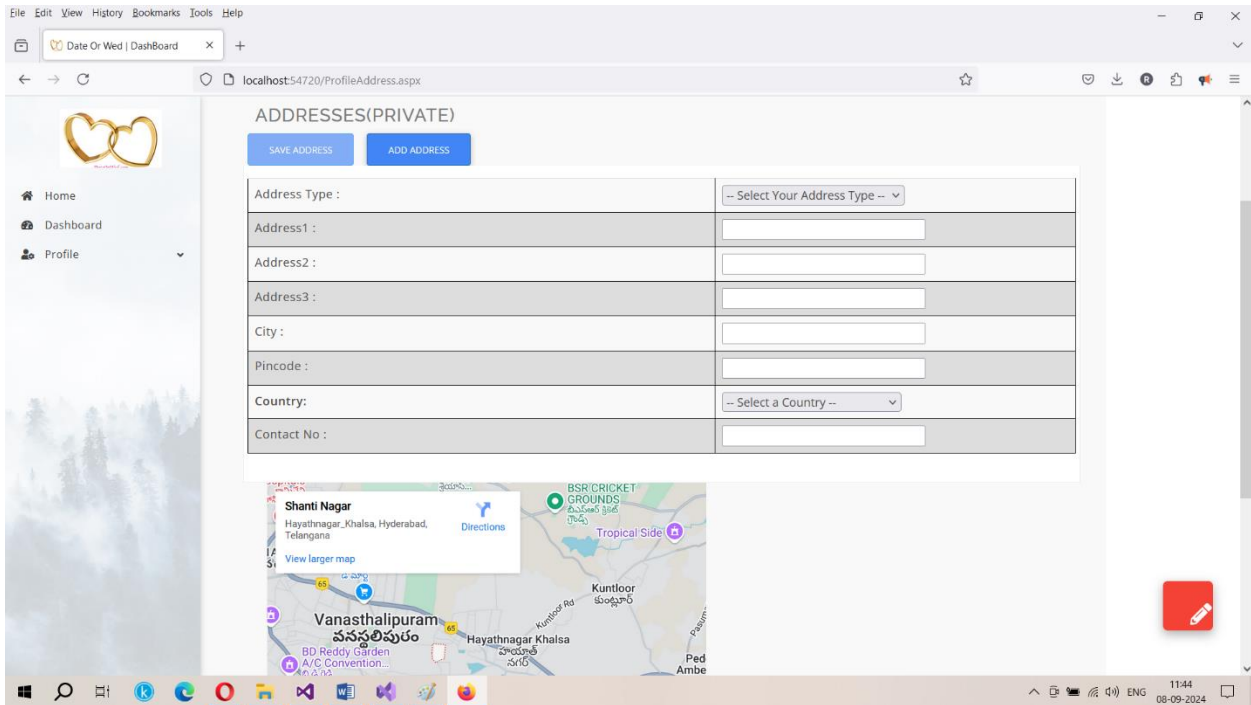
</div>
<!-- End Description -->
</div>
<div class="tab-pane fade in active" id="Maps">
<div style="width: 640px; height: 480px" id="mapContainer"> <!--%-->
<%-- <script async defer
src="https://maps.googleapis.com/maps/api/js?key=AlzaSyBSxyfN6Qe4VZNXau0NRbSIW4dgJWtYOLU&
callback=initMap">
</script>--%>
<div class="mapouter">

```

```
<div class="gmap_canvas col-sm-12"> <!--width="600" height="500" --%>
  <iframe class="col-lg-12 col-md-12 col-sm-12" height="500" width="600" id="gmap_canvas"
src="https://maps.google.com/maps?q=Shanthinagar, Vanasthalipuram, Hyderabad
500070&t=&z=13&ie=UTF8&iwloc=&output=embed" frameborder="0" scrolling="no" marginheight="0"
marginwidth="0" runat="server"></iframe>
  <a
href="https://www.embedgooglemap.net">embedgooglemap.net</a></div><style>.mapouter{position:
relative;text-align:right;height:500px;width:600px;}.gmap_canvas
{overflow:hidden;background:none!important;height:500px;width:600px;}</style>

</div>
</div>
</div>
  </div>
    </div>

<!--
  <script type="text/javascript" src="js/jquery.min.js"></script>
  <script type="text/javascript" src="js/bootstrap.min.js"></script>
  <script type="text/javascript" src="plugins/owl-carousel/owl.carousel.min.js"></script>
  <script type="text/javascript" src="plugins/FlexSlider/jquery.flexslider-min.js"></script>
  <script type="text/javascript" src="js/jquery.magnific-popup.min.js"></script>
  <script type="text/javascript" src="js/product.js"></script>
  <script type="text/javascript" src="js/magnific-popup.min.js"></script>
  <script type="text/javascript" src="js/slick.js"></script>
  <script type="text/javascript" src="js/slick-slider.js"></script>
  <script type="text/javascript" src="js/custom.js"></script>--%>
</div></section>
</asp:Content>
<!--<asp:Content ID="Content6" ContentPlaceHolderID="ContentFooter" Runat="Server">
</asp:Content>--%>
```



***** ProfileAddress.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 DOWEntity;
using DOWDal;
using System.Data;
using System.Data.SqlClient;
using System.Configuration;
using System.Drawing;
using System.Web.Security;
using System.IO;
using System.Net;

public partial class ProfileAddress : System.Web.UI.Page
{
    public string PageType;
    string pageid;
    int UserId;
    int PrId = 0;
    string user;
    DDLogTracer objlog = new DDLogTracer();

```



```

protected void Page_Load(object sender, EventArgs e)
{
    //dob.DropDownCalendarID = webcal.ID;
    string ip_add = GetIp();
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    if (Clsusr == null)
    {
        bool ret = new Common().guest_login();
        if (ret == false)
        {
            Response.Redirect("~/Login.aspx");
        }
    }

    if (Clsusr != null)
    {
        PageType = Page.Request.QueryString["PageType"];
        if (PageType == null)
        {
            PageType = "Edit";
        }

        Session["PageType"] = PageType;
        pageid = "4";
        if (Convert.ToInt16( Clsusr.GroupId) <4)
        {
            Common.ShowMessageHtml("Pl Login to View your Profile-"+Clsusr.UserName+"-
"+Clsusr.UserId+"-"+Clsusr.GroupId, 0, true, dvMsgBox);
            return;
        }
        //if (!this.IsPostBack)

        PopDlIs ObjPD = new PopDlIs();
        ObjPD.BindDropDowns(ddl_country1, (new PopDlIs()).GetCountries());
        Addressdet.Visible = true;

        //ddl_area
        string ProductId = Page.Request.QueryString["Id"];
        UserId = Convert.ToInt16(Clsusr.UserId);

        StringBuilder sb = new StringBuilder();

        List<Etetbluser_address> Lst = (new tblUsers()).getusersaddressbyuserid(UserId, "");
        if (Lst.Count() > 0)
    }
}

```

```

{
    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%'>Address</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.FirstName + " " + Clsusr.LastName + "</td> \n");
        sb.Append(" <td>" + add.address1 + ", " + add.address2 + ", " + add.address3 +
"</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.pincodes + "</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;&nbsp;Edit</a></td>");
        sb.Append(" </tr></Table> \n");
    }
}

```

```

        string adds = add.address1 + "," + add.address2 + "," + add.address3 + "," + add.city;
        if (adds != "")
        {
            gmap_canvas.Src = "https://maps.google.com/maps?q=" + adds +
"&t=&z=13&ie=UTF8&iwloc=&output=embed";
        }
    }
    sb.Append(" </tbody> \n");
    objlog.Trace_Log(1, "home", "1", "menu", "\n" + sb.ToString());

    addlist.InnerHtml = sb.ToString();
    addlist.Visible = true;
    //Addressdet.Visible = false;
}

}

}

protected string GetIp()
{
    String address = "";
    WebRequest request = WebRequest.Create("http://checkip.dyndns.org/");
    using (WebResponse response = request.GetResponse())
    using (StreamReader stream = new StreamReader(response.GetResponseStream()))
    {
        address = stream.ReadToEnd();
    }

    int first = address.IndexOf("Address: ") + 9;
    int last = address.LastIndexOf("</body>");
    address = address.Substring(first, last - first);

    return address;
}

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();
    IPEndPoint ipEntry = System.Net.Dns.GetHostEntry(strHostName);
    IPAddress[] addr = ipEntry.AddressList;
    string ip1 = addr[1].ToString();

    BALCreateUser objlogin = new BALCreateUser();

```

```

        DataTable status_login = objlogin.LoginGuest(Common.Encrypt(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 = new
tblUsers().getusername(Convert.ToInt16(objLoggedInUser.UserId));
            //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"]);
            objLoggedInUser.FirstName = status_login.Rows[0]["FirstName"].ToString();
            objLoggedInUser.LastName = status_login.Rows[0]["LastName"].ToString();
            Session["LoggedUser_Specs"] = objLoggedInUser;

            FormsAuthentication.SetAuthCookie(objLoggedInUser.UserName, false);
            HttpCookie myCookie = new HttpCookie("DOWCookie");
            DateTime now = DateTime.Now;
            myCookie.Value = objLoggedInUser.UserName;
            myCookie.Expires = now.AddYears(1);

            Response.Cookies.Add(myCookie);
        }
    }
    private DataTable GetData(SqlCommand cmd)
    {
        DataTable dt = new DataTable();
        String strConnString =
System.Configuration.ConfigurationManager.ConnectionStrings["DOWConectingString"].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();
    }
}
protected void SaveAdd_Click(object sender, EventArgs e)
{
    Addressdet.Visible = true;
}

protected void btn_SaveAddress_Click(object sender, System.EventArgs e)
{//save code
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"]; ;
    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.userid =Convert.ToInt16( Clsusr.UserId);
    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)
    {
        string adds = txt_add1.Text + "," + txt_add2.Text + "," + txt_add3.Text + "," + txt_city.Text;
        gmap_canvas.Src = "https://maps.google.com/maps?q=" + adds +
"&t=&z=13&ie=UTF8&iwloc=&output=embed";
        Common.ShowMessageHtml("Address Updated", 0, true, dvMsgBox);
        StringBuilder sb = new StringBuilder();

        List<Etetbluser_address> Lst = (new tblUsers()).getusersaddressbyuserid(UserId, "");
        if (Lst.Count() > 0)
        {
            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%'>Address</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>" + Clusr.UserName + "</td>        \n");
        sb.Append("        <td>" + add.address1 + add.address2 + add.address3 + "</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.pincodes + "</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();
}
Addressdet.Visible = false;

//if (Lst != null && Lst.Count > 0)
//{
//    WGrid_UserAddress.DataSource = Lst;
//    WGrid_UserAddress.DataBind();
//    WGrid_UserAddress.Visible = true;
//}
//else
//{
//    WGrid_UserAddress.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_addaddress_Click(object sender, EventArgs e)
{
    Addressdet.Visible =true;
    btn_SaveAddress.Enabled = true;
}
}

```

***** ProfileDetail.aspx *****

```

<%@ Page Title="" Language="C#" MasterPageFile="~/DOWMasterPage.master"
AutoEventWireup="true" CodeFile="ProfileDetail.aspx.cs" Inherits="ProfileDetail" %>

```

```

<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header" Runat="Server">
    <link rel="stylesheet" href="ProfDet/css/bootstrap.min.css"/>
    <link rel="stylesheet" href="ProfDet/css/font-awesome.min.css"/>
    <link rel="stylesheet" href="ProfDet/css/style.css"/>
    <link rel="stylesheet" href="ProfDet/css/animate.css"/>
    <link rel="stylesheet" href="ProfDet/css/magnific-popup.css"/>
    <link rel="stylesheet" href="ProfDet/plugins/FlexSlider/jquery.flexslider.css"/>
    <link rel="stylesheet" href="ProfDet/plugins/owl-carousel/owl.carousel.css"/>
    <link rel="stylesheet" href="ProfDet/plugins/owl-carousel/owl.theme.css"/>
    <link rel="stylesheet" type="text/css" href="ProfDet/css/slick.css"/>
    <link rel="stylesheet" type="text/css" href="ProfDet/css/slick-theme.css"/>

    <!-- WEB FONTS ProfDet/-->

```



```
<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: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">
```

```
<!-- Site Wrapper -->
```

```
<div class="wrapper">
```

```
<!-- Header -->
```

```
<div class="header header-static">
```

```
<!-- Content Bodey Here -->
```

```
<section class="main-contain shop-product bg-white">
```

```
<div class="containe">
```

```
<div class="row">
```

```
<!--Image Block-->
```

```
<script type="text/javascript" src="dowcss/js/bootstrap.min.js"></script>
```

```
<script type="text/javascript" src="plugins/FlexSlider/jquery.flexslider-min.js"></script>
```

```
<div id="imglist" class="signal-product-image mr15 pl-lg-5 pl-md-5 pr-lg-5 pr-md-5 mb-5"
runat="server"> <!--pl-lg-5 pl-md-5-->
```

```
<div class="col-md-5 col-sm-6 ">
```

```
<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 >
```

```
<a class="cd-lightbox-image cd-lightbox-type-image"
href="img/products/pro1.jpg" title="" style="text-decoration: underline">
```

```
<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>
        </ul>
    </div>
</div>

```

```
        </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>
```

```

</div>
</div>
</div>
<!--Image Block-->
<div id="shopdetail" class=" products-list-description " runat="server">
    <div class="shop-product-heading ">
        <h2 class="color-green">Printed Summer Dress</h2>
    </div>

    <ul class="list-inline product-ratings mb30">
        <li><span> </span><i class="rating-selected fa fa-star"></i></li>
        <li><i class="rating-selected fa fa-question-circle"></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 pull-right">
            <span>(1) <a href="#">Review</a> | <a href="#" dir="rtl"> 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>
    <label for="size-1">S</label>
  </li>
  <li>
    <input type="radio" id="size-2" name="size">
    <label for="size-2">M</label>
  </li>
  <li>
    <input type="radio" id="size-3" name="size">
    <label for="size-3">L</label>
  </li>
  <li>
    <input type="radio" id="size-4" name="size">
    <label for="size-4">XL</label>
  </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">checked</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>
  <button type="button" class="btn btn-product">Add to Cart</button>
</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>

```

```

</div>
</div>

```

```

<section class="main-contain dip-style" hidden="hidden">
    <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">Interests</a></li>
                        <li><a data-toggle="tab" href="#size">About</a></li>
                        <li><a data-toggle="tab" href="#custom">Videos</a></li>
                    </ul>
                </div>
            </div>
            <div class="tab-content">
                <!-- Description -->
                <div class="tab-pane fade in active" id="description" runat="server">
                    <asp:TextBox ID="Details" runat="server" Height="100px" Rows="7" style="resize:none"
                    TextMode="MultiLine" MaxLength="250" class="col-lg-9 col-md-9 col-sm-10 pull-left "
                    BorderStyle="None"></asp:TextBox>

                    <%--
                        <p id="about" runat="server">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.

                        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 -->
            </div>
        </div>
    </div>
</section>

```

```

        <!-- Size -->
        <div id="size" class="tab-pane" runat="server">
            <asp:TextBox ID="Interest" runat="server" TextMode="MultiLine" Font-Size="Large"
Text=""></asp:TextBox>
        </div>
        <!-- Size -->

        <!-- Description -->
        <div class="tab-pane media-block " id="custom" runat="server">
            <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">
                        <video width="540" height="360"
src="https://www.youtube.com/embed/Vpg9yizPP_g"></video>
                    </div>
                </div>
            </div>
        </div>
        <!-- End Description -->
    </div>
</div>
</div>
</div></section>
<br />
</asp:Content>
<asp:Content ID="Content6" ContentPlaceHolderID="ContentFooter" Runat="Server">
    <div class="related-products-style-in-productp">
        <div class="container">

            <div class="widget">
                <h4 class="text-uppercase bottom-line">Related Profiles</h4>
            </div>
            <div class="row">
                <div class="new-product slider" id="related" 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>Date</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 Looking for a Man</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>date</li>
        </ul>
        <a href="#" class="product-grid-img-thumbnail">
img" />
        
        </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="#">Man Looking for Dating</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>Wed</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="#">Pair Met thourgh our Portal</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>Date</li>
        </ul>
        <a href="#" class="product-grid-img-thumbnail">
img" />
        
        </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="#">Looking for Dating</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>Date</li>
        </ul>
        <a href="#" class="product-grid-img-thumbnail">

```



```

<!-- 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="#"><i class="fa fa-facebook"></i></a></li>
            <li><a href="#"><i class="fa fa-twitter"></i></a></li>
            <li><a href="#"><i class="fa fa-google-plus"></i></a></li>
            <li><a href="#"><i class="fa fa-twitter"></i></a></li>
            <li><a href="#"><i class="fa fa-google-plus"></i></a></li>
          </ul>
        </div>
      </div>
    </div>
  </div>
</div>
<!--End Suvbscribe -->

```

```

<!-- START FOOTER -->
<footer class="footer" hidden="hidden">
  <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>
```

```
</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>
    </ul>
  </div>
</div>
```

```
        <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 @ My
Laad Bazar</p>

</div>

</div>

</div>

</footer>

<!-- END FOOTER-->

<script type="text/javascript" src="ProfDet/js/jquery.min.js"></script>

<script type="text/javascript" src="ProfDet/js/bootstrap.min.js"></script>

<script type="text/javascript" src="ProfDet/plugins/owl-carousel/owl.carousel.min.js"></script>

<script type="text/javascript" src="ProfDet/plugins/FlexSlider/jquery.flexslider-min.js"></script>

<script type="text/javascript" src="ProfDet/js/jquery.magnific-popup.min.js"></script>

<script type="text/javascript" src="ProfDet/js/product.js"></script>

<script type="text/javascript" src="ProfDet/js/magnific-popup.min.js"></script>

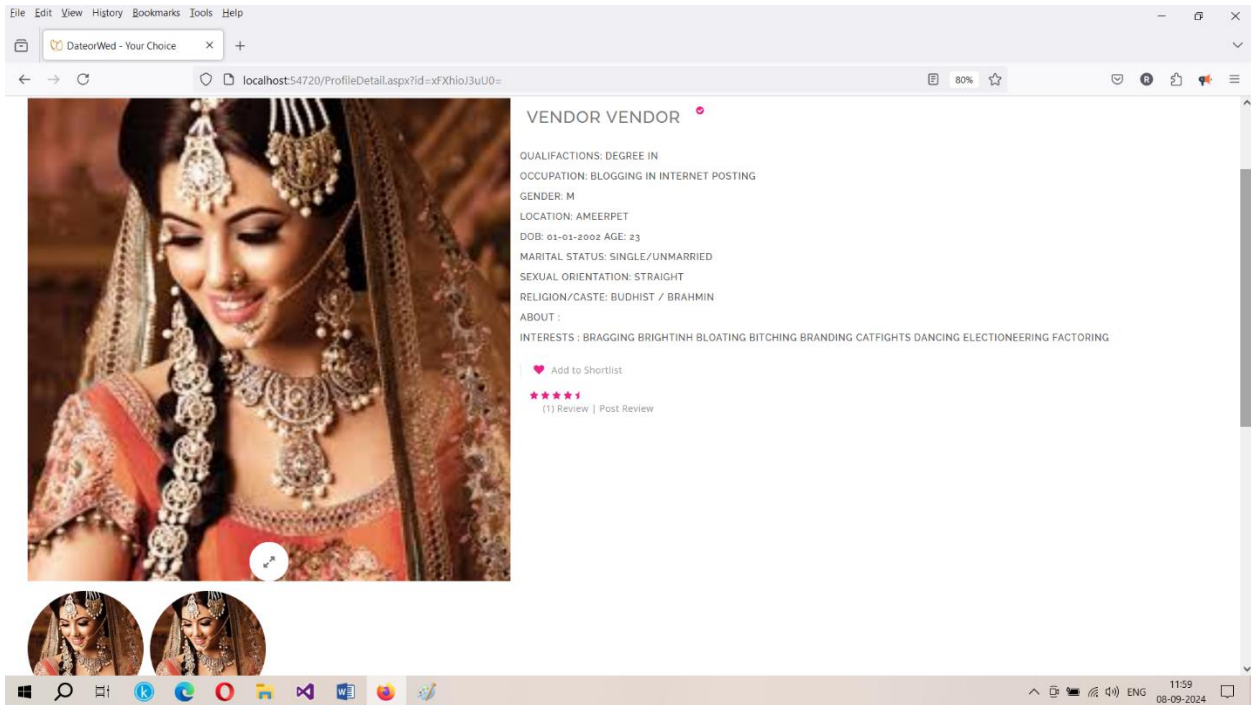
<script type="text/javascript" src="ProfDet/js/slick.js"></script>

<script type="text/javascript" src="ProfDet/js/slick-slider.js"></script>

<script type="text/javascript" src="ProfDet/js/plugins.min.html"></script>

<script type="text/javascript" src="ProfDet/js/custom.js"></script>

</asp:Content>



***** ProfileDetail.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 DOWDal;
using DOWEntity;
using System.Web.Security;
using System.Net;
using System.Web.Script.Serialization;
using System.IO;
using System.Data;

public partial class ProfileDetail : 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)
        {
            bool ret = new Common().guest_login();
            if (ret == false)
```

```

        {
            Response.Redirect("~/Login.aspx");
        }
    }
    //if (!IsPostBack)
    //{
        Bindetail(1, 5, 5);
    //}

}
private void Bindetail(int pageid, int userid, int groupid)
{ int cnt=0;
  string ids =Common.Decrypt(Request.QueryString["id"]);
  int psid=0;
  string epsid = Common.Encrypt(ids);
  string user;
  if (epsid != "0" && epsid != "") { psid = Convert.ToInt16(new tblUsers().getuserid( ids)); }
  StringBuilder sb = new StringBuilder();
  ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
  if (Clsusr == null)
  {
      bool ret = new Common().guest_login();
      if (ret == false)
      {
          Response.Redirect("~/Login.aspx");
      }
  }

  EtetblUser list = new tblUsers().getuserbyid(psid);
  List<EtetblUserImages> igs = new tblUsers().getuserImagesbyuserid(psid, "Profile");
  cnt = igs.Count;
  string path = "";
  if (cnt>0)
  {
      path = igs[0].ImageUrl;
  }

  string path1 = "images/DateOrWed1.png";
  string path2 = null;// "images/No Photo.png";
  string path3 = null;// "images/No Photo.png";
  string path4 = null;//"images/No Photo.png";
  string path5 = null;//"images/No Photo.png";

  if (cnt > 1)
  {
      path1 = igs[1].ImgPath;
  }

```



```

    }
    if (igs.Count() > 2)
    {
        path2 = igs[2].ImagePath;
    }
    if (igs.Count() > 3)
    {
        path3 = igs[3].ImagePath;
    }
    if (igs.Count() > 4)
    {
        path4 = igs[4].ImagePath;
    }
    if (igs.Count() > 5)
    {
        path5 = igs[5].ImagePath;
    }

sb.Append(" <div class='col-md-5 col-sm-6 '>\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> \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");
if (path1 != null)
{
    sb.Append("<li> \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");
}
}
if (path2 != null)
{
    sb.Append("<li> \n");
    sb.Append(" <a class='cd-lightbox-image cd-lightbox-type-image' href="" + path2 + "" title=""> \n");
    sb.Append(" <img title="" src="" + path2 + "" alt="" ><i itemprop='image' class='fa fa-fw fa-
expand'></i> \n");
    sb.Append(" </a> \n");
    sb.Append(" </li> \n");
}
}

```

```

if (path3 != null)
{
    sb.Append("<li> \n");
    sb.Append(" <a class='cd-lightbox-image cd-lightbox-type-image' href='" + path3 + "' title='> \n");
    sb.Append(" <img title=' src='" + path3 + "' alt='><i itemprop='image' class='fa fa-fw fa-
expand'></i> \n");
    sb.Append(" </a> \n");
    sb.Append(" </li> \n");
}
if (path4 != null)
{
    sb.Append("<li> \n");
    sb.Append(" <a class='cd-lightbox-image cd-lightbox-type-image' href='" + path4 + "' title='> \n");
    sb.Append(" <img title=' src='" + path4 + "' alt='><i itemprop='image' class='fa fa-fw fa-
expand'></i> \n");
    sb.Append(" </a> \n");
    sb.Append(" </li> \n");
}
if (path5 != null)
{
    sb.Append("<li> \n");
    sb.Append(" <a class='cd-lightbox-image cd-lightbox-type-image' href='" + path5 + "' title='> \n");
    sb.Append(" <img title=' src='" + path5 + "' alt='><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> \n");
sb.Append(" <img title=' class='img-circle' alt=' src='" + path + "'> \n");
sb.Append(" </li> \n");
sb.Append(" <li> \n");
sb.Append(" <img title=' class='img-circle' alt=' src='" + path1 + "'> \n");
sb.Append(" </li> \n");
sb.Append(" <li> \n");
sb.Append(" <img title=' class='img-circle' alt=' src='" + path2 + "'> \n");
sb.Append(" </li> \n");
sb.Append(" <li> \n");
sb.Append(" <img title=' class='img-circle' alt=' src='" + path3 + "'> \n");
sb.Append(" </li> \n");
sb.Append(" <li> \n");
sb.Append(" <img title=' class='img-circle' alt=' src='" + path4 + "'> \n");

```

```

sb.Append(" </li> \n");
sb.Append("<li> \n");
sb.Append(" <img title=' class='img-circle' alt=' src=' + path5 + "'> \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_Log(4, "Gordersearchlist", "1", "SliderList", PreHtml);

    imglist.InnerHtml = PreHtml;
}

PreHtml = "";
EteTblUser_profile lst = new tblUsers().getuserprofilebyid(psid);
lst = new tblUsers().GetUserProfilePrivacyDetails(userid,psid,lst);
    EteTblUser det = new tblUsers().getuserbyid(psid);
    EteHeartInfo info = new tblUsers().GetHeartInfo( Convert.ToInt16( Clsusr.UserId), psid);
sb = new StringBuilder();
long age = new DateTime(DateTime.Now.Subtract(lst.dob).Ticks).Year;

sb.Append(" <h2 >" + det.FirstName + " " + det.LastName + " " + "</h2> \n");
sb.Append(" <ul class='list-inline product-ratings pull-center ' > \n");
sb.Append(" <li class='product-review-list pull-center' > \n");
sb.Append(" </li> \n");
if (det.Status)
{
    sb.Append("<li> <i class='rating-selected fa fa-check-circle'></i> </li>\n");
}
else
{
    sb.Append(" <i class='rating-selected fa fa-times-circle'></i> \n");
}
}
sb.Append(" </ul><br> \n");
sb.Append(" </div><br> \n");
//sb.Append(" <p class='br15'>" + "" + "<br></p> \n");

sb.Append(" <h3 class='shop-product-title'>" + "Qualifactions: " + lst.Eduqualification + "</h3> \n");
sb.Append(" <h3 class='shop-product-title'>" + "Occupation: " + lst.Occupation + " In " +
lst.Occupationtype + "</h3> \n");

```

```

sb.Append(" <h3 class='shop-product-title'>" + "Gender: " + det.Gender + "</h3> \n");
sb.Append(" <h3 class='shop-product-title'>" + "Location: " + lst.Location + "</h3> \n");
sb.Append(" <h3 class='shop-product-title'>" + "DOB: " + lst.dob.ToShortDateString() + " Age: " + age
+ "</h3> \n");
//sb.Append(" <ul class='list-inline shop-product-prices mb5'> \n");
if (Convert.ToInt16(Clsusr.GroupId) > 3)
{
    sb.Append(" <h3 class='shop-product-title'>Marital Status: " + lst.StrMaritalstatus + "</h3> \n");
    sb.Append(" <h3 class='shop-product-title'>Sexual Orientation: " + lst.StrSexornt + "</h3> \n");
    sb.Append(" <h3 class='shop-product-title'>Religion/Caste: " + lst.StrReligion + " / " + lst.StrCaste +
"</h3> \n");
    if (lst.About != null)
    {
        sb.Append(" <h3 class='shop-product-title'>About : " + lst.About.ToString() + "</h3>
\n");

        sb.Append(" <h3 class='shop-product-title'>Interests : " + lst.Details.ToString() + "</h3>
\n");
        //sb.Append(lst.About.ToString());
        //sb.Append(" <asp:TextBox ID='Int' class='show' visible='true' lines='7' runat='server'
TextMode='MultiLine' Font-Size='Large' Text='Interests: " + lst.Details + "'></asp:TextBox>");
        sb.Append("<br> ");
    }
    if (Convert.ToInt16(Clsusr.GroupId) > 3)
    {
        if (!info.HeartListAdded)
        {
            sb.Append(" <div class='mb30'> \n");
            sb.Append(" <a href='MyHeartList.aspx?pid=" + epsid + "' class='btn btn-product fa fa-
heart'>Add to List</a><br> \n"); //<i class='fa fa-heart'>
            //sb.Append(" <button type='button' class='btn btn-product fa fa-heart' OnClick
='Button1_Click'>Add to List</button> \n");
            sb.Append(" </div> \n");
        }

        sb.Append(" <ul class='list-inline add-to-wishlist add-to-wishlist-brd'> \n");
        if (!info.ShortListAdded)
        {
            sb.Append("<span> </span> <li class='wishlist-in'> \n");
            sb.Append(" <i class='fa fa-heart'></i> \n");
            sb.Append(" <a href='MyShortList.aspx?psid=" + epsid + "'>Add to Shortlist</a> \n");
            sb.Append(" </li><br> \n");
        }
        if (!info.SentProfileAdded)
        {
            sb.Append(" <li class='compare-in'> \n");
            sb.Append(" <i class='fa fa-exchange'></i> \n");
            sb.Append(" <a href='SendProfile.aspx?psid=" + epsid + "'>Send Your Profile</a> \n");
        }
    }
}

```

```

        sb.Append("    </li> \n");
    }
    if (lst.Privacy)
    {
        sb.Append("        <li class='compare-in'> \n");
        sb.Append("        <i class='fa fa-question-circle'></i> \n");
        sb.Append("        <a href='SendRequest.aspx?psid=" + epsid + "'>Ask to Show Data With-held</a> \n");
        sb.Append("        </li> \n");
    }
    sb.Append("    </ul><br> \n");
    sb.Append("    <ul class='list-inline product-ratings pull-left col-lg-6 '> \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-selected fa fa-star '></i></li> \n");
    sb.Append("    <li><i class='rating fa fa-star-half'></i></li> \n");
    sb.Append("    <li><i class='rating fa '></i></li>\n");

    sb.Append("<br> <li class='product-review-list '> \n");
    // sb.Append("    <div class='shop-product-heading col-sm-10'> \n");
    sb.Append("    <span>(1)</span> <a href='#'>Review</a> | <a href='#' > Post Review
</a></span> \n");
    sb.Append("    </li> \n");
    sb.Append("    </ul> \n");
}
}
//size.InnerHtml = lst.Details.ToString();
// description.InnerHtml = lst.Details.ToString();
Details.Text = lst.Details;

    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(Common.Decrypt( Request.QueryString["id"]));
    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 tblUsers().InsertCartList(cart);
    if (ret)
    {
        Response.Redirect("~/MyList.aspx");
    }
}
protected void ImageButton2_Click(object sender, ImageClickEventArgs e)
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];

    int psid = Convert.ToInt16(Common.Decrypt( Request.QueryString["id"]));
    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 tblUsers().InsertShortList(lst, true);

}
public string GetIPAddress()
{
    String address = "";
    WebRequest request = WebRequest.Create("http://checkip.dyndns.org/");
    using (WebResponse response = request.GetResponse())
    using (StreamReader stream = new StreamReader(response.GetResponseStream()))
    {
        address = stream.ReadToEnd();
    }

    int first = address.IndexOf("Address: ") + 9;
    int last = address.LastIndexOf("</body>");
    address = address.Substring(first, last - first);

    return address;
}
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();
List<Location> locations = new List<Location>();
string City;
string Country;
string ipAddress = GetIPAddress();
string APIKey = "744627112f51a720c93d8e11741a02c6f50ad135ea28e733d10f52dc0596b193";
string url = string.Format("http://api.ipinfodb.com/v3/ip-city/?key={0}&ip={1}&format=json",
APIKey, ipAddress);
using (WebClient client = new WebClient())
{
    string json = client.DownloadString(url);
    Location location = new JavaScriptSerializer().Deserialize<Location>(json);
    locations.Add(location);
    City = locations[0].CityName;
    Country = locations[0].CountryName;
}
BALCreateUser objlogin = new BALCreateUser();
DataTable status_login = objlogin.LoginGuest(Common.Encrypt(user), user, ipAddress, "", 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 = new
tblUsers().getusername(Convert.ToInt16(objLoggedInUser.UserId));
//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"]);
objLoggedInUser.FirstName = status_login.Rows[0]["FirstName"].ToString();
objLoggedInUser.LastName = status_login.Rows[0]["LastName"].ToString();
Session["LoggedInUser_Specs"] = objLoggedInUser;

```

```

FormsAuthentication.SetAuthCookie(objLogedInUser.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;
//  string page = new tblUsers().getuserpage(objLogedInUser.UserName);
//}

}
}

}

***** ProfileImages.aspx *****
<%@ Page Title="DateorWed - Your Profile" Language="C#"
MasterPageFile="~/DashBoardMasterPage.master" AutoEventWireup="true"
CodeFile="ProfileImages.aspx.cs" Inherits="ProfileImages" %>

<%@ 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">

</asp:Content>
<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
  <div class="container">
    <div class="row-fluid" style="padding-top: 20px; padding-left:50px">
      <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="Content5" ContentPlaceHolderID="ContentMainPage" Runat="Server">
    <!-- Content Bodey Here -->

    <section class="dip-style ml-5"> <!--main-contain --%>
        <div class="container bg-gray">
            <div class="login-area mb30">

                <div class="tab-content ml-5">
                    <!-- Description -->
                    <div class="tab-pane media-block active" id="Images">
                        <div class=" col-lg-12 col-md-12 col-sm-12 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;"><h3>Images</h3><h4>(1st 2 images are public and others are Private) Add Images
Here</h4></div>
                                    <ul class="col-lg-12 col-md-12 col-sm-12" runat="server">
                                        <li class=" col-lg-2 col-md-2 col-sm-12">
                                            <asp:image ToolTip = "ASP Image Control" ID="Image1" runat="server" ImageUrl
="ManageProducts.aspx?ImageID=1" Height="156px" Visible="false"></asp:image> </li>
                                            <li class=" col-lg-2 col-md-2 col-sm-12">
                                                <asp:image ToolTip = "ASP Image Control" ID="Image6" runat="server" ImageUrl
="ManageProducts.aspx?ImageID=1" Height="156px" Visible="false"></asp:image> </li>
                                                <li class=" col-lg-2 col-md-2 col-sm-12">
                                                    <asp:image ToolTip = "ASP Image Control" ID="Image2" runat="server" ImageUrl
="ManageProducts.aspx?ImageID=1" Height="156px" Visible="false"></asp:image> </li>
                                                    <li class="col-lg-2 col-md-2 col-sm-12">
                                                        <asp:image ToolTip = "ASP Image Control" ID="Image3" runat="server" ImageUrl
="ManageProducts.aspx?ImageID=2" Height="156px" Visible="false"></asp:image>
                                                    </li><li class="col-lg-2 col-md-2 col-sm-12">
                                                        <asp:image ToolTip = "ASP Image Control" ID="Image4" runat="server" ImageUrl
="ManageProducts.aspx?ImageID=3" Height="156px" Visible="false"> </asp:image>
                                                    </li><li class="col-lg-2 col-md-2 col-sm-12">
                                                        <asp:image ToolTip = "ASP Image Control" ID="Image5" runat="server" ImageUrl
="ManageProducts.aspx?ImageID=4" Height="156px" Visible="false"> </asp:image>

                                                    </li>
                                                </li>
                                            </ul>
                                        </div>
                                    </div>
                                </div>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </section>

```

```

<%--    <li class="col-lg-1 col-md-1 col-sm-1" dir="auto">Add Images</li>--%>
<li>
    <asp:FileUpload ID="FileUpload1" runat="server" />
</li><li>
    <asp:Button ID="btn_upload" runat="server" Text="Upload" OnClick="btn_upload_Click"
CausesValidation="false" />
        </li>

</ul>
<%--
        <table>
            <tr>
                <td class="col-lg-2 col-md-2 col-sm-12">
                    <asp:image ToolTip = "ASP Image Control" ID="Image2" runat="server" ImageUrl
="ManageProducts.aspx?ImageID=1" Height="156px" Visible="false"></asp:image> </td>
                <td class="col-lg-2 col-md-2 col-sm-12">
                    <asp:image ToolTip = "ASP Image Control" ID="Image3" runat="server" ImageUrl
="ManageProducts.aspx?ImageID=2" Height="156px" Visible="false"></asp:image>
                </td><td class="col-lg-2 col-md-2 col-sm-12">
                    <asp:image ToolTip = "ASP Image Control" ID="Image4" runat="server" ImageUrl
="ManageProducts.aspx?ImageID=3" Height="156px" Visible="false"> </asp:image>
                </td><td class="col-lg-2 col-md-2 col-sm-12">
                    <asp:image ToolTip = "ASP Image Control" ID="Image5" runat="server" ImageUrl
="ManageProducts.aspx?ImageID=4" Height="156px" Visible="false"> </asp:image>

                </td>
            </tr>

<tr>
    <td class="col-lg-1 col-md-1 col-sm-1" 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_upload_Click"
CausesValidation="false" />
        </td>
    </tr>

        </table>--%>
</div>
<div class="row-fluid">

</div>

```

```

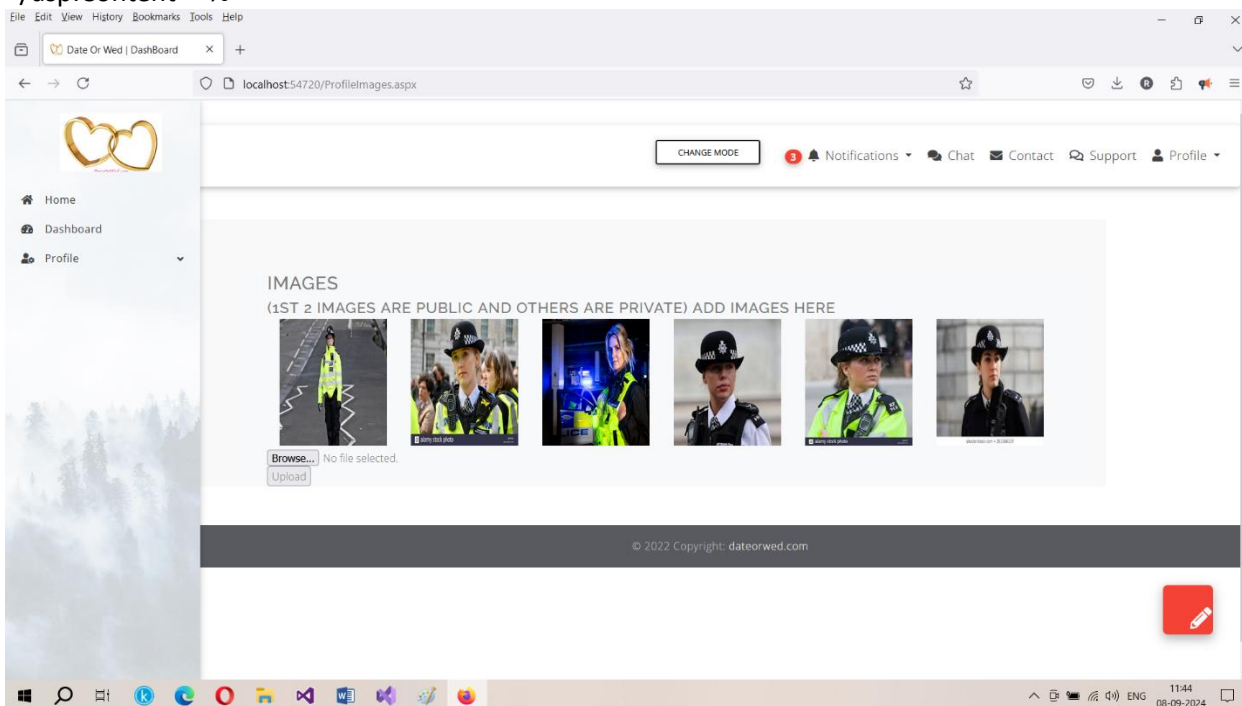
                <!--Notes Ends----->
            </div>
        </div>
    <!-- End Description -->
</div>
</div>
</div>

```

```

</div></section>
</asp:Content>
<%--<asp:Content ID="Content6" ContentPlaceHolderID="ContentFooter" Runat="Server">
</asp:Content>--%>

```



***** ProfileImages.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 DOWEntity;
using DOWDal;
using System.Data;
using System.Data.SqlClient;

```

```

using System.Configuration;
using System.Drawing;
using System.Web.Security;
using System.IO;
using System.Net;

public partial class ProfileImages : System.Web.UI.Page
{
    public string PageType;
    string pageid;
    int UserId;
    int PrId = 0;
    string user;
    DDLogTracer objlog = new DDLogTracer();

    protected void Page_Load(object sender, EventArgs e)
    {
        //dob.DropDownCalendarID = webcal.ID;
        string ip_add = GetIp();
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr == null)
        {
            bool ret = new Common().guest_login();
            if (ret == false)
            {
                Response.Redirect("~/Login.aspx");
            }
        }
        if (Clsusr != null)
        {
            PageType = Page.Request.QueryString["PageType"];
            if (PageType == null)
            {
                PageType = "Edit";
            }
            Session["PageType"] = PageType;
            pageid = "4";
            if (Convert.ToInt16(Clsusr.GroupId) < 4)
            {
                Common.ShowMessageHtml("PI Login to View your Profile-" + Clsusr.UserName + "-" +
                Clsusr.UserId + "-" + Clsusr.GroupId, 0, true, dvMsgBox);
                return;
            }
        }
        //if (!this.IsPostBack)
        {
            PopDlIs ObjPD = new PopDlIs();

```

```

//ddl_area
string ProductId = Page.Request.QueryString["Id"];
UserId = Convert.ToInt16(Clsusr.UserId);
if (UserId > 0)
{
    int srno = 1;
    List<EtetblUserImages> igs = new tblUsers().getuserImagesbyuserid(UserId, "Profile");
    foreach (var imgs in igs)
    {
        if (srno == 1)
        {
            Image1.ImageUrl = imgs.ImageUrl;
        }
        else
            if (srno == 2)
            {
                Image6.ImageUrl = imgs.ImageUrl;
            }
            else
                if (srno == 3)
                {
                    Image2.ImageUrl = imgs.ImageUrl;
                }
                else
                    if (srno == 4)
                    {
                        Image3.ImageUrl = imgs.ImageUrl;
                    }
                    else
                        if (srno == 5)
                        {
                            Image4.ImageUrl = imgs.ImageUrl;
                        }
                        else
                            if (srno == 6)
                            {
                                Image5.ImageUrl = imgs.ImageUrl;
                            }
                    }
            }
        }
    }
//
}

```

```

List<EtetblUserImages> Pics = (new tblUsers()).getuserImagesbyuserid(UserId, "");
foreach (var pic in Pics)

```

```

    {
        switch (pic.srno)
        {
            case 1:
                Image1.ImageUrl = pic.ImageUrl;
                Image1.Visible = true;
                break;
            case 2:
                Image6.ImageUrl = pic.ImageUrl;
                Image6.Visible = true;
                break;
            case 3:
                Image2.ImageUrl = pic.ImageUrl;
                Image2.Visible = true;
                break;
            case 4:
                Image3.ImageUrl = pic.ImageUrl;
                Image3.Visible = true;
                break;
            case 5:
                Image4.ImageUrl = pic.ImageUrl;
                Image4.Visible = true;
                break;
            case 6:
                Image5.ImageUrl = pic.ImageUrl;
                Image5.Visible = true;
                break;
            default:
                break;
        }
    } //for each in pics
}
}
}

protected string GetIp()
{
    String address = "";
    WebRequest request = WebRequest.Create("http://checkip.dyndns.org/");
    using (WebResponse response = request.GetResponse())
    using (StreamReader stream = new StreamReader(response.GetResponseStream()))
    {
        address = stream.ReadToEnd();
    }

    int first = address.IndexOf("Address: ") + 9;
}

```

```

int last = address.LastIndexOf("</body>");
address = address.Substring(first, last - first);

return address;
}

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();
IPEndPoint ipEntry = System.Net.Dns.GetHostEntry(strHostName);
IPAddress[] addr = ipEntry.AddressList;
string ip1 = addr[1].ToString();

BALCreateUser objlogin = new BALCreateUser();
DataTable status_login = objlogin.LoginGuest(Common.Encrypt(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 = new
tblUsers().getusername(Convert.ToInt16(objLoggedInUser.UserId));
//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"]);
objLoggedInUser.FirstName = status_login.Rows[0]["FirstName"].ToString();
objLoggedInUser.LastName = status_login.Rows[0]["LastName"].ToString();
Session["LoggedUser_Specs"] = objLoggedInUser;

FormsAuthentication.SetAuthCookie(objLoggedInUser.UserName, false);
HttpCookie myCookie = new HttpCookie("DOWCookie");
DateTime now = DateTime.Now;
myCookie.Value = objLoggedInUser.UserName;
myCookie.Expires = now.AddYears(1);
}
}

```

```

        Response.Cookies.Add(myCookie);
    }
}
private DataTable GetData(SqlCommand cmd)
{
    DataTable dt = new DataTable();
    String strConnString =
System.Configuration.ConfigurationManager.ConnectionStrings["DOWConectingString"].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();
    }
}
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_saveimage_Click(object sender, System.EventArgs e)
{
    string path = FileUpload1.FileName;
    byte[] Photo = GetPhtoStream();
    string retrpath = "";
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    UserId = Convert.ToInt16(Clsusr.UserId);
}

```



```

EtetblUserImages ep = new EtetblUserImages();

ep.UserId = UserId;

ep.Images = Photo;
retpath = new tblUsers().InsertImages(ep);
string server = Server.MapPath(retpath);
string server1 = Server.MapPath("thumb/" + retpath);
File.WriteAllBytes(@server, Photo);
ReduceImageSize(0.2, server, server1);

List<EtetblUserImages> Pics = (new tblUsers()).getuserImagesbyuserid(UserId, "");
foreach (var pic in Pics)
{
    switch (pic.srno)
    {
        case 1:
            Image1.ImageUrl = pic.ImageUrl;
            Image1.Visible = true;
            break;
        case 2:
            Image6.ImageUrl = pic.ImageUrl;
            Image6.Visible = true;
            break;
        case 3:
            Image2.ImageUrl = pic.ImageUrl;
            Image2.Visible = true;
            break;
        case 4:
            Image3.ImageUrl = pic.ImageUrl;
            Image3.Visible = true;
            break;
        case 5:
            Image4.ImageUrl = pic.ImageUrl;
            Image4.Visible = true;
            break;
        case 6:
            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;
    string retpath;
    retpath = new tblUsers().InsertImages(ep);
    string server = Server.MapPath(retpath);
    string server1 = Server.MapPath("thumb/" + retpath);
    File.WriteAllBytes(@server, Photo);
    ReducelImageSize(0.2, server, server1);

    List<EtetblUserImages> Pics = (new tblUsers()).getuserImagesbyuserid(UserId, "");
    foreach (var pic in Pics)
    {
        switch (pic.srno)
        {
            case 1:
                Image1.ImageUrl = pic.ImageUrl;
                Image1.Visible = true;
                break;
            case 2:
                Image6.ImageUrl = pic.ImageUrl;
                Image6.Visible = true;
                break;
            case 3:
                Image2.ImageUrl = pic.ImageUrl;
                Image2.Visible = true;
                break;
            case 4:
                Image3.ImageUrl = pic.ImageUrl;
                Image3.Visible = true;
                break;
            case 5:

```

```

        Image4.ImageUrl = pic.ImageUrl;
        Image4.Visible = true;
        break;
    case 6:
        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();
    }
}
}

```

```

private void ReduceImageSize(double scaleFactor, string sourcePath, string targetPath)
{
    try
    {
        //using (System.Drawing.Image image = System.Drawing.Image.FromStream(sourcePath))
        using (System.Drawing.Image image = System.Drawing.Image.FromFile(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);
        }
    }
    catch { }
}

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;
    string retpath;
    retpath = new tblUsers().InsertImages(ep);
    string server = Server.MapPath(retpath);
    string server1 = Server.MapPath("thumb/" + retpath);
    File.WriteAllBytes(@server, Photo);
    ReduceImageSize(0.2, server, server1);

    List<EtetblUserImages> Pics = (new tblUsers()).getuserImagesbyuserid(UserId, "");
    foreach (var pic in Pics)
    {
        switch (pic.srno)
    }
}

```

```

    {
        case 1:
            Image1.ImageUrl = pic.ImageUrl;
            Image1.Visible = true;
            break;
        case 2:
            Image6.ImageUrl = pic.ImageUrl;
            Image6.Visible = true;
            break;
        case 3:
            Image2.ImageUrl = pic.ImageUrl;
            Image2.Visible = true;
            break;
        case 4:
            Image3.ImageUrl = pic.ImageUrl;
            Image3.Visible = true;
            break;
        case 5:
            Image4.ImageUrl = pic.ImageUrl;
            Image4.Visible = true;
            break;
        case 6:
            Image5.ImageUrl = pic.ImageUrl;
            Image5.Visible = true;
            break;
        default:
            break;
    }
}

}

}

***** ProfileInfo.aspx *****
<%@ Page Title="DateorWed - Your Profile" Language="C#"
MasterPageFile="~/DashBoardMasterPage.master" AutoEventWireup="true"
CodeFile="ProfileInfo.aspx.cs" Inherits="ProfileInfo" %>

<%@ Register Assembly="Infragistics4.Web.v14.2, Version=14.2.20142.1028, Culture=neutral,
PublicKeyToken=7dd5c3163f2cd0cb" Namespace="Infragistics.Web.UI.EditorControls" TagPrefix="ig" %>
<%@ 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() {
            //alert('Hidepopup()');

```

```

    $find("popup").hide();
    //$find("pnlAddEdit").hide();

    return false;
}
function Hidepopup1() {
    alert("Hidepopup1()");
    $find("popup").hide();
    //$find("pnlAddEdit").hide();
    return false;
}
</script>
<script>
$(function () {
    $("#popup").dialog({
        resizable: false,
        autoOpen: false,
        height: 350,
        modal: true,
        buttons: {
            "OK": function () {
                $(this).dialog("close");
            },
            Cancel: function () {
                $(this).dialog("close");
            }
        }
    });

    $("#InkFake").click(function () {
        $("#popup").dialog("open");
    });
});
</script>

```

```

</asp:Content>
<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
    <div class="container">
        <div class="row-fluid" style="padding-top: 1px; padding-left:50px">
            <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>
</div>
</asp:Content>
<asp:Content ID="Content5" ContentPlaceHolderID="ContentMainPage" Runat="Server">
    <!-- Content Bodey Here -->

    <section class=" dip-style"> <!--main-contain --%>
        <div class="container bg-gray">
            <div class=" mb30"><!--login-area--%>

                <div class="tab-content ml-5">
                    <!-- Profile -->
                    <div id="Profile" class="tab-pane active ">
                        <div class="col-lg-12 col-md-12 col-sm-12" style="background-color: #f8f9fa;">
                            <div class="row-fluid">

                                <!--Profile Account----->
                                <div class="row-fluid">
                                    <div class="col-lg-10 col-md-10 col-sm-12" style="text-align: left;">
                                        <asp:Button ID="Button4" runat="server" CssClass="btn btn-primary" Text="Save
Profile" OnClick="Button4_Click"/>
                                        <h3>Profile(Public-Check for Private)</h3></div>
                                    </div>
                                    <div class="row-fluid"> <!--table-detail--%>
                                        <div class="col-lg-10 col-md-10 col-sm-12" style="background-color: #ffffff;
border:1px solid #eefef; border-top: none; padding: 5px;"> <!--border: 0px solid #eefef;--%>
                                            <table class="table table-striped col-lg-8 col-md-8 col-sm-12 ">
                                                <tbody >
                                                    <!--
                                                    <tr >
                                                        <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Your Name :</td>
                                                        <td>
                                                            <asp:DropDownList ID="ddl_users" runat="server"></asp:DropDownList>
                                                        </td>
                                                    </tr>--%>
                                                    <tr>
                                                        <td class="col-lg-4 col-md-4 col-sm-3" dir="auto">Your Interests :</td>
                                                        <td>
                                                            <asp:TextBox ID="Details" runat="server" Height="100px" Rows="7"
style="resize:both" TextMode="MultiLine" MaxLength="250" CssClass="col-lg-9 col-md-9 col-sm-10 pull-
left"></asp:TextBox>
                                                        </td><td>
                                                            <!--
                                                            <asp:CheckBox ID="Details_chk" CssClass="show" runat="server"
Checked="false" /> --%>
                                                        <div class="form-check" id="d_Details_chk" runat="server">
<input

```

```

        class="form-check-input"
        type="checkbox"
        value=""
        id="Details_chk" runat="server"
    />
    <label class="form-check-label" for="Details_chk">

</label>
</div>

    </td>

</tr>
<tr>
    <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">About Yourself :</td>
    <td>
        <asp:TextBox ID="txt_about" runat="server" Height="100px" Rows="7"
style="resize:both" TextMode="MultiLine" CssClass="col-lg-9 col-md-9 col-sm-10 pull-
left"></asp:TextBox>
    </td><td>
        <!-- <asp:CheckBox ID="about_chk" runat="server" />--%>
        <div class="form-check" id="d_about_chk" runat="server">
        <input
        class="form-check-input"
        type="checkbox"
        value=""
        id="about_chk" runat="server"
        />
        <label class="form-check-label" for="about_chk">

</label>
</div>
    </td>

</tr>

<tr>
    <td class="col-lg-3 col-md-3 col-sm-3 auto-style1"
dir="auto">Occupation</td>
    <td class="auto-style2">
        <asp:TextBox ID="txt_Occupation" runat="server" MaxLength="40"
CssClass="col-lg-9 col-md-9 col-sm-10 pull-left"></asp:TextBox>
    </td><td>
        <!--<asp:CheckBox ID="Occupation_chk" runat="server" />--
%>

        <div class="form-check" id="d_Occupation_chk" runat="server">
        <input
        class="form-check-input"
        type="checkbox"

```



```

        value=""
        id="Occupation_chk" runat="server"
    />
    <label class="form-check-label" for="Occupation_chk">

</label>
</div>
</td>
</tr>
<tr>
    <td class="col-lg-3 col-md-3 col-sm-3 auto-style1" dir="auto">Area of
Occupation</td>
    <td class="auto-style2">
        <asp:TextBox ID="txt_Occupation_type" runat="server" MaxLength="40"
CssClass="col-lg-9 col-md-9 col-sm-10 pull-left"></asp:TextBox>
</td><td>
        <!--<asp:CheckBox ID="Occupation_type_chk" runat="server"

        <div class="form-check" id="d_Occupation_type_chk" runat="server">
<input
class="form-check-input"
type="checkbox"
value=""
id="Occupation_type_chk" runat="server"
/>
<label class="form-check-label" for="Occupation_type_chk">

</label>
</div>
</td>
</tr>
<tr>
    <td class="col-lg-3 col-md-3 col-sm-3 auto-style1" dir="auto">Education
Qualification</td>
    <td class="auto-style2">
        <asp:TextBox ID="txt_Edu_qualification" runat="server" MaxLength="40"
CssClass="col-lg-9 col-md-9 col-sm-10 pull-left"></asp:TextBox>
</td><td>
        <!-- <asp:CheckBox ID="Edu_qualification_chk" runat="server" />-->
        <div class="form-check" id="d_Edu_qualification_chk" runat="server">
<input
class="form-check-input"
type="checkbox"
value=""
id="Edu_qualification_chk" runat="server"
/>
<label class="form-check-label" for="Edu_qualification_chk">

</label>

```

```

</td>
</div>
</tr>
<tr>
<td class="col-lg-3 col-md-3 col-sm-3" dir="auto">School Attended:</td>
<td>
<asp:DropDownList ID="ddl_school" CssClass="show" runat="server">

</asp:DropDownList> </td>
<td>
<div class="form-check" id="d_school_chk" runat="server">
<input class="form-check-input"
type="checkbox"
value=""
id="school_chk" runat="server"
/>
<label class="form-check-label" for="school_chk">

</label>
</div>

</td><td>
<asp:LinkButton ID="lnkFake" runat="server"></asp:LinkButton>
<cc1:ModalPopupExtender ID="popup" runat="server" DropShadow="false"
PopupControlID="pnlAddEdit" TargetControlID="lnkFake"
BackgroundCssClass="modalBackground">
</cc1:ModalPopupExtender>

<asp:Button ID="Button1" CssClass="btn btn-primary btn-rounded" runat="server" Text="Add"
OnClick="Button1_Click" />
<!--<asp:CheckBox ID="CheckBox2" runat="server" />--%>

</td>
</tr>
<tr>
<td class="col-lg-3 col-md-3 col-sm-3" dir="auto">College Attended:</td>
<td>
<asp:DropDownList ID="ddl_college" CssClass="show" runat="server">

</asp:DropDownList> </td>
<td>
<div class="form-check" id="d_college_chk" runat="server">
<input class="form-check-input"
type="checkbox"
value=""
id="college_chk" runat="server"
/>

```

```

        <label class="form-check-label" for="college_chk">

        </label>
    </div>

</td><td>
    <asp:Button ID="Button2" CssClass="btn btn-primary btn-rounded" runat="server" Text="Add"
OnClick="Button2_Click" />
    <%--<asp:CheckBox ID="CheckBox2" runat="server" />--%>

        </td>
    </tr>
    <tr>
        <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">University
Attended:</td>

        <td>
            <asp:DropDownList ID="ddl_university" CssClass="show"
runat="server">

                </asp:DropDownList> </td>
            <td>
                <div class="form-check" id="d_university_chk" runat="server">
                <input class="form-check-input"
                type="checkbox"
                value=""
                id="university_chk" runat="server"
                />
                <label class="form-check-label" for="university_chk">

                </label>
            </div>

        </td><td>
            <asp:Button ID="Button3" CssClass="btn btn-primary btn-reddit" runat="server" Text="Add"
OnClick="Button3_Click"/>
            <%--<asp:CheckBox ID="CheckBox2" runat="server" />--%>

                </td>
            </tr>

    <tr>
    <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Date of Birth :</td>
    <td>

    <ig:WebDatePicker ID="dob" runat="server" DisplayModeFormat="dd-MM-yyyy" EditModeFormat="dd-
MM-yyyy" Nullable="false" MinValue="1950-01-01" MaxValue="2002/01/01" CausesValidation="true">

```

```

</ig:WebDatePicker>
<!-- <ig:WebMonthCalendar ID="webcal" CssClass=" bg-success"
runat="server"></ig:WebMonthCalendar> --%>
<asp:TextBox ID="txt_yrest" runat="server" MaxLength="20" Visible="false"></asp:TextBox>
</td>
<td>
<!--<asp:CheckBox ID="dob_chk" runat="server" Checked="true" />--%>
<div class="form-check" id="d_dob_chk" runat="server">
<input
class="form-check-input"
type="checkbox"
value=""
id="dob_chk" runat="server"
/>
<label class="form-check-label" for="dob_chk">

</label>
</div>

</td>

</tr>
<tr>
<td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Time of Birth :</td>
<td>
<!--<ig:WebDateTimeEditor ID="WebDateTimeEditor1" runat="server" DisplayModeFormat="HH:mm"
MinValue="00:00">

</ig:WebDateTimeEditor>--%>
<asp:TextBox ID="txt_tob" runat="server" MaxLength="40" ></asp:TextBox>
</td>
<td >
<!-- <asp:CheckBox ID="tob_chk" runat="server" Checked="true" Visible="true" />--%>
<div class="form-check" id="d_tob_chk" runat="server">
<input
class="form-check-input"
type="checkbox"
value=""
id="tob_chk" runat="server"
/>
<label class="form-check-label" for="tob_chk">

</label>
</div>

</td>

</tr>
<tr>

```

```

<td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Place of Birth :</td>
<td>
  <asp:TextBox ID="txt_pob" runat="server" MaxLength="40" ></asp:TextBox>
</td>
  <!--<asp:CheckBox ID="pob_chk" runat="server" />--%>
  <div class="form-check" id="d_pob_chk" runat="server">
    <input
      class="form-check-input"
      type="checkbox"
      value=""
      id="pob_chk" runat="server"
    />
    <label class="form-check-label" for="pob_chk">

    </label>
  </div>

</td>
</tr>

  <tr>
    <td class="col-lg-3 col-md-3 col-sm-3 " dir="auto">Marital Status:</td>
    <td>
      <asp:DropDownList ID="ddl_Martial_status" CssClass=" show"
runat="server">
  <asp:ListItem Value="0">-- Select Your Martial Status -- </asp:ListItem>
  <asp:ListItem Value="1">Single/Unmarried</asp:ListItem>
  <asp:ListItem Value="2">Single/Divorced(no children)</asp:ListItem>
  <asp:ListItem Value="3">Single/Divorced(with children)</asp:ListItem>
  <asp:ListItem Value="4">Single/Widowed(no children)</asp:ListItem>
  <asp:ListItem Value="5">Single/Widowed(with children)</asp:ListItem>
  <asp:ListItem Value="6">Single/Seperated(no children)</asp:ListItem>
  <asp:ListItem Value="7">Single/Seperated(with children)</asp:ListItem>

  </asp:DropDownList>

</td>

    <td>
      <!--<asp:CheckBox ID="Martial_status_chk" runat="server" />--%>
      <div class="form-check" id="d_Martial_status_chk" runat="server">
        <input
          class="form-check-input"
          type="checkbox"
          value=""
          id="Martial_status_chk" runat="server"
        />
        <label class="form-check-label" for="Martial_status_chk">

```

```

        </label>
    </div>

    </td>
</tr>
<tr>
    <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Sexual
Orientation:</td>
    <td>
        <asp:DropDownList ID="ddl_Sex_ornt" CssClass="show" runat="server">
<asp:ListItem Value="0">-- Select Your Orientation --</asp:ListItem>
<asp:ListItem Value="1">Straight </asp:ListItem>
<asp:ListItem Value="2">Gay</asp:ListItem>
<asp:ListItem Value="3">Bi-Sexual</asp:ListItem>

        </asp:DropDownList>
</td><td>
        <%--<asp:CheckBox ID="Sex_ornt_chk" runat="server" />--%>
<div class="form-check" id="d_Sex_ornt_chk" runat="server">
<input
    class="form-check-input"
    type="checkbox"
    value=""
    id="Sex_ornt_chk" runat="server"
/>
<label class="form-check-label" for="Sex_ornt_chk">

    </label>
</div>

    </td>
</tr>
<tr>
    <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Religion:</td>
    <td>
        <asp:DropDownList ID="ddl_Religion" CssClass="show" runat="server">

        </asp:DropDownList>
</td><td>
        <%--<asp:CheckBox ID="Religion_chk" runat="server" />--%>
<div class="form-check" id="d_Religion_chk" runat="server">
<input
    class="form-check-input"
    type="checkbox"
    value=""
    id="Religion_chk" runat="server"
/>
<label class="form-check-label" for="Religion_chk">

```

```

        </label>
    </div>

    </td>
</tr>
<tr>
    <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Caste:</td>
    <td>
        <asp:DropDownList ID="ddl_caste" CssClass="show" runat="server">

            </asp:DropDownList>
            <%--<asp:CheckBox ID="Caste_chk" CssClass="show"
</td><td>
runat="server" />--%>
        <div class="form-check" id="d_Caste_chk" runat="server">
            <input
            class="form-check-input"
            type="checkbox"
            value=""
            id="Caste_chk" runat="server"
            />
            <label class="form-check-label" for="Caste_chk">

        </label>
    </div>

    </td>
</tr>
<tr>
    <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Sub-Caste:</td>
    <td>
        <asp:TextBox ID="ddl_subcaste" runat="server" MaxLength="40"
        CssClass="col-lg-9 col-md-9 col-sm-10 pull-left"></asp:TextBox>
        <%--
            <asp:DropDownList ID="ddl_subcaste" runat="server">

            </asp:DropDownList>--%>
    </td>
    <td>
        <%--<asp:CheckBox ID="Sub_Caste_chk" runat="server" />--%>
        <div class="form-check" id="d_Sub_Caste_chk" runat="server">
            <input
            class="form-check-input"
            type="checkbox"
            value=""
            id="Sub_Caste_chk" runat="server"
            />
            <label class="form-check-label" for="Sub_Caste_chk">

```

```

        </label>
    </div>

    </td>
</tr>

<tr>
    <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">No of Family
Members</td>
    <td>
        <asp:TextBox ID="txt_family_nos" runat="server" MaxLength="40"
></asp:TextBox>
        </td><td>
            <%--<asp:CheckBox ID="family_nos_chk" runat="server" />--
%>

            <div class="form-check" id="d_family_nos_chk" runat="server">
            <input
            class="form-check-input"
            type="checkbox"
            value=""
            id="family_nos_chk" runat="server"
            />
            <label class="form-check-label" for="family_nos_chk">

            </label>
            </div>
        </td>

    </tr>
<tr>
    <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Web-URL</td>
    <td>
        <asp:TextBox ID="txt_web_url" runat="server" MaxLength="40"
></asp:TextBox>
        </td><td>
            <%--<asp:CheckBox ID="web_url_chk" runat="server" />--%>
            <div class="form-check" id="d_web_url_chk" runat="server">
            <input
            class="form-check-input"
            type="checkbox"
            value=""
            id="web_url_chk" runat="server"
            />
            <label class="form-check-label" for="web_url_chk">

            </label>
            </div>

```



```

        </td>

</tr>
<tr>
    <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Income Per Year:

    </td>

    <td >
        <asp:TextBox ID="txt_Income_year" runat="server" MaxLength="40"
        CssClass="col-lg-3 col-md-3 col-sm-3" ></asp:TextBox>
        <asp:Label ID="curr" runat="server" CssClass="col-lg-3 col-md-3
        col-sm-3" align="right" dir="right">Currency:</asp:Label>
        <asp:DropDownList ID="ddl_currency" runat="server" CssClass="col-lg-3
        col-md-3 col-sm-3 show" ></asp:DropDownList>
    </td><td>
        <%--<asp:CheckBox ID="Income_Year_chk" runat="server" />--
        %>

        <div class="form-check" id="d_Income_Year_chk" runat="server">
        <input
        class="form-check-input"
        type="checkbox"
        value=""
        id="Income_Year_chk" runat="server"
        />
        <label class="form-check-label" for="Income_Year_chk">

    </label>
</div>

    </td>

</tr>
<tr>
    <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Location :</td>
    <td>
        <asp:TextBox ID="txt_Loc" runat="server" MaxLength="40"
    ></asp:TextBox>

    </td>
    <td>
        <%--<asp:CheckBox ID="Location_chk" runat="server" />--%>
        <div class="form-check" id="d_Location_chk" runat="server">
        <input
        class="form-check-input"
        type="checkbox"
        value=""
        id="Location_chk" runat="server"
        />
        <label class="form-check-label" for="Location_chk">

```

```
</label>
</div>
```

```
</td>
</tr>
```

```
<tr>
  <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Weekly working days
:</td>
```

```
  <td>
    <asp:TextBox ID="txt_weekly_days" runat="server" MaxLength="40" CssClass="col-lg-3 col-md-3
col-sm-3" ></asp:TextBox>
  </td><td>
    <%--<asp:CheckBox ID="weekly_days_chk" runat="server" />--
%>
```

```
  <div class="form-check" id="d_weekly_days_chk" runat="server">
  <input
  class="form-check-input"
  type="checkbox"
  value=""
  id="weekly_days_chk" runat="server"
  />
  <label class="form-check-label" for="weekly_days_chk">
```

```
</label>
</div>
```

```
<%-- <asp:Label ID="tet" runat="server" class="col-lg-3 col-md-3 col-sm-3 " align="right" dir="auto">In
Unit of Measure:</asp:Label>--%>
```

```
</td>
</tr>
```

```
<tr><td>Public Images</td><td></td><td></td></tr>
```

```
  <tr>
    <td>
      <asp:image ToolTip = "ASP Image Control" ID="Image1" runat="server" ImageUrl
="images/nocolor.png" Height="156px" Width="174px"></asp:image> </td>
    <td>
      <asp:image ToolTip = "ASP Image Control" ID="Image6" runat="server" ImageUrl
="images/nocolor.png" Height="156px" Width="174px"></asp:image>
```

```

</td>
<td>
  <%--<asp:CheckBox ID="Age_to_chk" runat="server" />--%>
    <div class="form-check" id="d_Age_to_chk" runat="server">
      <input
        class="form-check-input"
        type="checkbox"
        value=""
        id="Age_to_chk" runat="server"
      />
      <label class="form-check-label" for="Age_to_chk">

      </label>
    </div>
  </td>
</tr>
</tbody>

</table>

</div>
<!--Bank Account END-->
</div>

</div>
</div>
<!--ModalPageStarts-->
<asp:Panel ID="pnlAddEdit" runat="server" CssClass="modalPopup" Style="display: block;">

  <div class="row-fluid">
    <div class="col-sm-8" style="background-color: #ffffff; border: 1px solid #eefef; border-
top: none; padding: 5px;">
      <%-- panel table removed--%>
    <div class="row-fluid">
      <div id="popup_head" class="col-sm-8 customer" style="text-align: left;"
runat="server">Add School to list</div>
    </div>
    <table id="Table9" class="table table-striped table-detail col-lg-4 col-sm-12"
runat="server">

    <tbody>

```

```

        <tr>
            <td align="right" id="namescu" class="label_txt" dir="auto"
runat="server">College Name:</td>
            <td>
                <asp:TextBox ID="txt_name" runat="server"></asp:TextBox>
            </td>
        </tr>
        <tr>
            <td width="14%" align="right" class="label_txt" dir="auto">City:</td>
            <td>
                <asp:DropDownList ID="ddl_City" CssClass="show"
runat="server"></asp:DropDownList>
            </td>
        </tr>
        <tr>
            <td width="14%" align="right" class="label_txt" dir="auto">Country:</td>
            <td>
                <asp:DropDownList ID="ddl_country" CssClass="show"
runat="server"></asp:DropDownList>
            </td>
        </tr>
        <tr>
            <td align="right" class="label_txt" dir="auto">Zipcode/Pincode:</td>
            <td>
                <asp:TextBox ID="txt_zipcode" runat="server" ></asp:TextBox>
            </td>
        </tr>
        <tr>
            <td>
                <asp:Button ID="btnSave" runat="server" Text="Save" OnClick="btnSave_Click"
CssClass="btn btn-primary" />
            </td>
            <td>
                <asp:Button ID="btnCancel" runat="server" Text="Cancel"
OnClick="return Hidepopup()" CssClass="btn btn-primary" />
            </td>
        </tr>
    </tbody>

```

```

        </table>

    </div>
</div>

</asp:Panel>

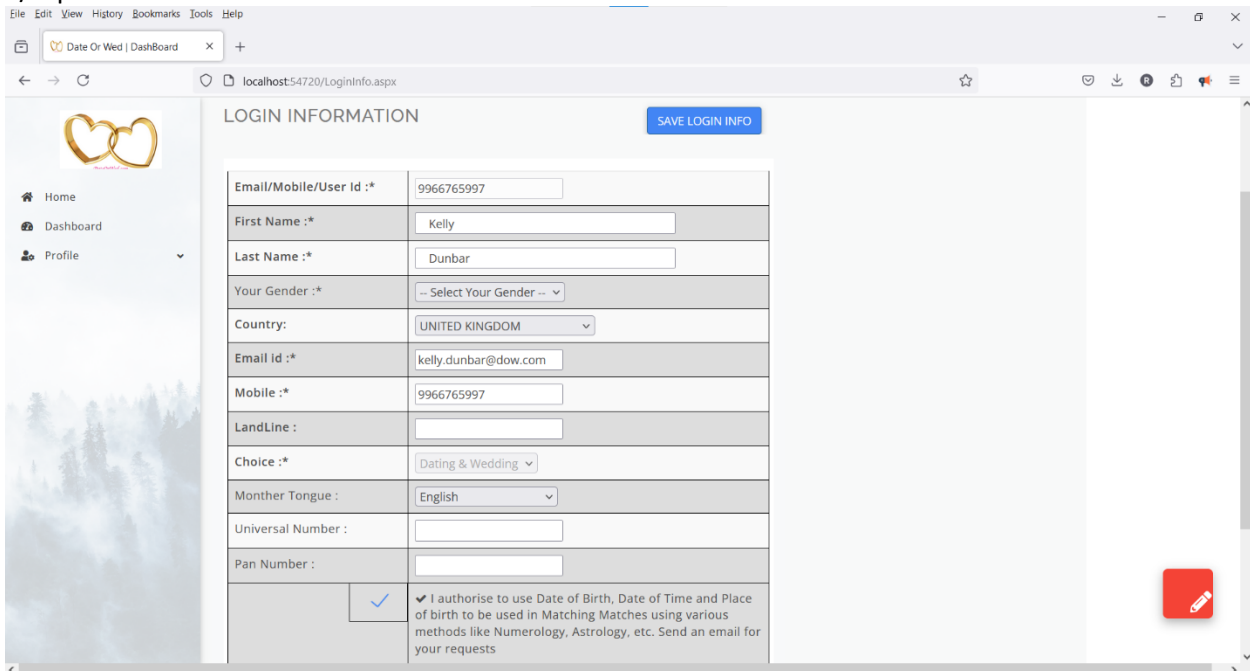
<!--ModalPageEnds----->
</div>
</div>

</div>

</section>

<script>
    // Material Select Initialization
    $(document).ready(function () {
        $(' .mdb-select').materialSelect();
    });
</script>
</asp:Content>
<%--<asp:Content ID="Content6" ContentPlaceHolderID="ContentFooter" Runat="Server">
</asp:Content>--%>

```



***** ProfileInfo.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 DOWEntity;
using DOWDal;
using System.Data;
using System.Data.SqlClient;
using System.Configuration;
using System.Drawing;
using System.Web.Security;
using System.IO;
using System.Net;

public partial class ProfileInfo : System.Web.UI.Page
{
    public string PageType;
    string pageid;
    int UserId;
    int Prid = 0;
    string user;
    DDLogTracer objlog = new DDLogTracer();
    PopDlls ObjPD = new PopDlls();
    public static int reset = 0;

    protected void Page_Load(object sender, EventArgs e)
    {
        //dob.DropDownCalendarID = webcal.ID;
        string ip_add = GetIp();
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr == null)
        {
            bool ret = new Common().guest_login();
            if (ret == false)
            {
                Response.Redirect("~/Login.aspx");
            }
        }

        if (Clsusr != null)
        {
            PageType = Page.Request.QueryString["PageType"];
            if (PageType == null)
            {
                PageType = "Edit";
            }
        }
    }
}

