

# Shree M.M.Ghodasara Mahila College

## Junagadh

---

### Computer Science Department

### B.C.A.

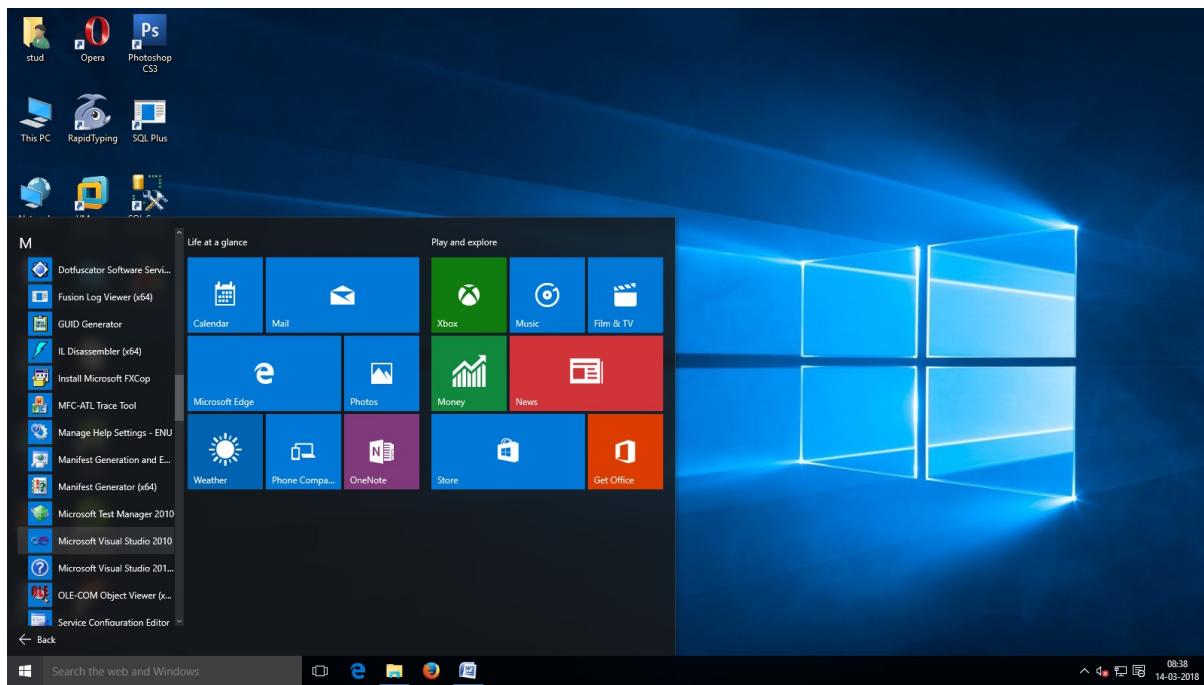
### Lab Manual

ASP.NET

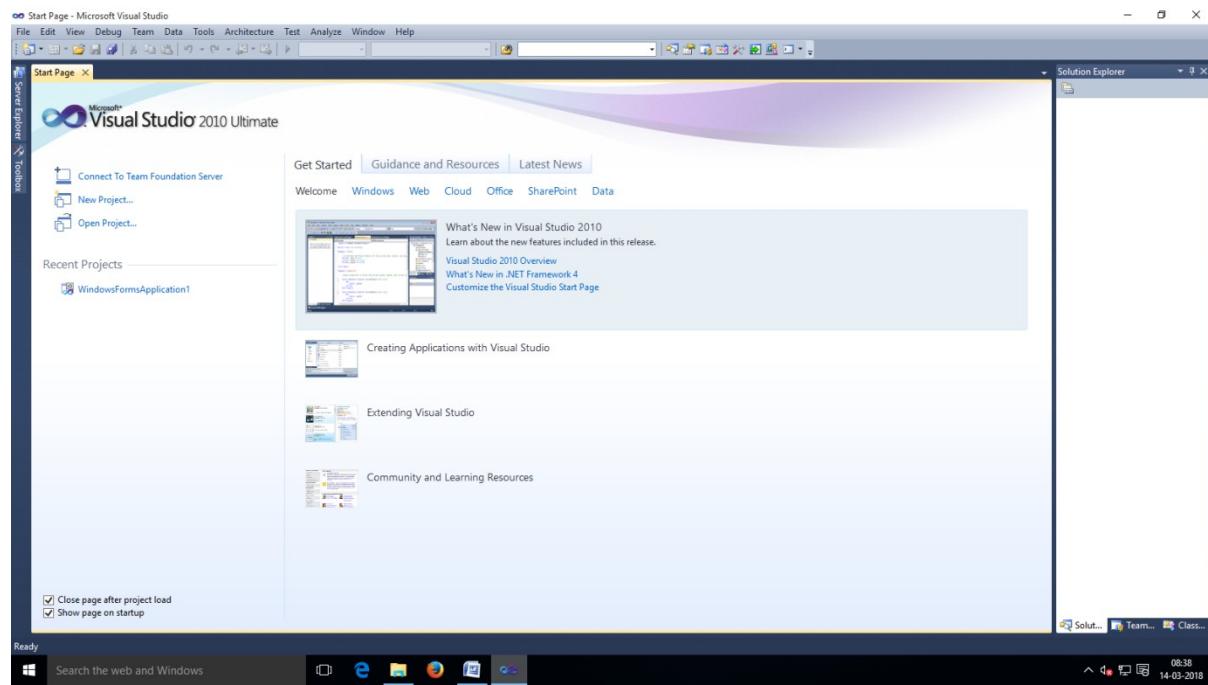


**Shree M.M.Ghodasara MahilaCollege**  
Opp. Motibaug, Junagadh - 362 001  
Ph. (0285) - 2670523, 2671523

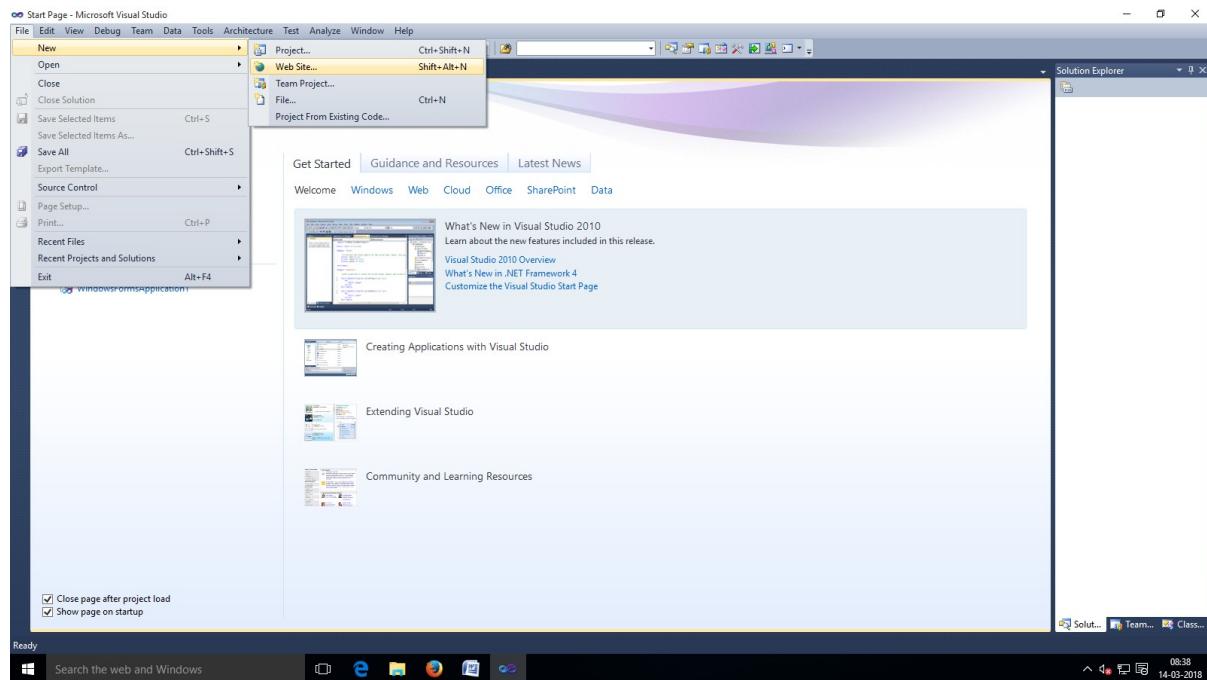
## 1) How to open ASP.NET?



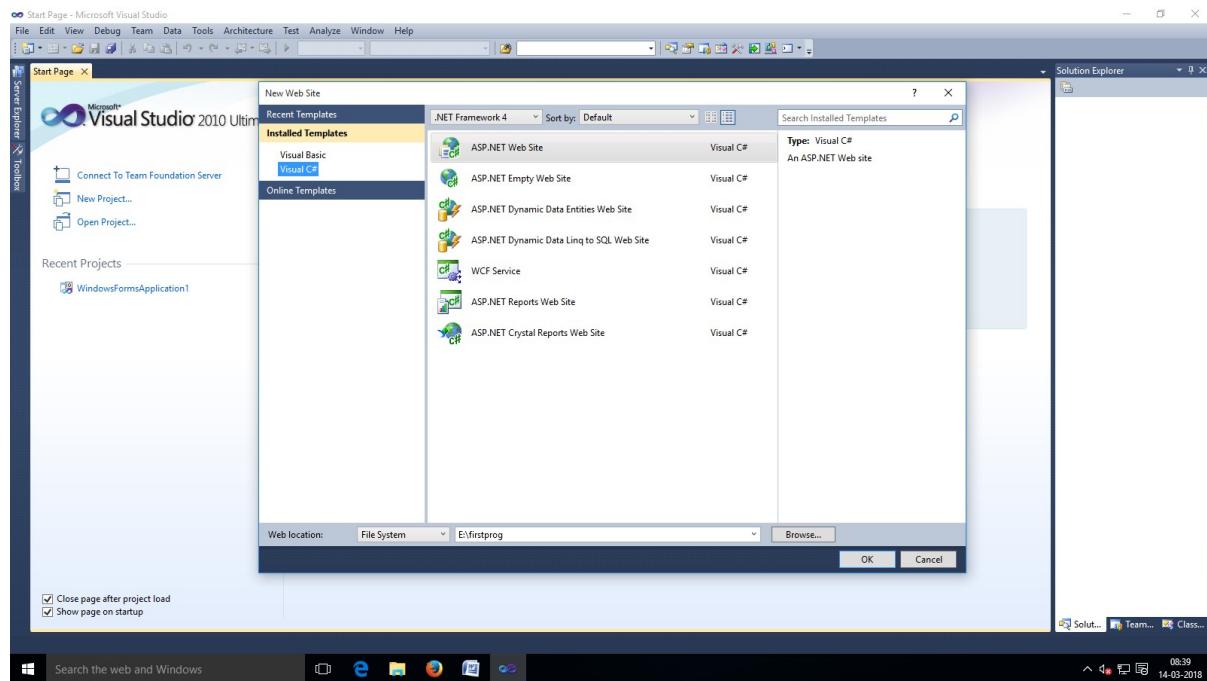
After Going to Microsoft Visual Studio 2010 you can see the frame like this



