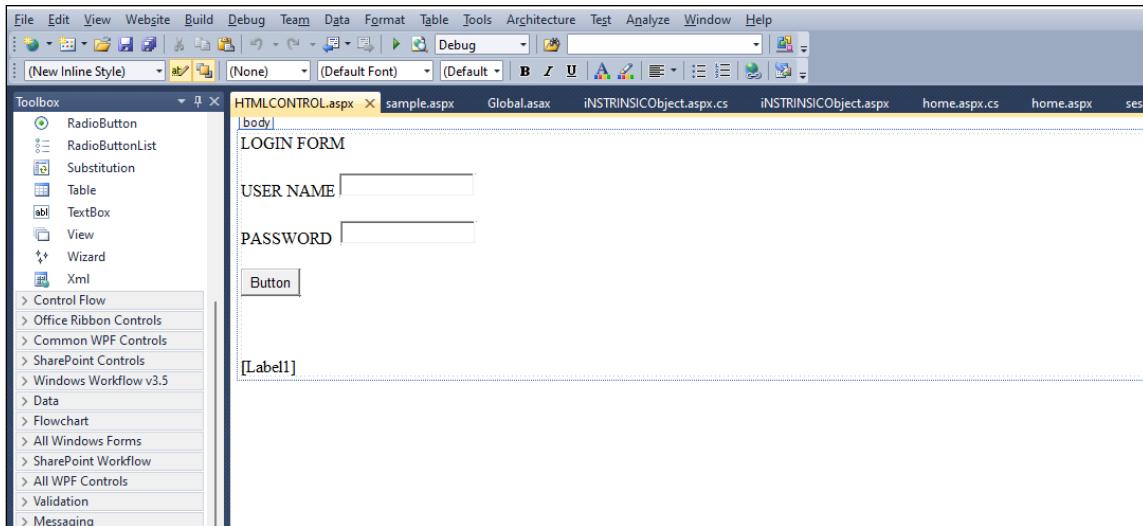


Assignment No.1:
Write an ASP.Net Program to Demonstrate Use of HTML Controls.

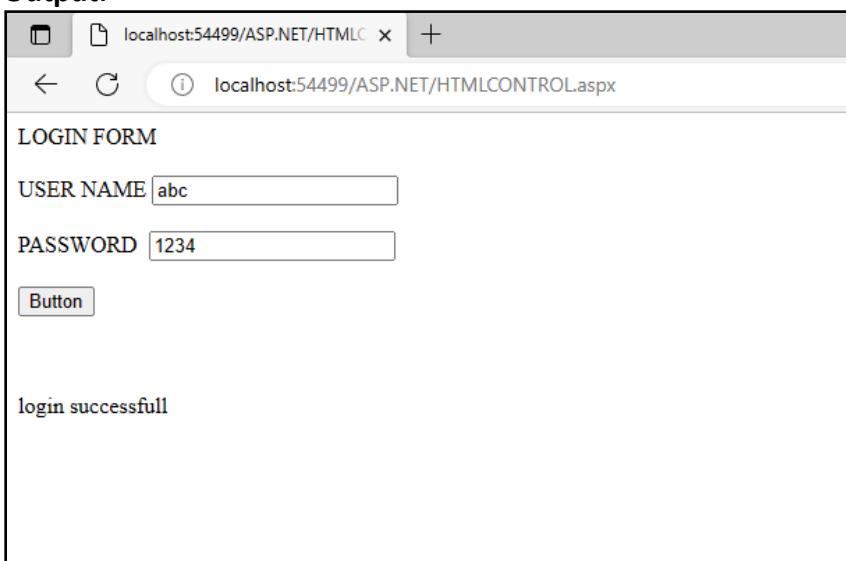
Step1:Type equation here.



Button Coding:

```
if (Text1.Value == "abc" && Text2.Value == "1234")
{
    Label1.Text = "login successfull";
}
else
{
    Label1.Text = "user name and password incorrect";
}
```

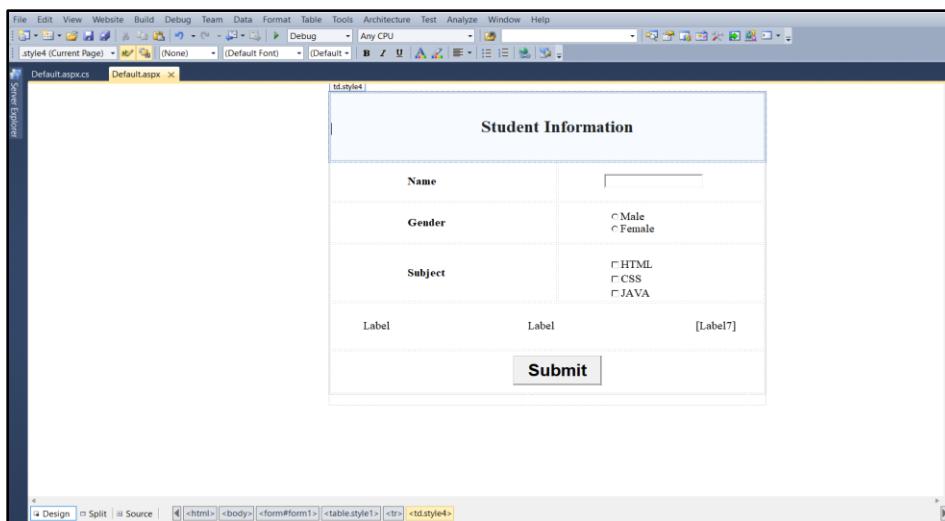
Output:



Assignment No.2:

Write an ASP.Net Program to Demonstrate Use of Web Controls.

Default.aspx :-



Default.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 _Default : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {

    }

    protected void Button1_Click(object sender, EventArgs e)
    {
        Label5.Text = TextBox1.Text;
        if (RadioButton1.Checked == true)
        {
            Label6.Text = RadioButton1.Text;
        }
        else
        {
            Label6.Text = RadioButton2.Text;
        }
        foreach (ListItem Item in CheckBoxList1.Items)
        {
            if (Item.Selected)
            {
                Label7.Text += Item.Text + " ";
            }
        }
    }
}
```

Output :-

The screenshot shows a web browser window with the URL `localhost:51660/Diksha/Default.aspx`. The page title is "Student Information". The form contains the following fields:

- Name:** Ram Patil (input field)
- Gender:** Male (radio button selected)
- Subject:** HTML, JAVA (checkboxes selected)
- Submitted Data:** Ram Patil, Male, HTML JAVA
- Submit:** A button labeled "Submit".

Assignment No 3:

Write an ASP.Net Program to Returns The Windows Name of Your Computer And URL of the Page That You are Visiting.

Default.aspx.cs file

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

public partial class _Default : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        string Computername = Environment.MachineName;string
        currentUrl = Request.Url.ToString();

        Response.Write("Computer name:" + Computername + "<br>");
        Response.Write("current URL" + currentUrl + "<br>");

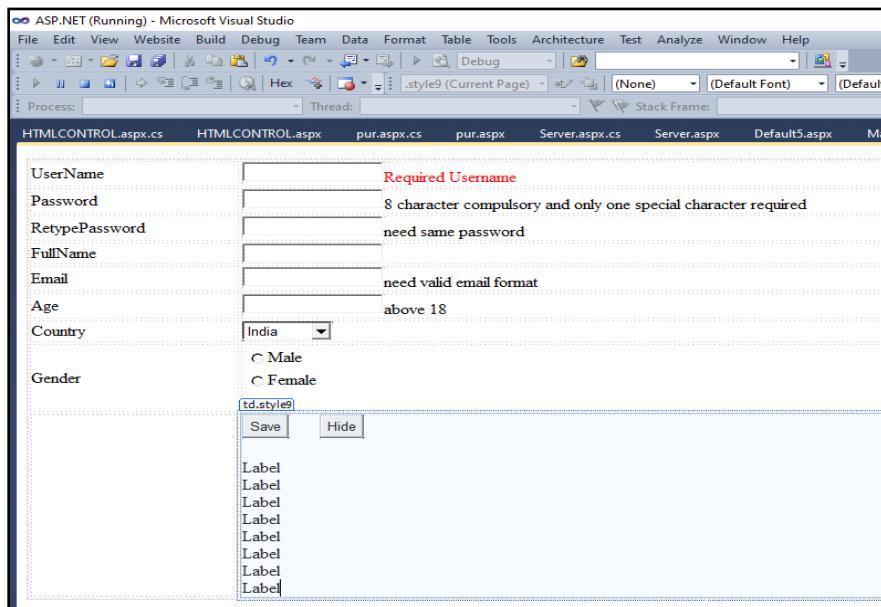
    }
}
```

Output:



Assignment No.4:

Write an ASP.Net Program to Demonstrate Use of Validation Controls.



Save Button Coding:

```
protected void btnSave_Click(object sender, EventArgs e)
{
    Label1.Text = txtUserName.Text;
    Label2.Text = txtPassword.Text;
    Label3.Text = txtRtypePassword.Text;
    Label4.Text = txtFullName.Text;
    Label5.Text = txtEmail.Text;
    Label6.Text = txtAge.Text;
}
protected void Button1_Click(object sender, EventArgs e)
{
    Label1.Visible = false;
    Label2.Visible = false;
    Label3.Visible = false;
    Label4.Visible = false;
    Label5.Visible = false;
    Label6.Visible = false;
}
```

Output:

The screenshot shows a web browser window with the URL `localhost:54499/ASP.NET/Validation.aspx`. The page displays a form with the following fields and values:

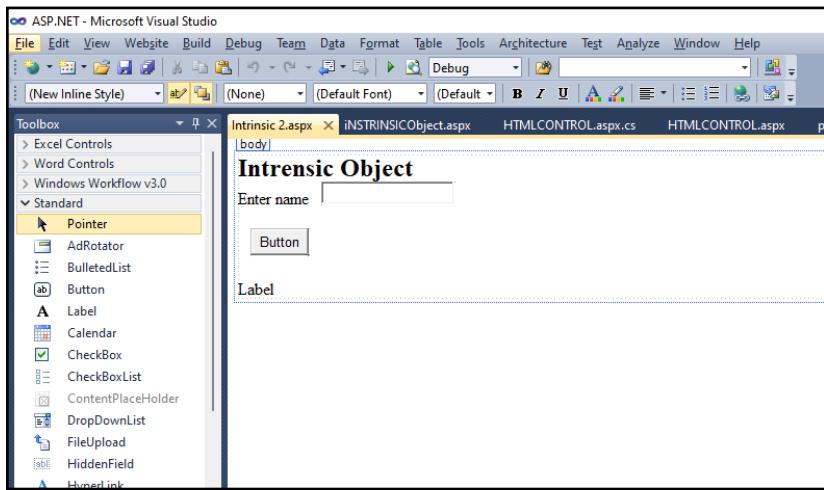
UserName	RCP Imrd
Password	(empty)
RetypePassword	SaiRam
FullName	SaiRam
Email	sairam@gmail.com
Age	25
Country	India
Gender	<input checked="" type="radio"/> Male <input type="radio"/> Female

Below the form are two buttons: `Save` and `Hide`. Underneath the buttons, the following text is displayed:

RCP Imrd
123456789@
123456789@
SaiRam
sairam@gmail.com
25
Label
Label

Assignment No.5:

Write an ASP.Net Program to Demonstrate Use of Intrinsic Objects.



Save Button Coding:

```
Label1.Text="welcome" +Server.HtmlEncode(TextBox1.Text) +"<br/> The  
URL is:" +Server.UrlEncode(Request.Url.ToString());
```

Output:



AssignmentNo6.

Write an Asp.net Program to demonstrate Application and session scope variable using Global.asax file.

Default. aspx :



The screenshot shows the Microsoft Visual Studio IDE. The main window displays the 'Default.aspx' page in 'Design' view. The code-behind for the page is visible in the 'Source' tab:

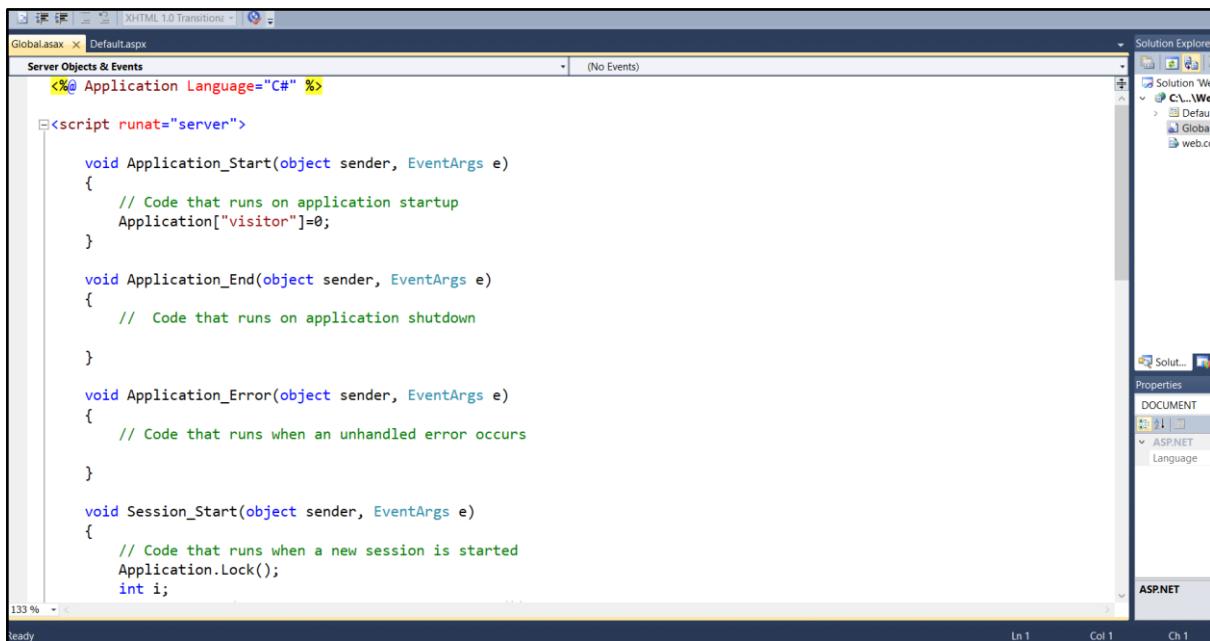
```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Default.aspx.cs" Inherits="_Default" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            <p>there are <%Response.Write(Application["visitor"]); %> online now!</p>
        </div>
    </form>
</body>
</html>
```

The Solution Explorer on the right shows a project structure with files like 'Default.aspx', 'Global.asax', and 'web.config'. The Properties and Tools palettes are also visible on the right side of the interface.

Global.asax :



The screenshot shows the Microsoft Visual Studio IDE. The main window displays the 'Global.asax' page in 'Design' view. The code-behind for the application is visible in the 'Source' tab:

```
<%@ Application Language="C#" %>

<script runat="server">
    void Application_Start(object sender, EventArgs e)
    {
        // Code that runs on application startup
        Application["visitor"]=0;
    }

    void Application_End(object sender, EventArgs e)
    {
        // Code that runs on application shutdown
    }

    void Application_Error(object sender, EventArgs e)
    {
        // Code that runs when an unhandled error occurs
    }

    void Session_Start(object sender, EventArgs e)
    {
        // Code that runs when a new session is started
        Application.Lock();
        int i;
```

The Solution Explorer on the right shows a project structure with files like 'Default.aspx', 'Global.asax', and 'web.config'. The Properties and Tools palettes are also visible on the right side of the interface.