```

```

    }
    Session["PageType"] = PageType;
    pageid = "4";
    if (Convert.ToInt16(Clsusr.GroupId) < 4)
    {
        Common.ShowMessageHtml("PI Login to View your Profile-" + Clsusr.UserName + "-" +
Clsusr.UserId + "-" + Clsusr.GroupId, 0, true, dvMsgBox);
        return;
    }
    //    if (!this.IsPostBack)
    {
        //Session["CAN_SUSP"] = "School";

        ObjPD.BindDropDowns(ddl_Religion, (new PopDlls()).GetReligions());
        ObjPD.BindDropDowns(ddl_caste, (new PopDlls()).GetCastes());
        ObjPD.BindDropDowns(ddl_currency, (new PopDlls()).GetCurrency());
        ObjPD.BindDropDowns(ddl_college, (new PopDlls()).GetCollegesList());
        ObjPD.BindDropDowns(ddl_university, (new PopDlls()).GetUniverssityList());
        ObjPD.BindDropDowns(ddl_school, (new PopDlls()).GetSchoosList());
        string ProductId = Page.Request.QueryString["Id"];
        if (reset == 0)
        {
            BindDetails();
        }
    }
}
protected void BindDetails()
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    if (Clsusr == null)
    {
        bool ret = new Common().guest_login();
        if (ret == false)
        {
            Response.Redirect("~/Login.aspx");
        }
    }
    reset = 1;
    UserId = Convert.ToInt16(Clsusr.UserId);
    if (UserId > 0)
    {
        //-----user infor

        Etetbluser_profile Prd = new tblUsers().getuserprofilebyid(UserId);
        if (Prd != null && Prd.userid > 0)
        {

```

```

Details.Text = Prd.Details;
txt_Occupation.Text = Prd.Occupation;
txt_Occupation_type.Text = Prd.Occupationtype;
txt_Edu_qualification.Text = Prd.Eduqualification;
ddl_Martial_status.SelectedIndex = Prd.Martialstatus;
ddl_Sex_ornt.SelectedIndex = Prd.Sexornt;
dob.Date = Prd.dob.Date;
txt_yrest.Text = Prd.dob.Date.ToShortDateString();
txt_tob.Text = Prd.tob.ToString();
txt_pob.Text = Prd.pob;
txt_family_nos.Text = Prd.familynos.ToString();
txt_web_url.Text = Prd.weburl;
txt_Income_year.Text = Prd.Income_year.ToString();
ddl_currency.SelectedIndex = Prd.currency;
txt_Loc.Text = Prd.Location;
ddl_college.SelectedIndex = Prd.College;
ddl_school.SelectedIndex = Prd.School;
ddl_university.SelectedIndex = Prd.University;
txt_about.Text = Prd.About;
ddl_Religion.SelectedIndex = Prd.Religion;
ddl_caste.SelectedIndex = Prd.Caste;
}

```

```

int srno = 1;
List<EtetblUserImages> igs = new tblUsers().getuserImagesbyuserid(UserId, "Profile");
foreach (var imgs in igs)
{
    if (srno == 1)
    {
        Image1.ImageUrl = imgs.ImageUrl;
    }
    else
    {
        if (srno == 2)
        {
            Image6.ImageUrl = imgs.ImageUrl;
        }
    }
}
//
}

```

```

Etetbluser_privacy_chk data = new Etetbluser_privacy_chk();
Details_chk.Checked = data.Details_chk;
//d_Details_chk.InnerHtml=SetChk("Details_chk",data.Details_chk);

```



```

about_chk.Checked = data.about_chk;
Occupation_chk.Checked = data.Occupation_chk;
//d_Occupation_chk.InnerHtml=SetChk("Occupation_chk",data.Occupation_chk);
Occupation_type_chk.Checked = data.Occupation_type_chk;
d_Occupation_type_chk.InnerHtml = SetChk("Occupation_type_chk", data.Occupation_type_chk);
Edu_qualification_chk.Checked = data.Edu_qualification_chk;
d_Edu_qualification_chk.InnerHtml = SetChk("Edu_qualification_chk", data.Edu_qualification_chk);
Martial_status_chk.Checked = data.Martial_status_chk;
d_Martial_status_chk.InnerHtml = SetChk("Martial_status_chk", data.Martial_status_chk);
Sex_ornt_chk.Checked = data.Sex_ornt_chk;
d_Sex_ornt_chk.InnerHtml = SetChk("Sex_ornt_chk", data.Sex_ornt_chk);
dob_chk.Checked = data.dob_chk;
d_dob_chk.InnerHtml = SetChk("dob_chk", data.dob_chk);
tob_chk.Checked = data.tob_chk;
d_tob_chk.InnerHtml = SetChk("tob_chk", data.tob_chk);
pob_chk.Checked = data.pob_chk;
d_pob_chk.InnerHtml = SetChk("pob_chk", data.pob_chk);
family_nos_chk.Checked = data.family_nos_chk;
d_family_nos_chk.InnerHtml = SetChk("family_nos_chk", data.family_nos_chk);
web_url_chk.Checked = data.web_url_chk;
d_web_url_chk.InnerHtml = SetChk("web_url_chk", data.web_url_chk);
weekly_days_chk.Checked = data.weekly_days_chk;
d_weekly_days_chk.InnerHtml = SetChk("weekly_days_chk", data.weekly_days_chk);
Income_Year_chk.Checked = data.Income_Year_chk;
d_Income_Year_chk.InnerHtml = SetChk("Income_Year_chk", data.Income_Year_chk);
Religion_chk.Checked = data.Religion_chk;
d_Religion_chk.InnerHtml = SetChk("Religion_chk", data.Religion_chk);
Caste_chk.Checked = data.Caste_chk;
d_Caste_chk.InnerHtml = SetChk("Caste_chk", data.Caste_chk);
Sub_Caste_chk.Checked = data.Sub_Caste_chk;
d_Sub_Caste_chk.InnerHtml = SetChk("Sub_Caste_chk", data.Sub_Caste_chk);
Location_chk.Checked = data.Location_chk;
d_Location_chk.InnerHtml = SetChk("Location_chk", data.Location_chk);

```

```
List<EtetblUserImages> Pics = (new tblUsers()).getuserImagesbyuserid(UserId, "");
```

```
foreach (var pic in Pics)
{
```

```
    switch (pic.srno)
```

```
    {
```

```
        case 1:
```

```
            Image1.ImageUrl = pic.ImageUrl;
```

```
            Image1.Visible = true;
```

```
            break;
```

```
        case 2:
```

```
            Image6.ImageUrl = pic.ImageUrl;
```

```
            Image6.Visible = true;
```

```
            break;
```

```

        default:
            break;
    }
} //for each in pics
}

protected string SetChk(string id, bool val)
{

    StringBuilder sb = new StringBuilder();
    string chkval = "";
    if (val) { chkval = "checked"; };
    sb.Append("<input class='form-check-input' type='checkbox' value=" id="" + id + ""
runat='server'" + chkval + " />");
    sb.Append("<label class='form-check-label' for="" + id + "">");
    return sb.ToString();
}
protected string GetIp()
{
    String address = "";
    WebRequest request = WebRequest.Create("http://checkip.dyndns.org/");
    using (WebResponse response = request.GetResponse())
    using (StreamReader stream = new StreamReader(response.GetResponseStream()))
    {
        address = stream.ReadToEnd();
    }

    int first = address.IndexOf("Address: ") + 9;
    int last = address.LastIndexOf("</body>");
    address = address.Substring(first, last - first);

    return address;
}

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();
    IPEndPoint ipEntry = System.Net.Dns.GetHostEntry(strHostName);
    IPAddress[] addr = ipEntry.AddressList;
    string ip1 = addr[1].ToString();

    BALCreateUser objlogin = new BALCreateUser();

```

```

        DataTable status_login = objlogin.LoginGuest(Common.Encrypt(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 = new
tblUsers().getusername(Convert.ToInt16(objLoggedInUser.UserId));
            //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"]);
            objLoggedInUser.FirstName = status_login.Rows[0]["FirstName"].ToString();
            objLoggedInUser.LastName = status_login.Rows[0]["LastName"].ToString();
            Session["LoggedUser_Specs"] = objLoggedInUser;

            FormsAuthentication.SetAuthCookie(objLoggedInUser.UserName, false);
            HttpCookie myCookie = new HttpCookie("DOWCookie");
            DateTime now = DateTime.Now;
            myCookie.Value = objLoggedInUser.UserName;
            myCookie.Expires = now.AddYears(1);

            Response.Cookies.Add(myCookie);
        }
    }
    private DataTable GetData(SqlCommand cmd)
    {
        DataTable dt = new DataTable();
        String strConnString =
System.Configuration.ConfigurationManager.ConnectionStrings["DOWConectingString"].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();
    }
}

```

```

private void ContentSaveControlClicked()
{

```

```

    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    UserId = Convert.ToInt32(Clsusr.UserId);
    try
    {

        if (Details.Text == "")
        {
            Common.ShowMessageHtml("Enrer you interests", 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.ToInt16( Clsusr.UserId);
            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..");
}

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 Button4_Click(object sender, EventArgs e)
{//update profile details
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"]; ;
    UserId = Convert.ToInt32(Clsusr.UserId);
    string updooption = "Edit";
    try
    {

        if (Details.Text == "")
        {
            Common.ShowMessageHtml("Enrer you interests", 0, true, dvMsgBox);

        }

        else
        {
            Etetbluser_profile Prd = new tblUsers().getuserprofilebyid(Convert.ToInt16(Clsusr.UserId));
            if (Prd == null)
            {
                Prd = new Etetbluser_profile();
                updooption = "Add";

            }

            Prd.userid = Convert.ToInt16(Clsusr.UserId);
            Prd.Details = Details.Text;
            Prd.About = txt_about.Text;
            Prd.Occupation = txt_Occupation.Text;
            Prd.Occupationtype = txt_Occupation_type.Text;
            Prd.Eduqualification = txt_Edu_qualification.Text;
            Prd.Martialstatus = ddl_Martial_status.SelectedIndex;
            Prd.Sexornt = Convert.ToChar(ddl_Sex_ornt.SelectedIndex);
            Prd.dob = Convert.ToDateTime(dob.Text);
            Prd.tob = txt_tob.Text;
            Prd.pob = txt_pob.Text;
            Prd.familynos = Convert.ToInt16(txt_family_nos.Text);
            Prd.weburl = txt_web_url.Text;

```

```

Prd.Income_year = Convert.ToDecimal(txt_Income_year.Text);
Prd.currency = ddl_currency.SelectedIndex;
Prd.Location = txt_Loc.Text;
//Prd.adhaarid = txt_adhaar_id.Text;
//Prd.panno = txt_pan_no.Text;
if (txt_weekly_days.Text == "") { txt_weekly_days.Text = "0"; }
Prd.weeklydays = Convert.ToInt16(txt_weekly_days.Text);
Prd.GeoLocation = "";
Prd.logopath = "";
Prd.Religion = ddl_Religion.SelectedIndex;
Prd.Caste = ddl_caste.SelectedIndex;
Prd.Sub_Caste = ddl_subcaste.Text;
Prd.College = ddl_college.SelectedIndex;
Prd.School = ddl_school.SelectedIndex;
Prd.University = ddl_university.SelectedIndex;
bool Result = false;

Prd.userid = Convert.ToInt16(Clsusr.UserId);
Prd.UpdatedOn = DateTime.Now;
Prd.CreatedOn = DateTime.Now;

if (updooption == "Add")
{
    Result = new tblUsers().InsertProfile(Prd);
}
else
{
    Result = new tblUsers().UpdateProfile(Prd);
}

if (Result == true)
{
    if (updooption == "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");
}
Etetbluser_privacy_chk data = new Etetbluser_privacy_chk();
data.userid = UserId;
data.User_privacy = 0;

```

```

data.Details_chk = Details_chk.Checked;
data.Occupation_chk = Occupation_chk.Checked;
data.Occupation_type_chk = Occupation_type_chk.Checked;
data.Edu_qualification_chk = Edu_qualification_chk.Checked;
data.Martial_status_chk = Martial_status_chk.Checked;
data.Sex_ornt_chk = Sex_ornt_chk.Checked;
data.dob_chk = dob_chk.Checked;
data.tob_chk = tob_chk.Checked;
data.pob_chk = pob_chk.Checked;
data.family_nos_chk = family_nos_chk.Checked;
data.web_url_chk = web_url_chk.Checked;
data.weekly_days_chk = weekly_days_chk.Checked;
data.Income_Year_chk = Income_Year_chk.Checked;
data.Religion_chk = Religion_chk.Checked;
data.Caste_chk = Caste_chk.Checked;
data.Sub_Caste_chk = Sub_Caste_chk.Checked;
data.Location_chk = Location_chk.Checked;
data.Age_to_chk = Age_to_chk.Checked;
data.GeoLoc_search_chk = false;// GeoLocation_chk.Checked;
data.Religions_Choice = false;// Religions_Choice.Checked;
data.Caste_Choice = false;// Caste_Choice.Checked;
data.Areas_Choice = false;// Areas_Choice.Checked;
data.Cities_Choice = false;// Cities_Choice.Checked;
data.Countries_Choice = false;// Countries_Choice.Checked;
data.Languages_Choice = false;// Languages_Choice.Checked;
data.uid = UserId;
bool ret = false;
ret = new tblUsers().GetPrivacyCount(UserId, 0);
if (ret == false)
{
    data.UpdatedOn = DateTime.Now;
    data.CreatedOn = DateTime.Now;
    ret = new tblUsers().InsertUserPrivacyCheck(data);
}
else
{
    data.UpdatedOn = DateTime.Now;
    ret = new tblUsers().UpdatePrivacyCheck(data);
}
if (ret)
{
    reset = 0;
    Common.ShowMessageHtml("Profile Data Updated Sucessfully", 1, true, dvMsgBox);
    //RefreshScreen();
}
}
}

```



```

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..");
//RefreshScreen();
}

protected void Button1_Click(object sender, EventArgs e)
{
    //Common.ShowMessageHtml("School button clicked", 0, true, dvMsgBox);
    popup_head.InnerHtml = "Enter School Details";
    namescu.InnerText = "School Name";
    ObjPD.BindDropDowns(ddl_City, (new PopDlls()).GetCities());
    ObjPD.BindDropDowns(ddl_country, (new PopDlls()).GetCountries());
    txt_name.Text = string.Empty;
    Session["CAN_SUSP"] = "School";
    popup.Show();

}

protected void Button2_Click(object sender, EventArgs e)
{
    popup_head.InnerHtml = "Enter College Details";
    namescu.InnerText = "College Name";
    ObjPD.BindDropDowns(ddl_City, (new PopDlls()).GetCities());
    ObjPD.BindDropDowns(ddl_country, (new PopDlls()).GetCountries());
    txt_name.Text = string.Empty;
    Session["CAN_SUSP"] = "College";
    popup.Show();

}

protected void Button3_Click(object sender, EventArgs e)
{
    popup_head.InnerHtml = "Enter University Details";
    namescu.InnerText = "University Name";
    ObjPD.BindDropDowns(ddl_City, (new PopDlls()).GetCities());
    ObjPD.BindDropDowns(ddl_country, (new PopDlls()).GetCountries());
    txt_name.Text = string.Empty;
    Session["CAN_SUSP"] = "University";
    popup.Show();

}

protected void btnSave_Click(object sender, EventArgs e)
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    if (Clsusr == null)

```

```

{
    bool ret = new Common().guest_login();

}
int rets=0;
//save school college university details
string savetype = Session["CAN_SUSP"].ToString();
EteEdInst edata =new EteEdInst();
edata.Name = txt_name.Text;
edata.City = ddl_City.SelectedIndex;
edata.Country = ddl_country.SelectedIndex;
edata.ZipCode = txt_zipcode.Text;
edata.CreatedON = DateTime.Now;
edata.UserId =Convert.ToInt16( Clsusr.UserId);
if (savetype == "School")
{
    rets = new EdInstitutions().InsertSchoolsList(edata);
} else if(savetype == "College")
{
    rets = new EdInstitutions().InsertCollegesList(edata);

}
else if (savetype == "University")
{

    rets = new EdInstitutions().InsertUniversityList(edata);
}
if (rets>0)
{
    ObjPD.BindDropDowns(ddl_college, (new PopDlIs()).GetCollegesList());
    ObjPD.BindDropDowns(ddl_university, (new PopDlIs()).GetUniverssityList());
    ObjPD.BindDropDowns(ddl_school, (new PopDlIs()).GetSchoosList());
    if (savetype == "School")
    {ddl_school.SelectedIndex=rets;
    }
    else if (savetype == "College")
    {ddl_college.SelectedIndex=rets;
    }
    else if (savetype == "University")
    {
        ddl_university.SelectedIndex = rets;
    }

    Common.ShowMessageHtml(savetype + " Data Saved", 1, true, dvMsgBox);
    txt_name.Text = "";

}
popup.Hide();

```

```

        //ObjPD.BindDropDowns(ddl_college, (new PopDlls()).GetCollegesList());
        //ObjPD.BindDropDowns(ddl_university, (new PopDlls()).GetUniverssityList());
        //ObjPD.BindDropDowns(ddl_school, (new PopDlls()).GetSchoosList());
    }
    //Common.ShowMessageHtml(Txt_Description.Text, 1, true, dvMsgBox);

}
***** ProfileSettings.aspx *****
<%@ Page Title="DateorWed - Your Profile" Language="C#"
MasterPageFile="~/DashBoardMasterPage.master" AutoEventWireup="true"
CodeFile="ProfileSettings.aspx.cs" Inherits="ProfileSettings" %>

<%@ 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">

</asp:Content>
<asp:Content ID="Content3" ContentPlaceHolderID="CPH_TopMessages" runat="Server">
    <div class="container">
        <div class="row-fluid" style="padding-top: 20px; padding-left:50px">
            <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="Content5" ContentPlaceHolderID="ContentMainPage" Runat="Server">
    <!-- Content Bodey Here -->

    <section class=" dip-style ml-5"> <%--main-contain--%>
        <div class="container bg-gray">
            <div class="login-area mb30">

                <div class="tab-content ml-5">
                    <!-- Description -->

```

```

<!-- Settings -->
<div class="tab-pane media-block active" id="Choices">
    <div class="row-fluid">
        <div id="Div1" runat="server" class=" col-lg-12 col-md-12 col-sm-12 " span3="" style="">
            <asp:Button ID="Button1" runat="server" CssClass="btn btn-primary btn-md btn-rounded"
Text="Save Settings" OnClick="Button1_Click" />
<%-- <a class="navbar-brand" href="UserChoices.aspx">
    <p>Other Search Options</p>

</a>--%>
    <a href="UserChoices.aspx" class="btn btn-primary btn-md btn-rounded"><i class="fas fa-
search"></i> &nbsp;&nbsp;&nbsp;Other Search Options</a>

        <div class="row-fluid">
            <h3>Choices/Setting(Private)</h3></div>
            <table id="Table1" class="table table-striped table-detail col-lg-12 col-md-12 col-sm-12 bg-
secondary" runat="server" visible="true">
                <tbody>
                    <tr>
                        <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Searching for :</td>
                        <td>
                            <asp:DropDownList ID="ddl_choice" CssClass="show" runat="server">
                                <asp:ListItem Value="0">-- Select Your Choice --</asp:ListItem>
                                <asp:ListItem Value="W">Women</asp:ListItem>
                                <asp:ListItem Value="M">Men</asp:ListItem>
                                <asp:ListItem Value="B">Bi-Sexuals</asp:ListItem>
                                <asp:ListItem Value="O">Others</asp:ListItem>

                            </asp:DropDownList>
                        </td>
                    <tr>
                        <td>
                            <asp:CheckBox ID="Gender_Search_chk" CssClass="show" runat="server" />
                        </td>
                    </tr>
                    <tr>
                        <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Age Search (Between) :</td>
                        <td>
                            <asp:TextBox ID="Age_min" runat="server" MaxLength="100" Width="150px"
Text="21"></asp:TextBox>
                            <asp:TextBox ID="Age_max" runat="server" MaxLength="100" Width="150px"
Text="25"></asp:TextBox>
                        </td>
                    </tr>
                    <tr>
                        <td>
                            <asp:CheckBox ID="Age_from_chk" runat="server" />
                        </td>
                    </tr>
                </tbody>
            </table>

```

```
</tr>
<tr>
<td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Religion Search:</td>

<td>
<asp:DropDownList ID="ddl_Religion_search" CssClass="show" runat="server">
</asp:DropDownList>

</td>
<td>
<asp:CheckBox ID="Religion_search_chk" CssClass="show" runat="server" />
</td>
</tr>
<tr>
<td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Caste Search:</td>

<td>
<asp:DropDownList ID="ddl_caste_search" CssClass="show" runat="server">
</asp:DropDownList>

</td>
<td>
<asp:CheckBox ID="Caste_search_chk" CssClass="show" runat="server" />
</td>
</tr>
<tr>
<td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Area Search :</td>

<td>
<asp:DropDownList ID="ddl_area" CssClass="show" runat="server">
</asp:DropDownList>

</td>
<td>
<asp:CheckBox ID="Area_search_chk" CssClass="show" runat="server" />
</td>
</tr>
<tr>
<td class="col-lg-3 col-md-3 col-sm-3" dir="auto">City Search :</td>

<td>
<asp:DropDownList ID="ddl_city" runat="server">
</asp:DropDownList>

</td>
</tr>
</tr>
```

```

        <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Geolocation :</td>
        <td>
            <asp:TextBox ID="Geolocation" runat="server" MaxLength="100"
Width="250px"></asp:TextBox>
        </td>
    </tr>
    <tr>
        <td align="right" class="label_txt"><strong>Country Search:</strong></td>
        <td width="25%" align="left">
            <asp:DropDownList ID="ddl_countrysearch"
runat="server"></asp:DropDownList></td>
    </tr>
--%>
    <tr>
        <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Search by Mother Tongue :</td>
        <td>
            <asp:DropDownList ID="ddl_mtonguesearch" CssClass="show"
runat="server"></asp:DropDownList>
        </td>
        <td>
            <asp:CheckBox ID="MTongue_search_chk" CssClass="show" runat="server" />
        </td>
    </tr>
<%--
    <tr>
        <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Search by Languages Known :</td>
        <td>
            <asp:DropDownList ID="ddl_languages" runat="server"></asp:DropDownList>
        </td>
    </tr>
--%>
</tbody>
</table>
</div>
</div>
</div>
</div>
</div>
</div>

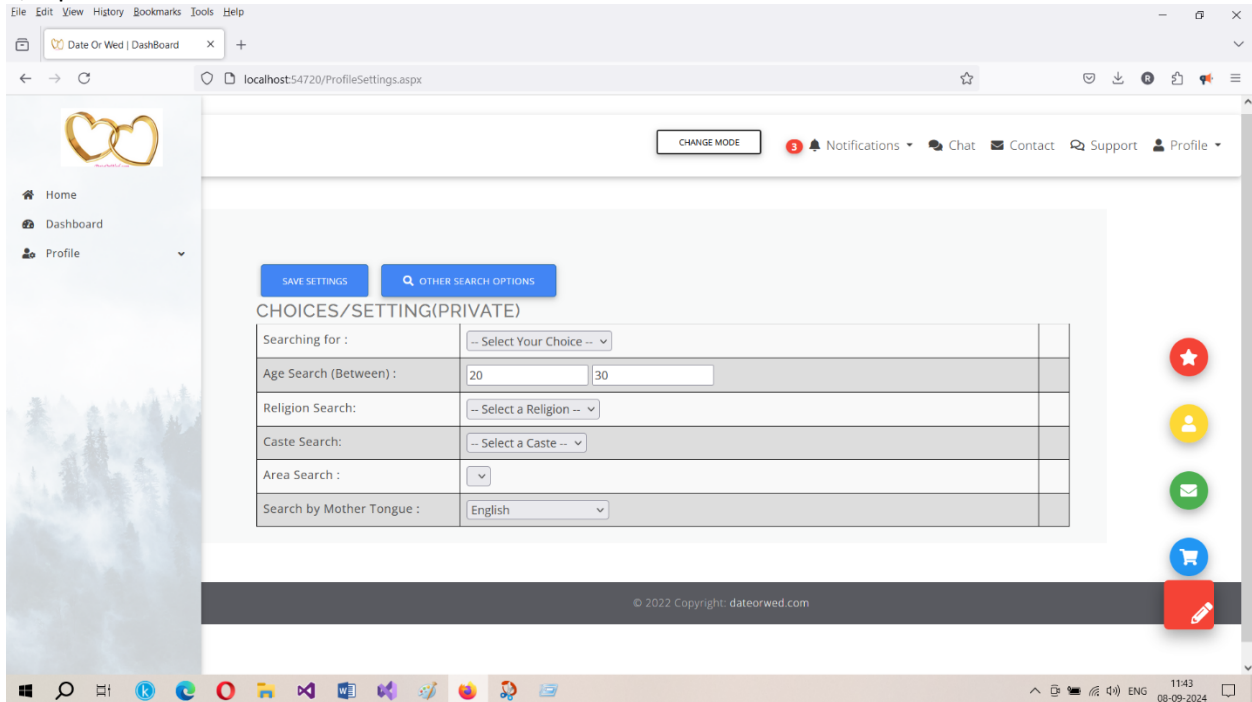
```

</div></section>

</asp:Content>

<%--<asp:Content ID="Content6" ContentPlaceHolderID="ContentFooter" Runat="Server">

</asp:Content>--%>



***** ProfileSettings.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 DOWEntity;
using DOWDal;
using System.Data;
using System.Data.SqlClient;
using System.Configuration;
using System.Drawing;
using System.Web.Security;
using System.IO;
using System.Net;
```

```
public partial class ProfileSettings : System.Web.UI.Page
{
    public string PageType;
    string pageid;
```

```

int UserId;
int Prld = 0;
string user;
DDLogTracer objlog = new DDLogTracer();

protected void Page_Load(object sender, EventArgs e)
{
    //dob.DropDownCalendarID = webcal.ID;
    string ip_add = GetIp();
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    if (Clsusr == null)
    {
        bool ret = new Common().guest_login();
        if (ret == false)
        {
            Response.Redirect("~/Login.aspx");
        }
    }

    if (Clsusr != null)
    {
        PageType = Page.Request.QueryString["PageType"];
        if (PageType == null)
        {
            PageType = "Edit";
        }

        Session["PageType"] = PageType;
        pageid = "4";
        if (Convert.ToInt16( Clsusr.GroupId) <4)
        {
            Common.ShowMessageHtml("PI Login to View your Profile-"+Clsusr.UserName+"-
"+Clsusr.UserId+"-"+Clsusr.GroupId, 0, true, dvMsgBox);
            return;
        }
        //if (!this.IsPostBack)
        // {
        //     PopDlIs ObjPD = new PopDlIs();
        //     ddl_GroupId.Enabled = false;
        //     ObjPD.BindDropDowns(ddl_mtonguesearch, (new PopDlIs()).GetLanguages());
        //     ObjPD.BindDropDowns(ddl_Religion_search, (new PopDlIs()).GetReligions());
        //     ObjPD.BindDropDowns(ddl_caste_search, (new PopDlIs()).GetCastes());
        //     ObjPD.BindDropDowns(ddl_mtonguesearch, (new PopDlIs()).GetLanguages());

        //ddl_area
        string ProductId = Page.Request.QueryString["Id"];
        UserId = Convert.ToInt16(Clsusr.UserId);
        Etetbluser_options opt = new tblUsers().getuserchoies(UserId);
    }
}

```



```

if (opt != null)
{
    ddl_choice.SelectedValue = opt.Gender_Search;
    Age_min.Text = opt.Age_from.ToString();
    Age_max.Text = opt.Age_to.ToString();
    //ddl_Religion.SelectedIndex = opt.Religion_search;
    //ddl_caste.SelectedIndex = opt.Caste_search;
    ddl_area.SelectedIndex = opt.Area_search;
    ddl_mtonguesearch.SelectedIndex = opt.MTongue_search;

}
else
{
    //ddl_choice.SelectedIndex = 0;
    //Age_min.Text = "21";
    //Age_max.Text = "25";
    //ddl_Religion.SelectedIndex = 0;
    //ddl_caste.SelectedIndex = 0;
    //ddl_area.SelectedIndex = 0;

}

```

```

    Etetbluser_privacy_chk data = new Etetbluser_privacy_chk();
    Religion_search_chk.Checked = data.Religion_search_chk;
    Caste_search_chk.Checked = data.Caste_search_chk;
    Area_search_chk.Checked = data.Area_search_chk;
    MTongue_search_chk.Checked = data.MTongue_search_chk;
}
}

```

```

protected string GetIp()
{
    String address = "";
    WebRequest request = WebRequest.Create("http://checkip.dyndns.org/");
    using (WebResponse response = request.GetResponse())
    using (StreamReader stream = new StreamReader(response.GetResponseStream()))
    {
        address = stream.ReadToEnd();
    }

    int first = address.IndexOf("Address: ") + 9;
    int last = address.LastIndexOf("</body>");
    address = address.Substring(first, last - first);

    return address;
}

```

```

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();
    IPEndPoint ipEntry = System.Net.Dns.GetHostEntry(strHostName);
    IPAddress[] addr = ipEntry.AddressList;
    string ip1 = addr[1].ToString();

    BALCreateUser objlogin = new BALCreateUser();
    DataTable status_login = objlogin.LoginGuest(Common.Encrypt(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 = new
tblUsers().getusername(Convert.ToInt16(objLoggedInUser.UserId));
        //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"]);
        objLoggedInUser.FirstName = status_login.Rows[0]["FirstName"].ToString();
        objLoggedInUser.LastName = status_login.Rows[0]["LastName"].ToString();
        Session["LoggedUser_Specs"] = objLoggedInUser;

        FormsAuthentication.SetAuthCookie(objLoggedInUser.UserName, false);
        HttpCookie myCookie = new HttpCookie("DOWCookie");
        DateTime now = DateTime.Now;
        myCookie.Value = objLoggedInUser.UserName;
        myCookie.Expires = now.AddYears(1);

        Response.Cookies.Add(myCookie);
    }
}
private DataTable GetData(SqlCommand cmd)
{

```

```

        DataTable dt = new DataTable();
        String strConnString =
System.Configuration.ConfigurationManager.ConnectionStrings["DOWConectingString"].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();
        }
    }

```

```

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 Button1_Click(object sender, EventArgs e)
{
    //save settings or choices
    Etetbluser_options opt = new tblUsers().getuserchoies(UserId);
    if (opt == null)
    {
        opt = new Etetbluser_options();
    }
    {
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        UserId = Convert.ToInt32(Clsusr.UserId);
        opt.userid = UserId;
        opt.Gender_Search = ddl_choice.SelectedValue;
        opt.Age_from = Convert.ToInt16( Age_min.Text);
        opt.Age_to = Convert.ToInt16(Age_max.Text);
        //opt.Religion_search = ddl_Religion.SelectedIndex;
        //opt.Caste_search = ddl_caste.SelectedIndex;
        opt.Area_search = ddl_area.SelectedIndex;
        opt.MTongue_search = ddl_mtonguesearch.SelectedIndex;
        bool res = new tblUsers().UpdateUser_options(opt);
    }
}

```

```

}
}

```

***** RcvdProfile.aspx *****

```

<%@ Page Title="" Language="C#" MasterPageFile="~/DOWMasterPage.master"
AutoEventWireup="true" CodeFile="RcvdProfile.aspx.cs" Inherits="RcvdProfile" %>

```

```

<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">

```

```

            <div class=" col-lg-12 col-md-12 col=col-sm-12 fa-heart-half">

```

```

                <div class="row">

```

```

        <div id="dvMsgBox" class="span12 ordersave" runat="server" style="display: block;">
            <asp:Label ID="Label1" runat="server" Text="" Height="1px"></asp:Label>
        </div>
    </div>
</div>
</div>
</div>
</div>
</asp:Content>
<asp:Content ID="Content4" ContentPlaceHolderID="CPH_MainContent" Runat="Server">
    <div class="content_holder">

        <!-- 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">Profiles Received</h6>
                </div>
                <div class="row">
                    <div id="NewProducts" class="new-product slider" runat="server">

                        </div>
                    </div>
                </div>
            </div>
        </section>

        <!-- 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>
        <script type="text/javascript" src="plugins/FlexSlider/jquery.flexslider-min.js"></script>
        <script type="text/javascript" src="js/jquery.magnific-popup.min.js"></script>
        <script type="text/javascript" src="js/magnific-popup.min.js"></script>
        <script type="text/javascript" src="js/slick.js"></script>
        <script type="text/javascript" src="js/slick-slider.js"></script>
        <script type="text/javascript" src="js/custom.js"></script>

    </div>

</asp:Content>
<asp:Content ID="Content5" ContentPlaceHolderID="ContentPlaceHolder1" Runat="Server">
</asp:Content>
<asp:Content ID="Content6" ContentPlaceHolderID="ContentFooter" Runat="Server">

```

```

<!-- START FOOTER -->
<footer class="footer" id="footer" runat="server">
  <div class="container">
    <div class="row">
<section class="home-main-contant-style3 bg-white">
  <div class="parallax-block parallaxBg">
    <div class="container">
      <div class="parallax-block-text">
        <p>DateorWed.com - YourChoice <span class="color-green">We are a social networking
site.</span> <br> Thank you! </p>
      </div>
      <div class="pre-btn-group">
        <a class="btn btn-product mr15" href="#">Search Now</a>
        <a class="btn btn-product mr15" href="#">Post a Profile</a>
      </div>
    </div>
  </div>
</section>
<!-- End Parallax -->

```

```

<!-- START NEW -->
<section class="home-main-contant-style3 bg-white" style="display:none">
  <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>
  </div>

```

```

        <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>
</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="fab 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="#"><i class="fab fa-facebook"></i></a></li>
                        <li><a href="#"><i class="fab fa-twitter"></i></a></li>
                        <li><a href="#"><i class="fab fa-tumblr"></i></a></li>
                        <li><a href="#"><i class="fab fa-instagram"></i></a></li>
                        <li><a href="#"><i class="fab fa-linkedin"></i></a></li>
                    </ul>
                </div>
            </div>
        </div>
    </div>
</div>
<!--End Suvbscribe -->
<!-- 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 an Agent 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="#">dating</a></li>
```

```
<li><a href="#">Dating & Wedding</a></li>
```

```
<li><a href="#">Wedding</a></li>
```

```
<li><a href="#">Option</a></li>
```

```
<li><a href="#">Dating or Wedding</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="#">Dating <span class="pull-right">12</span></a></li>
```

```
<li><a href="#">Wedding <span class="pull-right">11</span></a></li>
```

```
<li><a href="#">Date or Wed <span class="pull-right">10</span></a></li>
```

```
<li><a href="#">Date and Wed <span class="pull-right">14</span></a></li>
```

```
<li><a href="#">Others <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 Profiles</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 Profiles</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="#">Looking for</a></h6>
          <div class="widget-posts-meta">Wed</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="#">Searching for</a></h6>
          <div class="widget-posts-meta">-Date-</div>
        </div>
      </li>
      <li>
        <div class="widget-posts-image">
          <a href="#"></a>
        </div>
      </li>
    </ul>
  </div>
</div>
```

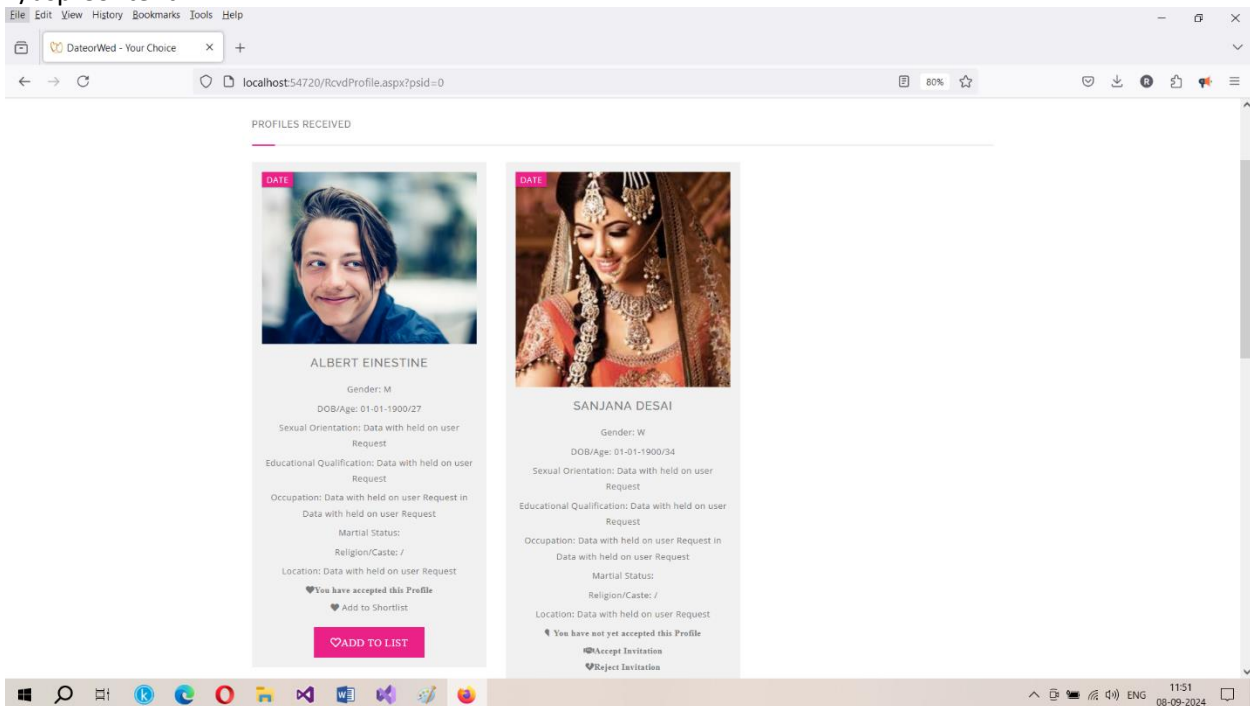
```
</div>
<div class="widget-posts-body">
  <h6 class="widget-posts-title"><a href="#">Date or Wed</a></h6>
  <div class="widget-posts-meta">-Wed-</div>
</div>
</li>
</ul>
</div>
</div>
<!-- END RECENT POSTS WIDGET -->
```

```
</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:Content>
```



```

***** RcvdProfile.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 DOWEntity;
using System.IO;
public partial class RcvdProfile : System.Web.UI.Page
{
    DDLogTracer objlog = new DDLogTracer();
    bool IsMobile = false;
    protected void Page_Load(object sender, EventArgs e)
    {
        // string SrchStrng = "";
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr == null)
        {
            bool ret = new Common().guest_login();
            if (ret == false)
            {
                Response.Redirect("~/Login.aspx");
            }
        }
        if (Clsusr != null)
        {
            string psid = (Request.QueryString["psid"]);
            { Bindetails(psid); }
        }
    }

    private void Bindetails(string SrchStrng)
    {
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"]; ;
        int sno = 0;
        string cat = "";
        string PreHtml = "";
        StringBuilder sb = new StringBuilder();

        cat = Common.Decrypt(Request.QueryString["cat"]).Trim();
        // int len = cat.Length;
        // cat = cat.Substring(0, len - 15);
        // cat = cat.TrimEnd('\0');
    }
}

```

```

//      sb.Append("<div class='directory-list'>");
Etetbluser_options opt = new Etetbluser_options();

//try
{
    List<EteGetSearchResults> Lst = (new
tblUsers()).GetRcvdProfilebyUserId(Convert.ToInt16(Clsusr.UserId));
    if (Lst.Count > 0)
    {
        var pst = (from c in Lst orderby c.UserId descending select c).FirstOrDefault();
        Lst=(from c in Lst orderby c.UserId descending select c).ToList();
        if (Lst != null && Lst.Count > 0)
        {

            //IList<EteGetSearchResults> Lst1 = new List<EteGetSearchResults>();

            foreach (var dt in Lst)
            {
                string path = "images/NoPhoto.png";
                string path1 = "images/No Photo.png";
                string href = "";
                string OutId = Common.Encrypt(dt.UserName.ToString());
                href = "ProfileDetail.aspx?id=" + OutId;
                var list = new tblUsers().getuserImagesbyuserid((int)dt.UserId, "").ToList();
                int cnt = list.Count();
                if (cnt > 1)
                {
                    path = list[0].ImgPath;
                    path1 = list[1].ImgPath;//"http://dateorwed.com/"+
                }
                else
                {
                    if (list.Count() > 0)
                    {
                        path = list[0].ImageUrl;
                    }

                    if (path == null || path == "")
                    {
                        path = "images/Nocolor.png";
                    }
                    if (File.Exists(path))
                    {
                        path = list[0].ImageUrl;
                        path1 = list[1].ImageUrl;
                    }
                    if (path1 == null || path1 == "") { path1 = path; }
                }
            }
        }
    }
}

```

```

string martsttus = new tblUsers().getmatstatus(dt.Martialstatus);
string relgn = new tblUsers().getreligion(dt.Religion);
string caste = new tblUsers().getcaste(dt.Caste);
//slick code
sb.Append("      <div class='col-lg-3 col-md-4 item'>      \n");
sb.Append("      <div class='products grid-product'>      \n");
sb.Append("      <div class='product-grid-inner'>\n");
sb.Append("      <div class='product-grid-img'> \n");
sb.Append("      <ul class='product-labels'>      \n");
sb.Append("      <li>Date</li> \n");
sb.Append("      </ul> \n");
sb.Append("      <a href='ProfileDetail.aspx?id=" + OutId + "' class='product-grid-img-
thumbnail'> \n");
sb.Append("      <img src="" + path + "" alt='img' class='img-responsive primary-img' />
\n");
sb.Append("      <img src="" + path1 + "" alt='img' class='img-responsive socendary-img'
/> \n");
sb.Append("      </a> \n");
sb.Append("      <div class='product-grid-atb text-center'><a class="
href='ProfileDetail.aspx?id=" + OutId + "'>Detailed View</a></div>      \n");
sb.Append("      </div> \n");
sb.Append("      <div class='product-grid-caption text-center'> \n");
sb.Append("      <h4><a class='p-grid-title' href='ProfileDetail.aspx?id=" + OutId + "'> +
dt.FirstName + " " + dt.LastName + "</a></h4> \n");
//new code for details
sb.Append("      <p class='mb5'>" + "Gender: " + dt.Gender + "</p>      \n");
sb.Append("<p class='mb5'> DOB/Age: " + dt.dob.ToShortDateString() + "/" + dt.age + "<p>
\n");
sb.Append("      <p class='mb5'>" + "Sexual Orientation: " + dt.Sexornt + "</p> \n");
sb.Append("      <p class='mb5'>" + "Educational Qualification: " + dt.Eduqualification +
"</p> \n");
sb.Append("      <p class='mb5'>" + "Occupation: " + dt.Occupation + " in " +
dt.Occupationtype + "</p> \n");
sb.Append("      <p class='mb5'>" + "Martial Status: " + martsttus + "</p>      \n");
sb.Append("      <p class='mb5'>" + "Religion/Caste: " + relgn + "/" + caste + "</p>
\n");
sb.Append("      <p class='mb5'>" + "Location: " + dt.Location + "</p>      \n");
if (dt.Active)
{
    if (dt.Sltype)
    {
        sb.Append("      <p class='mb5 fa fa-heart'>You have accepted this Profile</p> \n");
    }
    else
    {

```

```

        sb.Append("    <p class='mb5 fa fa-heart-half'>You have not yet accepted this
Profile</p>    \n");
        sb.Append("<br>    <a class='fa fa-handshake mb5' href='AcceptProfile.aspx?psid="
+ Common.Encrypt(dt.UserId.ToString()) + "'>Accept Invitation</a><br> \n");
        sb.Append("    <a class='fa fa-heart-crack mb5' href='RejectProfile.aspx?psid="
Common.Encrypt(dt.UserId.ToString()) + "'>Reject Invitation</a><br>    \n");

    }
}
else
{
    sb.Append("    <p class='mb5 fa fa-heart-broken'>You have Rejected this Profile</p>
\n");

}
if (Convert.ToInt16(Clsusr.GroupId) > 3)
{
//        sb.Append("<br>    <i class='fa fa-paper-plane'></i>    \n");
//        sb.Append("    <a href='SendProfile.aspx?psid=" + OutId + "'>Send Your
Profile</a><br>\n");
        sb.Append("<br>    <i class='fa fa-heart'></i>    \n");
        sb.Append("    <a class='mb5' href='MyShortList.aspx?psid=" + OutId + "'>Add to
Shortlist</a><br>    \n");
        sb.Append("<br>    <a href='MyHeartList.aspx?pid=" + OutId + "' class='btn btn-
product fa fa-heart'>Add to List</a>    \n"); //<i class='fa fa-heart'>

}

//new code for details ends
sb.Append("    </div> \n");
sb.Append("    </div> \n");
sb.Append("    </div> \n");
sb.Append("    </div> \n");
//grid code
sno++;
if (sno > 3)
{
    sno = 1;
}

} //for each list

```

```

        PreHtml = sb.ToString();
        if (PreHtml.Length > 0)
        {
            objlog.Trace_Log(4, "Dashboard", "1", "NewProducts", PreHtml, "Log".ToString());
            NewProducts.InnerHtml = PreHtml;
        }

    }
}
else
{
    //data not found
    //Common.ShowMessageHtml("Search Options or Data not found", 1, true, this.dvMsgBox);

}

}
// catch (Exception ex)
//{

//Common.ShowMessageHtml(ex.Message, 1, true, this.dvMsgBox);
// ExceptionLogging.SendExcepToDB(ex);

//}
}

}
***** Register.aspx *****
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Register.aspx.cs" Inherits="Register" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head>
    <title>DateorWed.com</title>
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <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"/>

</head>

<body>
  <form id="form1" class="box login" runat="server">
    <div>
      <div>

        <div class="container col-sm-12" style="text-align:center; padding:20px 0px 20px 0px;">

          <%-- --%>
          

          <%-- <div style="text-align: left;width:30em;margin: 2em auto 3em auto;">--%>
          <div class="col-sm-12" style="text-align: left;">
            <h3 class="btn btn-rounded" style="color: forestgreen;align-
content:center">Register</h3>

            <fieldset class="boxBody bg-secondary">
              <table class="fields">
                <tbody>
                  <tr><td></td><td>
                    </td></tr>
                  <tr>
                    <td class="editor-label fa-pull-left align-content-center" style="width:8em;">
                      <strong> Email / Mobile (User Name):</strong>
                    </td>
          <%--
                    </tr>
                  <tr>--%>
                    <td>
                      <asp:TextBox ID="txtuname" runat="server"></asp:TextBox>
                    </td>
                  </tr>
                  <tr>
                    <td class="editor-label fa-pull-left align-text-top" style="width:8em;">
                      <strong> Password:</strong>
                    </td>
          <%--
                    </tr>
                  <tr>--%>

```

```

        <td>
            <asp:TextBox ID="txtpassword" runat="server" MaxLength="50"
TextMode="Password"></asp:TextBox>
<!-- <asp:RequiredFieldValidator ID="RequiredFieldValidator1" runat="server" ErrorMessage="Please
Enter password"
ControlToValidate="txtpassword"></asp:RequiredFieldValidator>
<asp:RegularExpressionValidator ID="RegularExpressionValidator1" runat="server"
ErrorMessage="Password not valid"
ValidationExpression="^.*(?!{,}){?!.*\d}{?!.*[a-z]}{?!.*[A-Z]}{?!.*[!*@#%$^&+=)].*$"
ControlToValidate="txtpassword"></asp:RegularExpressionValidator>--%>
        </td>
    </tr>
    <tr><td style="height:5px"></td></tr>
    <tr>
    <td class="editor-label fa-pull-left align-bottom" style="width:8em;">
    <strong> Your Choice:</strong>
    </td>
<%--
    </tr>
    <tr>--%>
    <td>
        <asp:DropDownList ID="ddl_GroupId" style="width:22.5em; height:25px" runat="server"
Enabled="true">
        <asp:ListItem Value="0">--- Select an Option ---</asp:ListItem>
        <asp:ListItem Value="4">Dating</asp:ListItem>
        <asp:ListItem Value="5">Wedding</asp:ListItem>
        <asp:ListItem Value="6">FriendShip</asp:ListItem>
        <asp:ListItem Value="7">Date or Wed</asp:ListItem>
        <asp:ListItem Value="8">Date and Wed</asp:ListItem>

    </asp:DropDownList>
    </td>
    </tr>
    <tr><td style="height:5px"></td></tr>
    <tr><td></td>
    <td>
        <asp:Button ID="Button1" runat="server" CssClass="btn btn-secondary btn-md btn-
rounded pull-right" Text="Register" OnClick="Button1_Click" />
    <%--
        <asp:ImageButton ID="ImageButton2" CssClass="pull-left" runat="server"
style="margin-top:20px;" ImageUrl="~/images/register.jpg" OnClick="ImageButton2_Click" />

        <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>
    <tr><td>
    </tr>
    <tr><td>

```

```
        <asp:Button ID="Button2" runat="server" CssClass="btn btn-primary btn-md btn-rounded
pull-left" Text=" Login " OnClick="Button2_Click"/>
    </td></tr>
</tbody></table>
```

```
    <div class="validation-summary-valid" id="validationSummary"><ul><li
style="display:none"></li>
</ul></div>
</fieldset>
```

```
<script type="text/javascript">
```

```
    //
    if (!window.mvcClientValidationMetadata) { window.mvcClientValidationMetadata = []; }
    window.mvcClientValidationMetadata.push({ "Fields": [], "FormId": "form0",
"ReplaceValidationSummary": true, "ValidationSummaryId": "validationSummary" });
    //]]&gt;
&lt;/script&gt;</pre></div><div data-bbox="129 500 190 530" data-label="Text"><pre>&lt;/div&gt;
&lt;/div&gt;</pre></div><div data-bbox="129 549 507 598" data-label="Text"><pre>&lt;script type="text/javascript"&gt;
    document.getElementById("userName").focus();
&lt;/script&gt;</pre></div><div data-bbox="129 650 183 665" data-label="Text"><pre>&lt;/div&gt;</pre></div><div data-bbox="129 701 197 733" data-label="Text"><pre>&lt;/div&gt;
&lt;/form&gt;</pre></div><div data-bbox="111 751 796 835" data-label="Text"><pre>    &lt;div id="footer" style="text-align:center; position:relative; top:750px; margin-left:15px;"&gt;
&lt;%--    DOW Version 1.1.10. &lt;a
href="http://portaltest.DateOrWed.com:699/Heartbeat"&gt;Heartbeat&lt;/a&gt;. Copyright © 2021
DateOrWed.--%&gt;
    &lt;/div&gt;</pre></div><div data-bbox="111 852 382 885" data-label="Text"><pre>&lt;/body&gt; &lt;%-- #030548#eb2188--%&gt;
&lt;/html&gt;</pre></div>
```



DateOrWed.com

Register

| | |
|--------------------------------------|---|
| Email / Mobile (User Name): | <input type="text" value="9966765997"/> |
| Password: | <input type="password" value="....."/> |
| Your Choice: | <input type="text" value="Wedding"/> |
| | <input type="button" value="Register"/> |
| <input type="button" value="Login"/> | |

***** Register.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 DOWEntity;
using System.Web.Security;
using System.Net;
using System.IO;
using System.Web.Script.Serialization;

public partial class Register : 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 "";
}
public string GetIPAddress()
{
    String address = "";
    try
    {
        WebRequest request = WebRequest.Create("http://checkip.dyndns.org/");
        using (WebResponse response = request.GetResponse())
        using (StreamReader stream = new StreamReader(response.GetResponseStream()))
        {
            address = stream.ReadToEnd();
        }

        int first = address.IndexOf("Address: ") + 9;
        int last = address.LastIndexOf("</body>");
        address = address.Substring(first, last - first);
    }
    catch { }
    return address;
}

protected void Button1_Click(object sender, EventArgs e)
{
    lblStatus.Visible = false;
    lblStatus.Text = "";
    int utype = 3; //-1-mobile,2=email,3=username
    FieldValidator fld = new FieldValidator();
    string uname = txtuname.Text;
    if (fld.ValidateNumeric(txtuname))
    {
        utype = 1;
    }
}

```

```

}
else if (fld.IsValidEmail(uname))
{
    utype = 2;
}
EtetblUser user = new EtetblUser();
user.UserName = Common.Encrypt(uname);
user.Password =Common.Encrypt( txtpassword.Text);
user.RoleID = ddl_GroupId.SelectedIndex;
user.Status = true;
user.FirstName = "";
user.LastName = "";
user.CreatedOn = DateTime.Now;
user.GroupId =Convert.ToInt16( ddl_GroupId.SelectedValue);
user.Department = "";
user.Region = "1"; //
user.UpdatedOn = DateTime.Today;
user.DateOfBirth = DateTime.Now;
user.TimeOfBirth = DateTime.Now;
user.adshowchk = false;
user.MatchingChk = false;
user.MTongue = 0;
user.Gender = "";
user.EmailID ="" ;
user.ContactNum ="" ;
user.IsRemoved =false ;
user.DealerId =0 ;
user.Landline ="" ;
user.PlaceOfBirth="";
string ipaddress=GetIPAddress();
List<Location> loc=GetLocation(ipaddress);
Etetbluserlog log = new Etetbluserlog();
log.PerformAction = "Register";
log.PerformActionTime = DateTime.Now;
log.UserId = 0;
log.LoginId = 0;
log.LoggedCity=_City;
log.LoggedCountry=_Country;
log.LoggedIPAddress=ipaddress;

if (utype == 1) { user.ContactNum = uname; } else { user.ContactNum = ""; }
if (utype == 2) { user.EmailID = uname; } else { user.EmailID = ""; }
if (new tblUsers().UserNameExists( uname))
{
    lblStatus.Visible = true;
    lblStatus.Text = "User Already Exists - Retry or Login";
}

```

```

else if (new tblUsers().UserEmailExists( user.EmailID) && utype==2)//check for email
{
    lblStatus.Visible = true;
    lblStatus.Text = "User Email Already Exists - Retry or Login";
}
else if (new tblUsers().UserMobileExists( user.ContactNum) && utype == 1) //check for mobile
{
    lblStatus.Visible = true;
    lblStatus.Text = "Mobile no Already Exists - Retry or Login";
}
else if (ddl_GroupId.SelectedIndex==0) //check option seleted
{
    lblStatus.Visible = true;
    lblStatus.Text = "Pl select your option - Retry ";
}

else
{
    bool ret = new tblUsers().Insert(user, log );
    if (ret)
    {
        Response.Redirect("~/Login.aspx");
    }
}
}
public List<Location> GetLocation(string ipAddress)
{
    List<Location> locations = new List<Location>();

    string APIKey = "744627112f51a720c93d8e11741a02c6f50ad135ea28e733d10f52dc0596b193";
    string url = string.Format("http://api.ipinfodb.com/v3/ip-city/?key={0}&ip={1}&format=json",
APIKey, ipAddress);
    using (WebClient client = new WebClient())
    {
        string json = client.DownloadString(url);
        Location location = new JavaScriptSerializer().Deserialize<Location>(json);
        locations.Add(location);
        _City = locations[0].CityName;
        _Country = locations[0].CountryName;
    }
    return locations;
}
protected void Button2_Click(object sender, EventArgs e)
{
    Response.Redirect("Login.aspx");
}
}

```

```

}
***** RejectProfile.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/DOWMasterPage.master"
AutoEventWireup="true" CodeFile="RejectProfile.aspx.cs" Inherits="RejectProfile" %>

<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">
      <div class=" col-lg-12 col-md-12 col=col-sm-12">
        <div class="row">
          <div id="dvMsgBox" class="span12 ordersave" runat="server" style="display: block;">
            <asp:Label ID="Label1" runat="server" Text="" Height="1px"></asp:Label>
          </div>
        </div>
      </div>
    </div>
  </div>

</asp:Content>
<asp:Content ID="Content4" ContentPlaceHolderID="CPH_MainContent" Runat="Server">
  <div class="content_holder">

    <!-- 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 ">Profiles Sent</h6>
        </div>
        <div class="row">
          <div id="NewProducts" class="new-product slider" runat="server">

            </div>
          </div>
        </div>
      </div>
    </section>

```



```

<!-- 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>
<script type="text/javascript" src="plugins/FlexSlider/jquery.flexslider-min.js"></script>
<script type="text/javascript" src="js/jquery.magnific-popup.min.js"></script>
<script type="text/javascript" src="js/magnific-popup.min.js"></script>
<script type="text/javascript" src="js/slick.js"></script>
<script type="text/javascript" src="js/slick-slider.js"></script>
<script type="text/javascript" src="js/custom.js"></script>

</div>

</asp:Content>
<asp:Content ID="Content5" ContentPlaceHolderID="ContentPlaceHolder1" Runat="Server">
</asp:Content>
<asp:Content ID="Content6" ContentPlaceHolderID="ContentFooter" Runat="Server">
<!-- START FOOTER -->
<footer class="footer" id="footer" runat="server">
<div class="container">
<div class="row">
<section class="home-main-contant-style3 bg-white">
<div class="parallax-block parallaxBg">
<div class="container">
<div class="parallax-block-text">
<p>DateorWed.com - YourChoice <span class="color-green">We are a social networking
site.</span> <br> Thank you! </p>
</div>
<div class="pre-btn-group">
<a class="btn btn-product mr15" href="#">Search Now</a>
<a class="btn btn-product mr15" href="#">Post a Profile</a>
</div>
</div>
</div>
</section>
<!-- End Parallax -->

```

```

<!-- START NEW -->
<section class="home-main-contant-style3 bg-white" style="display:none">
  <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>
</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="fab 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="#"><i class="fab fa-facebook"></i></a></li>
          <li><a href="#"><i class="fab fa-twitter"></i></a></li>
          <li><a href="#"><i class="fab fa-tumblr"></i></a></li>
          <li><a href="#"><i class="fab fa-instagram"></i></a></li>
          <li><a href="#"><i class="fab fa-linkedin"></i></a></li>
        </ul>
      </div>
    </div>
  </div>
</div>
</div>
</div>
</div>
</div>
<!--End Suvbscribe -->
<!-- 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>

```

```
</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 an Agent 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="#">dating</a></li>
```

```
        <li><a href="#">Dating & Wedding</a></li>
```

```
        <li><a href="#">Wedding</a></li>
```

```
        <li><a href="#">Option</a></li>
```

```
        <li><a href="#">Dating or Wedding</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="#">Dating <span class="pull-right">12</span></a></li>
```

```
        <li><a href="#">Wedding <span class="pull-right">11</span></a></li>
```

```
        <li><a href="#">Date or Wed <span class="pull-right">10</span></a></li>
```

```
        <li><a href="#">Date and Wed <span class="pull-right">14</span></a></li>
```

```
        <li><a href="#">Others <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 Profiles</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 Profiles</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="#">Looking for</a></h6>  
          <div class="widget-posts-meta">Wed</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="#">Searching for</a></h6>  
          <div class="widget-posts-meta">-Date-</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="#">Date or Wed</a></h6>  
          <div class="widget-posts-meta">-Wed-</div>  
        </div>  
      </li>  
    </ul>  
  </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</p>

</div>

</div>

</div>

</footer>

<!-- END FOOTER-->

</asp:Content>

***** RejectProfile.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 DOWEntity;

using System.IO;

public partial class RejectProfile : System.Web.UI.Page

{

DDLogTracer objlog = new DDLogTracer();

bool IsMobile = false;

protected void Page_Load(object sender, EventArgs e)

{

// string SrchStrng = "";

ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];

if (Clsusr == null)

{

bool ret = new Common().guest_login();

if (ret == false)

{

Response.Redirect("~/Login.aspx");

}

}

if (Clsusr != null)

{

string psid = (Request.QueryString["psid"]);

if (psid != null)

{

psid = Common.Decrypt(psid);

int pid = new tblUsers().getuserid(psid);

EteUserShortList lst = new EteUserShortList();

lst.PSID = Convert.ToInt16(Clsusr.UserId);

```

        lst.UserId = Convert.ToInt16(pid);
        lst.Uid = Convert.ToInt16(Clsusr.UserId);

        lst.CreatedOn = DateTime.Today;
        lst.Active = false;
        bool result = new tblUsers().UpdateSendProfile(lst, false);

    }    { Bindetails(psid); }

}
}
private void Bindetails(string SrchStrng)
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"]; ;
    int sno = 0;
    string cat = "";
    string PreHtml = "";
    StringBuilder sb = new StringBuilder();

    cat = Common.Decrypt(Request.QueryString["cat"]).Trim();
    // int len = cat.Length;
    // cat = cat.Substring(0, len - 15);
    // cat = cat.TrimEnd('\0');

    // sb.Append(" <div class='directory-list'>");
    Etetbluser_options opt = new Etetbluser_options();

    //try
    {
        List<EteGetSearchResults> Lst = (new
tblUsers()).GetRcvdProfilebyUserId(Convert.ToInt16(Clsusr.UserId));
        if (Lst.Count > 0)
        {
            var pst = (from c in Lst orderby c.UserId descending select c).FirstOrDefault();
            Lst=(from c in Lst orderby c.UserId descending select c).ToList();
            if (Lst != null && Lst.Count > 0)
            {

                //IList<EteGetSearchResults> Lst1 = new List<EteGetSearchResults>();

                foreach (var dt in Lst)
                {
                    string path = "images/NoPhoto.png";
                    string path1 = "images/No Photo.png";
                    string href = "";
                    string OutId = Common.Encrypt(dt.UserName.ToString());

```



```

href = "ProfileDetail.aspx?id=" + OutId;
var list = new tblUsers().getuserImagesbyuserid((int)dt.UserId, "").ToList();
int cnt = list.Count();
if (cnt > 1)
{
    path = list[0].ImagePath;
    path1 = list[1].ImagePath;//"http://dateorwed.com/"+
}
else
    if (list.Count() > 0)
    {
        path = list[0].ImageUrl;
    }

if (path == null || path == "")
{
    path = "images/Nocolor.png";
}
if (File.Exists(path))
{
    path = list[0].ImageUrl;
    path1 = list[1].ImageUrl;
}
if (path1 == null || path1 == "") { path1 = path; }

string martsttus = new tblUsers().getmatstatus(dt.Martialstatus);
string relgn = new tblUsers().getreligion(dt.Religion);
string caste = new tblUsers().getcaste(dt.Caste);
//slick code
sb.Append("        <div class='col-lg-3 col-md-4 item'>        \n");
sb.Append("        <div class='products grid-product'>        \n");
sb.Append("        <div class='product-grid-inner'>\n");
sb.Append("        <div class='product-grid-img'> \n");
sb.Append("        <ul class='product-labels'>        \n");
sb.Append("        <li>Date</li>        \n");
sb.Append("        </ul> \n");
sb.Append("        <a href='ProfileDetail.aspx?id=" + OutId + "' class='product-grid-img-
thumbnail'>        \n");
sb.Append("        <img src="" + path + "" alt='img' class='img-responsive primary-img' />
\n");
sb.Append("        <img src="" + path1 + "" alt='img' class='img-responsive socendary-img'
/>        \n");
sb.Append("        </a>        \n");
sb.Append("        <div class='product-grid-atb text-center'><a class=""
href='ProfileDetail.aspx?id=" + OutId + "'>Detailed View</a></div>        \n");
sb.Append("        </div> \n");
sb.Append("        <div class='product-grid-caption text-center'>        \n");

```

```

        sb.Append("      <h4><a class='p-grid-title' href='ProfileDetail.aspx?id=" + OutId + "'>" +
dt.FirstName + " " + dt.LastName + "</a></h4> \n");
        //new code for details
        sb.Append("      <p class='mb5'>" + "Gender: " + dt.Gender + "</p> \n");
        sb.Append("<p class='mb5'> DOB/Age: " + dt.dob.ToShortDateString() + "/" + dt.age + "<p>
\n");
        sb.Append("      <p class='mb5'>" + "Sexual Orientation: " + dt.Sexornt + "</p> \n");
        sb.Append("      <p class='mb5'>" + "Educational Qualification: " + dt.Eduqualification +
"</p> \n");
        sb.Append("      <p class='mb5'>" + "Occupation: " + dt.Occupation + " in " +
dt.Occupationtype + "</p> \n");
        sb.Append("      <p class='mb5'>" + "Marital Status: " + martsttus + "</p> \n");
        sb.Append("      <p class='mb5'>" + "Religion/Caste: " + relgn + "/" + caste + "</p>
\n");
        sb.Append("      <p class='mb5'>" + "Location: " + dt.Location + "</p> \n");
        if (dt.Active)
        {
            if (dt.Sltype)
            {
                sb.Append("      <p class='mb5 fa fa-heart'>You have accepted this Profile</p> \n");
            }
            else
            {
                sb.Append("      <p class='mb5 fa fa-heart-half'>You have not yet accepted this
Profile</p> \n");
            }
        }
        else
        {
            sb.Append("      <p class='mb5 fa fa-heart-broken'>You have Rejected this Profile</p>
\n");
        }
    }

    if (Convert.ToInt16(Clsusr.GroupId) > 3)
    {
        //sb.Append("      <i class='fa fa-exchange-alt'></i> \n");
        //sb.Append("      <a href='SendProfile.aspx?psid=" + OutId + "'>Send Your
Profile</a><br><br> \n");
        //sb.Append("      <i class='fa fa-heart'></i> \n");
        //sb.Append("      <a href='MyShortList.aspx?psid=" + OutId + "'>Add to Shortlist</a><br>
\n");
        //sb.Append("<br>      <a href='MyHeartList.aspx?pid=" + OutId + "' class='btn fa fa-
heart'>Add to List</a> \n"); //<i class='fa fa-heart'>
    }
}

```

```

        //new code for details ends
        sb.Append("    </div> \n");
        sb.Append("    </div> \n");
        sb.Append("    </div> \n");
        sb.Append("    </div> \n");
        //grid code
        sno++;
        if (sno > 3)
        {
            sno = 1;
        }

    } //for each list

    PreHtml = sb.ToString();
    if (PreHtml.Length > 0)
    {
        objlog.Trace_Log(4, "Dashboard", "1", "NewProducts", PreHtml, "Log".ToString());
        NewProducts.InnerHtml = PreHtml;
    }
}
else
{
    //data not found
    //Common.ShowMessageHtml("Search Options or Data not found", 1, true, this.dvMsgBox);
}

}
// catch (Exception ex)
//{

//Common.ShowMessageHtml(ex.Message, 1, true, this.dvMsgBox);
// ExceptionLogging.SendExcepToDB(ex);

//}
}

```

```
}
```

```
***** SearchList.aspx *****
```

```
<%@ Page Title="" Language="C#" MasterPageFile="~/DOWMasterPage.master"  
AutoEventWireup="true" CodeFile="SearchList.aspx.cs" Inherits="SearchList" %>
```

```
<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: none;  
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>
```

```
                <%--<button class="navbar-toggler" type="button" data-toggle="collapse" data-  
target="#navbarSupportedContent-1"  
                aria-controls="navbarSupportedContent-1" aria-expanded="false" aria-label="Toggle navigation">  
                    <span class="navbar-toggler-icon"></span>  
                </button>
```

```
                <div class="collapse navbar-collapse " id="navbarSupportedContent-1">--%>  
                    <div id="sidebar" class="col-md-3 col-sm-10 col-xs-10 hidden-xs sidebar"  
runat="server">
```

```
                        <!-- CATEGORIES WIDGET -->
```

```
                        <div id="Filter" class="widget mb30" runat="server">
```

```
                            <h6 class="text-uppercase bottom-line">Showing Results for</h6>
```

```
                            <ul class="icons-list">
```

```
                                <li><a href="#">Age Wise 25-30 <span class="pull-right">12</span></a></li>
```

```
                                <li><a href="#">Age Wise >30 <span class="pull-right">11</span></a></li>
```

```
                                <li><a href="#">Religion wise <span class="pull-right">10</span></a></li>
```

```
                                <li><a href="#">Location Wise <span class="pull-right">14</span></a></li>
```

```
                                <li><a href="#">City Wise <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 Searched</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 Results</h6>
  </div>
  <div class="small-thumb-proudct mb30">
    <div class="thumb-product">
      

      <div class="thumb-product-inner">
        <h6><a href="SearchList.aspx?SrchStrng=">Dating Profiles</a></h6>
        <span class="thumb-product-cat">Men's Profiles 10</span>
      </div>
      <ul class="list-inline overflow-h">
        <li class="thumb-product-price line-through"></li>
        <li class="thumb-product-price">women's Profiles 15</li>
      </ul>
    </div>
  </div>
  <div class="small-thumb-proudct mb30">
    <div class="thumb-product">
      

      <div class="thumb-product-inner">
        <h6><a href="SearchList.aspx?SrchStrng=">Wedding Profiles</a></h6>
        <span class="thumb-product-cat">Men's Profiles 25</span>
      </div>
      <ul class="list-inline overflow-h">
        <li class="thumb-product-price line-through"></li>
        <li class="thumb-product-price">Women's Profiles 100</li>
      </ul>
    </div>
  </div>
  <div class="small-thumb-proudct mb30">
    <div class="thumb-product">
      
```

```
<div class="thumb-product-inner">
  <h6><a href="SearchList.aspx?SrchStrng=">FRIENDS LIST</a></h6>
  <span class="thumb-product-cat">College Friends 124</span>
</div>
<ul class="list-inline overflow-h">
  <li class="thumb-product-price">School Friends 65</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="#">Dating Profiles</a>
    <a href="#">Wedding Profiles</a>
    <a href="#">Dating or Wedding</a>
    <a href="#">Dating & Wedding</a>
    <a href="#">Friends</a>
    <a href="#">Lists</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="pull-right"><!--view-mode--%>
      <li class="grid-list-icons">
        <asp:ImageButton ID="ImgBtn1" runat="server"
ImageUrl="images/icons/list.png" OnClick="ImgBtn1_Click" />
        <asp:ImageButton ID="ImgBtn2" runat="server"
ImageUrl="images/icons/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>
<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> Profiles </h2>
        <small class="category-num-coun">9 Results</small>
      </div>
    </div>--%>
  </div>
</div>
</div>

```

```

</div>
<section class="home-main-contant-style2 dip-bg-style dip-style bg-white">

<div id="appvendors" class="category-products col-grid-style slick-slider" runat="server"
style="display:block">
<%--dynamic code APPVENDORS from here--%>
<div>

</div>

<%--dynamic code APPVENDORS ends here--%>
</div>
<div id="AppGrid" class="category-products col-grid-style slider" runat="server" style="display:none">

<%--dynamic code APPGRID from here--%>
<div>

</div>
<%--dynamic code APPGRID ENDS here--%>
</div>

<div class="row">
<div id="NewProducts" class="new-product slider" runat="server" style="display:none" >

</div>
</div>
</section>
<script type="text/javascript" src="plugins/owl-carousel/owl.carousel.min.js"></script>
<script type="text/javascript" src="plugins/FlexSlider/jquery.flexslider-min.js"></script>
<script type="text/javascript" src="js/jquery.magnific-popup.min.js"></script>
<script type="text/javascript" src="js/magnific-popup.min.js"></script>
<script type="text/javascript" src="js/slick.js"></script>
<script type="text/javascript" src="js/slick-slider.js"></script>
<script type="text/javascript" src="js/plugins.min.html"></script>
<script type="text/javascript" src="js/custom.js"></script>