## To Create New website

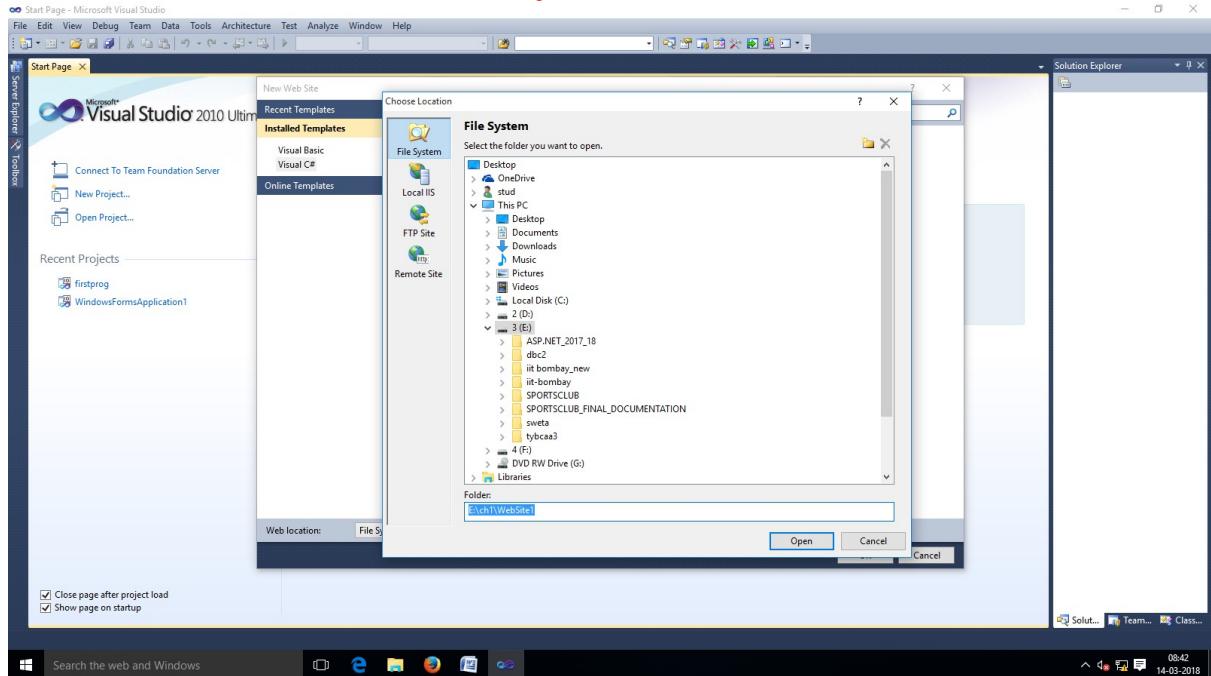


After Clicking on Website you can see this. Just click on ASP.NET website & select your path from Browse button.



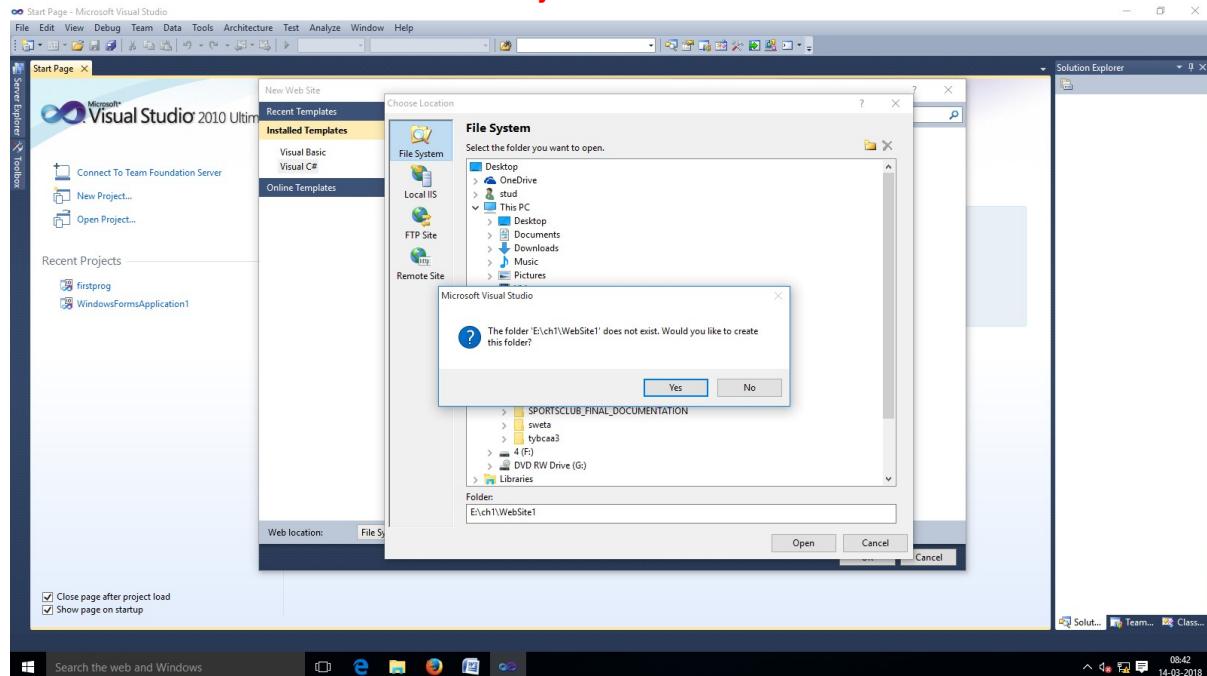
Click on Browse button & select where you want to create the website. If there is no folder is created there after your folder will be created by Dot net as you written in your path.

## Subject :- ASP.NET

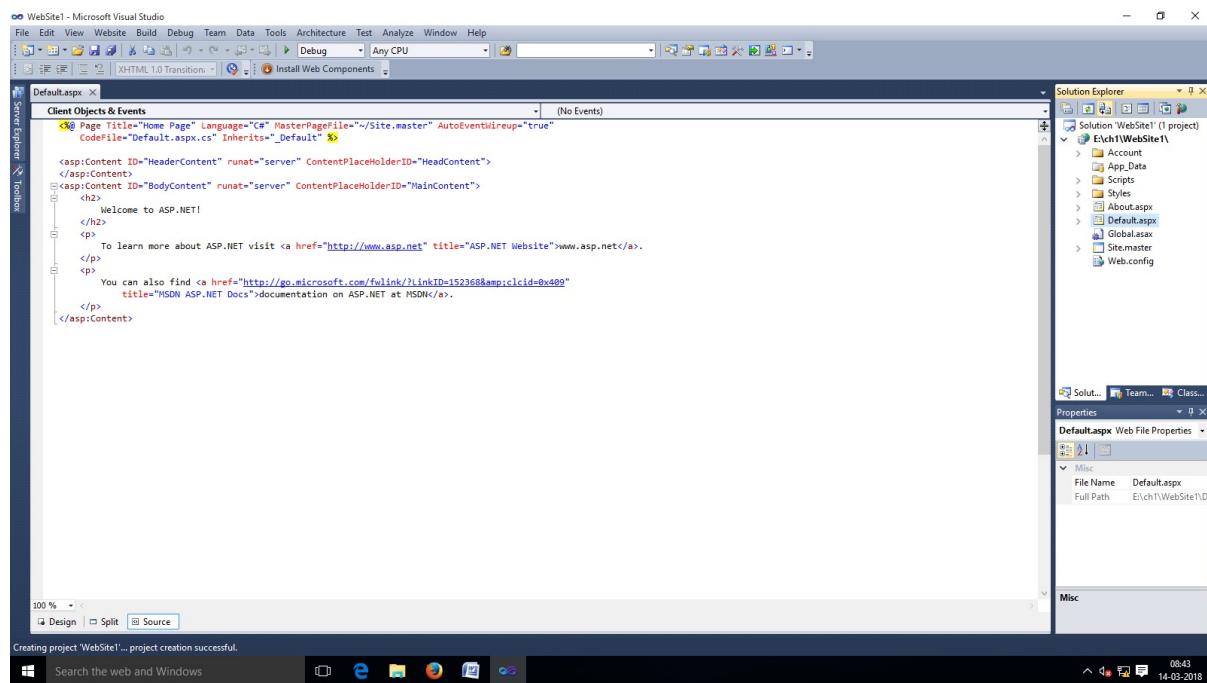


We have created website in E: drive there is folder named ch1. & then write name firstprog even if there is not any folder of this name. But if you click on Open button the system will ask you about your surety to create a folder of that name or not?

## Subject :- ASP.NET



After Clicking Yes button you can see the frame like this.



Here you can see at source side. & the code will be

```
<%@Page Title="Home" Language="C#" MasterPageFile="~/Site.master" AutoEventWireup="true" CodeFile="Default.aspx.cs" Inherits="_Default"%>
```

```
<asp:Content ID="HeaderContent" runat="server" ContentPlaceHolderID="HeadContent">
</asp:Content>
<asp:Content ID="BodyContent" runat="server" ContentPlaceHolderID="MainContent">
    <h2>
```

```
        Welcome to ASP.NET!
    </h2>
```

<p>

To learn more about ASP.NET visit <a href="http://www.asp.net" title="ASP.NET Website">www.asp.net</a>.