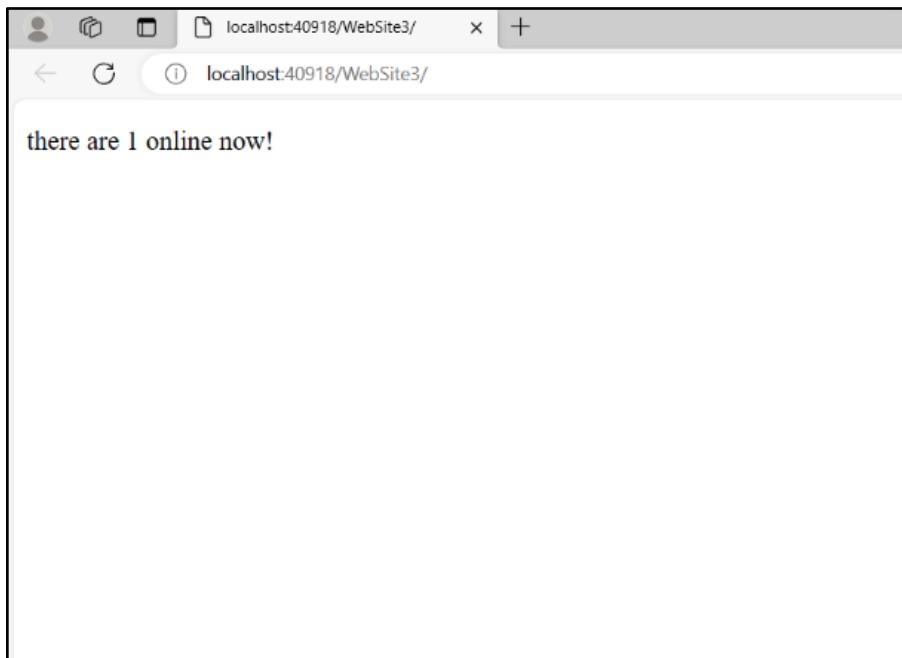
The screenshot shows the Global.asax file in Visual Studio. It contains two event handlers: Session_Start and Session_End. The Session_Start handler increments a counter in the Application object and stores it back. The Session_End handler decrements the counter and stores it back. The code uses Application.Lock() and Application.UnLock() to ensure thread safety.

```
void Session_Start(object sender, EventArgs e)
{
    // Code that runs when a new session is started
    Application.Lock();
    int i;
    i = int.Parse(Application["visitor"].ToString());
    i = i + 1;
    Application["visitor"] = i.ToString();
    Application.UnLock();
}

void Session_End(object sender, EventArgs e)
{
    // Code that runs when a session ends.
    // Note: The Session_End event is raised only when the sessionstate mode
    // is set to InProc in the Web.config file. If session mode is set to StateServer
    // or SQLServer, the event is not raised.
    Application.Lock();
    int i;
    i = int.Parse(Application["visitor"].ToString());
    i = i - 1;
    Application["visitor"] = i.ToString();
    Application.UnLock();
}

</script>
```

Output :

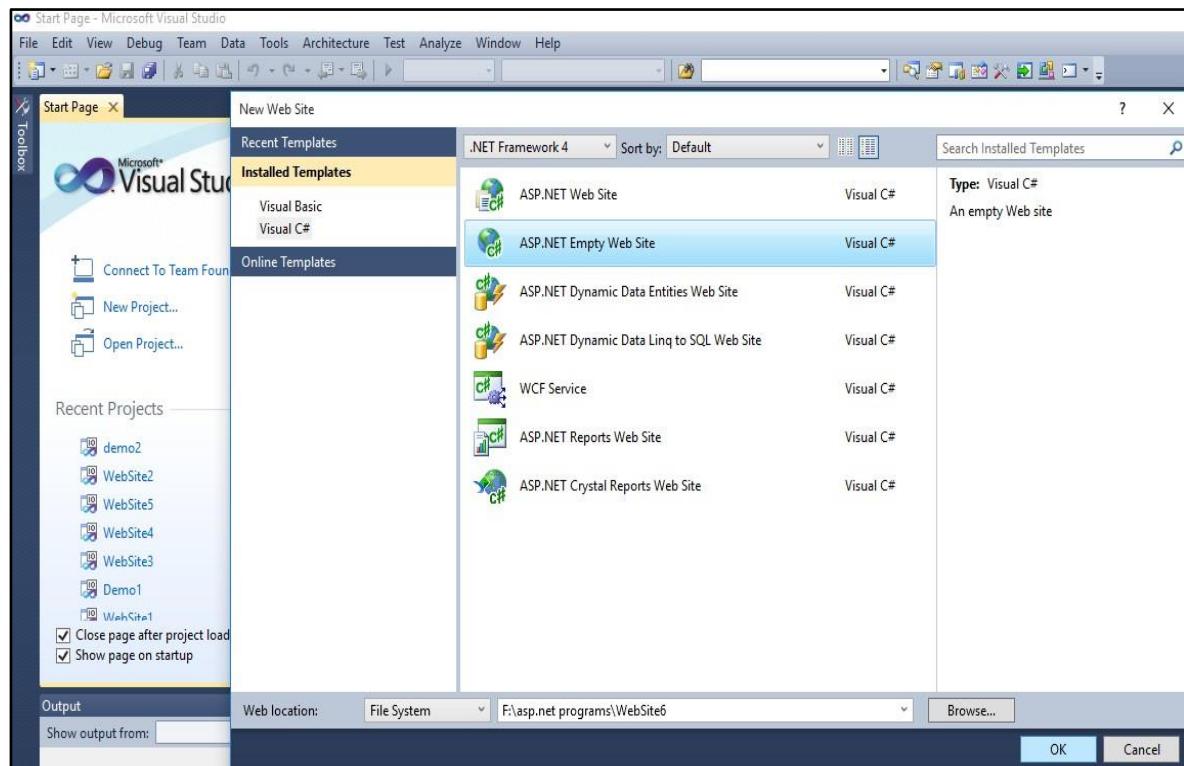


Assignment No.7:

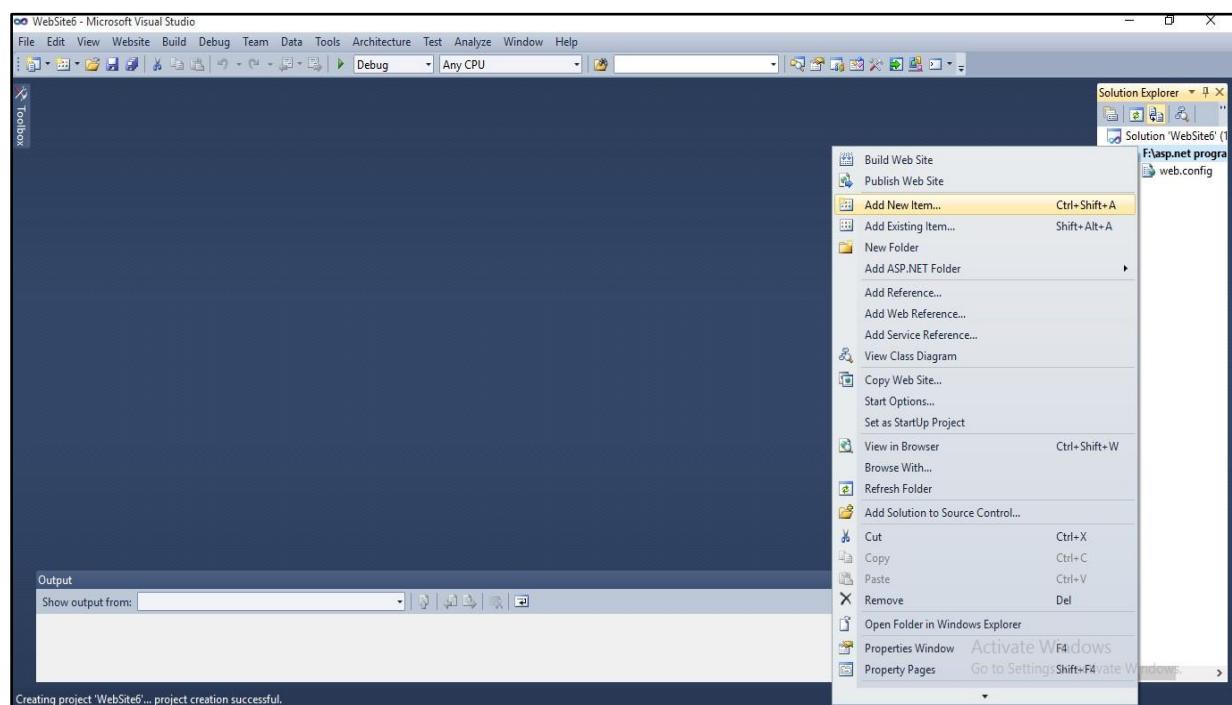
Write an ASP.Net Program to Demonstrate Use of Master Page.

Master page

Step1:-First You create ASP.NET empty website

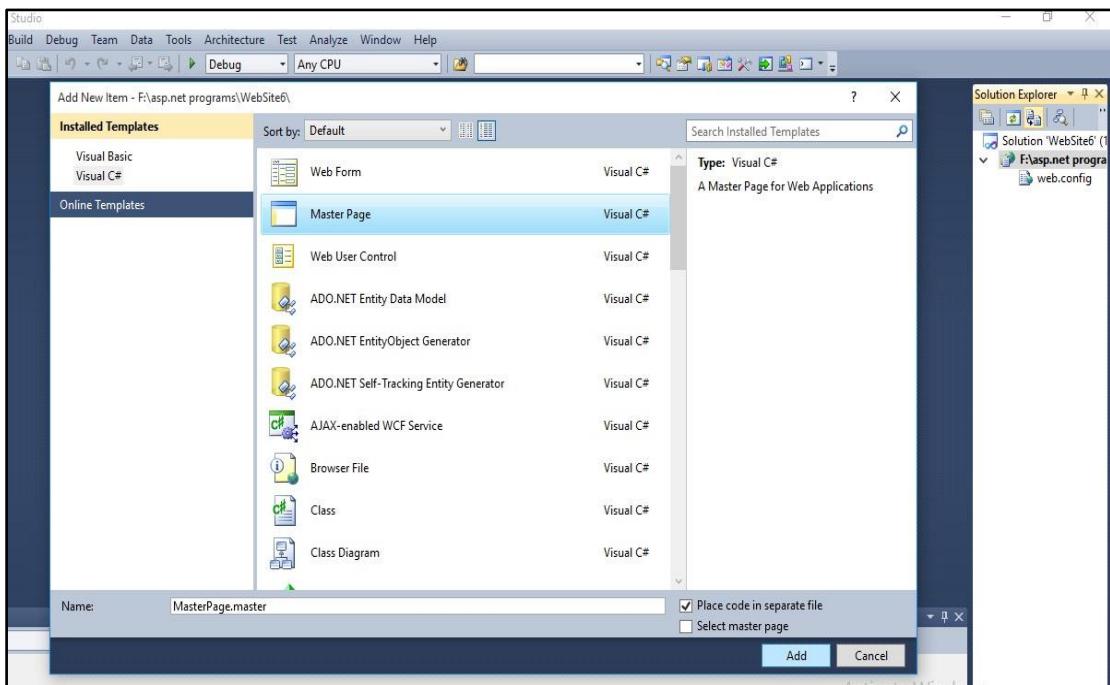


Step2:-In Solution Explorer, right click on the name of your website and click on 'add new item'.



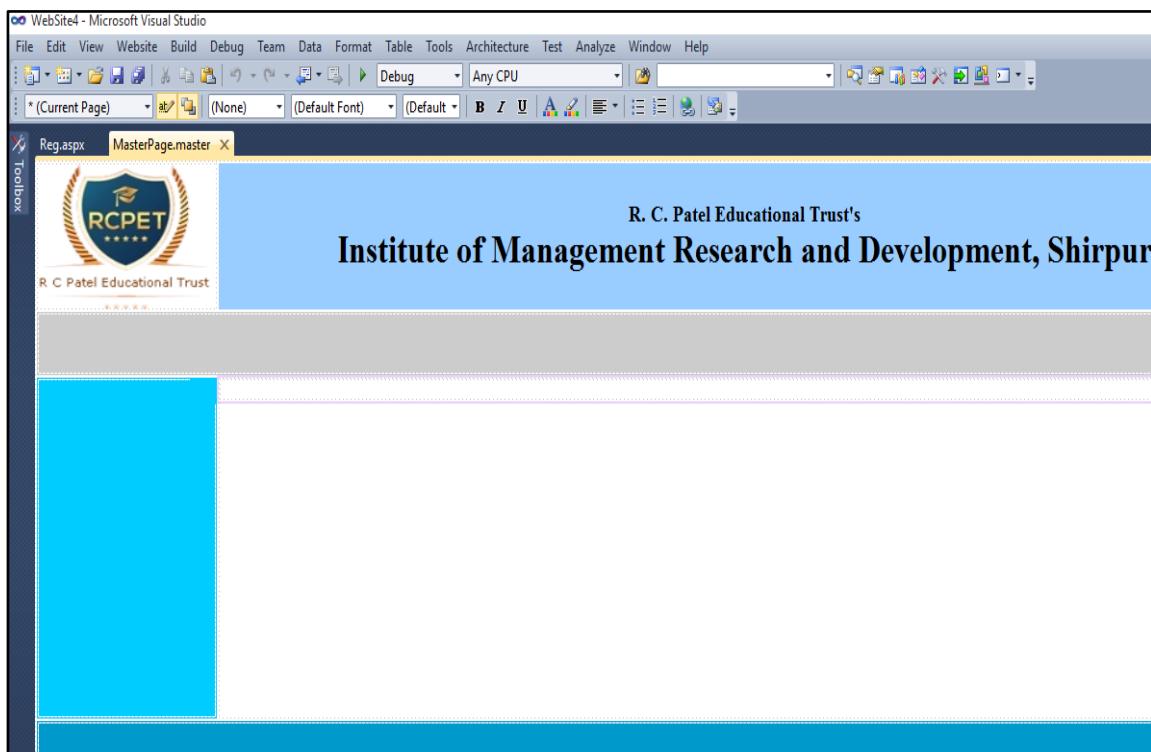
Step 3:-Under Visual Studio Template, click on master page and In name box type the name if you want to give it

Step4:-Then Click on add button



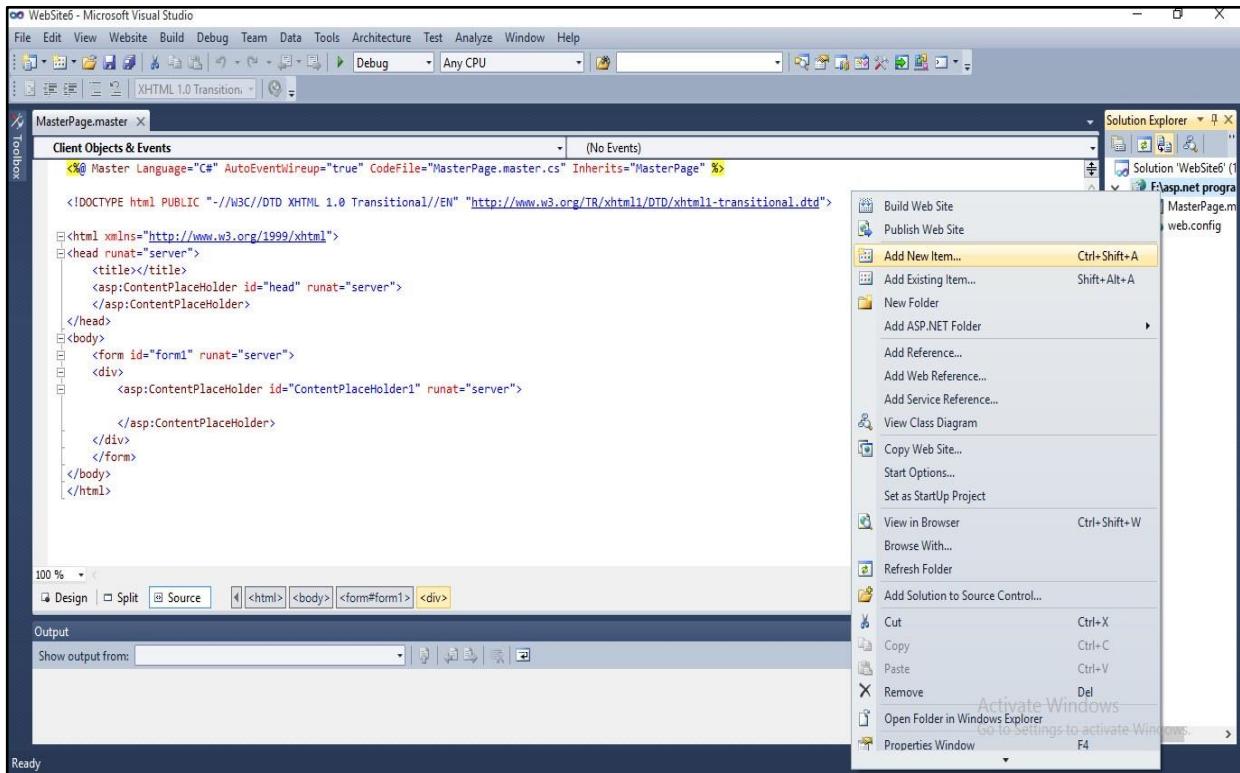
Step5:-Then new master page opens in source view and Design it.

Step6:-Add the Content Placeholder control in content section of master page then Save it.