<%--dynamic code ends here--%>

<!-- Start Pagination-->

```



```
<div id="paging" class=" cd-pagination "" runat="server" visible="false"> <!--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>

```

```

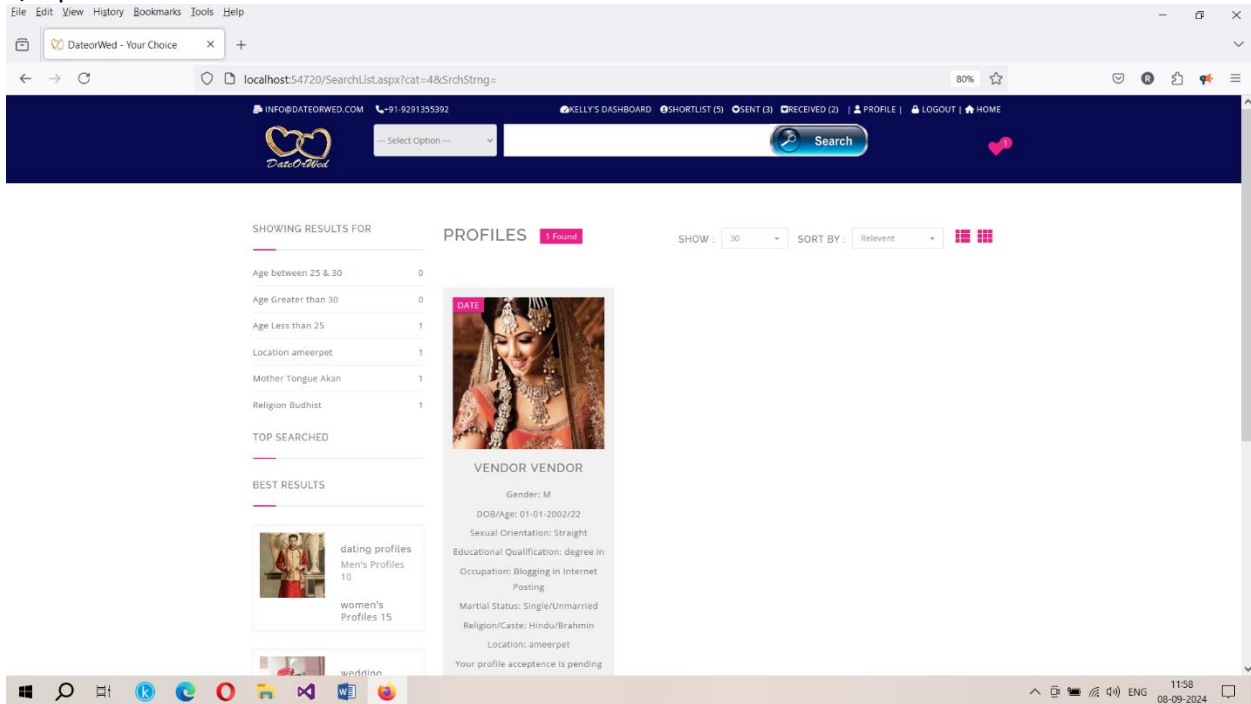
<asp:Content ID="Content6" ContentPlaceHolderID="ContentFooter" Runat="Server">

```

```

</asp:Content>

```



***** SearchList.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 DOWEntity;
//using LogTracer;
using System.Text;
using System.Text.RegularExpressions;
using System.Drawing;
using System.IO;

public partial class SearchList : System.Web.UI.Page
{
    DDLogTracer objlog = new DDLogTracer();
    bool IsMobile = false;
    string urlqry = "";
    BalCommon blc = new BalCommon();
    protected void Page_Load(object sender, EventArgs e)
    {
        string SrchStrng = Request.QueryString["SrchStrng"].ToString();
        if (Request.Browser.IsMobileDevice) { IsMobile = true; } else { IsMobile = false; }
        //if (qry.ToString().Substring(0,3) != "pid")
        //{
        //    Common.ShowMessageHtml(new BalCommon().Decrypt(qry), 1, true, this.dvMsgBox);
        //    urlqry = new BalCommon().Decrypt(qry).ToQueryString();
        //}
        if (SrchStrng == null) { SrchStrng = ""; } else { SrchStrng = Common.Decrypt(SrchStrng); }
        if (IsMobile)
        {
            appvendors.Attributes.CssStyle.Add("display", "none;");
            AppGrid.Attributes.CssStyle.Add("display", "none;");
            NewProducts.Attributes.CssStyle.Add("display", "block;");
        }
        else
        {
            appvendors.Attributes.CssStyle.Add("display", "none;");
            AppGrid.Attributes.CssStyle.Add("display", "none;");
            NewProducts.Attributes.CssStyle.Add("display", "block;");
        }
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr == null)
        {
            bool ret = new Common().guest_login();
            if (ret == false)
            {
                Response.Redirect("~/Login.aspx");
            }
        }
    }
}

```

```

    }
}
if (Clsusr != null)
    {
    }
else
    {
        //Response.Redirect("~/login.aspx");
    }
if (Request.Browser.IsMobileDevice) { IsMobile = true; } else { IsMobile = false; }
if (IsMobile)
    {
//        sidebar.Attributes.CssStyle.Add("display", "none;");

    }
//if (!IsPostBack)
    { Bindetails(SrchStrng); }

}
private void Bindetails(string SrchStrng)
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    if (Clsusr == null)
    {
        bool ret = new Common().guest_login();
        if (ret == false)
        {
            Response.Redirect("~/Login.aspx");
        }
    }
    string title = "", qstring = "", pagesort = "";
    int pid = 0;
    string page = "1";
    int srno = 1, perpage = 30, dvno = 1, sno = 1;
    int curpage = Convert.ToInt16(page);
    string cat = "";
string ppage    = "";
    try
    {
        title = "Profiles";
        //if (HttpContext.Current.User.Identity.IsAuthenticated )
        {
            page = Session["page"].ToString();
        }
        ImgBtn1.Enabled = true;
        ImgBtn2.Enabled = false;
        cat = Request.QueryString["cat"].ToString();
        pagesort = Request.QueryString["psort"];
    }
}

```

```

ppage = Request.QueryString["ppage"];
if (ppage != null) { perpage = Convert.ToInt16(ppage); }

SrchStrng = Common.Decrypt(Request.QueryString["SrchStrng"]);

if (SrchStrng != null && SrchStrng != "")
{
    // len = SrchStrng.Length;
    // SrchStrng = SrchStrng.Substring(0, len - 12);
    // SrchStrng = SrchStrng.TrimEnd('\0');

}
}

catch {}
    if (SrchStrng != null) { SrchStrng = ""; }
    if (pagesort == null) { pagesort = "N"; }
    curpage = Convert.ToInt16(page);
    string PreHtml = "";
    StringBuilder sb = new StringBuilder();
    StringBuilder sbg = new StringBuilder();
    StringBuilder sbs= new StringBuilder();

    sb.Append(" <div class="">>");//directory-list
        qstring = "cat=" + Request.QueryString["cat"] + "&SrchStrng=" +
Request.QueryString["SrchStrng"];
        Etetbluser_options opt = new Etetbluser_options();

        string fromage = Request.QueryString["fromage"];
        string toage = Request.QueryString["toage"]; ;
        string Location = Request.QueryString["Location"];
        if (Location == null) { Location = ""; }
        string Religion = Request.QueryString["Religion"]; ;
        string MotherTongue = Request.QueryString["MotherTongue"]; ;
if(fromage==null)
{
    opt = new tblUsers().Gettbluser_options(Convert.ToInt16( Clsusr.UserId));
    opt.Location_search = "";
}
else
{
    opt.Gender_Search = "0";
    opt.Age_from =Convert.ToInt16( fromage);
    opt.Age_to = Convert.ToInt16( toage);
    opt.Location_search = ( Location);
    opt.Religion_search =Convert.ToInt16( Religion);
}

```

```

opt.MTongue_search = Convert.ToInt16(MotherTongue);
opt.Area_search = 0;
}
if (cat == "") { cat = Clsusr.GroupId; }
    //try
    {
        IList<EteGetSearchResults> Lst = new List<EteGetSearchResults>();
        Lst = (new tblUsers()).getsearchprofile(Convert.ToInt16(Clsusr.UserId), SrchStrng,cat,opt);
        if (Lst.Count > 0)
        {
            var pst = (from c in Lst select c).FirstOrDefault();

            title = "Profiles"; // pst.Department;
            sb.Append("<h2>" + title + "</h2>\n <small class='category-num-coun'>" +
Convert.ToString(Lst.Count) + " Found</small>\n");
            results.InnerHtml = sb.ToString();
            sb = new StringBuilder();
            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<EteGetSearchResults> Lst1 = new List<EteGetSearchResults>();
                if (pagesort == "H")
                {
                    Lst1 = (from c in Lst orderby c.age descending select c).ToList();
                } else if (pagesort == "L")
                {
                    Lst1 = (from c in Lst orderby c.age ascending select c).ToList();
                }
                else if (pagesort == "R")
                {
                    Lst1 = (from c in Lst orderby c.Religion descending select c).ToList();
                }
                else if (pagesort == "C")
                {
                    Lst1 = (from c in Lst orderby c.Caste,c.SubCaste descending select c).ToList();
                }

                else if (pagesort == "A")
                {

```

```

    Lst1 = (from c in Lst orderby c.Location descending select c).ToList();
}
else if (pagesort == "T")
{
    Lst1 = (from c in Lst orderby c.GeoLocation descending select c).ToList();
}
else if (pagesort == "M")
{
    Lst1 = (from c in Lst orderby c.MTongue descending select c).ToList();
}
else if (pagesort == "G")
{
    Lst1 = (from c in Lst orderby c.Gender descending select c).ToList();
}
else
{
    Lst1 = (from c in Lst select c).ToList();
}

}

foreach (var dt in Lst1)
{
    string path = "images/NoPhoto.png";
    string path1 = "images/No Photo.png";
    string href = "";
    string OutId = Common.Encrypt(dt.UserName.ToString());
    href = "ProfileDetail.aspx?id=" + OutId;
    var list = new tblUsers().getuserImagesbyuserid((int)dt.UserId, "").ToList();
    int cnt = list.Count();
    if (dt.UserId == -1)
    {
        foreach (var imgs in list)
        {
            byte[] bt = (byte[])imgs.Images.ToArray();
            string server = Server.MapPath(imgs.ImgPath);
            string server1 = Server.MapPath("thumb/" + imgs.ImgPath);
            File.WriteAllBytes(@server, bt);
            //FileInfo fi = new FileInfo(@server);
            //FileStream fs = fi.Open(FileMode.OpenOrCreate, FileAccess.Read, FileShare.Read);
            //StreamReader sr = new StreamReader(fs);

            //string fileContent = sr.ReadToEnd();

            ReduceImageSize(0.2, server, server1);
        }
    }
}
if (cnt > 1)

```

```

{
    //path = list[0].ImageUrl;
    //path1 = list[1].ImageUrl;//"http://dateorwed.com/"+
    path = list[0].ImgPath;
    path1 = list[1].ImgPath;//"http://dateorwed.com/"+
}
else
    if (list.Count() > 0)
    {
        path = list[0].ImgPath;
    }

if (path == null || path == "")
{
    path = "images/Nocolor.png";
}
if (File.Exists(path))
{
    path = list[0].ImageUrl;
    path1 = list[1].ImageUrl;
}
if (path1 == null || path1 == "") { path1 = path; }

//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-6'> \n");
sb.Append("        <div class='product-grid-img'> \n");
sb.Append("        <ul class='product-labels'> \n");
sb.Append("        <li>" + dt.GroupName + "</li> \n");
sb.Append("        </ul> \n");
sb.Append("        <a href='ProfileDetail.aspx?id=" + OutId + "' class='product-grid-img-
thumbnail'> \n");
sb.Append("        <img src="" + path + "" alt='img' class='img-responsive primary-img' />
\n");
sb.Append("        <img src="" + path1 + "" alt='img' class='img-responsive socendary-img'
/> \n");
sb.Append("        </a> \n");
sb.Append("        </div> \n");
sb.Append("        </div> \n");
sb.Append("        <div class='col-sm-5 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='ProfileDetail.aspx?id=" + OutId + "'>"
+ dt.FirstName + " " + dt.LastName + "</a></h4></li> \n");
sb.Append("        <li class='pull-center'> \n");
sb.Append("        <ul class='list-inline product-ratings'> \n");
if (dt.Status)

```



```

        {
            sb.Append(" <li><i class='rating-selected fa fa-check-circle'></i></li> \n");
        }
        else
        {
            sb.Append(" <li><i class='rating-selected fa fa-times-circle'></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");
        string martsttus = new tblUsers().getmatstatus(dt.Maritalstatus);
        string relgn = new tblUsers().getreligion(dt.Religion);
        string caste = new tblUsers().getcaste(dt.Caste);

        sb.Append(" <p class='mb5'>" + "Gender: " + dt.Gender + "</p> \n");
        sb.Append("<p class='mb5'> DOB/Age: " + dt.dob.ToShortDateString() + "/" + dt.age + "<p>
\n");
        sb.Append(" <p class='mb5'>" + "Sexual Orientation: " + dt.Sexornt + "</p> \n");
        sb.Append(" <p class='mb5'>" + "Educational Orientation: " + dt.Eduqualification +
"</p> \n");
        sb.Append(" <p class='mb5'>" + "Occupation: " + dt.Occupation + " in
"+dt.Occupationtype+ "</p> \n");
        sb.Append(" <p class='mb5'>" + "Marital Status: " + martsttus + "</p> \n");
        sb.Append(" <p class='mb5'>" + "Religion/Caste: " + relgn+ " / "+caste + "</p>
\n");
        sb.Append(" <p class='mb5'>" + "Location: " + dt.Location + "</p> \n");
        sb.Append(" <p class='mb5'>" + "Mother Tongue: " + dt.MotherTongue + "</p>
\n");
        if (Convert.ToInt16(Clsusr.GroupId) > 3)
        {
            if (!dt.SentProfileAdded)
            {
                sb.Append(" <i class='fa fa-exchange-alt'></i> \n");
                sb.Append(" <a href='SendProfile.aspx?psid=" + OutId + "'>Send Your
Profile</a><br><br> \n");
            }
            else
            {
                if (dt.SentSltype)
                {
                    sb.Append(" <p class='mb5'>Your profile has been accepted by them</p>
\n");
                }
            }
        }
    }
}

```

```

    }
    else
    {
        if (dt.SentActive)
        {
            sb.Append("    <p class='mb5'>Your profile acceptance is pending</p> \n");
        }
        else
        {
            sb.Append("    <p class='mb5'>Your profile has been rejected</p>    \n");
        }
    }
}

//sb.Append("<ul class='list-inline add-to-wishlist mb15'>    <li class='product-
wishlist'>
    \n");
//sb.Append("    <i class='fa fa-paper-plane'></i>    \n");
//sb.Append("    <a style='text-decoration: underline' href='SendProfile.aspx?psid=" +
OutId + "'>Send Your Profile</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='MyShortList.aspx?psid=" + OutId + "'>Add to Shortlist</a>
\n");
//sb.Append("    </li></ul>    \n");
//sb.Append("    <ul class='list-inline add-to-wishlist'>    \n");
//sb.Append("    <li class='product-wishlist '>    \n");
//sb.Append("    <i class='fa fa-heart'></i><i class=' fa fa-exchange '></i><i class='fa fa-
heart '></i>    \n");
//sb.Append("    <a style='text-decoration: underline' href='MyHeartList.aspx?psid=" +
OutId + "'>Check compatibility</a><br> \n");
//sb.Append("    </li></ul><br> \n");
//sb.Append("    <div class='product-grid-atb text-center'><i class='fas fa-shopping-
cart'></i><a class=" href='ProfileDetail.aspx?id=" + dt.UserId + "'>Add to Cart</a></div> \n");
////sb.Append("    </ul>    \n"); sb.Append("    <button type='button' class='btn btn-
product fa fa-heart'>Add to Cart</button>    \n");
//sb.Append("    <a href='MyShortList.aspx?psid=" + OutId + "'>Add to Shortlist</a><br>
\n");
if (!dt.HeartListAdded)
{
    sb.Append("    <a href='MyHeartList.aspx?pid=" + OutId + "' class='btn btn-product fa
fa-heart'>Add to List</a>    \n"); //<i class='fa fa-heart'>
}
}

```

```

        sb.Append("        <br> <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-half'></i></li>          \n");
        sb.Append("        </ul> \n");
//        sb.Append("<a href='MyHeartList.aspx?pid=0'></a><img border='0' alt='Compliment'
src='images/DateOrWed_ico.png' width='50' height='50'/><p>Check compatibility</p> ");
    }
    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-2 col-sm-4 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>"+dt.Sexornt+"</li> \n");
    if (dt.Status)
    {
        sbg.Append("        <li><i class='rating-selected fa fa-check-circle'></i></li> \n");
    }
    else
    {
        sbg.Append("        <li><i class='rating-selected fa fa-times-circle'></i></li> \n");
    }
    }
    sbg.Append("        </ul> \n");
    sbg.Append("        <a href='ProfileDetail.aspx?id=" + OutId + "' 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='ProfileDetail.aspx?id=" + OutId + "'>Detailed View</a></div>          \n");
    sbg.Append("        </div> \n");
    sbg.Append("        <div class='product-grid-caption text-center'> \n");
    sbg.Append("        <ul class='list-inline overflow-h mb15'> \n");
    sbg.Append("        <li><h4 class='title-price'><a href='ProfileDetail.aspx?id=" + OutId + "'>
+ dt.FirstName + " " + dt.LastName + "</a></h4></li>          \n");

```

```

        sbg.Append("        <li class='pull-center'> \n");
        //sbg.Append("        <ul class='list-inline product-ratings'> \n");
        //if (dt.Status)
        //{
        //    sbg.Append("        <li><i class='rating-selected fa fa-check-circle'></i></li> \n");
        //}
        //else
        //{
        //    sbg.Append("        <li><i class='rating-selected fa fa-times-circle'></i></li> \n");
        //}

        //sbg.Append("        </ul> \n");
        sbg.Append("        <p class='mb5'>" + "Gender: " + dt.Gender + "</p> \n");
        sbg.Append("<p class='mb5'> DOB/Age: " + dt.dob.ToShortDateString() + "/" + dt.age +
"</p> \n");
        sbg.Append("        <p class='mb5'>" + "Sexual Orientation: " + dt.Sexornt + "</p> \n");
        sbg.Append("        <p class='mb5'>" + "Educational Qualification: " + dt.Eduqualification +
"</p> \n");
        sbg.Append("        <p class='mb5'>" + "Occupation: " + dt.Occupation + " in " +
dt.Occupationtype + "</p> \n");
        sbg.Append("        <p class='mb5'>" + "Martial Status: " + martsttus + "</p> \n");
        sbg.Append("        <p class='mb5'>" + "Religion/Caste: " + relgn + "/" + caste + "</p>
\n");
        sbg.Append("        <p class='mb5'>" + "Location: " + dt.Location + "</p> \n");
        if (Convert.ToInt16(Clsusr.GroupId) > 3)
        {
            sbg.Append("        <i class='fa fa-exchange-alt'></i> \n");
            sbg.Append("        <a href='SendProfile.aspx?psid=" + OutId + "'>Send Your
Profile</a><br><br> \n");
            sbg.Append("        <i class='fa fa-heart'></i> \n");
            sbg.Append("        <a href='MyShortList.aspx?psid=" + OutId + "'>Add to Shortlist</a><br>
\n");
            sbg.Append("<br>        <a href='MyHeartList.aspx?pid=" + OutId + "' class='btn fa fa-
heart'>Add to List</a> \n"); //<i class='btn-primary fa fa-heart'>

        }
        sbg.Append("        </div> \n");
        sbg.Append("        </div> \n");
        sbg.Append("        </div> \n");
        sbg.Append("        </div> \n");

        srno++;
        sno++;
        if (sno > 4)
        {
            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");

    }
    //slick code
    sbs.Append("        <div class='col-lg-3 col-md-4 col-sm-12 col-xs-12 item'> \n");
    sbs.Append("        <div class='products grid-product'> \n");
    sbs.Append("        <div class='product-grid-inner'>\n");
    sbs.Append("        <div class='product-grid-img'> \n");
    sbs.Append("        <ul class='product-labels'> \n");
    sbs.Append("        <li>Date</li> \n");
    sbs.Append("        </ul> \n");
    sbs.Append("        <a href='ProfileDetail.aspx?id=" + OutId + \" class='product-grid-img-
thumbnail'> \n");
    sbs.Append("        
\n");
    sbs.Append("         \n");
    sbs.Append("        </a> \n");
    sbs.Append("        </div> \n");
    sbs.Append("        <div class='product-grid-caption text-center'> \n");
    sbs.Append("        <h4><a class='p-grid-title' href='ProfileDetail.aspx?id=" + OutId + \">\" +
dt.FirstName + \" \" + dt.LastName + \"</a></h4> \n");
    //new code for details
    sbs.Append("        <p class='mb5'>\" + dt.Gender + \"</p> \n");
    sbs.Append("<p class='mb5'> DOB/Age: \" + dt.dob.ToShortDateString() + \"/\" + dt.age +
\"</p> \n");
    sbs.Append("        <p class='mb5'>\" + dt.Sexornt + \"</p> \n");
    sbs.Append("        <p class='mb5'>\" + dt.Eduqualification +
\"</p> \n");
    sbs.Append("        <p class='mb5'>\" + dt.Occupation + \" in \" +
dt.Occupationtype + \"</p> \n");
    sbs.Append("        <p class='mb5'>\" + martsttus + \"</p> \n");
    sbs.Append("        <p class='mb5'>\" + relgn + \"/\" + caste + \"</p>
\n");
    sbs.Append("        <p class='mb5'>\" + dt.Location + \"</p> \n");

```

```

if (Convert.ToInt16(Clsusr.GroupId) > 3)
{
    if (!dt.SentProfileAdded)
    {
        sbs.Append(" <i class='fa fa-exchange-alt'></i> \n");
        sbs.Append(" <a href='SendProfile.aspx?psid=" + OutId + "'>Send Your
Profile</a><br><br> \n");
    }
    else
    {
        if (dt.SentSltype)
        {
            sbs.Append(" <p class='mb5'>Your profile has been accepted</p>
\n");
        }
        else
        {
            if (dt.SentActive)
            {
                sbs.Append(" <p class='mb5'>Your profile acceptance is pending</p> \n");
            }
            else
            {
                sbs.Append(" <p class='mb5'>Your profile has been rejected</p> \n");
            }
        }
    }

    }
    //sbs.Append(" <i class='fa fa-exchange-alt'></i> \n");
    //sbs.Append(" <a href='SendProfile.aspx?psid=" + OutId + "'>Send Your
Profile</a><br><br> \n");
    //sbs.Append(" <i class='fa fa-heart'></i> \n");
    //sbs.Append(" <a href='MyShortList.aspx?psid=" + OutId + "'>Add to Shortlist</a><br>
\n");
    //sb.Append(" <a href='MyShortList.aspx?psid=" + OutId + "'>Add to Shortlist</a><br>
\n");
    if (!dt.HeartListAdded)
    {
        sbs.Append("<br> <a href='MyHeartList.aspx?pid=" + OutId + "' class='btn btn-
product fa fa-heart'>Add to List</a> \n"); //<i class='btn-primary fa fa-heart'>
    }
}
sbs.Append(" </div> \n");
sbs.Append(" </div> \n");
sbs.Append(" </div> \n");
sbs.Append(" </div> \n");

```

```

//new code for details ends

} //for each list

//sb.Append("      </div> \n");
//sb.Append("      </div> \n");
sb.Append(" </div> \n");
sb.Append(" </div> \n");
sbg.Append("      </div> \n");
sbg.Append("      </div> \n");
sbg.Append("      </div> \n");
//grid code

PreHtml = sb.ToString();
if (PreHtml.Length > 0)
{
    //objlog.Trace_Log(4, "ShoppingList", "1", "appvendors", PreHtml, "Log");
    appvendors.InnerHtml = PreHtml;
}

// PreHtml = sbg.ToString();
if (PreHtml.Length > 0)
{
    objlog.Trace_Log(4, "ShoppingList", "1", "AppGrid", sbs.ToString(), "Log".ToString());
    //AppGrid.InnerHtml = sbg.ToString();
    NewProducts.InnerHtml = sbs.ToString();
}

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");
    }
}
else
{
    //data not found
    Common.ShowMessageHtml("Search Options or Data not found", 1, true, this.dvMsgBox);
}

sb = new StringBuilder();
int grp =Convert.ToInt16( Clsusr.GroupId);
List<getcount> prd = new tblUsers().gettotalgroupcount(Convert.ToInt16(Clsusr.UserId),
grp.ToString());
if (prd.Count > 0)
{
    sb.Append("<h6 class='text-uppercase bottom-line'>Showing Results for</h6>\n");
    sb.Append("<ul class='icons-list'>\n");
    foreach (var data in prd)
    {
        if (data.count>0)
        {
            sb.Append("<li><a href='SearchList.aspx?pid=1&cat=" + cat + "&SrchStrng=" +
data.SearchStr + ">" + data.Desc + " <span class='pull-right'>" + data.count +
"</span></a></li>\n");
        }
        else
        {
            sb.Append("<li>" + data.Desc + " <span class='pull-right'>" + data.count +
"</span></li>\n");
        }
    }
}
sb.Append("</ul>\n");
//objlog.Trace_Log(4, "SearchList", "1", "paging", sb.ToString(), "Log");
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='ProfileDetail.aspx?id=" + 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='ProfileDetail.aspx?id=" + 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='" + "SearchList.aspx?" + qstring + "&ppage=30" + "'>30</a></li>
\n");
sb.Append("    <li><a href='" + "SearchList.aspx?" + qstring + "&ppage=45" + "'>45</a></li>
\n");
sb.Append("    <li><a href='" + "SearchList.aspx?" + qstring + "&ppage=60" + "'>60</a></li>
\n");
sb.Append("    <li><a href='" + "SearchList.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 = "Age High to Low"; }
else if (pagesort == "L") { sort = "Age Low to High"; }
else if (pagesort == "N") { sort = "Relevant"; }
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='" + "SearchList.aspx?" + (qstring + "&psort=R") + "'>Religion
wise</a></li> \n");
        sb.Append("    <li><a href='" + "SearchList.aspx?" + (qstring + "&psort=H") + "'>Age High to
Low</a></li> \n");
        sb.Append("    <li><a href='" + "SearchList.aspx?" + (qstring + "&psort=L") + "'>Age Low to
High</a></li> \n");
        sb.Append("    <li><a href='" + "SearchList.aspx?" + (qstring + "&psort=C") + "'>Caste
wise</a></li>\n");
        sb.Append("    <li><a href='" + "SearchList.aspx?" + (qstring + "&psort=A") + "'>Location
wise</a></li> \n");
        sb.Append("    <li><a href='" + "SearchList.aspx?" + (qstring + "&psort=T") + "'>City
wise</a></li> \n");
        sb.Append("    <li><a href='" + "SearchList.aspx?" + (qstring + "&psort=N") + "'>Country
wise</a></li> \n");
        sb.Append("    <li><a href='" + "SearchList.aspx?" + (qstring + "&psort=M") + "'>MotherTongue
wise</a></li> \n");
        sb.Append("    <li><a href='" + "SearchList.aspx?" + (qstring + "&psort=S") + "'>Language
wise</a></li> \n");
        sb.Append("    <li><a href='" + "SearchList.aspx?" + (qstring + "&psort=G") + "'>Gender
wise</a></li> \n");
        sortbtn.InnerHtml = sb.ToString();
        //objlog.Trace_Log(4, "ShoppingList", "1", "sortbtn", sb.ToString(), "Log");
        //blc.encrypt
        Session["page"] = page;
    }
    // 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;
    NewProducts.Attributes.CssStyle.Add("display", "none;");
    sidebar.Visible = true;
}
protected void ImgBtn2_Click(object sender, ImageClickEventArgs e)
{
    appvendors.Attributes.CssStyle.Add("display", "none;");
    AppGrid.Attributes.CssStyle.Add("display", "none;");
    NewProducts.Attributes.CssStyle.Add("display", "block;");

    ImgBtn1.Enabled = true;
    ImgBtn2.Enabled = false;
    sidebar.Visible = true;
}
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);

}
private void ReduceImageSize(double scaleFactor, string sourcePath, string targetPath)
{
    try
    {
        //using (System.Drawing.Image image = System.Drawing.Image.FromStream(sourcePath))
        using (System.Drawing.Image image = System.Drawing.Image.FromFile(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);
        }
    }
    catch {}
}

}
***** SearchProfile.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/DOWMasterPage.master"
AutoEventWireup="true" CodeFile="SearchProfile.aspx.cs" Inherits="SearchProfile" %>

<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-12 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="containe">
```

```
<div class="row">
```

```
<aside>
```

```
<div id="sidebar" class="col-lg-2 col-md-2 hidden-xs sidebar" runat="server">
```

```
<!-- CATEGORIES WIDGET -->
```

```
<div id="Filter" class="widget mb30" runat="server">
```

```
<h6 class="text-uppercase bottom-line">Showing Results for</h6>
```

```
<ul class="icons-list">
```

```
<li><a href="#">Age Wise 25-30 <span class="pull-right">12</span></a></li>
```

```
<li><a href="#">Age Wise >30 <span class="pull-right">11</span></a></li>
```

```
<li><a href="#">Religion wise <span class="pull-right">10</span></a></li>
```

```
<li><a href="#">Location Wise <span class="pull-right">14</span></a></li>
```

```
<li><a href="#">City Wise <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 Searched</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 Results</h6>
```

```
</div>
```

```
<div class="small-thumb-proudct mb30">
```

```
<div class="thumb-product">
```

```

```

```
<div class="thumb-product-inner">
```

```
<h6><a href="SearchList.aspx?SrchStrng=">Dating Profiles</a></h6>
```

```

        <span class="thumb-product-cat">Men's Profiles 10</span>
    </div>
    <ul class="list-inline overflow-h">
        <li class="thumb-product-price line-through"></li>
        <li class="thumb-product-price">women's Profiles 15</li>
    </ul>
</div>
</div>
<div class="small-thumb-proudct mb30">
    <div class="thumb-product">
        

        <div class="thumb-product-inner">
            <h6><a href="SearchList.aspx?SrchStrng=">Wedding Profiles</a></h6>
            <span class="thumb-product-cat">Men's Profiles 25</span>
        </div>
        <ul class="list-inline overflow-h">
            <li class="thumb-product-price line-through"></li>
            <li class="thumb-product-price">Women's Profiles 100</li>
        </ul>
    </div>
</div>
<div class="small-thumb-proudct mb30">
    <div class="thumb-product">
        

        <div class="thumb-product-inner">
            <h6><a href="SearchList.aspx?SrchStrng=">FRIENDS LIST</a></h6>
            <span class="thumb-product-cat">College Friends 124</span>
        </div>
        <ul class="list-inline overflow-h">
            <li class="thumb-product-price">School Friends 65</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="#">Dating Profiles</a>
        <a href="#">Wedding Profiles</a>
        <a href="#">Dating or Wedding</a>
    </div>
</div>

```

```

        <a href="#">Dating & Wedding</a>
        <a href="#">Friends</a>
        <a href="#">Lists</a>
    </div>
</div>
<!-- END TAGS WIDGET -->
</div>
<%--</div>--%>
</aside>

<div class=" col-lg-9 col-md-9">

<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/icons/list.png" OnClick="ImgBtn1_Click" />
                <asp:ImageButton ID="ImgBtn2" runat="server"
                ImageUrl="images/icons/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>
    <div class="col-md-9 col-sm12 col-xs-12">--%>
        <div class="row hidden-xs">
<!--
            <div class="toolbar">
                <div id="results" class="col-sm-8 category-num" runat="server">
                    <h2> Profiles </h2>
                    <small class="category-num-coun">9 Results</small>
                </div>
            </div>--%>
        </div>
</div>
</div>

    <div id="appvendors" class="category-products col-grid-style slick-slider" runat="server"
style="display:none">
<!--dynamic code APPVENDORS from here--%>
<div>

</div>
<!--dynamic code APPVENDORS ends here--%>
    </div>

<!--dynamic code APPGRID starts here--%>
    <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 Profiles</h6>
            </div>
            <div class="row">
                <div id="AppGrid" 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>Date</li>
</ul>
<a href='#' class='product-grid-img-thumbnail'>
<img src='img/img (26).jpg' alt='img' class='img-responsive primary-img' />
<img src='img/img (6) (1).jpg' alt='img' class='img-responsive socendary-img' />
</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='#'>Janet Evanovich</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>Date</li>
</ul>
<a href='#' class='product-grid-img-thumbnail'>
<img
src='data:image/png;base64,/9j/4AAQSkZJRgABAQEASABIAAD/4gXYSUNDX1BST0ZJTEUAAQEAAAxitGlu
bwIQAAABtnRyUkdCIEFhZWIAHhgACAABgAxBhY3NwTVNGVAAAABJRUMGc1JHQgAAAAAAAAAAAA
AAAAAAAA9tYAAQAAAADTLUhQICAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAABFjchJOAAABUAAAADNkZXNjAAABhAAAAGx3dHB0AAAB8AAAAABRia3B0AAACBAAAAB
RyWFlaAAACGAAAABRnWFlaAAACLAAAABRiWFlaAAACQAAAABRkbW5kAAACVAAAABHBkbWRkAAACxAAA
AAIh2dWVkaAADTAAAAIZ2aWV3AAAD1AAAACRsdW1pAAAD+AAAABRtZWZzAAAEDAAAACR0ZWNoAA
AEMAAAAAxyVFJDAAAEPAAACAxnVFJDAAAEPAAACAxVFJDAAAEPAAACAx0ZXh0AAAAAENvcHlyaWdod
CAoYykgMTk5OCBIZXdsZXROLVBhY2thcmQgQ29tcGFueQAAZGVzYwAAAAAAAAAASc1JHQiBJRUM2MTk2N
i0yLjEAAAAAAAAAAAAAAAAABJzUkdCIElFQzYxOTY2LTIuMQAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAFhZWIAAAAAAAAAABvogAAOPUAAAQWFlaIAAAAAAAAAAGKZAAC3hQAAGNpYWVogAAAAAA
AAJKAAAA+EAAC2z2Rlc2MAAAAAAAAAAFkIFQyBodHRwOi8vd3d3LmllYy5jaAAAAAAAAAAAAAAAAAAAFkIFQyB
odHRwOi8vd3d3LmllYy5jaAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAABkZXNjAAAAAAAAAC5JRUMgNjE5NjYtMi4xIERlZmF1bH0gUkdCIGNvbG91ciBzcGFjZSAtIHNSR0IA
AAAAAAAAAAAAAAAAAC5JRUMgNjE5NjYtMi4xIERlZmF1bH0gUkdCIGNvbG91ciBzcGFjZSAtIHNSR0IAAAAAAA
AAAAAAAAAAAAAAAAAAAZGVzYwAAAAAAAAAASUmVmZXJlbmNlIFZpZXdpcmcgQ29uZGl0aW9uIGluE
lFQzYxOTY2LTIuMQAAAAAAAAAAAAAAAAALFJmVjZW5jZSBWaWV3aW5nIENvbmRpdGlubiBpbmBJRUM2MTk2Ni0yLjE
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAEwAA1yeAAAAVhZWIAAAAAAAAAAEwJVgBQAAAAVx/nbWVhcwAAAAAAAAABAAAAAAAAAAAAAAAAAAAA
```

AAAAAAo8AAACc2lnIAAAAABDUlQgY3VydgAAAAAAQAAAAABQAKAA8FAAZAB4AlwAoAC0AMgA
3ADsAQABFAEoATwBUAFkAXgBjAGgAbQByAHcAfACBAIYaiwCQAJUAmgCfAKQAqQCuALiAtwC8AMEAxg
DLANAA1QDbAOAA5QDrAPAA9gD7AQEBBwENARMBQGQEFASUBKwEyATgBPgFFAUwBUgFZAWABZwFuA
XUBfAGDAYsBkgGaAaEBqQGxAbkBwQHJAdEB2QHhAekB8gH6AgMCDAIUAh0CjglvAjgCQQJLAIQCXQJnAn
ECegKEAo4CmAKiAqwCtgLBAssC1QLgAusC9QMAAawsDFgMhAy0DOANDA08DWgNmA3IDfgOKA5YDogO
uA7oDxwPTA+AD7AP5BAYEEwQgBC0EOwRIBFUEYwRxBH4EjASaBKgEtgTEBNME4QTWBP4FDQUcBSsFOg
VJBvGFzWv3BYfIgwMbbUfXqXVBeUF9gYGBhYJGwY3BkgGWQZqBnsGjAadBq8GwAbRBUmG9QcHBxkH
Kwc9B08HYQd0B4YHmQesB78H0gflB/glCwgfCDIIRghaCG4lggiWCKolvgjSCOci+wkQCSUJOGIPCWQJeqmP
CaQJugnPCeUJ+woRCicKpQpUCmoKgQqYcQ4KxQrcCvMLCwsiCzkLUQtpC4ALmAuwC8gL4Qv5DBIMKgxD
DFwMdQyODKcMwAzZDPMNDQ0mDUANWg10DY4NqQ3DDd4N+A4TDi4OSQ5kDn8OmW62DtIO7g8JDy
UPQQ9eD3oPlg+zD88P7BAJECYQQxBhEH4QmxC5ENcQ9RETETERTxFtEYwRqhHJEegSBxImEkUSZBKEEqM
SwxLjEwMTixNDE2MTgxOkE8UT5RQGFCCUSRRqFIsUrRTOFPAVEhU0FVYVeBwBfB0V4BYDFIYWSRZsFo8
WshbWFvoXHRdBF2UXiReuF9IX9xgbGEAYZRiKKG8Y1Rj6GSAZRRlrGZEZtxndGgQaKhpRGncanhrFGuwbFB
s7G2MbihuyG9ocAhwqHfIcexyJHMwC9R0eHUcdCB2ZHcMd7B4WHkAeah6Uhr4e6R8THz4faR+UH78f6iA
VIEEgBCCYIMQg8CEclUghdSGhlc4h+yInllUigiKvlt0jCiM4I2YjCPCI/AkHyRNjHwkqyTaJQkIOCVoJZclxyX3Jic
mVyaHJrcm6CcYJ0kneierJ9woDSg/KHEooijUKQYpOClrKZ0p0CoCkjuqaCqBks8rAis2K2krnSvRLAUsOSXuLK
ls1y0MLUEtdi2rLeEuFi5MLoluty7uLyQvWi+RL8cv/jA1MGwwpDDbMRlXsJGCMbox8jIqMmMymzLUMw0z
RjN/M7gz8TQRNGU0njTYNRM1TTWHNcl1/TY3Nnl2rjbpNyQ3YDecN9c4FDhQOIw4yDkFOUI5fzm8Ofk6Nj
p0Orl67zstO2s7qjvoPCc8ZTykPOM9lj1hPaE94D4gPmA+oD7gPyE/YT+iP+JAI0BkQKZA50EpQWpBrEHuQjB
CckK1QvdDOkn9Q8BEA0RHRIpEzkUSRVVfMkXeRiJGZ0arRvBHNUd7R8BIBUHLSJFI10kdSWNJqUnwSjdKfU
rESwxLU0uaS+JMkKxyTLpNAK1KTZNN3E4ITm5Ot08AT0IPk0/dUCdQcVC7UQZRUFGBUeZSMVJ8UsdTE1Nf
U6pT9IRCvI9U21UoVXVvWlYPVlxWqVb3V0RXklfGWC9YfVjLWRpZaVm4WgdaVlqmWvVbRVuVW+VcNVy
GXNZdJ114XcleGI5sXr1fD19hX7NgBWBXYKpg/GFPYah9WJJYpxi8GNDY5dj62RAZJRk6WU9ZJl52Y9ZpJm
6Gc9Z5Nn6Wg/ajZo7GlDaZpp8Wplap9q92tPa6dr/2xXbK9tCG1gbbLuEm5rbsRvHm94b9FwK3CGcOBxOn
GVcfByS3KmcwFzXXO4dBR0cHTMdSh1hXXhdj52m3b4d1Z3s3gReG54zHkqeYI53pGeqV7BHtje8J8IXyBfO
F9QX2hfgF+Yn7CfyN/hH/IgEeAqIEKgWuBzYlwgPKC9INXg7qEHYSAhOOFr4WRhg6GcobXhzuHn4gEiGmlzo
kziZmJ/opkisQLMluWi/yMY4zKjTGNmI3/jmaOzo82j56QBpBukNaRP5GokhGSepLjk02TtpQglIqU9JvflcmW
NJaflwqXdZfgmEyYuJkkmZCZ/JpomtWbQpuvNByciZz3nWsd0p5Anq6fHZ+Ln/qgaaDYoUehtqImopajBqN2
o+akVqTHpTilqaYapoum/adup+CoUqjEqTepqaocqo+rAqt1q+msXKzQrUStuK4trqGvFq+LsACwdbDqsWCx
1rJLssKzOLOutCW0nLUTtYq2AbZ5tvC3alfguFm40blKucK6O7q1uy67p7whvJu9Fb2Pvgq+hL7/v3q/9cBww
OzBZ8Hjwl/C28NYw9TEUcTOxUvFyMZGxsPHQce/yD3IvMk6ybnKOMq3yzbLtsW1zLXNNc21zjbOts83z7jQ
OdC60TzRvtl/0sHTRNPG1EnUy9VO1dHWVdbY11zX4Nhk2OjZbNnx2nba+9uA3AXcit0Q3ZbeHN6i3ynfr+A
24L3hROHM4IPi2+Nj4+vkc+T85YtmDeaW5x/nqegy6LzPRunQ6lvq5etw6/vshu0R7ZzuKO6070DvzPBY8OX
xcvH/8ozzGfOn9DT0wvVQ9d72bfb794r4Gfio+Tj5x/pX+uf7d/wH/Jj9Kf26/kv+3P9t////2wCEAAMEBAQGB
AYHBwYICQgJCAwLCgoLDBINDg0ODRIcERQRERQRHBgdGBYYHRgslh4eliwyKigqMj02Nj1MSUxkZIYBAwQ
EBAYEBgcHBggJCAkIDAsKcgsMEg0ODQ4NEhwRFBERFBECGB0YFhgdGCwiHh4iLDIqKcOyPTY2PUxJTRkxh
/AABEIAfQB9AMBIGACEQEDEQH/xACmAAACAgMBAQEAAAAAAAAAAAAAEBQMGAIIHAQgJEAAcAgIBAwI
FAwIDBwMBCQEBAGADBbEhBRixIkEGEzJRcQcUYSOBQpHBFsqzUmKhsQhyglMWFyc0Q2ODkuHwAQACA
wEBAQAAAAAAAAAAAAADBAIFBgEABxEAAgIBBAAEbglBAwQAAAAAAAAECAxEEiExEYlyQUJm1FhcRSBQ
mKRwQYVJDRSgrH/2gAMAwEAhEDEQA/APt94C4ka5IJM2NimYptGoSYtuEoHVF5M6HbKL1NeTFLmLGA
uymUL/WMeBlupX+tHarKtbyWuegQ1bnoojFUhASdVaPOys/aqSOJZOm4nbcPEGVRs24NfqWMxiheTLrj
qewSawcySpfQJ5ZLZLyFU35gujwI4SJ6fAjhJYU9ISsC1m+poslEslirPNTYT2bQyRBmTzU3mTuDhCzoTJWEiYS
Jlk9oG8J9o5n9JEo9jHDHojZYjxbB26jZHEapktiF7E9zCpk07p7uNADxhK/1Uaw7f/aY7dwJXuqWD9rYP+kxL
VNeDnf6WN6dPxiftFG6VmMlhUngy3/vG1OcuGhhHAYW1MrRfKEMZNLfp4ysykNsZILDkxj0vLO9EyrFi8lx2
KvuchdL+QpnJ0xdDidXqsDCEjzKx07I7uJZFabKi1TgmZS6twm0Sj5PYyAPJkyeTx48mTJ4Z48ez2eT0Tx4yeiZ
PZ48eSF/EnkT+DOPo6uwFG2TjF1Tesw3uikJZiHlHDF54uMb1n0xKx/qxtUeIkH+aX7JWrhBe5H8zmbyMoD
HZbvYWWDCMJ7A2Uia/MaB8bHaJ7M9EjJeQs2xBUXZhfYJctzbbJswsI1c8TxVnmtNjHCJZk2yPsRkTskrHQ
mgM9KMW8M8mylpMhEyD8KJLez5YS5wXk37nUKWjbelBfjKtGzypM1ccYJv3Q15IYz7gSZlRopiDjuPcYvKb

aDRismVD+tHSyrV5QVxGwzVla5JNj6i2kOVMJWIRmiELmCdJZHPZ5wZYFHllzwF5E51XmAkToHTbe7tjUZRY
rOMsF9rHoEitk9f0SKyW3+BU/wCQRT4EbpFFPgRuniO0itgWskEiWSiPoVZIJlXJBGEDZ7qeTaZJ4OGk0YSaak
TmDoK3iL3MZuOlqti8w0C0TyDxGAvfUTKeTDIJ1AQcvZh3t90MkyiPM2bNHErulfirsdwYj2HMq3UfizoXT3T
97lpRvevmHthHbNLG47GmEnnazolmYv3IT6lMfOKicx6t+p/wVRWxHU6bP4Q7nlb/wBa8E3hU6fY1ZIHf3gR
K6ds00ixooogmmfQiEg+IRsTmHR/1A+Hs2pyLTU6KS6WDRAA3Hmf8TdlxsX5puDE1Gxa05ZgOeBKzPHEW
WLcVJ5aRe0sUDzJ625nx+36x/Pz/wBvTQlL8gfN2w7h/hbt8GXLp36o3L2jMxVhCQFettqdnUKqbljMSGYNP
DyfUuNmCpxs8S74mWlhADAz58xPiLBymUJZpmHABiX3oWUT1FUJ+8Z02rthdGLjxKSQhq9JCVUpZ5SOwA
zJoDNtzZGTPJk1M8BnMncG88nsydOGT0TybTx49mTJk8eMkT+DJZE/gzj6OrsrqNqwwvvipm1cZKLJTRswW
Tgesf6ka0tylkJ9cZVHiRpn53+zlkfKh0DNoMpm3fLdTWcVwbP9Jgl8yaxyRlk+oRO2WbEGisRcQj1IMK1PFE2j
0Y4QBvINZMBm9g4nqiQw97054B32ZBylWlgjDRgLYpVISMjwdxEyeBtTIJOP3JYZwWleZHfWITSOZ7cszMOa
OLKPI0jZIMzqyH/tOjZSeZS81B3RZw4YypcnObbGDjiGI9h9oW9S944jWulZm7s72X1eNiAEL68SUNZ9o5Wp
ZsUEXe5BVgFxs5sAnX+ii3aNic0xVHzhOxdHA7RG9Jl2LLE9VhVsuda+mDW+YyXxF13ma+SxBGWT8zJaY2Tx
FFMbp4h6gVgWvibiRrNXLBCrJhJRIRJRDoeYQT2eTaEIM81PjtMnTxA8T5EcvEWYwCkxe3oPV2LPmqnJkGfl4
uLhW5WbetOPWpZtnQAH3MgrZU77bD4HG/YT82/1k/U/L65n2YGNcU6bQ+uPN7j3gKeRuaGP6j/rr1TqFtu
H0TWHh8g3Ef1bB/oJ8xYa35/U60e6y93KD5jEkguQRyYv/AGj3A1opZifGjrj3M7B0fpvR+jYqNkbuzcingHgUg
c8fzG3DjhHoZUuei65Hw/0DGowrMvI7RkV9rWKCVDj3/gSkn4j6h0XLKJj02UB+O71d+vsx+8x+rNmB6bWB
+YF7CfHeo/1Eq1burUtWWGjx58e6wMaUvVyNeI31wy+5XxDhW3m7E3XVY/caz5Rvt+JesbqpyKvnFxuuntJ
Pnn0z5leiz5Dmo7cEBI92Eu3S+rM+MzfbSN9yuveMV1xUkgUrG4sc9Kwqe+w3lfmPvss89reNR6r24weu7uF
bb1Z5Q9w9/swM5Vf1Djx7vBIDnuX7keCJfukdVS+tKy/tpCfB2fpYGL3RabYzppR9Jd+ifEOXjXfY7G+F86/wC
pDPo74c/UP9nmUjN5qJA+b7oT7H+J8o5/RrVRsrEBDVjd2P5H/vr/ANRLR0bqdeXQgsAb2IldPY1vS67RZkV
cJfbhn6s9P6jj5WPXbXyRq6ghlOwRG3eJ8Pfpr8SHAz06e9rNiXeqhz/APpufKGFy1Lk+ZdabUKyH5RINZpJU2Y
9n0Ny89QwdYQgjabyllk4mxngm0MDNZtMns8eMmTJk8eMkb+DJJE/gzj6PFLuOrT+ZCLOZtkn1t+YuDczM2
Sw2XsFIDUNzGIDxKhjGnlhPVSakDsXBYF8TGPE3UcCaOOJetPYVa7NQhaYayIWvibTqpi0c3s1VpL3QQnRmK
+zJqa6I4C9cTWeiYYiZBBR7woT1hsSE45idTwxYJkINZBmRHY/sHyjhdPtNrpIPtPbpmGhiVrKHmU3NHMu
mT7yoZw5gn6WHj2ilOfUpzHNXil7vr/ALxtW3AmVu4sZoq/QhkCJ7Bg0IBgMhcBOMP6wnXekfSjYTHP9UTr
XSPAjukXzEi6v6bL6vil7owX6YBcOZrpehGWXqPaY3SJ6o2rPEJURsDB4kgkS+JKI+hRkokgkQkojEQbJtNzCo
gz2ZMmTpwieVnNG217Dkyv4IVznYBz2GK3vgZpXJ8w/q/8AGV3SOHnj1OBflbRSPZPcz84qEsycglqb+2p3H
9SOsnrvxVkdgfspscUICf8K+TKglGPh3mqsaPy/UT53O6aCUVntj9mWxxhYUJhU6Xta09oZzzsj2nOMq+6/le1z
p1vYj8GEW5961nu+rvY/5QlqXsYdviY8fcGOZwwbzLoCLn5ZUnRVhpvto+loyq6httWDT93J/n7iSrgWMnpHI
Gj/l8cwtOhX2pwvA+4i8rY5YaFM8AhqN9psQBbRww9nEDHzERgU0SfV7bl+/8y5YXQclfJI+3PEcXdBtdFIQ8
SC1EFjE3p5tdM5zmUG9QQDvfp+m4qw1urYFSQyHgeXB9jOsV9IsryWr7eSONzyroireT2cckg+xPKGQnqINB
l6ewsXSesV3V1BzqwAhHPkH7GRtWmH1FLiOsfJfkDwr/f8GKqul3ByFU9p9vfiXqjzZNT1WLyf8Asw8N+DK
K2ahY2un2XkFOVeH2uiy/D9xbqL1FtdpUiff/AEHP+bg45Zt2BSfuRPzwxcdLrvqddi0KUbF8Ecz6i+/f/iS2rsSxC
EJGx9vYGL6TUyr1P4awwWv03i6ZfeLyj6or5WFARX0/lpuoV62DKRwQY3Am6racUzATTUumjybTyegwwM9
mTJk8eMmTJk8eMkb+DJJG84+jxRMv63/MUg+qNcw+t/wAxMp9cyl3ZoavSN6oyoOniyqH1+ZyDBzXZaEYE
CbkbEW1OY0HiaKqanEqJRwz0TYkTWQFoZySiYPW5Mj0RzJf5hAWDUdxLODatp7Ndam0YwCAjEPM9moU
7m08jxoRzMnp8zJw8cBp9ptdPKfab3TGzNRErmQPMqeast2RkTmQX+LDrtFDyF0394Qp4nmT5P5myjiZLVf
UZo6PQjYPCRbAHnimV255Hcd3Hs/qidg6M3pE4jtt/VWdn6GfSJZaKXzUV+sj8tnSU+mB3DmGV/TBbptn6E
ZBeoHQxrX4ixBGIYM9X2cmGr4kokSyQSwQqyYsUSEsURiIjkk3mgm0KQZtMnkySlkzbj6i9eHSfhrqOTz3JU
e3X/M/pE7Da3as+F//AFFdZ+TgYXTgf6uRabrPxXwBER+cL7sc065PjLoxsa+3KYk64B/mDdRuP7pX7+SAP8vE
MsYYmNTSPDFUpzKxkWP3ITzoz+4MNDrJYSwslktsWy5dsPEvHsemWXNXtPT7Gc0FoSweokb8H2M6t0v4ix
MPDAK7cDzvwlpqz28djOkhBzeTqL9J6bRUHYhRrnUGGRipd8tOPHAHqP9vac3/wBs5mdbsnsAbhR7f/7Oo
9A6d8zt7UJG5npzlh1M0tcltclidYWH81tpUN/ceZ0fp3w+HT1iO+kdFrUDajj2nSsfFRQAAJ6DlIFY4R6OZ2/CO
C4BNZB/iej4QxdH0bnZ66E14kv7ZY/GjKK96nDOHn4Xx1fYr3o74EeVfDeOSGKabU6g1CgQYV8wUqkgquck
Ulfh2he4qigk8kjcdpomtKtdnP/aWStYelYI1AOCw8EXY8rI2+DsmqoPQxA7m4/g/adXE4GaTVYHG+33E7piXl
8dDve1HP3/mX3wfuYlXKqS5h/8AjM18VoUbFZF8TCJg8zJgmiKE2mTJk8eMmTJk8eMkbYSRP4M4+jxQc3/i
P+YIX644zj/Uf8xMn1TJXepmj9A7q8Q6vzAKvEP8zsAUx1UlerRdW0YINiXdD44Kuzs3Zpqj7M27BJRG1FuX
IDPB6Fm4mTQmG4RAkmTyezp4yZPJ5uePG0yeTJ48cBp9pLd4kNPtJrvExcZTxK7kSrZfiWjllayhxBprhvdFKyR

yfzNlHpEkyRyZsg9ImV1UfmM0OnfkQM44g/iHWelucyol2WcOgzHP8AVX8ztnQjwJw7G/4i/mdu6D4Ef0X1
UI636bOoVfSINDcavpEhum8/wMV/mQVxpX4i2uMkkqzkwkSQSITcRxC7JhJRIRJRDxBsIE2mgm8MDPZk8kb
nieyeIHPBY+B4n5L/AKR9dbq/x9IOGBrqtFFf8JXP1A+KeqLgdDzMj/6Vlt/kj+MZymt6l8xzyWZz+WO4q3mb/
CLHTx4yFddfbLrWwBoypNcToN7Rx1Owu7AkcMdSqMe5Rxo+/wCRJRlhlakszZtkWWC3Sgx5g4Wdc6FAx34
Mnx8bvSq0AntIDT6K+G+k16V0esb+w9Uq9VqdkS00mn3S7EXw38J5TOnep1Pp7ovS6sWtQFHEg6bjooAA/
vLtUFAmbcpTnls0WFGO1la46x5UwiSo8Q1G0TzHq5YwiThktVPIkxJAI3GshjsD7y5hPyFNOvzmjNAS/PA5k5
aQErFbJZGq449idWh9DcjZijg6Oplbu0doPJ4EXbCuGS6YISWryPI/8y49HBrq+UTwh4/EpfTH0ol4xCvz1JPkal
voFHxlyXfT/TM1rt3nj7docTweZkweZpzPm0yZMnjxkyZMnjx5In8GSGQueJF9HV2c+zz/AFH/ADEtZ9UadQP
9R/zElTeqZC5+ZmlqXkLJUelfX5i6rxGFfmSgAmN643T6YrpeaqJe6ZcZKq18m893M1PNR8WPS2hBWs5m9m
4KBsxa2cs4QWMUDOWibwZRqTiMxbwCZsxkWjJZk0cIntMkmpkHtZLJwGk8CEXeLT4ELt+mZKZpYldv8A
MrmSPTLJfK9kj0wcegr7RTMocmbv/QJvIdkyKs+kTMav6jL/AE/oRrb4ixljsrsSBqZUTg2yyhIDx1PzFnbOheBO
R1V6cTrnQ/aN6NNWoV1bzWzqNX0iRWySn6RNLpvF6DFP1ENfmMU8RdX5JFPEIAjnEkEiE3EaQFk4kgkYm
8OgTJRNpoJ7DjkdYzQ+TM3NNzjZ1I+cv1x6scP4Gy9Hm5kpH/zn5a4Y+Zbb5+kf+Z9p/8AqP6532dOwEf0p3
3P+T6RPjDpPqy7efAHB+0Ti/JN/dlvXF+RAOc67Gyfb9vtwYsTsJAPnfBHVDOqKQXG+Qe9f9RB8NA7BCfPIM9
lbEF58QvfQBSSa3+k++tzv/QMTIpCqLSa/bwJxzoWBuxQy6/B5n0Z0bEs7F22wP4ma1tizjPnDB4zg6JhH0D
UstUQY66Ucx9SRoStrLKSyzrZhDw5I45gVYh6Vx6CYpNxyGVO4P0mHtlKBzsfkQKoARkzL28yyrzsK6eN/QtO
dj+9gH54gxzadn1g/jmGvXWfYQ9um9iKy3ZG4+Hj3lzmM3CVk/yeBGOHS/f3Pyx95rUiRrWQJ6MW3ywU5J
J4RYMUITLthMC6fWZQsexe6XfpoBtSW2k4tivyJoa5eVt/YtO+J4PM8nomsMubTJkyePGTJk8njxqTArn0phL
nQlayMnzFb7FFB6oOTk1BvW8RUt640zW2DEtB9cydr8xpal5C3UngRjV5iug8CNKvMNVWKT9x1S4Y4TxEtUc
J4I/pXwU93ZNPJqTB/mR2U0gCizezxlagdZQvsySs8xXKIYmFw1ELE3Bkc93HEBJk0BmEzuThhMyR9pmQeZ
Hcl4DT4ENT+mAUngQ6z6ZkplmliV++I8gemPrvMSZHOGDiFZTcscmC1n0iG5Y5MCq12zNaz6pf6X0BqSTUjU
8SXcQ4Hecnigd4nT+inxOYjyJ0rop5ELp/qIBqPps6nT9lmt09o+gTLZul+hGNI6wdPMYJ4i9IlekEjIIE3EjEkEZQJ
k4m4kYm8OgTJBPZrPYQieEyG5+yImPspMmli7KO1C/bk/2g5vCZOKy0fIJ+uGcz/G1tQYsyU1g/lue2Wn4X/R
u5slW5/U7K77E38mtAwr37MT5Mp3xo1V/6xubOU/2pUrD8ETt2leqDr+SEvsH9Qt541+JnddqrKYVQivUst
m++DfDqtTDUWTF08YR8w/qD8G9U6FfUbfXWxPZco9L6/8ABnMun6axQvB8p+J90/HWWnUvhfgGLcn9Wt
Pmr+U52s+LU6bbTaB7Btqw9o1ptVG2l/cR12iiTesLhrJ3b4YHf2d/bvjwOZ9H4ICLWoE+bPhbluN6/M1xrx4n
1DjLupfxM7q180t9K/loKVSlpPIEESD5V+SqhKRpiOX/wCWCgvdzZbbeC1vl4uOnda4WVqz446UpclW79vk
giJqOmYgsFuQzWv93Ox/ICsrq/S8WvIaVGvsJawtikJyp56bM/8AvD6ICQe9SPI4MOxfjvod7hRcVPj1Aic0zsj4
eyw/clH1a3xKoeg4/d31Wa8akZamH5QSOmR9QY3VcS8brtVuSOI3rsB8GfNXSq8zHtbrB7n3/nO4dNyyah3
QEb054J26fbEtQfUX5fWcXGUmX9Qeyz0kgzjnxDndOpYNIzIClVoyuaaSQGNKfLLyPjiyyzWJjs5JnYvhXrfWrL
a2vpCr3Da+8+Tul/GPw7Q3ZS517IUJn1J8H5tOYamrsDA8iMae21Xw9SeRHW01fx5vEWsH0BPRMno8z6Kf
NT2ZMmTx4yeT2eTx4FvPoMoDuST+Ze8o6qb8TnpMpdC/NEs9GuJcZMPpMTUH1xrln0mJcc+uZ+zsV15S5
UH0iNajzE+Oel2q8xit9CVg6rMb1niJq4wqbmXenlHlVag+B2rDIPd9MftScGLx9SFhOp6L9GaNI02UjnOL4Y8
oxa5HyOGG5624Lj8LDDLmDcq02lyWJM1rYkScQRW00JBEb8EZIIEyabEyGyiB880/SifYR2xRW2IELsc9sx1j
NPBC+6Jr/AKDGdhJi28HsMhAJlp2ZrmKkcdoh2efP4iGtj2zM6/6hoNGvIOFeFAXMjGNK5VIsWghflnSejeROc
bAlnQ+jNylzR9RCt/02dYo+gSSyRY/OCTWeJuq/poxc/WwVYYKEEJWSRhxQkkHBkgMNFgmECSAGbbjCYPB
MDNiZCDNpPJHBJuLcnQqdvsDDgeBFHUGP7K0DyVMFY/KEgvMj8d/inN//EDNyPYZ4cfgPPuWtKMb93INrb
EAfjzPgb4u0fifqIb8XuAfwZ9xfDeSnVugYdhORfikf/5EHawIH8TrbrokI7YNz8DuSephnh4b/SZQ+udeLkJViVu
ee5j9pw/N6LmU7VKDdjnmqwfUq+yt+J9RVdF6f22E62OJVMXHxtzVHIAO3n/AJ5QRsnVyu9mm1FVNkWK
mtvucg+GK8k5C7BAE+qcNdY6D+JxXCpC53aNdqt/mZ3HHGqhJWWb5ZwVsKtiwFBliqyb9cSwlgMxsXHY6d
O6QxLHB1YycP8Aij4htwcJyg7n0dcz5fyMnrnUCWusf6e4KSR3e2gPafevUehdMaliMRPyTzKG3SsSpgRvryja
N8yw011VL88cv7gNRp7r4YhNx/BwH4d+Cb+qZtVdxsqSyiy1WQqxXtIAJkmRhdX+HepGoZvzK/IDHg/wfsZ3
erGxqe41s1fdw3YO3f8AlFw6TTfeFYodz7sN7hr9bVPiMco5o/h11bcpzFfSuoZGTyHwogHzPojo+ITVs+4IX6
V0R6F/qKgPsFGGj0rplYQFf4gNpp8zWVg7q78QeHk4x+pPXMjPobQICsXuY+AToCfL1HS+r9TyhZeuQ27e02
CvvHafcaZ7s+I+krIV0uB6q2P+TTnbdPycduBtd70RJeJ4E5fLy89kVTHUVRza1+DI3SegdApwMxM3DNt9lRgk
BSHUBe0HY5HM+lf0t+F/iDp2cj5mabUIQroD1aHGzK10+5DYOGJY7b/AMK8f3Jnfvh+2/5tfeAvl0o9oejUO2+
ttPckiu1ul8DTTUX3F5yddmw8yLc3Bn0JM+bEkyeT2dOGTww2eGePCrPbVDfiUDfEuVvw1jNKJviZ/XS+b/Rc
6Rflf7F2YfQYloPrjXLPpMT459cpLC4h0XPGPEcVeYlx/EcVeYev2ErfcdVwxYFXCWzBvVvorJh6vxlN05EDC0SW