</p>

<p>

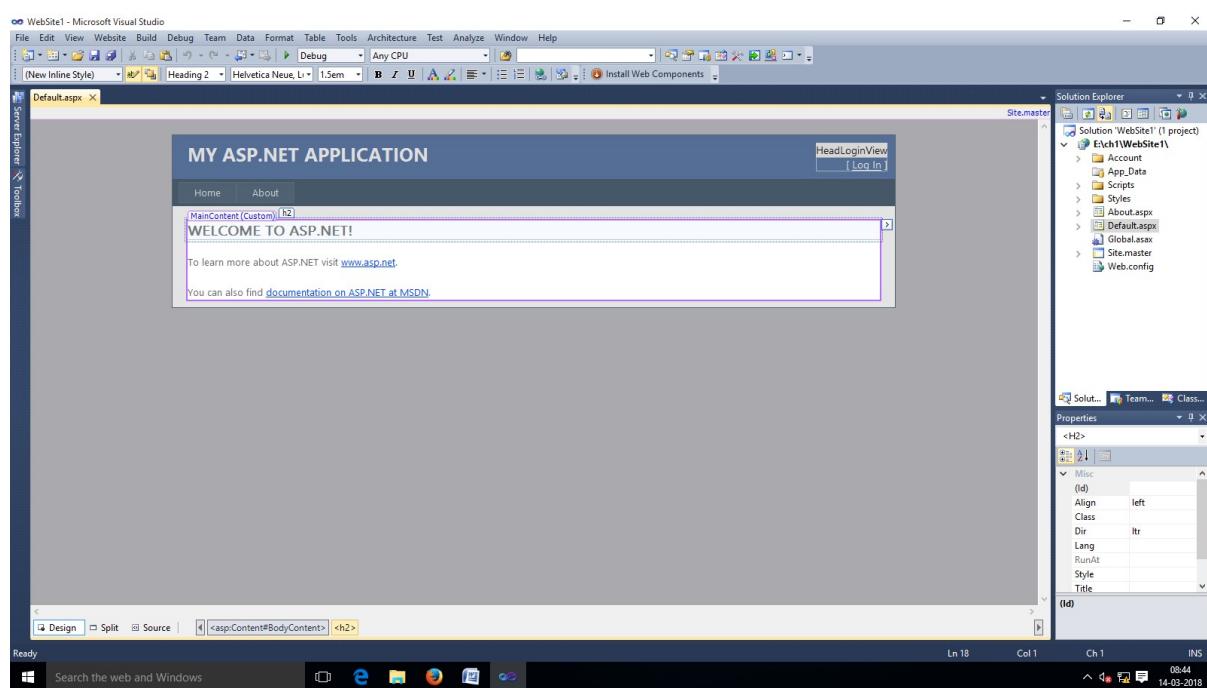
You can also find

<a href="http://go.microsoft.com/fwlink/?LinkId=152368&clcid=0x409" title="MSDN ASP.NET Docs">documentation on ASP.NET at MSDN</a>.

</p>

</asp:Content>

Now at Design side:-



- This is the form where you can take tools from Tool Box like Label, Text Box etc. Its extension is .aspx. & its name is Default. You can rename it.
- You can see Solution Explorer where we can see the Default.aspx file. There is one connected file with it Default.aspx.cs which is code behind file. Where you can code of page as per our needed output i.e. Page\_Load, Click, Double\_Click etc. events.
- There is one web.config file which comes default with website when we create it. In This file there is Configuration, System, Runtime etc. settings are declared.

Now you can put your tools in your form & write code in Code Behind file.

# Index

<b>Sr. No.</b>	<b>Definition of Program</b>
<b>Ch. – 1</b>	<b>Framework &amp; Web Contents Validation Control</b>
1.1	Enter your name in textbox and display it on click of button.
1.2	WAP to display list of selected checkbox and radio button.
1.3	WAP to fill dropdown of city on change event of dropdown of state.
1.4	WAP to find simple interest
1.5	WAP to create calculator
1.6	Create registration form and display it on another page on click of submit button
1.7	Create a web page to show use of adrotate control
1.8	Create a web page to show use of wizard control
1.9	Create a web page to show use of Calendar control
1.10	Create a login page using required field validation control
1.11	Create page with 2 labels and 2 textbox using compare validation control
1.12	Create page using range validation control(enter date in date textbox)
1.13	Create page using regular expression control(enter email in email textbox)
1.14	WAP to check length of name (enter name in textbox and check length using custom validation)
1.15	Create registration form using all validation control and also use validation summary control
<b>Ch. – 2</b>	<b>State Management</b>
2.1	Example of view state
2.2	Create a cookies page which gets name and value of cookies in textbox. It also has a WriteCookie to create and a link button to read the cookie. Initially the read cookie link is disabled.
2.3	WAP to get value of one page to another page using Hidden Fields
2.4	WAP to get value of one page to another page using Query String
2.5	Create login page using session state
2.6	WAP to get visitors count using application state
<b>Ch. – 3</b>	<b>ADO.NET And Database</b>
3.1	Connected architecture example for inserting record into employee table (EmpId, Empname, Designation) and display in gridview and also edit record
3.2	Dis-Connected architecture example for inserting record into employee table (EmpId, Empname, Designation) and also display in gridview and also edit record and next, previous, first, and last navigation
3.3	Insert (imageid, name, path) in image database and Display images in gridview
3.4	Example of repeater control on image database
<b>Ch. – 4</b>	<b>Master Pages and Theme caching, Application Pages and Data</b>
4.1	Example of master page using theme and css (as per .net book)
4.2	WAP of Page output caching

4.3	WAP of partial page caching
4.4	WAP of data caching
<b>Ch. – 5</b>	<b>Working With XML Asp.Net Application configuration and Deployment of Application</b>
5.1	WAP to fetch record from database and write data in XML file
5.2	WAP to fetch record from XML file and display in gridview.
5.3	Example of web service
5.4	Example of tracing
5.5	Example of error handling using custom error
5.6	WAP for authentication and authorization Form authentication Window authentication
5.7	Reading settings from web config file.

# **Chapter-1**

# **Framework & Web**

# **Contents Validation**

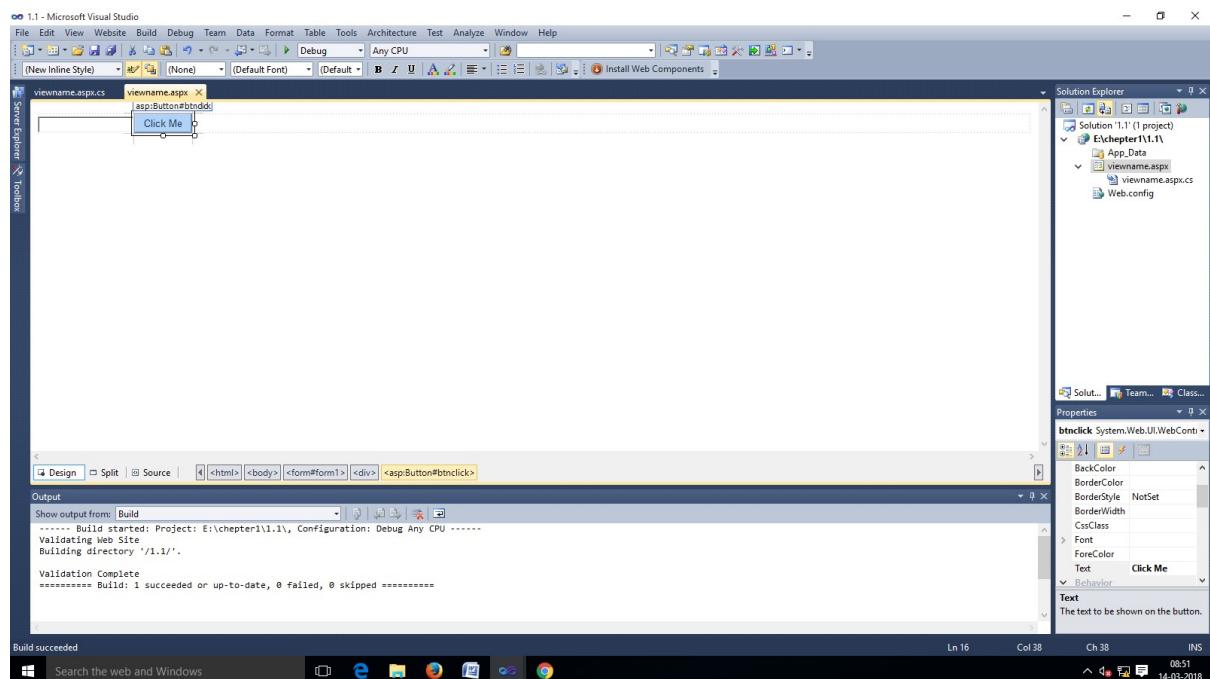
## **Control**

## **1.1) Enter your name in textbox and display it on web page when button is click. [viewname.aspx]**

- **At Design Time:-**

### **Properties:**

Control Name	Property Name	Value
Textbox	ID	txtname
Button1	ID	btnclick
	Text	Click Me