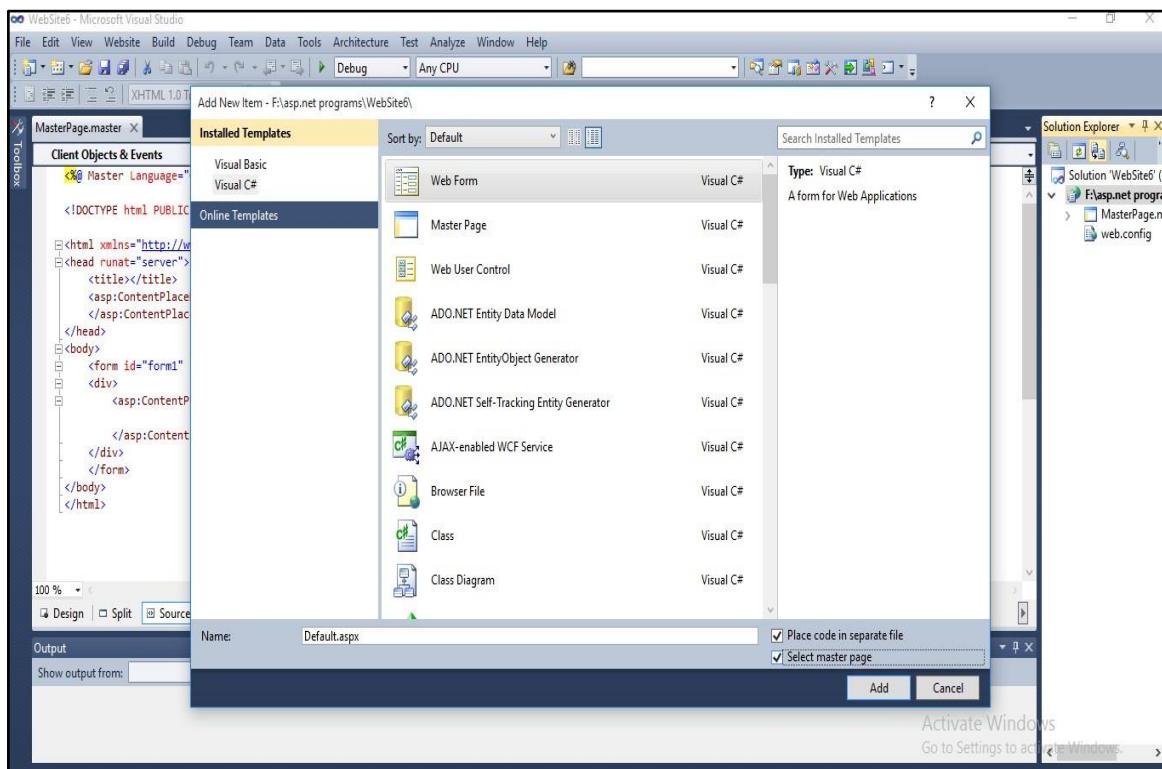
Content Page

Step1:-InSolutionExplorer,rightclickonthenameofyourwebsiteandclickon‘addnewitem’.

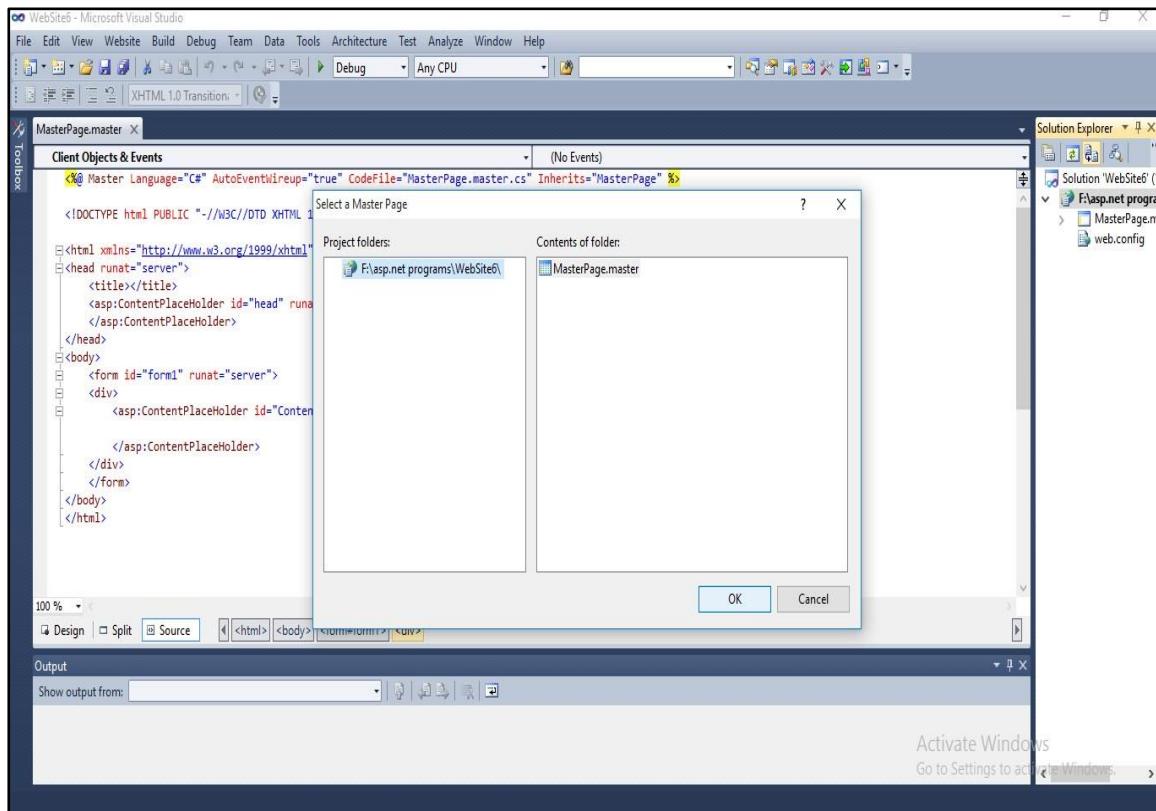


Step 2:-Under Visual Studio Template, click on Web form and In name box type thenameif youwanttогiveit.

Step3:Selectthe‘Select masterpage’checkboxandthenclickon addbutton.

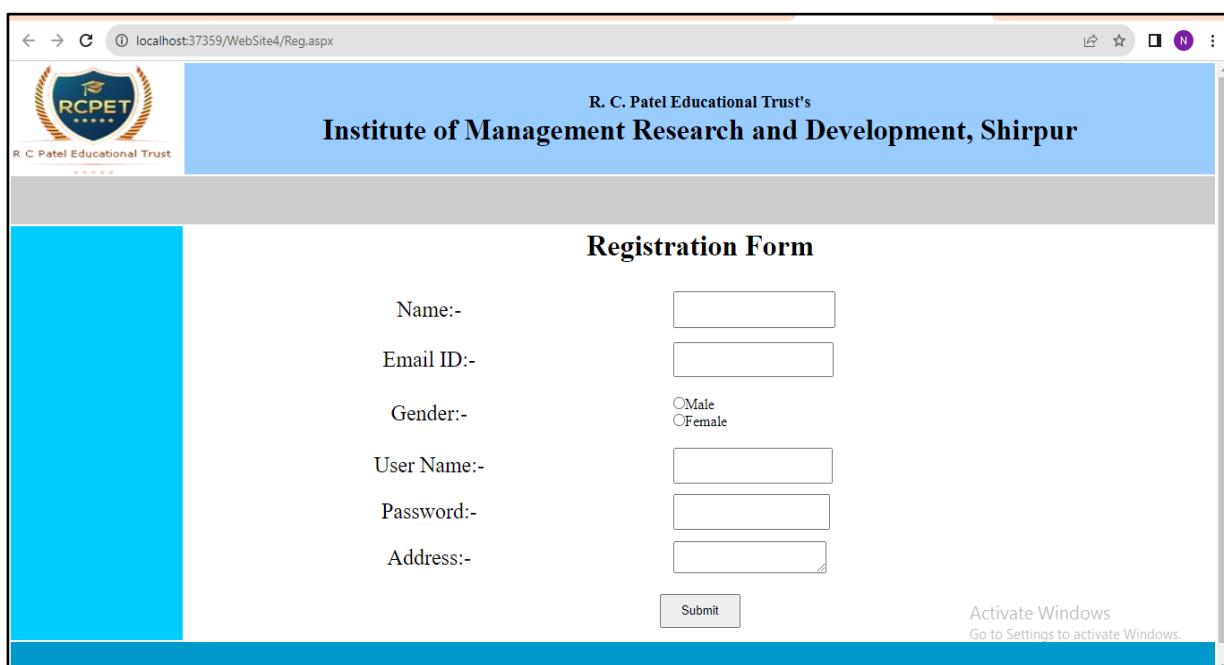


Step 4:- The select master page dialog box appears then click on .masert page and then click



onokbuuton

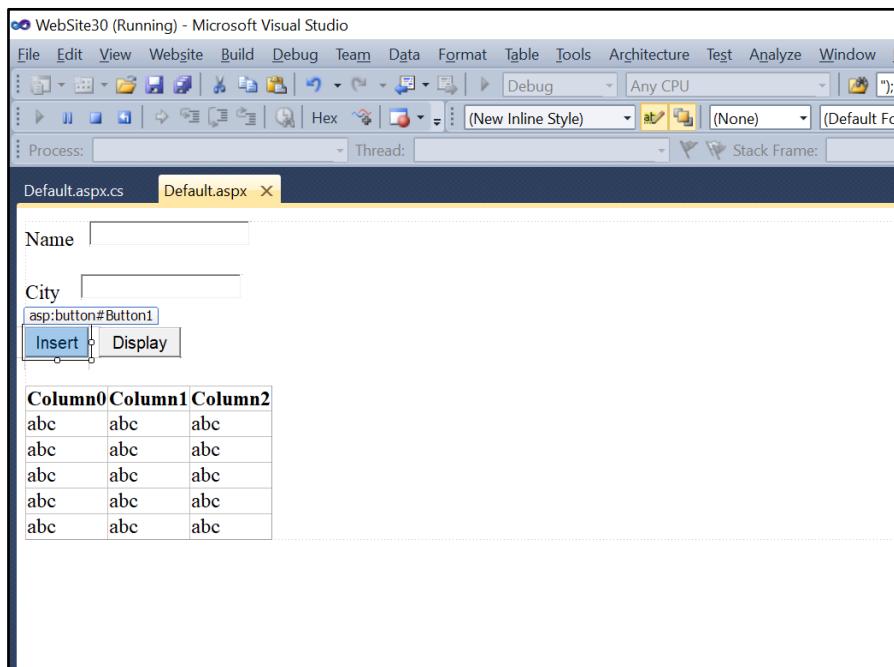
Step5:- Then new.aspx file is created and you can add content in this file.



Assignment No.8:

Write an ASP.Net Page that used The Connection object to connect the Database And Display Information using Datagridview Controls.

Default.aspx



Default.aspx.cs

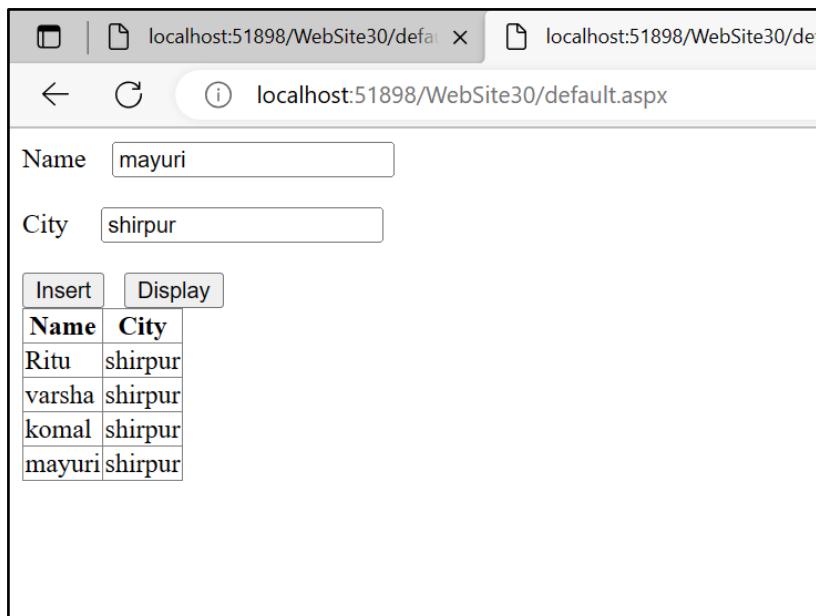
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data.OleDb;
using System.Data;
public partial class Default : System.Web.UI.Page
{
    OleDbConnection con =
        new OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data
Source=C:\\\\Users\\\\HP\\\\Documents\\\\ritu\\\\Database3.accdb");
    protected void Page_Load(object sender, EventArgs e)
    {
    }

    //Insert Data
    protected void Button1_Click(object sender, EventArgs e)
    {
        con.Open();
        OleDbCommand cmd = new OleDbCommand("insert into t1 values('" + TextBox1.Text
        + "','" + TextBox2.Text + "')", con);
        cmd.ExecuteNonQuery();

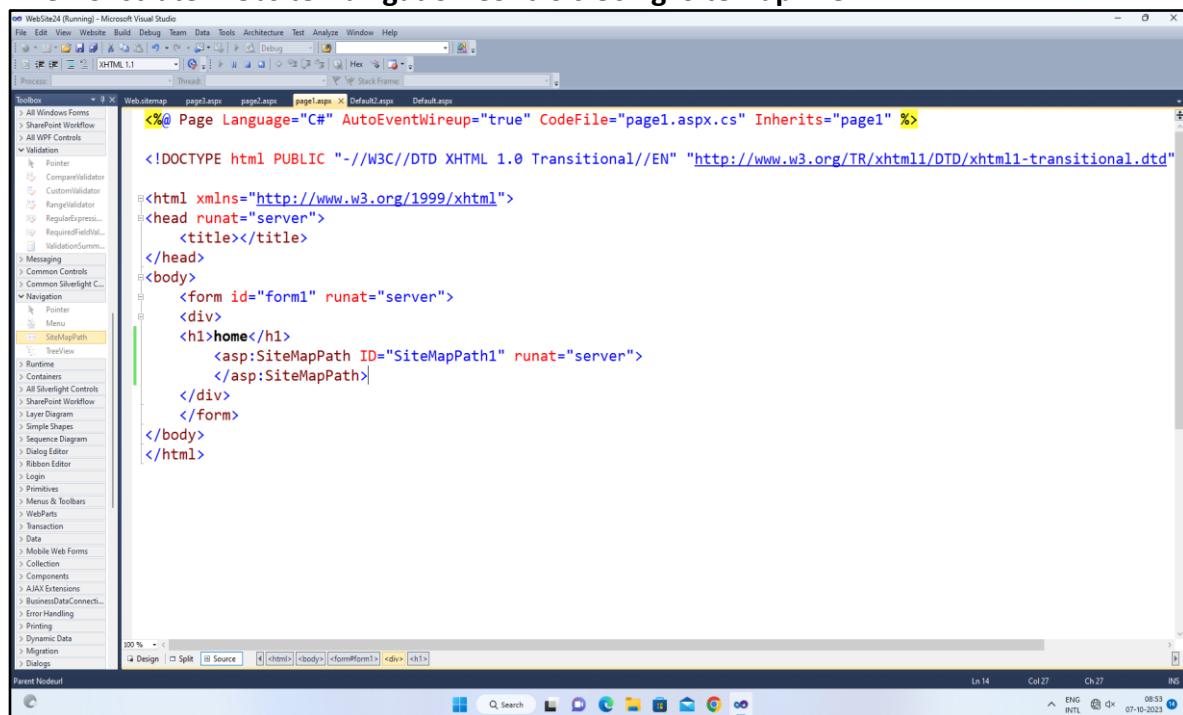
        con.Close();
        Response.Write("<script LANGUAGE='JavaScript'>alert('Data insert
successfully...!')</script>");
    }
}
```

```
        }
    //Display
    protected void Button2_Click(object sender, EventArgs e)
    {
        DataTable dt = new DataTable();
        con.Open();
        OleDbDataAdapter da = new OleDbDataAdapter("select * from t1", con);
        da.Fill(dt);
        GridView1.DataSource = dt;
        GridView1.DataBind();
        con.Close();
    }
}
```

Output :



Assignment No.9: Demonstrate Website Navigation Controls Using SiteMap File.

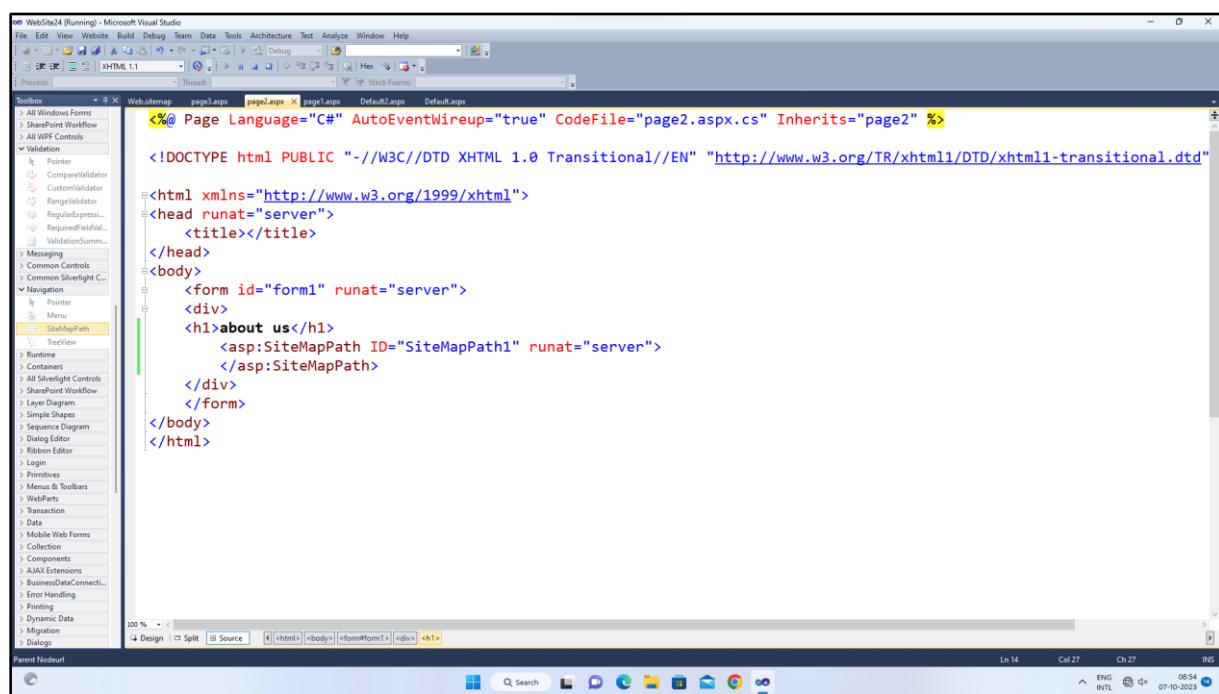


The screenshot shows the Microsoft Visual Studio interface with the title bar "WebSite24 (Running) - Microsoft Visual Studio". The menu bar includes File, Edit, View, Website, Build, Debug, Team, Data, Tools, Architecture, Test, Analyze, Window, Help. The toolbar has standard icons for file operations. The main window displays the code for "page1.aspx". The code uses ASP.NET syntax with C# code-behind. It includes a `<asp:SiteMapPath ID="SiteMapPath1" runat="server">` control within a form. The code is as follows:

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="page1.aspx.cs" Inherits="page1" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd"

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            <h1>home</h1>
            <asp:SiteMapPath ID="SiteMapPath1" runat="server">
            </asp:SiteMapPath>
        </div>
    </form>
</body>
</html>
```