CcpLAq0kLWUyAgx/8sQd6VMBPsvHDCRuA630IQ1o1AGRIBUMrq2OZGuVj8qR2aj2aK3MKEjNQHibqDDw
Uk8MHJpnpJmTeZC7SOT51qG1EZGrawKgHsjXu9EzU9pfxylHqAirJUdhja1xE17AqYDckHUWzn/UF5MSU1
ntlghzyYnpYama1nMzQaXij03phtD8QO2QVW8yq6ZZ4yh8zCX7obcic0DzovQzysNS34qF7l8tnZsY+gQmzxB
MX6BCrPE3tX00Yez6jBhJ1gwkymSycCAZtuQgzzcmmQwFBptuCGyUGGTItBAMkBkIM93CZIEimJuoHeLzr0
nmMweTEud6se9f4MFN+QLBeZH41/GZKfF/UfscizU7V+j3xbQWyui33dloc34jH+fqE5V+qWHZi/G+erb5cM
v4YTh/UL8nGyMfKotau6ptq68FSDwZOVcu0qj+FgsqNS9Pqd65WeV90friuLXkFv3NSBvdkOg05B1H4fsp6w
92LZvHba2bbWh7zlpWp+suJ1DHqxup2JjZi4+Z4rtn0Hlth0V4wflJrf51Tt3Kfp7DrX4O9gzlaim6G6MoYa7bN7
p9VTZtcLMqfUV7YKkMVUIIGtWKZ0uk6QCUDGuqffawYA+x3LvU3Alcj028juk/zGCjcvU63GwcaEcrWQPuF
VgGT3dKwrl9VKk/iDI/MOTIZSTvY14lIXGvGJIDONucxeGlv/s1g93CEe/ky0YfTMSICECiQNkMA39Upx6y7uAB
7mEj/ABq3nagFi1Viv5MP+SG5BEKpRlMqdXWLLPULbdh99cS04eQLe0++5OqcJNYBXV2xjzA3rr+Z3ofdYKc
dONqDqMg/wAu8GVbqWecfqNITnhgGQ/dWnNQ1B5f3wL6dSnLEftkdhEHiWzpa/qKd+85ymYGG5eOgOz
XCRqnB2R/Z3WVTWnm2/Y673TdXgReC/P082XipM+eutj4GbQFLQRC1OxG1JMA0Szxpk8aSIIX602qJsfawv
rT8KJU/azfWPN8i/Oq+ShPmH0mK8U/1lwzPpMWYh9cqplpHou2P4jmnzEuP4jinzGKvYRt9x5XCYNXChLaH
RWSJ6hto1UQKIYaJb0xxERm+T2eET2GAYvuXmSJ4kti7ED2RE5eWxsMuYhZmQcMxk24RSTItHpEYRFpki5
o9hnD6avRPbEOofR9EjtEy0omkTK9Yhi61PSY8sEW3L6TA7UGUjnfURoGVmgnR/Mt/UF8yr1J5/Mz+txNl3S
vymjniDKNGGWDiCdZKeS5LWlek6TOM8rOaKZ0fobcrDUR5iF7/ps7VifQIZZ4gOJ9Ahtnibyn6SMRb9RgYkyyE
eZuGkiJLMkYTYfYnUcNxlJQZDsTYMIVEGFazCZD3TUtCZl4NmaLLjsWfzCy0EsA0/4gZBYdn5l/+oTpoo6/gZftd
SVb81HRnynfWLANqifoH/6iumGzoGJla/4Wzpv4W1Z+eVvZfL5PIOv8o/pHmlfhtBLvUuW1Cjstv13+E+o21d
VoX5j9jHtK9x1K/IY/czuDyCTA8N/kZdb+NMGEYugp1NfgXoslXdcX2aP0H+HWFdw7T6bBO54zhqRPnfoWQ
lnTMe9Tw2mB+xnasDJ4AnzaziTPpSzLH6LclhUSUZy1FpsBQxdazbg97QzXBe5aTkgyVMxQZShe4GoJkdRKD
tXloRahpjWy00uGZ1oUBu4gkn0KPMo2RXnZ27XIHydqntPcHBtyckM7kkzr+NgUpR2641D112WvLYpZfVV0
ssTdK+IOmPQqM4VtaKnrjKjGQHAV1DAzmfXOh/KuaxBtSdysVZAql3Xc38AZktTDCSUI0djp6bE5RzyfSjz+8
giVv4rD3YeHanNid//wDWIOldRFiqFpsY/Zjx/wBpF1b4g6cmVbjDKosylQGypD3dn51wJO3U+JU1z+RVUeFd
FpYwx30Kxryq88zvvTKKsWld/UZ88fBK5F2cGK8HnX2E6Z1Hq/dnMEb0p6Rr+PMNoLUq97WWnhFX8WU5
2qtPCayzqzZSa8xJZm/1fM5+esN2zSnqHc3mXX8rODN/xzYdiw8j5niWmvwJz74cYubDOiKJotG3KvJQalJW
NEkjbxJfaQP4MffQoigdWs3br7SvlvTGPuT/Xf8xYvImQtebJGmqWK0Ic6whDFeFZt5Ysusdh2JXaawlm1is48
DUZF9xz6RHNPmlcU+kR5T5hKvYvt9x9XCxA6oUJb19lqpdjSvxChAajDAZc1vyoQl2bzeye90jHsB60DK7MmL
z1RAYxJk1IGKoAmxE2mGEwkiOQURzMkxEyAcAmTh2HcGrEKsijpqN5QjlxM0aAU2CLbh6TG1giu4jtME0ER
Quoe8q1bDbfmWjQJ8yoIOWmc13rNBpPSbWNB8yaxTBgDuUsmXEQpTjDcPKzntYnQuh+V6X50BvXy2
duwz6BDbDxF+Gf6Yhlh4m9pfl+jDWr5jBpGX1MBglzaE5N4R2K5NP3ID+ZuMkfeU7Myu1vMATNf7xSNzG
nUjo/7gfebC8ShLmtJ1zG+8MrwTqL2LhNvmCUxc0/eMa8kmHVyleEyxB5C7QJbONzfu51/Bnd+TigcM/Wbo
xzvgXqioD3JT3qB7ms98/l5iRaQBw3Op+4vXcdcjpmVQ2+2ytI4/kT8QuvYl2Jm5FXh8e96z/8AE6jujzOP6Z61
PamA2+gg+3v/llfurK2kfY7B/Ms1b13UgE63/5iPIRwRscgkS0yJSxk+iP06+JB/stsS8Hg6Vvl/Bn0bOnqasgQnlf
B+4nwf8L9TXD6lp2/p2jtM+p2uOO9dtZJRtMv+omM+16ZRulherlG2+G6hzoim8uHDPoSnJP3JjZdOu5y7pn
UK7FDA/Voy+41400ZnOnhmnjhxH1WOjzkXxVg9Xe1MS01W950w8mdepuAYGH2Uo9ndoeqEjhSi1jKBO
Xsz5a+C8XrGZbk0Zebk1ZfVhd2s55bf8A3WfRx+B809NxmrrzrbYF7vWeCZp1LouHfUy2UhgQRvWYXR1vqd
FOLTvmVoaX5+cncLFHhGI5H5I3B0ylykmwH/m4XgzTSfUlyLk/T3r7571jM9CoCS/qHMzpnwP1k9RyarpA
q00/tJ5bkcS9Yfx/fj5ly9Qw6wnaOx8Vzbv871JT+o/Ti1rr0/L7CV9XoBoh7gniFnRosLM8PPKBfyP+oPMv4yac
FhplpPxJ0Dop+DOod2Q4tsxXqAX0+t3CgaE558L/CCYFNVbLp7F7n/BnXeI9Q6t1vrNd1yhKKWZgij0jf335aX
SjpYyesMfFa+T7ACVV0cqMIN4csY+4WVtI07xtu/DILH+K9ohGKidJ6FbevFtg7KR/J95zKm69PJJnQOtXnKvG
hqqsdty/j7/AN5VbcfQln4CUEl7ly3jOdkpS7kwcZTGMsOw/NHMr9glh+DYfmQCypBHhxZ9JfCo1jk/czowlB+
GF/3JD9xL6JudD/68TD6r60v2SHxIX8GSyJ/Bj76FV2ct6if69n5iut9Rh1l/7xZ+YnUzFWyxNmsqXy0S5HaazqIU
A74ytB1F1aHviM7W2NRgsFnxFAj6nzEWPwBHlHmOvdlRt9x/VcxA6oYJcQ6KmfZOjahgeBqhk/aZYvuSQR
LSQu0IAkKwgRilJmpUtCT0z0QqSyQyZl5KTbz5npM6iSZPOZk4dOF4QQllidYRK1iZHohLZBlmVckkaJRbZvfaO
ZXr7/MnuckGlrGx3FN+WMqGCvZtmzENf1tG166B/MQqx+Y0o9csNF1o+g1lgbgAwjZg7mUsi2gao86H0JuV
nNxcrc8A0I8rOU5Vi03r5b07YZ/piF2eIDhH+mldZ4m8o+kv0YW5fMYGpgWsfTCdwDjB0yNnpJQ9Rz7qBPF
FgJjTOPqMTs4AITJ4ZYJ+UJDPJQzxN+55h9VuxlwnkGpJsYK79wlixmbUrSMNiWPGI4h0yXBZKh6YR3bMcrPE
3L6Edg8IXccsX5twCEbn5Pfqz0kYfxp1DsA7Lwt4/wDnwZ+omdkHmfC368dMRj0/OQcgvQ5/PqEZ0lmNQvz

wTsr+T+uT4wQsjBT/AP8AESfLVXrWzf1cN+fvJMqo6DCb4hV0dT4YbE0aKh5zgrNg7ezjg+/3n0v8H9Z/ddJFN
uj2bH8ifMtpYdy+3nUu3wPnNT1ZVDEd49P8n7RDXUKyh/dcod+HXuvUr7S4Z9O4t9uHbyf6ZPE6p07qSOoO
ZzLFispUMvPbxC6678dgU00+0wNuM89n0arKXHR26rJ5BBI4w8LKxsZgmL1Fu331Lbg9XCOOeIsshHHJ2V
l2srGbgB/KgiNsLQqsQeqWotKm8gR2KU12L75QeTjzdORSeX/wDMOxsfEUjdbMd+/AnVbMehf8IgrVoDQ
/ynpUzT9Y5/PsiHGZDTpOQoxRXVWqAnwBLVlkUYvyV+uzlz/H2inpFKmzZGIxk/gQi1jZaznyXlhpK3nc+cclYH
xCxb9q93lic1wG6jiWM1z1enX2nhTLGUkkV5ng5zjMTwJidPyPmDSHmdkxegAa7hLRT0msa9MQe+UsKD
BS1kYjTovJrW6x/EtggGPWEQAQ8Tc6aOKYL8GatImbZtIrPpMlKnh9JjUuga7OR9Sf/eLPzFfM36k/8Avdv5kF
JmHseZM11a+XH9IMZNIa9mmjdeRIGSKSjyFjlKpaPcc8yvpHeKeY3UxS1cMs1UOTzAKoeh5I3V7FRMZovEk7
Zoh4k8u4pYK5mgUTwyYTQyWfg5kjBkkGJ5kgaDUiWCUyH3k01YSbRFG0yeA8TJ3g4fLONvmM0QkRfi+THI
Q4Mx9kcmphIDNY0YquQR+48xNclKEVkm5MpGUvB/Mrir/VaWrM8N+ZWQ2rTKX4guUXOifBMUOoM6Q
7uGoO4IFLBCrfIGBoy79Eb1CUhmAlr6G+2H5nIPzonYvls79gH+mLwtPpirp5/piNLPpm407+Uv0Y9fMYpZor
vs4MPsHmJMo6EhZnBKHZUc1/UZXL8hRDep2kEyhZV77IPdPCZoc2kxkcoF5YKLD2CUDFcmYXqk7ri1Hpb11
dZGa3ciWrFs9InOvnEPqXHF9IjVM8tk4NtlzSz0yC6/SQIXaSLLrtiPZGlxFGdeWOGZwj9Tsjc4PyxrZTtcfkGdpt
PdYZUviTj5nQOo0KPVZQ4H9hDV5Uil8hJY2tfg/MC7uCJ/JAMR4W6s11P+Eltf+ZaMmn0dxH+KVvJ3TlfM1
41NbBpxyJ2JqSyA5tesxiF8yTEVlyqXB8Mp3GOeiMtbL+P8511YrFJ+9gPH8cT035X+jlcfmLn3Prz4f7cnBTwdi
WYyx2Vile+CKA/SMffkpvf2O50x8UnyN/Yz5nqsK2a/LPqWmeaofpFOOGynY3COVTww/ui7NbD22J58hT7al
duaHcDLp5yq9djhh9vBl4xep5S+llaU7DqG51Lo9Dsy8yddj3C9mEnlInx7brEJKkCbYgyLsjtVCQT5nVmxqlorX
sXetniAldTTQ0knGLcu0ZafxOKckqzUSjG+Up2x5dv9JClfcdCSah2OoDCWEYJJdFHOyUpOT7Y5wemodFuZ
dKcKtR4izCYaEstbCXOnprx0Ut9k8mLQokwrE3BE33LBU1/ZCOWYBN5ruew6WCJ7uCXuAhkjuAJUuo56qrD
cV1F6hBjFFtnJHNOo2bzLfl6bBFORf3Wu33Mhrv00x+cs1ij5EXatxqEAblfpvjuiwFhOYywTDkpEOoXTTQqiv
6o3GKQnOThIUNEBq8Q4Srz6K6YzpeEj+ptGMBLmqWYor5rDPdzUmemR6JhW2QByeZsDJjWJoF5gNssh
MrAUs9Ingm0ZBHnZMnsyewePljG8mPavelMc+qO6j5mTmaWJs8UWiNWMWwUeyZTM0fVKid/OMuO
cPqIPb/Al8pPiXsXWhC9SjgdQgTYruZ+SyXSYmcGWbof1j8xS9Yjvo41ZIQhiaJzImDO89O/wCGI5ccrJ00+gR+
w4m30v0kYrU/UYIsErmZ4MtF0qmceDJWrg5X2c96koJlPy6R2nUs3UbtPKtIXaG5Q3zUcsYkOogWKgVpaFvU
LwZzu7LbfEZYeUzgCva1XOEvn8lbsIshJL8GXTBJ7BKlJVEsDlPQvastdPBpZZY1J4yMLHPbFnrMV5h1jemA6Jl
jBDSZAimbXhKq3cnOHgywYHSsrIBZE9I9zwlbm/D+KtRe9rbeddtfAEe8G1wzGP8AblwnW5pOX9Ls/Kv4s6V
+26vn1oNVtYbKvWfacy6jSjIrezlR/ccz7c/UX4d6df8Au78Kkn9uNtZ8zYYk8p+Z8d5NYAYa+lu4fj3lXo5y8JKXaF
ddSo2vGVkqNb99KjXgEf5cyyW4ndjUb5JZJW6ICZABHD7H+UvHzddOrYD6AOP7bjs3hFdT6j6E/TnquNellG
wHp47f49jPoWupXA9jPzp6L1bK6T1anOr2UBBsUf4q38/5GffXTer4l1VFiXKRYO5eflMwHxXTOq5SXpmb74
ZqfFpa6lHhJ3D88a5gK0ckES2Vsti+RBrKBvYEzsi8jBxcf1idk6BhbdZzfCp26zt/RKe2Im+wjWihvuiVvnx3bT
Jjm1tkxa0PZYBZwZumYRHgEU6MD7pIH4kUdwPKc1kjavrAHmU4PPO+GjbJdMC6oP2Ohp1io+8PTqdR95y
4eZOO8e8MtTaBelrOrnnVH3kpyk15nJzbcvhj1j1DJUeYX+ZNewL+GvuX/N6iioeZyHO6m1lr+riC5+ZkPvmVjT
blTqbpzKw2mojBBb3czK3Z2B27kiLpohzksWvKPabCl8pu8SshhoSdL9GFFWi+1ZGxGuO22IKxruJbMJwTDQ
byhSyPDLdV4hggVWtQwGXEOipl2FIORGgiyojYiWtHQjZ2bTdVms2jQEyaam8zU8eME9nkydPGTJ5ozJ48f
KtH1R3UYhp+oR5T5mTn0aSJl0WWxuRFlogl2EKfnD6pSbDq4S8Z/lpQMg/wBYSk+J9lufh/Yxv4SGipGh6+Jn
Uy8aNNMa9KP9WJ2BjXpf/Fnk/MjzXlZ3jph9AikPiVjpZ9CyzE+mbTR/SRjdV9QTZB8ymZ7yy5luty15b9xnbpH
qkUPqY7niRsYul46m4DQCu5dCU01FvkZe3HJX8nDUDxB8WjssBBliY2XsiCq7doErrqoJrCELoRysHQMFIIesQ
YdsrOF9PEeA8S2qztQ3XnBuz7lw6X0mr5Byss9ICjYB47oiwasZF/c5R1Up9Ke9rfYRR1TqWdn27tfsB9Fa+AJ
Y1yhXHfJZb9Mf+WHjTZbLbF7Y+8v+EXg/GdJTVWMygHSqSF4H4gD/HhTi7EBHuUb/Qyk1UDWIRiZMeivd9a
aEJ/O1jXE1n9cDn/b/h64cP7zylfi7rXw51DouaKewXPUR2ORWw/kb4n5z9cwb8bOei5ey3uJUdvaN/8ALzP0
lyfgo1yEXY4sB8gyq9U/Sv4UzqBXbivx9JW1wVha9dJPM4c/wCkhZpINyhNtf6j8xr6tyOBwNMB/7fjKw6p
vr/wCk9s+nfiz9FeoYdLZHTHfKReWqbi0fyNcNPIO7uoy2VhrQB0f4OjLiF9dsMxecFK6LkP+ZYTH1fTrrcRXUb
XsHH3B4In0H+luSr9OfEcjvocNj/A0pnwiuNY9WLZUHS0j8+pdaEvHwNgY2J8cJ0/IZkTJVxTb/PJHn8aMpdBh
xq7Kv8vEv8ARYqlXbztacZH0ZjPoaMb7BAldCvTb22bALMK7P8ADYF89u46pJLDcWvILGTjJYaNhhNJosuBjd
zgzWJX2YwH3IG6DQHnQbgyMFI0AOJoPhNXO8ynxW7nYeMOIrvHEOL8QC08TSszqQtkoMj7huad/MiiR6z
amgfmRu3MhYyJ1DZGhgYSvq+hDUtGpJM40MmYQG3Uw28QF7Z1s8kA3osQtoNqPHelsmCkuBiDPQQJuX
Goke+ai/iIN4Y5nKHHZpstks/MhdTTkXyDkuC1UW61LdgZA3KJWToR/hselxHsVmuDqFF41DfnCVKmxTQ00
0sYzeCslBZLVtkDujyp9zn6u0eYmWwOjH9Pfh4YnbV7otoM22YHXZ3QoCXCeSvaPZtMnsmcMaaAzaaGcPH

syCG0g+JkH4kSW1ny3SfUI/p8yu0HIY/pPqmZl0aJdhipi+6HmAXQSJIM6h5b8TneT/xlnQ+onk/ic5yTq1ZTFE/
SXXw/1BKRjWYqBhdBgTMI8xoV2IzwF1bEwuGoywLf6sIsZBvOGdx6UfQsszn0So9Kf0CWW1wEM1ukkvCM
nqU/FKj1Czkyptttx11C4d5ibfBkZck4I5t8QWdGmoFPVj3a3xLn8TdxBnF7K7BYCspL1P2EdU7P8TptmX3LwY
RiY5Z+6UzCyCWAM6b0xQYOJLfj6iW58lqwqtKI+ro71dj9KKWP+g/JgFK60BG2e3Z24aEDt9VzD7n2/tLuCiot
vpFnWt00kJVqsvvNvYquVvWI+IQvgLHtGBQp8dx+5kNwTALwBHtI4EgpKUsIs8qOOkF01aHAhRr4mqNCwZ
YQxgQm3kBNc3F5mFGeDgyE0jsWyAYq90+NP14/TCI8U9bwawpB1kov3bjvn3AiwTqnT6M7puTiWjaX1Mjf
/KAhN1T3I9LE+JdH459JyX+SmnKsqLsj2I43Ovdb6l0+/wCFumZdXfTndMyENy7Pc6OeXB/M5/17pL9O6m7l
vCWtXen2ZSQZPi4f712xxf2/NXtVW4DqfHPsRLO6UHtszx7nalTjurf6P0lwMet+k4v7tvm4Js+dVfQNKVcbC
2jyvnkibZnRjiFLabPnYrgFLAd634B1OH/AKZ/GOQnQq+IXZlMNI4jtjv3+ul+3x+DO5dBz/2GVZi5nTXr+aCTdVz
jWr/AH0tKTU6eixOL4a/z+xcVW6pJTXOf8PudC+HWSuixz4Ubp9uZ0hxTIYyOhBDoGU/mc6opxUxLv21/fVYr
BedsVhic+/SL4uvy6c7p2Yyi/BbtA9ym/Mn8ObrxTJLrsqtdQ7VO6Ocx4a5OsMWUkHyIE7R31FAuRseGG4gs
MuGU0Re7AGQNZNb20YOTxOEgpn4kXcTBw+xNA/q0Jw6FKx3CAxEB7iJt3nU8eD++DMywc2iC2Pz5nGSSN
ridxbcfTJLLG1ALCSs5kmK7tGABtE9+0xf8yJwXGIMZh+Y2xzK4rxxQ+ovF8k5ItiNwl+wjKmlksmA29R2InPov
VHiHCAUeIel7HorpEyiNKVEVAw+q0Q8MZAzzgtuMOlyETYVoZrDuDL+lpwRUzXmZvMnk9hwZk8M9mTx4
hKiZJZkjHj5GoPCR/SfVK1QfSksFJ9QmXl0aRdJSA3QuBXGAQQpvUfqM5rltqwfmdJ6l5nK85tWD8yp+JeguP
h/qDg/E87+YCj7EnmVwzSLAetscdOY/OlfiP+nsPmwkVygU+mdx6W4FYjPjydieZUMPJAUDc2zcr0GaCqzE
MGdtrzMRZW3ZBAMIDeiV5VJtLRr3Hth4yygTjgqHWag+5zS7CIBOp1TqC7JlVvqHZFEZJRjtZzuolb9anUel3
aAAIGtoAtBEtWH6FESi2pvCKuPraSOnYvtiiy8BT8kBV43BabWZS7HbWMWMCtsNfQxz6rrCT+F4ntJA0PSa
I7bnqMY/jL/s0+hpxXua5blRQY6rfxK1S/iO67QBOVTHLjysw1WERLbDUsMsq5or5wGJaYDswXukqmEmwa
Q8pHE8tEjx2ORC8hfQTFrF5AKeLD4S/WD4cbD67+/UH9pngCz7Jeo/1ny81duNkdvi7WDKBx+dT9TPiHouH1
no2Vg3j02LwfdWHhh+J+avVcPIw8y7AywVuocpv8e4jGkt3xx+OUNzTSX3XTLn8K9TCfEnzcl1+RnUim0t4Lr9
DN/4n1/hX9W6ZilUrXmXcVFLcsB/0n3H8T89qvmhTTYAwbwRwG/IT/hefUnwF1fr79ORcexM35I7baHPZeA
PcD/HFNVTouXtG8cYH6LoTrcZRTWctM750snMxlzMUcu5H9dPfyfEhu6Jj4/xCvVcOkUZ1an5wAlpyKzww3
7PEfya84L14yWU5CH66mB5HIXH+hnUOm/E1VtaU52OVu8Nx6WH3nNPZVN4fkl2n7ZPapWJboreunH3S/
wCUXfuozMNBefYP/DOtfkGVA3jcvONh2VV6J36i2vzzEfWts7bl8P9X8GaSyEtiljnHJjozj4jSfGeCiZJgfdJMo8
GLkYxTlZgLBkBCbPsIHbOHV2FtbxJVckaisWCS1sdzmToaUMecOIWW/mRs6z2DqBB3e8XZBMMscDwYE7gi
RJFayATF4Me3KTuKTX6oGSygyWRq5DR1S3ETsg35hVLaiLTTGW8os9by39MPiUKt5dulNwIzB8ili4OieIeIe
oPph4MsY9FZLsnEkkIMmTlglRAy1dNTVe4/EVYnFYEaiaChYrRT2PM2biezUT2Mgj2ZMmTx48mT2ZPHj46xj
6ElgqPIIzXj/SSWCs8iZV9GI9xxuBXHgyfcDuMAiZUupTkvUSe/8AvOtdQG5zHOqJbx7xDWxzEstHLEhdWxhw
ZtTWuriGCqZx0l8rgP5zCHYeUVtnhxgZoMchpKfJCdx0jAyi2uY6t27Sq9NqIAlrWNLhIRfLAzX65K66E22O+a5
TgLH4+gUl6iuZfMquWR2iOcy8SoXOxgOyLXAP27aWbGo9AldorLOJcKK+AJ1QQGNSXJbNZGTzVYFCVFFrA8
6I7IT/ADzGyWcmJes5fzuoZh0AEyDWAptWAsnD+rcV1T+dP8PH+xptMvIQ4xwWqm4CNqrNyn1WEsBLRj+
BA1Nth7Ekh/XGSRXUDGtSy4rKmxoLUCFKs9rUQ9FEbUciUpkaRqzd2M/3AMhRBD0rEn4eU0KynymUPHuB
Yz5T/W74UL9vU6F9egLAPftn1X1LCfEyQ67+W7f5RL1vEqzel21MAfSZSVsps/MWXuK7lpr0yWD8sact/lcK
GAHrqPuB7rLb0rrFleQt9D2K1RGnXixNf8AMB5Aif4v6Dk9M6hkFQewHf4HsYm6blfPfurcu5YHpK8C3X/hp
pc12VZxpaK6PiVW4faPtr4d+P8ApmbfWnUXWvKYaTIU9hf+O8cN+DOzNRk2qAtyPo7rYqUZfzY0J8CdLycLq
TPXk9tN3hiUHy2Pj1g+PzOzfd13V+IstRyba6ifR6w1f+T8iZnUVquXbTL+rE45jho+4ehfEV2XQQ9ai2klIDA8i
PraD/UpavVnhZg++EM+Yq8zrON1Kq4CrsyiqWsjalYeD/HE+n8DK+bWvd7gHW9+JqPh+od9WJdoyXxHSrT2
KUUtsvb7HK83Heu10PkGKvr1Oo/EmGB8u5FGtdrTnbjRnZ17JtC1dm6CYI25EykwppCYIKgYVAGenX2np8y
YLxOHcgFjlyGbwTDbhuV9iVfzxINskhh83nxB3sO573jtibJulkGSClBf35gnncVqzM0NCnUimdPGkC2+ua2tzf
/cdwVkQsZFrqsl76S/AnL8eydF6Q3Ag4dnLOjqeO3phwaJ8ZvTGlaWEXwVklYgGwqo+sRcGhtJ9YhovkFJcF7x
x6RGQj3EPoEaCaWr0oo5+pm88mTlcGezJkyePHkyZMnjx8X4p/pLLFX5ErGlf6Q/MsSN4mUfRpl2NielI4JkgJ
MKSsa2YBZbDJFevxiZsvA37e86b2Bj4gmRhgxjB2w3lfpSTRYkzEKEBBQ06FIYg0eJRcurSaUtleGpmgdZOpU
xT3QhHOxAqJxsvGFrtEbWppZX8Oz0iM7H9M8yKpVs9UgzbPRAhbppBmWgpGVP5YBrzFUzHO4tXZUSTKs
G4FXb6BzOR6iscY6gOJa8LsbKpVjoF1BJ443KNVePmDmP8ACyQmTW7L3Kh7yvnYXkw0FmS/ZHAIZGr/AHG
UU12tkWkaO/8AEYxps3Uh+6iVit+/Frfn1ju58+rmH4VgIKHyDsSotlmyf7Zp1HEllyxj6gZcmfhdMUF8iWO23
sphKlhZAWvJYqbe4x9TKnhH0iWiki1qfBV2LkcpDVYRYhhaGPxYJDDGfZidGEYVmGFJlly8dMjHeth5HE5U
O+ux6n8g6nX0nOPiSj5OWlo8PwZW66Hku/ddj/wAOs+Y6/Z8o+bfjf4eS/VwrB7fIPuJ8W9b+DcvE6ytOL3M

MgfNxf+v717/5xP0yyqEvoZSN7E4r134Yod0+7FQ9I6OLsWzx2WryP7N4Mq6NXZTPH+WRe20Qtr5XmifKfS
Fu6j2fKJqz6Nq6kdpshg7B47h/PBncOgZ91dNVKY4NhOmwrGAl+vP7Zn8P96m/tNMz4Tt6xfR1jp7nFz7NJcp
OgmTXwyWfYn7zpnQMTpXxPg5WBn1HH6pjaXlpK9rqR4bXuvuCI5ZZGxenK917xYGCdfvz9/uAucTqPTrRh2
WlvKPS21NTD/CQeUYS7/p78Q9WruWrOvBRR8le7ggr9/zOZ/EHw58TdHy0tIsvdRpchB3uUA+m3/6ifY/UJ7
0T4lp9bWKO2xe3lp/j/mG/tEq52ae1Si/LnksZ116nTyi0m2uD7zupGbgWKCpAIM49dW6OQ3twY9+Arv3HR
nuovZ6jkMqsPsPYrCPIHAau/wCZvYfz+Zr7LPEjXNL1RMH4XhXWV7s7ZFP2NyQAESDt5k+h2+YAmDNXswkK
AJpo+03IOpxHRTeOTElipvce3rsRK1JB5g2ERAOFeAyMdyY8qpQxiKAV8TmMnih10OrQ9kbUbX0agOjIJJHhH
cpiwg7lgsrgBpJM8+giB6CQ06T0h+BOcisq0uvSLRxF8YkSfKot4z+mMQOr+NYO2MlsjSYk4jVWhIT+oRORwq
uwbEmpA3E6bgttBHIIx6bZtBLOJqdNLNaM/csTZJMnk9jYayezyZPHj2ZPJk8ePiXEb+n/AHlhqBOPVMJ9o35I
0xFBAmUxlGm6Y2rr4nrvzoTe1wqwKrlotJLgapjnkY6iHvUO2C0RmfohtqwHcnkqWZSNgcY6rXoGdczNaM
5X1gitaU2pSLCvLRQy8mS0RC9+iZouRK9dk2jpGJcNCOHsHbKNhX+JZTZ6lvN8nUiGy4B4tzMlfi+YFm5IUyr5
OWWh4pulJ4TNMvl5glVxQCQV/VzJEpJEdhHyik3lhtdxDAy9DF6nrWMG+7fj6TKAtZEtPw4LV63hlfJsOPfkjUJ
WsWR/aF3nDBdL8tQNeIKj/KvVv55kr7U6bggkGD2AFZnJvFj/ZulrMF+joWL9cJyriSi/wAxJ0zIDYwO+QNGM
8dTzKCMp8JL3K+Sw3n2L1hL6F/EstQiTHAAEeVGW1awirmxkghlBg9cPURYApNnqkxrSYAFhdcaiKzHaGVz4
kxfnYBI8qdiPajJrqxZSymcuhuqIH7oDVNwuhL7M4LVYCNQW7FDHcddRxfkZLADgwdBsTHTg08M2sJppSXT
EVXSuzJsurTa3cZYFh1EeLF/6hGmf8M43UnxskXHG6jff/l86oetf+lx4dD7qZYcdRHilPPvGKMxfDer2n2JfmZ2
RgnJdSxfl3DxdTt6H/IT5T8NOB9d6VUUdvljvQ/VrRn1bRaZQPitPtZdn16bBo/mH1Ed0MpHtDd4dm19Hv6P
5fb8N5NAr5XJtI/uAZ170xjl9H79L39vce3kbH23Od/pfjV19EywFAYZli7/ACBO4VowDe4mx0UN+iqy+NiwZP
4nZGOuvcVh+lfNbdyZFE6R8T9MRCmQigdx04H3+858BoxSyDhNpnq5qUuWV0YSDZMY2PAteXnBhQV6j5
gxpDR72Ht8TQVakGiSKJEqKHwoarRq1KkbEAsUId6OZyA31kxWauY7JJEgFRJg2ySEllGz4gLOMPaXhMfc0fD
nGdTOe3YzEx90/HKgrz+xO/EZU4utQTTJqQXSSfhq2GaLXxJhUZBto7hBSOYrQKus7hwqM6pMg4ouvSb+
BzL9W2xOUYDFHE6Ni27UTQ6C/jDKPWWYIIDIzNAZvL4qjJ7ueTJ04ZMmTJ48fCeEfrl3xbgEEpOGOXliqJAEYL
niJqHMHh/daWIEMoEU1DZj+hYCGXfSSiNaodv0wVBjz4jnsQ9xBnNwZyLrVuladU6gfM4v1+zSNKLVpKt6I+U
5XfkEWNz7yBcJnzFV9v9Q/mDi3mLpHGdLwb+RLa9+qzvpz7lIulZlikl5iXsKs5+73imuokeZYXxLWI9BhH7Fgv
0SzhriJXTI5it/J5h9dREbp0+9vCQlenZC+VhAQAtApKR50tVp6hjWf8lqH/vN6aXLa7Y4TAc6kklZSOrUPR1HJr
fytzj/vFRbctHxVrenVgsgdfnltoIgv1qU1XJmf1UMXzX+pmw0091EHn2Q56bkdljp7EcS/dJGz3Tk/f2WBvsZ
1XpTg0gj3E9S+UC1Efd5fKwJqoyr47yxUvLaEionEeVxikVVGNAhHqxGYYSIWelsJCR6KEZSjJgSniLVEKQw
mBeRRviXE3X3gciUOK8TsfU6hZQR/E5AaHrsKke8zWsqxdk0+guzRhvldSltSwUuCsri6AEa0vxF4LDDWcJgHR
kZWwuRiWVsPlmgeY1gAhc4F0uUwP8ATihF6tUzE2rkja+2ivBncFsAYg8czlfrGgVl6teFIFttY37HtWdtdWc
BkbR3o+4Im00CktFUzB4i4S19v8AXP5wbX41V1LV2Ltw4M4P1jpt2DIlG2UPKN9xPoAKlAdcxR1Xp9GdjGpv
r1tG+xjGopU48epdFfta4S569zgDcj3niKSytxbabGrsUhlOiiAEEOi6TRsVOpFrnkwxNfaasnd4njxHsdsCbncM
CFZL2AyDOpi6uvmf/twDJNAGFjlfEjkkR1ooh/yll8QRyYqcESUQcgNscfaQGsiOujBYv8AE7KPBxMVqYSDIW
Q93iFpUYmw6JOEMAg61mFqhklEg2TV7B3Ljg3eJU1QxpuiVaNOscZIVujuidDrbYhAijGtBURoDNZVNSiignH
DJJ5ua7mu4XIPBJuZnJk4dwfDeH/xnEs6i00Sp4p1PLIV9ImQlZFGsh6mMKFljoSKcZJZKEha4hpSJ0SbuvELVJ
66cRiUeAUZ8IF6iPM4T8SEhGn0H1FODOF/EIw0aUWoh5i9on5T55scmxvzMUcwiyki5vzCFqkHFAdxaulnXO
vdJw0dQWnGsJ+0CdO6T1ZEABMWjFeLydsb2HXcfoMwHEy3o2PviDYXV6eweoQ1+p1E+ZdYhgp3nJtR05
BxDLOm0MORB6s9CRGaXq58ySUSOZCYdLoDiNVwKdRxVXUw5E9t+Uvir2Ht5xb4+oCV9PYDj11k/9xOOBp
9JfE+Cmf0e+rv7WQfMrb7MnM+ZaW+b2IPDciUfxGt74y+6x/san4XYnS4/ZkzpzPRbf93AijA6FZbosJO/pvRvl
qB2xSmixvOBzU2w24yeUMZYqGaFvDM/iO6cDWUjBQ080Us76yGjce1Cb14wEJ+XqWnDbRW2WJkytClab
hYWGjomwkSQ1RNUyeBZNxUMupS+pYI5IEuujAsmsMhiV8FKI1RY4TRy0cHUJrJhGZT2WcQVToSl2cmg35
QxD6HM8ppvy8laKvLeT7KpuYv7rbLErrQvY50qj3M6z0JaQwESo7a60Eu48ce34EsdHo3dZl+hdiOr1Spr45m
15V/yWHCw6cbFroqHpQa58n+TCK6yqudAEkmSrPdd3aVYeQf4Im1UEkkkYnfJttvt5bJEA87gzowZSBzsf2h
Y1zzNW7hOSjwcJzFS6/0wX0G5R/UQC/yJyGzYM+ijpk5E4d8R9ObFyz2/8ADflZU62rDU10+yx0ln+D/or62jf
mFKSfEUVDnkxtVqVZYSIHnkSfsBEiYAiD95UyOTuDCjmGUIK2sA53CarZBdkn0GwAbmV2aM8f1DcysczuXkG
Nlb+Js3M8r1JhCZIC4jnxGNCiBuDvxPEsktAdSc+w9FYkoVRFwu4nouMLwDwxrPA0BW2RmzmepYLfiX+
OZaKn2JzSm8gy3YeSDLTSXpcMrtrV7lnE2kCOCJPuXyaaKoyZPZkkcPhHFBOY0vIFfAIHwn/3xhOhKQFEyKXkj
+jWLO9/sZ0+ZZKJV6DzLJQOLWyc1wOVm7DiQI0I9o00Le5W82vYM411/CZkbQnerq9iVHOWA4PEr7aclhVd
g+LsrGuS9trxljg/afRmd8OI5J7ZUL/h017IESIBoNvRytFsGul8xkYjzHLYBVtEQyjGIMEocnJT4BVOQifUYdRI27

5ZoYyjkPK8cE8QjBlitGDIE167pZcTJIIBaUdcexV4motyFbzxPKTTI4O3Y2RXr6pvZZX3eZzTEyHK77o8rtB8tGlyy
Bcc5JVW/Gpz3i+E6a+sWuiAJYfmKB4BPKS84dhIAEtVVCNokDf3hXVGyOH98k6rZ1SyvdYK7h9IRFHEfV4qLG
5CgSDfMpgGqESErpyzklkdCwv5aie1+JDfaBDtJIVTk5HJEtwaMUtKcw+h9wUGmw04NIOCzcTYEQWywCEsaS
AwUmxiJfAiVj90N+YdXkRaFmQk6ZlckCDOvBDOWuSk7E7NcAllFD6mkqyF7LFrrUs7HSqPeXzqONdbta07
mnnS8DCwWx7i1gvt3WwP3POu0faJ0aSdInPEc8suHqYQp+88cfdG6RThL3OwbldfU32H2WWHbLYW9J00
Bvn+YPdiu3rViH1xqL0rvGSfomt27q0NN3MT9zvgGaqElwiklhLozLbtm5OWW+xjfmWowYdugNuPqPb7FQP
M26TIYVisMdhrcfw39Jb2/iJ6ek0JjMDC7MA2rC2iob8cRf0q29Gwx9QSwT28rrQA2ZHe1NDTqqdM0u0dK9
W5sTqVe65iastCQRWAUY8Hv8fg7lioexqllr2t2juH2MaTTbKidbjFM2A1xKr8RYQyenONbdPUstJ4gd/IA9jA
WxTrIEIW2rEz5oAYNGNT695P1LHrqzLqyPpc6ipQfaZdpo0SaaHnpli63Unrazt9oryHcbkMnUSdxKmYl2ouRi
d8mRrvC42dLbTftZot3MvYqsR4h7VEc6knnAleUXAxsgBITqftMsnVylfMnB5ISDXrUiKbU5jMWRBbSJOcUci
2LhYQdSUPzBLDzxJ6gGgAuAr5k3V9yRKYclAEEmoyZDMUDLGMlcQ08FQ1PQujCRg0Dk00XfFt2BHCmVLEt8
SzVsCJodNPMcFLdHEgmZPJkeFj4CxmsTqTidGWz0iUG4ducTLsLdBZipTwkbSuGzf2WzHeWkl+JRcW/Zlopt
haphbayyo8OR4gSyHo8sYyK+URqQDBLKQZKjwkaMnjIDLrW7cMH2lczMAFTxOjGsGLbSCH2i86gsLDhWf0
7zxKyaWTyJ3bKwVbfEouf08falThgaUsnNiG7vEOpUhwdrMmIhoQ+IwXYMXaeMndxvWysPEhen+IIO9W5
MdVoHTlpGPJ7IFQrq4Illox6y2iloFHPmWzp9BjEUSiPcOsqBLGl3aIBTRXJ3TUcimkee1smbIhNB2YgZtGNMe0
akoy5PTqahwh93aER5dpEMa8aiXJbukrHIA6a8S5AVcl5YKH0Inqph3gSMYSSGLNsngaNkaEr2Vnckbg2Vk9o
MqNlzO8rb7ZN4HdPpV2y2U3lmlooBOJUOnVk6l4qTsiN0RxEt1UluwglWMMR4tY6ki2Ts5pCWzKGC2MN9
qgnn+P8zAL1L0/OreurJ7FBdtMK9ydiriv+0n3iY6ITwHe1VRh2WI67NgXxwPAj1FqUSEKIKzvHP2zkbYfWcrB
xbLesZFNy7xXXwFLke4A+8JzMyxs+tyyql8chSzAL52DKTriVftaxZWuYq8u9mztzwTz7S5dK6e4DIMdETegjH
ZTX237RjxZzeEMSq09cpTeM8r7R/2E/Rr+udQqtN2Tj1p88hBj7cEVHTB2f2adPxVBUSA+1oHjYdadpBbhe3R
4jfcbbg1yyo1uphZJqEUo+2FjBnyq/mB+31a1v+JuhrD2EeTrfP24iy3qOLWLN2L6F23PgGKMHqjWVWW241t
QNvYFI2xHjuA8hYR2Rz2Kx09zg24v2RcCelus5Tz9jNq8ICSncCy67v7zV99rCDm00DjGUZYzW/4iZR1a3nyFM
rqOxbgxt8SFbOqXH7aX/IStLYUMzFj88v2aCC8i/RaEUfYsupXchryHPvJWZyeZ7COHTnCbkzUoreJFXZoyS24E
bElhYOBI DhWEaO6IJURc/dHFbFknvY40evaAY0xbgRK5bWO6OcmQPaBjncdxwWNRJtAyJWXUmVhGQQuv
SC1P22Rlk8rxK+QweCx5gq6LjVapHiGCwalcptJUQR50aU0LOI3a/OgrZY3F9rt2xQXdmkXN5JBdqMr1CXLfu
2Jy6guNS44N54jtFmGJ3QTReAZkESzaiZLITKvaz4SuctbuHZGWEROZvfQAN/wASi9Qy/UoB8TAOeYg+pjixH
R8HK2PMuGPKtXt8n0jmXnFyJ2m0dtryjoNVsaJZKnRbHVdktq5IPbDBYEsh6WRCjw5Hj8ZFfO19VwZsVB1
Hh6PCi+AK2kSr5mLsHiXggEQC2kERSyGQ9cjlNuIA3iK700tAtP12Dv2ipulsx8RjxfQz5Wc2qw3ZuRH+P04geJ
dquika4jirC1Oxr/AAdeOp1XTuRxLPjYYUeI7XGAK4rAh4wi7wZawBAR+BGfjgCIMq3g6hJswA1UG5CPKvC7ip
OqhTqR5SWOTFQwnlfJzzwammGn2eYtidR7veH029xIPrpZlxqyO0wsLHnkBbp4NPYXhANSK5gBFSZydvmb
35oIPMDlbHaVMdLZv6Fma5LTTExe94HZYC0f4DrsSthGLsyWlqnCrCLbh4qqBHXboQGmwaEje5QJYycVeZT
U3MHsfUB/ccwDlygPeV05yh/Mz993m7LinTtxOiUuH1GJ6fVbprdcHzuc+qzGkbVjuOMN+oW3WK9h+SU2j
fz/ADH9JfB4Ti2IX6exNtTucFpxsDjBkte75YqGvknBAI9Xd99wPrXXun9IphzLBWnb3b8kksAAP5MG6ZRM
UlyE97eLAO002viHOodZiJkFRbTVaQdgsaoXPiPb5YtP7ijph4ydk1KCS4XAkz/jnpudXjd7fNtySRTWnqLH+3
sN8mWLG6hnW2UWNbWILAI17d/jn2lV630v4crGVnZqpVWuP2WkjtBUHegf5IR+Hk6nk4TfPzyfnXLZUpKI
UrQ77Aq79JHmCd9sZJOWR6Gk0c6HNJxaeJOSz39v0dCxOrdHz68jsrftVpR/Tr1o2gDxyNx1Vkv4mKbczIhp
X/iuf7+PaLMbpeNifMlp5a17hvle8886lcbqIvyxg2GzlZrHFjqoWuva9wRvHdwZLxZRSz2QdNdrIsz4a5f3wXrC
uN2Q11Sp8uxU03lADgx3lXJVTZYx4Vdn+0R1YvyEx/8A9uvsA2dAaA/Ole+Ls2wYqY6fVzy34Em7HGqWeyn
tjGd0dvpXj/Y5Tk5PzbnclmJ/wA5CACvieLiv7mTovbKhD5JTXz4jtafTBKF7o0AIHmESBNia5XBKAUNDLbu8y
SkD7SPudyALUAYfWQDJ7F2PEF2fsJ5I7kMdfEao+jNkZe2Bn6jPOJxMsFVzFYWLG+8T40cqgnVFnG0asWYeY
v048mNxWpVbrk7ROODPKZFXf5BhAbz43Ej2gMODHNJ2oMLEHIksc+JBUJG3cNKEyVEIktvJHdwEoNQ+m7t
eLO+Ds1h8QmcA9uTo9OQCg5mSn0ZFgrA3MjSu4Qq6eT556iQmOT/E4Zbk91jg+xnaur7NBA+04ccd/3jzIU
PO5Gwbw0y0YNxAeVOHfKJTURLFjPqQaakWEZJxOmYt3iWWmyc+xlVet2PZLGmwr7oForeHK0Svvd1eWkI
IXKA2V4alkTq0IV42pC7gPksk+gYISyHJbPng9oUawZgpEkVxCgRA4R7LQMKRNuWCEyJp3B1NkLECBWW6k
1m4scGBbY1CKI3cmBNV3Q0JzC0rnUgrntERwwfaeHCXXiWtsAgthAkmkcVs2ypX44AlMzbOzxL9luNGUjKx
2cmV9yb6L3R2Ya3FZXOt7tRtXc7LI06cd+I5qwiB4ga6p+5c26ml4xgVktDKczsk9uPoSv3oROWtwWUSq8Ozs
vtPWE7eTMu6ymvqnJbsh095WMzrLIDzKyetsfB5/DqN2UdSzutIAfVKaes91nDTIGX1p2JAMk6c91lg8xCTsk
8sZVEIRPonp2exr2JeH65lphK1WF3EWL5PtmUDoGPqsd06QOnZDvRdXkBAR9xXXBGtaMtNL4yj5ZFFf4Hi+

ZLgfflx6X8Wvm4vzKsRlXTfWCCSpII0ZW+kZnxHn5GTfmZNYwnBRMVKijADj1sT5lp7sOil2vyVrXf8Ah48wPK
s6fZg5NWBlomQ1bKjckBtcHRLw5XNLddn8NpCK/iwlnQoxuaxLa5JfflnKPjCunKv6bgV49BpNj9113c9a2KB2
1kA8FpZB1rovw+lbHpFyWl0qCYqEjn3MrfSekVdGOR87qWM2VkBBrE+Y9nITn08ksxBliGZ1bLZclbFxmBGu
tbidwFp7TsOzA6AIHiAU5KfqWUWcq4zilMquectZOxYvWHsawvj2VVjsCd/1sSNn0+wEEqPSK862IXrNxsH
ciaLKX93nPMT47x26NIZlpRCLLBjqQT3BfB0NyX4c6B01cBTlqLMu/Vlrtw4LE2BePtHXq3LbiSf7KN/D4Vqx2Q
nBjPjR5ydhDKdD7Tl3X8gP1W3nhAFlz6UMijALZDFmXvcnZ15Gz7mcUvusttssJ5ZiT/edttbhdjv4VKN00n
lR4yMrHXXmLhcA3mBFTv6obUikj3i3IUOquXcN/crqDLWn2Mif+FhctA8G1I53zJ6bRsaih0sJmqvcvkicyzuC3
95l8RRdZpjxqC1W2HjZml1bE7JmK3wcSJEt9Xmbsrd2wYIIB8xwily+ZFdHSTHdhrmWaogruVQBvjKjIA4k0y
DRYwJ6y7WALbuFLZJ5QPDFr1jc3ps7Trclv0RuIXfR8wW5JhFHKLQrfzNmcAeZW6sgeNwo2giTUyDgSNkr3c
GNKXVhKwaW79ql4o7gsIFvJ6SWBv3ATIqe0gzLJLDBxrLXuqnNrcbWVvU6VYwNcqt1YNsz+mRfXNrAsCaE9T
gxp8s6gbbpqStiMUTHWLdLli2zndD6MtuLbA1Swxm2OUXql+lzRpWabY3rslvCZVTgOlaTq0VrZCA8cjMA4jNX
hSWRQHkgshNxDaWku2MEsIWS2NK7YPcRdY+DTeAI8LUwsZCzRG6wNq4zkRE60Tjji8JJNgSYiCWSOcBU
mj2gCKbrCZvaTBdEwDbY5GKQuDCTNBjbjtaYQKZNQOSTei4q/aY1IAjpwAltM9JpI7CUmyv3pK1IJ7AS3WDc
UX4hY7ErrU5luqb1E5Z1IFVM451S1i5E+huoYiujAA8TkWf0ZzaSBKWdTjlu6bkzn+LivY4navh/o/wBjlipzXR/
UNidy6RhKijSrhmRDVXqMOBxg4KioAxjd/Q6deybftrYqm/q17R1VUvbGtVahSPv5ltVSm8GUs1O15fPKZxB
MWvreZh1Z4vl/cOOctZer5bKndpvPcCPedc6d0DofTu441CV78nez/wB4wYIGIB0fvqcc651PHouyUqtuDU1o
97lBoj+Xb/FrxGjyhVH0Rk89sNXC3Vzwpjrt9Ky0MLLvg9fi2oqgOXcvzfRwvoOhYw8Ej2axJ6OnYOFIW1472/
N+ta1BZyePacnwuiUZGVjdQ+YqJbiqXK67nBGx6j4Cx5X8R5n+08WmioHDWwJ8yqwOG7uF7gedfcyuWow
5bkuyynpbZbVGcmoxWcywmo+xPkYIFFIKDogNW1ZdasdDYAHPb/GgDqWrExeo2Zx+dY6Yu1JU9vc7Nsdja
9h7RlI3viUNf8AtbbrHZUCVjbMWPA/hRHuDUiXWqq2ejRO/Hc/OgT9pKNTdiznsTu1cvBbwums5z/ayZ1hhT
0a4LxsBAP4M4vtd6M638Ts37BF+9n/AIE49YCGlnZ9RL7Izledmfu8kjsk3FwU8CCbEzevacJDJcv2Am63ux8R
Oe7e5lGskskdo7U/iAWt6oE1j/eAfNt7uTOSkjqTLLSqk73DXrrK+YioyAPMY/MQycUmiLzkAtclZlJSeDB7lHd
B0VQ3iB5UieModWXLcMc7li0INw6n0yfuRLbV2aEMGojot2NRj3QuATCSyxTcqbK5aa6JEBKlBcXlhAtNNfd
4jIKU14gK6EPW5dcwsSEibtA9pvwIOLgT4k4YEeiQGL7SC8ybvr+zmZBvOSZwprulpsb1TDZsAAZR6MudR0i
QNXB7DCxXBLF5jlkWRpkQq3Mf41h4ijHYuJbKsllEiOvSzfI9O+CWMh9F0dVuYopxBj6qqWtWknjJW2aiCY
QjmFrZNVoj8QqvGML4E0A8aDMDGTBjCBRqbGqRcGjqmiFXh9dsAKETASIFh44ZZqrYzR5VqrY4qsnlz5BTrH
gMwwZGhEajISccMjYQKwQ8wdhOtBlstOk1VlwZCRILgYzlGyqJ45AkTWai+28T05pl9Gttnt1sT2WczHt7m
mvy9xXmTHEoxlwDJwsIWsz08RiMEkcUssSZOKSGJ95VLunoT4l7tcaie3Urr4xLahySK3RiojeJccO1V1KvdYFg
a55DSv3xiMSrczr9N6kRslglE6XY9up0SjHTtGzLjSxnIzWsUINor/Uacq+vSWtQGPqZdFtf38TiWfl4/8AtivFzOo
0pjoRU9fqO3I8M/8AqZ3HrdfU3xnqwW+XY3b226BC88+Zsc/4Ox7VVPk/NsYs/eeAGc8mC1NOZN4ci00Osh
GpKTis5wl2irdVRL+jZ2JUB6a2FivF6rWFJ0wJ8jtl+6H8J4VNOO/yQny0Covf36A8dx92Ald6h8F5GN0mjEwj
Woe92tBG17WHsp861O3oiqqqAAAPbicp0vmanHrDwd1nxF+DDwZ+pyy884PUQSVrFV0X3ab9rdp0dGCl
W7lZ/qdgAPIXQ9pYNNYwZmOJdiD4kANNP8A7mnLLq+Z1X4jUDGx9eNmcrubmLXJ+K/6C1NeGgcV/wAT3j
7SVH2JHY+pxtpe8JsiKkwV1lhIPM1uZQuwJHDwS4F/k63NnxCR9UEN6d0mGSPuZ5LPZ5sg7WX3MNptTXkR
PZbt42wqEY+Budg3nAOQW5GtiZWymM2oULFNml8GTkmciO17e2e94BiEZZ1rmejlsPtltsFrruUahf7jcpyX
WnmWLGQusnGbYOURgLM34hK2tNFpMJWofeFyCarC/tMqd+7lYe9lI1nKGQ9ya6HNaEwz5fHmKaLxJWY
BuGWALTyY6AMfUJkGbTHEjMg+CZ85qYYjRUr8xjXuUWk4L3UIYDkSMqNzdJrYDqWLaFYRYww1BaX/ERSB
KNhroCXTEfWpcaVxwhLU7iyrioR4kgxdGS49o0I3UqRLyKTKKU5pglNEapjieVIYyQiSnCOCEbJZBPKCQmiOAA
ZsaxKe2CQ/C1lcmL3qloueAWUyqn2WlVhXhsGMabJ5ZTIFUgwWCxUk0WkP4xVpX6Xjet4aDE7YBshYSQG
e6jKYp0BFZCYxiVkbjObQimV68GILu8ky321bix8eAcGPQtSRX6ajHNdRhFVIEOCACFhFIFZ2wE1jUX3AARzZ
rUR5LcGcsnhBKexFdZowB22JDksQTFH7obH2ILZntmiilsNM08GVksQ+ydR1ZlYosYWER2p3aP8AiH8ffUOHS
nycZWTS7AYAnXcD7HXiKulyZ1XqPbxn3PONDfxqAvfcgJPE7H0/Ntt9Phu0MVPkA+DOVYXw1n/PazKsqeh3U
CqtNIQT5JM7hRjVAoe0bXfb/GxriW+kqt93hFN8Rs02Fte5sNrrJ8wv0LNQJv2S8SwuDMN5flvyrqK6w9xAxu
ABI3yfAEPQDQlfvyc5SLeCBUxdGOWCw4/7biq7rC5D5dNLIXTdaNo67iBz/PmJTuri3l8/Ys69HbZfBU8Yy37F
5ZO6sgHnXBguKjitQ9iuQLCDx3e8j6dQ+N0+qti9hRAPj+ptcbhVKVtQfSFvIB459wRCRWXF4w8CkntU45y
t2MiX4ir309T/yulxXjYjeJO9dTQ39PzKwOUTYP3I5nzzY+z5gNQsTz90SozsafsYlvE9azYmV6hJQa3F8DCYGjc
8yC+w/YxotW4b+2BXZk8cEfco5rDt9jDFxuPeNL6aEG4FTIItQQRLLBzR2neoxoylXjgGaWtscRMXG5Btp8Elh
lpObFFzkmL0vpjckL7g25sDOWBFkT3RjrQ8RfSdiMFOxzDQzgfLSixVv/HEdUXkeOIFte2b1CtiJzOGdxksCZFh9

4ULbPvPKK6iojAV1QuWCaQDOWNQC75pljr5dc3FSSLbZ1JlW1Fk8w9GBO9Q5S5W5E2xgTCxIkHJDgH+Jk1Ct
9xMkwJ8p0IS8sIVPEjxMb+JY66JRVcGjsWWAJTJUoJbkr2lIhqY8OnkjhIVV1FY4qJEnXHkwx43VOUQE4JhIV+
o1TjIQVgbgES3q1TKuzSxZaKsmNa8iUlbcIbXk6jj1CaEXpsMv1dohYsEpVeaI3pymVI+oQaFDLD27nhpmUN
uNFUGkx5JtulifGgFmNLeaoJZUIbYTheyodhUwut4XdXfp4ME1gslz3lbo8KBiVHhyPPKYKcA+YRIg03jCKLNE
DrBGQQ4WR55sLAG7QJozTxmgNtmoKU8DChky1+JXcl5mXmIFP8AhtFqNZbT3HgnwIpu3PA1sclbuMCTJr7
7t9xA7SDKzhMx6jbSe1wIjUt0E1/0kQbq/VOp4vUPI2UAUPxVaAW0wG/XrwJWWz+rVHhsPZYhLjIPev8AR2f
Ttl+24GUUn6R6pWsjNYawjp2F8N4d1lwdKjXxrj1bPnmOq+idPwEUtFYK+ERS7cdx8Cc46N1RemXU0vbdb8
yKkCz9/qrPPLaJJ3Bviv4pWwVjL3e9MdKHauytCw+YfHdO5q2cR5AuF7v2ufl+59EdOprauxCSNkjZ35janqGN
ZmW4qWD9wIQSkKHgNwDOO/CPxP0rLGRZj3vbW1eI7HCs48LLZl59Gf03KsUX0fKtspYISjBPrCa559o5Tb5
OuvYTs0zduJZ5x5joXS7r7MBGe6uyz1bdBpdg/bZ8SjZ/X+oY9+TQEr+ay+gNaB3WEJQRfJEWfAPxH0XKxzgYu
JfjvQD3pYN/3Le5aWfVt76epLqmkFK/mtcw5Bc9oUN9zD3SzVGW7GOztNG3WWwU3nJrBUb87q+VmZF
FWPYvUB7bVC7Pd3Ht354kPw5Ycrqgya3f5dzu7V+UAUa3v7E+w43LeOr4mavyMW+t7LVPIBKqBw2yJnT
vhy7EoLVkdwAK0/wCh0cqV43zK9UuVsXHLXuy0nqFVRKEoqtvhL98ZLNx1V/8AbI2I/YgStPIg/VYzeokfWajZ
AygaiHV3H6cled8wqWJDEP47F40gj4uuiXdzanz7mR1DqzFQ/wDis/skTm/nlSuiJwHqGEKc2+oDQVYB+J3Kn
OpewhOSDozlvxhW1PVFs9rkB/uODA6iKcE1zhnKXJWYaxmJUaq+dah4p41AaLDDhcBFVJA085ILNqlms6my
EIH5NymV4oGjGw3nglkhdlZVr77RxA6Gbu5Jv9trmCW4yK8NIOLySTWCcZBA9opuyAW8CMkoDDkwazFB9
555Z5GtCoTzGTUa8GDYwX7ciH/PULoqJNPgi1yBqrBuGMNqsl2GgbsDyvBg6Nb3c6kNyTJbcj8NuaV3hX1CK
alZN7G5s2Mutkiel+jyH+JeOOY8WwalFS4LrUfV2ekSUZZRGSICHEw1YjQmEDuk8EBhcyycqWbpCQSJEAQ
J6OVI61HaNDcN+ZkAGijKOLnMqEWOqlEyZKGS0khjWiw9FEyZGkBYWFEmCiZMhogWSBfMgtZkyHQBgVi
ARc5ImTIVN4IkS2vzvLjH2NMmROzsl7F5xGPEstUyZGKcTuDdcQSwTJkefQpHsRxXdyEzkyJzLejo0VjuGoTM
mQA1IPQwkTJkYgJTPYM4mTIR9Hldi2yJ8n6ZkyKTH4exRc+9xTYIi6Zm5BZFLcEbmTIPvX0WNqX8aXAV1zq
F9H7ML2kWxdrAj21OydeLvVe3eynHdba+3QGyPB+4mT17d6hXsdlCfdGdRnjJussAuLoCR6NeAvHgQX4b6
NwAmN+6yWx7shO6ln2o99LsbEyZFKvRItNRxZ/SOWX4GD0Kt7sKhFY6Hq2QO4+REef1LNxL+oiq59Urh9nc
S/1Hn6t8mZMk33/YTR8988G/Xeu5nR+q9Lw8BKqEzcqw5DKvrfTIfJ74jetqermymuztSpAH2QFzTETJK9Nvd
D+g+m+nu923l+/qR1npPTMOqqmpBXVTawrROAOH+EfxL4DMmS30nPF7Md8Uk3ZDLfTFD5Fg3/NGEE6h
ohF12NA8+x3MmSWX4cxekSuj+zbplNdSN2/5nyYo+Ma0foq2Eeq5e0/+7gzJkIh/wBaX6AWt/y//scardt+YS
HffmZMLauh19iy927/ADIF8zJkhHs6RI2i2yxlt4PvMmT0jsRzUx1IHJ75kyRJABeH9ibgksZkyQ9zoWK1AgLkox1
MmTkjqHWMxKbhlMe0iZMk/Yga0HcsWMSa5kySqlzGaniThzMmRoATKxmrrnRmTJw4ha1rBjMmTIPLCH//
Z' alt='img' class='img-responsive primary-img' />

<img

src='data:image/png;base64,/9j/4AAQSkZJRgABAQAAQABAAQ/2wCEAAkGBxITEhUTEhIVFRUVFRUVGB
UVFRUWFXUVFRUXFhUWFRUYHsGgGBolHRUVITEhJSkrLi4uFx8zODMtnYgtLisBCgoKDgOOGxAQGy0mICUt
LS0tLAtLy0tLS0tLS0tLS0tLS0tLS0tLS0tLS0tLS0tLS0tLS0tLS0tLS0tLS0tLS0tLS0tLf/AABEIAOAA4QMBIGACEQEDEQH/
xAAcAAACAgMBAQAAAAAAAAAAAAAEbQMGAEECBwJ/xABBEAABAwIEAggEBAQFAwUBAAABAIRAawQF
EiEXQVEGEyJhcYGRsTKhwdEUUnLwBOJi4SMkM4KyFXpXQ1OSosIW/8QAGgEAAwEBAQEAAAAAAAAAAAAA
AAQIDBAAFBv/EACKRAAICAgIBAwMEAwAAAAAAAAABAHEdIRixQRMiyYQqIUSNxgFAUQqH/2gAMAWEAA
hEDEQA/AL50ewQNaHP1cniW0bGyCw/EWkaFGOrSsycUijbFrLBrXSAoMbv3UWsgk+Caayo7+gHN1SJ0m
ww2yJ//ANZWn/Rcrhglw5wDnCECLBnIjzYMAGiWGBm6ovOFIKu7ghpHeKdJBNy57zJLz6L2e8GhXi2PUpu3
5jpmMK7uyHgsNhWZIEDhtC7cMx2hEYBYyAnT8N7IncGx2kVupbudo0apnh3R15iU5sbEM1KrnSTpiSTQtTG
4dV5RvB4Dv/ZpGNdicbYRjL7Sj2HODn/IzQr+rkk9F5OtEDwdmHoY1PdCrLsQDCSACf8A3H6DybufExvuiLLGi
XAlxfrOglE8xAJRZVQXktuH3lywEuDagbJDWRPdlcZB32nw5HWnS6mXBlWk6m7kd44HfvQeG3bag7Dgd5B
LhJ7gePdKUdlb+WKAHO0zqkASQNew8Rpy5a8NuTCOX6jdsJjMDOxB0Pd4ot1IleP2PSI1A2mTDpAD3Akj+I3
GRzV86N9liXdTWC4aNeNnH8uvFH9xvotDcPjXNCiWogo62vmkb68l1UqNPinpeCjMzZcEKsntFOcaYFuv
QDgtFQVrvLug2kcBtwpgPwgnmVOzsiFeZEA7Yoe6uEirtBbbOlyGHkaxtwAAoqb+a7dWhdxXYE6GpVDksQH
44rFw3lnFmBGijFSraPfalzTQl/RcRDUxXDiaucU0cnRUa9Ks0wduYT3DmmBK7ulKltgoQxqMtDvI5dk1WjIX
mnSzA8teTBDu0PqvUnv0VB6W3M1RPDRVnomM+jlqAwJ25gCRYFdAtATHEbwMpue4wGtJpGAm5JIamU/
p5jpb/l6R7b/iM7N5TWxneJxabcSIOMbuPMqXE8S0fWqHtVCSpanh7eSrNvGoc9XMAadhE+EjfySpXtIPbPH
Vd9WprHZnR2mXycpqVQNaMxc8jhs0eZ3RVChpAqBo/SDB/wC1O/ij2Mp6Zx1gE69TBHdmB08CCuckcosm

wXpCabgWt0lj/UBPe3Vv14qzVb63umAOhrzsey4zG2Zokf8Ax81XBRpEhoq9S4xAeHsBG2h+GduKhubq5tP
hqZmncgU6w8HRqB5BLVjXRbdUH21UFzQ5uYQ85SNDpO/EnUc0SLuo2pnOkulxZ8JDoOYabiEAzpF1nZfIIP
COz5E6hMslxOmf8J+rSDIJ3b0404d2yZ35FVHoWG3pqsY8HUiD+ob/AHVmw6g7cleddHrl1F3Vky0mWH9/
vZeh4NeZ1ONchJrY9pN0XRauqa2QtJOziEDfUwQUc9yW3VZLJBSrttQe2oQNU5/DmJIWMA3RYuW5VJQ4
jqYGyELeOgoZ+IdswuK1Ykyu5Ao465yxcSVpdZ1F3yrYC7hZCtQtnMKCrTIFAKNy5o5AooBSU2QpQtgJVFIJH
W2VExnD3VLg/IEeyvz2ygalgJJXSVhQnsLPIk3T6/FO2LT/P2flgz8lbX2y8n/Air1nGsyk0/C2TyBdz8oU5J9FI0U
eqTVdMTGjGnYAcfBcvaxplxzu/8Ao3u7/ZTFwYzkDwtLjzP70SWtULjpsqIDGFTFaTf/AEw4nIHodD6Babjzso
GgNHDIb3cRx4wljWtG7QT/uB9/opqFMnQN+390XFAUpMeDpPVy5Q6eYcM2p3kOkH09FC+5D4d8D26h1
JzmieRpu07oBhcUcEqEAwjrLBqnGB5D6KTyQXRb0pvsVvmtqawGv5DRru8D+U/XxUNu85gSntNRPd6/ZXC
hgExOsa7ckNiNj1AzRI5H77goRzJ6BLC1tjfbKwe0RUZmbqGF2XbUfPmG+HZ816RglXsgxEx5HivIMKxq3ByvY
9moh1N0EekfOV6B0lvRUDw2o54BBlwjeeEnkOSD0wdo9Ftrkc0R1gVZNbKtjFQOKdZCTgPKz94St4B1IDVM
WEH7pR/1sCe9c5oDQ+cIEpTd3kac11Tvi5qEuGF0Icjlta5weTwKYOXSRK6NuIUVR0JaOTGVPBYushWijOX
IBbAXIKwFaSJ0onBSStQgziOfuFigrIXUdZxC5c1TBq6yo0dYDWEAk8BK8B6d4mHXVR411ho7hDQfOCfNe4
dLbkU7d5J4evGP3zXzlj5LnucNZKjLcQLQ1FsVXdw52+pOngOXco3g/CPMrGNjvPsunmNOJTGJbS1zEafvxVy
wjDGgCAq5h9q06ve5o270581Z7LBmRNOtWHg8H5QsWedurN/0+NpWkO6dp3KRlr3Jba3ISm/Ka2cDg5o
B9QrG2s0szaBZetGn5YMKYaJMADcnSPNVjptF0alCoGVA5zQC114gaHijBQZVqHO51TXZzoZ5NH1IS9lqTTbV
abWtAawnsG DYZu CtCk0QyJtM8xtq3a79/Tf8AfcvTf4XvluCODmeQlcPuPVeT5ocCOB9I61/DYf4jSOLi3yDes
P8AxaFuyLwY3pnp9S9Jmdknu8LmYT65vGsbqQlrLk01kKckgWVF7nhxaQSp4a52pMK12tgHOc6FNdWzWt
J5IRh+RG2Vy3plggzaNBASmretDOVObBnYBTJbDGxDil64VSwbaKTD3S8EoTE2zXKMtKJJEIM5NlfZgsQmQr
F1sPJFzatgod1VTUmq6ZI7C6lucWiUUzjuVhK5W0bOO2LpaYoL+qWshvxHQJvADzH+MWOHK2jTjgOLXEbF
xAOUeA3/UF4zcVTJG4AjXnxV16a3HXXXVsMimCJPFz+092niB/tS63w+jT1cMzu/n4KHNlbnCg2qRX6Nscud
wge57lCxm7ztsO8ptidxnMu+EaADieSWVXHTxGnAdw8NEVILjTosmFWGalGhkcVlt+No9hgBGurhMeYI+aa
9GxoFa22wK815ak0z1Hi0qZQ7bD6rZqVHEud5AdwCuFtbl1u09BYmzt5RsrDYMHVZUeXLYrjxVHnrsArteer
e9on85glu3DnW9pcPqPLiaVQniCS06ydZ4KzjDgHcUj/iIHizcymJzFueNxBknv1AHgSmjlcPkJOWNRi5HkTK
eZ4A2JA+a9N/ha/M4a/DUC7yLXT83NXn1OGDaXEaeMLOPoHRNKhniC8u9JA//K2ZZaPPiqPRcYo526alCWF
NzBBGiht8VbEEou1rGpOUSOaRSTJB1rc5Z10Q2KX46t3eCFLa4W6DJhD18KicwlfYdBSrdhTOVmsXENgqKjZ
NHABGEgBD5OUmCVrcF0wFPQpALqdNEDcV3A7JY5EKxnIWJT+JcsTc0dQ8pYmHV2sPEH1VkaV5zijCyuSDE
ajuTNvSOpliBPNZ8f1ag5Kf5HcGW2rXExXTWYq7hNVzjncd0/pVgBqtWLKsisRollOW2IKbzF2zlbqSmdrsqRn
GTpHE4VZ6aYwKFN8EzXtkf0gmC7xktAVIJXnv4QXTa1eq4tFSu6IHFrWVRTY1sjkJJ11fpGsnLKISGxxt7PG/wA
a4OcW7uOpOpJJ4/Nd1ahl3knTwQIXSo4cnfcfRarh+zZ1/e/BQqzatIhu3hoA3/epQjqkgIw2kDtHfgPZBV3a/J
WjRnld2Xjo1cgtaQrdUug1mZxgAaleV4BcuY7TbkvQsPvA9gB9CvMzY+Mz1sOTnBATxile9oVW+ZlRdlj4Byhz
HTsGAvCZ5EaDxKldhjDrljw+yMsMPa3WF0eA7qiWnemC+o0MY0Egk6wNSXcAqX0n6RC0otYczqmgH5WD
83L/AMqT9jDswt2CWDWoQYzOGrWfpG59FV7VoAzEy92/coAC0Ysa9zMOWd/aiS2shAYB2nHtP3MchyH
2XqTcO7LkbdAAGjwGipeB2etOd6jgY5MbqT5/Reisrjfkz51Ey5FoYN6OUy2NZjeVrCqHUksOgnQpphd5mb
st4g0ESjScecGQ6JfyrW6le5uA7YJPvqGLWHYiwvyl2vJQ9aWTSYaC7ig8jRKqgdsSVY679FX8Te4GWtJuc4oK
ih3Y0RkCivWgg6LnCrrMwSIPJbr0yTAQ42tHJoV9UfTgF9LdzWlelMa0I8Su87y7mh2VZhGvwGoSirXo4/ipSx
Ska2o0M8lrgtA5Iq8vYbuorfB3N2Kgv8JqOEBXrJGNUQcYtgeFVwahJ8lbbO5gKsWnR97dZTm2s3jcrsHqQf
QJwXgbuqSD3g+yptpeAWZqcBdVHgHgDXOp7gD6K2ZMjs47AE+QEleNYt0hf+FNrbt1carqtQ7NaX53BvkGE
nhpG4Wqbk+wY4opeIP6y4dkgAuJJIANZibu68DLT3iM3HvQNM5SQd+PfoWpuiqkgSeMAeaaatospaYLeuyje
XbT3IKoNPAqW5PanvJ+a5ayWT/UFaOkRltjPCaWqtlg0iIvewpnbb3j6K3UKcBed9RLZ6n06qIXQxQt0cyfAw
pq2KvcCGtyz318OSXtaCUZSpKkHkV/FbURMfvmuKnRhuUVGOGrmtNM8SQdWnyOnen9zZBw1RGFYUHc
zlXCZjmsYPcRI81oU3RmIFCywTks5+rcC9uRstIDWN4CdyeJ8FaMJwmpoam3JXfDblrmS5kNMQ1wECNNER
VtKJ/pJ5H6KqwxlswZeTklrWmGtS3F7sDinFTAiT/AK0D9Ovuqr0j6P3NNwc2atPcubu39Td47xPkpZ4ZFDUSJ
rqcw3QT8NAeHjcl14WN2ld4XcOq76LyVhmpcoer1XTHVG8EaplaVGEaQklzYECQVNH4W2OTLcdNHJJo6r
VMrzGy23EACIKlnOqV1rfKUmvZYq0FJWNv+pNWJTIW1P1s3wNxRd22w5KVtEKYBbXOCgibmyLqInVBTLsbi
heTlerXTaakhR3NTK0IDiuw23oq/TOu+oPw1N2QO1q1BHYPnAc/wA8pGnN4nQQfN7WxdWp1eqZlPkOdH
EtZLmsLjrlA7TjxcfbWLpvCuHWjPbc0+bHMaIHi6kqxZ4q42tKm8ihQyjNII664iJA/LT4T8jOmOcuTZtxw4op/w
CGDqktJLARqeOkAevyWr/VwA2kfM/YfNmCqumkkUm5WCcoP8s/OUq1LtOHH6p4uzpKgK6pyD4rKI7IHep

6x7Lu8wPAFdYfQmBzVbpEeNvQ5wq1dLXRofY7ipCNwe3aKQEawpKlKCV582pOz0opxVAWHMJHimQbCKt
LcAlgUQhoDZDZwUyidArRQpU4bSpiZ+i8SOKU21rqPYb/ANlALC1yN2An96ncplalZjpbXVaROXSNIMeHBCc
nl+CJ4udq4+fBaQV2tGp9fslD5iusN9f7K2OUv9f7/fgzwhlm6itf8GTrotkudKltcSB0mVXmsc9up/m1J4BdVLjJ2
KXmeJ8eXgmx5ZW7/I2TDTpnfSnBWIprUhtq9o5cXAe/r40+leCk4RsvQMLu9cjjMjXl4Kh41hPU3D2fyzLf0O1
b6beSlmio/fH+TNKFMcjEg9ui3ZOgpbSDCeWdIngp+7bCojOlchAXxkro0SCuagTy2qOoGhYperW1PhEOy+B
dLkBDL10RMWLaxMAx4w6GDx+hRyV9IHRTH6voUmR1Fj41c0eV4nUNRwrOMzWNLL+VvDzXm1aoQ4yT
oSBJ2A2Er0/pBYw17mSCBJAlghurHa7GOz5Lya/ac5MzPudVgwq2z0MrpIKNwPGPIA8FEyqTtoOJ+gQ7QJgu
JPIKbrMvAacN473d/ctCSRFts1eO2HnHlcPNNcBpZiP6R7mfoPVLKVOe07+5VwwfDHUgMw1cA7yPDy28IP
NkSVFMUHdj+x0aB3BH0hKGtaBMABPbS1DRrqqfkskYuRqIoiCjkb9w5ouhSk5WiT+9+SKZSzJnbWwaNE05Rx
6XZncnI5o0mUmy46/vZR18UnRvqV3Vw4OM5nT6qNuDOOXEcSfmrYY438sGP8Ax7ubtg1GoXkk68kQ3DQ
T2jlnh/mft5o+nQZS1jXmfpyCHqOLjoPNNmqD5lt6zk/s0ge+qZwTfYwAanbh9z3oBuiMxBmUN5yfmGkjoEK
MZtq2Ta2TWbznBHBFdNLTPSp1wNWkNd+i23o7/AJLjB6Uu8IYepFSk+mdiCPT81WEOUWn5IZq0VPBcOBG
YhP2WwAXFISytARRKIGNHNC6tSUJtSmcKQMT0ikPwHJt1a2koahusWLF6pjMWLFi44xJulLopNP9Y9inK
rvT616yzeNdC0iHZTVGh4bqeb2Mph96PL+n+iQKdIEjrDlcRwYT/afJUjH8hyimNCSRzOwBPk6B03FrYh7TUN
UBhENqvD6jo4CNm95S5rZgSdN4jYeHqseKNUzbnkduhc05dBvxdx8uQUjB6cPuoqb0ywagalQDKTqC7XYT8
leWieOPJpIYYRhr3EOyZo2H9l6Fgdg97MtRpaAZY4xIPERyRGDWLgtGUKyWdCfBefky848ZL5PaIOEMPppfz
5sV0qYZ2WiOZO/mVLSMuhupTK+ssw7Oh/e6lw+xFMcydyh/klJrs8uX5JrW2yjXUooBaC7bVARhgc3Zmjll
wSsaBv6K0vftb3+CFq1y4/0+6DuWzo1a6WNcei0MCcrkvbL5zuyNAPbe5IAIAElu41U9sNJOGzqLho1uEVG
ktHeMVBk679Eopu3RmMvGQAOfqh7VqzKFKmIk4od4RTifBOLbcpdhrdD5Jhbuk+oWvHqjFI22Dvpdo+JXR
p6lt7dVG5BxoVSsGFNdLtwXOVLQxwsUmVYhQbGSxYsXomMxYsKHfWhBugpWTyIPsg/5ap/t/wCQRhrFJO
mFyW2lV35QDHOHDRTyO4MpjVSR5J0ssZZmnUHWNBHfz8VSKhJzNAOjRMsX6R1KjiNAOWpCT/iHOOp+iz
4oOMdmnLNSlo4pUiiA1M+i9Z6M9DHU6TYLZd2jMySqz0I6N9a4VHjsyNlln+y9Xr34ZDganieAUismaU58Mf
8jYfIjOsa2DURdqdXOBngEzpxGmgScuJMnVT0K3omy4OUbfZ6k8ba+52xi64dwRFB5P3UFvSnU7Iyi0jwWW
ONeUY8jiti5NEyTzUNSm/iPTgOXijphZKvHNx0jI8zgI61cmI0H73RdMyAealu7umOzAceXLzWUKcCSAOMDY
Ls2SoLI2alkuN8aO2W8/Eu3UhsPKbwRouazw0SsbuRBZG3XkUY2xoygbzP0UNq1TXVlvDnn9wuLfgRvXGpQ
qPY9w7ZMGbhAYdsj2bhb8fR5+X3M7qKEqavwUJSz7Ej0aWQsW0gxpBWLETg5alDmuuTWWzkQ4hLigX7rt
1Qodz0kmMkSqt/xEYTh9eJ0AMDjDgnb60Jb0lGe0rtOxpKJPTHito+d327RgB4ifhB9yi+jtqKtcN0Eat0BBI1g
g8xKX4raltVzQZDTEj2Vg6FtFKs3rGlsyWujQkwNTw008wpZJVCzViinPaPR7aqKdIZRqdPuUVQoMcJBP9+9C
YlBkEEbffUKCjXLDI9FgXQ4xfF0z28WJOFxe3/aHPUt5n2TG1tWgSW+APuhLljd3kPumJMjXQH5oxnkl2zHm
m+idvcuqRgamUO6sGju5KIXQ1OvohOfF0uzHLHJRRO+sQ7hljfkgrvEi4EUvM/ZA3Vd799G8lDQfIMnZXx/Tu
K5Xv8ABpx/Tqk5bYtaSHSrtz905p3APclzHSDedRt+az5Zeo7kDNT2wqllbyAQtetmOmwrTmOmOSypTydNJ7t
/kq4KiuJfXU7oWXGLn/ghLVO/wwwLRxBHjASege89u/yXjNNMeYcNEeEFYfCEYAJ7wtkPbowZPcyS7+Enlqg
RWTKoJecx7pK0QYPDRJn00wYugxhXcqGmpFNDG5W1ytonHAXQC4zLfwLWTZl7Kjcf01VBWroNhsNVACUPj
LP8ALVv+272W6dwCVB0iugLWR3tj10U21TZRJ2jx26wBry05oh0kcxYVowS1blbIEDu8ksDtU0wqpDgvMySk
+z18cEui31rYBjYGw8o4BQPIA+Hu5Qmds8Op68j7TPyS2u8DeO9NPGnFSJQbc6fg3b0mjU6nv2Hku69+0aTJ
4IR1YcF1bWLXOlw99UrzpLijTwXumEUKbnmTsjqNMNEBSU2jYLosUVHyY8ma3Xggq2rXbiD3lCpg7y6Q4R3
yE1LDGm66NyBpueQ+6vinkf7Cxyzj7GZZ2jWjbVGNpqGnUU1R5jRFrt7ITcm9kdR4Bgb+y4A5LVO2MzzRvr
Tg68FoUEISObUVo1ZD/EA5Sk1Snle4cnH3TvDRlye4+6XYmyKrvGfUSqTj+mn8gxy+9r4DrP4QikQ+aHtvhCl
aVeC0iE3thnBJ75sVJ5ifoU3GyAxGnIB5H3/8IfURulmPs4onRShQUgpggRKnSxaWJjilhcOUsqOq5XYiBpMrb
qMqRikhKkFggfFXemr8INrj+Ik+Tf7lW8NVA6cXOavl4MaB5nU+4SZaUSmHc0VbPqmNIVS3qtUZa0iFglE9S
Mi9YNLqD3nYaDxhJryHfzR3HZN6ZcyzbTA7TzMcddfaEspWgaZf2nfl4Dx5laljqCRIWV+o2mZZW9Q/zQ0bm
B8uZTTR8l4mPKfNBVbmBqZjghOtLuKzPjXWy1OT2xscRd/Kira9cdwPZKbWnJTOk2EY472TmohdW600EIU
O119l0u2hXTrFi+KGNFwgahTsSoMU7ajuZUuDsnKlye6BK4DoZ3mUL1h05UxHZHitEvolxpE+EScxIhC403/
ABAebR7IMcMb2SeZ9kY7TfA+6s41hROMv1TVA6KaVBS2XeZGPQsuw+gezvKjriQVuz2K25M19tCeQJikA
UVD6qYLKl7MwLaxMA/9k=' alt='img' class='img-responsive socendary-img' />