### **For viewname.Aspx.cs page:-**

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

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

    }

    protected void btnclick_Click(object sender, EventArgs e)
    {
        Response.Write(txtname.Text);
    }
}
```

**At Run Time:-**

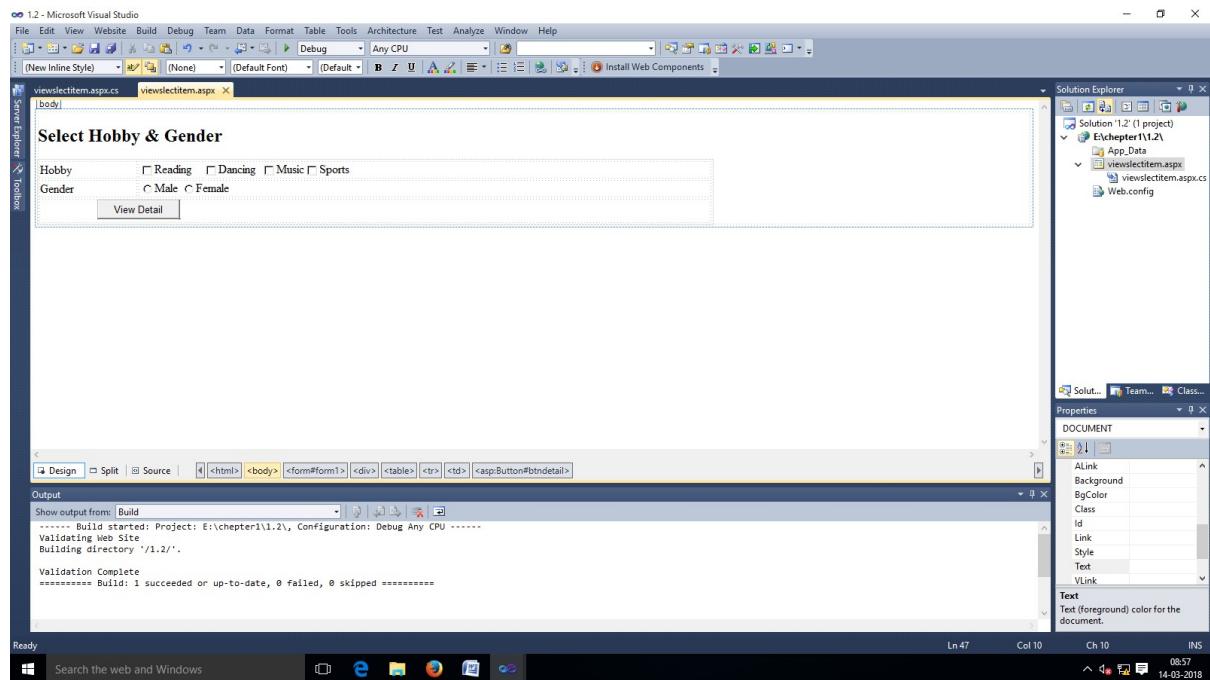


**After Pressing ClickMe button:**



## 1.2)WAP to display list of selected checkbox and radio button. [viewselectitem.aspx]

At Design Time:-



For viewselectitem.Asp.cs page:-

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

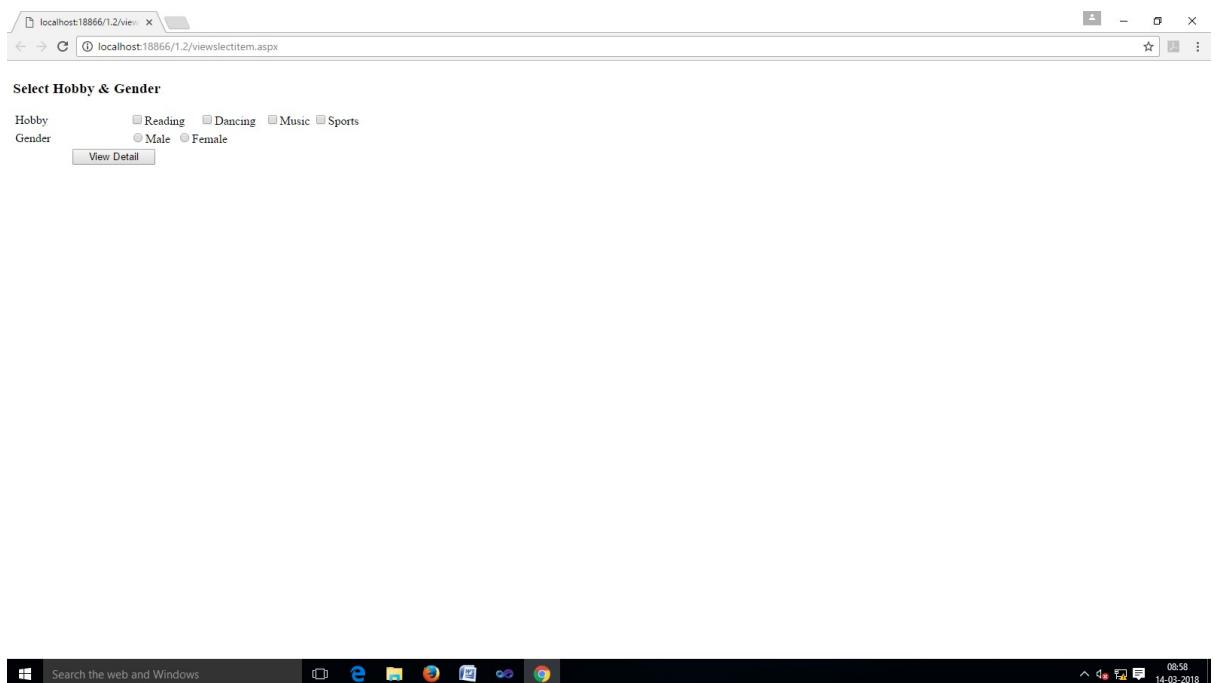
public partial class viewselectitem : System.Web.UI.Page
{
    Protected void Page_Load(object sender, EventArgs e)
    {

    }

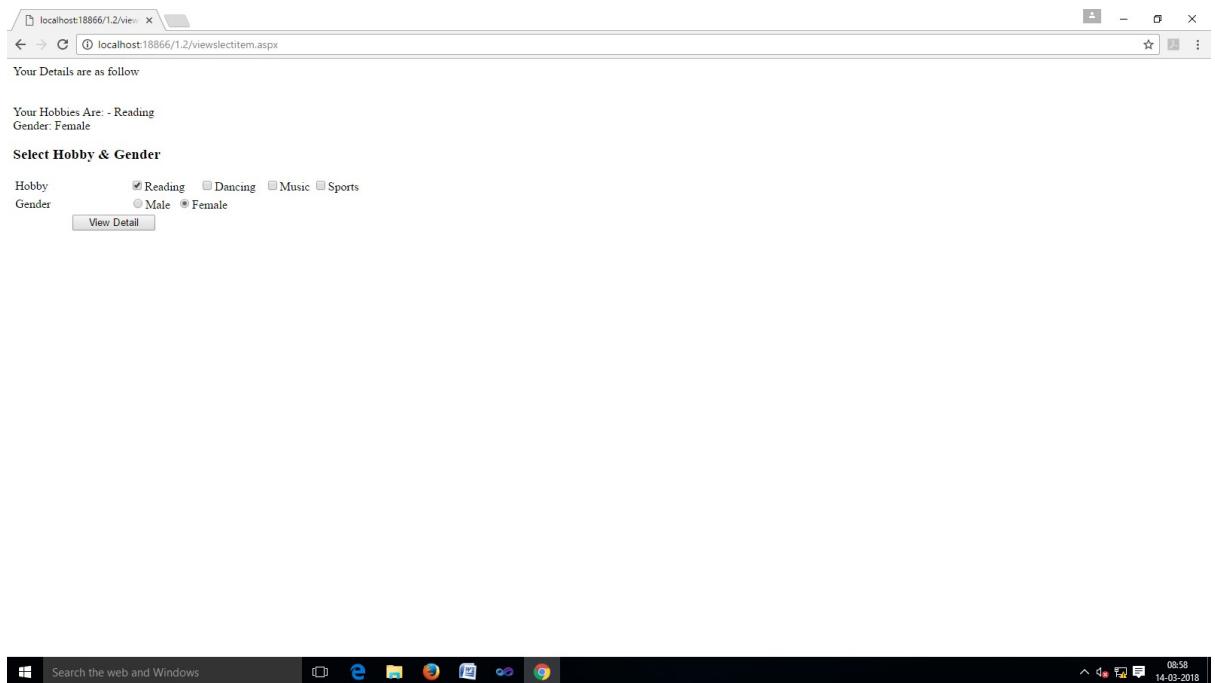
    Protected void btndetail_Click(object sender, EventArgs e)
    {
        Response.Write("Your Details are as follow<br><br>");
        if (chkdancing.Checked == false && chkmusic.Checked == false && chkreading.Checked == false && chksports.Checked == false)
        {
            Response.Write("<br>No Hobbies Selected");
        }
        else
        {
            Response.Write("<br>Your Hobbies Are:");
        }
        if (chkmusic.Checked == true)
        {
            Response.Write("- Music");
        }
        if (chkreading.Checked == true)
```

```
{  
    Response.Write(" - Reading");  
}  
if(chksports.Checked == true)  
{  
    Response.Write(" - Sports");  
}  
if(chkdancing.Checked == true)  
{  
    Response.Write(" - Dancing");  
}  
  
if(rbtfemale.Checked == true)  
{  
    Response.Write("<br>Gender: Female");  
}  
else  
{  
    Response.Write("<br>Gender: Male");  
}  
}  
}
```

### **At Run Time:-**

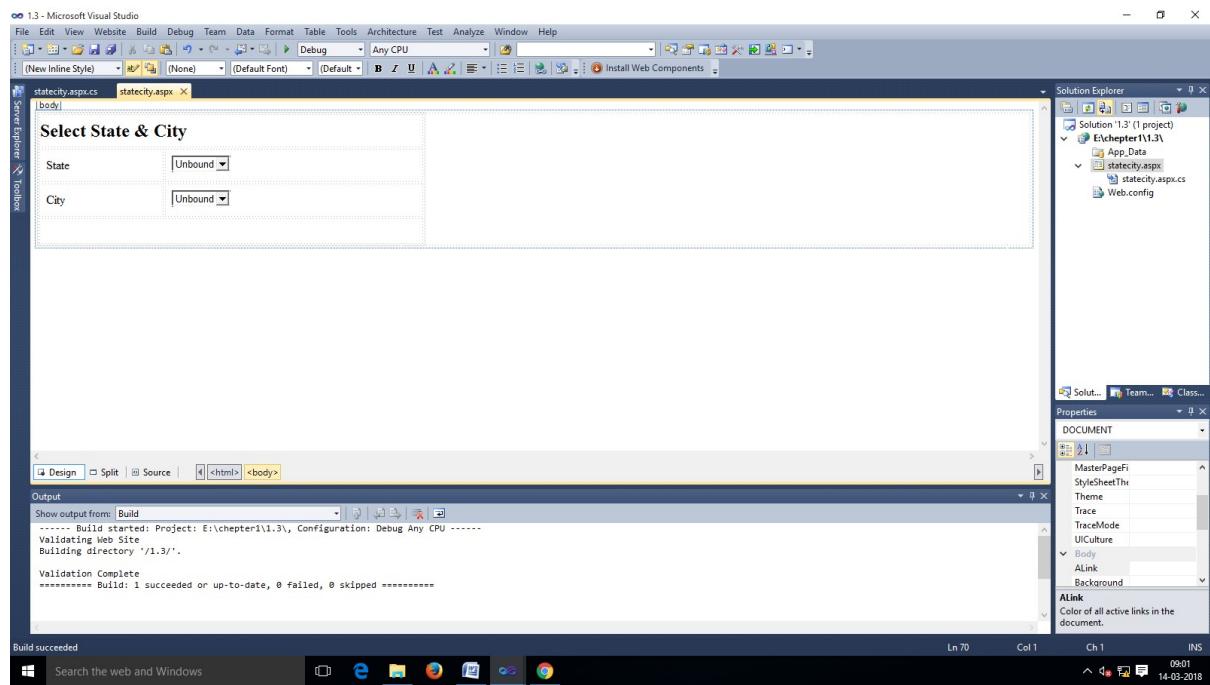


**After Pressing View Details button:**



### 1.3) WAP to fill dropdown of city on change event of dropdown of state.[selectcity.aspx]

**At Design Time:-**



**NOTE:-**In this form you have to set **AutoPostback** property **true** for state dropdown list.

**For selectcity.aspx.cs page:-**

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

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

            this.AddDefaultStateCities();
        }
    }

    public void AddDefaultStateCities()
    {

        ddlstate.Items.Add("Gujarat");
        ddlstate.Items.Add("Maharashtra");
        ddlstate.Items.Add("Rajasthan");
        ddlcity.Items.Clear();
        if (ddlstate.SelectedValue == "Gujarat")
        {
            ddlcity.Items.Add("Rajkot");
            ddlcity.Items.Add("Jamnagar");
        }
    }
}
```