The screenshot shows the Microsoft Visual Studio interface with the title bar "WebSite24 (Running) - Microsoft Visual Studio". The menu bar includes File, Edit, View, Website, Build, Debug, Team, Data, Tools, Architecture, Test, Analyze, Window, Help. The toolbar has standard icons for file operations. The main window displays the code for "page2.aspx". The code uses ASP.NET syntax with C# code-behind. It includes a `<asp:SiteMapPath ID="SiteMapPath1" runat="server">` control within a form. The code is as follows:

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="page2.aspx.cs" Inherits="page2" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd"

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            <h1>about us</h1>
            <asp:SiteMapPath ID="SiteMapPath1" runat="server">
            </asp:SiteMapPath>
        </div>
    </form>
</body>
</html>
```

```

<%@ Page Language="C#" AutoEventWireup="true" CodeFile="page3.aspx.cs" Inherits="page3" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            <h1>help</h1>
            <asp:SiteMapPath ID="SiteMapPath1" runat="server">
            </asp:SiteMapPath>
        </div>
    </form>
</body>
</html>

```

```

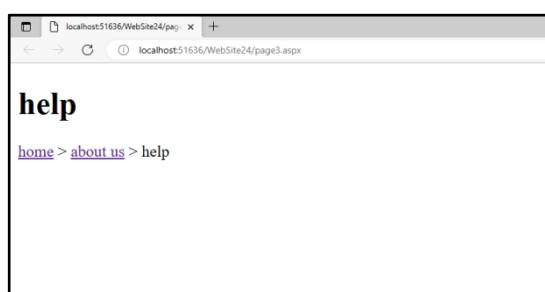
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Default2.aspx.cs" Inherits="Default2" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
</head>
<body>
    <div>
        <asp:SiteMapPath ID="SiteMapPath1" runat="server">
        </asp:SiteMapPath>
    </div>
</body>
</html>

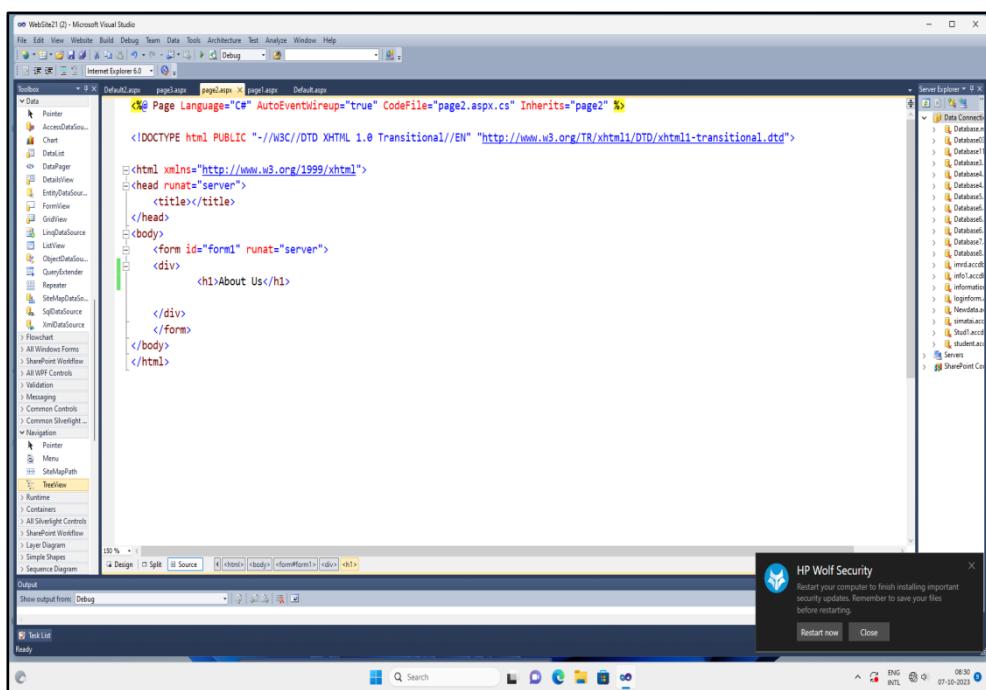
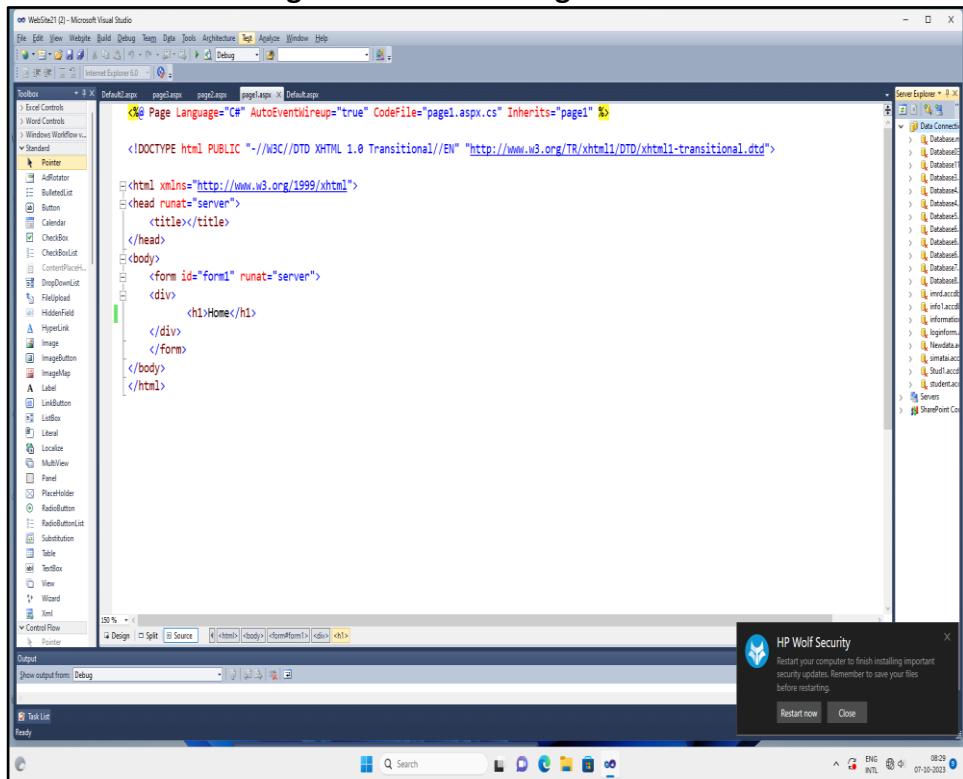
```

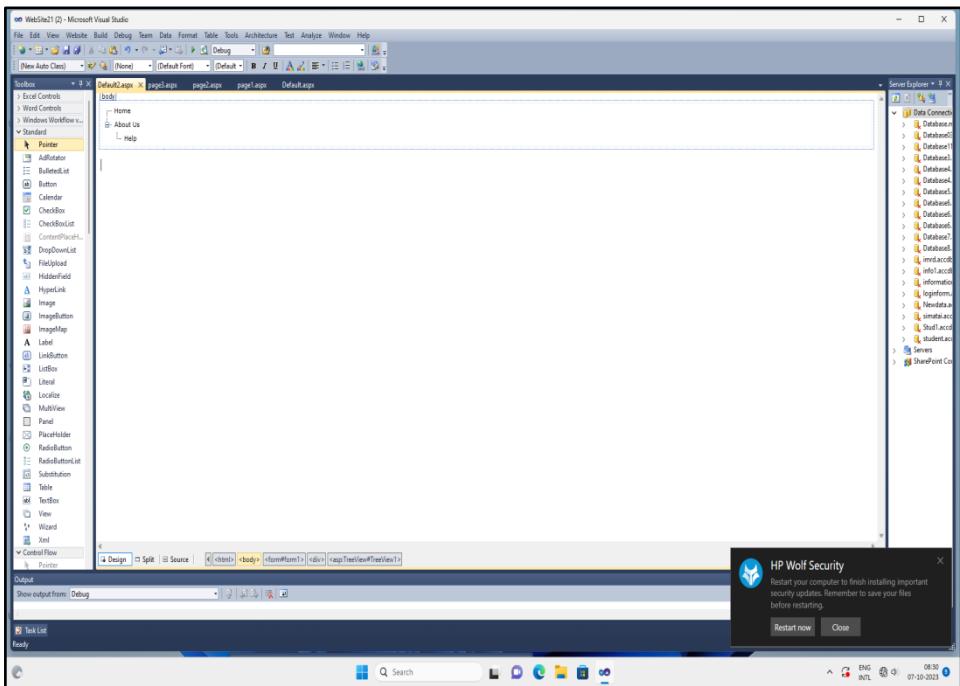
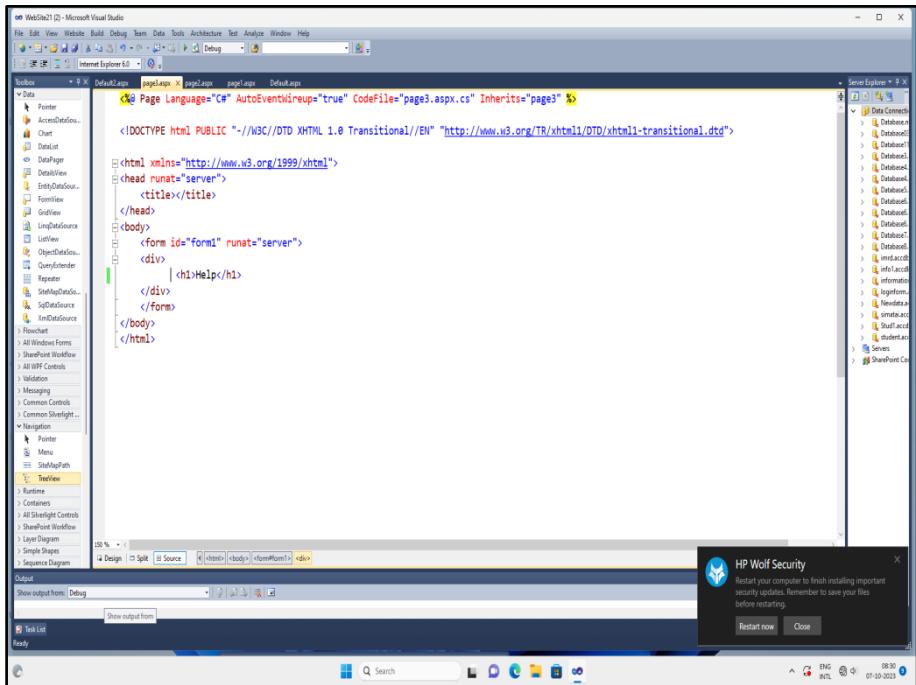
Output:

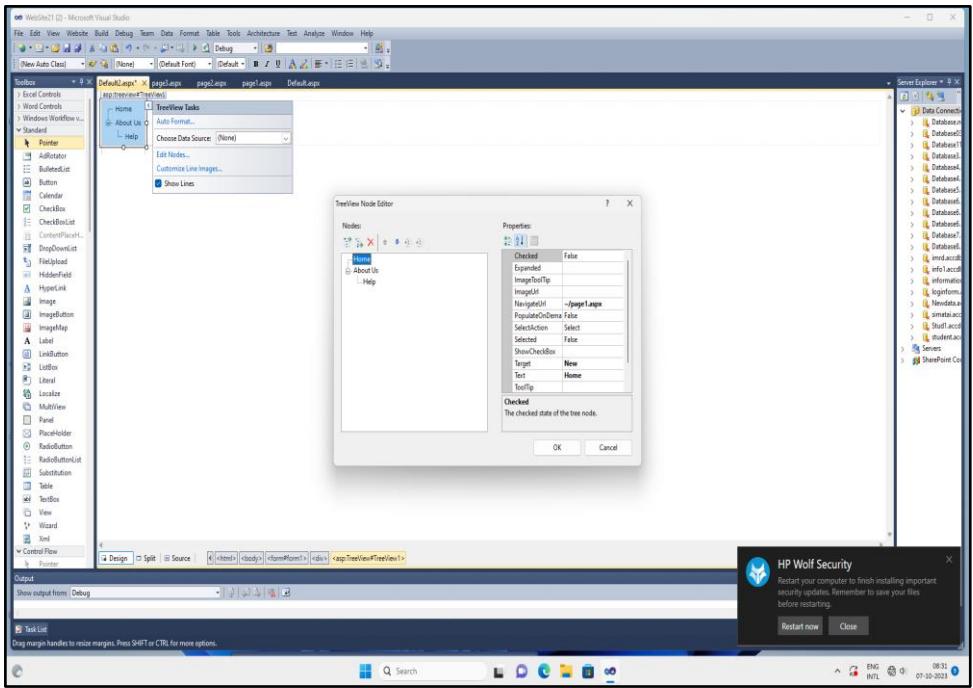


Assignment No.10:

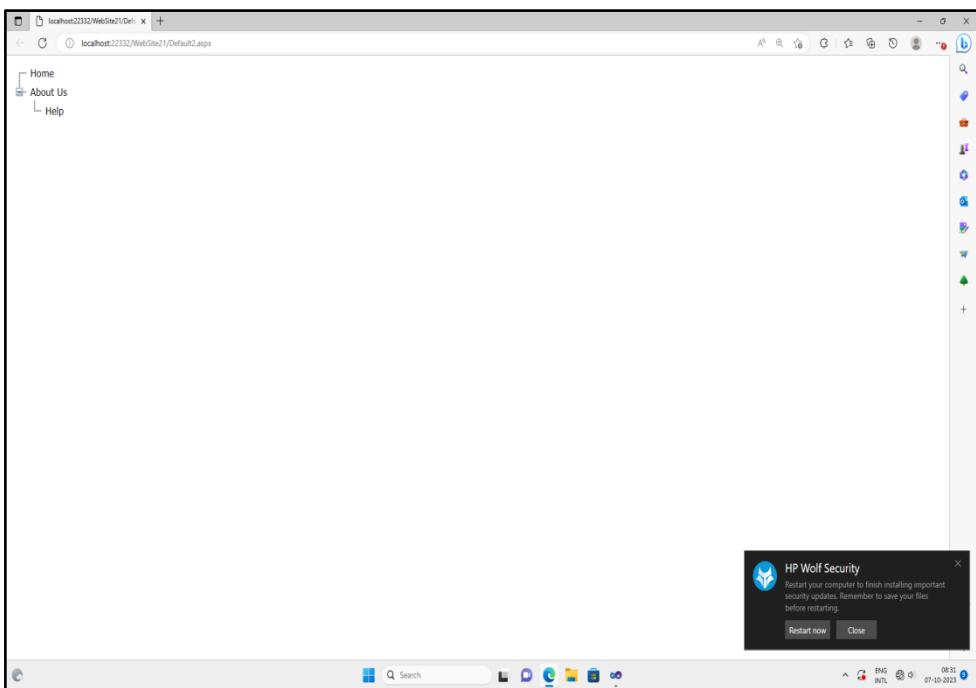
Demonstrate Website Navigation Controls Using TreeviewFile.

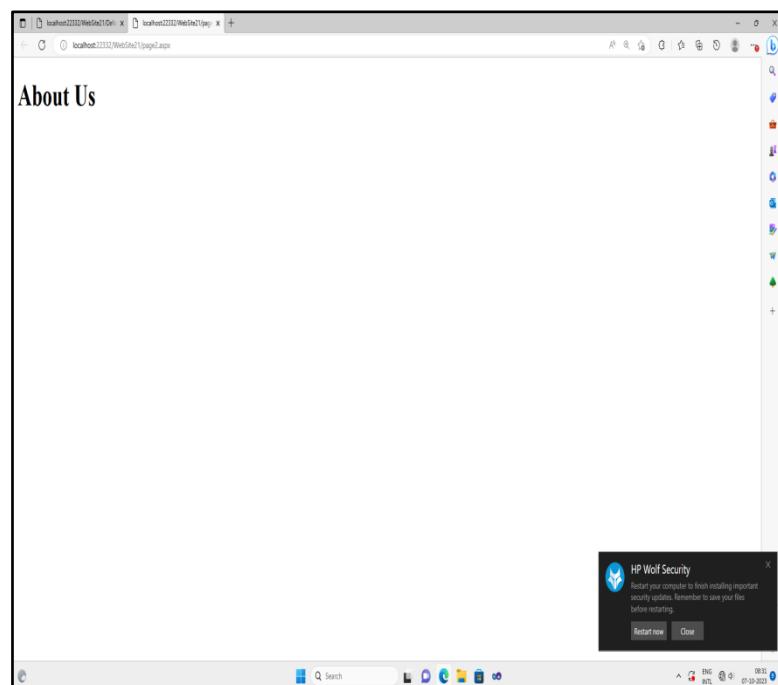
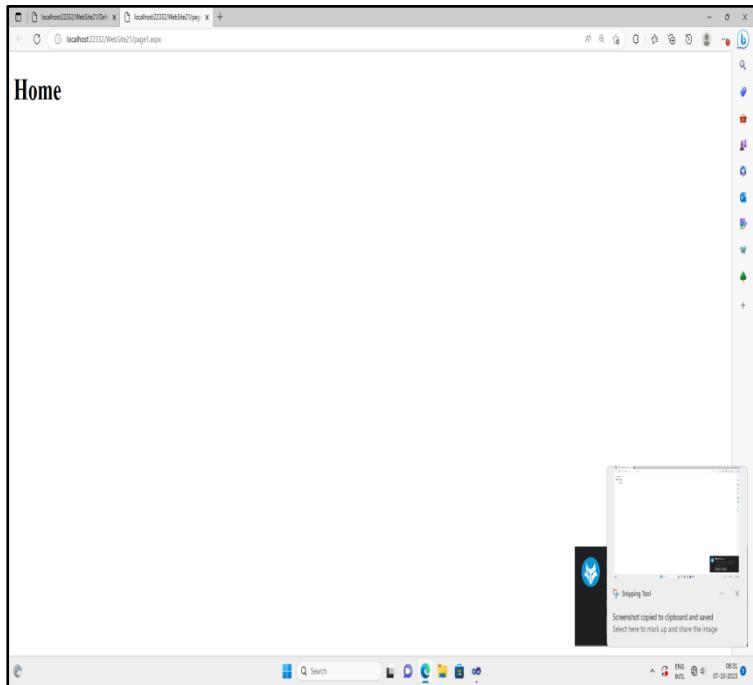






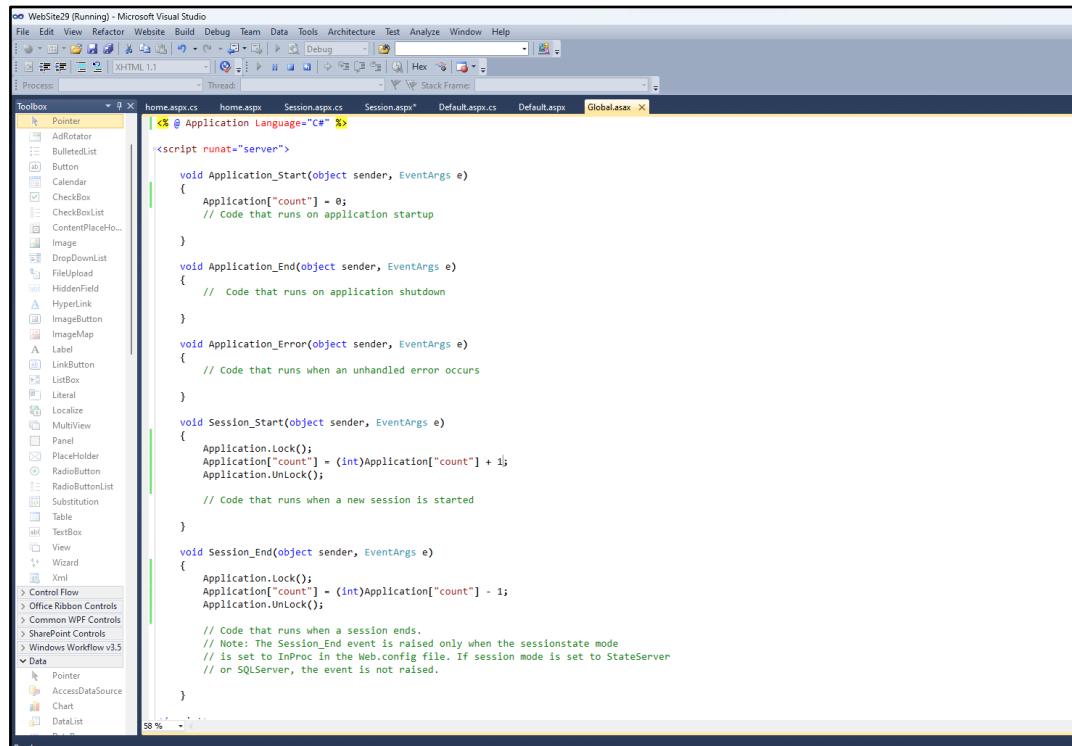
Output





Assignment No.11:

Demonstrate of Asp.Net objects (HTTP ApplicationState, HTTPSessionState), HTTP ApplicationState



The screenshot shows the Microsoft Visual Studio IDE with the Global.asax file open. The code defines event handlers for Application_Start, Application_End, Application_Error, Session_Start, and Session_End. It initializes a counter in Application state and increments it in Session state.