<!-- <div class='product-grid-atb text-center'><i class='fa fa-shopping-cart'></i><a class=''
href='#'>Add to Cart</div> -->

```
</div>
<div class='product-grid-caption text-center'>
<h4><a class='p-grid-title' href='#'>Tracy Summer</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>Date</li>
</ul>
<a href='#' class='product-grid-img-thumbnail'>
<img src='img/img (27).jpg' alt='img' class='img-responsive primary-img' />
<img
```

src='data:image/png;base64,/9j/4AAQSkZJRgABAQEASABIAAD/4gxYSUNDX1BST0ZJTEUAAQEAAAxiTGlu
bwIQAAbtbnRyUkdCfHhZWiAhZgACAAkABgAxAAbhY3NwTVNGVAAAABJRUMgc1JHQgAAAAAAAAAAAA
AAAAAAAA9tYAAQAAADTLUhQICAA
AAAAAAAAAAAAAAAAABFjchJOAAABUAAAADNkZXNjAAABhAAAAAGx3dHB0AAAB8AAAAABRia3B0AAACBAAAAB
RyWFlaAAACGAAAABRnWFlaAAACLAAAABRiWFlaAAACQAAAABRkbW5kAAACVAAAABHkbWRkAAACxAAA
AAIh2dWVkaAADTAAAAIZ2aWV3AAAD1AAAACRsdW1pAAAD+AAAABRtZWZzAAAEDAAAACR0ZWNoAA
AEMAAAAAxyVFJDAAAEPAAACAxnVFJDAAAEPAAACAxvVFJDAAAEPAAACAx0ZXh0AAAAAENvcHlyaWdod
CAoYykgMTk5OCBIZXdsZXROLVBhY2thcmQgQ29tcGFueQAAZGVzYwAAAAAAAAAASc1JHQiBJRUM2MTk2N
i0yLjEAAAAAAAAAAAAAAAAABJzUkdCIElFQzYxOTY2LTIuMQAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAWFlaIAAAAAAAAAAPNRAAEAAAABFsxYWVogAAAAAAAA
AAAAAAAAAAAFhZWIAAAAAAAAAABvogAAOPUAAAQWFlaIAAAAAAAAAAGKZAAC3hQAAGNpYWVogAAAAAA
AAJKAAAA+EAAC2z2Rlc2MAAAAAAAAAAFkIFQyBodHRwOi8vd3d3LmllYy5jaAAAAAAAAAAAAAAAAAFkIFQyB
odHRwOi8vd3d3LmllYy5jaAA
AAAAAAAAABkZXNjAAAAAAAAAC5JRUMgNjE5NjYtMi4xIERlZmF1bH0gUkdCIGNvbG91ciBzcgFjZSAtIHNSR0IA
AAAAAAAAAAAAAAAAAC5JRUMgNjE5NjYtMi4xIERlZmF1bH0gUkdCIGNvbG91ciBzcgFjZSAtIHNSR0IAAAAAAAAA
AAAAAAAAAAAAAAAAAZGVzYwAAAAAAAAAsUmVmZXJlbnNlIFZpZXdpbmcgQ29uZGl0aW9uLGlulE
lFQzYxOTY2LTIuMQAAAAAAAAAAAAAAAAAALFJlZmVyZW5jZSBWYWV3aW50aW50bW9uZGl0aW9uLGlulE
lFQzYxOTY2LTIuMQAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAHZpZXcAAAAAAAAABOK/gAUXy4AEM8UAAPtzAA
EEwsAA1yeAAAAVhZWIAAAAAAAAAEwJVgBQAAAAVx/nbWVhcwAAAAAAAAABAAAAAAAAAAAAAAAAAAAA
AAAAAAAAo8AAAAACc2lnIAAAAABDUlQgY3VydgAAAAAAAAQAAAAABQAKAA8AFAAZAB4AlwAoAC0AMgA
3ADsAQABFAEoATwBUAFkAXgBjAGgAbQyAHcAfACBAiYAiWcQAJUAmgCfAKQAqQCuALiAtwC8AMEAxg
DLANAA1QDbAOAA5QDrPAA9gD7AQEBBwENARMBGQEFASUBKwEyATgBPgFFAUwBUgFZAWABZwFuA
XUBfAGDAYsBkgGaAaEBqQGxAbkBwQHJADeB2QHhAekB8gH6AgMCDAlUAh0CjglvAjgCQQLAIQCXQJnAn
ECegKEAo4CmAKiAqWcTgLBAssC1QLgAusC9QMAAawsDFgMhAy0DOANDA08DWgNmA3IDfgOKA5YDogO
uA7oDxwPTA+AD7AP5BAYEEwQgBC0EOwRIBFUEYwRxBH4EjASaBKgEtgTEBNME4QTWBP4FDQUcBSsFOg
VJBVGfZwV3BYFFlgWmBbUfXQXVBeUF9gYGBhYJGwY3BkgGWQZqBnsGjAadBq8GwAbRBuMG9QCkBxkH
Kwc9B08HYQd0B4YHmQesB78H0gflB/gICwgfCDIIRghaCG4IggiWCKolvjSCOcl+wkQCSUJOGlPCWQJeqMP
CaQJugnPCeUJ+woRCickPQpUCmoKgQqYcQ4KxQrcCvMLCwsiCzkLUQtPc4ALmAuWC8gL4Qv5DBIMKgxD
DFwMdQyODKcMwAzZDPMNDQ0mDUANWg10DY4NqQ3DDd4N+A4TDi4OSQ5kDn8OmW62DtIO7g8JDy
UPQQ9eD3oPlg+zD88P7BAJECYQQxBhEH4QmxC5ENcQ9RETETERTxFtEYwRqhHJEEgSBxImEkUSZBKEEqM

SwxLjEwMTIxNDE2MTgxOkE8UT5RQGFCcUSRRqFlsUrRTOFPAVEhU0FVYVeBWbFb0V4BYDFiYWSRZsFo8
WshbWFvOXHRDBF2UXiReuF9IX9xgbGEAYZRiKKG8Y1Rj6GSAZRRlrGZEZtxndGgQaKhPrGncanhrFGuwbFB
s7G2MbihuyG9ocAhwqHFicexyjHMwC9R0eHUcDcB2ZHcMd7B4WHkAeah6UHr4e6R8THz4faR+UH78f6iA
VIEgbCCYIMQg8CEclUghdSGhlc4h+yInllUiigiKvlt0jCiM4I2YjIcPCI/AkHyRNJHwkqyTaJQkIocVoJZclxyX3Jic
mVyaHJrcm6CcYJ0kneierJ9woDSg/KHEooijUKQYpOClrKZ0p0CoCkjuqaCqbKs8rAis2K2krnSvRLAUsOSXuLK
ls1y0MLUEtdi2rLeEuFi5MLoluty7uLyQvWi+RL8cv/jA1MGwwpDDbMRlxSjGCMbox8JlqMmMymzLUMw0z
RjN/M7gz8TQRNGU0njTYNRM1TTWHNcl1/TY3Nnl2rjbpNyQ3YDecN9c4FDhQOIw4yDkFOUI5fzm8Ofk6Nj
p0Ori67zstO2s7qjvoPCc8ZTykPOM9lj1hPaE94D4gPmA+oD7gPyE/YT+iP+JAI0BkQKZA50EpQWpBrEHuQjB
CckK1QvdDOKn9Q8BEA0RHRlpEzkUSRVVfMkXeRiJGZ0arRvBHNUd7R8BIBUHLsJFI10kdSWNJqUnwSjdKfU
rESwxLU0uaS+JMKkxyTlpNAk1KTZNN3E4ITm5Ot08AT0IPk0/dUCdQcVC7UQZRUFGBueZSMVJ8UsdTE1Nf
U6pT9IRCvI9U21UoVXVvWlYPVlxWqVb3V0RXklfgWC9YfVjLWRpZaVm4WgdaVlqmWvVbRVuVW+VcNVy
GXNZdJ114XcleGI5sXr1fD19hX7NgBWBXYKpg/GFPYaJh9WJJYpxi8GNDY5dj62RAZJRk6WU9ZJI52Y9Zpjm
6Gc9Z5Nn6Wg/aJz07GIDaZpp8Wplap9q92tPa6dr/2xXbK9tCG1gbbLuEm5rbsRvHm94b9FwK3CGcOBxOn
GVcfByS3KmcwFzXXO4dBR0cHTMdSh1hXXhdj52m3b4d1Z3s3gReG54zHkqeYI553pGeqV7BHtje8J8IXyBfO
F9QX2hfgF+Yn7CfyN/hH/IgEeAqIEKgWuBzYlwgPKC9INXg7qEHYSAhOOFr4WRhg6GcobXhzuHn4gEiGmlzo
kziZmJ/opkisqLMluWi/yMY4zKjTGNml3/jmaOzo82j56QBpBukNaRP5GokhGSepljK02TtpQgllqU9JVflcmW
NJaflwqXdZfgmEyYuJkkmZCZ/JpomtWbQpuvNByciZz3nWsd0p5Anq6fHZ+Ln/qgaaDYoehtqImopajBqN2
o+akVqTHpTilqaYapoum/adup+CoUqjEqTepqaocqo+rAqt1q+msXKzQrUStuK4trqGvFq+LsACwdbDqsWCx
1rJLssKzOLOutCW0nLUTtYq2AbZ5tvC3alfguFm40blKucK6O7q1uy67p7whvJu9Fb2Pvgq+hL7/v3q/9cBww
OzBZ8Hjwl/C28NYw9TEUcTOxUvFyMZGxsPHQce/yD3lvMk6ybnKOMq3yZbLtsw1zLXNNc21zjbOts83z7jQ
OdC60TzRvtl/OsHTRNPG1EnUy9VO1dHWVdbY11zX4Nhk2OjZbNnx2nba+9uA3AXcit0Q3ZbeHN6i3ynfr+A
24L3hROHM4IPi2+Nj4+vkC+T85YTmDeaW5x/nqegy6LzpRunQ6lvq5etw6/vshu0R7ZuKO6070DvzPBY8OX
xcvH/8ozzGfOn9DT0wvVQ9d72bfb794r4Gfio+Tj5x/pX+uf7d/wH/Jj9Kf26/kv+3P9t///2wCEAAIDAwMEA
wQFBQQGBgYGBggIBwCICA0JCgkKCCQ0TDA4MDA4MEXEUEQ8RFBEeGBUvGB4jHRwdlyolJSo1MjVFRVwB
AgMDAwQDBAUFbAYGBgYGCAGHBwglDQkKCCQoJDRMMDgwMDgwTERQRDxEUER4YFRUYHiMdHB0jKiUl
KjUyNUVFXP/AABEIAfQCcQMBIgACEQEDEQH/xAC0AAAABBAMBAQEAAAAAAAAAAAAAGBAUHCAACAwkBC
hAAAQMDAwIFAwMCAwUHBQEBAQACAwQFEQYSITFBbXmMiUWEUMnFCUoEVI2KR0qgWM3LBJENTgpKx
0SU0Y30iJ1QBAAICAwEBAAAAAAAAAAAAAAAAAQFAwYAAQIHCBEAAGIBAwIEBAQEBAUDBQAAAQIAAxEEEIE
FMRMiQVEyYXGBBHrCkSNSobEzcsHRFSRigpIQ+FTc7LC8P/aAAwDAQACEQMRAD8A9Y2sSrf0a1dw1Sw
bE0a1LGNXxoSloWpubtat3NXVoWzgsnelxTBCIWOCjOYciUqxwV0JA4kWXMelyr7fujYa6D0uVeb/0emun7
iltV2Mqxf3Yc5R7A8bijfUDvU5RpTyYcrfUPJKVYfPjLpHjhHNK8cKLqSXoJyKl6LTLNq0kancMBE0Lggimk4RND
KhCsMUwwhItqwnNQ1DlneOThDIZMDHYNaugY0puEqVxyZK4KyQGLWwpS2BascnWNDtCFwZpFTJ0ZS/C
WwRp8jCCcwtFg+Kceywwlp8gLQwlctCQsFvIXB8CKjAkj4FGXkgSBM0HVDC8KkeWnOCmCam+F2GkbLI2m
h+EwyxKSjQ4TJJRk9kQRhI8dEfzFwWfCMHUR9lgoneyl3CQEGDH06Ry0meyP20TvZd/6eT2WBpwVMiG
Si56Jskoiphltz9qa5Lcf2olWg7CQzJSOCbHwPCL2a2n2Q/PQH2UoaDkSOHbgEy1MnCOkmcOyB62CQZ4U
yyJBSZ/KRiXlbTxv8AZM5Dw5HKBBmPMK4JUv0zwcKPYHnRjSEkhckTamSHSgHCOaRoQNRNPCKSiYUvcw
5IYUjOAI2BnRMVHHwEYQRObjGSKI4wlyjSiOJK/KQxMPURM1qcY0nwurSh2hCx7ienRrwh1j+U4NIQhEME
edy4FySiVc3ylZpKs7l65F6QmVcTKohJCBFrncJomPVdDkKer1MIO0aJRyk2F3lckoctNIYsjTknNkbuU6siQjSQ
RzidyIKFyG40/woSEL2j40rukBEozwuwZ3PjuibPQEtKQSKUGRGNEuE0PkwnOUFMEzXlhQDBmOJ9NXhYlgB
3Q7M16Y5HSBTisGDmzEkT+qADqkz7w39yi6eplAKEam4tyyuxp8yM6jEnP+sN/csVbv6tL7lYU/ysj/ADM9Qg
F2AWALsAtSzTzoShoWrQIDQskk6NC2cFs0LdwXGZkZZhwUJ1YRIMOEk1bOCpBlmKTXQelyrnqEcPVI7ozhyr
tf4+Hpppylh1Q8plOtR5DnKJoXnzCpm1JefWoaiAepV1oI2SiXZ8Qw1pCeEd0jjwgWkI4RvSrGmhDemeeEU
QEoUpuyK4B0QRlc4EIYXFPMBkyRNTxGOFERCysylcJ5Tfgpwp2EkId5IsJYGfYJ6eBnTHD0RxTwdEtseNK0m
QQIijp8LpTwdOERxQJa7xkiRkFPnssNMiwU4X004QpaGCuBxpj7LmaP4RqKcey3+mHsoy0kCSP30XHRND0
WnspSdS8dEgfSfC6DSNkkSSW74Tc+2j9qlqSmHsmt9MPZEq5gjVyK3W4ey3bbR7KQ3UwW7Kceyl3yHZAllr
HsljbWPZH0dOMdEtBtb3zPDkXPtP+FNstph7VMhpfhJn0g9l2HMjNqkCT2geyFqi0H9qsjLQtpZD9Rbm+y
mFsFaiVhqrQf2oBrrMefSrZVftb+1BNba288lpLIC9RIQKq0EE+IDE1swTwrO1tsGTwgeotgyeEckXOmJCTLe4
Hoi+joJxwiz6AA9E609MGlBz5pVGZ3oaQ8KRqKl6JlpY28l4pWdEA7GMqwiQUtOOEYQU/ATBTahFtOgJaR

ZHTJWaYYWzSuhfwoCDCgRGWWHCb3cd04zvQ5LIVrHE3ui8SY7ruJx7oWdKfdcvPPuhmEKVoZCpHutX1Ax
1QI2fnquhmyEK8IUx7FuP9T8pidIPdaeYoAOZITH/AOoXF8wwmEy4SZ9QiAsFYx0dKEhdLgpkkqcd01vrQO
6l8OCF/nDWOOp5T1DUBROlGAeqclro0fqQz1GdraPeTFDMM9UQwyhQzBdW/uRTT3Vp7oBq2EMWwSVm
SBLQ4lCguAPdEMNSHd1Fgyfcl+4yuLosreKQFOLQCPBNQefTpvfR57l4EIK+/TD2U4kRWRfLQn2TJNbieymd
1GD2SJ9AD2U6sYO1cgOothweECVtodzwrSS2wHsh2ps4l+1FLZBHpMqt/R3+yxWN/on+FYIPF+cg8Ay2AC7
gL4AuwCFlum7Qu4C1AXYLkmbm4C3cF9AW5CizMjVKENVTECiyQlfqW9VKDOWkvXNnDlXu/R8OVlrlzLXX
A77F6XI+k8xLqV4MpqjaPG9V8Lg2dystqiP71WOcbakq66VvJKJqRiyGdE7OFINleAoxpJAAEd0hmeMtY4j3X
btIEEKsleOEX07xgKMoZXsxlvoqjeeqEZulUsPo3BOzHDAQHfPyn1k7CBzhCtbCBH8EJ9pWjIQgyZnGQUZOL
Q4twCg7LQBzCqkZmAahzRgZCO6ZmclFp5ooyAMuJRPT3OFuMuASO3VV54Ms1OiuIHEOoNueSnUVUDHh
pyCemQhWkuNCXZ82MAcnnsF3/wB4rM6QMkqoj5gcWtyOjervwEbvUaxqmlydxDD6h7ATJHsaOrieEsje1
4yRj4UXVN9oDRSOjk82mLPtPTafYn3Vcbd430IHepqOtt9XBAZSI6l7QYg0d3OHQBav2YHAJkpp295esNC6b
FGdo1darhTCemqIJoT0kbICOUfRvrXjOFGL0zzOzpnxFDmpE4JT9VA79WCtTyeCCiBYh9YM1Tj0MapGgppkj
T/JH84Ta+MhTqwgjKYxujWzI04mMrA3CnBgpWbMjS1rF0jblwrWLeZvbEojWhiTMghfdq3mZtjBJCEzTQlTe
1NcrV0DImWak8CDKyn68KT5mBCdXEFOrwKxOJDFbSjQbPSDPRS1Vw8lCFREExVonsTmR2+jC4ft47lwkIG
UkMYUuYMFicIjOQj2ji6lhbZgowpCOEM5zC6+8Jol0TQs4THCU9RvwEKY0TtHENXJ4PsvrJEOPIUckgzPnlDM
xIRIUNQlUsUgnJg/l8hlfMOUqlam8jBQ7rJUaKw5dS8pE0ldyCgHEPQz4XrUyLQtK4ua5RASZsz4+VNM5s2O67
SZTHOXIxQIG8Q1VVjPKDai5EZ5ThWF2CoypRHjcmNaAxRaSl/yXjH6kmF8wfuUS1VVICUxfXPB6o78sDF5vllj
qe9kn7kcUV1Jx6lVakuB3DISTb6/OOUBbpQPSTV6k57y0NHcCcepHNLXE45VeaCsHHKkuhq+nKUPTGyXZk3
U1WUVwTEqLqKYHCPqR/RCIMQ9WhpE7KdWNymaniKI4gtYhSzBGunkBOjIwlgIXU2wdNMD2SV9E09kYe
SsMKzMzZAP6AeyxHH06xbzi9ggjF2AXILuFNmMZ1AXcBcQuwKjM3O4C6EcLk0pRlcyQARA8JjqBwUQPTNO
OFIsilkd17OCOMvsXpcp/r2cFQnflzhyLRPMV3jiU01RBw9VNuYDZ/5VydUsla/hUyvzi2c/IWzRsJR9YuGimlq
NpacA490TCulecveSVGEVtyE9sqflMHGIukoU1ST+rCNKCPnDlQitBO7laO/JU0PmuDcNKU3OAO8KqBJHE
lqCrZuDWN3Eoxa52GML2tcew5KjK2wTyTtZHzn7iE93zUtHaJG0tI5r6st9cnaIJBhQAGAHJj/R6O27J+FQcEyS
xiFzRMRu/SwcuK51urbbb481E0cLR+ku5/lee2pvGWK3PmhpJTNUkkPIVebNa9ca/uUGZJK6MHL5TnY1droL
rV332eGgj5LKNQOta7ml/trf7R+mbcJlqfdVvb+zhqrRdf9o7Xd0zHbaJsLDwC1peUaWbwd0jYGNnuj3V9X2h
8zbHn/Enxn9LkqoZYbbS0dE8CGCEEMjd2z7oipOnp8GnL4/U0kZ9W4ybQmewEXW+m8Wbto2qcfqW1dY8
RN/QGxjG56tlp/RdxpZGT1tSZn+RHSsjLsARsGXfy49UnpaaV9DTurqmSBoazEbDgk5KJp77RUxcXvLGMDY2
nqSTycfKr99pfhUUck8CN6kiOWcngDkx71Fc7o2lpoYRASWBzg44aMcDgdgoQbJqMhINNQ0s0cryZNg+5va
MA9T3cegTBUarudfM+VuaG2MBAkcMy1EmfTGz345cVIVhoqqGjkna90c84aJqupdgQMceA0dvho5JUbv+
GnIXMkVi7eVmxDKz2awWBzaiKkcyf8ATAC58MRd1AYz7n/KJaO86muNa9k1FXww7wlyGBjxfloJKD3Xm1
W9kjIHZ1IT0Jc4Ndn5xwwfHVD8l+rwWRAOdI9vEEMr3Pyegc7qB7lBIHfPHf1MlyF7fsJalUGpY8Opbg52P+7f
6h/Kc2awnoZWx3KhljPA85jD5ZKo3V+LkWn52/1nUMrqnf6LzB5W7GtH/iyFH9p8cLvct51L9HNAR/8AbbQ2
QAE7zwVs6LUkoO319DORqkISM4PqJeuC/UMrA7e4A9MghPMc9PMPS8OVW6TUzLrGZPqYqeqY3wTyNe
3+HNPBT3BPax5bBcGNk67C/OfgFCiy6s47yQ002DPaWOMGftctRC8HkhRdb73d12kuiEoBwRIHdJfqKpw12
Y3ftei69ahOCcH5xfboLAMqMj5QhY0hKsLk1gfy3kLN5HBTWVwz6xSykdxFC+riHLbepZFOL01yJe96a5HrJwCR
qmQpU90SzPQnUydVKpgTwPqccoPqAMIEtXKDOwTrNyUapiuzEbZU3HqtpJgku8KfJgWBHGMp/pn4KHYIE
Q07eVCTJk7wshkOAnMSphtZ0ToIlDuh65ihkycWzcJAYapciHYXJlk2DE0sgKHKjnKkzTn2TXNTH2XQYTzBgPI
1I3Q5Rc6k+F9+l+FEExkRTBBkSWiHIRCKQDsu4p8dkA5EYIsHBT6aUeyKG04Sj6fhDzhe2AMtGPZMs1CD2U
kywJokg+FMrSFkkR1VuBB4UZ3C19eFYqanzhBNdRZB4TGqyLLqWZVW4WwglRtVUz2norXRC38HhRPLfy
eE9puiC6qRjTB4KPaGV4lSaOhx2RHSUgHZTWMPeAVCDDSiqnDCIG3VfDeVE0MJGMI4otwwk1gUxjWSJO1
vqs45UpUE2cKcra4+IS9bh9EosEdVMZLlJzhF0LUGUJyAjanKFjhI9xNto1iSxBOzGriGqOjZDFv5aWBq67Fqd4
jd5YWJy2LFkzEHguwXBZIESWKNy3DkgL188xZNx0D0pD+ExiRLGyLWJkVPcmid/CVvemOeTgrYmj2g7Wv4K
hq8vbhyk+ul4KhO8zfcia+8XWjiV61OwOa9Uv1FTZkcrMxySN+QThVrvtAXF3CsekckRRkdrayZBEcXITzHTt4Lz
gdh3Kcl6B3mY7d0rFHPi97w3gBM7bR7xKqH2mUjnOmwwBg7n4R7QvmmfsaSGlCkCbhJ56uPIR9b3RUltqK
yZwAYMNyq/qbAASPOi001TW3rWPqY4X/AFXTWCOfsBH1Eg2j4+v546p15VuE0FPOTJIT50vv8Bbeleq6ior
XRMIJ9/jKE/DzSrtQ6hhheD5TDvkPwEdpNJXTUbbOT3lpvvCIK04VeJIPhJ4X1Wpq1k9aHx0uc/Mi9I2VfttFlu
dstMUcc0VOHwxsGN3lnDv8gkVtrLdbqC5RUjGj6KnkxdGkqvlfqYWvUFsexokk9kZ7OImb6v9C17Ndq7SSOF
5VZzRtRN7cZ4jbO6rkrJa6ume+n3hrdvPmEfpYpB3K30LqGndeZZ5GMipmybGRs/UfclR54wM/pb4aCOcmI

bfLIB5dCfUCFGNBfppprfBGNrm7cY7jqSfchOVofm3ejDj5CRm0reAf0nke5l0otW1tzvrYtldrZwwc8Z3f8A
QKV/roZqisfMzFPA6RxcSD9oDflUn0o+v+pMbzgPa5/5PLifZShrDUTqWJ0DXkSOeWNYP1SO5z+GhI79MD
eiL/LHdNv8Fmb+aTDYaxl3vTqh4DKeiZtjbjifvZoh7j3TRrLX5qqz6Gln8uOm3B7wcfj1Yw/v8A3v7dAowv95f
pLRsNPA/FbUNG+TuHP5J/IVVbXeQ3bK+N5Y53DM5dL7N9s8uKko0QtLW91XhBOLtX4Ywv9R5aXQt90ipa
AVk04p4WJDHu5e4DqW54AQzPrCsusU1HaYpoo5sl72Z86UHj17n5JwFXKru93vNwLqiRIPtwANDizlAb+xnc
oxotYUtPSVENHSEtOzzXyy4MhH/AIrhj/09AizpcHO3c/8ARYMLww+Lav8AUyz2lvC/w/pJxPcuZxN3ZDjO/Pc
kn0hWdtOmtHO9EFqkqWADD6mQv/yjbgBefkF7rauCkFLO2LzjuayEk4jb7k9ASrCaX1Q1ulnAyRknLncNIHfJ
KS6mnUnJa1ifrGNLUdlQD7S0IVQUFrhaygobZSZeHaxrooXH8mTKjC8nVMjGTSVU1GYxwWOHQonfJPGE0V
Vx0PwMdhPP621oB53xlzj8ghC1Fp3wofVh7Yaqz500ijq5mt/lpQClg5ZwJ99u7+5k7CzPIIA9g2P7CSTZ/FR9
BIBVtkhMOGSS8S083s4PZ9qs3Q6zs1e1m7DNwGHZy059iqN13hbX09U6sslbQXbzGYLHudSTuHbdzsk/IRtJ
qO+aREAU2nZKalleWPcx7g2N34duDfhrW6Cm/wDwWGF5c4/oZJVrbKT/ABVIHvj/AGnq7Hc62AB9JUIrVdi
NbZyqgqilZ3CKb9ruF5oWfVvblxtZZLkyshHL6dzsStHx2Klu26ysd8jAe8Q1TDgh3DgUoNeq057Egdx7RoRpt
QO4Dht856FluRljgUjdJhV805qqroy2KeUTRdnZyQp8hnpK6EPieDIMaNajYB4iPU9PdMIZxklTVJKtarzoTzyP
dDk1Sm64MrthK8GKZ5kl1c3BXaepwDyg2rquvKJURe7xprp+TygKpqsZ5S2uq+vKj2rquvKMRYpseOr6nJ6r6
yce6BnVwB6reKtG7qiSvEC8TmSnTSA45RvR44UR0ISCQpKojxxygnEPraSLABgljYChmmkCKoHBBtGyRdHG
nFsa5RkXjbhRQtYnEQKTPgBTyGrCxakmBBo0oWwpRhEHLr6l1wTJIWdZpRnotTSj2RR5QWphQzwhRBoUo9
l2+mCIBCu4gUMJEDH0oTVLRqRTT/CRYUvCzJnJEimajQtV0RwVnJ6MHsh2qoeDwiUaBuuZXsvoeDwopuNB
yeFaCvoDzwoouNvOTwmNdkUXVYEBR47JdFBgosdREE8Jl6AtKLLxWUxEcTMFFtIBkIgYzBTxTnDgoGadKOZ
KFvA4UrW7soat8qlS2zdEtsjSoiTRQdkfUzVHVteCApKpOiDjqEcLeE8Mam+BPMYXAMYCdmtXYNWzQlAau
pKBE20rEq2rFk3AJcHOW7ikEjkrMn10iTmVI5HpvfMsmo9iZOEcqDxMcpyilWTAY/SSIfqeCu8kiH6iRZMMH
a+XgqFLy/hylqtdkFQ9eD6XLtWglNaQBd3ZeVG9QxpjchDIlyjy7P/uuQPI4Eke6Zl5lVbAzAiShaxsrh7cLeiYGxv
bgZcESxBr2yxHu04/ITVHPEXwDYBkZySpXtYqRF6ooYGCxpyZ8Ho0ZKifXd+/8Ap7ImvxGCTj3AUt3yV0cL/LG
HskD8Aqpnio9gfsDvSxjWj5Kn0yeJcmYdpV8Jbn9cSr9dPU1kkh5LnK7fh0W2TTwFPGH3CuJbGP2j3KpzblG
yXKEOHG7or6aSoG0IBNOHDz3MOHHpGweyc6/aKwvzHElpQ2sOeB3k0WempYaI2yWoDH1cBdNM7vI4qo
niJSXCG40daliPppWRSdp/eRdD+C1FFXqGCYQs88Ndh0Ye8kbm59+xBRs2WDUFNNBNABXRQNZLGT6apjft
laff5QlKvQwY8g/FC7Ntq7VOMY2yJdcRR33SLKuNzvqLe8kBw5fBJyR+WqHNEMa3UllVY/fTukleT0DWs7hW
DssgthlT9aDLSE+iRw9TGU4Id+O6jZ9ufateXbgtp56eUxHqHRYjHcMRsJbWO2CV+hkZALVue+RuhVoTb9TA8
kkvqXNdk8kNO4k/krrV1hqNZy1VVh7aMOeGdsk5Gfyh7TtSyllLcHax/pcO+2PP+q1he1rJ5H+p1XVj/0R8/6l
Duv8axvdcD7wld/BrX2bJg54hXp1TUMilO55aC/43ckBAdC+srypfN8tsbeAB6ImDJOB1d2CS3qdtRd55nt3F8
h2MHxn8JXc611DssomHEsgzORwGkjho/CaqgWtFA5xFrXtdj2zNrhflWRR09M0Njijy3eeXO9yT7IMIE2oq
5qSjX4lk9Mf/u4oTDiMZ6DnlPtFWTxuqJGZEjoHQsd+wO+4j+FOFAXiQsSTkmSBddVNNxnkDi6FgbFTWj0t2
MGBuwmhmq71M8sge6Nuftj4H8k5UbMplZpQxgc454aBkn8AI9g0hfpmxsfTzNa77WnbGDj8R7a1AGBWP
WTcnPmMexers07vrYhIO7qgE/6FHdH4laljjZHKKarDQBuY/MuPz3CDqbSNLA8fUV1BTbfu82YvP/APLCKVoL
Zp2enYJbIT1YaCG/T5jfn4dgZQ9ngkDcm77TtDaudrEfeT/ozxQ03VMhbJJJSvbctOXHYcqWVJ4h297HUV2pJL
QVRMYmkj82HHS8/pB7FeVFwh05FVv/AOW3fe13D3luBj8cqWtO6yt1G8RmOUwPbh586Vh9sjOW5/KR6jp
ITZdFf6RpVrnXCuV+svBU+DTaeV9x0nVfVz6luzke8R7j4Q1LbaC8PLa3dQXGHgyj0OB7b2nlKdH6prLO+mJr
TW2apcGw1TuH0z3ceXKW9Af0uUwaorKGRgfd6N1TEGlsVfE3FTG0diR923uCKDPeLarsWP6XHxRmErCEqA
oPdD2kiHUmoNFytjun/aKVxGypbyAPIWx0T4lOU/lyU9U18Unyq31MZZaQ4mK62p49MzPUA0+47KHqjSs1
sf9dYpylJOTCDlo+QhrNLRcP5Hz8Q7H6j0MOr1NqcfGmOx+lf7ie1INc6evgBDxkhCtyppogXDoqHeHPiZPLG6
lqj5dRHwrj2TVIPcYvlll3patl+msKuOBM1Gjp1Ne5IPVvbjlygesr+vKLNSUESadIB6fdV+rmbnIWuhlsUMs83
1SvU5VxgiOFdcBzyo2rrmBnlNVwuXX1KL6+5fKcV1yvW2wtluoDvuSmC68/coLnrzuPKV0lxORyijXfVjHMTn
b7l05UsW6vB28qqNurs45UxWytPHKXWpGdFss/RVgOEaQVI4UE0FX05UhU1QeOUsYsWVPxJXhnBTwyRR
5Tzle0MxKHjFWha16VBMSb05teozC1i0ALcNwJslIiUJMJJE+Bq28vldg1KWMUJhCjmcWwpY2IJSxiVtYuZNGs
whcnQBPuxcjGsmQcdThM89MMHhGjo01TR9VkjYCRdWUTtnhRlcbcdnhTvUxdeECVsHVTKYDYgMr5UUO
1x4Q3UuuD0UyVtMMnhBFVBwUYGiixQJGsseCtWPw5PvTHjKJHDgqSBMcGGtHPgjlSbban1BQDDVYcOVI
Vtrhkcod1k1dktPaqjopcon8BVrs1cCW8qdbfUgtHKXEYmFUPmSha5PsZQhTTdESxPXhrGymPTUqCQMkXN
K6he3wVi+8LFkyRu9NUpt29qZpmFd55mRgmV8ieJmHIMMJHZXWZoifWycp0jITM1pStmVvM0BHeSTh
MszktJ4TXMuSZswZrHcFRHeD6XKWavove13+1y5B5gtnaVxvJw5y9zuVlt2awvO44Cj2SEtPdshNa2ErN/ed

/qgx8b/KaeeV3npKN042HG4Zwm+MBxLPdKqSF5wT1Y87T+VE5weDicV8g8ZgHeadzrjSsBIDck/wqo6moH
196qy//gwkEq69/iluD8Nwfl3D4yFXqS3smrpow0kNZvcf3I/RXY83/TGwpydsrNSOUVvk+tfqOIJEY7Mb3JT9
W61qoGiFr9jOBub3afcLvr6n8iKliZjOAOIH55AQNBQmoZ9bG4Uk7QPPaN3IE9+PZWxAjoHYZ74ix96PsU4A
xmH9v04+60shp6hs247mgHac98A906Uxv8ApupibclpWwtO6GqjG4wn3I9vcJ/03o29W25udSuZX07GB0rI
ZOXwuGd7FN2mL/TMnjoK2WSW0yuDBK9gL6cv6HJ6s7OCAu1BG7GLF747GF00g44KN2+UGDU0d6pjM5r
GziPEjW/bK1/SSM9whqtoI32Z84Jc+nhcx4//ABv9v+VylDU3hZW2zzpbTIY3BziKYA7CDyXRnsD7KKrPPJPoy
Koa5rp90EzDwW7xtJP4cAUGjoybq3yAc49RDSrBtrrgkd5X4zmGkidvycOJ/hmESNnHn0DSSQ2ASux7v5AQh
dY5oylSzbSlewtl6FvCemgCekeCAOCJjvxyINWAOPnmBISM/LEyrJR+fqGvqHgGOjY+R57ZHAx/Kj24VP1Iye9
vpDnZJPJwOpU00VJLFo6vnYD51xrnxf4lwTICdDp+CkonVLU1zyWgtjH+gP5RSMCzH57R9oMwwF+mZXH0
ckhM0j/KhJO0u5LsewUq0NJa6GORPkiDzUvMu6U4a2NnpBP/MeiMtMaBu2o7oysrwY6YuB28Ny0dsdmBE
Gt7ta6eqFFZ4IZi1m+SqkaCjRxuGf9FE1ytaK18xHLEdhOxW3h7ydo9BlhdfLm6nbBarYyASEjz9mHyfgnlIX2b
VALXzykv42hzgXH4a1HdgtN9uMcTLfHNU1dTuo8DaloRwXuceGh3ZTjSae01pq2vrK6tdNK9vD4eXyO9oye
3yubLkqbA5Yn4RyZutGsXttUfqMrW2m1hSxkyGqcB0jfh7P5B6/gJyorTe7i7d/SZ6h3dppC0fhjztwie4eJlZSG
WKgo2UjH29Aw+X8veclRhWax1dKWSTXKRjXDLGiXk8LtRqD+IFzMJq/mZpOlt0dV1LdvlvtCS0NBmjy0H4L
MnC+XTww1bGyWTbGNhw6aObGf4Cg6m8SdZUrh5VxkaB2JJU+aY8d7pDPG250kdTESNxDigrB1FMIAjfk
EquhYYYsvzgVpi636w3F3IVIDXwSZbUUYmA85meRteB6l6b6cuNLX2qKSCeRtG8BhbKp7tNL2bIPj37hVW1D
bdK6zthrLcl5pT6pKaQASN+Y3Dkn3Ci7RGqa7TV6fSzyST0krSMSEI7NvWN/uPYpTqavzVZdV22L8S+sPofwH
CM2Ub4T6S382nry6yrumnXMcGvP9Qtu0mJ7j1cxp6bvZZbnWq6RPqblSx2C6otruHxn9WwFR3rTUuptJV
dFf7NOailBv1UL/WyeF3LXKQqKq0zr+GO8aed/Tr5Btlkgzt3/ACPFKWNu5pWw8gjg/wBvkwhauFtKgYlOdv8
AqsFbhZqWoP1UBMcv6445HZwX21aOrLdXMgqGuY4dHqSGsfelZi6MU11i4kZja2Uj3HuoduojqKZ8NVFsm
ZnqMEFCrtfyOuR/URgCR5kbB/oZ6BaZ1XS3ijEExa84UTax0sYpXyQOWD2VJtL6nr7DcmgyI8eeOV6F23UNtv
8AafU4b9vCBK3aG4EEmtPFqtPRr6DIQLFIE73JUUsjmyKJ6quJJ5UpeITKuluD2P8AtzwVXmafJXo+lw9St7ieG
6wGu505GDjmOjquTjSuOQhWJ+SiiIPRGMMCLwcmSRb53NxyptlaRjIQtRckKS7e08JZasYVE5IglbXcBSfR
Vmccqv9C94AUIU54Sh0j6lZj8pJgQOUWQS8KJKGoPHKPaebICAYcx3W2RD+GVPUbkH07ieAoYxiklIovam
2MpxYVCYcsVtCWsCTNS1iiMIXtFrAlQCTtKUArU7nXC1IC+5XwIzNzg4Jtlb1TmSm+QrJqDU7EH1UOQUczlb
qB1WQRpFlbB1UeVjAMqXa1nVRhcW4yikMU3CRdXenKA6mUDKMrm/AKiqrqAM8pii5iK1uZsasB2copoLg
ARyoQrq8RuPK50l65A3KdqSRAXdgy7NmuXLeVYW03HlaMqiWn7tk9SszZbgCW8pJchBj7TXZxzLZUVTKDI
GtPLwFCdtrA4N5Um0k/A5QUstT5EP4npyY5DUMmQnmN62lwBjlvWJNlYtzriCrgm+RidyEmc1ZOhBeWJNL
4EXvjTc6JdAzIPcNwGnIRA2PldXRDC3mZzBMsKbJmorkITJMxcGagLVt4Kii7N9LIM1YzggKrrHlrlwO8Dt7Ss1
7j+5RkHFhIPQqaL0wjdGKIjPc5wa4AAdymtZ8sq1x88Sww2ynJyimOV0IO6SIDp6h8pphNftLH6iH9/ZcoH/AE
M8vq9PXkqJ+T25m6iVhCYMe99LXxev0yeWWZUPyWySC7O3gD1xgnsWqXWQU0hbMyQNY72KR3igMkA
cHFrwMHuC1R1ttYj0MslB3Vqe5X+olTvEnRV2ra6aegpvO3AEtaRkYUCWJtdTXGC33KkftNlfgF7Dtz846K1F9
qKyKSB/mmf8fDZmEgOA7EhP1pvluuwjusMdQWOa/eQGuOz2c3nIVlr1dtdAUqGUDuO4kL6Wp7iwbax9D
2Miui05XmCYWuolprlbGSTUrGPG8AnljXdJlZnc3uE826uhvlOZKqBtHcWtMfXgA2xVDZOG1MePteHfcpQve
kpZWRVtguINRT5dG0u4mizkMLuz29igyOd0dZTXGSjPE5hq4iz7PMHV3wVEL1sUkHJ/Yg+30M7FDIQQCB+4
MsTYtUzTaUpBW/3JYwftTz67mcDge4Cr3qi0RmvZUwP9b3DJ/e149B/IIS+DzaStrYMuw53nMIP3A/9QIinY
+piqKdw3ExiWP55yQgkAqdmXsf7GHkelqg9xKwaxs7WVrZNh/usMpx+4qNvL2RVg9pmObkdjwSFbvUFqrJ
qGSJrh5gjLckZlAdn/NRZLpriWRo9LowJAP0OP/QprTqV8NcmDWaYlJgRF9NBT6ZtrA3Lts0A9nP7pPQ2aa7V
8UW8R26nlknJ5O2MZWplykaCy1L7Nbc3F3qHv05RtdrL/TNKsjZkGoMMMmOSd3rkA+eQFyNSF4B8xY8zT
aUtjI4AHECLnX1Q0xTR0m2OpvL5GsYz/uabO3P+QURaa8PqnVF2fJO59LZoH7S/o+oMfBEf/yp9+kNbdC6Ye
VTUz2xjAw5zImBojHsCRkp8r9RQ0kNDT4j3SAHYMBkcY3EN+Aoxe6KvrGGbkn2nBoVmDP2HYRgvt0oaaao
i07bgIaWfuaryTwQORdu6/kqkGrdXvt1uNRI7hkXoia1xayJg4GMKYKCW41outVFG6SaQSNhaOrnSHGV8o/
Ctttp4aq7uOOCIXzud0A98d01oFNHc+bt8yfWLLvFtI8p2/OAlYaQXqMkmlh3cf22dGuf23n2HsvkNq1BXP3x
UM8pefu2YBPwSrgVdBaaaSOKjo4hO1pL5Hs80R98BzuPMPx0CVUFMyoYydwkka87QXHMs5P6Y/Zv44RZ1
HGQMSFVAODgmVK/wB1NSscwGm9TujWuD3fwG5KcxpS7xvAqWQxZ7vnZE4flrirSVd/tNsIfDEKeDaChilbv
yHPPLj/AKILI8RbO10m210j428GWeJsJnn2a1cC/UnkV5E6NdJ4NgB+QjJpfTus6SUVFv21DG8g08rZHAj4B5R
3q2OrqKVIZLsvp53Pa6cFpbskbwJBns7umal8Wry17N2m7btDwS4sdHJkdx5i4VwNM670Pq63z0NfRbZzmg
Ojfl4Ozn9pd2SvUajUVotj6c4Hcj2h1NFvINWt4z3GfeAuk6xmoNIMPzGwOpWoge1v7HchzfwqaSuu2k9Si

WIMIPlxzyYSACHDgP/AHwvQay+GTLJfnPtFZJLSyA/VUEvE8LXD0ysH62DvhRB4oaZkqKU1UUR86KyeZGDxl
5vDZWezi3gqLSamgal0Ug12c4Ppmd6mi06dXIIlevvj1xJh0H4g2/XNHGPMjpl1TD1ZPEuP/dqkW8UYvMRZL
B5Naz0yAjr8g9x8rx8ZXVntu8VfQvkp6mF+XMPByOuP+oXpZpPxEj1ZamMimZFdkVnmCB3Amb3DD7oXXd
PNTb6x5c/+P/xJtJrA+FY+b+//AMyKNT6bkp3gRuLZWnqUE6e8QrjabiyF52OY/Dhnhytb51t1ZRvDAYqmEkTt
6OYR3IVldXaerla58dVHscXExTN6OW9Ka7Vaq4cj0MlvLoRZUeD+xnpXJQWLW9hZMwjzQ1UZ1TpGa11b4
w/OCgnw98TLnpW7MgqHEwucGuV4dUWu2antrayknG9zccGk39P1AR2Jpb4T3AiXqei0/UdO11Vf8dfjXsT
KExNex+CjCkHITTXW2qoax8coOQeCnyhYSQrtkMolOOZ5KFKuQRggw7oR0UpW9vRRrQs5CIW3NHpS60RI
VD2kZOR1Rs6ISpAMBGVLnISpxHNeMCSBQjgI9peyBaLsjqm6Ja4jqjGvKieFCVMimEoRhGiR/jKcmFM7ClzHI
cw9THlhSxjk0tclbHklwhY8scu+5NbXLrvXMkjhvWF6bt6wvWTlqL0je5cXspl+RZOGM4TFDs56p1lkQ3UPW
QdoM1p4Kiu6P4KkOul4KiS6z8ORVYim4yJLxPjcoNuVYwK8qTb1NklQbdn8FPKR2IXuJyE3OtyeqFGXBzZBg
re4PyShtrfUmgC4i1gZyRTI3dvaMq3en7jkM5Xn9aJDG9pVp9N1rvRyq9q0EzaVyGI8bLWZDeVNFDPkDIVSsd
dw3IT1bqzICr57y5UPJrp5eif45FHdNU5RRBN0W44Rsw3rE0ecsW5Nmd8LmQIC1IW5LEDmpE5idyElc1ZO
4gDF3LOF12rphjZNXhmYmCdqLJmocqAtzgwIqwMFRddANrIKVb0KiG7EgOwVgXDOWzgzSDL2eHdIX24HbK
VNd8lfl6rxdah4e446J1RXKXq7fNE7Kvy6lp7Hgo4MDqmaSN+P7keWfKhv6mQvAaME/qPYKVLHQ1bWMM
kn3RZyCTJ3eoXnMzRI3YKATzCewaWuj42YwYBnlKlbpTNhiZGyEOIHynaLUzY4mQ00WU6T3NnkGSVsYeG
9uElZ7C2SjedLpqqIAVsnHMq7drfmJxmoS+LOSA4tIhwoldZLPLPvoHvhlac7JcQ7j214KmvUupGODjuDW+4K
iW3VIRcq8R08biM5Mjm8fxlOKjatZY5A+vEKK1lgOCfpzFtqhulBUMdBmflccgOblrx3Z+QpwwpxXRVD9pEkj
Q14Px047omtFkmkZ5csHmbsbjhTnaNEsErXx9QBkO/wCqRX69d3zlhq6e235StbNLVL6iGRkXrzj26+4/BUq2
nSDT62NJGzjjnDuMFWmotKmNsR2gEHg/n3UiU1gpoXtcMdCCMIFtbY0KXRospPJolz6oNd0kilPHsgap0A+C
tqSIMMeyMEe20lejUtrpvPjeBg9FxrHDNJubHTBHuoHQ7B2MI/Kp7TzjtujpPjtuY3FrZiDngYOUSu+mJag0jX
uDYozXOIAzKjuVfUaZpg1jNgw2TdnCQ3DTcEsblo2N5LsnH7I2urfdmcPpkxgTyyu1kuRqC+nAMYj8zHycj/A
UaV1guMIQwY5FOWMHXBdvwWp2haVwfmPPAaR0yOpyudFoCh+vdJ5HRwdnuSOyZ1ErjLH0SnM83LDo
eGy0RllaTKQAxmFFeobtW3G5VfV1xZC0R0zDw1m3jcfhq9Y7voP6qoqpCzDiwJ/ANfHHSQoof4N0sZmkdD
uY1oJZ77egz7EounqS7yz5JMct0PIABwJ5SGJITvxiXeIZnNOOQ57OpGO2ep+E7amv/0dtkmhDWz118mma
HEMTfSxqXSu3hLVbpCyDdKWkf8ArOdo9vcqv2qfDG4RvYwRF5jYI48cjPs1Pquoaax1ye3pE1nT7kVtvcygVa
2Ux5JOWHkE+p7u5QIL5j/Xw1vbsB+Fbe++GldSVbYnwpFw1p28lzyQFFt+0u9ocTjzGtwGM5GBxx7+2e5Vor
1FVgGGBIdei2onKyG46+rbgNnf+M8KSdNX6rhrGO3O/5genw4HIIUb1tvno5yx7cOb1HXB9k/2eV1PX08g/4
cJgHe2CpygZGBGZGWcspBwZ6m6cuc2pdPVUVNN5N4ooHVdse0kPzCNz4vwwQOAivU1ay/afiuVNEI5Z6Vsz
4ug3xjJ/B6qDfDLfRa1oYuP7UrQ13fB7fyETaju0VuvssDagCJ9RHHS7Rvc6T0qgWV7dTtQfDhx9D3EuKNmnL
HhhtP1IWNVaVbcYJa2InY6U+rpjeeuD7PUHWauq6WsaYo3RVsB3xEPSSooVh7qx+n7/U0kjc2+qe2SEnoG
u6AqE9VWyOiu7SyXayXEkJecY+A8eyulDbkAJyCuV+kqli4c4XBDdeb5GWTodbVN2p47rRBkNyp3AVdMz0+e
33PuVFPfCLFq+0JfDhzeJGfjgd7rZxpLjJTXOCuhkbFK0gT8gsk+eO6n6z3230N1NbTVZ+oJy+NjCGSnd+Un1Wi
AGUBBHkkenyjtP6vJ8xBB+If6wJ1lpb6KpdDke5MMvZwR94eXu7UsRp3vcWt6KzkkNn1BamNmia+GTLrv1R
OUONsv9Eu/kyg+ST6H/CBbVC6g1uvmHpDE0/hXhw3IPENaQ3QXhhcAN4QWLNUUzyHN6JNXxios14ZLHIO
EnUqdqaWmusDJWEchEaW61FUE5UjiVLRGgqZ2sQYYHzCAFDSnjhSPRO+McJ2prQG/prVBbiAOEa1oMrNd
BE4U0Z4RhsDkJPDRkdK/wU+ChmYGHKhEKKM9EZ07xgcoQpoyERRAhBslyThzTygliimCAYnnCe4pDhCMSy
oTDqKUJzblgqKRyemSFBMIwQmE7ZErbKhcSIKhMhzCQYtIw5mQz9QuLqr5XMk3Qq88LUzoQNZ8pO6vA
WTW4wqfOkpncEn14x1Te64BbkRaFss4Q3UzjB5TNJcB7oZq7gMHISKsgd5lfVDBUNXeqbh3KebjcgM8qFrX
cxh3KpQsJb7O8EbrVguPKhqt7Tgp8uNwy93KjyuqNwPKelMAlhJyTBedwcSkETdz1s93KVUbcydFKxwJwEhvb
4uinGwvcxzVFVvgPcli2twQkGoeGJVggyzdkQ1Tza6vpyqr2iYtLeVNVBW4xykZjupsSylHUggJzqgBjqoJobg
MDIH9NXAgrQjiuyST9R8rEffWBYu4T4smRfMLZbYWRjOBC4lqWELkQsnYiPC3wumF9wsm40zBDNSOqKp
gh+dqycmAFawkFRHdoyQ5TdVM4Ki27Q8Fdg8xfvEq1e4j61XS6CJkrt/Q9Vaq+Ux9arZd6Bz5S0DkpxS4IL1
dfPaM1NRRVc8baZo2Y9TynWeqqPPFPA3LWcOceiJJoWWy10tJf/xpeXu9gUF3W4w2+EtiaY4Yz7ID5ZuBn
2li0em2KM8MQccenyi6o1BHSt+mgjLpnDr7fKb21FY+mLX1Mb3P6vc7DWhQ++vpXzyeZM/jmXYeSf2goVu
N7NSzDaNgYz7Q52GN/PuUXXpSccfUxubAohvcP6CepZBBVgqcD6ntGlwfhW/wBCaHijpYpZBh72gqg/hhbJ
bxdmZaHxxEH0t2sXqvbqVkedAODp0wkPV9R4ZFSn6yydH028G1h9JxtlIgia0ABSXbqFkcYBABBByMJvpGxkH
hFsQc0DkKnBuZdSOMR3iHDSNXUsJ6rmx+O4SsOB6ITF4MEwe0SASDG1uR3ShvLrkLo3G/hdQzjrtPstAmdn
E1DSAScflLoIG7t+3lco8lvHUdQlJmg+xUqwawZBE6ywxlpw0ZXZIOXuMNXVvQJQAiBF5XiNjaWNrnEDJcclcn

0kboyC0cnJCdiBuX3BW5GRAOXT8Uj5HOwcg4GOhKGanRlvkaP7YJbJbx0wpgLeUnkBws7TMEemVI1ToCjq4
mRsj24BDpAPUqN6l8NKOgkmMLGtcXYDztt4xwvXOsjBBaZnGMquerLQJlzgd2muvpYYY4hC6Gm3Uj+f7X
Wip7dO9zu+doA/1KhK3RvkZsBGT/wC4PC9UvE+xB1JLlpJaCV5g2eINuBB6skx/qvWOLA46igse4nm/WenjT
WqF7MZe3Qxnk1xE3OQySnc53cbByFDGubkanXOOqYZi82rY6P4miH/Vs/4TSmbU92qnZDIIPM/+QKHNZW
2SuutTcGk7nSb8/IA1Mi9RcN6VKPv3k71O+gQr/OT9pKW+O/aUpXvjY98e6CfzOMB/I57cqC2tgr6eotdSzeYj
/acThzHj7SD89Cpa8IL5T1FfJR1QaYqg+tjuhJ+4KPFEC3z2bVPmxREsy5j24wcNPF5HYoqg7b7Ke2PMsBv81SX
e/DSvIXDJQ1skJ3sa/AIKLqG4yXCmFPJl1IVT8wyYA8xo/S759iji+UkF9o2VDOJXM9BA/WOx/KggMqqecOc0s
ew85/8AYpyp8QdsMlpYbD3yCJaXROuamge6OcZiHERt1b8hWhqZqK70Aj8wH05jf3C81XVm2SOZh9D+vw
7uCP10hqT6argp5nFOEp9Dv2n2SXW6IN/EQYAnNHqyv8N+R2zJdbvjc+IrmB8ZOGPUw6TtZppGtjk3Rk8IKu
9uBiZURgyR9XAKSNkZqBGFj8qv2WsfBHPMctSjgqwxLk0lR3Rg4Twy2kfpX3TVUyb05ypYFE32U62FgDK
ZbpyjKykYtoSOyclqP4R8aMey6spB7KTMh2fKb7KdwTiyMhE5pB7LiafCzMkCRsYE7Qrh5eClcbeihaErHalOz
CmyMJc0oJoxTtO+/lfDKkrnJG96Hik8WvqcE25VfPVNc8+Ekz1mCeVgE5JhY+tx3TVLcB7oGnr8DqheoumM8
qUJIWeSNJdAO6apLuPdQ5VXRgFug+ovhHRYlWnMCe2TtPfgGfchWqvYIPqUB1WoDz6kKVgonbT6kamm
MAe+S9cLyDn1KILdAd3qQPu38nPqQFWXbdn1JrVpsRTbdo1fxjeeULyVm5CITXku6pEKk9UxNeBALbzQ
sD9xRzBvYsvGQgikJdhSVbQMhKb+BGda5kiUTMYUh0XBCAKVwyEd0buir9ojFRJRojNuFIVNWbccqK6d4ABT
o2uw4cpbtJM6PEnujuGMco7pblgdVW2muWO6J4rxgfcuwlW2WC/qg91igf+t/4li72GSePPTcL6sC3UMt81
wtCF2WuFk7ETkLRdnLiVk3G+QJknCfZzTLRmjBCqHBUb3JnBU1I6oAr48grMwO2QDd4M7ICFRbnmpD9v
DTIWZuNMPUSoUv0hipZDH16llH7Cl3pDOCZAV3rgXT1D3YAz/AAAqwXq9VVVI9jXgOk4jP7B7qSte3uCGJtJ
GckEPeR3z2UI0tE6uuMMMDT0AdM4e5/SnWnrAXeRgAk9o9UdMyCkdNJktwWR/4ndymD+iz1hhFQ8sj/TG
3qQptv5orRTU25rJZWjMUZ+1v+Ipo0NFV3y+XvPqYHdcYyfhF+OUpazsPedeEhtWv+kun4Taeht1rbllducK2k
GD3CAbfSinhp4WgnaMuUhwZABwvJ9XcbLnY+pnrgjPCUoo9BCaAlrcADp1TpG4NOR/KYAQPTwUta8gdUu
3RqFhPE8ZGBJkcmv8AY5QzHNw3HZOsTWO0n0n3XYeRskfWY24HJPVKnpDmHcfZnZmbuAER3ZwEQpgbAg
xYMF+ehHRKmuP8Jv3bcZy5dS5oIIPBU4MgKx6DwMJW0hMzHcA7shLWv54U4MDdu6nstsJMHjv2XVrwc
nKIBEFKmdOCk7h2x1WxOMri5xXJMwCMtUwEFQzfmGSNzGgd856KbKh/plUTXsEMeQAMISwRrpycz919
QzuoqkFnBa7AXjg4Cn1FJH2+o3uC9ttebjFN15ByvFa+QtZq+r5wC8n/NX38Nudlyn2IX/ABLWCIDD+eW38M
x9Dpq83Bx2uqJfJi+cu5TpVW6LdKBnaeCOwQxba9jbRaoW4bD5m1rR+OD7j8kqVfp3S1G0Da5zf4dtUlZMLH
sP6mJ+w4kGkVdip6KoEpbj3WbV0kJJbueHwnOOT2/ITH4l1TbYLVWgnzWylj3jqQW+klAXijTic60DmDaS3
lcOuUUWWP/qWkq6KVo3Ab2td2kYeSVYw29NNqPXgH+oqlqbLdVp/uv95ENguVZTzuohJv81m5jT3cedqaL
tQw1uJ4HH+5nAPUuHVjvZ47e6brhSVDI3VUMxf5co3uxh8bzyN3/Qpa64b5fq44gPMwKilSzoPl9s9QnGPN
uX7xLnYbW9O0BKcxvd53MefWwdSnsV2S6Jvpid9o7hO92pKatp/rKY+sD++zuf8f/ygERNynA9WnhTDDDtl
WGPUS4Hh/rt0cZoq0GRgG0/j3Cm+z3CjorpljswPPH8rzmtda+GcPDsPZYpkKzljvcUnlFxzHJw74KrmT0a5ZIH
DDkR9o9UxARjyvZ0600fJq2SRuzHlrXU7N8TTheXGI9Xy22tgp5H+ZC4jaV6WacubKukYWnPAVcqD1na32
Mm1tav51+4hlacL62BPG3IWwYjeYgxGjyQkr4UQFiTuYtzelMujWoYnl8aTlq4YyRRzPjAlYHC5MbylWECOPW
IXJrmKeHjqOihkkGKqTAKAKqfkoXRyBUY1snVSJInMaauqwDyGkurPILa2owSo6r6vryjq0yYC7Rvra7BPqQT
U3Dj7k23Gu5PKjuquBweU7rp4ET2PH6qrzn7kLVVadh9SEqI4HPVM01cSOqZrTF7vxHearPumCapPukL5yU
2TS11VxAHM+vmy7qIULySEO7juT/TgcLLAAJzX3kh2/oEfU9bIbKAjARhCq7d3MsNQ4kj0s3RHHOOFFFI/oje
keQQIDqIzJSZUeIIJKwh/VNDZvSmaechyhWsZgtrYEPY693uur7s5oPqKjxlUSEiqqsgHlSeHamsIEPv66/3KxRB
9cfdYpfCEi8Y+8/RoFsvgXQKvz1OYublOJSdxW52JycVwJX1xXBxWtc4vKZ5k5PKa5SFozRjBUNQjUx5BRnOha
r6ELiC2SJLuG7S0d1WvV1c2gpZHEB2wEq01fExokeR0C82vGTU7YI5YIyC9yJ06Gy1VEAIEqRdK76ivnq3kFrXn
YPdylnR9sggp56moe0AZkmce57McrZa9sstJCw5ex4Lh+4nITfKx9Faqekb97xuPYEu6l3wFaL1OEQcZ/sIRQR
5nI4X+8FLXOLXtbv+GX+ouPL/YAdmhXa8I7E2ACrsfbsMNH8IUZO/QOouoa0vq5w4OJxiKMn47lerui4BDdb4
WZJeWZ4SHrN/h0rWOxjrpFBsulKciSpBGdKe+ERQujADs/KaKcNY3CdGBrcNAGAF5o7ZJnqFSYURd5jsAgY
9ko3kYyckpqMnLgStBJ6un29fdQGghYURSYcBt4wniKbkcoLZMdw9HbnpnqnN8ga0gNw7hbBmmSGUcw
GeErjKdu34QtE/kJ4Y5xcHAcA8oITAXrEIgyPzldw9pxjghNgc73yClMT5CGlv4RQJgZWOjSMhKGPaON2UIYe
BwSlLdxbjCnEEYRU1x/KzzeeThcgMD2XPAAHyu+ZBgRWJGO7IN0xPRrlhb0xwkEnsQuSZIijM5I3oxuOfdRZ
epcxv5RvUI2ep6KNLy0FpAPOEKzRhUgEp/rp7hBLh/ODnH/ALLyG1ITil/meCN83A+AV68axgDoZGtGRyDnu
vLrWFEGaii443fnpyrt0NwrN9JXut1b6B8mEW/U7ayhpw/YlocD2yOAVbO2OE1EyTAP2OyD9rtuCFRMTGa+
U/rIzBg/wVd3QBE9olaHerGR+WFOepV7dOjf/wBzK50q3dqbf+Z/piU8RLBV3GS2CMbqiN8nmDoCH8twob

0Xdplbw2mncDFUMLf+Vw6K+94pYm3G2yBh5cWjHck4Xn9WWWWK6PYw52yvkjGNsgaCQ4AdyiOmagW
6dqm4AXj9zB+raY16hLlzkthvsBHHUIC6guXmMA8mfdDIOxyA5QqJBR1rmPy+Jww73LOx/IVp7vRG52Bsn/
jRtG/3e3gO+DngqqIQNztkxw9pl56j3B/IWPSPUqG9xwZvtWgRwQODyl/CV1vqg4BsrHNYROEjHdx+U1XKKP
eDEf7bxvjd3x+13yFyoZBLF9HM8N9eYHn9Dj2P+FyXysc2HBHqjcd7PYjgo31i4jjjt3ECXhzHgjSHYLP5Uga4+
h/X8oeqqZoia4ODmu5x7JiY7Y/AKxIDKRNqzAg+qy2La+RkDJGuz5ZB/hepfgzqFlxtkPqycDheO2la8VGYZOo
GPYf6H+CD3UVc5m/wBOVU9ZUF4PcHiPkum4dmGCPmJ6lsaC0LsGpLA/dE09eEsaUsBgJAmpYFwcwJUSk
z3Lc1gRskaE2uCcpHJvcuTMAnNnVKtyRA8rqSoDCImPKYqg8FOrimacqEiSQJrujlfVf3UqV3QqLbgOqISDvltr
yeVF1x6OUUnV+clRfcDwU1pi62Q/ciclRtWE8qSryj4qNq0dVYavSJLO8CpidybpAU5yD1pMW5KZCL2iIAPNKD
hO+xJ3xlbzICInOU90zncL7HS5en6OkwQo7HGJ1UjEwjo3OACN6Z6FqOJGVKwZCr9xGTH9QOIR0rTkI4pey
YaVjcDhFELAEoZhmGEGOWSAmWZ5LiiEMy1M8zPUt1sIbBGOPTHXVJ2nBT9IzgoHuGQUYObMV2kxs853u
sTV5ixE7YFmfqFC3WoXwlU2ezTUJHoxVxSF7k7mhcUrk5ly5FyyZ8cU2SOSp701yOWjlyY3TuzwEO1AwC
nuQoerH+nCORBnMizUtZHS22aRx6A4HuV4ceJl9dPfaqQ87O3yV62+LV5ittgqJHfeWERT+SvC3UH1DpJnzZ3
SOLIZOkUgksfpV9m1TiOmhmT1l6mmccNjGf81luprk6orny4JJOyFg9h3TXo6kFNYJ53fdMeD7AIQuVYY5ZJ
x6n/bGPbtX8pw2H1bkD4fKJMmU0SZ4LeYmTboeNrLpDG9znSZB8lvz3evVywiRIHGHNDeBgLy98HKRjQ7zH
Hc8u9RHcr1Lt7dob8AYXnfX7f+Z2+wnofQaf+XDY7mSFCctA44TgduCeeibIhmMbT+CjflDgAcjplUiXtQJzBlac
D0+5XHa0dFuDGCOpwlUbGkgndj24XELWKAZHYllwRg9E7RxtbjD+TytYysOJa3thPsUGW52chdgSjAm8TS
ckk4OE9xQDIwSQtiYHBgBO4jrhPkclaA0dAjUUXdY4nBsIOOEuYyRjhkZBW7KdzSC13+adWNPscUuqxe9g+s
0YzuSITWtx3XYAAFdwDjLEBYAzRKxhxyFjozwUtw8Hhwwey2LAey72yLdzGI8JLu6QyQNHUomLEkkjBYeAfh
RIJ2thgFUtAJI5yoxvL27XNa05xnopqqY3Oac4Ciy60+Q9BOI1oYGUv1Yx7mvBdjH3YC8ztxeFI03Do1rl6n6tpg
+J+AeF5v+JFLHHSzZDO/OefhWTPd4uA9+IN1RM6Vz7DMq/RTj+s0jnPwwsZn/AM3Ctr4cXgwVM9OSHGU
Pe0/tadjj/lqgmMcrWXKgcRtD8tP88hSXZrw+06qjne47BMxs3yyQbXf+y9H1dAu07J7px9RPI9Jf4OpV/Tfz9D
PSe/UpZS09U3OiauGU45xG84J/wDKVTTX1hnjus9UwO2G6zNbhv8AwzK3eAT7E9FeJojmtFRS125rGujoyO8
UgywqsuoJv6vZ9QU8sJZV/TRVDQ05Pm0L9m8fBaeVQumXPXePYHB+jT0PqVkvpz9Mj6rK+6N1FG501vrT/
AG3sMkU5HlbnD2Se4Cj/AF9Y6y13Cc+W18bgHBxAcCHdHNPsUpraZramCZrNkdSx8kbv2TtG17f+VxRpa6/
+uafFsqPTUuXfTvd1LD1j/8AhX4/w3Fq/CeHHT8559jeppp4hyjf1xKsNeHNG7IHtFVw1ljZK8MnaAxsh6Pb0
DXH39ihitpJaSpnhlBDg308dR2KY4pcHa7onhwyilcMGPHbul7SRyU1XLDIHDDiCCKD2DeisTRVUMcV57loxD
UAct/wv92r5WWuaOEOcztYRyPyD7LQPozWR6j7zja6ySmqWSRkclq9H/CS6NmukDw72yvLt7Sw9VY/wA
KNTy267MjdnIOCluVp31EjuIXpbCrbc5Df0M/Q3bZz5DAfZP+9Qto7U1Bdrfe+KQH0jLVJ5mwqkAZO5wTH
YyJI+VNzqhJHzLeDit0WPkSF0iROmSYy5KwibDR1a5dS5NbZF9dKhmhSGdZJAEyTS9V2lkQ5PKopJmNNdJkF
RtXOzIGVXKMFAFBLwVOog7mR3X91E1yI5UnXCTqohuUvVNqF5iu1pGlby4qPq1HVWcuKcazoU/rHaJXMC
JANxSfulcg5KSGeIHwQktu1qVNpyV3ggyQiGODhRM2JoDJXTUmXokFIALLT8omZTSu54xpQRkp6dFVLT8h
fW023siSih5CT2vG1ax5paXgIkipSci1HT4aETQ0wJSV7DmE7elyinwzomSaEB3RSWacbumVMHrXVb8wC5
OICzxHBwEB3CLIKI+aEbeijq4RclMq3iS5ZGXIFYiTyD7LEZ4kX7RPOs5WpK5ZWEqqT2czi8pveUpeU3PKybnJzl
xc9c3OSR7uF2BOZjpE3yOXJz+VrvGFrEiMSyFMSrckIPMibJSFoiDPKG+N9rq6+OBraS1p3OXkL4iyxm+imhxhj
GsOPde7viZ5QslUe+w5PsF+fm5bq3Ur+2+Y4HwrZ0jkMT2QRHqjgBR3dhJic8UlpHPxn0QNdgdYVElxBEKymf
sa1vqwOSe+PITjWQxvqXh/pZA1m4+wafAskrKionuNQD5TXltPCP1HsP8AqUbpeSzf7mNNUMKq/b7CXN8
FYJA4yPAaAfrGOjWr0gtrg/aMqg3gm8y0Mj5MbnnoOgV/rPC3Ib7BeWdabOusnqvRUxoaj8ofwgAM9OSn
BwB6nH8LpSU8krz7AlvpLJITu/nKrQQnsJZfEre5jJHDkNJaCccFOMVKeOgP4RXHaizlxSltFtDdxI6AqUVH1Ei
OoT0MZI4ffITw2J2B6v5W5iilLegXTLegP5C7CzgvmlxiBjtnqE8MGcZGcPJYQDkcZCUtmD28OwiFg7gmEiHA
wf80sY84zwmI54cEvY/wC4kooEQBkj0xzS0cFbuJx7JsEgyQMru2YAqbMDKHMWta0t5XcYwUz+a4kgDlx1W
4nHIW8iclGjsCcDnlfCckDJWlsHnK67ylqREs7CRwECXKjcsjuFJmAm6qga5hUfleRJ6rtrCUl1faDJTyEx9iOF5f
8AinaX09uqSA7DW917f320NnjcMdl5oeOGmnR2C4yfafSAun2NXq6wTwWEc6grbo7AO+wzx5r/wDgU8o
lyC13+Rwi27OjC3sPNhwPyPUE11dG+SwQzNZw2R8bj8kZCRmZ8tsh4y6ECQD4BwV7SvOPkxE8JtG3IPqq
mej3hVqQXTT8cUr8ywxCl/lb9qZtYmBQVm6aJ3kSSBzi07SCW7SfwQqt+GGoHW7Ugia/Dahvo/5xyF6CXejtl
9tflytLo6hhx2dG74PuCvPdfUNL1EtjyPzx856T02/wDNdNayC6cc+4lLrZpiovFFdaeCdvom0MjauBhaT5sbxhw
aoVskxfVCnGWzNeRAC4IcOdpVqbbTy2PVdDum9UUjoJmn0kseMtz2IUCeJNm/pV+NfSxmKMzB5A6A5yCP
wVZ9LeXtZCcrYoZD85V9Xp9ta2AYatirj3EFtV04qqWksj6b2s/uNzkDPJwoQcAWgqcZbk2qkqZWbdpduLQO
NjuuPgKM7xaZKSdzo+YzzkdAn1DEKEP2leuALbwfrGGF7h6QeCpCst4jiJpKof9M8/wAsPu1RqyUBLpWkNa4

lsgMulDlIcEQ4utglpnF7HiWB/Mcreh+D7FDdvmlpqpr2HBBS+0Xyqo37d+WO4cx3LXD5CKKiWlq3udFQUxf
+wZY/8jBw5RgsOGGR7zlwv6ePIPQnwjr611LFVUc5dtP92LPIxo7QXGSopmucOclxK8L9RV1nu8BjJDHuW9i
9gbbdopKOOVuaHBVXVFLjjsYV4m6sZHM03VZ90ldVfKd5a5pOQ5JjWZPVDYgpeGRqMrZsqEWVCWNqPl
RNolaFYIXx0yH21C1dUIMgwxWEdJUO1Ei2fUfKHqmoHusA5nZaNFbPjuo+rKjryn2tqByo2rqjryj60gjb+41
A55USXKbqi241XXIRXX1GSeU7priqx41TuyEF1rkTPkBaUF1jzkpoi8xW5jA/qtYxysJXaHqijBc8x8p2okhYmOn
KJYMIoyE1x5pWAORpBCDhB1OfUpEom5aEn1BjagTg6BP1BDyuxgCfaODokbvWY3VYWUkPoCf449q0oou
An4Q8JQzcmEkDEbcplqWjcid8IAQ5UNJXdxZ5i66CVWRgoBqmZcUf1cZQxPF6SmCENENogX5IWIJ32LETugW
J+hUFFSVxBX0II56+YneU2vKXPKb3roKZqISUlKSwhcXtUsyMj0m3YS+UJpkXEiM5SyoaqaoDKXVD+EC18pAK
5MX2GQr4mVT5bHUxM5L2kALxgkshpdRsdKfUZCvZPUYM0bwV5TaxAGr2tDuGqxdNchbF91MWcNanrhh
GjVeJ4bPNscPNqZS0++1qgS7TNjMNM08U7A0f855cVO+qHxiSiHDgyPeR+VV+5PH1Dvd2Sf5T7QqPDH7wr
Xk7yB8hPSTwEeH2EEf+M7K9H7O12WEDleZn+zxUodaJGdmTr1Jt89PTwRPeergMD/4C8m6ymOpX/5p6z0
mwDpem/yCSra2Cnu9xBxychE0d9ZvcNzGNaMnPXAUDXS+zOqmRNJbCTjcm/6hAl+p6qN+6N0Jg4cEulz+U
lGcgCMWUvywlpjq2ke94jZhp5JXKHUFLMCd7AM4zuHP4Xm5qGpv5aGN89jWnOH5P8A/Q7KOJb7fKONz
YA+N2Q5zSS5rvkFMk0tjj4xBmZU/TPWluoLY94AqGFOMVbc9x2ubj4K8mtP6xv0M53bpGu5AJ4UjUniLc4L
gMF2xzwHMIIGHLh9LapPY4hFbKQO4npJ5jQeCvolAGFW6z6sc9zWveS1/wBpJUw64iqkeGv+zHCXEMMA
vHeSF5xAABCXxyMJOX5OEHeQ8gAnhLYp3YJ/jC7VphTihyo6jdzhdBOdvJCG2SHYSR/11Xx00mRgqYPBGqG
Y/fUjcBu5X0VPPYhapqQwFwxMBDVTfGU7XZPbgn39z8BYGkZrBE1tYwN5KUR1THHg8KIN48WbdQbmGb
dj/Qdj/PZQ5J4/vNWWRTowcBgG6QlGpXewyEJgr1ovqJ6j/VNH6gvhrYtxGcrzFHjpcnu8uOLbu5zMR0/AtS
eJlfPJunrlgHfaxrg1dNXeP0wTwi9xPQ2sfSuBBI5VQPFW1wSWC6Mkw5r6aTaRzyBIJKHWZq5vLa6V7243Db
vx/ITTrGWqqrBU7ZiCGHcw8ZBQXm3rkYIIMLrynZsgieEFvnDbdcoCT97jt7O2HchOj2sl/Y17mf+WQcKQYaBr
Lnfqd2Q6GYAP7c7XKM3ANqjE12A5pZx03NOQvbKCrB/sf3E8j1Ssvh59Ny/sZ8he6mfTzNJaYpANw9s8H+F
6TaS1DFcLstH/FUXNeAeGVMY9Q/wDOOQvNpnqqpYCCbjnkp8hWR8G7myWsqLNPgCqw6FxoPLqYvsIK
XdZ0y26Ut6pz9od0PVmNv7PSzj7jtJ61/Ssiko5g3EzgTvl4fsOcOVXb9ebIC5j3ZYx7B5kTx5kTmnuAeoV+L/b4
7lpwSsBZLC8lwPB5vpkaVQXUj5qajnhGH+RKTslyMHu1lej2BlVCMIDiWbrCEB2BwGXMjD6mnmZMXRMEL2
PyGgks2O4cAPbunstrJcAWVwGf+E8dCPb+EPxm21tlyZr/llicGPY/mNwd33Dovgp6s7opmFwJ4maQ/a/s717
FXcqrflieeh2U+4MYq+x1kbnEQnHuOiZoWyPiLSDlqlKyXqWnnFLW/28EtDpOaf8Lif9Cim7UUDZPMpngNc3
DoiG7ue4eOq2LWDbWHPofQzGpBryrcex7iV9YBsO7sUsgkw5uScBLbpQ1EErd0bmh3TK1pIN7TncjARiAOD
69/aWF0BiW4xsl3NyDlep9qqRDb4ml3ZeTXhvXfT3cB3Yr0rpKzfTtI6EckT68YuB+UmrJOn+8kw17SthVj3Uc/
VEFdRWY7pfA2bEkxtZ8pS2t+VGTK75XdtD8rhIkXiSU2VnyvprPIRyyvHutzXD3Q5SEraPeHD6vjqhupq+vKYX1
/yh6prevK6Wudm2dayrJB5UdV9T15S6qq+Dyo9r6rrymINcEd4wXGp68qMqyp5TzX1XJ5Uc1dRyeVYKq+BF
rvHbz8tPKGKmxLkn+o9KbJJsFhIEWnUvSiF6anP4X2GVbKyEnmGkEmESxSjCA4ZURU8uUJYkJraG9I/LgpUt3
lCiWg5IUvWsANakOqjmhoW7BwiCljCbG4OETUjBwqzYY7QwvvolxgllAAmSkGMliJ9CWTJielxToYkAJKI6koY
lKlrEW2mMNU0IUqsYKJaqtGoHqpcJgiGjBTEaxNn1CxFeG0AZp0KAr6VzC2SoAT16JnpE4Je5cC1dTli2rIIeUw
uEgWTlwTBMco6ohmCH5lyZE0GKI3BUdXCTqj+sOAVFV0lwCsAyYrt7SjL/UhIPMFzPzXkhrSqfDfzpnccr1Pvkg
cx7T3Xln4t0rmXEfo+5ysXTQPFIPqInDkFjBe91fmwQSZ+6ANKgOv5rHj2Up15eyhp2O6+nH8KOJo3y18z8dX
Kz6dQE47QnVMRYd3cAT0V/2aqQvoKr/wDcvSyqbBvjAyzYMZA6EqkP+zFaP/o1TJ2M6vRU08UldJKHhaerS
vH+tMDr9R/mnrVrvJ0OIB9Ez+5jdBTzPPLsjuD1ypBpaWJ8QEvrg3ac9/yheHYxp2tAA9znCdmVT8NHQFVz
Mt/hAiPjLRbAMEsw+2eeE0VWltO1PDqWMedOOi7mpja/1O5Sse80MWd8zGn5cFirsOxMEakQsrPDqySM
ljAbkc+xUc1/hrKly3ZeP8AUBSqdQ0j3YZUMI/5gnKC9RAhpJRQtuHrIvBikCwaeuFHM57t5acEAdiP+qnfT8j
43Ek4JZz2JIRQ2eGVnMYIK7eRCXtcB0Qtjlp2DgYhHDO5/GE5R72vPQjHCH4DszyESw7pDwFAJPKfaLo2kkD
kDC+yyMaDz2TkyhmcM8hNifSyCMk8uRYBxAmdS3xSOBxXlgaN3Z2VW2+XuqrKQMi3B0jn7sfpyCpkvMbH
Pa4uO4EqPY4KKDDWt3H3UIRA5xmd4JGU6q0/cahkrvIkeSSRnqXe5+AmizeF15lkdJldm88kDBKu5C6LaNr
GhO5nY0DuR0ATQa61QQoAzBjQrEZkA2/wAI4m0pZ5wMjj65Tyf4RtQ+D9i9BmL5dq16mq8HBIB7IEcVZCS
QXJe+svJPnMIFC9oitml7fSYZDE1gAwA3hc7zp/NNKGO4xkkjy9gETGvp8feEyXS7RNp3f3eyHVsnM2ajPFT
UtmFt8WjBlwiGvjK9sl4x3+VAGo7CYbltiH3sLm+4fF1Vx/HOINntt0DyXwVI3H4JyoQ1XJG+IFbEdwimjqGEjq
yUYeF6job38PTP8AZsP1WUPqOkTfq0l+FxYPow5lXrgMSRysP3t3D4IT/aLIJSXmmqmZGS1/HHI6purGNc6qj
Y37HGVg68fqH+Sa7f63Oj5LmnfH/1H8hWzhqyD7Y+xlBwy28dwePqJ7B2Wvir7PHWGRp+sjBkezp5zeA//A
M4GCPdUf8Src2nr5g+p725Y5h2k92nClLwJvDK2Ku0692Xlr5aZrjrd+5idPFC2MqLYA8bailOx5/cx5y138Feea