```
ddlcity.Items.Add("Junagadh");
}

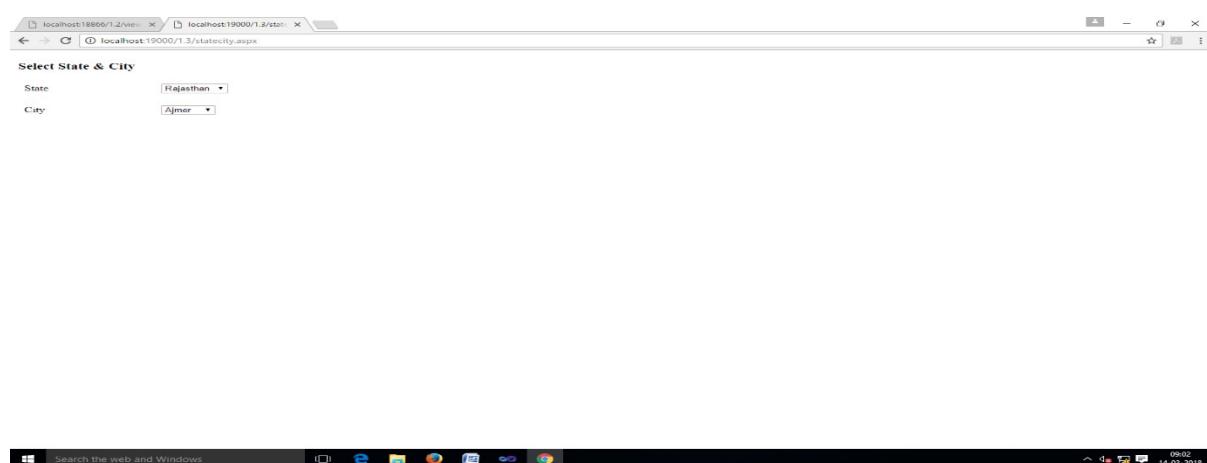
elseif(ddlstate.SelectedValue == "Maharastra")
{
    ddlcity.Items.Add("Mumbai");
    ddlcity.Items.Add("Pune");
    ddlcity.Items.Add("Panchgini");
}
elseif(ddlstate.SelectedValue == "Rajasthan")
{
    ddlcity.Items.Add("Ajmer");
    ddlcity.Items.Add("Pushkar");
    ddlcity.Items.Add("Jaipur");
}

}

protectedvoid ddlstate_SelectedIndexChanged(object sender, EventArgs e)
{
    ddlcity.Items.Clear();
if(ddlstate.SelectedValue == "Gujarat")
{
    ddlcity.Items.Add("Rajkot");
    ddlcity.Items.Add("Jamnagar");
    ddlcity.Items.Add("Junagadh");

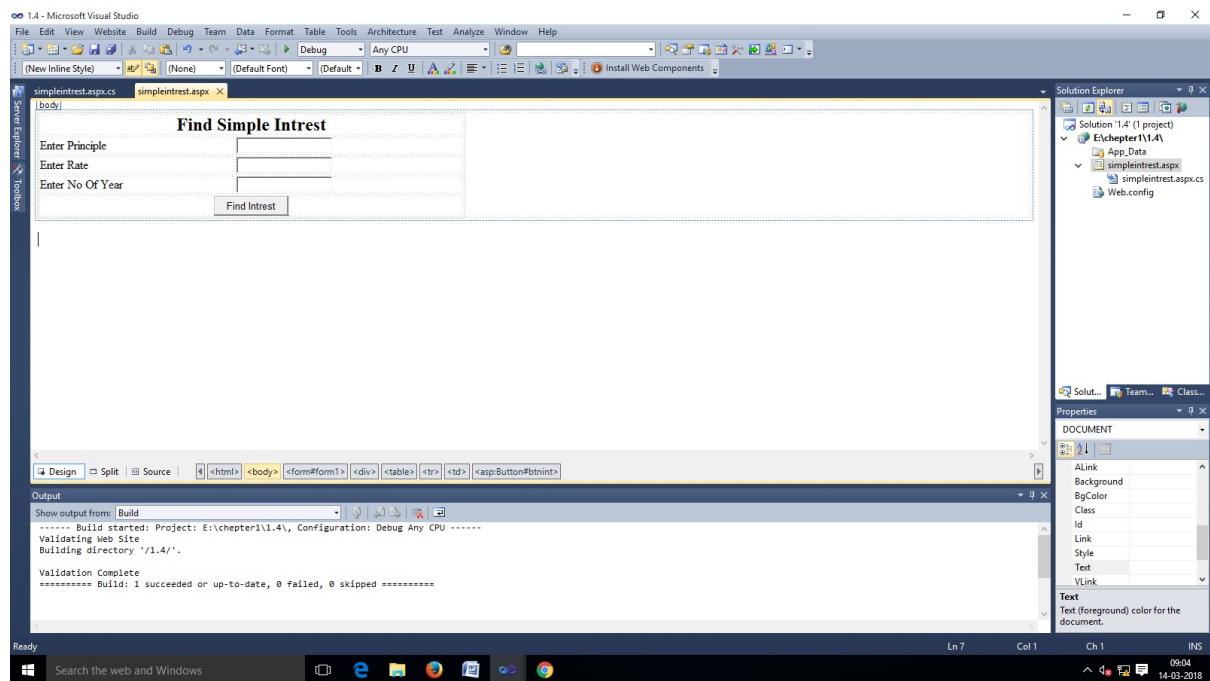
}
elseif(ddlstate.SelectedValue == "Maharastra")
{
    ddlcity.Items.Add("Mumbai");
    ddlcity.Items.Add("Pune");
    ddlcity.Items.Add("Panchgini");
}
elseif(ddlstate.SelectedValue == "Rajasthan")
{
    ddlcity.Items.Add("Ajmer");
    ddlcity.Items.Add("Pushkar");
    ddlcity.Items.Add("Jaipur");
}
}
```

### **At Run Time:-**



### **1.4) WAP to find simple interest.[simpleinterest.aspx]**

**At Design Time:-**



**For simpleintrest.Aspx.cs page:-**

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

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

    protected void btnint_Click(object sender, EventArgs e)
    {
        int p = Convert.ToInt32(txtp.Text);
        int r = Convert.ToInt32(txtr.Text);
        int n = Convert.ToInt32(txtn.Text);
        int simplist = (p * r * n) / 100;
        Response.Write("<h1>Simpl Intrest Is=" + simplist + "</h1>");
    }
}
```

**At Run Time:-**

localhost:18866/1.2/view... localhost:19000/1.3/sta... localhost:19116/1.4/simpleintrest.aspx

Find Simple Intrest

Enter Principle: 1000

Enter Rate: 10

Enter No Of Year: 2

Find Intrest

**After Pressing find instrest button:**

localhost:18866/1.2/view... localhost:19000/1.3/sta... localhost:19116/1.4/simpleintrest.aspx