```
<script runat="server">
    void Application_Start(object sender, EventArgs e)
    {
        Application["count"] = 0;
        // Code that runs on application startup
    }

    void Application_End(object sender, EventArgs e)
    {
        // Code that runs on application shutdown
    }

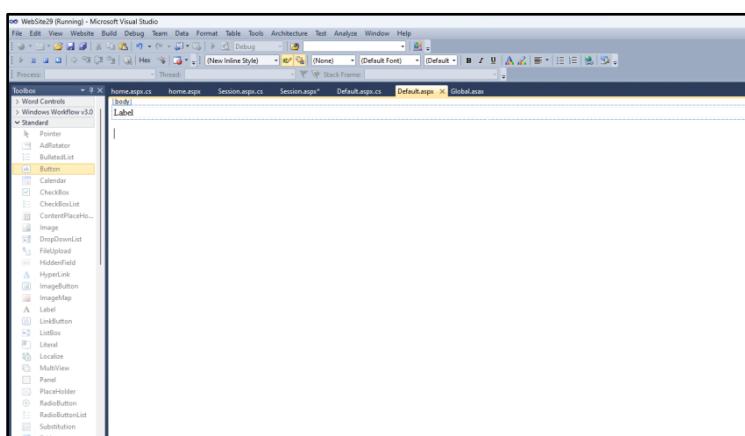
    void Application_Error(object sender, EventArgs e)
    {
        // Code that runs when an unhandled error occurs
    }

    void Session_Start(object sender, EventArgs e)
    {
        Application.Lock();
        Application["count"] = (int)Application["count"] + 1;
        Application.UnLock();

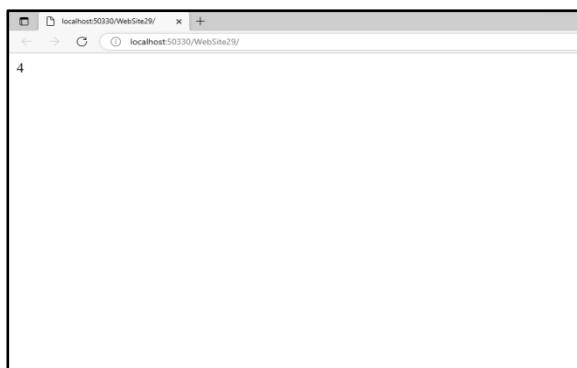
        // Code that runs when a new session is started
    }

    void Session_End(object sender, EventArgs e)
    {
        Application.Lock();
        Application["count"] = (int)Application["count"] - 1;
        Application.UnLock();

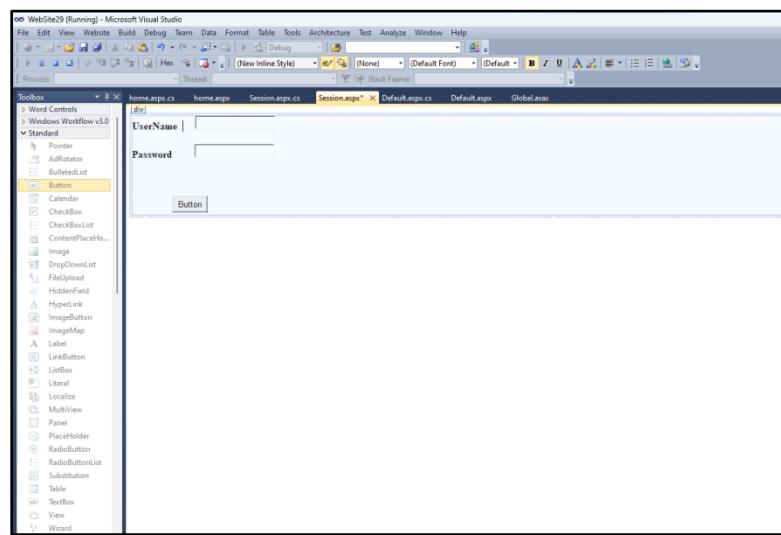
        // Code that runs when a session ends.
        // Note: The Session_End event is raised only when the sessionstate mode
        // is set to InProc in the Web.config file. If session mode is set to StateServer
        // or SQLServer, the event is not raised.
    }
}
```



Output:



SessionState

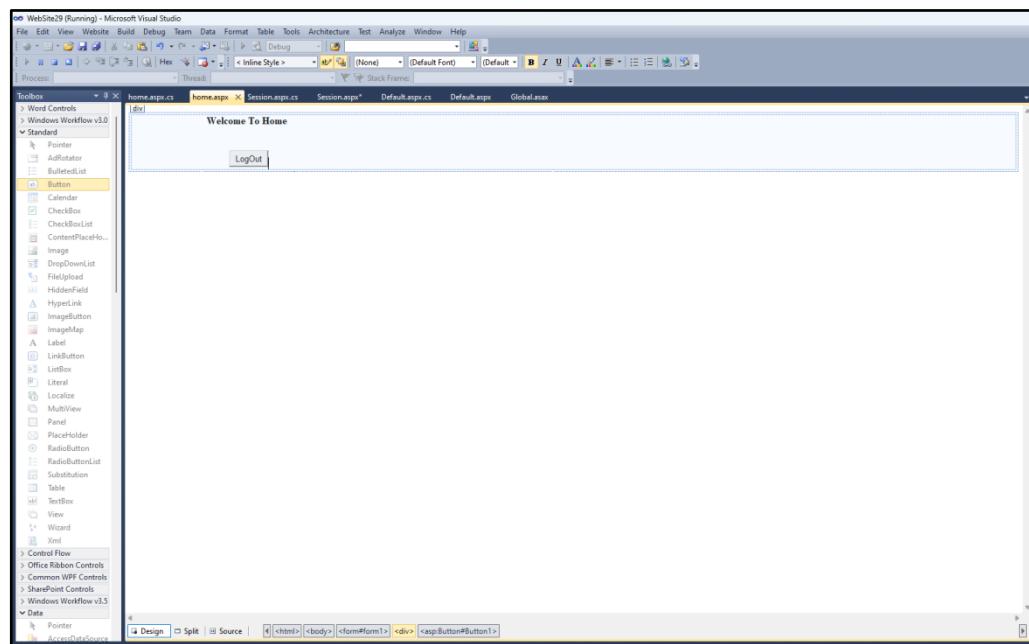


```
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

public partial class Session : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {

    }

    protected void Button1_Click(object sender, EventArgs e)
    {
        if (TextBox1.Text == "rcp" & TextBox2.Text == "imrd")
        {
            Session["id"] = TextBox1.Text;
            Response.Redirect("home.aspx");
        }
        else
        {
            Response.Write("does not match");
        }
    }
}
```



The screenshot shows the Microsoft Visual Studio interface with the title bar "Website29 (Running) - Microsoft Visual Studio". The menu bar includes File, Edit, View, Refactor, Website, Build, Debug, Team, Data, Tools, Architecture, Test, Analyze, Window, Help. The toolbar has icons for Save, Undo, Redo, Cut, Copy, Paste, Find, etc. The status bar at the bottom right says "Ln 21".

The code editor displays the file "home.aspx.cs" with the following C# code:

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

public partial class home : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        if (Session["id"] == null)
        {
            Response.Redirect("Default.aspx");
        }
    }

    protected void Button1_Click(object sender, EventArgs e)
    {
        Session.RemoveAll();
    }
}
```

The screenshot shows a web browser window with the URL "localhost:50330/WebSite29/Session.aspx". The page contains two text input fields: "UserName" with value "rcp" and "Password" with value "imrd". Below the password field is a "Button" control.

The screenshot shows a web browser window with the URL "localhost:50330/WebSite29/home.aspx". The page displays the message "Welcome To Home" and a "LogOut" button.

Assignment No.12:

Write an ASP.NET Program to Demonstrate Use of ASP.NET TextBox, CheckBox, RadioButton and Button Control in Simple Web Form.

The screenshot shows a Microsoft Visual Studio interface with a web form titled "Registration Form". The form contains fields for Student Name, Student Address, and Student email, each with a corresponding text input box and a label. There is a section for Student Gender with two radio buttons: Male and Female, and a label [Label9]. Below this is a section for Sports with three checkboxes: cricket, hockey, and football, and a label [Label10]. A "Save" button is at the bottom right.

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

public partial class regist1 : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {

    }

    protected void Button1_Click(object sender, EventArgs e)
    {
        Label6.Text = TextBox1.Text;
        Label7.Text = TextBox2.Text;
        Label8.Text = TextBox3.Text;
    }

    protected void CheckBoxList1_SelectedIndexChanged(object sender, EventArgs e)
    {
        Label10.Text = CheckBoxList1.SelectedItem.ToString();
    }

    protected void RadioButtonList1_SelectedIndexChanged(object sender, EventArgs e)
    {
        Label9.Text = RadioButtonList1.SelectedItem.ToString();
    }
}
```

Output:

localhost:21440/WebSite20/reg1.aspx

Registration Form

Student Name vaishnavi

Student Address shirpur

Student email vaishnavi@gmail.com

Student Gender

Male

Female

Female

Sports

cricket

hockey

football

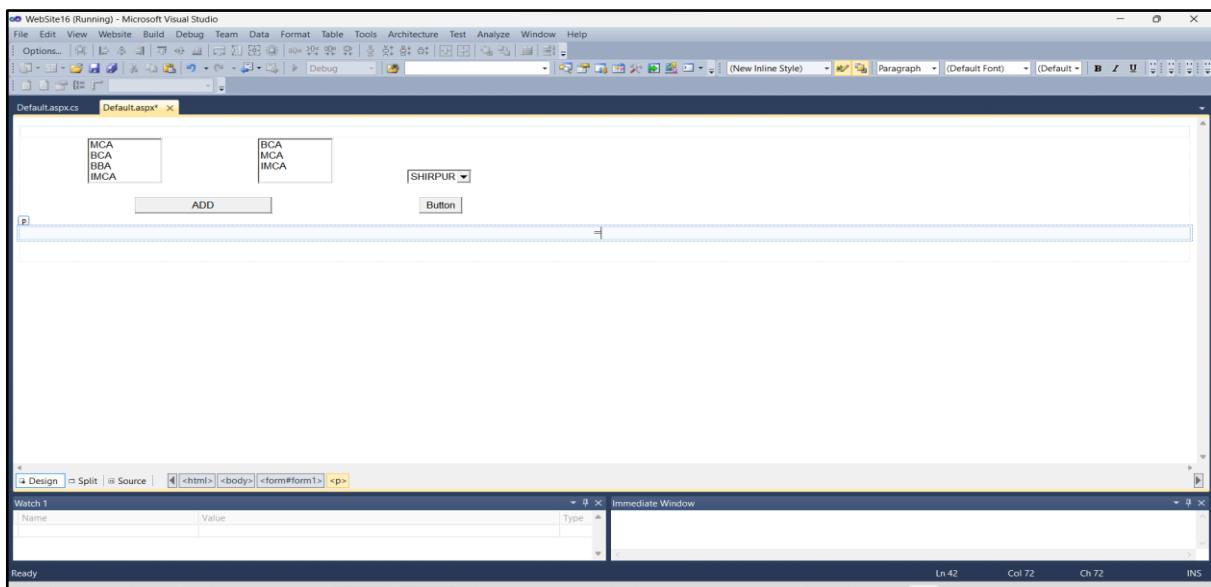
cricket

35°C Mostly sunny

Search ENG IN 06-10-2023 15:51

Assignment No.13

Write an ASP.Net Program to Demonstrate Use of ASP.Net Dropdownlist, ListBox and Button Control in Simple Web Form .



Default.aspx

Default.aspx.cs

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

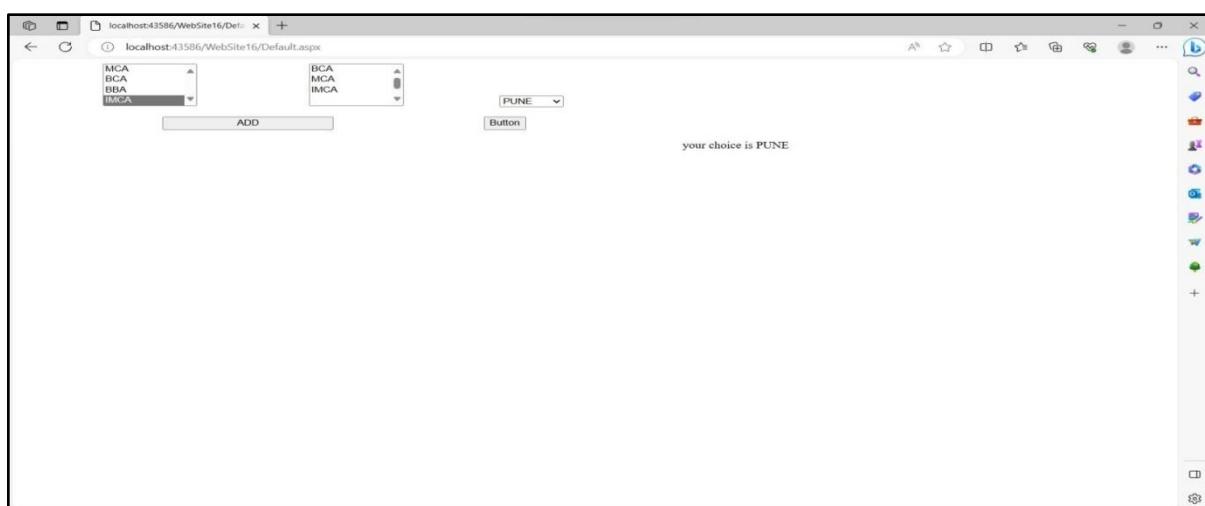
public partial class _Default :System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
    }

    protected void Button1_Click(object sender, EventArgs e)
    {
        if (ListBox1.SelectedItem != null)
        {
            string SelectedItem =
                ListBox1.SelectedItem.ToString(); ListBox2.Items.Add(SelectedItem);
        }
    }
}
```

```
        }

    }
    protected void Button3_Click(object sender, EventArgs)
    {
        if (DropDownList1.SelectedValue=="")
        {
            Label1.Text="Please Select a City ";
        }
        else
        {
            Label1.Text="your choice is "+ DropDownList1.SelectedValue;
        }
    }
}
```