ZDpOqNWe2ePoexno19g1fSw477f6juQazmjKqjG58jRKC17cDD/wAfKR1bDb53hj5CAcEEABwPRNk1NLT1
sjH/ANssf3RILWw1UDivLlnP4imOC4O7s/HsvR/XPoZ5ocdsdosguU9VQgYczADC98glx+k7Xp6o7zW0tM6G
SaOytONomADgfggKGo3Fk7g/kOBa/KTP4zG8dDwVw1an0EIUkHgntJcv0gmowfpaZjhg74pS/HxjKjSCZzJmn
2K2pJg6Exu6joVuyME4PB7FSouFxBbG8xzJQtMoiusLwcCQBejGmakyWuLLslq81KD1VNMzOcYwVfXR1QY
6fyiewSLqK/wwfaS08hvYmS7JMm91ThbOc0hD078ZSJLIFchEd/rCO6wXDHdBU1RgJhfXEHqmKqDErviSyLl8
r4bl8qlchD7rV9wd7qXwpwLjJvfc+vKaZq/jqo0Nefdc3VvypVpkothVU1vB9SA62sHPK5VFWUE1tSfdMKqxm
aLmIK6p68oCqp+eqd6mUklBdU9OKwll5M6uqOEm83JTQZCSurXlJAg5zHQyLmyTBSMvK478Faxl8EwqilRN
SydFHkcyKqObooHTiYrYMmK3OHCliglAAUF0E5BCkejg8YVc1NcdUWDEmKnlBwjijcFdtJVdI5UiUVW0Y5VZ
uqjuu0Ykq05wAlz5gAhGksbt6r4+ryeqWFDmTG0R0llzICITNgFK5KgBp5QbV1Q5RVVcAtsER1M/XIadbMcnl
Laqs4PKBa2sGDyndVRiexhO31IWIW+tCxHeDA8z9NrSumEljKXgKoT2OCCFyISshcCFkyJSEmkCWFJZOi3MjF
Mh+bun+ZD054K1IngBXHAKhm7yYDIMNwPpKhC8vwHLtBFF/YyC7zLw5Uw11ZH3OpiDW55Vrb5UYLwop
gaySpjLhwHrUWrbYdwikqceOoPYkZIRtUacqI7fEWt9cY6KNrXamzQTzObywZP5V89b2kt8t7W8OVeKqyzU
dDMAzHny8KXp3UC9RQnndxPUev9JRXq1KDysg3D544Mv1/s00gg0PPK8AZlkKsLJVEOcox8HKT6Dw6hDhg
u3IEc5k5cXnAPQDsvPuoPv1lx93aWbpNZXS0j1CLHupudPS075ZphGxoJLiq5Vmv9eJDb5WUVCx2DWDtDq
B3aCk2tbrTvxDLE8x9os4L8KhWtr1qy7EQxUr/AKGJ4lp4AQ0tB5x7ozp2g8Y5OB827CMdbrPar4Vmb0Cy0N
bqqjqak00VTdb1OGlz/LeWs/A2qT7XoXWFFSiZmkooxs3EzLz/V+nr1CnDwnn0Y+3UDKWmZTYGJm+PYGuB
A6HKu/Tx0Zhc2N4JTRXRW2rWRj3P+gIR1Or1ozl9vyAnjDryzXfT9+pLbTOtc1dJcXxeY+HYD2zlRpD4q3yx17
qWtEtNNGfWwv8+MD+eUeeOejbvPrrU1RLVs3yMbJBGtTc1rRxsXm5HBdZLhurKhzvLzuc52SvRq+m6O3TV
hqt5Kjke5nmo691RNVcV1QARj5G9cT2a0r4t0VwbEx7gx7ujgdzH/gq0Vt1DFLgh4Pwvzu2y53e0GKqgDjDP
M5gj7SBvXH4916j6Uv1eyipZH72h7W4J9ivOur9IrOzjYeDnE9b6R1T87UdyYdcZnodFcWukBaRhSxZWecWv
b0VVNPVvnuua3POeiuNpylBrPwqMo/igS03Lt05Ih8IHYDdvGEBX7LIHDpwpVcBtUPalc4ui9wmdxCrK7pVL2S
oeoa8RyHHcnoo2be42ynDxkKR9SWmqkZI4c5ySqqXay3SDzZZQY4wN248DCmpRCnJjncC2JKFz1zQ0Ebnv
mHpGSogd4k6iurZqXrQwk//AHE/pafwFUW5anpKi6CONwmd5pYzefQ0ju4d1fwg0RYb5Q/UV721UhPd/
pZ/AT78nXTUGsTJPvF9uqADbOw7kSG5dS3Fz2+dqiocXZ/+2gwOOuE0DxF01TVLqepvd5MwAlvuG0H9XAX
sFbtJWOjpxHDbqZ5A9OY25APXleLPi3o7U9s1lJeoljDvOrXNw0jJh6Yx7J3oNLo9RuDDbgDsF/2ID6h13W0bd
iqdzYGSZL9v1vpG57Y6LV9dZtkhuJXh4JPwRIDOo9Z62sMzmyvhudN2e0bJNvxjgrzbt9HfK2GaaYOY2EGQSH
0nPSUrW7W1yoaOjp615lbM3dyN2BnHPtIGaroa14dAHU/pZQG/cQjp34jNzPXaSjL+pTuX+smLUI+pdTWct
EEp37cmN3BY4diFFdnqRWaZbHIM7GOikHwe/8ACb57naGVQla2SOR3Bc0ZHPb5TFaaplNU18Wd0byHt/5
XqOqjZSygHAYMuf2MN1WpD6mtyVJZGrbHY+okd1JdTVMLiOWEsPzjj/UlaL3QVLJGcYdx/BUnXmhMj59oz
6A9nzhRdJzkdnaOCTFlh1IA1VbV2/KT7ZaoWPV1ouMDnMEgngOcAkcpYSr0+MELJLNQ3aAZgqmhk2PaQZ
Xm9G8VmlJAcie21DHjnrFN6Xf5EK/fh/eotWeD9XbJ35qbdml+5Z1jKqnUqmW3T39/DfY/wDIPYyz9OvUpd
QOBYniJ9R3E8/NR28ysFZHszPzk/Dmj0n+Qo5bLIGObn5/kKyFVbZaUGWaMmjqs09SRz5bhwhkfbVebIQz0
NdNBKPE/Bx0I7EfBctWnsDLjPaVjUVEOTjvFdRE2SmbVx4IJZm55a7934cmmqla2TPq4a7/oUrjJIITtaH
Du09HMPUFb1lPE2TMTHiN3LA/qi/lBeM5jTtkNkGU8MJ3bSmt7dpaU+YBfG/HB6rBILRk5iINsrZ4sEg7gvQH
S0M0VLSTO6OGCVRcnbi4wg8jcF6X6fpmS6XgHcDgqvdx2VV/M4MN6fXuNw9gMR6e8gJiqH8Jxy7yyx3Dg
mCocqjU5zNamoYg9Uv6oSnl5RNOOqFZ28qw0vKhevMQmcrn57lyc1JXEhMwwgEWmdJzUJASUme4qYGS
LmLjpihaqflLnyjneMIhGxDAsY53oVqin+c8pgmCYo05NcZ+63DI8LVqiN0gKzpkqrdoXfyyVrdlu04MKI6Nx
ymIkBT5Sx7StFhiDuQTDyjqByjamm+UA0/GERwvxhKLoTW2JKNFNyOUeU1QB3ULU9TtrPBWnl5SS2vMZ
pZJoJrsd0qbVZUZQVLj3T42YkJaapKbDCeorgnIAVbXjn1LnWTPweVHNbUuGUwopEW2WHMX1VdweVht
bcDyNyTVNaeeUEVU5yeVYKqRF72GPP1x/csQX55Wlwl9pDI5+tuFydmnhDUMieWP4Xl89yi4rgVm5fCVky
ckkkSpJJeiyZB6oPVC070T1J4KDKo8FZB3MDrhLwVB17IBDIk0kIDIX2+1WA7IE1rEWpfCyC9QyDLsFRXTTO3
HB7hO+pLmGF2So7ttaJC8p0UlpJlZqcNqIHlzloJ6F1tkZAzgj/RRHrq1ft0dL6cYKlvwyq/qz8T+cPKzxMpAym
HHDQsQHSzVdSKemZ9F2XDU9AqPchRn7Sa/D2D/APzqglB9UJP+ZW88LnMwHYIRboajNB2wD//AJwuEtIC
9J9S38d/qYd09R4C/wCUf2ka12IDXV75Jsta7qAeyl4NJU0NKlmwRuZvAA25lapQfTMxy/CQeeIndcDPUroa
mwDv2jIVAtAlJh8t4dDFsaH9CMY/IPcVxqKVjg2Z4eHYJa4jv1T5LqOghG1+93w1pKC6u/xzFzYKN4Du+3CJTU
3n0MIOnqPDICI032LTN0rRPcrRDVzBga2SV7iQPbhBdTR6PEbY4NPUTf08N6pymhuNTLgQgZP6uEpZZxDy/
MrvYcAJou1JUA3Nj23QT/h+gVt4oQN7gcwOk0XpEsa/wDp7Wk85A6FJY4+aeCPGxj84aOgHZSE+lnnaG+X
t7Fp/wCq70VoxlHPaBjgHCD1F+B5nLEe5hWn0oLEhQBjR0oB9Y13lxb2wrv2BpLWuz/CqLYoGQTY7eo/KtN

pyU4aCc7R1VfQ5uBk+tXFDAegkyj/hIRPe2CRxClFjuEB3Nu55ICY6keQsqalgWmQ7VUAcWkt7dFH2ql6Cop
TBV0+Yv1YHCnt0DXDBGB3QneLJBUQSNJzkcEKCl+QCYc6GNmeftbojw1gmLxZY3lx3Enjk9+E72mp0nazsp6
F8DWngxSEKRL1pR0bJCC/B7KHq3TdQzedr8HkOHITjdaw5uYiMqU02OVlrrXrpghBhrZux9ZD/SOyCtSaesGo
619ZNXVktPgGyMgDSHEdCQ5VrpXXaifkRh7M8EckUrXqmlYS2ppiWu6nJGV0mr1WnbKmA6rovTtQvmqB
nKLwN8MpXTSF1dcMSAGF8wEzPudiFdS+H1oYwxRW+nazbsYY2ANaAFONFqDTcMBEUGG7ySA48lyR3O+
U0sbhDShvC1b1fW3Eb7GOIJpukaLSFhVSFB78CeeV28LIld8jHv57BV6v+nZbZW0077JN8RP/NyF6b1UM9U
4+a0sz2PHCrz4IaVfNp2rlYzL4MSsHU+jlNtH1K02otj8E4/eBa3QVGlyicr5h9uZT2SAMbG97T/ZcBJ/+s8H/JQd
d6Z1HXTQED+1I4A+7XchWoraf/sMcmwEVEG4fh4VeNVDfPSzbgrLTtGfmP0q6aG0s8onVaQtYPOP2MJbPT
yOooJmtBjq2S0M3/7AMsz/AMwUh+EGpJLRep6Wb0Q18Rppt3Z4+w/wU0+EVNFe5brp6SQMfXUxmpHft
qqb1Mx+Qolnrr3O+QbJvMMnH72nn/AFCmdVtbUUt7D9j2MWBjWmmvXPBI+44lIn9Sslp46xkTtpdmVjv
8Q6gj2KiClktOpGiKro3R1sTfTjTKn8xg7Fp6uCmWvq47ITU87TjzIQSPnCha32yWm1C2Sj5bgh7T+UPQQKnz
w6jIMotrd9RUF5G5GbBEmCzeFdBJAydleSGgjEsRacFRhrLtsNvmjhZl147OAxgr000p5dTR4cBuLcSD591XbXl
sDi/eG9AkGm6le2sxY5xLhf0nTrpGFdYBIB66lswCMFOFGGSQFp6jkl7eH75i0jDm8H5XO3My4K+r8APynlV2
PFYDtulkiWK2/vvKj/aQvQjTULorQWkcdITXSNrnNZCQ04Lgr6xUxipGNAX6QqN1zU8omY46fVhXbEYJ2h8A
e37mdR7hCE53DKN3sLOnfqhGth8s8dDyFX6GyZxqkZ/eCUp5TJMwFO02cpokJvkqzXRqBzGqRjU2vATq8
OPZN5hIP6UzWKil2uCSuaCE7Gml/ak7qaX2Uu4e8lrWC8zAENzEZwjOop5dp4QdNTTbjwjK2EOURnlamKRp
yiOSKXuE0yRuHZGo0kKxjc3lfWxZS8M5ThHECpt0iKzjU2EpeyFPDKfKXNp1rxIE9camxABbtGCnTyly8pYggb
LHSnyQiGPomKmaeESwsJIQthnSDidI92U9wlwK0ZDnHCdmQH2S1yIWgMfaV/ARLHINqGoYyMJEZCAG8ZM
JwZwrZuCo0rZc5RfVvJQHnEXJjSmBarBzAOoByUI1LX5KkmogGEKzQA9k6QwPw4GbHLESeQFik3TvbP1Ow
Sp/jl4QFTzIkiXlk9kEJWvXXcmhkiWtcsnUVJHKUpBSKY8LjHak8FBdW7AKLqk9UEVruCtiB2GRfd5DhyrHqS
U4erGXit0uVYtSSDD0wpHDOW1jcSoWqpXbncoUszyl3H4TzqmYbnJlsBa+B3yCrDYMaavjRt/zglivCq5eXcx
GT95U3eJcZkt7+MqpmjZxTX6hOeryCrm6vp3VFC7uHMxnnUVFfUaXhrPoD8Ok39K1NZPwk4kmeHsxl8Prc
7PQbP8iwxN2OcVG3hkXM00IT1jqpApdjy1xPtgKvavAvb6mWbp+fAUGAVRC93q5PseMy5D3AjJ/wAypb
moiYgMZz7pqltDnEEgN2j4JKHWWJMSJ8sa15f92f5CQsLnucGAjGSSApX/AKax+dzB+duMrm61xx4bkbBiBgd
1OGAEICZPaRaykqC7kZL+SOvH5TpTUNOwk7vyUfeSGy48ogDqVo+FhDixgLRn45XDxtiTeAPUQMFLF5uGja
epKfaOg8x2AOA7hcCwwqfM5GQAR7KQ6lBo6JexyZtztGBFMFFEHNAyCBhTbPsh02Awt2qPKWjMhbwzwpYs
sQJOERUPOsValh+XcdziSPloGN3KF6gEkg9EUbWEcjlNc7PUU2uBKyn0kBoFyWEZITc9rXBwPU8hFczR3CYJGI
j8gdCIXAaNIlfMQRq7a2dmCzBUfVunXM4aMg9ipufLzQQEILfMjdtz/IR6WHEwAjtK2zaYje7+5EAE3yaUpH4J
ha7PQFvKs0ymJj9cYPHZaihjl4ZyOi73kyfxCJWhujra3BFOB/C6nTtLEHFrMEKxH9Nc5vq+5cJbaw/oHHsEOdx
m/F57ys01p3EZGAD7IGu1hhmglt3Nc1zSD7FWtq7eQCABymqNlHr7JOpynecLjcyMjXDD8jpbWRaa6k5
8y1zuHz5ZOQqk6miLInsxjyptzfw/qvR/VNjDb/FSopX+mK6UfA6AvVfUD6aWRuOWudG78tK9L6XfutX/rA
YffvPOesUAaZz/8ATJU/T0kZ6KvjrHqu0XAHApquN5/5c4Knfxt03S2vU1XUU+Ayads8eO8VSN7T/BVU/LcAS
RjAvsPEC4tvHh9pW5b90jKb6GX/AJoDkZvivUrrNM4PxrZv7iU7TMG0WprP6cWD+Xnbw4abnJDTuaTsGc/
CkOWzxU9weCzmN5A/CivwZuBg1FCHexCthqijDLrvaOHpDrWZNYydgVzLh0Vvft1ueSGwYeaXIDI3R8Oa3/
MeyctV2VlXbpbmt6NjLfnmlKeTc0BTBqCl8mx1EnR21xVltYrqlwfWehWL/BZvPEjUFBMLtOWjLd3UJXYKO
SW5UsTWZLiFMEsIr7hUARNOXkBWGM/nCsb4aeF5/qMNZLFjByAV6ldrkp053cELPATp3svBBzuf8AbmSPp
XS0dM7a+PDsNIUx1FB6cYRZc7c2krKSZg4+1ye5qdrmggLYa+x7XDk95aztRSBIJqqLg8lRqqRzm4IU+VFCD2Q
9JbW+yPo4xEGo5zK9SWl7j0WNsRPZT5/Sx+1LI7YPZOBewErz6cEyu7rJ/hWosv8AhVzkZmkdFx/ow/auxqTB
vygz2lczZf8ACK77H/hVk/6OP2rk+zjH2rPzJhaaUSrE9jOptQIPYju+1W8mso/ah+ayj9qJTVeesJGmEqLNY+vp
QvU2YjPpVwZ7lOfShKrsY/amCas+82aBkmG0u/avrLY4HorGvsX+FIH2X/Cjxq4I9Eg5tE5vZKfp8DkKXH2g56J
tltL8cNUy3AxbZXgyK305XHyCpEdapv2IJTb5O7EULB7xayQbp6c+yLKamdxwnCloHZ+1GIPQgY4QdlsrqlLFS
DhOzKYYRE2kAC6eRhLTZDgkYPKwE2yMIRTJHGFmzwpEadmuDE0Zl6JmdCeUbmIuHRJ20nwmdbjEFevmRr
UU+UOS0jVMUIAD2TfJbuOILfWEGNUib6P4WKTV6aPZYuvGnHhz3ipJsglqgkUfUb+il4Hrz2eqQuienRjkPQu
TxGVqSCOoKRzFKGpFMVkwwbqT1QHxU4KN6IAVw+0rod4DZ2kLXyXAcqralqeHqzd+Bw5VD1S4gPTbTjKSp
648GVN1LUEvfym7TNSdxbnmzUchMj0x2Gp2VjvZ7Ezpz9JVnk+NUp+cmmmkd18Eg42ytK9L6OlbC7FC4
cksXmnXMzTPc3qBkK6/hVqplVZYoi71NGMLznrNtvp0sUcoZ7n+FtStOttpJ4tXl+ZEnfrtskplLVQngmrJH4w
pliYAcxd0Wlv8p9GXDgl2Sl8cYbk55KpjszYJ9Z6ZvtDuB7xGGPaHoc8kF3Ax0+F0khD3Ny7A+Rx/KWgEAYHb
qk9Q+TbhrSc9cHIRCPK+SI2y0sYDw45yQR3x8hNraR7Cchfk5GRhEOGHYA3BSZ20vftcct4WYrD1YgGMzhlw

by3H4GVyNCTFI4IJREyHfLghca2byYsYyeyhbJkb2gYA7wChgjidtxnCPrbTSPDARwOqaqK0vETpX8I3KkqlpDF
AzA5AyVKtcW22xfDCxgAcDhGtsDNuAe6A5ZuQ1GtqcjdA5uDnqiagPEgGoDDT5kixElo4yQtJoQeT1XKJwBX
eaQ44TY42yoebxOIPVUeGE90Jkv7o1mjc9jj3CYplmFgyIj15MbUvxG1j4yw9MhcHHDT2yg+8TOpTuY7B9kx
01/8AOWEKXKEAiNqqGYZEK1rnjb8pVHyehQPT3EPIHq6J/ZLncWynJHA7IhLAROrKcVbHAzGeQu/IxuBAC40
8npaHde6d2luOESORFNvB7QZqqVu3OOVDN+t7vU5rCSOWOMqxUrQ5uOiBLnThwcNvAUFGkuntIM8rPH6
2GO32O7x8PpK5kb/fbJ0VPPFi2Bs7p9oDZvKl4/xjBXpd41WGWq0beYgTy7yTNF8ui9S83tS17Latomy17mBrZ
aUteezZlzhWjpnj7aD/ACWFT9G5EV9UUVCLx/PWGH/bwZTm40wNspph92+Rjv4OQph07S1Nf4aXmnA3iK
RtQxpP2lhw4tSFJjkq9M10bT/ciqtzRj/NWp8J/Dy4iw3emqliC/Y+M9y2VuCrvq9XWIBJYZSwHH3IB0ujsa9QF
OLKSM/MjH+khO22T+kSW+qiyPNgie4excreMt9xuVNFilYcJWfDatNsvEU7TupqNroz39Ksjoeagk0IQPDAXO
jDXe+5ip2r1+9d6+Yg4z9ZetHphQxTsCucfPsYt0pplxQRSvH3sDkyeJ7fJsUjGHGRhTJaayUUFly8p7m/x1CDN
ZWWe4WwkN4SDThn1Su3YGFdS1vh6N19WUgCVF0FoQ1crJXRg5OTwrz2+wR0bWMZHzARD4aaW8m1xu
cxTo+yjcTtTLVWWW2H2nn9W1UBEq/drc+WB7Q35CaYaSfyGhzDIWlksbD+Ij9n7EMqYGMSoX2JIYJKGT9
pTe62yk/YrWHTzD+hfp93Y/2IlePSL2UmVWbbJf2JU22S/sVohp1n7F0GnmftXW4+0i8EytOdski+xdf6TJ+xW
VZYWD9KUIxs/atZadCmVg/pMn7VwdapP2K0hsTP2ri6xMP6FxlPMK5U6W1SfsTHLaX8/2yrhP0/Gf0pA/TkZ
/SugzTvZKZS2d5/Qh6oscp/Qrvv0zH+1Nsml4z+hSLawnJSUVksE//hptOnKgn7Ffe6Wj/YthpWL9iLW94K1co
e3Sk5P2JWNIPPVivUNMRj9C+HTcf7FONQ8DbT5IHf8AcwHrGkcuhegejFec6eYP0pK6wjP2qQXv7yA6VfaUZ
Giy39C3OIJW9GK8H+77f2r5/u20/oWje/vNDTDEopLpyoH6Sm02Cq9lfGTS7D+hNr9LNH6Fx+YabGm5IEZrD
UgdEPvss4/SVfOo0w0/oQzPpZv7FKmpaYaMSm7LNNt6LoyTY6K2R00B+hbN06P211dSZAaZUR1Im9kkdZ5
/2q3z9ON/YkJO1z9qkGqMhNBISf6PL+xYrZf7t/4Fi3+aMj8EY3FKMYRZAmMODCe4WkJL3H2FPcaZIQnqML
JOI5NKRzJW3okcyyYe0GajugG4nAKPKk8FRxcn8FdqIDZIUvjxh6qJqrkSK09+k4eqh6qm4enOIXzCVLXEbTKba
m9L3kiDtk22sj57ou1JMC9wUb08m2oYc9CrgFzVj5Smrw/OMs19S4PY132uaifS14nslUyRpOww5CjSavhdSw
ODhkBOUdwgkodpPqzIVw6fcm0rweDLumq8Ozcr4ZeVM9I9G3KC42kTRuBDGzEPC1riCqKeAWqHPqpbZK/
7oy6P+FeWYnOT3yvL9dpT7Wr9jx9DPc+k64aum3PLDzflibmTc1aSSnc3IPPHAXBuSwwafLuHZZkFJpd0Y
ATjI7DsYx7FaRNOXhx6LpKHNYMgcnuhOuuccDCC7n8rRhPieXiFwmjY/lcCharq2Szx7ur+ieY7p9S4hpyfhlas
VUVXBKQNoeMlaQcyJVy2TJoFSGvjHlaQnmG5B+4k9eAgKSWUwxOaCQHcrnNXbHBw6Y5CwsZpkXHaHBn
eJMklyo7u0MHHKjCGqjmjBalclUynhc7lz2XQYg5zNIK7FckSelrtHtweM9Cu5r2NcC5/BVbjqVkuEZHGIer9ftd
KGMdu/BRYuciKT09N/fa9Zc3+qRYc3cOU1wVTH7uwBVW6PU8krGx8HHHKICC4BtMHb8cZkHFrkzl9GILx6x
HrGuwBs55UMFIZF/c9TT7KVZKZ9zrmynPIM/1K73CjgEOxpzhd2KCozCq3CYVfQcwt9+IeGnj3R9S3UOJyDn
sq7XeJqqR5liOe+E72LUMM42F+Hjg+6D5X6RoNjr25lraq4b89eUVRz+nrhQnSVQDWguyEaxVzXZ5PTBCKS
zilb6RngSQzMA0c9U1Vhy0pm+pa5g5PBGfPJUj/5UheLhXgyJNb2qOts04cOWtP+RGcVc+oYaLRmqLY9+W
0Vyljaf0B/svfu7Fs1NM0jO5pC8S9d6aDtX11lyN2a5rKlxB4zHlrgPIWLo9i77FY8eVv/ABMWdRRjWhA58y/+Y
jH4fWqSt0lfHuaC+KdmPc5aMr1BstMILXQuihG/yIxxewCqB4JUkc2ntRs2BxdOwt/AjwvQJR1Gx1upg/nawA
ZXHVbibr7P/ccQ9OrArq/yn+hMGomVNT9dln/ABWFhUX6HtMVpZX0tQ4hsc4e3PQByuE63wNY3aAF5te
L+uG2G++TEeJ4yDj3S3R1W3Mal7tz+0P1I9VSi1uApx+8N9U+IVLb9SxUVO8OZK8NJHZwV57dQRXCyU7XDI
7AV4BWGW5VWozWTEvBkyWZ5HBX6ENAVTq230bSwjbe1XDUaJNPXUo5P6jPM31dmpvtZuAV8o9uZLtk
s0dJRRsaOgRA6jHsieGECMBdTEEuWJPJAagYaley1+gHsjPylnkreBNbYF/QN9I9+hHsjPyfhffJcZamtsDPoR7L
79C32RI5IWeSFmBM2wPFEPZb/RD2Rb5K++StYE3tggaleyONE32RI5IXzyQswJm2BRom+y4mhb7I48kLUwB
ZtEzbAM0A9knNub7KQTAPZcjTj2WYE1tke/04eyz+nN9kfmnC1MAXXE52yPXW8eyTOt/wpGNouRpx7LeZ
GUEjR1vH7UINuHspNMPZcFb7LsGRGuRuLcPZdv6ePZSD9KPZffph7LC0wVyPDbx7JO+2NP6VJf0y1NMPZ
c5nXhIRfJaWn9KZ5bk0/pU2upB7JE6jHst5E4NUgI9jb+1cP6K39qnF1EPZcDQj2UgMiNMhA2Rh/SuzsbP2qc
PoB7LX6Aey3unHg/KQZ/QY/2rFOP0DfZysZNeBApsaWMYurWpS1q3GsURNTuwJBGE5sC1OoraOE3zpx7Jr
n6FZNEwXqzvwVF10fw5SVWHgqJrs/hymXvAbZA+oJcB6pzuqovv9Qu4eqXatecSKw6RQWEp+u7GVlv026
odygJ0m1yILu8/UO/KE5CrftwoLXpXPJ9TH2Gvk4bu4RdRVZ45UvtfgooopsEcnaCJJYhTks6PhHd2UmvbE9x
w10xjP/nGF671zy3PK8EtO3CWLulDNH90c8bm/wV7muqBPSQy9Q9jX/wCYyVl/AMTU7ban911/aeyfgjUBq
r0/lcN/5DH+kc45g5vCW7iG5zgdUIQSEP8AUV0rbi2OI54wvOSZ7aB2jTdrv5bXhxwAoTZVTXOulGElgPlk1X
65y1dUYo3Ep60xSTsDgRsG44HcroLhcw1nUACTBbqKgniaPZO08kbhgtymWOaRjQDknHVntVUPDHEH8K
DDZkYcEx2N2dTjYhCqrrp6gMY0hh6uyhWuq5i1zsnAUBvvU7JRguH/AECLRDnNpPLi0sITFGAHpDck+ueAxx
SR7hREy+VzYxkH+SiWjuskOeXn1DsFhVs5wJGGZR2Bjfcmlz2O3blC9RU36hmEzY43DP2OU9vqpX8ZCba1p

dNsLc57oymzHBUESB2dj2jRbfE20texlXRS0z+7mjexT7ZtRf1drfp5SYc8uxhQxHo6jqJWZj6nsrS6d07DQU7I4
m9PhdWjTYBQMD7Z4mjY4XBhrTVbmwNjjaWtXexF5ZLjnundIOzHPVvZNIUyRoOEKQSYELR7YgDc/LfEQQB
x3VXNQ0tXS1TaqidtkB5HZw9irH3Rji9xc7oOnuovqWCQEP289BldlulUtwxPumtcr1LNko8uRvDmuPRTxR3
ZrtnqHKpLqazy0rG1VMP7rOcDuPYot0rjq62jje1xx0c09QR1BUL1YAdfh/tJ9y2cesumLI6mDr3S36xr2IzTyoat
dYRwCSDnO45PKMSXAEDrjoCogcwN1AhPK/c0jPPdUA1bYt2vrRjJ0kzte72b1V54ZfSMnggPUNshE0VTsblkj
SCUbp7TW5PuCP3EXOgfg+4P7HMqd4KtkFi1ARF5RFwnDf8AIHAU8ay1i2D/ALO4k7DjKA/DKBzbXqLDtzX3
OqH45RnadNsbK47cucUfrXQ22E+pgWg8qLxnCz5r3xUfZ7RUzRsJLWnavF7UGrbjqG4Oq6kklRv/DXL0f8AH
oRUmnpWD2wvMCx0zJDUMfxvj4/I6L0r8K6Glt1uzndjPynmn4q11q6sVB8KEzt+ZluvDtsNRdqaN7PQZ2f5E
4JX6K9JW0moaCFrGjhoX5sdFXM0tTuwC6NrSWnuGnkL9GXh5qOC5WWjeHgh8TCw5+Oi31vSvS6+xlF6brE
vfvn6SbdoWwC6DBW+FUZasTjtX3au+F9wtZEycNqzalGFthazMibYs2JVhfclMzlk2LNqV4X3CzMyI9i+bEswv
m1ZmZEexa7Eu2r5tWzMYiHYtdiX7VrtWZmYEQeWFqY04bVrtW5rEbTEtDGE57V82rczEaTEvnIJ12L5sWT
WI0+Uvnlp22L5sCyaxGnyvhfPKTrsC+bFkzEaDEFwMIT4WLMwLrAjAYFyMHwn8xrTy1k52xi8hfPIT95a+eW
tzW2MX06xPvIlyszM2yC2hKGhcWpS1FyOK2BOTEhjCcGrgybM6nommfoU6u6JmnPBWxlyYJ1p4Khu8O4c
pdrjwVDV5dw5EV94HZ2ldNQS+l6phq6TiRW/1C7h6pZq9/EitGiXzCU/XdjKiXV2ahyGXIO9xdmd35TC4q2ER
HQvlE0TtTSYKZ8hKY3YK1EiEWLIZJFFUEEYOcVzXwv1GL1oe2y8eZTAwSflnQleIkExAV7v9nfVzKe61tonmw2s
aHwt7b2KpfiHSeNoGdRzWd329Y//Cms/K9XRGOftBT7ntPrx7g1xxygS/1MjaN7mHnB4KKauUR+o9EOXGI
VNE7HGeSvCW4YT6lp5XMrvpWtmlqK2Z7dzmykBTTHXtp9peQ00HP5QTabUKWiqZtuN73H8gKMr1qWZ8
E0EOWSs4cfz3CP2+LcQo4nB8o5MtLDdYpIwTzJkHguToJWEAHkLzLli8SAHTUN4kjDeWxjussXjqxq3Vbqa9
W9sob6TLC3a7+Wpv8A8HudCarEcjuoPMAGrVHAdHUE8MRkf0nodcZo2Ow6MchRDWB7ZS7ys+wwhEeK
mmK4U8v1kTdjg4Mkd5Z/BBT43XFtreIKujyTwBKxxQa6PUp3rIjxDUQP4i8/ONXmVbZgdpOfjKemTVjly5rHD
8hOjK+q5c1rS32aAlOIP2vpm7e/wDCl2v/ACCEGjdjBzGGO5Vb3NbtCSBghG9spquol3vi00Jjk1DSxOHIQAU
efhlzqutDQ1jmsxjq14TnsmJ0dJxnOJZm1WeTAJ2kHkZ6hStRQvied0mTt6KjMXiDdqZr8zMGO54Q/U+M9
VTMyaukc8d94XIOV7dkgNull72qBPRp8jtx45HOQha41sjS8H/XsvPCT/AGh68se1hgcfjLIft78b9b1bJBSUW4
/vcC0BFV9J1rn4MfU4il1rQZ3hvkoLGXevd/mYXbSCB/Kg2u1U1UA7Zh3u4DCqRSReMmqKkA1zqaEnkxx4C
ka2eFNwEzXV1fUUVLweQ4pg+gooT+LqE3ey8mLke928tDKPdsCWqZc6SptzgeWnjIrhPwght+pqmP7qar9bP
+YdQkiU0W2kNMPTGB09sKbNE0jbnZqOoe0CRjtwOFXLCa6n9VfiN6h/EHulFPZLSPY/ILeCQjqkuLZouoOOQ
VFGs6h8FG1sY9W9vA+U62SR5pmu6ZHOUvVTsvved2EEGSvTSBx+Oq4XpnnUjW9y4JupZyFvVWjTtHfKIQE
tAHO0SKfDq1SwMvInu5N0ld/DIYWmtjYckBRj4d0xkuF9mzlrqx2P4Cnevc2GmkcezSiNUSbfsIbPfg49S25Kf7
TFzG6KmDvucqFW5uw5HBwp78b71/UtZPjDstiULQR4avpT8MaTwuk0gilhn9583/irW+J1O8g9m2j7SRbbIY
Z4qgc8+pX58PPFK6WklhDcvpgendq876GowCwnAd/oVzFQ9fTGF0coHUJ31HRJaqFk3AHBHyMpOi1j1qb9
hYblb2Yf7ie3mhPF6x3+NsyNa2YDIhKszFMyRoLTlfms1RUvtlroq+21DonsIOWleifg8A7QcV6ENBcXBISMAH
s9ecdS6A9SeLTK9R6iXfpn4kzd4GpAU5wrjsZ6LdNVLVMmjDgc5TIIUMggz0kEEZE6LdcsrbK5m5usXxbLcyf
QF9XxYtiZMwvi+rF1MmYXxbLFrEYfF8WyxaxMmmF8wuixaxMnLC1wuq+LUycsL5hdloVkyC8L5hdF8W5k5Y
XzC64XxbzMNehakLutFuanEha7V2K1W5rE5bV82rqsWTMTjtCxdliYziV6aljExMnCXNnHujcQWPsaWhMkc4
Tg2UFamRc7omKpKc3SDCYkMUYKyZBSufwVDV4dw5SrXyjBUM3iUYciqhzA7exlDDRO4eqSawfjerl6jk4eqO
6xqOJfCNCvllN1x4MqrXOBmcmR5SyskzK5NBdlWz0BSnkWdMprG4JEF2CjhDCPUb0X2O8z2q60lBET5IPK
17efZAbXpS160UDKQRkEYMXISHDDgg5BnvTYb/R6hsFHXwEYnjBI9ndwIEbi3ew9OioB/s9a0ENTU2SonDW
Teunyf1dwvQOWPjl/yXz11jp50mssr/T3Q/lz6k6B1Ma3p9VufOPLYP8AqEb7hFm2Ssb3aqmRWOoiurZziSjz
Dnq0HsfhW5keDGGnvwUP1Nnj4fsy1xwcJNRc1ZYe8tbKHA+RgXR2WkYzdENU7sop1VoCirpPPghDZcjrJso
3MaGDPC6eSQ5pLooRIOqtqfcGM1gAjgSlnfoujiIgljjeG+kux1PuozrPDK3TSb/la0EccYK9J57BQVsZBADvfCj
uu0xPTkuMW9nbB6YVh0/VLPRyDHCfk71C21q31EpdBPDq/0h3W291IN/gbIXD/Jye5aLxjo3jFVBVD/APLdt
J/kKz8VA8AFu4bSjejq1sQAdkDPVFN1Ak5atH+qwhul6RVzVIPkrIf6dpSvPixK/abZRR0PG7c4pXfo/xNuTwj
bqlQeCIIgP8AUq60l3qlyNwaQXDkgZ/CapNUVstnhhAJPsuRrefLRWPtn+8mr0AZDud349bCP/xxK7Un+zrcK
5rZa+71M2eSjPSB/kFJEHgXoK1QtNQ0TPPCNYr/AKlqvQwOAdnBRNBNN1VU9rqyYvJPQngLH12pxzZtHygb
aPTUkkisfRef3MjyDS9rhpZ4qOjijY5owGNzGousvh3RSOHnRekAFxI4+MKe6GzUIMACOMnvCIWN84hrcMa
Ak1uts5wxie/VrghFgVSWKjRTUsTY2NHXCW1lkpqOneQRulFuKPw6Omj2sblxQRcqWorjh0hCTO7HkmKIZ
meVavNgrgJpYWScyEB78bnH4CIHQ0H90tclO4keW84JRtT2OlomvfjJUd11wEe8Nwckj2UTu9gCDtmFjau4/
Ka3XNdX/bw0oijDo3Mw7DB1b7oXon7BvPJp2SiWuLGOcVptJIA9INnCsfxNYOYKH3MyT5AJx1QE+4Tzd

se+FJWk7e6aoDnNyAUetQRSTFF9hJwJNWhbc6itj9w9UsjpHfly5eIF3jt9gqpS7GlypDp2COMAcKhH+0lqj6K
wupmP9cvpXGiobVa+pB3dxINVaml0Nth7VoTPJ+61j7he62qcc75Dhd4H9kzU7cRp2h65X15p6hXUiDsABPk
TWWG2x2PcmEYAwVltlqzEeDgqL2PLnYCKGSGNiOIDCVa3KsD6yZ7hK+toHguzgKktlVVTtalpHE8td57MEe
+U+W26RsieJHcYXDTLIGagpql3LGTte78AoV0wjDHPm3jYc9zP0zaNnlfaKVzlxjblSkHktWgtV2qutVOYKhHG
wd1O7KyNw4cCvnPU1OtrgqRzPpDQ31Ppqtrg+UesINy3DK0CcLuJUEVjTmcw5bApv8xdA9c4m8xdlfcPhv
WwetYm4rW2UIDlvuWuZk7r6uG5fdy3mZOq+rllZlazMnRfCVzYvm5ZmZOi1yueVruWTJ1ytcrllfMreDMnbK
zk45XzkZyEysrVcsr5uW8TJ0ytMrmXLQuXeJrInXK1yuBeue9bxOcxVklXckhkXMyLc1kxduWJv8xYsmvVklirS
llWfdCb3rRsyabBFe+SDFWH3TsysKjiKZO0c5UZSSb4cvq+EwVNXwm91QdqHaqoODytBZheJK+sGDyogu1U
OeUTXGo4PKhq71hAdyj6q+RabbJfmoZ8h/KoxrRzvWrZ32t4cqhaqldw9W/QoQwR1zAgyslRnzSuGEunb/
dKTgJ6RzNlw2L9JzwtwFvhfclMTRMwBdQ5c1izE5ljXQXGqoayCpgeWSxPDM09iF7ReHWtaTVOMOkwYE7c
MqGdw8LxFypX0Brq5aUvUVTCXPhcQJ4ez2qr9c6UNdpfLjxU5T5/KWroHVT0/V5bPhWYDj2+c9n6uPDwUR
UTo5YdpQVZr3b75a6etpX74pmBw+Pgp9pC6Kct7LwC2t1ZIYEMpwQZ9HVWoyKysGVhKER7dAASMHhchB
0yE7SZLQ4JMwOLsDkKEGSE8zSCDZyDz2TiObWEPZklrEezktwHZW9xmAmCTaOimBcabyXnlc0HKY6izU7e
WzSD1dEYyQYce3PBTdK14AAU62t7mHJqbVxhoCTWqEkESOy3kZGU3sssY5JcT+PdHTWodkOb+E5w0+R1
HTIEi9/eSnXXAcGClupGOwwDhvbBUs0cUMfJG4jCY4aUZDj+AUW00TWJ3wuDYTFd1tlnLMYpkdCJlw9+0v
O1nuT1wE5Y8vGCEnBBDSMfBXR4w0HGSSulrbPaaPY/h2Op6pRhjY8kdspCGS7nFzsAdPgJHcqoQ0xOROUR
kycQHv8AcYoo3NDsEqMtS+oqi7B2golvidJPO4deo4TBBCWtHuEdVWEXPqZyWLH5QiZUsijyc57IcqJxNLjce
ey+ySAZByXJnlnCHHHJKPqr5gd1mB3hTsxAvYxv3FWq0xQCnp2e/dQNPw3uMglkbnorR0DQyMIXV2Y8g+
8i09ZJ3H7R+qqhkNM95OMBeG3jzqo3bVP0zH5ZCvTrxb1IDY9O1L9+HbDheEs9XNXV09TK4I0jy5eg/grpps1
Lalh5U4X6zz/wDGfUBVpV0ynl+W+gi6EJx3ABJluAEoIK+gxPnN+WjjAQ31FO7KuKUyYg2rnLGYQ3DcfKkPK
w2qpAnIOLXIW9fSSrIW4DwnlFdE80sQPdyjm1b6mUPPRFVbVtbi1vspgQRmJL6mV/DByfWW20hdq6KFro
aiRh/wulVmbR4j6poS3bXvePZ/Ko9pW6xtha1TITvcwEk1WlpdjurU/USbTX2VqNrSPYy/1h8bJctbXQcfvYr
OWfW1muMbXQ1TCT2zyvImGpaCEZ0VSW4LHlp9wcKm6roWmflMoZdNH17WV4DNvHznsAy4xOHDwl7
Ktp7ryQl1Rqejbmnucox2Jyiy+M+paRzRvMkdoPP6Sq7Z+H9SBlGvPzq/xPTuAsrZfn3nqo2YFdxlq76T8Q7P
e6cPhqG7/wBTM8gqZla6J44Kq1tFtTIXUgiXPT6um+sMjhgYUiRdA9MrZfldhlsQ3Md96+701iRbeYFrE6zHP
es3Js81Z5qzEzMcT6+b03eatfNWymsxy3r5uTb5qzzVvAmZjhuXzcm/wAxa+YsmsxxL1rvTeZfZMo91uazHLet
DImp0490ndUBZOdWjwZfYmQy3VKTOqR7rrBnBeP5IXEzBDzqoe6RPqx7re2cGyE5qAkxqfCL64e6bJLg0d1
2EkJsh39V8rFG/8AU2+6xb2fKc+KJCL4ykbmOBRT5bStjTNRK2yloYOxkpxYUqFMF1MGF3vE1gxJJwharm4
KeKpxAKAa6pwCpVnDcQauFR1UJ3mo4cpBuVvW7lQZeaoydym9C8iKlM4MhzUFWQHcqqt+q9znDknDUd
Xnfyqr3WcmVyuWITC5I51DFnCj1gxLy8pOQvpJJXMIgwtQQJssXPJWcrU6m/CwAk4CWwUc8xADUf2/Tsxw
XNUyVs3pa7tVvWwO+T7QKgt8kmEU01ie79ClujsbYwCWJ4mZTQN5cExtTr7Zleu11hz5sCPXh/qav0tVgb3S
UrziSHPA+QvRukuVLX00FVTSCSN4yCF5G3K7wtB28lj8P8AxSqNO3UNmc6ShmcBMzrs/wATVQ/xN+G69U
hvpAF6jlf5x/vPTPwh+JtRpWFF+W0zHyse6H/aex8UodTAHsEkaOTtJyeQmuz3ClrKOCop5WyrTMDmObOIK
ISwholwV88shRiCMHPIn0mj+CDkHsZzD9smCeic45w3d6fyEzu5OCPyujG5IIOFFCwO15ve18bSABnthN7qV
jSBvx85XfeXsK2DQ8g4A4HX2XYhSoMTg2hYXMy3/VPkVuhwCl+FxijYHNGSc8fhEICdz27Nob/quszT18d4li
ogMuBPX7fZojad23hoclo4wQcD8paKc4+0BdRW7QfjiLueP8koMDsjLhgDsE/Np+PtwtRSYPx1W+YGWEHn
ZYCF8A3UNahqsktD+vXCmC6kNa7PsoNri987txHpJXSjmc7uJGklKN253GeiaahwjaOQMj9uVcyHducAMFQ
Pc72ZZCyI5+U2ppewyBrISP9dc2NOG8kIO9ioZaqbzHg9UK2i0TVMjXPyrNWWgbCxo2oy6xKUwveCKrWtlu0
NrRSNijbwj9p9Wycnc9xwGhDUEgYMIVw8WnEx2myz7JPUWkBJKKbNReqKMLjiMrbUooZ2OFUZMpL4966f
dr19BDJmOM+tVYpoxgJpfVtVtdNUSnLpHEkohgWAvqro2hTsaKqtQc/WfLHXNdZqtXZYfU8fIRWG4XUu2t
WpcEy1tXsbhWikASppW1jgYiKpk3vPKYBTulqWtbzyklRUuPRH2mqMn+68JeMW2bfuTLDYfymlZyfTAEkSg
hbTUjR3wmqqhdI7clNTU5djsi2gpmTUp45TrAxiedl3rbxD3y5YJ3NcG5U30Ne5hALICFJRzsn9DT1UjRUIR6S
920KN1HrBC5LkqCZNNLXNeByiyCt2NzLQIL5Q0LcGQOIWjdd0UjwwjCXNSx7CGpZgd5PFXfBsIBUSXS8P5w5
dxW0s0O9r8qPK2ZskuAVNTQoPaDai5iveLYL1dqacT0tXLDIDw5jiFcnwy8fL9HVx0d4PnR5AE441/5lRepmZT
wHnkp+0ZG+epe/soddotLqKmFIYPHB9Z1odTq9PcjVWMPzz7Gfodst/pq6njejQODhkEFGgmXkrobX89mq
mU80hMBPBJ+1eiVn1NSVdOxzZAQQvDupdMt0tvbKnsZ790nq9WsqwSA47iS4J1989C7Ktjxw5bGdIZZYTG
daGpQo6pwrqv5W5omGBqR7pOasZ6oLfW8JrdcDu6rJGWkjirHutvqx7qOBXn3W/1591nEzdJF+rHuvhqx7
qO/rj7rU1x91uZukgurB7pl6uHugB9c73SF9a73W8TKvJafXD3SF9f8oAdWH3SN1WfdbEhLmSA6vHukprx7q

PXVbvdJX1ZHdStksYfSXEAduOVN6a39SAqy4FoPKi+43ZwJ9SmRcmDWORJkqNQMb+pCFVqpic+tV6r764A
+tRHc9Subn1plXp8xTZeRLh/wC98f71ioD/AL2O/esRf5QwX8yZ6zshCVCPHnsdXH7pcKmPHVVGyAy5ZE0MX
K5viK7idnUSDkz3UgnHEDa6F2ConubHjPCnOoLC0qOLIA1wKIqKGROoIlaRq6RoKgS9VDSOVpbxStldwqwakp
NgcQrFpWyREOpXCmVgv8zjvVcLgFW5T7f37S7KrzXvy4q708VypEZ1AjWCvpXJq7KWHGahpJRda7U+eRvCS
0NE57hwrAWC2bA0lqOppzyYi1mrx5FMclTp2ONoJYi90NPA3nAuw1bcIqSnPlzhV6vGpJ5HkApktYxk8CV8F
3fZWuT6mSfc73BC0hjlBl1vsr3H1IbqLII/OXZQZUTue45K4tvVfWsf6LpZL7rOYunuDnnkIjXKSmprSSnFrUpDu
xyTLb4VaAACXx8BPeyWgrYrDXynyJ3f9kld0jf8AtJ9ivVONzywcD+F+cCOofFK1wPQg4K9mfAzxlg1PY/pJpQK
6hY1jxnmRnZwXkf4r6QATq6l4P+IB7+89P/DHVSCumtP/ANs//rLSmLL3ZK7MZtd78JRgE47ZXE MH+CF5Liev
q0T5G5zQCC7v7JRJEHw85G1Oe1vGRgALsBG3AyclDgSxEBggRM1kW0ZHPBCeKbDng7jwEjLG4OG9U4RbQ
/GCP+qwrOd5IhTTRR5B3EA/KemklwwTkdkMwFhDdvACIKebp+eq6EW25joMHnaQQVrO1jGEpV5zG8uB
QRerrFEz70QBFHjZaA/VYdnLsBqgK63mngHqHyE36u1IBCXYCecKq9XcKy4THLzlgIIndPoy3LcCQteF4HJj/fr3
JWy+XF0SyzWNz3Nc9a2q1tBBIUw0NM1mEwttVE2JIEqLNuaFFroGRNapKgaGglRpcABKkm4siYclV6zLGMV
4E7X29R0dK8lwGAVH3xW1fLd7w6Bj8xsKtR4p63FNRSta/kggLzWEj5qh8rjkudlepFhXpXn8dx2+Gec/ifqOK/
AU9+WjtT+gBPjHkJsY3KcGcYXt6cT27BMWixDSgitLpCiSplAYUDS1DMIQahwaAATC9DT5i2J8iZ5k7G/Ksjb
7flyjDWHBHVV0oWkv3jSpxtGr6mlh8pzAQt6NgoJPrAut1XWhFTsvCQ5jsEjxl3CK6KClpiHSBRXNqyRwKE6i/
TP7pobF3IOTQ6ywgFf3k+VN+o6Zp2AEqPa3VFRM45kUQy180h5cUh8x5PVD+Kvol6q6Tgedvt6Qxqa6WV
xlem/dMe/KRwtPUpyY1znNAW8kwrYIyWAMCStYLhUMpi1xJGETUr90hcSgimligARFS+ZlZawEko4cCUTUD
dazAcBuJxqhlW1bYo8nJVjLLb2Wy3gD7iOSG6w2dlLiR4zIVlsvqZglL7W3EAdoZWNqwrqbnl2UqX9J68vVuD
WOJfF256KD7lRvySAmGK7z0pDDyPZQ26aq6vayhhMq1VtFoYOykeonqpp7xOinDcyc+xVgbdq2kqAAZBkrx
ut96DtuMtcpMpdWXXh2u3lzVQ9b+HK2JNfB9p6FofxPYgAt8w9xPXCVMcgy1ySSPKovpbxSJ2NkfkK2Ntv9J
XRnc2QHK861Wh1GmYh1M9L0mv0+prDVuD8oQTTEBMLp/UnCo6IYcTvS3MMxH9sxSkSImrMpUcT5mRy
81aGVlSrQkrqcxU6UpG6QrQkpK4rrM4In10iSukK+OXEGreZxtM+7ykM0iWbSkM0bithhN7TAm4zkAqFltUS
c4U5VtISCotudG0A06luYHasgO4STFpUHXP7wHZJU83uWKfJlT7VmoY49wBVm0qliOJW9QyqJy+p+Vihb/e
NyxPvAMVeL8p7sw6gb+9Ogv7cfcvP6DXzOPWiSLXELv8AvVtToXHoZahrkPrLyMvzSfvSv+uNx96pTFq+M9J
cp5j1WCPvWvyje0IGsWWylvrcH1oZqr2wg+pVpm1T/jQ5Uaqbz61Mukb2nJ1Y95Nd0u8eHZKrhqS5xFr+Uw
3LVTdp9SgC/wCpQ4O9SeabSMCOlp1GqBU8wK1PXsL3AFQRPKXvT7ca100jihcfcroOFAi2hOWcxUxPFLFve
E0tRpbIsuClqXcwEi1VmytjJGstGwEZCkt1bHTR8FR5HOIY8oVrLrIXEAqyBFUCUMC65yB9zH+73Uy7vUogq5
C5xTjUVLnZQ/17JQ1r5EtOh0oqGYgkTQ9vKeXJA5uSIngzLTUcTjE1Kn8BbsbgJPM7AXONqTrO543k5KmTw7
1RXab1DBX00ga5vDmnhrx3a74Kxgy5FVMz0KAUJcjo4yrAgiS23NVtZThII9CJ756Q11Y9UWeOtoZQqEjI3
HD43DsQpGFQzj1D+F+eCy6svum7j9VbquSGQdGfS74c3uFdfTf8AtO0z4mMvNuex4ADp6flpPvsK8S6p+GN
TRcx0/wDERzWP1CeydK/E2nupQajyP6tjyempJrQyM5IByyv9QixgkcdFSuj8Z9C1oYGXtjXOxx1zCEWQa0tE/
MVxhlGerZFU20OqT4qnX6qRLrXrdG48t6N9GETmLjEXN5AKUOrY3k+o4CrEy+PwXsc45xytv8AEgY53bhgnp
0UH5az2hlup/nEs7Hc2bBh/PdOkV4jHR4GO5VSnallDPTncO2EyT6guUr3eXE/HueFluMsMhe6rnnMthdNWR
RQnL8kKsOrdFp8qRkAL3cjhBMzK+qePNmcG45af8/pLMYACY101IQWO4xVYzPwOBIA+1Dda909UfljJ4YC
pUt9sEe1GTLc2Nd9jQeEdbqGfAAAA9BB66EX3J9SY40kTBhF0LmBA7Jdqcm1fpSxlyWuHBrQxvVRXqG+COF
5344SusrtsZOVtbxL1Q6mpXRsd65OEw0Gia+9FA7mLdZq1oodz6CQLr2/wD9RubmNflrCoziy0hI4PW8uceS
U/siC+g9DplqoRV7ATwTXalrbnd+7GOkCXHjqEY2rtLJ6OqsC9pVHXLxqq5MtKj+QF0ie6yocMhN9EN03KQa
lxZaqS16VDVSWHnQs8tgyiH+0W/KbhCSMk9F8+ojZx1KbphFA9ljtzY5lyTNznuaUmDyVu7zH8luFu2PBWuS
ZMMKvPeZG15KeY2Djhc440uPpCIVcRfbZkZxAAClbfGMbig87y4FSbbKN0sA7BE18mJdawrp794op4paqds
bATyptt1LDSMYA3Lu5QIQ0zKdgmXU9fFUbnOcpWzKo1gJ4HbtDqnkeSMog2kjKFKWRoP2EonM7WxZxh
BsJKMRayJr2SBkpsm0qx5Mw5Pso+u2pH0jvRjIqIPEitYDjCzbYOQQJ0gFnGwnEM545aap5jIDSjqnq2SxAO
UJ0mo7rcCXvDce6fmCsZ6y/GeoClkZaziDFjWxAB+clNwZad8TsFshpjWs9JKweZj4zwVXFlwJaaSuE7Z2ncxA
36Gq9ClghtGvvosD1HGO4nrrnp/VdPclGguGUAeAuyvKTSutKqgqGNkecAhei+ltS0tzpWEPBOF431jo1ukcsBl
DPaejdao11YBOHHcSUGDhKAFq0LuFusy3FTOeFrhKML5hd5keljcEnc0px2rXy1mZrbGssK2EJKd2QEzjPgT
TrbGVlLwtJacAlpLGtah2tkw0rc3l/uDmsaVAl+rMncpcu9RgOVatRVPDUz04yRFWobAMr9qy4ERv9SoHqe
tdJUu9StjrCtxHJz7qjV3n3zvPyr9ok215lQsy96iNfm/KxNO9YmW6MPBEkxt6rW9JSI7NTV8f/eKMDVFcHVDI
MWr9oANFJnj1pWN6lPkHiDVMly4quDpnrgZH+6gZq/5RCBoP+th95b9niAyZn3YckM2ri4cOKqcJHg8FO8V
ZlBy5dlav5ZDZobR2tjK3VeoZX5w5A9TVyTOyXlcbUE90oD8owbfQQTWGU88zd4TePuKvveAEmYfKLTThFO

TcuCN6NwYAguHgp9jlw9NwxcTq0LjEJqqs9PVCb5i5xSWpqCSmsTHcmL28yPT6QIkdn9EzvPJSzCUkcoGIIj
KtcTQpMBly6OK1YEKeSlaOFMUcAJlmdlyZDgJjPJUNx4Ak9A7mKIQSUE0jAIM/CEoY8BFkTv7eETp1wlu1rF
sAe8Crg7+65MKfK4f3CmlpHqfjMfabHhLOdsqSNJahkt9awF3oJUCFc8kEFKbFVlIPIMZozKwZThIOQZ696Q1
RDNCzc4EFWaihPz49zQMFep2idXyUsjlZX8Z4K9AtN6rwxmX7mlea9R6c9Tkr2Paeo9N6hXfWM8EcMPaT+
bcGnlSaWna1d6a6wzxxggpoulcyNp5VaActiWU7ds4SyRxpCa6NoQDPcy95w5aMmc7ujxSYCXEMH1m4dU
mZNYhp8uzuuP1nYFSeHxlt5hY94J6pM6owE0xyPc1cZSWA5K42yXdxGa71+yJ5J7Lzz1ZdTctIM+hWfZHC
1F9JRPY0+p/AVNo2PcS8r0z8O6LANpHyE8x/EetyVpDduWmriXtfe8f2BNTH5KcY2nC9LqULnE8yvYsBmLm
OBTfVyhokVgDGUK3CXspbn21kyDT177RGCol3vS6iZl3HVNDW5ciW0YNUMqu0Ze9SfUyzX4ShsegjyyeTG1
o5SuFjYjukavtURT1GWjIKVodJKWHyxwriByQTkiVdmyoiGFbucxeGCduWtwAuHlBpwVyiqaSkMZFGJV5b94
a7qUQMh0gBDK2CQB6DOTPrfdJ5HuJA7l3R7XBoGSUBWuzMY4TTJHQLvaTxA7dTVUu5vslhtdkkftkn9LP
b3R4+YR7WxtwG9Aks8xztJSHzSYyAR+UYqhRKlbb25u3oIQQ3J7Th3HuiKqEjWuD8OCh8SkOIdn8pXTV
UscnDi4LODI205wcGWLt827Y8Z9inO71WyAY691HtmrgdgyRuRXX04ILY5AcHuCoCuGEG3HYRINr/OqZnYy
cllI9LXCYgsjJBvj7fpiMHOAQVKFBZ2MwMYGFy9tYhtDahcBeBKjQacuFMM+ds+ESUhDSGvny78q2stttbmE
TxlI91CN90vTPd5tAN7QecLmvUlxxjEhvqt+IkN/eRjWz1EUUnROttvDXEMkXCoelSpojlCNVG5ji5rSAjMAiLIP
m7yYpY6dzd7Ee6R1jLaq6MGQ7MqtVFepWna8HCKHtErPMjKDu01dtbl4yphqX3ae1bEO1hzPaPTV+prlRvx
a8EkKQGF5T+F+uJ7dXR0s7zscAlepIrlqumY9pByF4H1rpdmi1BH6D2M+geidWr1+IB7OvDCLw1biMpeGB
dw0KtZloxG4QpU2AJXwvhetzU+tYAuUQkxetS9dZmprM7hBFwlwCiOok4Kjy5TcO5XQMhaRneZ+Hcqr+o6n
h/Kne9VBw7lVY1LU4a/lPtIvliLUnqyqms6z+3Jyqf1r8vKsLrKqJ3DKrfUOy4q/UjFQldoG69jECxYsW46nValdVz
XcjESlcyuhWhURhImi6tJXJbhaE6MXsenBkhTS1L2o1CYBYoi7OUpakjUraiBAHi1hSoOSJuLc8gI6kwErkxPM
/lNgf6l0mem4HlcWP5o0qrG2PrXrCkbHcLv5gUysMSEoQzqQILG4C5tcCuPpDV2AJGxPaNdQ5I4W5ct5Tucl
9PFwhcF7IYSEqi0DACfzPzLSUOSSbSuct0Lli1qK8VEJJPALXptsACjvONEBvKH3ALSSofI5bbHFqR2OtejEgR/VW
a0UExG5c10cCFGCXecw8EYmrXuY4EFWH0drJ8LmRTP46AqvRCOZl6NwIKGuqR0KsODCKbrKrA6HBH9Z6q2
i9OKjaYpeqeaavn+56oJpbWk1I9jJH8K4tk1VR1cbcvGVR9VoXpfiX9DPrtD1GnUJjdhwvYwWxPHUJ2AcB1X
Vk8DhkOC4SSx46pSSfaNiAljk+7lyxnVJJmZWgnZ7rcjhVDPtb0Q1drIHfBI9xwAFzdWADAVbvEXUflQfSxv9b
+qJ0eka/Ulg9TA9Zqko072H0HA9zIP1NeH3S6yP3ZjacNTCzkYTUzhd2TYK9v01aU1qgGABPDtS1l1jOTkk5M
X7NpyE9QPa5qQRPa7hKmsDeQU3TiJLTkYPec6hwjYUBVExc4olrZc5GUKvwlescnGODXhcnvNI+6JLU3Eh
ch4fYiy3sliJx1UGkX+lvyEn1bfwW+fEW1bzI38LhSv3DA3EpbKz0+yaljG/c04bnlN3yLAYmrCmogfaFsJfEA53
ASmlukl9lJWsDWzgerlCfloH05a7aS132/KOUdvalLbFut/P7RzgpS0hwOT3Uow0QntTJWh2Wuw5RxBUOef
VJsHsOgUi6aqo5ZJ6cvd/caQD2RgwBKpqBYx59IK1OGuc0odje8PLXDgopuj20lW0PjJldghFcltoKilinj4JABBX
RlZlq3KVjKEhhoJ7bucrrTQHzehJzudlylnAC+26FxJPXl4WvWFG0eAWBhbTkwtYQ48ckYU3WF9HcA0NIDZG
n7D3UCj6hwa2Dj/Nd6ZlW2dro97HA9R2K267h3iZDtbJLlxxwXgAu/k8JNNfKGmar5jSR1GVXWou92EDWT
Fwfjh+OP5UO3C71b5HNLjkHkID8v6s0ZI7u+EUfeW8m1Tb5XYe4OB/SCkrKqjdc+Shc5mDy3Kqfb6a4TvDm
gk9cZR5Hb7tI0yQ1JhLpVBOMKbwCyFyRZgsp+Y9JMDTTVMgBM+lyHpuGCmGurqKmmMNRrkqcbgEG01
dO9xFQ+Nzo+jv3J7q5sxMJdua8dCu9pBEGLDn5esVsg0zURI4OzCcqOTTrtFS1RnLRN5dG7ju1AF0t72EvjcQ
umU4OGM7pCPYAdq+xxLSft27h8Uwz1BBV0/C/XLHsZSTTZeZAXjxb7vUwvDHvOFMnh1DVUNxgqWSHDX
Dcm9QkfUenLrNI6E5YDKx9o9Vd03Wo+0bT3x2In6Alpg+MOB4KVBygbQOqoLnbITvBjAFOIK+eLqnqtZGGC
DPoSm5LakdTKMMxSXLTK0XxQ5ks2ytHOW2Fxf0W5uM1U/gqMLrL6SpCrHYaVEd3l4cpU7yCw8SGr3Pw5
VN1TV4a9WNvs/3qn2rqoBkitWiTLCVjvthTKk6nqt87hnoofOXI0vM++d5+UDOPqV3PCgRbo18pPvMWL7h
YtRlmdVYK3XJywkCcCuZW5XIIQmFCfF0C4hdwsE6aKwpa1I2pY1FrAXixqVhI2pZHyQiVi946sjyxlJ+AiGJn9t
MVdwmYTCZI91ulOyOSTcujuKuUrduZZUXiLA/hfC8ri1alZuOJm0ZjnA4lOT3elN8DSnExOcQExrzi6zbvIKKI
veilRhjEspaJwGcJDWTCMotawiZMXtcbBqQ+kR/SPkY4oOq43MfgovjuzWMIKEKqczSFyUas0msYOWjbrjU
Cxy4X0iill027whsJ2pZjGUBp2AfmMNQRmNHeKkin56JG2ApwkrGEpOaxnsi38Dd3EGQ3bQMRG+IhInMKe
HVMbgkuQUI6VnsYSju04jU1xa5HlPv1TSubh5wgWRuCTWOIKWEDISMiGgsMMpwZcS0a1e9jQ56kqHUH
mNBD8qhtPVPYQQ5G9Ffp48AuKU3dOrbly90/WbFwtn7y5Yuf3Slk7yq60OpM4BcpFpL2xwHKSwaRk/TLR
TrarBw0N7ldl6SkklecbWqIF1uElfXyzvPU8KRdZ6hNS8UOTvSPuUPbsDct/R9EKqzYw8zSkda1vjXctTIU/qYoH
qXwjBWjQV3wrbKmTgxVC8ghOz5MMTPE1fZJRIFq2FgL1hnjdM17nEpoe05T5LMzamPcXSJVftyOckmN6N2
O06kdAjyYPAQlZ1TAI3ZtjA5Rmj+Jj9oBrslFX17xXVHbEU0UcsiY8SHgrtVzAx4JQi5x3jCnvu2OCIPpqN9RBJ
GTJRpopKaPz2OLqbcGvd3YSjSlPNA9zml4aOOcJmsrmMt5glbmOQerHfKQCQuBpaOXq0hzH9SwpSnlVfyj