Simpl Intrest Is=200

Find Simple Intrest

Enter Principle: 1000

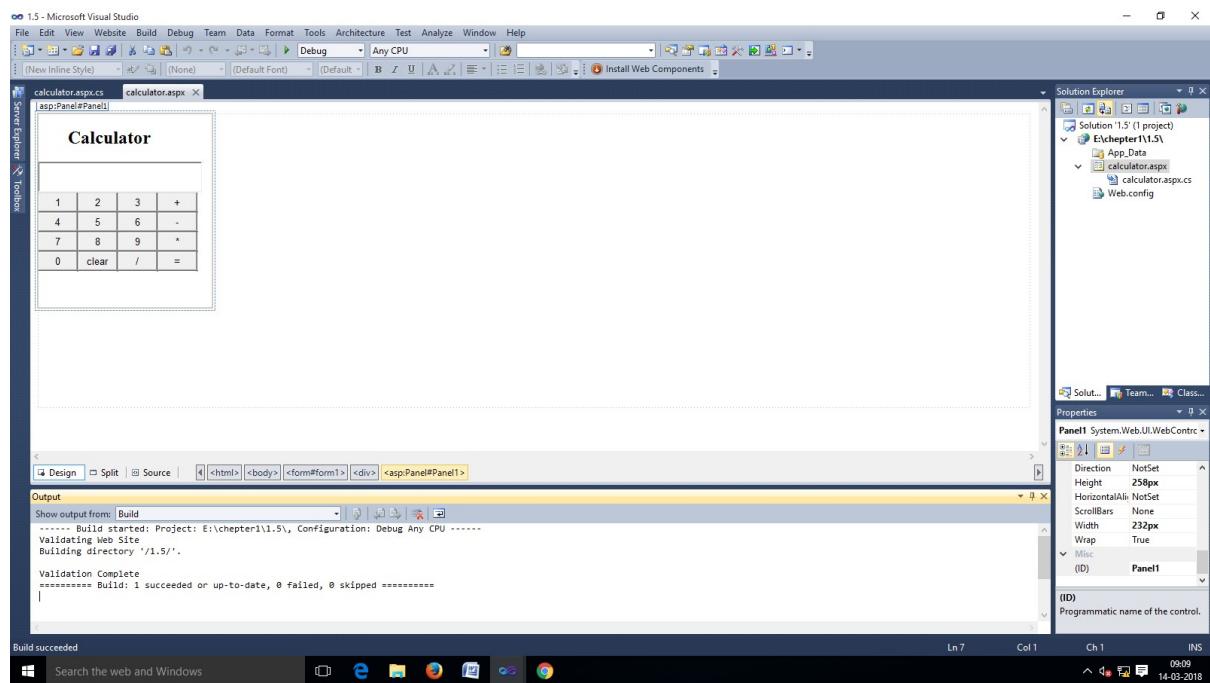
Enter Rate: 10

Enter No Of Year: 2

Find Intrest

## 1.5)WAP to create calculator.[calculator.aspx]

**At Design Time:-**



**For calculatorAspx.cs page:-**

```

using System;
using System.Collections;
using System.Configuration;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.HtmlControls;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Xml.Linq;

public partial class calculator : System.Web.UI.Page
{
    static float a, c, d;
    static char b;
    protected void Page_Load(object sender, EventArgs e)
    {
    }
    protected void btn1_Click(object sender, EventArgs e)
    {
        if ((TextBox1.Text == "+") || (TextBox1.Text == "-") || (TextBox1.Text == "*") || (TextBox1.Text == "/"))
        {
            TextBox1.Text = "";
            TextBox1.Text = TextBox1.Text + btn1.Text;
        }
        else
        {
            TextBox1.Text = TextBox1.Text + btn1.Text;
        }
    }
}

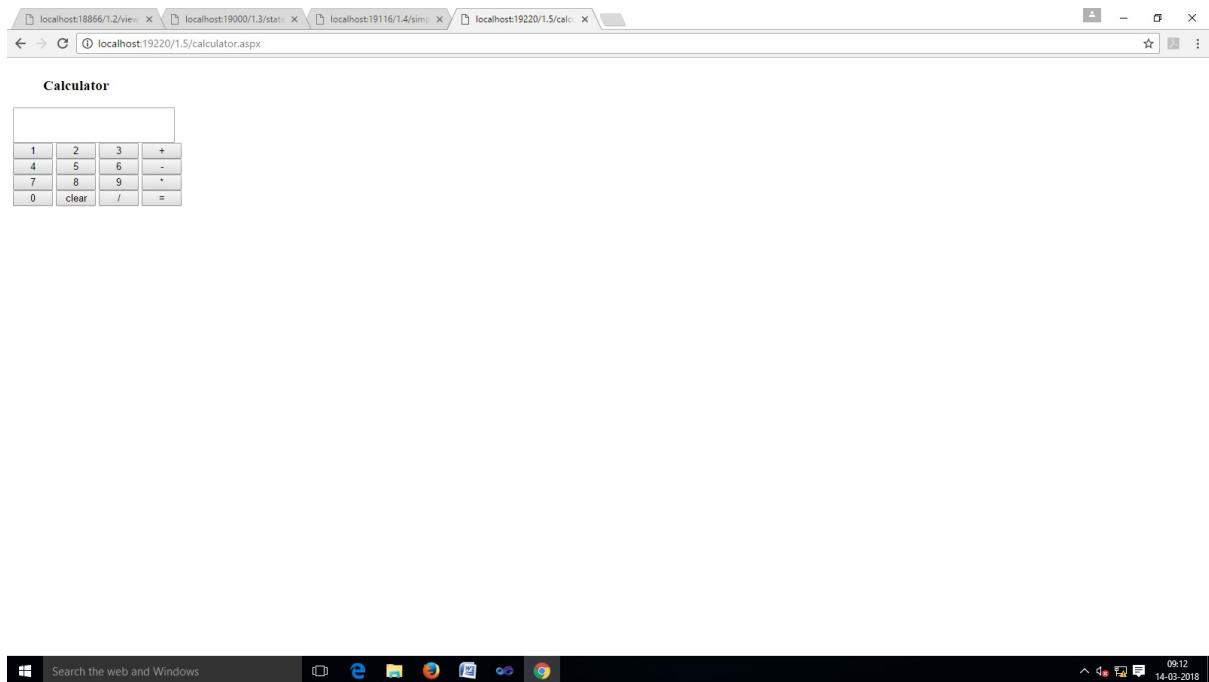
```

```
        }
    protectedvoid btn2_Click(object sender, EventArgs e)
    {
        if((TextBox1.Text == "+") || (TextBox1.Text == "-") || (TextBox1.Text == "*") || (TextBox1.Text == "/"))
        {
            TextBox1.Text = "";
            TextBox1.Text = TextBox1.Text + btn2.Text;
        }
    else
        {
            TextBox1.Text = TextBox1.Text + btn2.Text;
        }
    }
protectedvoid btn3_Click(object sender, EventArgs e)
{
if((TextBox1.Text == "+") || (TextBox1.Text == "-") || (TextBox1.Text == "*") || (TextBox1.Text == "/"))
{
    TextBox1.Text = "";
    TextBox1.Text = TextBox1.Text + btn3.Text;
}
else
{
    TextBox1.Text = TextBox1.Text + btn3.Text;
}
}
protectedvoid btnpls_Click(object sender, EventArgs e)
{
    a = Convert.ToInt32(TextBox1.Text);
    TextBox1.Text = "";
    b = '+';
    TextBox1.Text += b;
}
protectedvoid btn4_Click(object sender, EventArgs e)
{
if((TextBox1.Text == "+") || (TextBox1.Text == "-") || (TextBox1.Text == "*") || (TextBox1.Text == "/"))
{
    TextBox1.Text = "";
    TextBox1.Text = TextBox1.Text + btn4.Text;
}
else
{
    TextBox1.Text = TextBox1.Text + btn4.Text;
}
}
protectedvoid btn5_Click(object sender, EventArgs e)
{
if((TextBox1.Text == "+") || (TextBox1.Text == "-") || (TextBox1.Text == "*") || (TextBox1.Text == "/"))
{
    TextBox1.Text = "";
    TextBox1.Text = TextBox1.Text + btn5.Text;
}
else
{
    TextBox1.Text = TextBox1.Text + btn5.Text;
}
}
protectedvoid btn6_Click(object sender, EventArgs e)
{
if((TextBox1.Text == "+") || (TextBox1.Text == "-") || (TextBox1.Text == "*") || (TextBox1.Text == "/"))
{
    TextBox1.Text = "";
    TextBox1.Text = TextBox1.Text + btn6.Text;
}
else
```

```
{  
    TextBox1.Text = TextBox1.Text + btn6.Text;  
}  
}  
protectedvoid btnmins_Click(object sender, EventArgs e)  
{  
    a = Convert.ToInt32(TextBox1.Text);  
    TextBox1.Text = "";  
    b = '-';  
    TextBox1.Text += b;  
}  
protectedvoid btn7_Click(object sender, EventArgs e)  
{  
if((TextBox1.Text == "+") || (TextBox1.Text == "-") || (TextBox1.Text == "*") || (TextBox1.Text == "/"))  
{  
    TextBox1.Text = "";  
    TextBox1.Text = TextBox1.Text + btn7.Text;  
}  
else  
{  
    TextBox1.Text = TextBox1.Text + btn7.Text;  
}  
}  
protectedvoid btn8_Click(object sender, EventArgs e)  
{  
if((TextBox1.Text == "+") || (TextBox1.Text == "-") || (TextBox1.Text == "*") || (TextBox1.Text == "/"))  
{  
    TextBox1.Text = "";  
    TextBox1.Text = TextBox1.Text + btn8.Text;  
}  
else  
{  
    TextBox1.Text = TextBox1.Text + btn8.Text;  
}  
}  
protectedvoid btn9_Click(object sender, EventArgs e)  
{  
if((TextBox1.Text == "+") || (TextBox1.Text == "-") || (TextBox1.Text == "*") || (TextBox1.Text == "/"))  
{  
    TextBox1.Text = "";  
    TextBox1.Text = TextBox1.Text + btn9.Text;  
}  
else  
{  
    TextBox1.Text = TextBox1.Text + btn9.Text;  
}  
}  
protectedvoid btnmulti_Click(object sender, EventArgs e)  
{  
    a = Convert.ToInt32(TextBox1.Text);  
    b = '*';  
    TextBox1.Text = "";  
    TextBox1.Text += b;  
}  
protectedvoid btn0_Click(object sender, EventArgs e)  
{  
if((TextBox1.Text == "+") || (TextBox1.Text == "-") || (TextBox1.Text == "*") || (TextBox1.Text == "/"))  
{  
    TextBox1.Text = "";  
    TextBox1.Text = TextBox1.Text + btn0.Text;  
}  
else  
{  
    TextBox1.Text = TextBox1.Text + btn0.Text;  
}
```

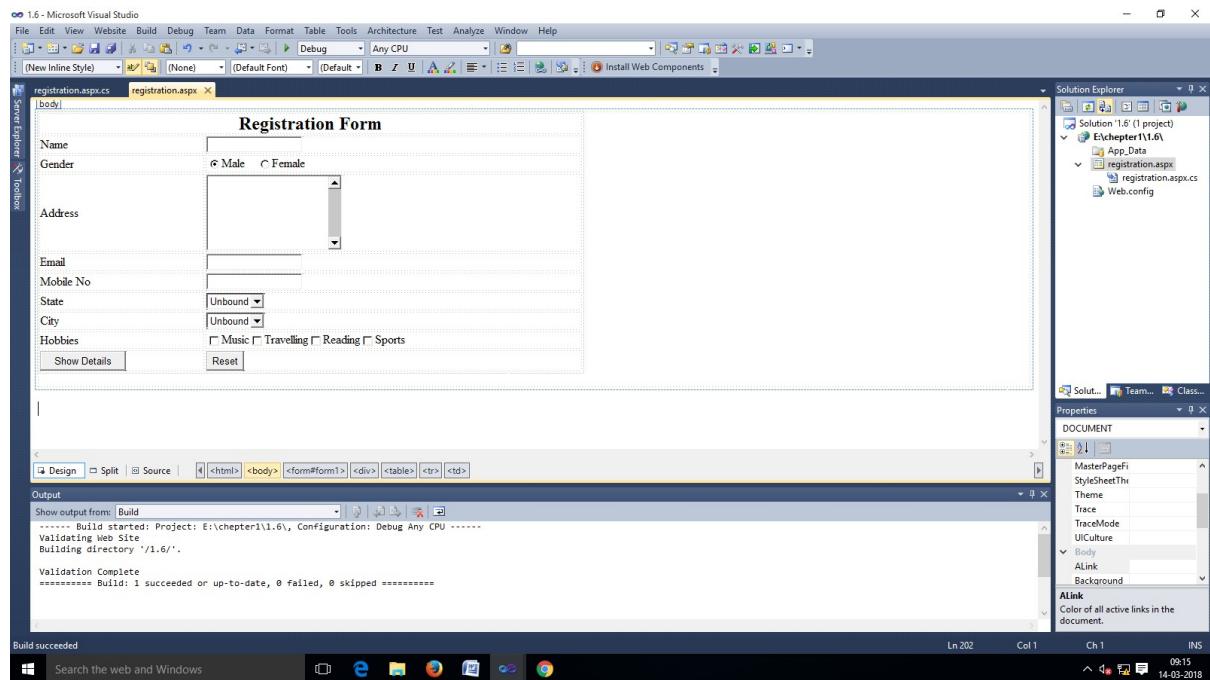
```
        }
    }
protectedvoid btnclear_Click(object sender, EventArgs e)
{
    TextBox1.Text = "";
}
protectedvoid btndiv_Click(object sender, EventArgs e)
{
    a = Convert.ToInt32(TextBox1.Text);
    TextBox1.Text = "";
    b = '/';
    TextBox1.Text += b;
}
protectedvoid btnans_Click(object sender, EventArgs e)
{
    c = Convert.ToInt32(TextBox1.Text);
    TextBox1.Text = "";
if(b == '/')
{
    d = a / c;
    TextBox1.Text += d;
    a = d;
}
elseif(b == '+')
{
    d = a + c;
    TextBox1.Text += d;
    a = d;
}
elseif(b == '-')
{
    d = a - c;
    TextBox1.Text += d;
    a = d;
}
else
{
    d = a * c;
    TextBox1.Text += d;
    a = d;
}
}
```