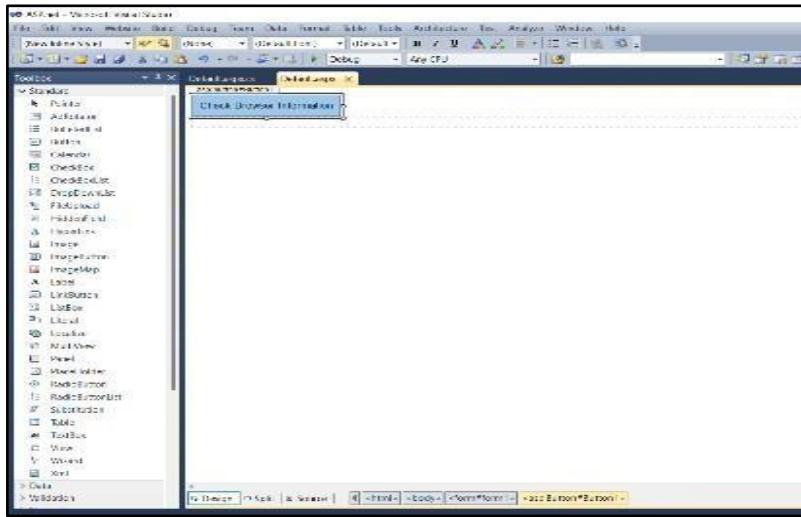
Output:



Assignment No:-14

Demonstrate How to Detect Browser Capabilities in ASP.NET

Default.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 _Default :System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
    }
    protected void Button1_Click(object sender, EventArgs e)
    {
        HttpBrowserCapabilities K;
        K = Request.Browser;
        Response.Write("<h1>Browser Information And Capabilities</h1>");

        Response.Write("Browser Name is :-" + K.Browser +
        "<br>"); Response.Write("Browser Version is :-" + K.Version +
        "<br>"); Response.Write("Browser Support to tables:-" + K.Tables +
        "<br>"); Response.Write("BrowserSupporttoJavaScript:-" + K.JavaScript+
        "<br>");

        Response.Write("Browser Support to VBScript :-" + K.VBScript +
        "<br>"); Response.Write("Browser Support to Frames :-" + K.Frames +
        "<br>"); Response.Write("BrowserSupport to BackGround Sound :-" +
        K.BackgroundSounds + "<br>");

        Response.Write("Browser Support to Cookies :-" + K.Cookies +
        "<br>"); Response.Write("Browser ID is :-" + K.Id +
        "<br>"); Response.Write("BrowserPlatform is :-" + K.Platform+
        "<br>");

    }
}
```

OutPut:

Broweser Information And Capabilities

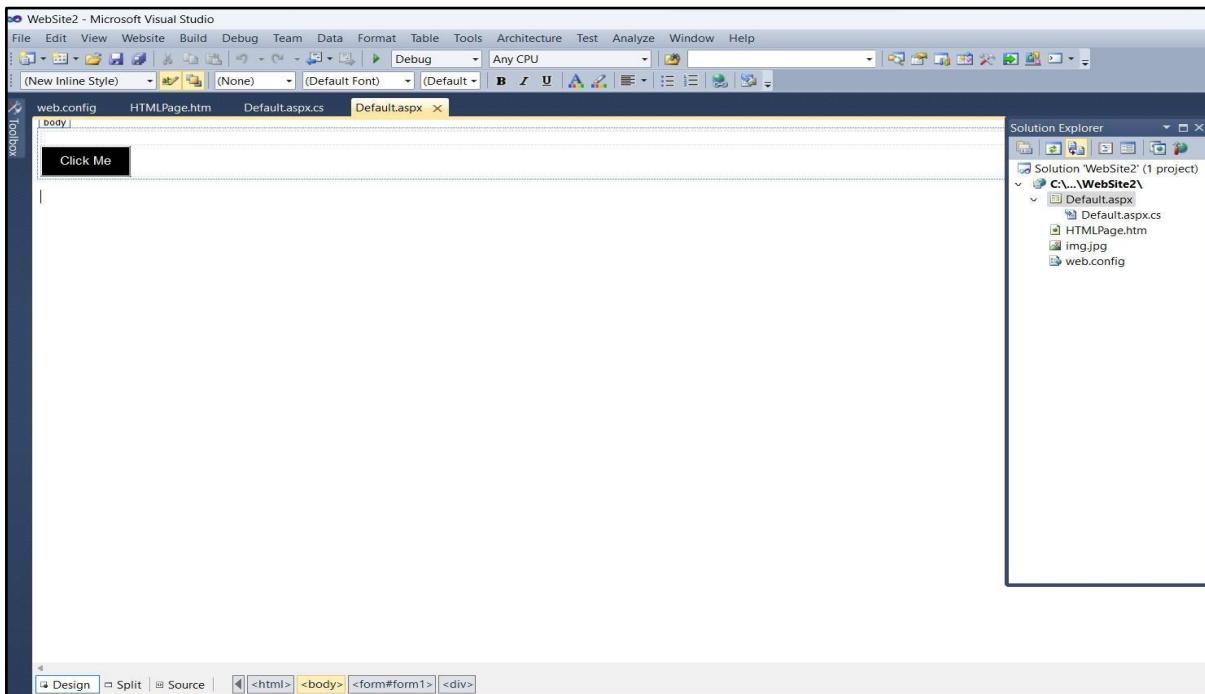
Browser Name is :-Chrome
Browser Version is :-115.0
Browser Support to tables:-True
Browser Support to JavaScript :-True
Browser Support to VBScript :-False
Browser Support to Frames :-True
Browser Support to BackGround Sound :-False
Browser Support to Cookies :-True
Browser ID is :-chrome
Browser PlatForm is :-WinNT

[Check Browser Information](#)

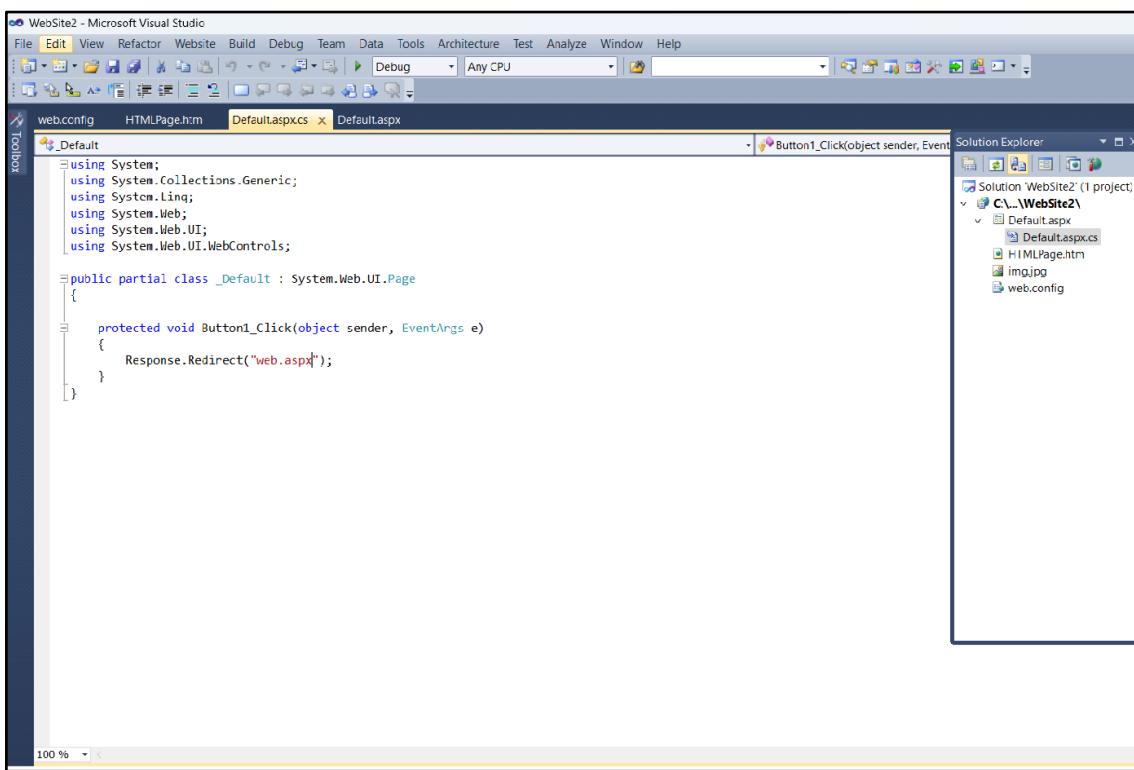
AssignmentNo:-15

Demonstrate how to handle Application Level Errors in ASP.NET.

Default.aspx File:-



1) Default.aspx.css File:-



2) HTMLPage.htmFile:-

The screenshot shows the Microsoft Visual Studio interface with the title bar "WebSite2 - Microsoft Visual Studio". The menu bar includes File, Edit, View, Website, Build, Debug, Team, Data, Tools, Architecture, Test, Analyze, Window, Help. The toolbar has standard icons for file operations. The status bar at the bottom shows "100 %". The main editor window displays the HTML code for "HTMLPage.htm". The code includes a DOCTYPE declaration, a title, and a body containing an image tag pointing to "img.jpg". The Solution Explorer on the right shows a project named "WebSite2" with files "Default.aspx.cs", "Default.aspx", "HTMLPage.htm", "img.jpg", and "web.config".

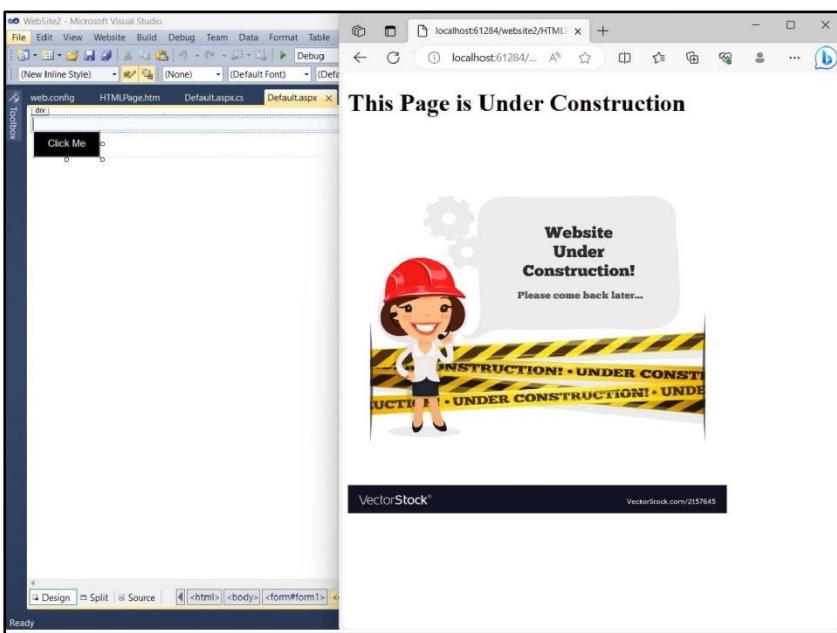
```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title></title>
</head>
<body>
<h1>This Page is Under Construction</h1>
<image src="img.jpg" height="500" width="500"></image>
</body>
</html>
```

3) Web.configFile:-

The screenshot shows the Microsoft Visual Studio interface with the title bar "WebSite2 - Microsoft Visual Studio". The menu bar includes File, Edit, View, Website, Build, Debug, Team, XML, Data, Tools, Architecture, Test, Analyze, Window, Help. The toolbar has standard icons for file operations. The main editor window displays the XML configuration file "web.config". It contains sections for system.web, including compilation and customErrors settings. The Solution Explorer on the right shows a project named "WebSite2" with files "Default.aspx.cs", "Default.aspx", "HTMLPage.htm", "img.jpg", and "web.config".

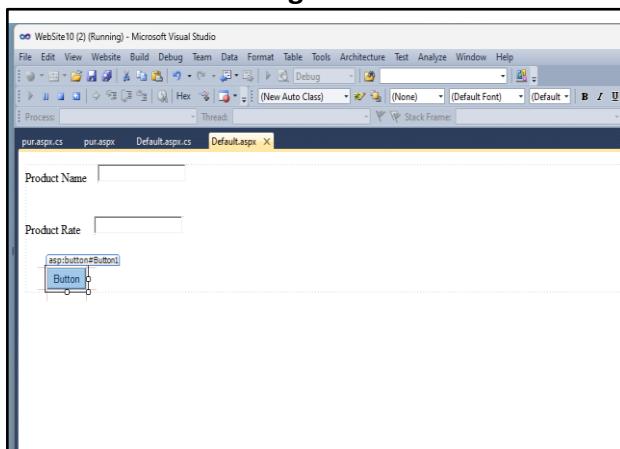
```
<?xml version="1.0"?>
<!--
  For more information on how to configure your ASP.NET application, please visit
  http://go.microsoft.com/fwlink/?LinkId=169433
-->
<configuration>
  <system.web>
    <compilation debug="true" targetFramework="4.0"/>
    <customErrors defaultRedirect="HTMLPage.htm" mode="On"></customErrors>
  </system.web>
</configuration>
```

Output:



Assignment No.16:

Write an ASP.Net Program to Demonstrate use of Server.Transfer() and Request.QueryString.

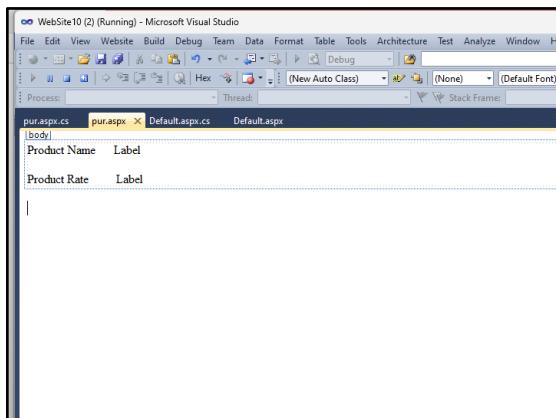


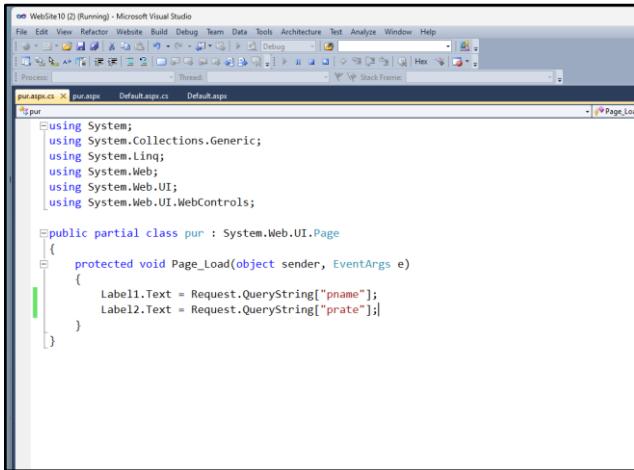
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

public partial class _Default : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
    }

    protected void Button1_Click(object sender, EventArgs e)
    {
        Server.Transfer("pur.aspx?pname=" + TextBox1.Text + "&prate=" + TextBox2.Text + "");
    }
}
```

The screenshot shows the code-behind file "Default.aspx.cs" in Microsoft Visual Studio. The code uses the `Server.Transfer` method to redirect to "pur.aspx" with query string parameters "pname" and "prate" set to the values from the text boxes. The code includes standard using statements and defines a partial class `_Default` that inherits from `System.Web.UI.Page`. It contains a `Page_Load` event handler and a `Button1_Click` event handler that triggers the transfer.