9jKlevi2uOd6NwR+pfeKGOcwePEw89Pj3R7bq2hgEbt7mHPUKO62pE8ke1hG0dcdl3gfTs5kPmhSK2CRIL9P
4tYLBgfVRDLU1S6aopyHlxcRzjCebdcYmjyWuBJI4JQi7yJK2DeSyHYA00OXAnokMbpaes8ogQFzj5T8c72np+
Cu92CYD4CvQqc5VSZJVzbE5+XyNaOCUz0dWNz3xQyyCPk4AbjB+VrQVNE36unmpy6QO3RvaS5wa7r6vYFJ
aq+QQueycwMeWFkn9zcSWHg4ZnrZsUd2Agi6ZyCoRn7EfeFcclc2sAddGwOaXAYOyC09D6e4Wzam7RTP
Bq2t9IexzGbgS0/KCKe9OHINZ9TKGlzQ5sGAY3D3eIMkty8hsj4j/AGuBvnDRj29IK1vQj1P2M2dNYjAYrX69/
wCslKSsrwwFlxc9h+9j4wQA45UbXdkgqWvdCGPJlJb9sg9wOyH4r9JEqx1Kz7A0kVbjuwc5OQu4u7KuQMm
Y9pA9G7H8bXDgrhXrPGCJMNIqam3EBIA9MZhhQ3ScNljkDmZxh2E4Pq7h9bzNhnbcCFHjxWetvn4wCWWh
wDsrpbkZKER1IMyaLOA+M7Xtz7BTkn2gbaZe4K894T15o5omvptgkb1YRw4ptp7ix0To5mmJ47E5H8IThp
TE97qOqJaOTFJ1wnCompKqkPI+meMeph7/hYDljQudowR74wR9ZvXVc9O1xDwRjqCo9ffJ3u9ZyEx1NZLG1
7dxweyaliXOCDe4lhiWrS9NrSsl1DexhZl3zQHtTxQ1FU0gZyFxo2ejolb2OYCCiVX1glrQa8A+2Zb/wz8QJ7X
Vw08r8MJ4K9ctOXyC40cb2vByF+dymqy4BwOC1egPgz4kuZlykqJeRxyvOPxJ0cWldRUvPdhLP+H+qNRYNN
afKThT7T1cC6gJjoqkx8LHtOQqNprl47PWwZ12pNKEvCQTnhanJ7QLr3YBULXiXhyLu5SdVBV7mwHluoZaBX
HiQJf58B6pVrOrwyRWq1FVYD+fdUY1rWjDwCrv06vzCVHWtxK6V0m57kwJZUPy4pCFZWPMKpTbWJusWL
FqTTouLuqxYtmcr3nArkVixQmFLPgXdqxYtr3m2itqVNWLEYkAeKWIOEH3rFiKTuIDb2MLYfsQvcTyVixOLP8
GJ9J/jwWckjuqxYq7Z3lvWd2L9QWLFIPhE4Pcwno2NJCO6OmhlxkLFis+IAwJTteSN3Mf6mNjITgKFRm9xkK
xYtdR4rkXROXMFytCsWKKGegifQlJRwsWKRPWcNEz+qSrFiHs7yVe03al8SxYpqe8js7GfZ2AJqWLFHf8AHOq
vgE7sKXNcVixL2nFkcYZpGu4KNYa6oZTvd2WLFpwCRx6zrTswL4JHBgVI9znEk5JW8YWLE/r9IsPwxYFs7grF
iO9IJ6zoD6U1zOKxYuLfgkIXxpcSt4lixJR/iCNj8M70/MwR2D6AsWJ3ovgb6xFrviSDtcT5uE204BmblyS9n+
P/AN0Lq/8AT/8AbJGtcrw5zc8ZCXamcY5aWZvDwBySWJ8xP5RpUMD/AirWMdw2f2jbUV1Qd3IHoyktPKW
OpcAN3usWLkk7zz6GEBVFAwAMsl928eZ5b3Elx3Nz8LtVzPmtUbpDuMfmOaT1y0rFim/9o/5YA4H5IfiYMF1
iWPdPbfqHuJJIOwcMGe2AlIJUGKWPpyoo48uGdrAMrFi1V8CH1IGTN393X0yePTtJAMUbowSOpK3p4IhJKw
D0kAkLFiz+olKLntbkxnlTRNeSl+6ZRExokiH288f7rFihYAeKeUO7JyxP1MHhUzuijeZDwTx2RHb7nUyVQidt
LSQencLFiFrJyOfURpq608FvKOAFSS1TwU8zAHwtPmbsnuMexUb173RSktPLOAVixFN6ysaTlwJHFzPrC2o2g
7SsWJb/wC6ZfR/6RZJNK0BgwLzmgTWLE4HwyjWk+IfRgPaI5/T3T5bq6ppK+KSF+1zXDBWLEHYAVMMbtn12
z2u8KbxXVImhMz93pVm2OKxYvmrXgDV2gD9U940JJ0dJz5RF7XHCbKh5wVixLowPaRtc3uw5V+v0jsOWL
Ewo+KLb+4lR9TyvDXnPuqGauMkdM4ErFi9B6f8Mpuo/xl+okKyE5XILFiZx2vYtBJWLFi5m5//9k=' alt='img'
class='img-responsive socendary-img' />

<!-- <div class='product-grid-atb text-center'><i class='fa fa-shopping-cart'></i><a class=''
href='#'>Add to Cart</div> -->

</div>

<div class='product-grid-caption text-center'>

<h4>Batman Superhero</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'>

Date


```

<img src='img/img (31).jpg' alt='img' class='img-responsive socendary-img' />
</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='#'>Janet Evanovich</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>Date</li>
</ul>
<a href='#' class='product-grid-img-thumbnail'>
<img src='images/girl9.jpg' alt='img' class='img-responsive primary-img' />
<img src='images/girl15.jpg' alt='img' class='img-responsive socendary-img' />
</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='#'>Ravi Kumar</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>Date</li>
</ul>
<a href='#' class='product-grid-img-thumbnail'>
<img src='images/boy4.jpg' alt='img' class='img-responsive primary-img' />
<img src='images/boy5.jpg' alt='img' class='img-responsive socendary-img' />
</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='#'>Venkat Edukondalu</a></h4>
</div>

```



```
</div>
</div>
</div>
```

```
<%-----old-----%>
<%--
    <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>Date</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 Looking for a Man</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>date</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="#">Man Looking for Dating</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>Wed</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="#">Pair Met thourgh our Portal</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>Date</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="#">Looking for Dating</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>Date</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="#">Looking for Dating</a></h4>

                </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>Wed</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="#">Looking for Marriage</a></h4>

```

```

        </div>
    </div>
</div>
</div>--%>

    </div>
</div>
</div>

</section>

<%--dynamic code APPGRID ENDS here--%>
</div>
    <%--dynamic code ends here--%>
</div>
<!-- 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',DivG6')"><i class="fa fa-angle-right" aria-
hidden="true"></i></a></li>
    </ul>
--%>
    </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>
```

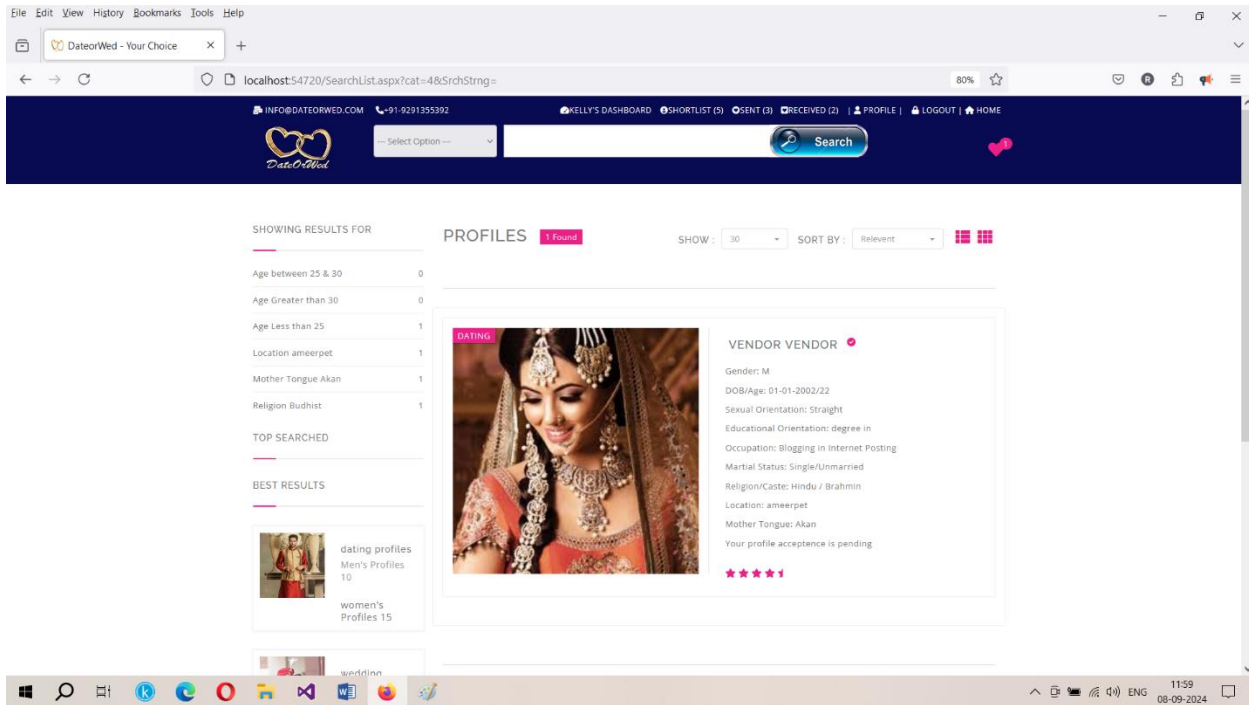
```
<!-- END MAIN CONTAIN -->
```

```
<!-- ./Content Bodey Here -->
```

```
</asp:Content>
```

```
<asp:Content ID="Content6" ContentPlaceHolderID="ContentFooter" Runat="Server">
```

```
</asp:Content>
```



***** SearchProfile.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 DOWEntity;
//using LogTracer;
using System.Text;
```

```
using System.Text.RegularExpressions;
using System.Net;
using System.IO;
using System.Drawing;
```

```
public partial class SearchProfile : System.Web.UI.Page
{
    DDLogTracer objlog = new DDLogTracer();
    bool IsMobile = false;
    string urlqry = "";
    BalCommon blc = new BalCommon();
    protected void Page_Load(object sender, EventArgs e)
    {
        string SrchStrng = Request.QueryString["SrChStrng"].ToString();
        //if (qry.ToString().Substring(0,3) != "pid")
```

```

//{
// Common.ShowMessageHtml(new BalCommon().Decrypt(qry), 1, true, this.dvMsgBox);
// urlqry = new BalCommon().Decrypt(qry).ToQueryString();
//}
if (SrchStrng == null) { SrchStrng = ""; } else { SrchStrng = Common.Decrypt(SrchStrng); }
ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
if (Clsusr == null)
{
    bool ret = new Common().guest_login();
    if (ret == false)
    {
        Response.Redirect("~/Login.aspx");
    }
}
if (Clsusr != null)
{
}
else
{
    //Response.Redirect("~/login.aspx");
}
if (Request.Browser.IsMobileDevice) { IsMobile = true; } else { IsMobile = false; }
if (IsMobile)
{
    // sidebar.Attributes.CssStyle.Add("display", "none;");
}
//if (!IsPostBack)
{ Bindetails(SrchStrng); }
}
private void Bindetails(string SrchStrng)
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"]; ;
    string title = "", qstring = "", pagesort = "";
    title = "Profiles";
    int pid = 0;
    var page = Session["page"].ToString();
    int curpage = Convert.ToInt16(page);
    string cat = "";
    ImgBtn1.Enabled = true;
    ImgBtn2.Enabled = false;
    appvendors.Attributes.CssStyle.Add("display", "none;");
    AppGrid.Attributes.CssStyle.Add("display", "block;");
    int srno = 1, perpage = 30, dvno = 1, sno = 1;
    cat = Request.QueryString["cat"].ToString();
    // int len = cat.Length;
    // cat = cat.Substring(0, len - 15);
}

```

```

//    cat = cat.TrimEnd('\0');
pagesort = Request.QueryString["psort"];
if (pagesort == null) { pagesort = "N"; }

string ppage = Request.QueryString["ppage"];
if (ppage != null) { perpage = Convert.ToInt16(ppage); }

SrchStrng = Common.Decrypt(Request.QueryString["SrchStrng"]);

if (SrchStrng != null && SrchStrng != "")
{
    //    len = SrchStrng.Length;
    //    SrchStrng = SrchStrng.Substring(0, len - 12);
    //    SrchStrng = SrchStrng.TrimEnd('\0');

}
string PreHtml = "";
StringBuilder sb = new StringBuilder();
StringBuilder sbg = new StringBuilder();

sb.Append(" <div class="">>");//directory-list
qstring = "cat=" + Request.QueryString["cat"] + "&SrchStrng=" + Request.QueryString["SrchStrng"];
Etetbluser_options opt = new Etetbluser_options();

string fromage = Request.QueryString["fromage"];
string toage = Request.QueryString["toage"]; ;
string Location = Request.QueryString["Location"];
if (Location == null) { Location = ""; }
string Religion = Request.QueryString["Religion"]; ;
string MotherTongue = Request.QueryString["MotherTongue"]; ;
if (fromage == null)
{
    opt = new tblUsers().Gettbluser_options(Convert.ToInt16(Clsusr.UserId));
    opt.Location_search = "";
}
else
{
    opt.Gender_Search = "0";
    opt.Age_from = Convert.ToInt16(fromage);
    opt.Age_to = Convert.ToInt16(toage);
    opt.Location_search = (Location);
    opt.Religion_search = Convert.ToInt16(Religion);
    opt.MTongue_search = Convert.ToInt16(MotherTongue);
    opt.Area_search = 0;
}
if (cat == "") { cat = Clsusr.GroupId; }
//try

```



```

{
    IList<EteGetSearchResults> Lst = new List<EteGetSearchResults>();
    Lst = (new tblUsers()).getsearchprofile(Convert.ToInt16(Clsusr.UserId), SrchStrng, cat, opt);
    if (Lst.Count > 0)
    {
        var pst = (from c in Lst select c).FirstOrDefault();

        title = "Profiles"; // pst.Department;
        sb.Append("<h2>" + title + "</h2>\n <small class='category-num-coun'>" +
Convert.ToString(Lst.Count) + " Found</small>\n");
        results.InnerHtml = sb.ToString();
        sb = new StringBuilder();
        if (Lst != null && Lst.Count > 0)
        {
            string pageshow = "block";
            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");
            IList<EteGetSearchResults> Lst1 = new List<EteGetSearchResults>();
            if (pagesort == "H")
            {
                Lst1 = (from c in Lst orderby c.age descending select c).ToList();
            }
            else if (pagesort == "L")
            {
                Lst1 = (from c in Lst orderby c.age ascending select c).ToList();
            }
            else if (pagesort == "R")
            {
                Lst1 = (from c in Lst orderby c.Religion descending select c).ToList();
            }
            else if (pagesort == "C")
            {
                Lst1 = (from c in Lst orderby c.Caste, c.SubCaste descending select c).ToList();
            }

            else if (pagesort == "A")
            {
                Lst1 = (from c in Lst orderby c.Location descending select c).ToList();
            }
            else if (pagesort == "T")
            {
                Lst1 = (from c in Lst orderby c.GeoLocation descending select c).ToList();
            }
            else if (pagesort == "M")
            {
                Lst1 = (from c in Lst orderby c.MTongue descending select c).ToList();
            }
        }
    }
}

```

```

}
else if (pagesort == "G")
{
    Lst1 = (from c in Lst orderby c.Gender descending select c).ToList();
}
else
{
    Lst1 = (from c in Lst select c).ToList();
}

foreach (var dt in Lst1)
{
    string path = "images/NoPhoto.png";
    string path1 = "images/No Photo.png";
    string href = "";
    string OutId = Common.Encrypt(dt.UserName.ToString());
    href = "ProfileDetail.aspx?id=" + OutId;
    var list = new tblUsers().getuserImagesbyuserid((int)dt.UserId, "").ToList();
    int cnt = list.Count();

    if (cnt > 1)
    {
        path = list[0].ImageUrl;
        path1 = list[1].ImageUrl; //"http://dateorwed.com/" +
        path = list[0].ImgPath;
        path1 = list[1].ImgPath; //"http://dateorwed.com/" +
    }
    else
        if (list.Count() > 0)
        {
            path = list[0].ImgPath;
        }

    if (path == null || path == "")
    {
        path = "images/Nocolor.png";
    }
    if (path1 == null || path1 == "") { path1 = path; }

    //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>" + dt.GroupName + "</li> \n");
        sb.Append("</ul> \n");
        sb.Append("<a href='ProfileDetail.aspx?id=" + OutId + "' 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='ProfileDetail.aspx?id=" + OutId + "'>"
+ dt.FirstName + " " + dt.LastName + "</a></h4></li> \n");
        sb.Append("<li class='pull-center'> \n");
        sb.Append("<ul class='list-inline product-ratings'> \n");
        if (dt.Status)
        {
            sb.Append("<li><i class='rating-selected fa fa-check-circle'></i></li> \n");
        }
        else
        {
            sb.Append("<li><i class='rating-selected fa fa-times-circle'></i></li> \n");
        }
        sb.Append("</ul> \n");

        sb.Append("</li> \n");
        sb.Append("</ul> \n");
        string martsttus = new tblUsers().getmatstatus(dt.Martialstatus);
        string relgn = new tblUsers().getreligion(dt.Religion);
        string caste = new tblUsers().getcaste(dt.Caste);

        sb.Append("<p class='mb5'>" + "Gender: " + dt.Gender + "</p> \n");
        sb.Append("<p class='mb5'> DOB/Age: " + dt.dob.ToShortDateString() + "/" + dt.age + "<p>
\n");
        sb.Append("<p class='mb5'>" + "Sexual Orientation: " + dt.Sexornt + "</p> \n");
        sb.Append("<p class='mb5'>" + "Educational Orientation: " + dt.Eduqualification +
"</p> \n");
        sb.Append("<p class='mb5'>" + "Occupation: " + dt.Occupation + " in " +
dt.Occupationtype + "</p> \n");
        sb.Append("<p class='mb5'>" + "Martial Status: " + martsttus + "</p> \n");
        sb.Append("<p class='mb5'>" + "Religion/Caste: " + relgn + " / " + caste + "</p>
\n");
        sb.Append("<p class='mb5'>" + "Location: " + dt.Location + "</p> \n");

```

```

        sb.Append("      <p class='mb5'>" + "Mother Tongue: " + dt.MotherTongue + "</p>
\n");
        if (Convert.ToInt16(Clsusr.GroupId) > 3)
        {

            sb.Append("<ul class='list-inline add-to-wishlist mb15'> <li class='product-wishlist'>
\n");
            sb.Append("      <i class='fa fa-paper-plane'></i>      \n");
            sb.Append("      <a style='text-decoration: underline' href='SendProfile.aspx?psid=" +
OutId + "'>Send Your Profile</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='MyShortList.aspx?psid=" + OutId + "'>Add to Shortlist</a>
\n");
            sb.Append("      </li></ul>      \n");
            sb.Append("      <ul class='list-inline add-to-wishlist'>      \n");
            sb.Append("      <li class='product-wishlist '>      \n");
            sb.Append("      '<i class='fa fa-heart'></i><i class=' fa fa-exchange '></i><i class='fa fa-
heart '></i>      \n");
            sb.Append("      <a style='text-decoration: underline' href='MyHeartList.aspx?psid=" +
OutId + "'>Check compatibility</a><br> \n");
            sb.Append("      </li></ul><br> \n");
            sb.Append("      <div class='product-grid-atb text-center'><i class='fas fa-shopping-
cart'></i><a class="" href='ProfileDetail.aspx?id=" + dt.UserId + "'>Add to Cart</a></div> \n");
            //sb.Append("      </ul> \n"); sb.Append("      <button type='button' class='btn btn-
product fa fa-heart'>Add to Cart</button>      \n");

            sb.Append("      <a href='MyHeartList.aspx?pid=" + OutId + "' class='btn btn-product fa
fa-heart'>Add to List</a>      \n"); //<i class='fa fa-heart'>

            sb.Append("      <br> <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-half'></i></li>      \n");
            sb.Append("      </ul> \n");
            //
            sb.Append("<a href='MyHeartList.aspx?pid=0'></a><img border='0'
alt='Compliment' src='images/DateOrWed_ico.png' width='50' height='50'/><p>Check compatibility</p>
");
        }
        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-3 item'> \n");
sbg.Append("        <div class='products grid-product'> \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>Date</li> \n");
sbg.Append("        </ul> \n");
sbg.Append("        <a href='#' 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'><i class='fa fa-shopping-
cart'></i><a class='" href='#'>Add to Cart</a></div --> \n");
sbg.Append("        </div> \n");
sbg.Append("        <div class='product-grid-caption text-center'> \n");
sbg.Append("        <h4><a class='p-grid-title' href='#'>" + dt.FirstName + " " + dt.LastName
+ "</a></h4> \n");
sbg.Append("        </div> \n");
sbg.Append("        </div> \n");
sbg.Append("        </div> \n");
//grid code
if (Convert.ToInt16(Clsusr.GroupId) > 3)

srno++;
sno++;
if (sno > 3)
{
    sno = 1;
}
if (srno > perpage)
{
    dvno++;
    srno = 1;
    if (curpage != dvno) { pageshow = "none"; } else { pageshow = "block"; }

sb.Append("        </div> \n");
sb.Append("<div id='DivL' + dvno + "' class='products-list products-list-description-brd
mb30' style='display:" + pageshow + "'>\n");

}

```

```

    } //for each list

    //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, "ShoppingList", "1", "appvendors", PreHtml, "Log");
        appvendors.InnerHtml = PreHtml;
    }

    PreHtml = sbg.ToString();
    if (PreHtml.Length > 0)
    {
        objlog.Trace_Log(4, "ShoppingList", "1", "AppGrid", sbg.ToString(), "Log".ToString());
        AppGrid.InnerHtml = sbg.ToString();
    }

    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");

    }
}
else

```

```

{
    //data not found
    Common.ShowMessageHtml("Search Options or Data not found", 1, true, this.dvMsgBox);
}

sb = new StringBuilder();
int grp = Convert.ToInt16(Clsusr.GroupId);
List<getcount> prd = new tblUsers().gettotalgroupcount(Convert.ToInt16(Clsusr.UserId),
grp.ToString());
if (prd.Count > 0)
{
    sb.Append("<h6 class='text-uppercase bottom-line'>Showing Results for</h6>\n");
    sb.Append("<ul class='icons-list'>\n");
    foreach (var data in prd)
    {
        if (data.count > 0)
        {
            sb.Append("<li><a href='SearchList.aspx?pid=1&cat=" + cat + "&SrchStrng=" +
data.SearchStr + "'>" + data.Desc + " <span class='pull-right'>" + data.count +
"</span></span></a></li>\n");
        }
        else
        {
            sb.Append("<li>" + data.Desc + " <span class='pull-right'>" + data.count +
"</span></span></li>\n");
        }
    }

    sb.Append("</ul>\n");
    //objlog.Trace_Log(4, "SearchList", "1", "paging", sb.ToString(), "Log");
    Filter.InnerHtml = sb.ToString();
}

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='" + "SearchList.aspx?" + qstring + "&ppage=30" + "'>30</a></li>
    \n");
    sb.Append("    <li><a href='" + "SearchList.aspx?" + qstring + "&ppage=45" + "'>45</a></li>
    \n");
    sb.Append("    <li><a href='" + "SearchList.aspx?" + qstring + "&ppage=60" + "'>60</a></li>
    \n");
    sb.Append("    <li><a href='" + "SearchList.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 = "Age High to Low"; }
    else if (pagesort == "L") { sort = "Age Low to High"; }
    else if (pagesort == "N") { sort = "Relevant"; }
    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='" + "SearchList.aspx?" + (qstring + "&psort=R") + "'>Religion
wise</a></li>    \n");
    sb.Append("    <li><a href='" + "SearchList.aspx?" + (qstring + "&psort=H") + "'>Age High to
Low</a></li>    \n");
    sb.Append("    <li><a href='" + "SearchList.aspx?" + (qstring + "&psort=L") + "'>Age Low to
High</a></li>    \n");
    sb.Append("    <li><a href='" + "SearchList.aspx?" + (qstring + "&psort=C") + "'>Caste
wise</a></li>\n");
    sb.Append("    <li><a href='" + "SearchList.aspx?" + (qstring + "&psort=A") + "'>Location
wise</a></li>    \n");
    sb.Append("    <li><a href='" + "SearchList.aspx?" + (qstring + "&psort=T") + "'>City
wise</a></li>    \n");

```



```

        sb.Append("      <li><a href=\"" + "SearchList.aspx?" + (qstring + "&psort=N") + "\">Country
wise</a></li> \n");
        sb.Append("      <li><a href=\"" + "SearchList.aspx?" + (qstring + "&psort=M") + "\">MotherTongue
wise</a></li> \n");
        sb.Append("      <li><a href=\"" + "SearchList.aspx?" + (qstring + "&psort=S") + "\">Language
wise</a></li> \n");
        sb.Append("      <li><a href=\"" + "SearchList.aspx?" + (qstring + "&psort=G") + "\">Gender
wise</a></li> \n");
        sortbtn.InnerHtml = sb.ToString();
        //objlog.Trace_Log(4, "ShoppingList", "1", "sortbtn", sb.ToString(), "Log");
    }
    // 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);

}

}

***** SendProfile.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/DOWMasterPage.master"
AutoEventWireup="true" CodeFile="SendProfile.aspx.cs" Inherits="SendProfile" %>

<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">

```

```
<div class=" col-lg-12 col-md-12 col=col-sm-12">
  <div class="row">
    <div id="dvMsgBox" class="span12 ordersave" runat="server" style="display: block;">
      <asp:Label ID="Label1" runat="server" Text="" Height="1px"></asp:Label>
    </div>
  </div>
</div>
</div>
</div>
</div>
```

```
</asp:Content>
```

```
<asp:Content ID="Content4" ContentPlaceHolderID="CPH_MainContent" Runat="Server">
```

```
<div class="content_holder">
```

```
<!-- 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">Profiles Sent</h6>
```

```
</div>
```

```
<div class="row">
```

```
<div id="NewProducts" class="new-product slider" runat="server">
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</section>
```

```
<!-- 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>
<script type="text/javascript" src="plugins/FlexSlider/jquery.flexslider-min.js"></script>
<script type="text/javascript" src="js/jquery.magnific-popup.min.js"></script>
<script type="text/javascript" src="js/magnific-popup.min.js"></script>
<script type="text/javascript" src="js/slick.js"></script>
<script type="text/javascript" src="js/slick-slider.js"></script>
<script type="text/javascript" src="js/custom.js"></script>
```

```
</div>
```

```
</asp:Content>
```

```
<asp:Content ID="Content5" ContentPlaceHolderID="ContentPlaceHolder1" Runat="Server">
```

```
</asp:Content>
```

```
<asp:Content ID="Content6" ContentPlaceHolderID="ContentFooter" Runat="Server">
```

```
  <!-- START FOOTER -->
```

```
  <footer class="footer" id="footer" runat="server">
```

```
    <div class="container">
```

```
      <div class="row">
```

```
      <section class="home-main-contant-style3 bg-white">
```

```
        <div class="parallax-block parallaxBg">
```

```
          <div class="container">
```

```
            <div class="parallax-block-text">
```

```
              <p>DateorWed.com - YourChoice <span class="color-green">We are a social networking  
site.</span> <br> Thank you! </p>
```

```
            </div>
```

```
          <div class="pre-btn-group">
```

```
            <a class="btn btn-product mr15" href="#">Search Now</a>
```

```
            <a class="btn btn-product mr15" href="#">Post a Profile</a>
```

```
          </div>
```

```
        </div>
```

```
      </div>
```

```
    </section>
```

```
  <!-- End Parallax -->
```

```
  <!-- START NEW -->
```

```
  <section class="home-main-contant-style3 bg-white" style="display:none">
```

```
    <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>
</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="fab fa-envelope-o"></i></button>
                    </span>
                </div>
            </div>
        </div>
    </div>
</div>

```

```

<div class="social-footer">
  <ul class="social-icons social-icons-simple">
    <li><a href="#"><i class="fab fa-facebook"></i></a></li>
    <li><a href="#"><i class="fab fa-twitter"></i></a></li>
    <li><a href="#"><i class="fab fa-tumblr"></i></a></li>
    <li><a href="#"><i class="fab fa-instagram"></i></a></li>
    <li><a href="#"><i class="fab fa-linkedin"></i></a></li>
  </ul>
</div>
</div>
</div>
</div>
</div>
<!--End Suvsubscribe -->
<!-- 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>

```

```
</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 an Agent 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="#">dating</a></li>
        <li><a href="#">Dating & Wedding</a></li>
        <li><a href="#">Wedding</a></li>
        <li><a href="#">Option</a></li>
        <li><a href="#">Dating or Wedding</a></li>
      </ul>
    </div>
  </div>
</div>
```

```
    </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="#">Dating <span class="pull-right">12</span></a></li>
      <li><a href="#">Wedding <span class="pull-right">11</span></a></li>
      <li><a href="#">Date or Wed <span class="pull-right">10</span></a></li>
      <li><a href="#">Date and Wed <span class="pull-right">14</span></a></li>
      <li><a href="#">Others <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 Profiles</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 Profiles</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="#">Looking for</a></h6>
```



```
        <div class="widget-posts-meta">Wed</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="#">Searching for</a></h6>
        <div class="widget-posts-meta">-Date-</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="#">Date or Wed</a></h6>
        <div class="widget-posts-meta">-Wed-</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:Content>
```

```
***** SendProfile.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 DOWEntity;
using System.IO;
public partial class SendProfile : System.Web.UI.Page
{
    DDLogTracer objlog = new DDLogTracer();
    bool IsMobile = false;
    protected void Page_Load(object sender, EventArgs e)
    {
        // string SrchStrng = "";
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedIn_Specs"];
        if (Clsusr == null)
        {
            bool ret = new Common().guest_login();
            if (ret == false)
            {
                Response.Redirect("~/Login.aspx");
            }
        }
        if (Clsusr != null)
        {
            string psid = (Request.QueryString["psid"]);
            if (psid != null)
            {
                psid = Common.Decrypt(psid);
                int pid = new tblUsers().getuserid(psid);
                EteUserShortList lst = new EteUserShortList();

                lst.UserId = Convert.ToInt16(Clsusr.UserId);
                lst.PSID = Convert.ToInt16(pid);
                lst.UId = Convert.ToInt16(Clsusr.UserId);

                lst.CreatedOn = DateTime.Today;
                lst.Active = true;
                bool result = new tblUsers().InsertSendProfile(lst, false);

            }
            { Bindetails(psid); }
        }
    }
    private void Bindetails(string SrchStrng)
    {
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedIn_Specs"]; ;
        int sno = 0;
    }
}

```

```

string cat = "";
string PreHtml = "";
StringBuilder sb = new StringBuilder();

cat = Common.Decrypt(Request.QueryString["cat"]).Trim();
// int len = cat.Length;
// cat = cat.Substring(0, len - 15);
// cat = cat.TrimEnd('\0');

// sb.Append(" <div class='directory-list'>");
Etetbluser_options opt = new Etetbluser_options();

//try
{
    List<EteGetSearchResults> Lst = (new
tblUsers()).GetSendProfilebyUserId(Convert.ToInt16(Clsusr.UserId));
    if (Lst.Count > 0)
    {
        var pst = (from c in Lst orderby c.UserId descending select c).FirstOrDefault();
        Lst=(from c in Lst orderby c.UserId descending select c).ToList();
        if (Lst != null && Lst.Count > 0)
        {

            //IList<EteGetSearchResults> Lst1 = new List<EteGetSearchResults>();

            foreach (var dt in Lst)
            {
                string path = "images/NoPhoto.png";
                string path1 = "images/No Photo.png";
                string href = "";
                string OutId = Common.Encrypt(dt.UserName.ToString());
                href = "ProfileDetail.aspx?id=" + OutId;
                var list = new tblUsers().getuserImagesbyuserid((int)dt.UserId, "").ToList();
                int cnt = list.Count();
                if (cnt > 1)
                {
                    path = list[0].ImgPath;
                    path1 = list[1].ImgPath;//"http://dateorwed.com/"+
                    if (!File.Exists(path))
                    {
                        path = list[0].ImageUrl;
                        path1 = list[1].ImageUrl;
                    }
                }
            }
            else
                if (list.Count() > 0)

```

```

        {
            path = list[0].ImageUrl;
        }

if (path == null || path == "")
{
    path = "images/Nocolor.png";
}
if (File.Exists(path))
{
    path = list[0].ImageUrl;
    path1 = list[1].ImageUrl;
}
if (path1 == null || path1 == "") { path1 = path; }

string martsttus = new tblUsers().getmatstatus(dt.Martialstatus);
string relgn = new tblUsers().getreligion(dt.Religion);
string caste = new tblUsers().getcaste(dt.Caste);
//slick code
sb.Append("        <div class='col-lg-3 col-md-4 item'>        \n");
sb.Append("        <div class='products grid-product'>        \n");
sb.Append("        <div class='product-grid-inner'>\n");
sb.Append("        <div class='product-grid-img'> \n");
sb.Append("        <ul class='product-labels'>        \n");
sb.Append("        <li>Date</li> \n");
sb.Append("        </ul> \n");
sb.Append("        <a href='ProfileDetail.aspx?id=" + OutId + "' 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'><a class="
href='ProfileDetail.aspx?id=" + OutId + "'>Detailed View</a></div>        \n");
sb.Append("        </div> \n");
sb.Append("        <div class='product-grid-caption text-center'> \n");
sb.Append("        <h4><a class='p-grid-title' href='ProfileDetail.aspx?id=" + OutId + "'> +
dt.FirstName + " " + dt.LastName + "</a></h4> \n");
//new code for details
sb.Append("        <p class='mb5'>" + "Gender: " + dt.Gender + "</p>        \n");
sb.Append("<p class='mb5'> DOB/Age: " + dt.dob.ToShortDateString() + "/" + dt.age + "<p>
\n");
sb.Append("        <p class='mb5'>" + "Sexual Orientation: " + dt.Sexornt + "</p> \n");
sb.Append("        <p class='mb5'>" + "Educational Qualification: " + dt.Eduqualification +
"</p> \n");
sb.Append("        <p class='mb5'>" + "Occupation: " + dt.Occupation + " in " +
dt.Occupationtype + "</p>        \n");

```

```

sb.Append(" <p class='mb5'>" + "Marital Status: " + martsttus + "</p> \n");
sb.Append(" <p class='mb5'>" + "Religion/Caste: " + relgn + "/" + caste + "</p>
\n");
sb.Append(" <p class='mb5'>" + "Location: " + dt.Location + "</p> \n");
if (dt.Active)
{
    if (dt.Sltype)
    {
        sb.Append(" <p class='mb5 fa fa-heart'>Your Profile has been accepted </p> \n");
    }
    else
    {
        sb.Append(" <p class='mb5 fa fa-heart-half'>Your Profile acceptance is still
pending</p> \n");
    }
}
else
{
    sb.Append(" <p class='mb5 fa fa-heart-broken'>Your Profile has been Rejected</p>
\n");
}
if (Convert.ToInt16(Clsusr.GroupId) > 3)
{
    if (!dt.ShortListAdded)
    {
        sb.Append(" <br> <i class='fa fa-heart'></i> \n");
        sb.Append(" <a href='MyShortList.aspx?psid=" + OutId + "'>Add to Shortlist</a><br>
\n");
    }
    if (!dt.HeartListAdded)
    {
        sb.Append("<br> <a href='MyHeartList.aspx?pid=" + OutId + "' class='btn btn-
product fa fa-heart'>Add to List</a> \n"); //<i class='fa fa-heart'>
    }
}

//new code for details ends
sb.Append(" </div> \n");
sb.Append(" </div> \n");
sb.Append(" </div> \n");
sb.Append(" </div> \n");
//grid code
sno++;
if (sno > 3)
{

```

```

        sno = 1;
    }

} //for each list

PreHtml = sb.ToString();
if (PreHtml.Length > 0)
{
    objlog.Trace_Log(4, "Dashboard", "1", "NewProducts", PreHtml, "Log".ToString());
    NewProducts.InnerHtml = PreHtml;
}
}
}
else
{
    //data not found
    //Common.ShowMessageHtml("Search Options or Data not found", 1, true, this.dvMsgBox);
}
}
// catch (Exception ex)
//{

//Common.ShowMessageHtml(ex.Message, 1, true, this.dvMsgBox);
// ExceptionLogging.SendExcepToDB(ex);

//}
}

}

***** 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.com</title>
<link rel="icon" href="images/DateOrWed.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 pt-1 pb-0">
  <a class="navbar-brand" href="Home.aspx"></a>
  <div class="row">
    <div class="col-lg-2 col-md-2 col-sm-12 pl-5">

      <asp:DropDownList id="DropDownList1" runat="server" Width="180px" Height="50px">

        <asp:ListItem Value="0">--- Select Option ---</asp:ListItem>
        <asp:ListItem Value="4"> Dating Search </asp:ListItem>
        <asp:ListItem Value="5"> Wedding Search </asp:ListItem>
        <asp:ListItem Value="6"> Friends Search </asp:ListItem>
        <asp:ListItem Value="7"> Dating and Wedding </asp:ListItem>
      </asp:DropDownList>
    </div>
    <div class="col-lg-4 col-md-4 col-sm-12 pr-5 pl-5 mr-5">
      <asp:TextBox ID="TextBox1" runat="server" CssClass="form-control search-slt" Width="361px"
Height="37px" MaxLength="50"
onkeydown="javascript:TextName_OnKeyDown(event)"></asp:TextBox>
    </div>
    <div class="col-lg-2 col-md-2 col-sm-12 pl-5">
      <asp:ImageButton ID="ImageButton1" runat="server" ImageUrl="images/services/Search2pg.jpg"
OnClick="ImageButton1_Click" Height="50px" />

```

```

    </div>
  </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-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> Cart</a>
    </li>
    <li class="nav-item">
      <a class="nav-link text-center" href="MyItinerary.aspx">
        <i class="fas fa-calendar-alt"></i><br> 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-->

```



```
<section class="buynsell pt-5 pb-3 mt-4">
  <div class="container-fluid">
    <div class="row">
      <div class="col-md-2" 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>
```

```
<a href='#menu406' class='list-group-item collapsed' data-toggle='collapse' data-parent='#sidebar' aria-
expanded='false'> <span class='hidden-sm-down'><i class='fas fa-vihara'></i> Dating Services <i
class='fas fa-chevron-right' style='font-size: 12px;'></i></span></a>
```

```
<div class='collapse' id='menu406'>
```

```
<a href='/ServicesList.aspx?sid=448' class='list-group-item' data-parent='#menu406'><i class='fas
'></i>Arrange Meeting</a>
```

```
<a href='/ServicesList.aspx?sid=449' class='list-group-item' data-parent='#menu406'><i class='fas
'></i>Accommodation</a>
```

```
<%--<a href='/ServicesList.aspx?sid=450' class='list-group-item' data-parent='#menu406'><i class='fas
'></i>Venues with Catering</a>
```

```
<a href='/ServicesList.aspx?sid=451' class='list-group-item' data-parent='#menu406'><i class='fas
'></i>High cost venues</a>
```

```
<a href='/ServicesList.aspx?sid=452' class='list-group-item' data-parent='#menu406'><i class='fas
'></i>Low cost venues</a>
```

```
<a href='/ServicesList.aspx?sid=453' class='list-group-item' data-parent='#menu406'><i class='fas
'></i>Banquet Halls</a>--%>
```

```
</div>
```

```
<a href='#menu404' class='list-group-item collapsed' data-toggle='collapse' data-parent='#sidebar'
aria-expanded='false'> <span class='hidden-sm-down'><i class='fas fa-pizza-slice'></i> Wedding Services
<i class='fas fa-chevron-right' style='font-size: 12px;'></i></span></a>
```

```
<div class='collapse' id='menu404'>
```

```
<a href='/ServicesList.aspx?sid=450' class='list-group-item' data-parent='#menu404'><i class='fas
'></i>Venues with Catering</a>
```

```
<a href='/ServicesList.aspx?sid=451' class='list-group-item' data-parent='#menu404'><i class='fas
'></i>High cost venues</a>
```

```
<a href='/ServicesList.aspx?sid=452' class='list-group-item' data-parent='#menu404'><i class='fas
'></i>Low cost venues</a>
```

```
<a href='/ServicesList.aspx?sid=453' class='list-group-item' data-parent='#menu404'><i class='fas
'></i>Banquet Halls</a>
```

```
<a href='/ServicesList.aspx?sid=443' class='list-group-item' data-parent='#menu404'><i class='fas
'></i>Catering</a>
```

```
<a href='/ServicesList.aspx?sid=444' class='list-group-item' data-parent='#menu404'><i class='fas
'></i>Cooking</a>
```

```
<a href='/ServicesList.aspx?sid=445' class='list-group-item' data-parent='#menu404'><i class='fas
'></i>Table arrangement</a>
```

```
</div>
```

```
<a href='#menu405' class='list-group-item collapsed' data-toggle='collapse' data-parent='#sidebar'
aria-expanded='false'> <span class='hidden-sm-down'><i class='fas fa-holly-berry'></i> Friends Meetings
<i class='fas fa-chevron-right' style='font-size: 12px;'></i></span></a>
<div class='collapse' id='menu405'>
<a href='/ServicesList.aspx?sid=446' class='list-group-item' data-parent='#menu405'><i class='fas
'></i>Arrange Reunion</a>
<a href='/ServicesList.aspx?sid=447' class='list-group-item' data-parent='#menu405'><i class='fas
'></i>Other Services</a>
</div>
<a href='#menu409' class='list-group-item collapsed' data-toggle='collapse' data-parent='#sidebar'
aria-expanded='false'> <span class='hidden-sm-down'><i class='fas fa-birthday-cake'></i> Shopping
Services <i class='fas fa-chevron-right' style='font-size: 12px;'></i></span></a>
<div class='collapse' id='menu409'>
<a href='/ServicesList.aspx?sid=459' class='list-group-item' data-parent='#menu409'><i class='fas
'></i>Wedding Cakes</a>
<a href='/ServicesList.aspx?sid=460' class='list-group-item' data-parent='#menu409'><i class='fas
'></i>Cakes & Pastries</a>
<a href='/ServicesList.aspx?sid=461' class='list-group-item' data-parent='#menu409'><i class='fas
'></i>Bakery Items</a>
</div>
<a href='#menu407' class='list-group-item collapsed' data-toggle='collapse' data-parent='#sidebar'
aria-expanded='false'> <span class='hidden-sm-down'><i class='fas fa-music'></i> Services <i class='fas
fa-chevron-right' style='font-size: 12px;'></i></span></a>
<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' style="display:none"> <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' style="display:none"> <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' style="display:none"> <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' style="display:none"> <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' style="display:none"> <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' style="display:none"> <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' style="display:none"> <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>International 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>
<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-7">
  <!--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>
    <li data-target="#carousel-example-1z" data-slide-to="3"></li>
    <li data-target="#carousel-example-1z" data-slide-to="4"></li>
    <li data-target="#carousel-example-1z" data-slide-to="5"></li>
    <li data-target="#carousel-example-1z" data-slide-to="6"></li>
    <li data-target="#carousel-example-1z" data-slide-to="7"></li>
    <li data-target="#carousel-example-1z" data-slide-to="8"></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>

```

```

<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="col-md-12">
        <div class="card mb-2">
          <div id="shopping" class="card-body text-center" runat="server">
            <h4 class="card-title font-weight-bold">Dating Profiles</h4>
            <p class="card-text">
              <a class="badge badge-primary"
href="SearchList.aspx?cat=4&SrchStrng=&fromage=21&toage=25">Ages 21-25</a>
              <a class="badge badge-primary"
href="SearchList.aspx?cat=4&SrchStrng=&fromage=26&toage=30">Ages 26-30</a>
              <a class="badge badge-primary" href="SearchList.aspx?cat=4&SrchStrng=&psort=R">Religion
wise</a>
              <a class="badge badge-primary" href="SearchList.aspx?cat=4&SrchStrng=&psort=c">Caste
wise</a>
              <a class="badge badge-primary" href="SearchList.aspx?cat=4&SrchStrng=&psort=A">Location
wise</a>
              <a class="badge badge-primary" href="SearchList.aspx?cat=4&SrchStrng=&psort=T">City
wise</a>
              <a class="badge badge-primary" href="SearchList.aspx?cat=4&SrchStrng=&psort=G">Gender
wise</a>

```

```
        <a class="badge badge-primary" href="SearchList.aspx?cat=4&SrchStrng=&psort=Y">Country
wise</a>
    <%--        <a class="badge badge-primary"
href="SearchList.aspx?cat=4&SrchStrng=&psort=M">Mothertongue wise</a>--%>
    <%--        <a class="badge badge-primary"
href="SearchList.aspx?cat=4&SrchStrng=&psort=S">Language wise</a>--%>
    </p>
    <a class="btn btn-secondary btn-md btn-rounded" href="SearchList.aspx?cat=4&SrchStrng="
>View Now</a>
    </div>
</div>
</div>
```

```
<div class="col-md-12">
    <div class="card mb-2">
        <div id="services" class="card-body text-center" runat="server">
            <h4 class="card-title font-weight-bold">Wedding Profiles</h4>
            <p class="card-text">
                <a class="badge badge-primary"
href="SearchList.aspx?cat=5&SrchStrng=&fromage=21&toage=25">Ages 21-25</a>
                <a class="badge badge-primary"
href="SearchList.aspx?cat=5&SrchStrng=&fromage=26&toage=30">Ages 26-30</a>
                <a class="badge badge-primary" href="SearchList.aspx?cat=5&SrchStrng=&psort=R">Religion
wise</a>
                <a class="badge badge-primary" href="SearchList.aspx?cat=5&SrchStrng=&psort=C">Caste
wise</a>
                <a class="badge badge-primary" href="SearchList.aspx?cat=5&SrchStrng=&psort=L">Location
wise</a>
                <a class="badge badge-primary" href="SearchList.aspx?cat=5&SrchStrng=&psort=T">City
wise</a>
                <a class="badge badge-primary" href="SearchList.aspx?cat=5&SrchStrng=&psort=G">Gender
wise</a>
                <a class="badge badge-primary" href="SearchList.aspx?cat=5&SrchStrng=&psort=Y">Country
wise</a>
            <%--        <a class="badge badge-primary"
href="SearchList.aspx?cat=5&SrchStrng=&psort=M">Mothertongue wise</a>--%>
            <%--        <a class="badge badge-primary"
href="SearchList.aspx?cat=5&SrchStrng=&psort=S">Language wise</a>--%>
            </p>
            <a class="btn btn-secondary btn-md btn-rounded"
href="SearchList.aspx?cat=5&SrchStrng=">View Now</a>
        </div>
    </div>
</div>
```

```
<div class="col-md-12">
    <div class="card mb-2">
        <div class="card-body text-center">
```

```
<h4 class="card-title font-weight-bold">Friends Profiles</h4>
<p class="card-text">
  <a class="badge badge-primary"
href="SearchList.aspx?cat=6&SrchStrng=&fromage=21&toage=25">Ages 21-25</a>
  <a class="badge badge-primary"
href="SearchList.aspx?cat=6&SrchStrng=&fromage=26&toage=30">Ages 26-30</a>
  <a class="badge badge-primary" href="SearchList.aspx?cat=6&SrchStrng=&psort=H">School
wise</a>
  <a class="badge badge-primary" href="SearchList.aspx?cat=6&SrchStrng=&psort=D">College
wise</a>
  <a class="badge badge-primary" href="SearchList.aspx?cat=6&SrchStrng=&psort=L">Location
wise</a>
  <a class="badge badge-primary" href="SearchList.aspx?cat=6&SrchStrng=&psort=T">City
wise</a>
  <a class="badge badge-primary" href="SearchList.aspx?cat=6&SrchStrng=&psort=G">Gender
wise</a>
  <a class="badge badge-primary" href="SearchList.aspx?cat=6&SrchStrng=&psort=Y">Country
wise</a>
<%-- <a class="badge badge-primary"
href="SearchList.aspx?cat=6&SrchStrng=&psort=M">Mothertongue wise</a>--%>
<%-- <a class="badge badge-primary"
href="SearchList.aspx?cat=6&SrchStrng=&psort=S">Language wise</a>--%>
</p>
  <a class="btn btn-secondary btn-md btn-rounded"
href="SearchList.aspx?cat=6&SrchStrng=">View Now</a>
</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>
```

```
                <div><a href="ServicesList.aspx?cid=414" 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">Videographer</h6>
                <div class="mb-4"></div>
                <a href="#">By: ABC</a>
                <div><a href="ServicesList.aspx?cid=415" class="btn btn-secondary btn-sm btn-
rounded">Get Best Price</a></div>
            </div>
        </div>
    </div>
</div>
</div>
</div>
</div>
</div> </div>
</section>
```

```
<section class="boxx-1 pb-3">
    <div class="container-fluid">
        <div class="card">
            <h3 class="h3-responsive font-weight-bold pb-0 mb-0 pl-3 pt-1">Venue (Function Hall)</h3>
            <div class="p-3">
                <div class="row">
                    <div class="col-md-3">
                        <div class="text-white card1img">
                            <div class="cardlist">
                                <ul>
<a href='/ServicesList.aspx?sid=448' class='list-group-item'><i class='fas '></i>A/c Venues</a>
<a href='/ServicesList.aspx?sid=449' class='list-group-item'><i class='fas '></i>Open Air Venues</a>
<a href='/ServicesList.aspx?sid=450' class='list-group-item' ><i class='fas '></i>Venues with Catering</a>
<a href='/ServicesList.aspx?sid=451' class='list-group-item' ><i class='fas '></i>High cost venues</a>
<a href='/ServicesList.aspx?sid=452' class='list-group-item' ><i class='fas '></i>Low cost venues</a>
<a href='/ServicesList.aspx?sid=453' class='list-group-item' ><i class='fas '></i>Banquet Halls</a>
<!--
                <li><a href="#">Banquet / Wedding Hall</a></li>
                <li><a href="#">Board Meeting Hall</a></li>
                <li><a href="#">Merriage Garden</a></li>
                <li><a href="#">Theater Type Hall</a></li>--%>
            <!--<br>--%>
        </div>
    </div>
</div>
```

```

                <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 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>
<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 class="col-md-4">
    <div class="card1 p-2">
        <div class="row">
            <div class="col-md-4">
                
            </div>

```

```
    </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>
```

```
<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">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 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">
```



```
        <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> info@dateorwed.com</p>
            <p>
                <i class="fas fa-phone mr-3"></i> +91 - 9876543210</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"> dateorwed.com</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() {

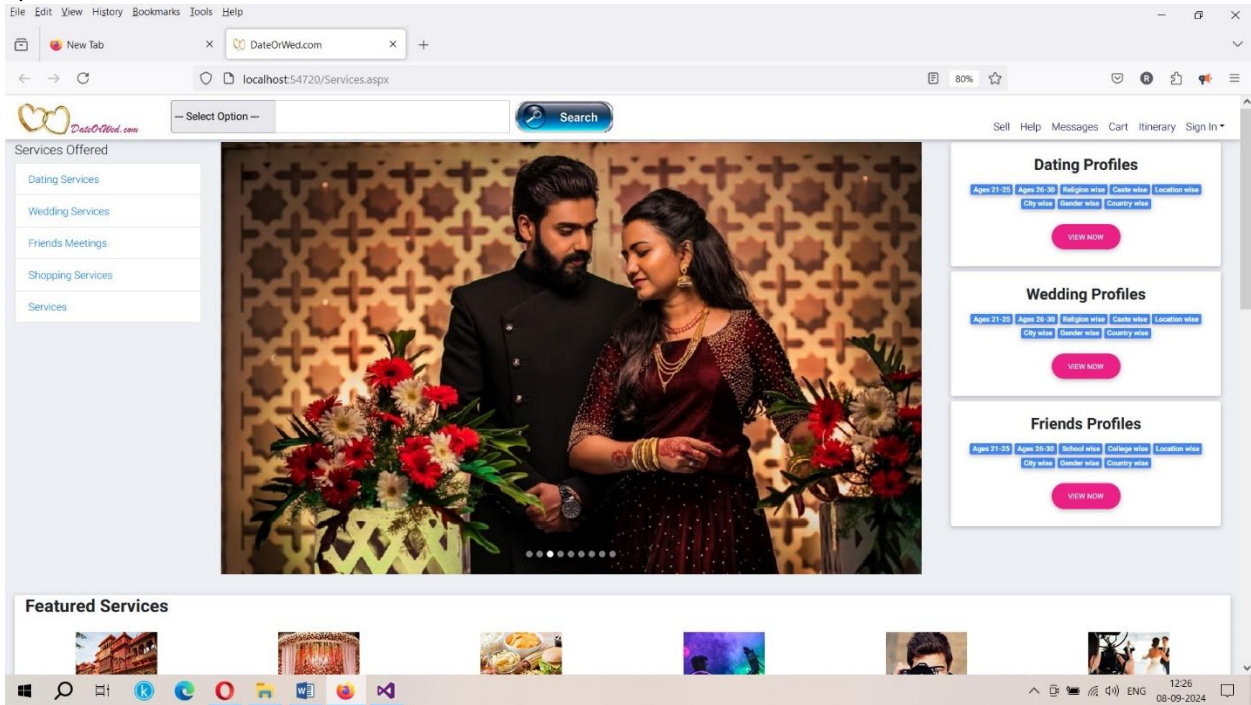
  var $btn = $('#btnTop');
  var $home = $('#topSection');
  var startpoint = $home.scrollTop() + $home.height();

  $(window).on('scroll', function() {
    if($(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 DOWEntity;
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)
{
    bool ret = new Common().guest_login();
    if (ret == false)
    {
        Response.Redirect("~/Login.aspx");
    }
}

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, "MyLaadBazar", "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();
    IPEndPoint 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("MyLaadBazarCookie");
DateTime now = DateTime.Now;
myCookie.Value = objLoggedInUser.UserName;
myCookie.Expires = now.AddYears(1);

Response.Cookies.Add(myCookie);

}
}
protected void ImageButton1_Click(object sender, ImageClickEventArgs e)
{
//ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedUser_Specs"]; ;

//string cat = ddl_location.SelectedValue.ToString();//
Common.Encrypt(ddl_location.SelectedValue.ToString());
//string srch = Common.Encrypt(txt_SearchString.Text);
//Response.Redirect("~/SearchList.aspx?pageid=198&SrchStrng=" + srch + "&cat=" + cat);

}
}
}
***** Startup.cs *****
using System;
using System.Collections.Generic;

```

```
using System.Linq;
using System.Web;
using Microsoft.Owin;
using Owin;
```

```
[assembly: OwinStartup(typeof(SignalRChat.Startup))]
namespace SignalRChat
```

```
{
    public class Startup
    {
        public void Configuration(IAppBuilder app)
        {
            app.MapSignalR();
        }
    }
}
```

```
***** UserChoices.aspx*****
```

```
<%@ Page Title="DateorWed - Your Search Choices" Language="C#"
MasterPageFile="~/DOWMasterPage.master" AutoEventWireup="true" CodeFile="UserChoices.aspx.cs"
Inherits="UserChoices" %>
```

```
<%@ 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">
```

```
</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 -->

<div class="container bg-gray">
  <div class="login-area mb30">
<section class="main-contain dip-style col-lg-2 col-md-2 col-sm-6">
  <div class="login-header">
    <div class="login-details ">
      <ul class="nav nav-tabs navbar-left">
        <li class="active col-sm-12"><a data-toggle="tab"
href="#Choices">Settings</a></li>
        <li class=" col-sm-12"><a data-toggle="tab" href="#Religion"> Religion</a></li>
        <li class="col-sm-12"><a data-toggle="tab" href="#Caste"> Caste</a></li>
        <li class="col-sm-12"><a data-toggle="tab" href="#Area">Area</a></li>
        <li class="col-sm-12"><a data-toggle="tab" href="#City">City</a></li>
        <li class="col-sm-12"><a data-toggle="tab" href="#Country">Country</a></li>
        <li class="col-sm-12"><a data-toggle="tab"
href="#Languages">Languages</a></li>
        <li>
        </li>
      </ul>
    </div>
  </div>

</section>
  <section class="main-contain dip-style col-lg-10 col-md-10 col-sm-6">
    <div class="tab-content">
      <div class="tab-pane fade in active" id="Choices">
        <div class="row-fluid">
          <div id="Div1" runat="server" class=" col-lg-8 col-md-8 col-sm-12 " span3="" style="">
            <asp:Button ID="Button1" runat="server" CssClass="btn btn-primary btn-md btn-rounded"
Text="Save Settings" OnClick="Button1_Click" />
          <div class="row-fluid">
            <h3>Choices/Settings</h3></div>
          <table id="Table1" class="table table-striped table-detail col-lg-8 col-md-8 col-sm-12 bg-secondary"
runat="server" visible="true">
            <tbody>
              <tr>
                <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Searching for :</td>
                <td>
                  <asp:DropDownList ID="ddl_choice" runat="server">
                    <asp:ListItem Value="0">-- Select Your Choice --</asp:ListItem>
                    <asp:ListItem Value="1">Women</asp:ListItem>
                    <asp:ListItem Value="2">Men</asp:ListItem>
                    <asp:ListItem Value="3">Bi-Sexuals</asp:ListItem>
                    <asp:ListItem Value="4">Others</asp:ListItem>
                  </asp:DropDownList>
                </td>
              </tr>
            </tbody>
          </table>
        </div>
      </div>
    </div>
  </section>

```



```

        </tr>
    <tr>
    <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Age Search (Between) :</td>

    <td>
    <asp:TextBox ID="Age_min" runat="server" MaxLength="100" Width="150px"
Text="21"></asp:TextBox>
    <asp:TextBox ID="Age_max" runat="server" MaxLength="100" Width="150px"
Text="25"></asp:TextBox>

    </td>
    </tr>
    <tr>
    <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Religion Search:</td>

    <td>
    <asp:DropDownList ID="ddl_Religion" runat="server">
    </asp:DropDownList>

    </td>
    </tr>
    <tr>
    <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Caste Search:</td>

    <td>
    <asp:DropDownList ID="ddl_caste" runat="server">
    </asp:DropDownList>

    </td>
    </tr>
    <tr>
    <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Area Search :</td>

    <td>
    <asp:DropDownList ID="ddl_area" runat="server">
    </asp:DropDownList>

    </td>
    </tr>
    <tr>
    <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">City Search :</td>

    <td>
    <asp:DropDownList ID="ddl_city" runat="server">
    </asp:DropDownList>

    </td>
    </tr>
    </tbody>
</table>
</%--

```

```

        </tr>
        <tr>
        <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Geolocation :</td>

        <td>
        <asp:TextBox ID="Geolocation" runat="server" MaxLength="100"
Width="250px"></asp:TextBox>

        </td>
        </tr>
        <tr>
        <td align="right" class="label_txt"><strong>Country Search:</strong></td>
        <td width="25%" align="left">
        <asp:DropDownList ID="ddl_countrysearch"
runat="server"></asp:DropDownList></td>

        </tr>--%>

        <tr>
        <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Search by Mother Tongue :</td>

        <td>
        <asp:DropDownList ID="ddl_mtonguesearch" runat="server"></asp:DropDownList>

        </td>
        </tr>
<%--
        <tr>
        <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Search by Languages Known :</td>

        <td>
        <asp:DropDownList ID="ddl_languages" runat="server"></asp:DropDownList>

        </td>
        </tr> --%>

</tbody>

</table>

</div>
</div>
</div>
<!-- Description -->
<div class="tab-pane media-block" id="Religion">
    <div class="col-lg-10 col-md-10 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;">

```

```

        <asp:Button ID="btn3" runat="server" CssClass="btn btn-primary" Text="Save Login
Info" OnClick="btn3_Click"/>
        <h3>Religions to Search</h3></div>
    </div>

    <div class="row-fluid">
        <div class="col-lg-8 col-md-8 col-sm-12 detailview" style="background-color: #ffffff;
padding: 15px 5px;"> <!--border-top: none;border: 0px solid #eefef;-->

        <!--<form class="form-horizontal">-->
        <div class="control-group">
            <table class="table table-striped table-detail col-lg-8 col-md-8 col-sm-12 bg-
secondary">

                <tbody>
                </tbody>

            </table>

        </div>

    </div>

    </div>

    </div>
</div>

    </div>
<!-- End Description -->

<!-- Profile -->
<div id="Caste" class="tab-pane">
    <div class="col-lg-10 col-md-10 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;">
                    <asp:Button ID="Button4" runat="server" CssClass="btn btn-primary" Text="Save
Profile" OnClick="Button4_Click"/>
                    <h3>Castes to Search</h3></div>
                </div>
                <div class="row-fluid"> <!--table-detail-->
                    <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 col-lg-8 col-md-8 col-sm-12">
                            <tbody>

```

```

</tbody>

        </table>

    </div>
    <!--Bank Account END-->
</div>

</div>
</div>

</div>
<div class="tab-pane media-block" id="Area">
    <div class="row-fluid">
        <div id="addlist" runat="server" class=" col-lg-12 col-md-12 col-sm-12 " span3=""
style=""><h3>Areas to Search</h3></div>
        <asp:Button ID="btn_SaveAddress" runat="server" CssClass="btn btn-primary btn-md btn-
rounded" Text="Save Address" Enabled="false" 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;">

        </div>

    </div>

    </div>
<!-- End Description -->
</div>
<div class="tab-pane media-block" id="City">
    <div class=" col-lg-12 col-md-12 col-sm-12 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;"><h3>Cities to
Search</h3></div>

```

```

</div>
    <div class="row-fluid">
        </div>
    </div>

    <!--Notes Ends----->
    </div>
    </div>
    <!-- End Description -->
    </div>

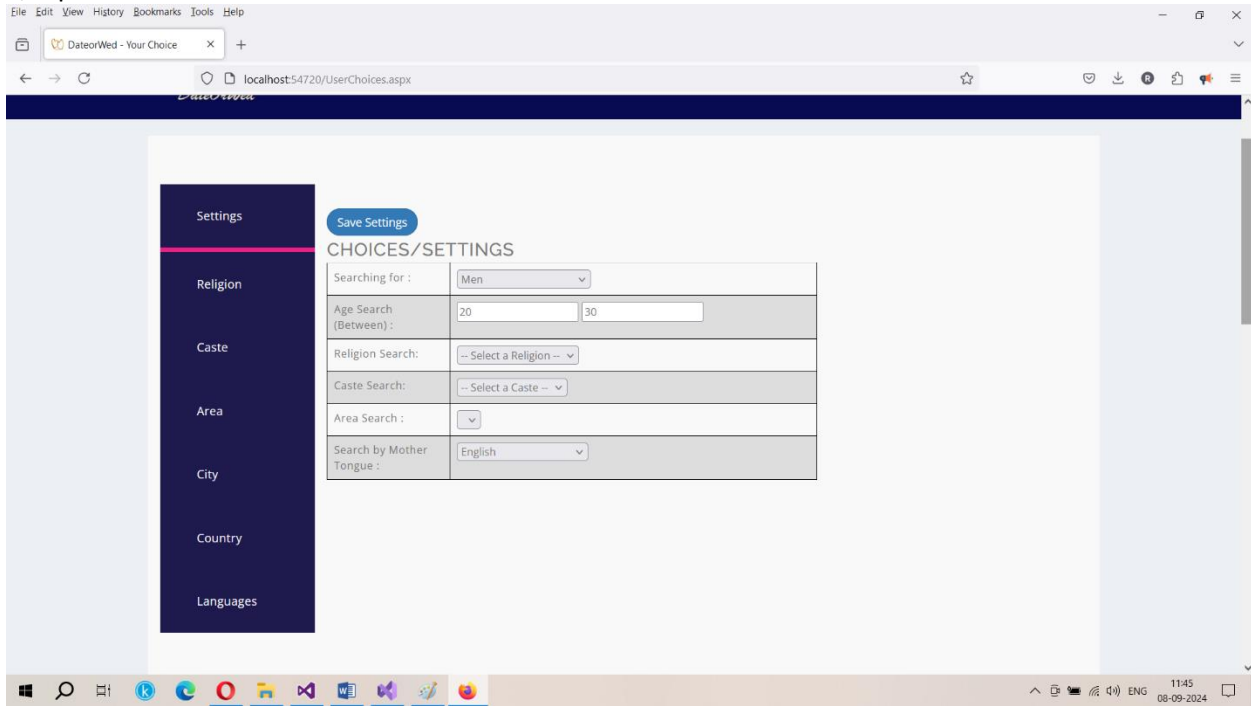
</div>

```

```

</section> </div></div>
</asp:Content>
<%--<asp:Content ID="Content6" ContentPlaceHolderID="ContentFooter" Runat="Server">
</asp:Content>--%>

```



*****UserChoices.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 DOWEntity;
using DOWDal;
using System.Data;
using System.Data.SqlClient;
using System.Configuration;
using System.Drawing;
using System.Web.Security;
using System.IO;
using System.Net;

public partial class UserChoices : System.Web.UI.Page
{
    public string PageType;
    string pageid;
    int UserId;
    int PrId = 0;
    string user;
    DDLogTracer objlog = new DDLogTracer();

    protected void Page_Load(object sender, EventArgs e)
    {
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr == null)
        {
            bool ret = new Common().guest_login();
            if (ret == false)
            {
                Response.Redirect("~/Login.aspx");
            }
        }

        if (Clsusr != null)
        {
            PageType = Page.Request.QueryString["PageType"];
            if (PageType == null)
            {
                PageType = "Edit";
            }

            Session["PageType"] = PageType;
            pageid = "4";
            if (Convert.ToInt16( Clsusr.UserId) <5)
            {
                Common.ShowMessageHtml("Pl Login to View your Profile-"+Clsusr.UserName+"-
"+Clsusr.UserId+"-"+Clsusr.GroupId, 0, true, dvMsgBox);
            }
        }
    }
}

```

```

    return;
}
if (!this.IsPostBack)
{
    PopDlls ObjPD = new PopDlls();
    //ObjPD.BindDropDowns(ddl_legalstr, (new PopDlls()).GetLegalstr());
    ObjPD.BindDropDowns(ddl_Religion, (new PopDlls()).GetReligions());
    ObjPD.BindDropDowns(ddl_caste, (new PopDlls()).GetCastes());
    ObjPD.BindDropDowns(ddl_mtonguesearch, (new PopDlls()).GetLanguages());
    //ObjPD.BindDropDowns(ddl_mtongue, (new PopDlls()).GetLanguages());
    UserId = Convert.ToInt16(Clsusr.UserId);
    if (UserId > 0)
    {
        //-----user infor
        Etetbluser_options opt = new tblUsers().getuserchoies(UserId);
        if (opt != null)
        {
            ddl_choice.SelectedValue = opt.Gender_Search;
            Age_min.Text = opt.Age_from.ToString();
            Age_max.Text = opt.Age_to.ToString();
            ddl_Religion.SelectedIndex = opt.Religion_search;
            ddl_caste.SelectedIndex = opt.Caste_search;
            ddl_area.SelectedIndex = opt.Area_search;
            //ddl_city.SelectedIndex=opt.city
            //ddl_countrysearch
            ddl_mtonguesearch.SelectedIndex = opt.MTongue_search;
            //ddl_languages.SelectedIndex=opt.

        }
        else
        {
            ddl_choice.SelectedIndex = 0;
            Age_min.Text = "21";
            Age_max.Text = "25";
            ddl_Religion.SelectedIndex = 0;
            ddl_caste.SelectedIndex = 0;
            ddl_area.SelectedIndex = 0;

        }
        List<Etetbl_moptions> mopt = new List<Etetbl_moptions>();
        foreach (var row in mopt)
        {

        }

    }
    StringBuilder sb = new StringBuilder();

```

```

        //download(dt);
    }
}
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(Common.Encrypt(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 = new
tblUsers().getusername(Convert.ToInt16(objLoggedInUser.UserId));
        //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"]);
        objLoggedInUser.FirstName = status_login.Rows[0]["FirstName"].ToString();
        objLoggedInUser.LastName = status_login.Rows[0]["LastName"].ToString();
        Session["LoggedUser_Specs"] = objLoggedInUser;

        FormsAuthentication.SetAuthCookie(objLoggedInUser.UserName, false);
        HttpCookie myCookie = new HttpCookie("DOWCookie");
        DateTime now = DateTime.Now;
        myCookie.Value = objLoggedInUser.UserName;
        myCookie.Expires = now.AddYears(1);
    }
}

```



```

        Response.Cookies.Add(myCookie);
    }
}
private DataTable GetData(SqlCommand cmd)
{
    DataTable dt = new DataTable();
    String strConnString =
System.Configuration.ConfigurationManager.ConnectionStrings["DOWConectingString"].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();
    }
}
protected void SaveAdd_Click(object sender, EventArgs e)
{
}
private void ContentSaveControlClicked()
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    UserId = Convert.ToInt32(Clsusr.UserId);
    bool Result = false;
    try
    {
        if (Result == true)
        {
            Common.ShowMessageHtml("Enrer you interests", 0, true, dvMsgBox);

```

```

    }

    else
    {
        Etetbluser_profile Prd = new Etetbluser_profile();
        //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

```

```

    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"]; ;
}
protected void Button2_Click(object sender, System.EventArgs e)
{
}

protected void btn_addaddress_Click(object sender, EventArgs e)
{
}

protected void btn3_Click(object sender, EventArgs e)
{//update tbluser
    objlog.Trace_Log(0, "DateOrWed", "MangeUser", "ContentSaveControlClicked", "Start Here..");
    ClsLoggedInUser ClsUser = (ClsLoggedInUser)Session["LoggedInUser_Specs"];

    try
    {

    }

    catch (Exception exc)
    {
        objlog.Trace_Error(4, "DateOrWed", "MangeUser", "ContentSaveControlClicked", "Exception:" +
exc.ToString());
        Common.ShowMessageHtml(exc.ToString(), 0, true, dvMsgBox);

    }
    objlog.Trace_Log(0, "DateOrWed", "MangeUser", "ContentSaveControlClicked", "Start Here..");
}
protected void Button1_Click(object sender, EventArgs e)
{
    //save settings or choices
    Etetbluser_options opt = new tblUsers().getuserchoies(UserId);
    if (opt == null)
    {
        opt = new Etetbluser_options();
    }
    {
        opt.Gender_Search = ddl_choice.SelectedValue;
        opt.Age_from = Convert.ToInt16(Age_min.Text);
        opt.Age_to = Convert.ToInt16(Age_max.Text);
        opt.Religion_search = ddl_Religion.SelectedIndex;
        opt.Caste_search = ddl_caste.SelectedIndex;
        opt.Area_search = ddl_area.SelectedIndex;
        opt.MTongue_search = ddl_mtonguesearch.SelectedIndex;
        bool res = new tblUsers().UpdateUser_options(opt);
    }
}

```

```

}
protected void Button4_Click(object sender, EventArgs e)
{//update profile details
    bool Result = false;
    Result = new tblUsers().UpdatePass();
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"]; ;
    UserId = Convert.ToInt32(Clsusr.UserId);
    string updooption = "Edit";
    try
    {
        if (Result == true)
        {
            if (updooption == "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..");
}
}

```

***** UserDetail.aspx*****

```

<%@ Page Title="DateorWed - Your Profile" Language="C#"
MasterPageFile="~/DOWMasterPage.master" AutoEventWireup="true" CodeFile="UserDetail.aspx.cs"
Inherits="UserDetail" %>

```

```

<%@ Register Assembly="Infragistics4.Web.v14.2, Version=14.2.20142.1028, Culture=neutral,
PublicKeyToken=7dd5c3163f2cd0cb" Namespace="Infragistics.Web.UI.EditorControls" TagPrefix="ig" %>