**At Run Time:-**



## **1.6) Create registration form and display it on another page on click of submit button.[registration.aspx]**

**At Design Time:-**



**For registration.aspx.cs page:-**

```

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

public partial class reg : System.Web.UI.Page
{
    Boolean a;
    protected void Page_Load(object sender, EventArgs e)
    {
        if (!IsPostBack)
        {
            rdgender.SelectedIndex = 0;
            this.AddDefaultStateCities();
        }
    }
    public void AddDefaultStateCities()
    {

        ddlState.Items.Add("Gujarat");
        ddlState.Items.Add("Maharashtra");
        ddlState.Items.Add("Rajasthan");
        ddlCity.Items.Clear();
        if (ddlState.SelectedValue == "Gujarat")
        {
            ddlCity.Items.Add("Rajkot");
            ddlCity.Items.Add("Jamnagar");
            ddlCity.Items.Add("Junagadh");

        }
        elseif (ddlState.SelectedValue == "Maharashtra")
        {
            ddlCity.Items.Add("Mumbai");
            ddlCity.Items.Add("Pune");
            ddlCity.Items.Add("Panchgani");
        }
        elseif (ddlState.SelectedValue == "Rajasthan")
        {
            ddlCity.Items.Add("Ajmer");
            ddlCity.Items.Add("Pushkar");
            ddlCity.Items.Add("Jaipur");
        }
    }

    protected void ddlState_SelectedIndexChanged(object sender, EventArgs e)
    {
        ddlCity.Items.Clear();
        if (ddlState.SelectedValue == "Gujarat")
        {
            ddlCity.Items.Add("Rajkot");
            ddlCity.Items.Add("Jamnagar");
            ddlCity.Items.Add("Junagadh");

        }
        elseif (ddlState.SelectedValue == "Maharashtra")
        {
            ddlCity.Items.Add("Mumbai");
            ddlCity.Items.Add("Pune");
            ddlCity.Items.Add("Panchgani");
        }
        elseif (ddlState.SelectedValue == "Rajasthan")
    }
}

```

```
{  
    ddlCity.Items.Add("Ajmer");  
    ddlCity.Items.Add("Pushkar");  
    ddlCity.Items.Add("Jaipur");  
}  
}  
protected void btnShowDetails_Click(object sender, EventArgs e)  
{  
  
    Response.Write(" Your Details are: ");  
  
    if (txtName.Text.Length == 0)  
    {  
        Response.Write("<p>Name is blank");  
    }  
    else  
    {  
        Response.Write("<p>Name:" + txtName.Text);  
    }  
  
    Response.Write("<p>you gender : " + rdgender.SelectedItem.ToString());  
  
    if (txtAddress.Text.Length == 0)  
    {  
        Response.Write("<br>No Address Entered");  
    }  
    else  
    {  
        Response.Write("<br>Address: " + txtAddress.Text);  
    }  
  
    if (txtemail.Text.Length == 0)  
    {  
        Response.Write("<br>No Email Entered");  
    }  
    else  
    {  
        Response.Write("<br>Email: " + txtemail.Text);  
    }  
  
    if (txtmobile.Text.Length == 0)  
    {  
        Response.Write("<br>No MobileNo Entered");  
    }  
    else  
    {  
        Response.Write("<br>Mobile: " + txtmobile.Text);  
    }  
    Response.Write("<br>State:" + ddlState.Text);  
    Response.Write("<br>City:" + ddlCity.Text);  
    Response.Write("<BR>Hobbies :");  
    foreach (ListItem li in chkhobby.Items)  
    {  
        if (li.Selected == true)  
        {  
            Response.Write(li.Text + " ");  
        }  
    }  
}  
  
protected void btnReset_Click(object sender, EventArgs e)  
{  
    txtName.Text = "";  
    txtAddress.Text = "";  
    rdgender.ClearSelection();  
}
```

```
foreach (ListItem li in chkhobby.Items)
{
if (li.Selected == true)
{
    li.Selected = false;
}
txtemail.Text = "";
txtmobile.Text = "";
}
}
```

**At Run Time:-**

The screenshot shows a registration form titled "Registration Form". The form fields include:

- Name: Text input field
- Gender: Radio buttons for Male (selected) and Female
- Address: Text area
- Email: Text input field
- Mobile No: Text input field
- State: Dropdown menu set to Gujarat
- City: Dropdown menu set to Rajkot
- Hobbies: Checkboxes for Music, Travelling, Reading, Sports (all unchecked)
- Show Details: Link
- Reset: Button

The browser taskbar shows multiple open tabs, and the system tray indicates the date as 14-03-2018 and time as 09:16.

**After Pressing ShowDetails button:**

The screenshot shows a web browser window with multiple tabs open, but the main content area displays a registration form titled "Registration Form". The form fields include Name (vibhuti), Gender (Female selected), Address (Timbavadi Swetalpark Society Behind Satname Garege Junagadh 362015), Email (vibhuti.vala@gmail.com), Mobile No (9016606539), State (Gujarat), City (Junagadh), and Hobbies (Music, Travelling, Reading, Sports). A "Show Details" button is visible on the left side of the form. The browser's status bar at the bottom shows the date as 14-03-2018 and the time as 09:17.

Your Details are:

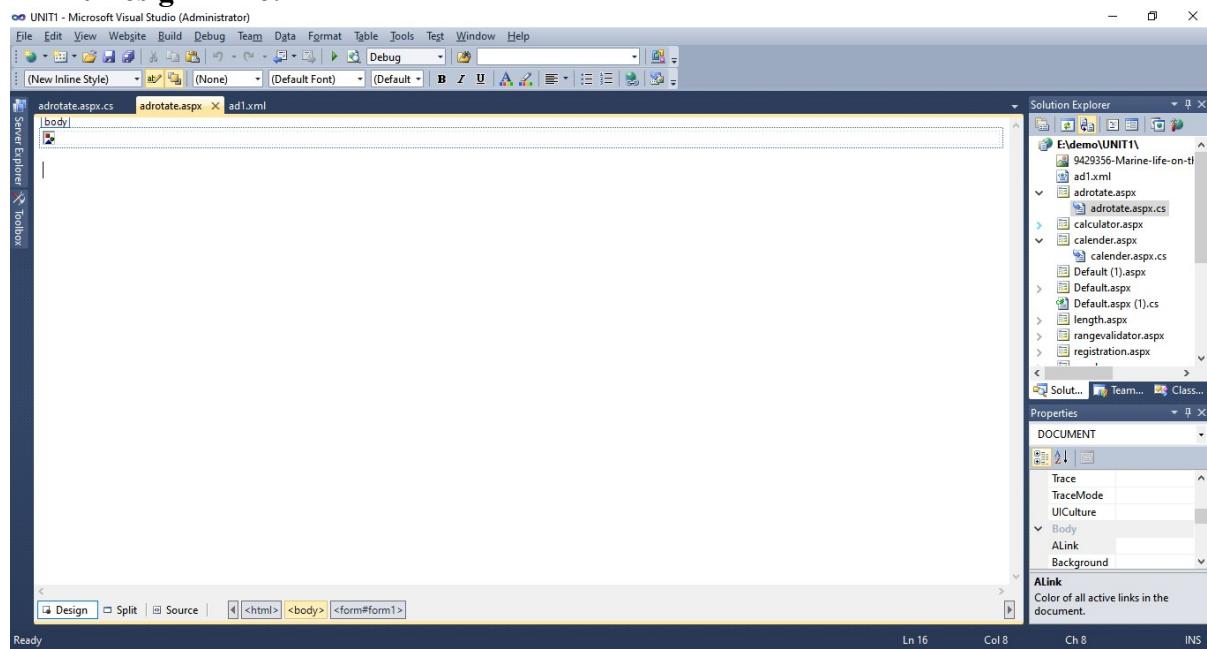
Name: vibhuti  
Gender: Female  
Address: Timbavadi Swetalpark Society Behind Satname Garege Junagadh 362015  
Email: vibhuti.vala@gmail.com  
Mobile: 9016606539  
State: Gujarat  
City: Junagadh  
Hobbies Are:  
- Music

**Registration Form**

Name: vibhuti  
Gender:  Male  Female  
Address: Timbavadi Swetalpark Society Behind Satname Garege Junagadh 362015  
Email: vibhuti.vala@gmail.com  
Mobile No: 9016606539  
State: Gujarat  
City: Junagadh  
Hobbies:  Music  Travelling  Reading  Sports  
[Show Details] [Reset]

## 1.7) Create a web page to show use of adrotate control. [adrotate.aspx]

At Design Time:-



For ad1.xml

```
<?xml version="1.0" encoding="utf-8" ?>
<Advertisements>
<Ad>
<ImageUrl>rose1.jpg</ImageUrl>
<NavigateUrl>http://www.1800flowers.com</NavigateUrl>
<AlternateText>Order flowers, roses, gifts and more</AlternateText>
<Impressions>20</Impressions>
<Keyword>flowers</Keyword>
</Ad>

<Ad>
<ImageUrl>rose2.jpg</ImageUrl>
<NavigateUrl>http://www.babybouquets.com.au</NavigateUrl>
</Ad>

</Advertisements>
```

For adrotate.aspx

```
<%@Page Language="C#" AutoEventWireup="true" CodeFile="adrotate.aspx.cs" Inherits="adrotate1"%>

<!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>
```