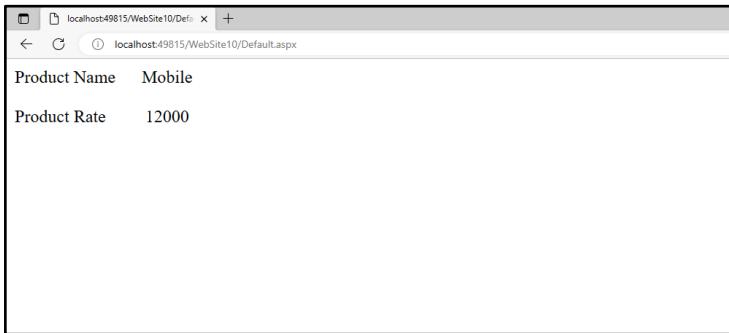
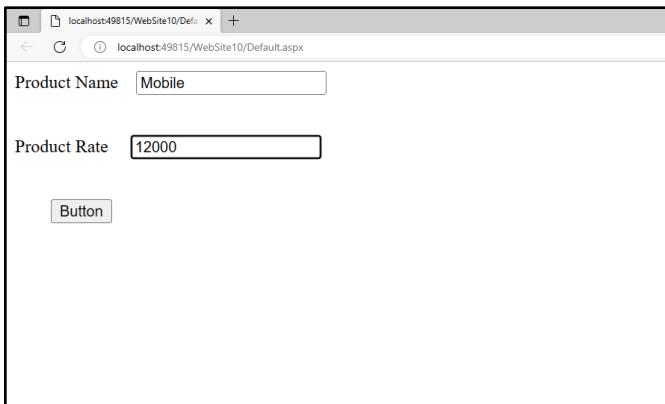




```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

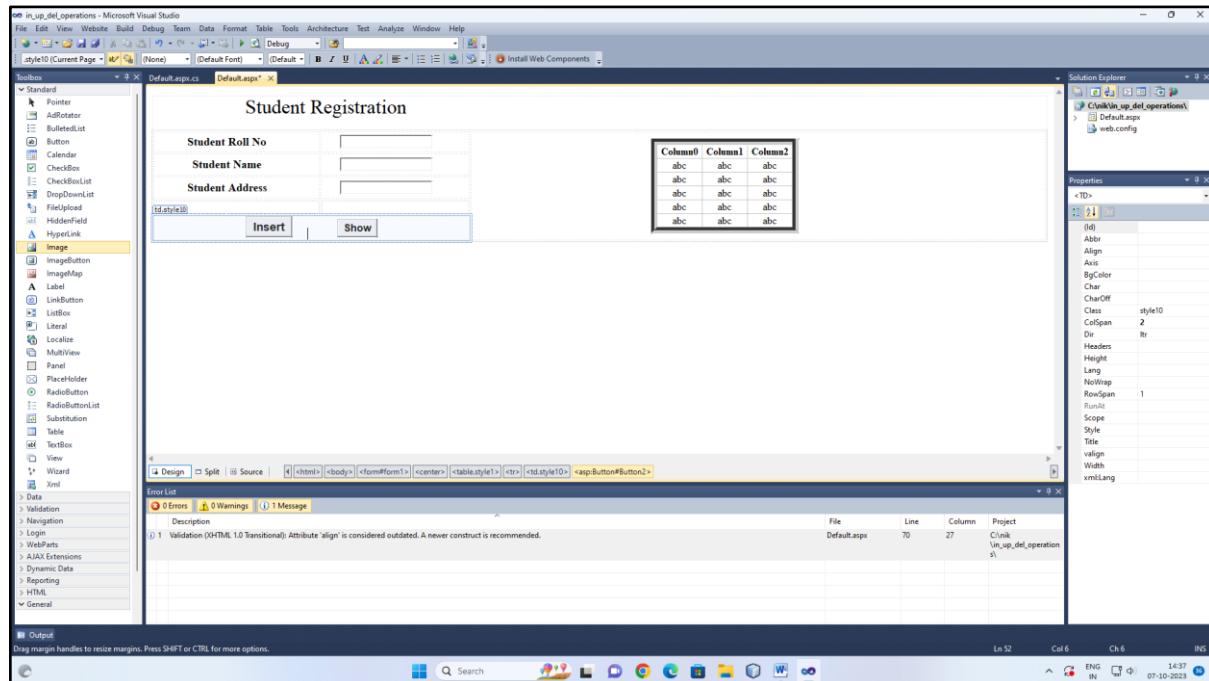
public partial class pur : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        Label1.Text = Request.QueryString["pname"];
        Label2.Text = Request.QueryString["prate"];
    }
}
```

Output:



Assignment No.17:

Write an ASP.Net Program to Demonstrate Insert , Display Operations with Microsoft Access Database.



Button Coding:

Insert button:

```
protected void Button1_Click(object sender, EventArgs e)
{
    OleDbConnection con = new
    OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data
    Source=C:\\nik\\database\\in_up_del_operation.accdb");

    OleDbCommand cmd = new OleDbCommand("insert into stud(rollno,sname,sadd)
    values (" + TextBox1.Text + ", '" + TextBox2.Text + "', '" + TextBox3.Text + "')", con);

    con.Open();
    cmd.ExecuteNonQuery();
    con.Close();
    Response.Write("insert successfully");
}
```

Show button:

```
protected void Button4_Click(object sender, EventArgs e)
{
    OleDbConnection con =
    new OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data
    Source=C:\\nik\\database\\in_up_del_operation.accdb");

    OleDbCommand cmd = new OleDbCommand("select * from stud", con);

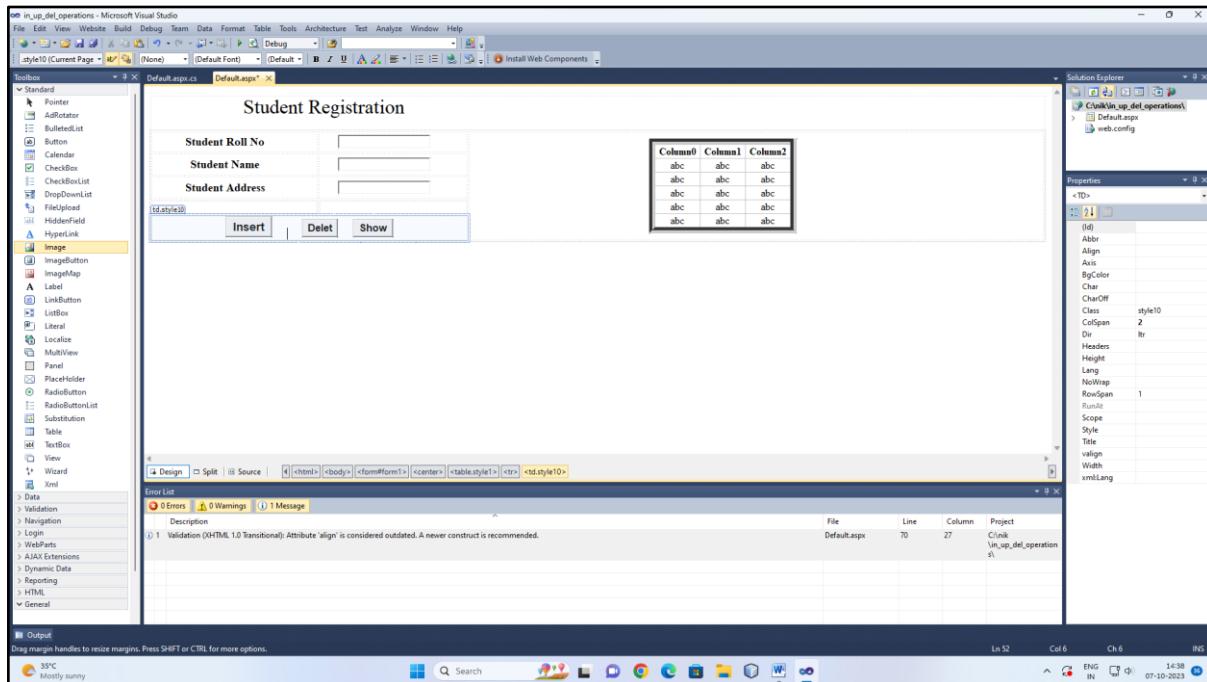
    con.Open();
    OleDbDataReader dr = cmd.ExecuteReader();

    GridView1.DataSource = dr;
    GridView1.DataBind();
```

```
con.Close();  
}
```

Assignment No.18:

Write an ASP.Net Program to Demonstrate Insert , Delete and Display Operations with Microsoft Access Database.



Button Coding:

Insert button:

```
protected void Button1_Click(object sender, EventArgs e)
{
    OleDbConnection con = new
    OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data
    Source=C:\\nik\\database\\in_up_del_operation.accdb");

    OleDbCommand cmd = new OleDbCommand("insert into stud(rollno,sname,sadd)
    values (" + TextBox1.Text + ", '" + TextBox2.Text + "', '" + TextBox3.Text + "')", con);

    con.Open();
    cmd.ExecuteNonQuery();
    con.Close();
    Response.Write("insert successfully");
}
```

delete button:

```
protected void Button3_Click(object sender, EventArgs e)
{
    OleDbConnection con =
    new OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data
    Source=C:\\nik\\database\\in_up_del_operation.accdb");

    OleDbCommand cmd = new OleDbCommand("delete from stud where
    rollno=" + TextBox1.Text + "", con);

    con.Open();
    cmd.ExecuteNonQuery();
```

```
con.Close();
Response.Write("delete successfully");
}
```

Show button:

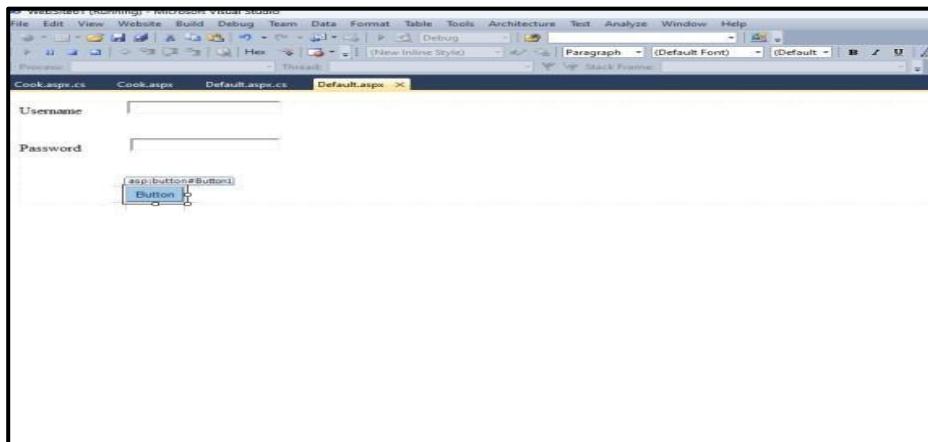
```
protected void Button4_Click(object sender, EventArgs e)
{
    OleDbConnection con =
new OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data
Source=C:\\nik\\database\\in_up_del_operation.accdb");

OleDbCommand cmd = new OleDbCommand("select * from stud", con);

con.Open();
OleDbDataReader dr = cmd.ExecuteReader();
GridView1.DataSource = dr;
GridView1.DataBind();
con.Close();
}
```

Assignment No. 19:

Demonstrate how we can create and Retrieve cookies in ASP.NET.



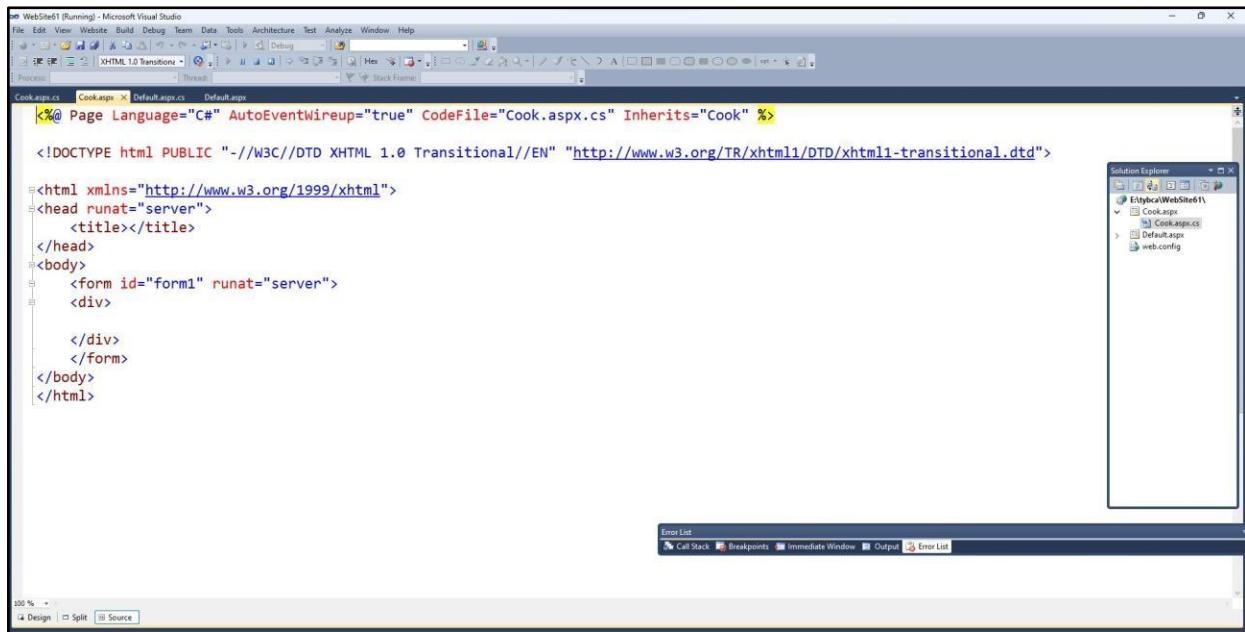
Default.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;usingSystem.Web;
usingSystem.Web.UI;
usingSystem.Web.UI.WebControls;

publicpartialclass _Default: System.Web.UI.Page
{
    protectedvoidPage_Load(object sender,EventArgs e)
    {
    }

    protectedvoidButton1_Click(object sender,EventArgs e)
    {
        Response.Cookies["rcp"].Value=TextBox1.Text;
        Response.Cookies["rcp"].Expires.AddDays(1);
        Response.Redirect("Cook.aspx");
    }
}
```

Cook.aspx

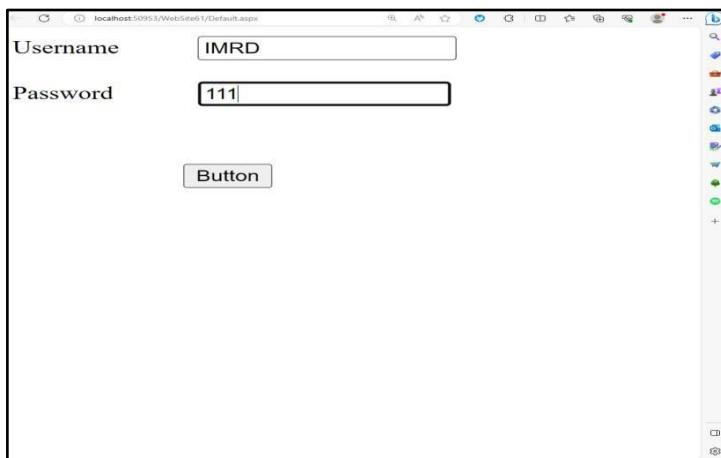


Cook.aspx.cs

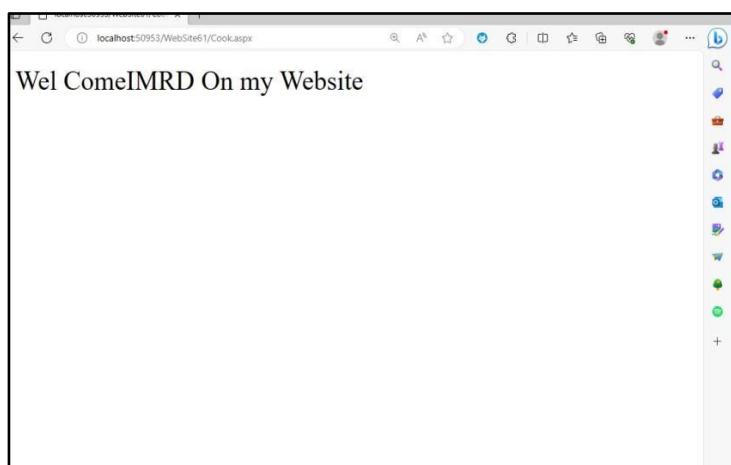
```
using System;
using System.Collections.Generic;
using System.Linq;using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