```

```
<%@ Register Assembly="AjaxControlToolkit" Namespace="AjaxControlToolkit" TagPrefix="cc1" %>
```

```
<asp:Content ID="Content1" ContentPlaceHolderID="CPH_MainMaster_Header" Runat="Server">  
    <script type="text/javascript" src="js/maps.js"></script>  
<script src="https://js.api.here.com/v3/3.1/mapsjs-core.js"  
    type="text/javascript" charset="utf-8"></script>  
<script src="https://js.api.here.com/v3/3.1/mapsjs-service.js"  
    type="text/javascript" charset="utf-8"></script>  
<%-- var platform = new H.service.Platform({ 'apikey':  
'sRtQ7lqu4sVI4RaHYe089ukcoH5i9R1AdOeQWB7yXto'}));  
    var defaultLayers = platform.createDefaultLayers();  
var map = new H.Map(  
    document.getElementById('mapContainer'),  
    defaultLayers.vector.normal.map,  
    {  
        zoom: 10,  
        center: { lat: 17.3850, lng: 78.4867 }pa  
    });--%>  
<script>  
    // Initialize and add the map  
    function initMap() {  
        // The location of Uluru  
        var uluru = { lat: 17.3850, lng: 78.4867 };  
        //var uluru="9FRC+R4 Hyderabad, Telangana";  
        // The map, centered at Uluru  
        var map = new google.maps.Map(  
            document.getElementById('map'), { zoom: 4, center: uluru });  
        // The marker, positioned at Uluru  
        var marker = new google.maps.Marker({ position: uluru, map: map });  
    }  
</script>  
<link href="CSS/PopupModal.css" rel="stylesheet" type="text/css" />  
<script type="text/javascript">  
    function Hidepopup() {  
        //alert('Hidepopup()');  
        $find("popup").hide();  
        //$find("pnlAddEdit").hide();  
  
        return false;  
    }  
    function Hidepopup1() {  
        alert('Hidepopup1()');  
        $find("popup").hide();  
        //$find("pnlAddEdit").hide();  
        return false;  
    }  
}
```

```

</script>
<script>
$(function () {
    $("#popup").dialog({
        resizable: false,
        autoOpen: false,
        height: 350,
        modal: true,
        buttons: {
            "OK": function () {
                $(this).dialog("close");
            },
            Cancel: function () {
                $(this).dialog("close");
            }
        }
    });

    $("#InkFake").click(function () {
        $("#popup").dialog("open");
    });
});
</script>

```

```

</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: 1px;">
        <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="Content5" ContentPlaceHolderID="CPH_MainContent" Runat="Server">
    <!-- Content Bodey Here -->

    <section class="main-contain ">
        <div class=" container bg-gray">
            <div class="login-area mb10">

```

```

        <div class="login-header">
        <div class="d-block d-d-none d-sm-block d-md-none " id="navbutton" runat="server">
        <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="fa fa-bars "></span>
        </button>
        </div>
<div class="collapse navbar-collapse " id="navbarSupportedContent-4">
        <ul class="nav nav-tabs navbar-right col-sm-12 ">
        <li class="active sm-12"><a data-toggle="tab" href="#description"> Info</a></li>
        <li class="sm-12"><a data-toggle="tab" href="#Profile"> Details</a></li>
        <li class="sm-12"><a data-toggle="tab" href="#Choices">Settings</a></li>
        <li class="sm-12"><a data-toggle="tab" href="#Address">Address</a></li>
        <li class="sm-12"><a data-toggle="tab" href="#Images">Images</a></li>
        <li class="sm-12"><a data-toggle="tab" href="#Maps">Maps</a></li>
        <li>
        </li>
        </ul>
        </div>
        </div>
<div class="container-fluid">
        <div class="row-fluid" style="padding-top: 1px;">
        <div class="col-sm-3" id="dvlogo" runat="server" style="display: none;">
        </div>
        <div class="col-lg-10">
        <div class="row">
        <div id="dvMsgBox" class="col-lg-10 ordersave" runat="server" style="display: none;">
        <asp:Label ID="Label1" runat="server" Text=""></asp:Label>
        </div>
        </div>
        </div>
</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-10 " style="background-color: #f8f9fa;">
        <div class="row-fluid">
        <div class="col-lg-8 col-md-8 col-sm-10" style="text-align: left;">
        <asp:Button ID="btn3" runat="server" CssClass="btn btn-primary" Text="Save Login
Info" OnClick="btn3_Click"/>
        <h3>LOGIN INFORMATION</h3>(Private Information used in Search only)</div>
        </div>
        <div class="row-fluid ">

```

```

        <div class=" detailview" style="background-color: #ffffff; padding: 15px 15px;" <%-
border-top: none;border: 0px solid #eefef;--%>
        <!--col-lg-12 col-md-12 col-sm-12--%>
        <!--<form class="form-horizontal">-->
        <div class="control-group">
            <table class="table table-striped table-detail bg-secondary"> <!--col-lg-12 col-
md-12 col-sm-12--%>
                <tbody>
                    <tr>
                        <td class=" col-lg-3 col-md-3 col-sm-3"><strong>Email/Mobile/User Id
:*</strong></td>
                            <td class="auto-style1 col-lg-6 col-md-6 col-sm-6">
                                <asp:TextBox ID="txt_username" runat="server" MaxLength="100"
></asp:TextBox> <!--TextMode="Email"--%>
                                    </td>
                                </tr>
                                <tr>
                                    <td ><strong>First Name :*</strong></td>
                                    <td > <!--class="col-lg-6 col-md-6 col-sm-6"--%>
                                        <asp:TextBox ID="txt_FirstName" runat="server" MaxLength="100"
onkeypress="return validatealpha(event)" CssClass="col-lg-6 col-md-6 col-sm-8 pull-
left"></asp:TextBox>
                                            </td>
                                        </tr>
                                        <tr>
                                            <td class=" col-lg-3 col-md-3 col-sm-3 align-right"><strong>Last Name
:*</strong></td>
                                                <td class="align-left col-lg-6 col-md-6 col-sm-6">
                                                    <asp:TextBox ID="txt_LastName" runat="server" MaxLength="100"
onkeypress="return validatealpha(event)" CssClass="col-lg-6 col-md-6 col-sm-8 pull-
left"></asp:TextBox>
                                                        </td>
                                                    </tr>
                                                    <tr>
                                                        <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Your Gender :*</td>
                                                        <td>
<asp:DropDownList ID="ddl_gender" runat="server">
                    <asp:ListItem Value="0">-- Select Your Gender --</asp:ListItem>
                    <asp:ListItem Value="M">Male</asp:ListItem>
                    <asp:ListItem Value="W">Women</asp:ListItem>
                    <asp:ListItem Value="T">Trans</asp:ListItem>
                    <asp:ListItem Value="O">Other</asp:ListItem>
                </asp:DropDownList>
                </td>
            </tr>
        </tbody>
    </table>

```



```

        </tr>

        <tr>
            <td align="right" class=" col-lg-3 col-md-3 col-sm-3 align-
right"><strong>Country:</strong></td>
            <td align="left" class="align-right">
                <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>Email id
:*</strong></td>
            <td align="left">
                <asp:TextBox ID="email_name" runat="server" MaxLength="50" TextMode="Email"
Visible="true" Enabled="true"></asp:TextBox>
            </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>Choice
:*</strong></td>
            <td width="25%" align="left">
                <asp:DropDownList ID="ddl_GroupId" runat="server"
Enabled="true"></asp:DropDownList>
            </td>
        </tr>

```

```

        <tr>
            <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Monther Tongue :</td>
            <td>
                <asp:DropDownList ID="ddl_mtongue" runat="server"
Enabled="true"></asp:DropDownList>
            </td>
        </tr>
        <tr hidden="hidden">
            <td width="17%" 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>
                <%-- <asp:RegularExpressionValidator ID="RegularExpressionValidator1" runat="server"
ErrorMessage="RegularExpressionValidator"
ValidationExpression="^\.(?=.{8,})(?=\.*\d)(?=\.*[a-z])(?=\.*[A-Z])(?=\.*[!*@#$$%^&+=]).*$"
ControlToValidate="txt_Password">
                </asp:RegularExpressionValidator>--%>
            </td>
        </tr>
        <tr>
            <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Universal Number :</td>
            <td>
                <asp:TextBox ID="txt_adhaar_id" runat="server" MaxLength="20"
></asp:TextBox>
            </td><td hidden="hidden">
                <asp:CheckBox ID="adhaar_id_chk"
runat="server" Checked="true" />
            </td>
        </tr>
        <tr>
            <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Pan Number :</td>
            <td>
                <asp:TextBox ID="txt_pan_no" runat="server" MaxLength="10"
></asp:TextBox>
            </td><td hidden="hidden">
                <asp:CheckBox ID="pan_no_chk"
runat="server" Checked="true"/>
            </td>
        </tr>
        <tr hidden="hidden">
            <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>
            </td>
        </tr>

```

```

<tr><td class="pull-right">
<asp:CheckBox ID="currency_chk" runat="server" Text=" Click to Authorise" />
  </td>
  <td ><i class="fa fa-check"></i> I authorise to use Date of Birth, Date of Time and Place of birth to be
used in Matching Matches using various methods like Numerology, Astrology, etc.</td>

</tr>
</tbody>
<tr><td class="pull-right">
<asp:CheckBox ID="ad_chk" runat="server" Text=" Click to Allow" />
  </td>
  <td><i class="fa fa-check"></i> I authorise to use Public / authorised Photos and Information to use
for Advertisement and to Post on the site.(Please send an email to confirm the same)</td>

</tr>
</table>

</div>

</div>

</div>

</div>

</div>
<!-- End Description -->
<!-- Profile -->
<div id="Profile" class="tab-pane">
  <div class="col-lg-12 col-md-12 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;">
          <asp:Button ID="Button4" runat="server" CssClass="btn btn-primary" Text="Save
Profile" OnClick="Button4_Click"/>
          <h3>Public Profile</h3><span> (Click Checkbox to make Private)</span></div>
        </div>
        <div class="row-fluid"> <!--table-detail-->
          <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 col-lg-8 col-md-8 col-sm-12">
              <tbody>
                <tr>
                  <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Your 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">Your Interests :</td>
<td>
<asp:TextBox ID="Details" runat="server" Height="100px" Rows="7"
style="resize:both" TextMode="MultiLine" CssClass="col-lg-12 col-md-12 col-sm-12 col-xs-12 pull-
left"></asp:TextBox>
</td><td>
<asp:CheckBox ID="Details_chk" runat="server" />
</td>

</tr>
<tr>
<td class="col-lg-3 col-md-3 col-sm-3" dir="auto">About Yourself :</td>
<td >
<asp:TextBox ID="txt_about" runat="server" Height="100px" Rows="7"
style="resize:both" TextMode="MultiLine" CssClass="col-lg-12 col-md-12 col-sm-12 col-xs-12 pull-
left"></asp:TextBox>
</td>
<td>
<asp:CheckBox ID="about_chk" runat="server" />
</td>

</tr>
<tr>
<td class="col-lg-3 col-md-3 col-sm-3 auto-style1"
dir="auto">Occupation</td>
<td class="auto-style2">
<asp:TextBox ID="txt_Occupation" runat="server" MaxLength="40"
CssClass="col-lg-9 col-md-9 col-sm-10 pull-left"></asp:TextBox>
</td><td>
<asp:CheckBox ID="Occupation_chk" runat="server" />
</td>
</tr>
<tr>
<td class="col-lg-3 col-md-3 col-sm-3 auto-style1" dir="auto">Area of
Occupation</td>
<td class="auto-style2">
<asp:TextBox ID="txt_Occupation_type" runat="server" MaxLength="40"
CssClass="col-lg-9 col-md-9 col-sm-10 pull-left"></asp:TextBox>
</td><td>
<asp:CheckBox ID="Occupation_type_chk" runat="server" />
</td>
</tr>
<tr>
<td class="col-lg-3 col-md-3 col-sm-3 auto-style1" dir="auto">Education
Qualification</td>

```

```

                <td class="auto-style2">
                    <asp:TextBox ID="txt_Edu_qualification" runat="server" MaxLength="40"
                    CssClass="col-lg-9 col-md-9 col-sm-10 pull-left"></asp:TextBox>
                </td><td>
                    <asp:CheckBox ID="Edu_qualification_chk" runat="server" />
                </td>
            </tr>
            <tr>
            <tr>
                <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">School Attended:</td>
                <td>
                    <asp:DropDownList ID="ddl_school" CssClass="show" runat="server">
                        </asp:DropDownList>
                    <asp:TextBox ID="txt_school_year" runat="server"></asp:TextBox>
                </td><td> <asp:CheckBox ID="school_chk" runat="server" />
            </td>
            <td>
                <asp:LinkButton ID="InkFake" runat="server"></asp:LinkButton>
                <cc1:ModalPopupExtender ID="popup" runat="server" DropShadow="false"
                PopupControlID="pnlAddEdit" TargetControlID="InkFake"
                BackgroundCssClass="modalBackground">
                </cc1:ModalPopupExtender>
                <asp:Button ID="Button3" CssClass="btn btn-primary btn-rounded" runat="server" Text="Add"
                OnClick="Button3_Click" />
            </td>
            </tr>
            <tr>
            <tr>
                <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">College Attended:</td>
                <td>
                    <asp:DropDownList ID="ddl_college" CssClass="show" runat="server">
                        </asp:DropDownList>
                    <asp:TextBox ID="txt_college_year" runat="server"></asp:TextBox>
                </td><td> <asp:CheckBox ID="college_chk" runat="server" /></td>
            </td>
            <td>
                <asp:Button ID="Button5" CssClass="btn btn-primary btn-rounded" runat="server" Text="Add"
                OnClick="Button5_Click" />
            </td>
            </tr>
            <tr>
            <tr>
                <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">University
                Attended:</td>
            </tr>

```

```

        <td>
            <asp:DropDownList ID="ddl_university" CssClass="show"
runat="server">

                </asp:DropDownList>
                <asp:TextBox ID="txt_univ_year" runat="server"></asp:TextBox>
            </td> <td> <asp:CheckBox ID="univ_chk" runat="server" /></td>
        </td>
        <asp:Button ID="Button6" CssClass="btn btn-primary btn-rounded" runat="server" Text="Add"
OnClick="Button6_Click"/>

            </td>
        </tr>

</tr>
<td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Date of Birth :</td>
<td>

    <ig:WebDatePicker ID="dob" runat="server" DisplayModeFormat="dd-MM-yyyy"
EditModeFormat="dd-MM-yyyy" Nullable="false" MinValue="1950-01-01" MaxValue="2002/01/01"
CausesValidation="true">

</ig:WebDatePicker>

<%-- <ig:WebMonthCalendar ID="webcal" CssClass=" bg-success"
runat="server"></ig:WebMonthCalendar>                                --%>
<asp:TextBox ID="txt_yrest" runat="server" MaxLength="20" Visible="false"></asp:TextBox>
</td>
    <td>
<asp:CheckBox ID="dob_chk" runat="server" Checked="true" />
</td>

</tr>
<tr>
<td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Time of Birth :</td>
<td>
<%--<ig:WebDateTimeEditor ID="WebDateTimeEditor1" runat="server" DisplayModeFormat="HH:mm"
MinValue="00:00">

</ig:WebDateTimeEditor>--%>
    <asp:TextBox ID="txt_tob" runat="server" MaxLength="40" ></asp:TextBox>
</td>
    <td >
        <asp:CheckBox ID="tob_chk" runat="server" Checked="true" Visible="true" />
</td>

</tr>

```

```

<tr>
<td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Place of Birth :</td>
<td>
  <asp:TextBox ID="txt_pob" runat="server" MaxLength="40" ></asp:TextBox>
</td>
<td>
  <asp:CheckBox ID="pob_chk" runat="server" />
</td>
</tr>

<tr>
  <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Marital Status:</td>
  <td>
    <asp:DropDownList ID="ddl_Martial_status" runat="server">
<%-- <asp:ListItem Value="0">-- Select Your Martial Status --</asp:ListItem>
  <asp:ListItem Value="1">Single/Unmarried</asp:ListItem>
  <asp:ListItem Value="2">Single/Divorced(no children)</asp:ListItem>
  <asp:ListItem Value="3">Single/Divorced(with children)</asp:ListItem>
  <asp:ListItem Value="4">Single/Widowed(no children)</asp:ListItem>
  <asp:ListItem Value="5">Single/Widowed(with children)</asp:ListItem>
  <asp:ListItem Value="6">Single/Seperated(no children)</asp:ListItem>
  <asp:ListItem Value="7">Single/Seperated(with children)</asp:ListItem>
  <asp:ListItem Value="8">asp:ListItem>--%>

    </asp:DropDownList>
</td>
<td>
  <asp:CheckBox ID="Martial_status_chk" runat="server" />
</td>
</tr>
<tr>
  <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Sexual
Orientation:</td>
  <td>
    <asp:DropDownList ID="ddl_Sex_ornt" runat="server">
<asp:ListItem Value="0">-- Select Your Orientation --</asp:ListItem>
  <asp:ListItem Value="1">Straight </asp:ListItem>
  <asp:ListItem Value="2">Gay</asp:ListItem>
  <asp:ListItem Value="3">Bi-Sexual</asp:ListItem>

    </asp:DropDownList>
</td><td>
  <asp:CheckBox ID="Sex_ornt_chk" runat="server" />
</td>
</tr>

```

```

<tr>
  <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Religion:</td>
  <td>
    <asp:DropDownList ID="ddl_Religion" runat="server">

    </asp:DropDownList>
    <asp:CheckBox ID="Religion_chk" runat="server" />
  </td><td>

  </td>
</tr>
<tr>
  <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Caste:</td>
  <td>
    <asp:DropDownList ID="ddl_caste" runat="server">

    </asp:DropDownList>
    <asp:CheckBox ID="Caste_chk" runat="server" />
  </td><td>

  </td>
</tr>
<tr>
  <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Sub-Caste:</td>
  <td>
    <asp:TextBox ID="ddl_subcaste" runat="server" MaxLength="40"
    CssClass="col-lg-9 col-md-9 col-sm-10 pull-left"></asp:TextBox>
    <%--
    <asp:DropDownList ID="ddl_subcaste" runat="server">

    </asp:DropDownList>--%>
    <asp:CheckBox ID="Sub_Caste_chk" runat="server" />
  </td><td>

  </td>
</tr>
<tr>
  <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">No of Family
  Members</td>
  <td>
    <asp:TextBox ID="txt_family_nos" runat="server" MaxLength="40"
    ></asp:TextBox>
    <asp:CheckBox ID="family_nos_chk" runat="server" />
  </td><td>

  </td>
</tr>
<tr>
  <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Web-URL</td>

```



```

        <td>
            <asp:TextBox ID="txt_web_url" runat="server" MaxLength="40"
></asp:TextBox>
        </td><td>
            <asp:CheckBox ID="web_url_chk" runat="server" />
        </td>
    </tr>
    <tr>
        <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Income Per Year:
        </td>
        <td >
            <asp:TextBox ID="txt_Income_year" runat="server" MaxLength="40"
            CssClass="col-lg-3 col-md-3 col-sm-3" ></asp:TextBox>
            <asp:Label ID="curr" runat="server" CssClass="col-lg-3 col-md-3
            col-sm-3" align="right" dir="right">Currency:</asp:Label>
            <asp:DropDownList ID="ddl_currency" runat="server" CssClass="col-lg-3
            col-md-3 col-sm-3" ></asp:DropDownList>
        </td><td>
            <asp:CheckBox ID="Income_Year_chk" runat="server" />
        </td>
    </tr>
    <tr>
        <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Location :</td>
        <td>
            <asp:TextBox ID="txt_Loc" runat="server" MaxLength="40"
            CssClass="col-lg-9 col-md-9 col-sm-10 pull-left"></asp:TextBox>
        </td>
        <td>
            <asp:CheckBox ID="Location_chk" runat="server" />
        </td>
    </tr>
    <tr>
        <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Weekly working days
        :</td>
        <td>
            <asp:TextBox ID="txt_weekly_days" runat="server" MaxLength="40" CssClass="col-lg-3 col-md-3
            col-sm-3" ></asp:TextBox>
        </td><td>
            <asp:CheckBox ID="weekly_days_chk" runat="server" />
        </td>
    </tr>
    <tr>
        <td class="col-lg-3 col-md-3 col-sm-3" align="right" dir="auto">In
        Unit of Measure:</td>
    </tr>

```

```
</td>
</tr>
```

```
<tr><td>Public Images</td><td></td><td></td></tr>
```

```

    <tr>
      <td>
        <asp:image ToolTip = "ASP Image Control" ID="Image1" runat="server" ImageUrl
="images/nocolor.png" Height="156px" Width="174px"></asp:image> </td>
      <td>
        <asp:image ToolTip = "ASP Image Control" ID="Image6" runat="server" ImageUrl
="images/nocolor.png" Height="156px" Width="174px"></asp:image>
      </td>
      <td>
        <asp:CheckBox ID="Photo_chk" runat="server" />
      </td>
    </tr>
</tbody>
```

```
</table>
```

```
</div>
<!--Bank Account END-->
</div>
```

```
</div>
</div>
```

```
</div>
<!-- Settings -->
<div class="tab-pane media-block" id="Choices">
  <div class="row-fluid">
    <div id="Div1" runat="server" class=" col-lg-12 col-md-12 col-sm-12 " >
      <asp:Button ID="Button1" runat="server" CssClass="btn btn-primary btn-md btn-rounded"
Text="Save Settings" OnClick="Button1_Click" />
<!-- <a class="navbar-brand" href="UserChoices.aspx">
  <p>Other Search Options</p>
</a-->
```

[<i class="fas fa-search"></i> Other Search Options](UserChoices.aspx)

```
<div class="row-fluid">
  <h3>Choices/Setting</h3>(Private Data for Search usage)</div>
  <table id="Table1" class="table table-striped table-detail col-lg-12 col-md-12 col-sm-12 bg-
secondary" runat="server" visible="true">
    <tbody>
      <tr>
        <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Searching for :</td>
        <td>
          <asp:DropDownList ID="ddl_choice" runat="server">
            <asp:ListItem Value="0">-- Select Your Choice --</asp:ListItem>
            <asp:ListItem Value="1">Women</asp:ListItem>
            <asp:ListItem Value="2">Men</asp:ListItem>
            <asp:ListItem Value="3">Bi-Sexuals</asp:ListItem>
            <asp:ListItem Value="4">Others</asp:ListItem>
          </asp:DropDownList>
        </td>
      </tr>
      <tr>
        <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Age Search (Between) :</td>
        <td>
          <asp:TextBox ID="Age_min" runat="server" MaxLength="100" Width="150px"
Text="21"></asp:TextBox>
          <asp:TextBox ID="Age_max" runat="server" MaxLength="100" Width="150px"
Text="25"></asp:TextBox>
        </td>
      </tr>
      <tr>
        <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Religion Search:</td>
        <td>
          <asp:DropDownList ID="ddl_Religion_search" runat="server">
            </asp:DropDownList>
        </td>
      </tr>
    </tbody>
  </table>
```

```

        <asp:CheckBox ID="Religion_search_chk" runat="server" />
    </td>
    </tr>
    <tr>
        <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Caste Search:</td>

        <td>
            <asp:DropDownList ID="ddl_caste_search" runat="server">
            </asp:DropDownList>

        </td>
    </tr>
    <tr>
        <td>
            <asp:CheckBox ID="Caste_search_chk" runat="server" />
        </td>
    </tr>
    <tr>
        <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Area Search :</td>

        <td>
            <asp:TextBox ID="ddl_area" runat="server" MaxLength="40"
class="col-lg-9 col-md-9 col-sm-10 pull-left"></asp:TextBox>--%>
            <asp:DropDownList ID="ddl_area" runat="server">
            </asp:DropDownList>

        </td>
    </tr>
    <tr>
        <td>
            <asp:CheckBox ID="Area_search_chk" runat="server" />
        </td>
    </tr>
    </tr>
    <tr>
        <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">City Search :</td>

        <td>
            <asp:DropDownList ID="ddl_city" runat="server">
            </asp:DropDownList>

        </td>
    </tr>
    <tr>
        <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Geolocation :</td>

        <td>
            <asp:TextBox ID="Geolocation" runat="server" MaxLength="100"
Width="250px"></asp:TextBox>

        </td>
    </tr>

```

```

        <tr>
            <td align="right" class="label_txt"><strong>Country Search:</strong></td>
            <td width="25%" align="left">
                <asp:DropDownList ID="ddl_countrysearch"
runat="server"></asp:DropDownList></td>

        </tr>
--%>

        <tr>
            <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Search by Mother Tongue :</td>

            <td>
                <asp:DropDownList ID="ddl_mtonguesearch" runat="server"></asp:DropDownList>

            </td>
            <td>
                <asp:CheckBox ID="MTongue_search_chk" runat="server" />
            </td>
        </tr>
<%--
        <tr>
            <td class="col-lg-3 col-md-3 col-sm-3" dir="auto">Search by Languages Known :</td>

            <td>
                <asp:DropDownList ID="ddl_languages" runat="server"></asp:DropDownList>

            </td>
        </tr>
--%>

    </tbody>

        </table>

</div>
</div>
</div>

<!-- Address -->
<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 "
style=""><h3>Addresses(Private)</h3>
        <table>
            <tbody>
                <tr>
                    <td>
                        <asp:Button ID='Button2' runat='server' CssClass='btn btn-primary btn-md btn-rounded'
Text='Del' CommandName='1' OnClick='Button2_Click' /></td>
                    </tr>
                </tbody>
            </table>
    </div>

```

```

        </table>
    </div>
    <asp:Button ID="btn_SaveAddress" runat="server" CssClass="btn btn-primary btn-md btn-
rounded" Text="Save Address" Enabled="false" 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-12 col-md-12 col-sm-12" style="background-color: #ffffff; border:
1px solid #eefef; border-top: none; padding: 15px 5px;">
<%--
    <h3>Add Addresses</h3>--%>
        <table id="Addressdet" class="table table-striped table-detail col-lg-6 col-md-6
col-sm-12" runat="server" visible="false">

            <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:ListItem Value="0">-- Select Your Address Type --</asp:ListItem>
<asp:ListItem Value="1">Home Address</asp:ListItem>
<asp:ListItem Value="2">Office Address</asp:ListItem>
<asp:ListItem Value="3">Factory Address</asp:ListItem>
<asp:ListItem Value="4">Billing Address</asp:ListItem>

                        </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=" col-lg-12 col-md-12 col-sm-12 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;"><h3>Images(Private)</h3></div>
                    <ul class="col-lg-12 col-md-12 col-sm-12" runat="server">
                        <li class=" col-lg-2 col-md-2 col-sm-12">
                            <asp:image ToolTip = "ASP Image Control" ID="Image2" runat="server" ImageUrl
="ManageProducts.aspx?ImageID=1" Height="156px" Visible="false"></asp:image> </li>
                            <li class="col-lg-2 col-md-2 col-sm-12">
                                <asp:image ToolTip = "ASP Image Control" ID="Image3" runat="server" ImageUrl
="ManageProducts.aspx?ImageID=2" Height="156px" Visible="false"></asp:image>
                                </li><li class="col-lg-2 col-md-2 col-sm-12">
                                    <asp:image ToolTip = "ASP Image Control" ID="Image4" runat="server" ImageUrl
="ManageProducts.aspx?ImageID=3" Height="156px" Visible="false"> </asp:image>
                                    </li><li class="col-lg-2 col-md-2 col-sm-12">
                                        <asp:image ToolTip = "ASP Image Control" ID="Image5" runat="server" ImageUrl
="ManageProducts.aspx?ImageID=4" Height="156px" Visible="false"> </asp:image>

                                        </li>

                                <li class="col-lg-1 col-md-1 col-sm-1" dir="auto">Add Images</li>
                            </li>
                            <asp:FileUpload ID="FileUpload1" runat="server" />
                            </li><li>
                                <asp:Button ID="btn_upload" runat="server" Text="Upload" OnClick="btn_upload_Click"
CausesValidation="false" />
                                </li>

                        </ul>

                    </div>
                <div class="row-fluid">

```


</div>

<!--Notes Ends----->

</div>

</div>

<!-- End Description -->

</div>

<div class="tab-pane fade in" id="Maps">

<div style="width: 640px; height: 480px" id="mapContainer"> <!---%>

<!-- <script async defer

src="https://maps.googleapis.com/maps/api/js?key=AlzaSyBSxyfN6Qe4VZNXau0NRbSIW4dgJWtYOLU&callback=initMap">

</script>--%>

<div class="mapouter">

<div class="gmap_canvas col-sm-12"> <!--width="600" height="500" --%>

<iframe class="col-lg-12 col-md-12 col-sm-12" height="500" width="600" id="gmap_canvas"

src="https://maps.google.com/maps?q=Nampally, Hyderabad, Telangana&t=&z=13&ie=UTF8&iwloc=&output=embed" frameborder="0" scrolling="no" marginheight="0" marginwidth="0" runat="server"></iframe>

embedgooglemap.net</div><style>.mapouter{position: relative;text-align:right;height:500px;width:600px;}gmap_canvas {overflow:hidden;background:none!important;height:500px;width:600px;}</style>

</div>

</div>

</div>

</div>

<!--ModalPageStarts---->

<asp:Panel ID="pnAddEdit" runat="server" CssClass="modalPopup" Style="display: block;">

<div class="row-fluid">

<div class="col-sm-8" style="background-color: #ffffff; border: 1px solid #eefef; border-top: none; padding: 5px;">

<!-- panel table removed--%>

<div class="row-fluid">

<div id="popup_head" class="col-sm-8 customer" style="text-align: left;" runat="server">Add School to list</div>

</div>

<table id="Table9" class="table table-striped table-detail col-lg-4 col-sm-12" runat="server">

```

<tbody>
  <tr>
    <td align="right" id="namescu" class="label_txt" dir="auto"
runat="server">College Name:</td>
    <td>
      <asp:TextBox ID="txt_name" runat="server"></asp:TextBox>
    </td>
  </tr>
  <tr>
    <td width="14%" align="right" class="label_txt" dir="auto">City:</td>
    <td>
      <asp:DropDownList ID="ddl_City" CssClass="show"
runat="server"></asp:DropDownList>
    </td>
  </tr>
  <tr>
    <td width="14%" align="right" class="label_txt" dir="auto">Country:</td>
    <td>
      <asp:DropDownList ID="ddl_cntry" CssClass="show"
runat="server"></asp:DropDownList>
    </td>
  </tr>
  <tr>
    <td align="right" class="label_txt" dir="auto">Zipcode/Pincode:</td>
    <td>
      <asp:TextBox ID="txt_zipcode" runat="server" ></asp:TextBox>
    </td>
  </tr>
  <tr>
    <td>
      <asp:Button ID="btnSave" runat="server" Text="Save" OnClick="btnSave_Click"
CssClass="btn btn-primary" />
    </td>
    <td>
      <asp:Button ID="btnCancel" runat="server" Text="Cancel"
OnClientClick="return Hidepopup()" CssClass="btn btn-primary" />
    </td>
  </tr>

```

```

                </td>
            </tr>
        </tbody>
    </table>

    </div>
</div>

</asp:Panel>
    <asp:LinkButton ID="LinkButton1" runat="server"></asp:LinkButton>
<cc1:ModalPopupExtender ID="ModalPopupExtender1" runat="server" DropShadow="false"
    PopupControlID="pnlAddEdit" TargetControlID="lnkFake"
    BackgroundCssClass="modalBackground">
</cc1:ModalPopupExtender>

<!--ModalPageEnds----->
    </div>

<%--
    <script type="text/javascript" src="js/jquery.min.js"></script>
    <script type="text/javascript" src="js/bootstrap.min.js"></script>
    <script type="text/javascript" src="plugins/owl-carousel/owl.carousel.min.js"></script>
    <script type="text/javascript" src="plugins/FlexSlider/jquery.flexslider-min.js"></script>
    <script type="text/javascript" src="js/jquery.magnific-popup.min.js"></script>
    <script type="text/javascript" src="js/product.js"></script>
    <script type="text/javascript" src="js/magnific-popup.min.js"></script>
    <script type="text/javascript" src="js/slick.js"></script>
    <script type="text/javascript" src="js/slick-slider.js"></script>
    <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 DOWEntity;
using DOWDal;
using System.Data;
using System.Data.SqlClient;
using System.Configuration;
using System.Drawing;
using System.Web.Security;
using System.IO;
using System.Net;
using System.Globalization;
public partial class UserDetail : System.Web.UI.Page
{
    public string PageType;
    string pageid;
    int UserId;
    int Prid = 0;
    string user;
    DDLogTracer objlog = new DDLogTracer();
    bool IsMobile = false;
    PopDlIs ObjPD = new PopDlIs();

    protected void Page_Load(object sender, EventArgs e)
    {
        if (Request.Browser.IsMobileDevice) { IsMobile = true; } else { IsMobile = false; }
        //dob.DropDownCalendarID = webcal.ID;
        ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
        if (Clsusr == null)
        {
            bool ret = new Common().guest_login();
            if (ret == false)
            {
                Response.Redirect("~/Login.aspx");
            }
        }

        if (!IsMobile) { navbutton.Attributes.CssStyle.Add("display", "none"); }
        if (Clsusr != null)
        {
            PageType = Page.Request.QueryString["PageType"];
            if (PageType == null)
            {
                PageType = "Edit";
            }
            Session["PageType"] = PageType;
            pageid = "4";
            if (Convert.ToInt16( Clsusr.GroupId) <4)

```

```

        {
            Common.ShowMessageHtml("PI contact Administrator as you have no profile", 0, true,
dvMsgBox);
            return;
        }

        if (!this.IsPostBack)
        {
            RefreshScreen();
        }
    }
}
protected void RefreshScreen ()
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    if (Clsusr == null)
    {
        bool ret = new Common().guest_login();
        if (ret == false)
        {
            Response.Redirect("~/Login.aspx");
        }
    }

    ObjPD.BindDropDowns(ddl_college, (new PopDlIs()).GetCollegesList());
    ObjPD.BindDropDowns(ddl_university, (new PopDlIs()).GetUniverssityList());
    ObjPD.BindDropDowns(ddl_school, (new PopDlIs()).GetSchoosList());
    ObjPD.BindDropDowns(ddl_choice, (new PopDlIs()).GetPartnerChoice());
    ObjPD.BindDropDowns(ddl_currency, (new PopDlIs()).GetCurrency());
    //ObjPD.BindDropDowns(ddl_currency, (new PopDlIs()).GetCurrency());
    //ObjPD.BindDropDowns(ddl_sexor, (new PopDlIs()).Gettbluom());
    ObjPD.BindDropDowns(ddl_Martial_status, (new PopDlIs()).GetMartialStatus());
    //ObjPD.BindDropDowns(ddl_addtype, (new PopDlIs()).GetAddType());
    ObjPD.BindDropDowns(ddl_country, (new PopDlIs()).GetCountries());
    ObjPD.BindDropDowns(ddl_cntry, (new PopDlIs()).GetCountries());
    //ObjPD.BindDropDowns(ddl_countrysearch, (new PopDlIs()).GetCountries());
    ObjPD.BindDropDowns(ddl_country1, (new PopDlIs()).GetCountries());
    ObjPD.BindDropDowns(ddl_GroupId, (new
PopDlIs()).GetGroups(Convert.ToInt32(Clsusr.UserId), Convert.ToInt32(Clsusr.GroupId)));
    //    ddl_GroupId.Enabled = false;
    Addressdet.Visible = false;
    ObjPD.BindDropDowns(ddl_Religion, (new PopDlIs()).GetReligions());
    ObjPD.BindDropDowns(ddl_caste, (new PopDlIs()).GetCastes());
    ObjPD.BindDropDowns(ddl_mtonguesearch, (new PopDlIs()).GetLanguages());
    ObjPD.BindDropDowns(ddl_mtongue, (new PopDlIs()).GetLanguages());
    ObjPD.BindDropDowns(ddl_Religion_search, (new PopDlIs()).GetReligions());
    ObjPD.BindDropDowns(ddl_caste_search, (new PopDlIs()).GetCastes());
    ObjPD.BindDropDowns(ddl_mtonguesearch, (new PopDlIs()).GetLanguages());

```

```

//ddl_area
    string ProductId = Page.Request.QueryString["Id"];
    UserId = Convert.ToInt16(Clsusr.UserId);
    if (UserId > 0)
    {
        //-----user infor
        EtetblUser Ist = (new tblUsers()).getuserbyid(UserId);
        if (Ist.Status==false)
        {
            Common.ShowMessageHtml("Pl contact Administrator to Approve your Account" +
Clsusr.UserName + "-" + Clsusr.UserId + "-" + Clsusr.GroupId, 0, true, dvMsgBox);

        }
        //txt_username.Text = Ist.UserId.ToString();
        txt_username.Text = Ist.UserName;
        txt_username.Enabled = false;
        txt_FirstName.Text = Ist.FirstName;
        txt_LastName.Text = Ist.LastName;

        email_name.Text = Ist.EmailID;
        if (email_name.Text == "" || email_name.Text == null)
        { email_name.Enabled = true; }
        txt_mobile.Text = Ist.ContactNum;
        ddl_country.SelectedIndex = Convert.ToInt16(Ist.Region);
        txt_mobile.Text = Ist.ContactNum;
        // ddl_GroupId.SelectedIndex = Ist.GroupId;
        ddl_GroupId.SelectedValue = Ist.GroupId.ToString();
        //txt_Password.Text = new BalCommon().EncryptPwd(Ist.Password.ToString());
        //txt_Password1.Text = new BalCommon().EncryptPwd(Ist.Password.ToString());
        if (Ist.Gender == "") { Ist.Gender = "0"; }
        ddl_gender.SelectedValue = Ist.Gender.ToString().Substring(0,1);
        ddl_mtongue.SelectedIndex = Ist.MTongue;
        txt_landline.Text = Ist.Landline;

        //-----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.Details;
        txt_about.Text = Prd.About;
        txt_Occupation.Text = Prd.Occupation;
        txt_Occupation_type.Text = Prd.Occupationtype;
        txt_Edu_qualification.Text = Prd.Eduqualification;
        ddl_Martial_status.SelectedIndex = Prd.Martialstatus;
    }

```

```

ddl_Sex_ornt.SelectedIndex = Prd.Sexornt;
dob.Date = Prd.dob.Date;
txt_yrest.Text = Prd.dob.Date.ToShortDateString();
txt_tob.Text = Prd.tob.ToString();
txt_pob.Text = Prd.pob;
txt_family_nos.Text = Prd.familynos.ToString();
txt_web_url.Text = Prd.weburl;
txt_Income_year.Text = Prd.Income_year.ToString();
ddl_currency.SelectedIndex = Prd.currency;
txt_Loc.Text = Prd.Location;
txt_adhaar_id.Text = Prd.adhaarid;
txt_pan_no.Text = Prd.panno;
txt_weekly_days.Text = Prd.weeklydays.ToString();
ddl_subcaste.Text = Prd.Sub_Caste;
ddl_college.SelectedIndex = Prd.College;
ddl_school.SelectedIndex = Prd.School;
ddl_university.SelectedIndex = Prd.University;
txt_school_year.Text = Prd.School.ToString();
txt_college_year.Text = Prd.College.ToString();
txt_univ_year.Text = Prd.Univ_year.ToString();
ddl_Religion.SelectedIndex=Prd.Religion;
ddl_caste.SelectedIndex=Prd.Caste;

}
Etebluser_options opt = new tblUsers().getuserchoies(UserId);
if (opt != null)
{
ddl_choice.SelectedValue =opt.Gender_Search;
Age_min.Text = opt.Age_from.ToString();
Age_max.Text = opt.Age_to.ToString();
ddl_Religion.SelectedIndex = opt.Religion_search;
ddl_caste.SelectedIndex = opt.Caste_search;
ddl_area.SelectedIndex = opt.Area_search;
//ddl_city.SelectedIndex=opt.city
//ddl_countrysearch
ddl_mtonguesearch.SelectedIndex = opt.MTongue_search;
//ddl_languages.SelectedIndex=opt.
//ddl_choice.SelectedValue = opt.Gender_Search;
//ddl_Religion.SelectedValue = opt.Religion_search.ToString();
//ddl_caste.SelectedValue = opt.Caste_search.ToString();
//ddl_area.SelectedValue = opt.Area_search.ToString();
//ddl_mtonguesearch.SelectedValue = opt.MTongue_search.ToString();

}
else
{
//ddl_choice.SelectedIndex = 0;
//Age_min.Text = "21";

```

```

//Age_max.Text = "25";
//ddl_Religion.SelectedIndex = 0;
//ddl_caste.SelectedIndex = 0;
//ddl_area.SelectedIndex = 0;

}

int srno = 1;
List<EtetblUserImages> igs = new tblUsers().getuserImagesbyuserid(UserId, "Profile");
foreach (var imgs in igs)
{
    if (srno == 1)
    {
        Image1.ImageUrl = imgs.ImageUrl;
    }
    else
    if (srno == 2)
    {
        Image6.ImageUrl = imgs.ImageUrl;
    }
    else
    if (srno == 3)
    {
        Image2.ImageUrl = imgs.ImageUrl;
    }
    else
    if (srno == 4)
    {
        Image3.ImageUrl = imgs.ImageUrl;
    }
    else
    if (srno == 5)
    {
        Image4.ImageUrl = imgs.ImageUrl;
    }
    else
    if (srno == 6)
    {
        Image5.ImageUrl = imgs.ImageUrl;
    }

}
//

```

```

StringBuilder sb = new StringBuilder();

```



```

List<Etetbluser_address> Lst = (new tblUsers()).getusersaddressbyuserid(UserId, "");
if (Lst.Count() > 0)
{
    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%'>Address</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.FirstName + " " + Clsusr.LastName + "</td> \n");
        sb.Append(" <td>" + add.address1 + ", " + add.address2 + ", " + add.address3 +
"</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> <asp:Button ID='Button2"+add.address_type+"'' runat='server'
CssClass='btn btn-secondary show' Text='X' OnClick='Button2_Click' /></td> \n");
    }
}

```

```

//          sb.Append(" <td>          <asp:Button ID='Button" + add.address_type + "' runat='server'
CssClass='btn btn-primary' Text='Del' CommandName='" + add.address_type + "'
OnClick='Button2_Click' /></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("          </tbody>          \n");
          sb.Append("          </tr></Table> \n");

          string adds = add.address1 + "," + add.address2 + "," + add.address3 + "," + add.city;
          if (adds != "")
          {
              gmap_canvas.Src = "https://maps.google.com/maps?q=" + adds +
"&t=&z=13&ie=UTF8&iwloc=&output=embed";
          }

      }

      objlog.Trace_Log(1, "home", "1", "menu", "\n" + sb.ToString());

      addlist.InnerHtml = sb.ToString();
      addlist.Visible = true;
      //Addressdet.Visible = false;
  }
  Etetbluser_privacy_chk data = new tblUsers().GetPrivacyCheck(UserId, 0);
  if (data != null)
  {
      Details_chk.Checked = data.Details_chk;
      about_chk.Checked = data.about_chk;
      Occupation_chk.Checked = data.Occupation_chk;
      Occupation_type_chk.Checked = data.Occupation_type_chk;
      Edu_qualification_chk.Checked = data.Edu_qualification_chk;
      Martial_status_chk.Checked = data.Martial_status_chk;
      Sex_ornt_chk.Checked = data.Sex_ornt_chk;
      dob_chk.Checked = data.dob_chk;
      tob_chk.Checked = data.tob_chk;
      pob_chk.Checked = data.pob_chk;
      family_nos_chk.Checked = data.family_nos_chk;
      web_url_chk.Checked = data.web_url_chk;
      adhaar_id_chk.Checked = data.adhaar_id_chk;
      pan_no_chk.Checked = data.pan_no_chk;
      weekly_days_chk.Checked = data.weekly_days_chk;
      Income_Year_chk.Checked = data.Income_Year_chk;
      currency_chk.Checked = data.currency_chk;
      Religion_chk.Checked = data.Religion_chk;
  }

```

```

Caste_chk.Checked = data.Caste_chk;
Sub_Caste_chk.Checked = data.Sub_Caste_chk;
Location_chk.Checked = data.Location_chk;
school_chk.Checked = data.school_chk;
college_chk.Checked = data.college_chk;
univ_chk.Checked = data.university_chk;
Photo_chk.Checked = data.Photos_chk;
//GeoLocation_chk.Checked = data.GeoLocation_chk;
////Gender_Search_chk.Checked = data.Gender_Search_chk;
////Age_from_chk.Checked = data.Age_from_chk;
//Age_to_chk.Checked = data.Age_to_chk;
Religion_search_chk.Checked = data.Religion_search_chk;
Caste_search_chk.Checked = data.Caste_search_chk;
Area_search_chk.Checked = data.Area_search_chk;
MTongue_search_chk.Checked = data.MTongue_search_chk;
//GeoLoc_search_chk.Checked = data.GeoLoc_search_chk;
//Religions_Choice.Checked = data.Religions_Choice;
//Caste_Choice.Checked = data.Caste_Choice;
//Areas_Choice.Checked = data.Areas_Choice;
//Cities_Choice.Checked = data.Cities_Choice;
//Countries_Choice.Checked = data.Countries_Choice;
//Languages_Choice.Checked = data.Languages_Choice;
//uid.Checked = data.uid;
//CreatedOn.Checked = data.CreatedOn;
//UpdatedOn.Checked = data.UpdatedOn;
}
//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:
            Image1.ImageUrl = pic.ImageUrl;
            Image1.Visible = true;
            break;
        case 2:
            Image6.ImageUrl = pic.ImageUrl;
            Image6.Visible = true;
            break;
        case 3:
            Image2.ImageUrl = pic.ImageUrl;
            Image2.Visible = true;
            break;
        case 4:

```

```

        Image3.ImageUrl = pic.ImageUrl;
        Image3.Visible = true;
        break;
    case 5:
        Image4.ImageUrl = pic.ImageUrl;
        Image4.Visible = true;
        break;
    case 6:
        Image5.ImageUrl = pic.ImageUrl;
        Image5.Visible = true;
        break;
    default:
        break;
    }
} //for each in pics

}

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();
    IPEndPoint ipEntry = System.Net.Dns.GetHostEntry(strHostName);
    IPAddress[] addr = ipEntry.AddressList;
    string ip1 = addr[1].ToString();

    BALCreateUser objlogin = new BALCreateUser();
    DataTable status_login = objlogin.LoginGuest(Common.Encrypt( 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 = new
tblUsers().getusername(Convert.ToInt16(objLoggedInUser.UserId));
        //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"]);
objLoggedInUser.FirstName = status_login.Rows[0]["FirstName"].ToString();
objLoggedInUser.LastName = status_login.Rows[0]["LastName"].ToString();
Session["LoggedUser_Specs"] = objLoggedInUser;

FormsAuthentication.SetAuthCookie(objLoggedInUser.UserName, false);
HttpCookie myCookie = new HttpCookie("DOWCookie");
DateTime now = DateTime.Now;
myCookie.Value = objLoggedInUser.UserName;
myCookie.Expires = now.AddYears(1);

Response.Cookies.Add(myCookie);
}
}
private DataTable GetData(SqlCommand cmd)
{
    DataTable dt = new DataTable();
    String strConnString =
System.Configuration.ConfigurationManager.ConnectionStrings["DOWConectingString"].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();
    }
}
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"];
    if (Clsusr == null)
    {
        bool ret = new Common().guest_login();
        if (ret == false)
        {
            Response.Redirect("~/Login.aspx");
        }
    }
    UserId = Convert.ToInt32(Clsusr.UserId);
    try
    {
        if (Details.Text == "")
        {
            Common.ShowMessageHtml("Enrer you interests", 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.ToInt16(Clsusr.UserId);
        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
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    if (Clsusr == null)
    {
        bool ret = new Common().guest_login();
        if (ret == false)
        {
            Response.Redirect("~/Login.aspx");
        }
    }
}
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.userid = Convert.ToInt16( Clsusr.UserId);
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)
{
string adds = txt_add1.Text + "," + txt_add2.Text + "," + txt_add3.Text + "," + txt_city.Text;
gmap_canvas.Src = "https://maps.google.com/maps?q=" + adds +
"&t=&z=13&ie=UTF8&iwloc=&output=embed";
Common.ShowMessageHtml("Address Updated", 0, true, dvMsgBox);
StringBuilder sb = new StringBuilder();
}
}
}

```



```

List<Etetbluser_address> Lst = (new tblUsers()).getusersaddressbyuserid(UserId, "");
if (Lst.Count() > 0)
{
    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%'>Address</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 + add.address2 + add.address3 + "</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;&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();
}
Addressdet.Visible = false;
RefreshScreen();
//if (Lst != null && Lst.Count > 0)
//{
//  WDGrid_UserAddress.DataSource = Lst;
//  WDGrid_UserAddress.DataBind();
//  WDGrid_UserAddress.Visible = true;
//}
//else
//{
//  WDGrid_UserAddress.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;
    string retpath;
    retpath = new tblUsers().InsertImages(ep);
    string server = Server.MapPath(retpath);
    string server1 = Server.MapPath("thumb/" + retpath);
    File.WriteAllBytes(@server, Photo);
    ReduceImageSize(0.2, server, server1);

    List<EtetblUserImages> Pics = (new tblUsers()).getuserImagesbyuserid(UserId, "");
    foreach (var pic in Pics)
    {
        switch (pic.srno)
        {
            case 1:
                Image1.ImageUrl = pic.ImageUrl;

```

```

        Image1.Visible = true;
        break;
    case 2:
        Image6.ImageUrl = pic.ImageUrl;
        Image6.Visible = true;
        break;
    case 3:
        Image2.ImageUrl = pic.ImageUrl;
        Image2.Visible = true;
        break;
    case 4:
        Image3.ImageUrl = pic.ImageUrl;
        Image3.Visible = true;
        break;
    case 5:
        Image4.ImageUrl = pic.ImageUrl;
        Image4.Visible = true;
        break;
    case 6:
        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;
string retpath;
retpath = new tblUsers().InsertImages(ep);
string server = Server.MapPath(retpath);
string server1 = Server.MapPath("thumb/" + retpath);
File.WriteAllBytes(@server, Photo);
ReduceImageSize(0.2, server, server1);

List<EtetblUserImages> Pics = (new tblUsers()).getuserImagesbyuserid(UserId, "");
foreach (var pic in Pics)
{
    switch (pic.srno)
    {
        case 1:
            Image1.ImageUrl = pic.ImageUrl;
            Image1.Visible = true;
            break;
        case 2:
            Image6.ImageUrl = pic.ImageUrl;
            Image6.Visible = true;
            break;
        case 3:
            Image2.ImageUrl = pic.ImageUrl;
            Image2.Visible = true;
            break;
        case 4:
            Image3.ImageUrl = pic.ImageUrl;
            Image3.Visible = true;
            break;
        case 5:
            Image4.ImageUrl = pic.ImageUrl;
            Image4.Visible = true;
            break;
        case 6:
            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"];
    if (Clsusr == null)
    {
        bool ret = new Common().guest_login();
        if (ret == false)
        {
            Response.Redirect("~/Login.aspx");
        }
    }
    UserId = Convert.ToInt16(Clsusr.UserId);
    EtetblUserImages ep = new EtetblUserImages();
}

```

```

ep.UserId = UserId;

ep.Images = Photo;
string retpath;
retpath = new tblUsers().InsertImages(ep);
string server = Server.MapPath(retpath);
string server1 = Server.MapPath("thumb/" + retpath);
File.WriteAllBytes(@server, Photo);
ReduceImageSize(0.2, server, server1);

List<EtetblUserImages> Pics = (new tblUsers()).getuserImagesbyuserid(UserId, "");
foreach (var pic in Pics)
{
    switch (pic.srno)
    {
        case 1:
            Image1.ImageUrl = pic.ImageUrl;
            Image1.Visible = true;
            break;
        case 2:
            Image6.ImageUrl = pic.ImageUrl;
            Image6.Visible = true;
            break;
        case 3:
            Image2.ImageUrl = pic.ImageUrl;
            Image2.Visible = true;
            break;
        case 4:
            Image3.ImageUrl = pic.ImageUrl;
            Image3.Visible = true;
            break;
        case 5:
            Image4.ImageUrl = pic.ImageUrl;
            Image4.Visible = true;
            break;
        case 6:
            Image5.ImageUrl = pic.ImageUrl;
            Image5.Visible = true;
            break;
        default:
            break;
    }
}

}

protected void btn_addaddress_Click(object sender, EventArgs e)
{
    Addressdet.Visible =true;
}

```

```

    btn_SaveAddress.Enabled = true;
}
protected void btn3_Click(object sender, EventArgs e)
{//update tbluser
    objlog.Trace_Log(0, "DateOrWed", "MangeUser", "ContentSaveControlClicked", "Start Here..");
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    if (Clsusr == null)
    {
        UserId = Convert.ToInt16(Clsusr.UserId);
        bool ret = new Common().guest_login();
        if (ret == false)
        {
            Response.Redirect("~/Login.aspx");
        }
    }
    UserId = Convert.ToInt16(Clsusr.UserId);

    //try
    {
        //if (BalCommon.IsValidEmail(txt_username.Text) == true)
        //{
            // email_name.Text = txt_username.Text;
            //}
        if (txt_tob.Text == "") { txt_tob.Text = "0:00"; }
        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 Match.", 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 &&
txt_landline.Text.Trim()!="")
{

    Common.ShowMessageHtml("Landline No is Numeric only", 0, true, dvMsgBox);
}
else if (new FieldValidator().ValidateAlpha(txt_FirstName) == false)
{

    Common.ShowMessageHtml("Name Alphabetics only", 0, true, dvMsgBox);
}
else if (new FieldValidator().ValidateAlpha(txt_LastName) == false)
{

    Common.ShowMessageHtml("Company Name Alphabetics only", 0, true, dvMsgBox);
}
//else if (new FieldValidator().ValidateAlpha(txt_region) == false)
//{

//    Common.ShowMessageHtml("Region Alphabetics only", 0, true, dvMsgBox);
//}
else
{

```



```

EtetblUser _tbluser = new EtetblUser();
_tbluser= new tblUsers().getuserbyid(UserId);
if (_tbluser.UserId == null) { _tbluser.UserId = UserId; }
_tbluser.UserName = txt_username.Text;
_tbluser.FirstName = txt_FirstName.Text;
_tbluser.LastName = txt_LastName.Text;
_tbluser.Department = ddl_GroupId.SelectedValue;
_tbluser.Region = ddl_country.SelectedValue;
_tbluser.GroupId = Convert.ToInt32(ddl_GroupId.SelectedValue);
_tbluser.DealerId = 1;// Convert.ToInt32(ddl_DealerId.SelectedValue);
_tbluser.EmailID = email_name.Text;
_tbluser.ContactNum = txt_mobile.Text;
_tbluser.Status = true;// chk_IsLoginAllowed.Checked;
_tbluser.RoleID = 1;// Convert.ToInt32(ddl_RoleID.SelectedValue);
_tbluser.Gender =( ddl_gender.SelectedItem.ToString().Substring(0,1));
_tbluser.Landline = txt_landline.Text;
_tbluser.MTongue = ddl_mtongue.SelectedIndex;
_tbluser.DateOfBirth = Convert.ToDateTime(dob.Text);
//_tbluser.TimeOfBirth = DateTime.ParseExact(dob.Text + " " + txt_tob.Text, "dd-MM-yyyy
HH:mm", CultureInfo.InvariantCulture);

_tbluser .PlaceOfBirth= txt_pob.Text;
_tbluser.MatchingChk = (bool)currency_chk.Checked;
_tbluser.adshowchk = (bool)ad_chk.Checked;

if (txt_Password.Text!="")
{
    _tbluser.Password = new BalCommon().EncryptPwd(txt_Password.Text);
}
_tbluser.PageId = Convert.ToInt32(Request.QueryString["pageid"]);
_tbluser.Uid = Convert.ToInt32(Clsusr.UserId);
if (PageType == "Add")
{
    //bool exist;
    //exist = new tblUsers().UserNameExists(_tbluser.UserName);
    //if (exist == false)
    //{
    //    _tbluser.CreatedOn = DateTime.Now;
    //    sresult = (new tblUsers()).Insert(_tbluser);
    //    if (sresult == true)
    //    {
    //        Common.ShowMessageHtml("User has been Created successfully.", 1, true,
dvMsgBox);
    //        Response.Redirect("~/UserProfile.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 updated successfully.", 1, true, dvMsgBox);
        //Response.Redirect("~/Home.aspx?pageid=153&msg=User has been edited
successfully.");
        RefreshScreen();

    }
    else
    {
        //ShowMessage("Update Un-Successful");
        Common.ShowMessageHtml("Update user failed.", 0, true, dvMsgBox);
    }
}

```

```

    }
    }
}
//catch (Exception exc)
//{
// objlog.Trace_Error(4, "DateOrWed", "MangeUser", "ContentSaveControlClicked", "Exception:"
+ exc.ToString());
// Common.ShowMessageHtml(exc.ToString(), 0, true, dvMsgBox);

//}
objlog.Trace_Log(0, "DateOrWed", "MangeUser", "ContentSaveControlClicked", "Start Here..");
}
protected void Button1_Click(object sender, EventArgs e)
{
//save settings or choices
Etetbluser_options opt = new tblUsers().getuserchoies(UserId);
if (opt == null)
{
opt = new Etetbluser_options();
}
{
ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
if (Clsusr == null)
{
bool ret = new Common().guest_login();
if (ret == false)
{
Response.Redirect("~/Login.aspx");
}
}
}
UserId = Convert.ToInt32(Clsusr.UserId);
opt.userid = UserId;
opt.Gender_Search = ddl_choice.SelectedValue;
opt.Age_from = Convert.ToInt16( Age_min.Text);
opt.Age_to = Convert.ToInt16(Age_max.Text);
opt.Religion_search = ddl_Religion.SelectedIndex;
opt.Caste_search = ddl_caste.SelectedIndex;
opt.Area_search = ddl_area.SelectedIndex;
opt.MTongue_search = ddl_mtonguesearch.SelectedIndex;
bool res = new tblUsers().UpdateUser_options(opt);

}

}
protected void Button4_Click(object sender, EventArgs e)
{//update profile details
ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];

```

```

if (Clsusr == null)
{
    bool ret = new Common().guest_login();
    if (ret == false)
    {
        Response.Redirect("~/Login.aspx");
    }
}
UserId = Convert.ToInt32(Clsusr.UserId);
string updooption = "Edit";
try
{

    if (Details.Text == "")
    {
        Common.ShowMessageHtml("Enrer you interests", 0, true, dvMsgBox);

    }

    else
    {
        Etetbluser_profile Prd= new tblUsers().getuserprofilebyid(Convert.ToInt16(Clsusr.UserId));
        if (Prd == null)
        {
            Prd = new Etetbluser_profile();
            updooption = "Add";

        }

        Prd.userid =Convert.ToInt16( Clsusr.UserId);
        Prd.Details = Details.Text;
        Prd.About = txt_about.Text;
        Prd.Occupation = txt_Occupation.Text ;
        Prd.Occupationtype = txt_Occupation_type.Text ;
        Prd.Eduqualification = txt_Edu_qualification.Text ;
        Prd.Martialstatus= ddl_Martial_status.SelectedIndex ;
        Prd.Sexornt =Convert.ToChar( ddl_Sex_ornt.SelectedIndex) ;
        Prd.dob =Convert.ToDateTime( dob.Text );
        Prd.tob = txt_tob.Text ;
        Prd.pob = txt_pob.Text ;
        Prd.familynos =Convert.ToInt16( txt_family_nos.Text) ;
        Prd.weburl = txt_web_url.Text ;
        Prd.Income_year= Convert.ToDecimal(txt_Income_year.Text) ;
        Prd.currency = ddl_currency.SelectedIndex ;
        Prd.Location = txt_Loc.Text ;
        Prd.adhaarid = txt_adhaar_id.Text ;
        Prd.panno = txt_pan_no.Text ;
    }
}

```

```

Prd.weeklydays = Convert.ToInt16(txt_weekly_days.Text) ;
Prd.GeoLocation = "";
Prd.logopath = "";
Prd.Religion = ddl_Religion.SelectedIndex;
Prd.Caste = ddl_caste.SelectedIndex;
Prd.Sub_Caste = ddl_subcaste.Text;
Prd.College = ddl_college.SelectedIndex;
Prd.School = ddl_school.SelectedIndex;
Prd.University = ddl_university.SelectedIndex;
Prd.School_year = Convert.ToInt32( txt_school_year.Text);
Prd.College_year = Convert.ToInt32(txt_college_year.Text);
Prd.Univ_year = Convert.ToInt32(txt_univ_year.Text);
    bool Result = false;

    Prd.userid = Convert.ToInt16(Clsusr.UserId);
    Prd.UpdatedOn = DateTime.Now;
    Prd.CreatedOn = DateTime.Now;

    if (updooption == "Add")
    {
        Result = new tblUsers().InsertProfile(Prd);
    }
    else
    {
        Result = new tblUsers().UpdateProfile(Prd);
    }

    if (Result == true)
    {
        if (updooption == "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");

}
Etetbluser_privacy_chk data = new Etetbluser_privacy_chk();
data.userid = UserId;
data.User_privacy = 0;
data.Details_chk = Details_chk.Checked;
data.Occupation_chk = Occupation_chk.Checked;
data.Occupation_type_chk = Occupation_type_chk.Checked;

```

```
data.Edu_qualification_chk = Edu_qualification_chk.Checked;
data.Martial_status_chk = Martial_status_chk.Checked;
data.Sex_ornt_chk = Sex_ornt_chk.Checked;
data.dob_chk = dob_chk.Checked;
data.tob_chk = tob_chk.Checked;
data.pob_chk = pob_chk.Checked;
data.family_nos_chk = family_nos_chk.Checked;
data.web_url_chk = web_url_chk.Checked;
data.adhaar_id_chk = adhaar_id_chk.Checked;
data.pan_no_chk = pan_no_chk.Checked;
data.weekly_days_chk = weekly_days_chk.Checked;
data.Income_Year_chk = Income_Year_chk.Checked;
data.currency_chk = currency_chk.Checked;
data.Religion_chk = Religion_chk.Checked;
data.Caste_chk = Caste_chk.Checked;
data.Sub_Caste_chk = Sub_Caste_chk.Checked;
data.Location_chk = Location_chk.Checked;
data.Gender_Search_chk = Gender_Search_chk.Checked;
data.Age_from_chk = Age_from_chk.Checked;
data.Age_to_chk = Age_to_chk.Checked;
data.Religion_search_chk = Religion_search_chk.Checked;
data.Caste_search_chk = Caste_search_chk.Checked;
data.Area_search_chk = Area_search_chk.Checked;
data.MTongue_search_chk = MTongue_search_chk.Checked;
data.GeoLoc_search_chk = false;// GeoLocation_chk.Checked;
data.Religions_Choice = false;// Religions_Choice.Checked;
data.Caste_Choice = false;// Caste_Choice.Checked;
data.Areas_Choice = false;// Areas_Choice.Checked;
data.Cities_Choice = false;// Cities_Choice.Checked;
data.Countries_Choice = false;// Countries_Choice.Checked;
data.Languages_Choice = false;// Languages_Choice.Checked;
data.school_chk = school_chk.Checked;
data.about_chk = about_chk.Checked;
data.college_chk=college_chk.Checked ;
data.university_chk=univ_chk.Checked ;
data.Photos_chk=Photo_chk.Checked ;
data.uid = UserId;
bool ret = false;
ret = new tblUsers().GetPrivacyCount(UserId, 0);
if (ret == false)
{
    data.UpdatedOn = DateTime.Now;
    data.CreatedOn = DateTime.Now;
    ret = new tblUsers().InsertUserPrivacyCheck(data);
}
else
{
    data.UpdatedOn = DateTime.Now;
```

```

        ret = new tblUsers().UpdatePrivacyCheck(data);
    }
    if (ret)
    {
        Common.ShowMessageHtml("Profile Data Updated Sucessfully", 1, true, dvMsgBox);
        RefreshScreen();
    }
}
}

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..");
RefreshScreen();
}
protected void Button2_Click(object sender, EventArgs e)
{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    if (Clsusr == null)
    {
        bool ret = new Common().guest_login();
        if (ret == false)
        {
            Response.Redirect("~/Login.aspx");
        }
    }
    Button btn2 = (Button)sender;
    string Id = btn2.CommandName;
    UserId=Convert.ToInt16(Clsusr.UserId);
    bool res = new tblUsers().DeleteAddress(UserId, Id);

}
private void ReduceImageSize(double scaleFactor, string sourcePath, string targetPath)
{
    try
    {
        //using (System.Drawing.Image image = System.Drawing.Image.FromStream(sourcePath))
        using (System.Drawing.Image image = System.Drawing.Image.FromFile(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);
    }

}
catch { }
}
protected void Button3_Click(object sender, EventArgs e)
{
    //Common.ShowMessageHtml("School button clicked", 0, true, dvMsgBox);
    popup_head.InnerHtml = "Enter School Details";
    namescu.InnerText = "School Name";
    ObjPD.BindDropDowns(ddl_City, (new PopDlls()).GetCities());
    ObjPD.BindDropDowns(ddl_country, (new PopDlls()).GetCountries());
    txt_name.Text = string.Empty;
    Session["CAN_SUSP"] = "School";
    popup.Show();
}
protected void Button5_Click(object sender, EventArgs e)
{
    popup_head.InnerHtml = "Enter College Details";
    namescu.InnerText = "College Name";
    ObjPD.BindDropDowns(ddl_City, (new PopDlls()).GetCities());
    ObjPD.BindDropDowns(ddl_country, (new PopDlls()).GetCountries());
    txt_name.Text = string.Empty;
    Session["CAN_SUSP"] = "College";
    popup.Show();
}
protected void Button6_Click(object sender, EventArgs e)
{
    popup_head.InnerHtml = "Enter University Details";
    namescu.InnerText = "University Name";
    ObjPD.BindDropDowns(ddl_City, (new PopDlls()).GetCities());
    ObjPD.BindDropDowns(ddl_country, (new PopDlls()).GetCountries());
    txt_name.Text = string.Empty;
    Session["CAN_SUSP"] = "University";
    popup.Show();
}
protected void btnSave_Click(object sender, EventArgs e)

```



```

{
    ClsLoggedInUser Clsusr = (ClsLoggedInUser)Session["LoggedInUser_Specs"];
    if (Clsusr == null)
    {
        bool ret = new Common().guest_login();

    }
    int rets = 0;
    //save school college university details
    string savetype = Session["CAN_SUSP"].ToString();
    EteEdInst edata = new EteEdInst();
    edata.Name = txt_name.Text;
    edata.City = ddl_City.SelectedIndex;
    edata.Country = ddl_cntry.SelectedIndex;
    edata.ZipCode = txt_zipcode.Text;
    edata.CreatedON = DateTime.Now;
    edata.UserId = Convert.ToInt16(Clsusr.UserId);
    if (savetype == "School")
    {
        rets = new EdInstitutions().InsertSchoolsList(edata);
    }
    else if (savetype == "College")
    {
        rets = new EdInstitutions().InsertCollegesList(edata);
    }
    else if (savetype == "University")
    {
        rets = new EdInstitutions().InsertUniversityList(edata);
    }
    if (rets > 0)
    {
        ObjPD.BindDropDowns(ddl_college, (new PopDlls()).GetCollegesList());
        ObjPD.BindDropDowns(ddl_university, (new PopDlls()).GetUniverssityList());
        ObjPD.BindDropDowns(ddl_school, (new PopDlls()).GetSchoosList());
        if (savetype == "School")
        {
            ddl_school.SelectedIndex = rets;
        }
        else if (savetype == "College")
        {
            ddl_college.SelectedIndex = rets;
        }
        else if (savetype == "University")
        {
            ddl_university.SelectedIndex = rets;
        }
    }
}

```

```

        Common.ShowMessageHtml(savetype + " Data Saved", 1, true, dvMsgBox);
        txt_name.Text = "";

    }
    popup.Hide();
    //ObjPD.BindDropDowns(ddl_college, (new PopDlIs()).GetCollegesList());
    //ObjPD.BindDropDowns(ddl_university, (new PopDlIs()).GetUniverssityList());
    //ObjPD.BindDropDowns(ddl_school, (new PopDlIs()).GetSchoosList());
}
}

***** UserList.aspx*****
<%@ Page Title="" Language="C#" MasterPageFile="~/DOWMasterPage.master"
AutoEventWireup="true" CodeFile="UserList.aspx.cs" Inherits="View_UserList" %>

<%@ 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>
    </asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentMainPage" runat="Server">

    <div class=" col-lg-12 ml-5" 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="col-lg-12"> <!--container pl-5 ml-5-->

        <!--<asp:ScriptManager ID="ScriptManager1" runat="server"></asp:ScriptManager>-->

        <ig:WebDataGrid ID="WDGrid_UserList" runat="server" AutoGenerateColumns="False"> <!--
        CssClass="wdglist" -->
            <Columns>

                <ig:BoundDataField DataFieldName="UserId" Key="UserId " Width="100px" >
                    <Header Text="User Id">
                    </Header>
                </ig:BoundDataField>
                <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="LastName" Key="LastName">
                    <Header Text="Last Name">
                    </Header>
                </ig:BoundDataField>
                <ig:BoundDataField DataFieldName="EmailID" Key="EmailID">
                    <Header Text="Email Id">
                    </Header>
                </ig:BoundDataField>

                <ig:BoundDataField DataFieldName="ContactNum" Key="ContactNum">
                    <Header Text="Contact Number">
                    </Header>
                </ig:BoundDataField>
                <ig:BoundDataField DataFieldName="GroupName" Key="GroupName" Width="100px">
                    <Header Text="Primary Choice">
                    </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="Status" Key="Status" Width="70px">
            <Header Text="Is Active">
            </Header>
        </ig:BoundDataField>

        <ig:TemplateDataField Key="userid" Width="130px">
            <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>
        <%--
        <ig:BoundDataField DataFieldName="DealerName" Key="DealerName"
Width="10px">
            <Header Text="Dealer Name">
            </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 [UserId], [UserName],
[FirstName], [LastName], [EmailID], [ContactNum], [CreatedOn], [Status] FROM
[tblUser]"></asp:SqlDataSource>
    </div>
</div>
</div>
</asp:Content>
//user list for admin use

```

| User Id | User Name | First Name | Last Name | Email Id | Contact Number | Primary Choice | Created On | Is Active | Open |
|---------|------------|--------------|-----------|---------------------------|----------------|------------------|------------|-----------|------|
| 2 | Guest | Guest | guest | gust@dow.com | 949499 | Guest | 01-01-2018 | True | Open |
| 3 | Business | ravi kumar | ravikiran | ravikpcom@gmail.com | 585858 | Others | 20-02-2018 | True | Open |
| 4 | Staff | staff | staff | staff@b2bportal.com | 999999 | Staff | 01-01-2018 | True | Open |
| 5 | Buyer | Buyer | Buyer | Buyer | 999999 | Wedding | 01-01-2018 | True | Open |
| 6 | Vendor | Vendor | Vendor | vendor@b2bportal.com | 999999 | Dating | 05-05-2018 | True | Open |
| 7 | Shopping | ravi k p | acdf | ravi_kp_com@yahoo.com | 9999999 | Friends List | 22-11-2019 | True | Open |
| 8 | service | ravi outlook | outlook | ravi_kp_com@outlook.com | 848484848 | Dating & Wedding | 22-11-2019 | True | Open |
| 9 | 89999999 | Sanjana | Desai | sanjana.desai@dow.com | 89999999 | Wedding | 14-01-2020 | True | Open |
| 10 | 12345678 | Albert | Einestine | albert.einestine@dow.com | 12345678 | Wedding | 08-05-2021 | True | Open |
| 11 | 7893642582 | Shivani | Kapoor | shivani.kpoo@dow.com | 7893642582 | Wedding | 08-05-2021 | True | Open |
| 12 | 8885269384 | Renu Sree | Sakaria | Renu.Sree.sakaria@dow.com | 8885269384 | Friends List | 08-05-2021 | True | Open |
| 13 | 9291355392 | Staff | Staff | staff@dow.com | 9291355392 | Staff | 08-05-2021 | True | Open |

***** 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 DOWEntity;
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("Login.aspx");
}
//}
}
```

```
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;
    }

}

}

*****VendorHome.aspx *****
<%@ Page Title="" Language="C#" MasterPageFile="~/VendorMasterPage.master"
AutoEventWireup="true" CodeFile="VendorHome.aspx.cs" Inherits="VendorHome" %>

<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="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>

  </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 DOWEntity;
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)
        {
            if (Convert.ToInt16(Clsusr.GroupId) == 3 || Convert.ToInt16(Clsusr.GroupId) == 1)

```

```

    {
    }
    else
    {
        Response.Redirect("~/Login.aspx");
    }

    string msg = Request.QueryString["msg"];
    if (msg != null)
    {
        Common.ShowMessageHtml(msg, 1, true, dvMsgBox);
    }
    else
    {
        bool ret = new Common().guest_login();

        if (ret == false)
        {
            Response.Redirect("~/Login.aspx");
        }
        else
            if (Convert.ToInt16(Clsusr.GroupId) == 3 || Convert.ToInt16(Clsusr.GroupId) == 1)
            {
            }
            else
            {
                Response.Redirect("~/Login.aspx");
            }
        }
    }
}

```

}

This concludes the Windows of the application

The following is a link to download css files used in the application, unzip and copy folders to Application folder

<https://www.dropbox.com/scl/fi/gv3ytznmefgzicoqfhe1t/DOWCssFolders.zip?rlkey=a6tq31f1d0yqpsrpsnp8bnhnm&st=bj79j5pa&dl=0>

The following is the link for signalchat.dll used in the application, download and copy to bin folder

<https://www.dropbox.com/scl/fi/klhq206s42gxcl5zcxszn/SignalRChat.dll?rlkey=6zq6gz35eihtgfee03mb0y1x0&st=kccuz695&dl=0>