public partial class Cook : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        Response.Write("WelCome" + "" + Request.Cookies["rcp"].Value.ToString() + "OnmyWebsite");
    }
}
```

Output:



Output:

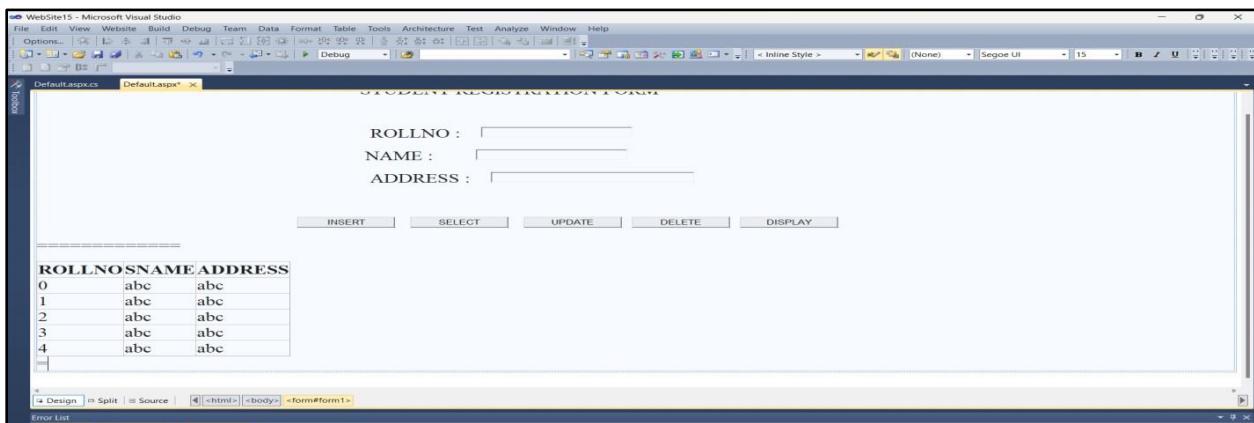


AssignmentNo20:-WriteASP.NETProgramtoDemonstratedDifferentDatabaseOperation(Insert, Select, Update, Display And Delete).

Step1:-OpenVisualStudio2010.

Step 2 :- Create new Website.

Step3:-Gotosolutionexplorer>rightclick>Addnewitem>webform>Ok.

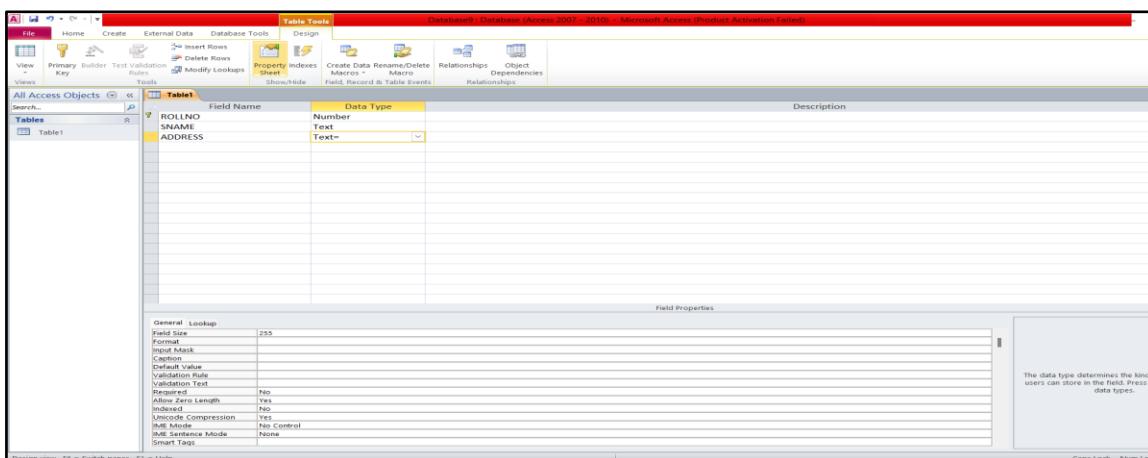


Step4:CreateMicrosoftAccessDatabase

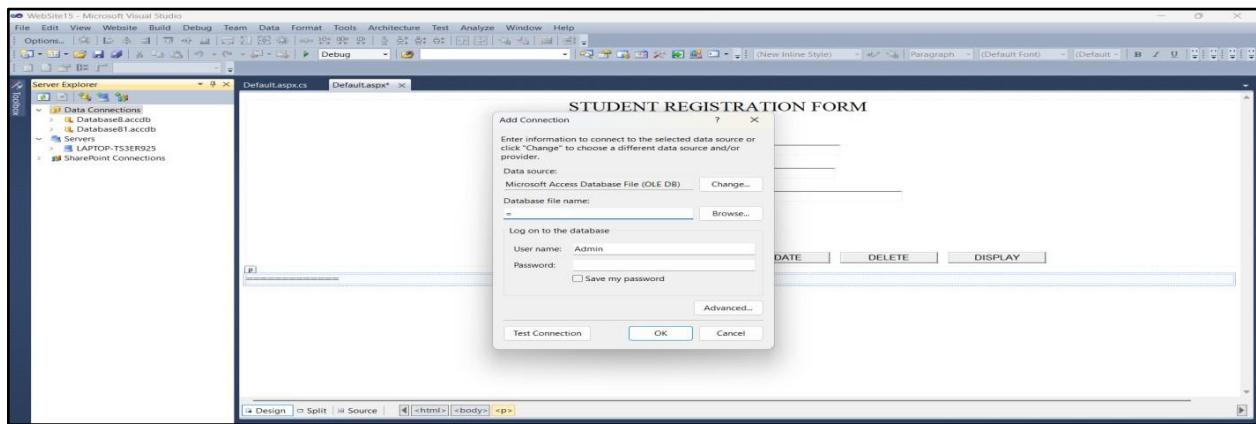
Step 5 : Select blank Database option

Step 6 :Give File Name

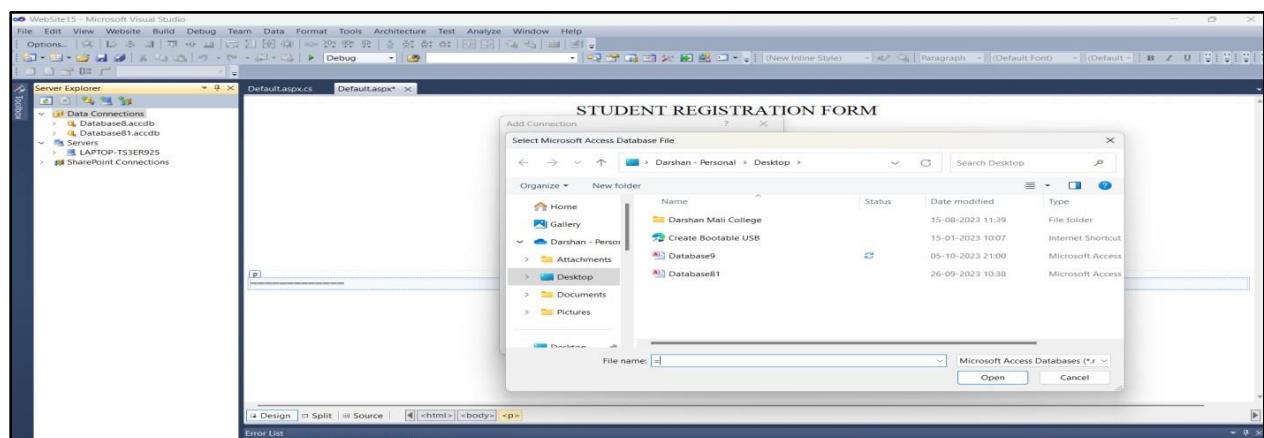
Step7:Create Table



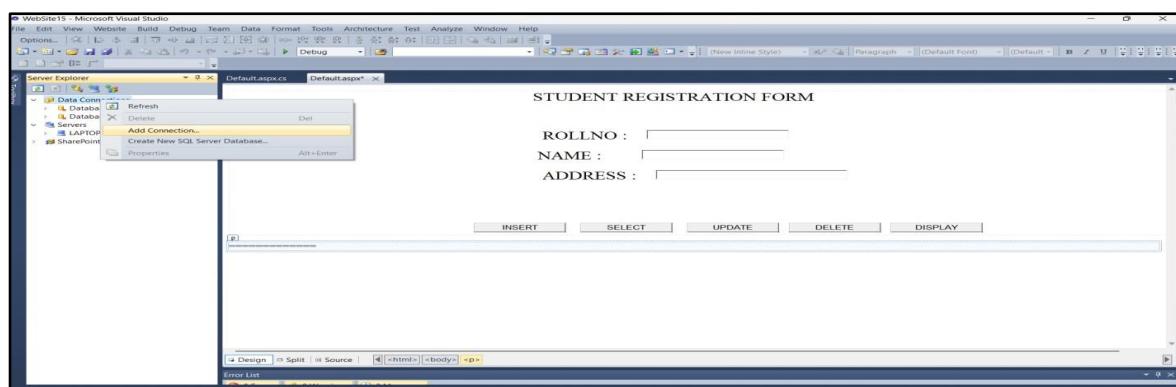
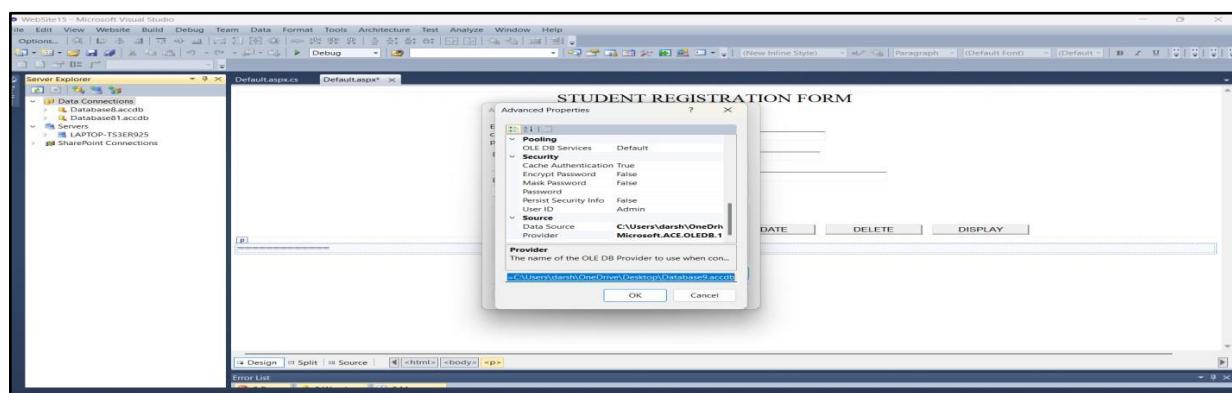
Step8:Connect DatabaseFile with WebForm



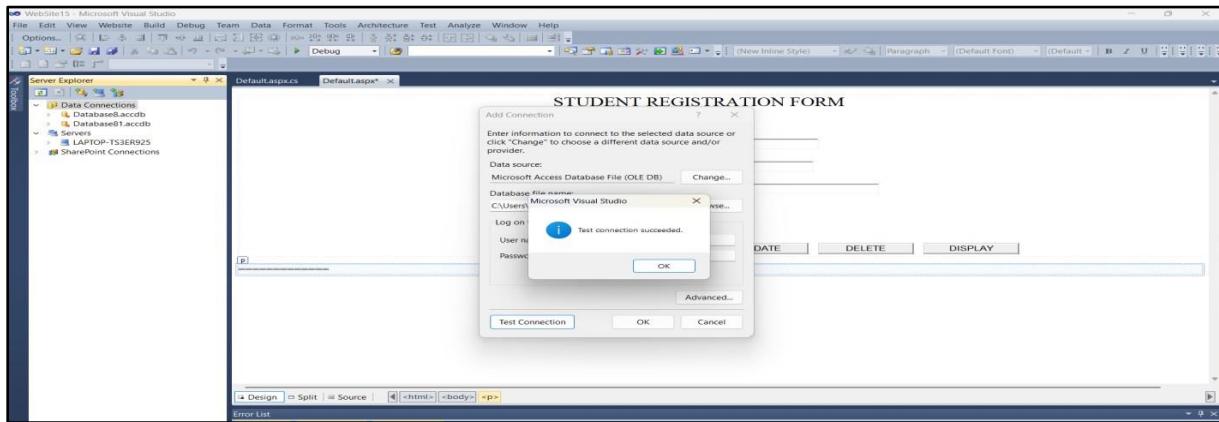
Click on Browse And Select Database



Click on Advanced and Copy Link



Check Connection



Default.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.Data.OleDb;
public partial class _Default : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        }

    protected void Button1_Click(object sender, EventArgs e)
    {
        OleDbConnection con = new
        OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data
        Source=C:\\\\Users\\\\darsh\\\\OneDrive\\\\Desktop\\\\Database9.accdb"
        );

        OleDbCommand cmd = new OleDbCommand("insert into
        Table1(ROLLNO,SNAME,ADDRESS)values('" + TextBox1.Text + "'"
        + TextBox2.Text + "','" + TextBox3.Text + "')", con);

        con.Open(); cmd.ExecuteNonQuery();
        con.Close(); Response.Write("INSERT");

    }

    protected void Button2_Click(object sender, EventArgs e)
    {
        OleDbConnection con = new
        OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data
        Source=C:\\\\Users\\\\darsh\\\\OneDrive\\\\Desktop\\\\Database9.accdb"
        );

        OleDbCommand cmd = new OleDbCommand("select*from Table1", con);
        con.Open();
    }
}

```

```

        OleDbDataReader dr=cmd.ExecuteReader(); GridView1.DataSource = dr;
        GridView1.DataBind();
    Response.Write("SELECT");
    }

protected void Button3_Click(object sender, EventArgs e)
{
    OleDbConnection con = new
    OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data
    Source=C:\\\\Users\\\\darsh\\\\OneDrive\\\\Desktop\\\\Database9.accdb"
    );
    OleDbCommand cmd=new OleDbCommand("updateTable1 set
    ADDRESS='"+TextBox3.Text+"' where ROLLNO='"+TextBox1.Text+"'", con);
    con.Open(); cmd.ExecuteNonQuery();
    con.Close(); Response.Write("UPDATE");
}

protected void Button4_Click(object sender, EventArgs e)
{
    OleDbConnection con = new
    OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data
    Source=C:\\\\Users\\\\darsh\\\\OneDrive\\\\Desktop\\\\Database9.accdb");

    DataTable dt=new DataTable();
    con.Open();
    OleDbDataAdapter da=new OleDbDataAdapter("select*from Table1",
    con);
    da.Fill(dt);
    GridView1.DataSource=dt;
    GridView1.DataBind();

    con.Close();
}

protected void Button5_Click(object sender, EventArgs e)
{
    OleDbConnection con = new
    OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data
    Source=C:\\\\Users\\\\darsh\\\\OneDrive\\\\Desktop\\\\Database9.accdb");
    OleDbCommand cmd=new OleDbCommand("delectfromTable1where ROLLNO="
    + TextBox1.Text + "", con);
    con.Open();
    cmd.ExecuteNonQuery();
    con.Close();
    Response.Write("DELETE");
}

```

AssignmentNo21:-WriteASP.NETProgram usingAdRotatorControl.

Step1:-OpenVisualStudio2010.

Step 2 :- Create new Website.

Step3:-Gotosolutionexplorer>rightclick>Add newitem>xmlfile.

Step 4:- Write a below code in xml file.



The screenshot shows the XMLFile.xml file open in the Visual Studio editor. The tab bar at the top has 'XMLFile.xml' (highlighted in yellow), 'Default.aspx', and 'MasterPage.master'. The XML code in the editor is:

```
<?xml version="1.0" encoding="utf-8" ?>
<Advertisements>
  <Ad>
    <ImageUrl>banner.jpg</ImageUrl>
    <NavigateUrl>Default.aspx</NavigateUrl>
    <AlternateText>Image1</AlternateText>
    <Impressions>4</Impressions>
    <Keyword>Ad1</Keyword>
  </Ad>
  <Ad>
    <ImageUrl>bg 2.jpg</ImageUrl>
    <NavigateUrl>Default.aspx</NavigateUrl>
    <AlternateText>Image1</AlternateText>
    <Impressions>4</Impressions>
    <Keyword>Ad1</Keyword>
  </Ad>
</Advertisements>
~
```

Step5:-GotoToolBox and place AdRotator Control on form.



Step6:-Go to properties and select Advertisement File property and pass xml file Url.

Step 7:- Run program.