**Subject :- ASP.NET**

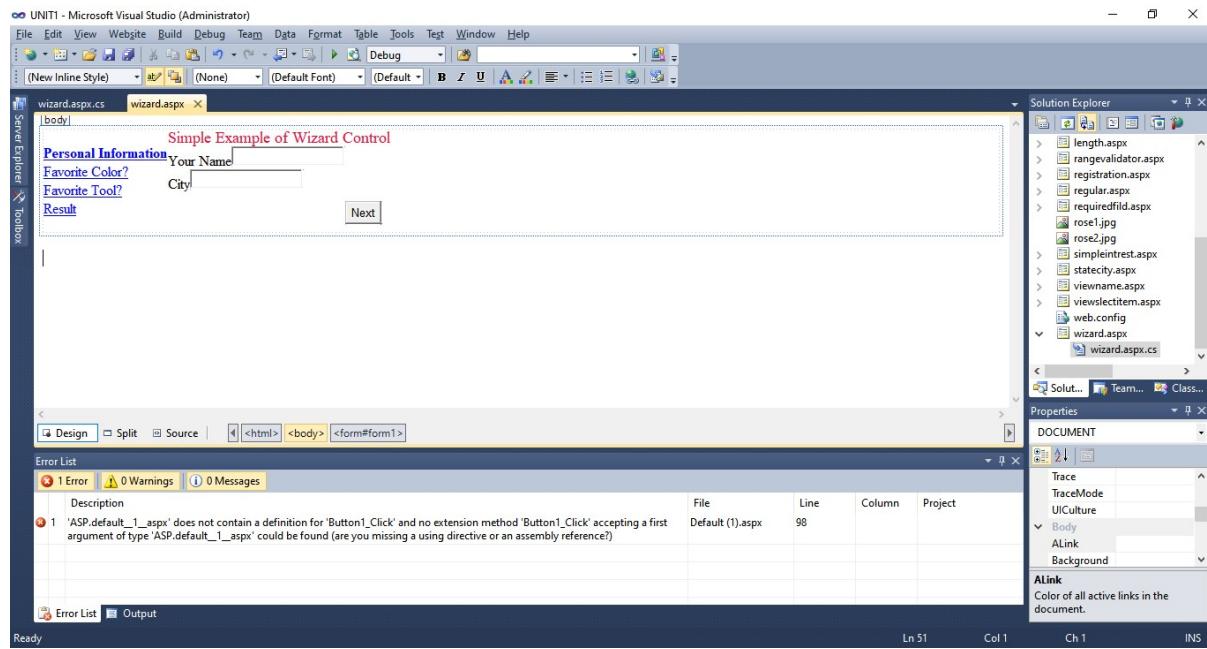
---

```
<asp:AdRotatorID="AdRotator1" runat="server" AdvertisementFile="~/ad1.xml"/>
</div>
</form>
</body>
</html>
```

**At Run time**

## **1.8) Create a web page to show use of wizard control.[wizard.aspx]**

**At Design Time:-**



## For wizard.aspx

```
<%@PageLanguage="C#"AutoEventWireup="true"CodeFile="wizard.aspx.cs" Inherits="wizard_demo"%>

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

<htmlxmlns="http://www.w3.org/1999/xhtml">
<headrunat="server">
<title></title>
</head>
<body>
<formid="form1"runat="server">
<div>
<asp:WizardID="Wizard1"runat="server"HeaderText="Simple Example of Wizard Control"HeaderStyle-Font-Size="Larger"HeaderStyle-ForeColor="Crimson">
<WizardSteps>
<asp:WizardStepID="WizadStep1"runat="server"title="Personal Information">
<asp:LabelID="Label1"runat="server"Text="Your Name"AssociatedControlID="TextBox1"></asp:Label>
<asp:TextBoxID="TextBox1"runat="server"></asp:TextBox>
<br/>
<asp:LabelID="Label2"runat="server"Text="City"AssociatedControlID="TextBox2"></asp:Label>
<asp:TextBoxID="TextBox2"runat="server"></asp:TextBox>
</asp:WizardStep>

<asp:WizardStepID="WizardStep2"runat="server"title="Favorite Color?">
<asp:LabelID="Label3"runat="server"Text="Favorite Color?"AssociatedControlID="RadioButtonList1"></asp:Label>
<asp:RadioButtonListID="RadioButtonList1"runat="server">
<asp:ListItemSelected="True">Red</asp:ListItem>
<asp:ListItem>Green</asp:ListItem>
<asp:ListItem>Blue</asp:ListItem>
</asp:RadioButtonList>
</asp:WizardStep>

<asp:WizardStepID="WizardStep3"runat="server"title="Favorite Tool?">
<asp:LabelID="Label4"runat="server"Text="Most Favorite?"AssociatedControlID="RadioButtonList2"></asp:Label>
<asp:RadioButtonListID="RadioButtonList2"runat="server">
<asp:ListItemSelected="True">Asp.Net</asp:ListItem>
<asp:ListItem>ColdFusion</asp:ListItem>
<asp:ListItem>PHP</asp:ListItem>
</asp:RadioButtonList>
</asp:WizardStep>
<asp:WizardStepID="WizardStep4"runat="server"Title="Result"OnActivate="WizardStep4_Activate">
<asp:LabelID="Label5"runat="server"Font-Size="X-Large"ForeColor="DarkGoldenrod"></asp:Label>
</asp:WizardStep>
</WizardSteps>
</asp:Wizard>

</div>
</form>
</body>
</html>
```

## For wizard.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 wizard_demo : System.Web.UI.Page
{
protected void Page_Load(object sender, EventArgs e)
{
}

protected void WizardStep4_Activate(object sender, System.EventArgs e)
{
    Label5.Text = "Your Name: " + TextBox1.Text.ToString() +
"<br />City: " + TextBox2.Text.ToString() +
"<br />Favorite Color: " + RadioButtonList1.SelectedItem.Text.ToString() +
"<br />Favorite Tool: " + RadioButtonList2.SelectedItem.Text.ToString();
}
}
```

### **At run time:-**

#### **Personal Information**



### **Favorite color?:-**

The screenshot shows a web browser window with the URL `localhost:58419/UNIT1/wizard.aspx`. The title bar says "Simple Example of Wizard Control". On the left, there is a vertical menu with links: "Personal Information", "Favorite Color?", "Favorite Tool?", and "Result". The main content area asks "Favorite Color?" with three radio button options: "Red", "Green", and "Blue". The "Blue" option is selected. At the bottom are "Previous" and "Next" buttons.

### **Favorite Tool?:-**

The screenshot shows a web browser window with the URL `localhost:58419/UNIT1/wizard.aspx`. The title bar says "Simple Example of Wizard Control". On the left, there is a vertical menu with links: "Personal Information", "Favorite Color?", "Favorite Tool?", and "Result". The main content area asks "Most Favorite?" with three radio button options: "Asp.Net", "ColdFusion", and "PHP". The "Asp.Net" option is selected. At the bottom are "Previous" and "Next" buttons.

### **Result:-**

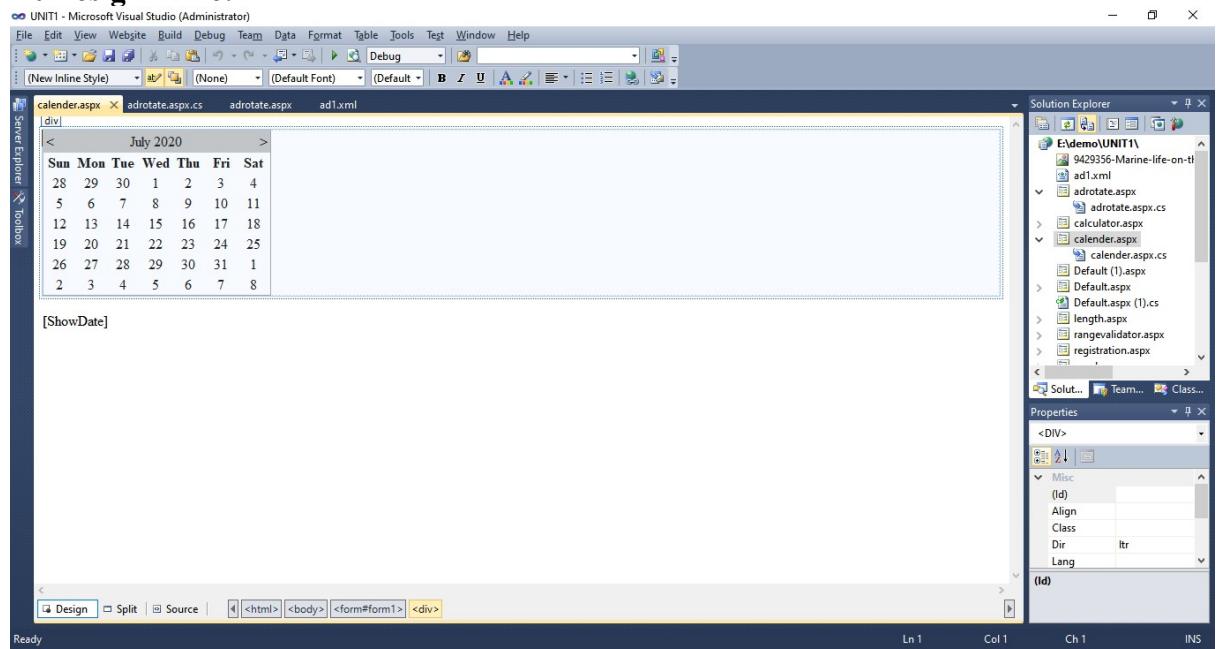
The screenshot shows a web browser window with the title bar "localhost" and the URL "localhost:58419/UNIT1/wizard.aspx". The page content is titled "Simple Example of Wizard Control". It displays a series of questions and their answers:

- Personal Information**: Your Name: BHAVISHA
- Favorite Color?**: City: JUNAGADH
- Favorite Tool?**: Favorite Color: Blue
- Result**: Favorite Tool: Asp.Net

At the bottom of the form are two buttons: "Previous" and "Finish".

## **1.9) Create a web page to show use of calendar control. [calender.aspx]**

### **At Design Time:-**



### **For calender.aspx.cs**

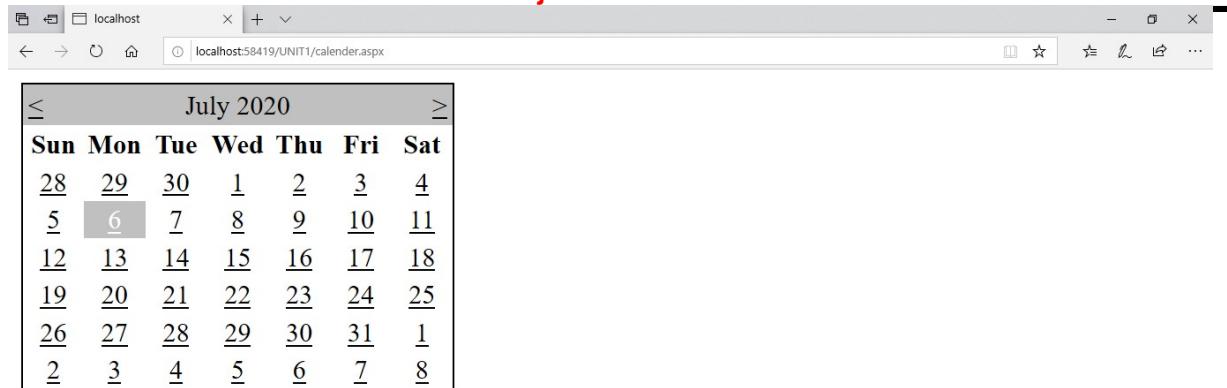
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

public partial class calender1 : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
    }

    protected void Calendar1_SelectionChanged(object sender, EventArgs e)
    {
        ShowDate.Text = "You Selected: " + Calendar1.SelectedDate.ToString("D");
    }
}
```

### **At Runtime:-**

**Subject :- ASP.NET**

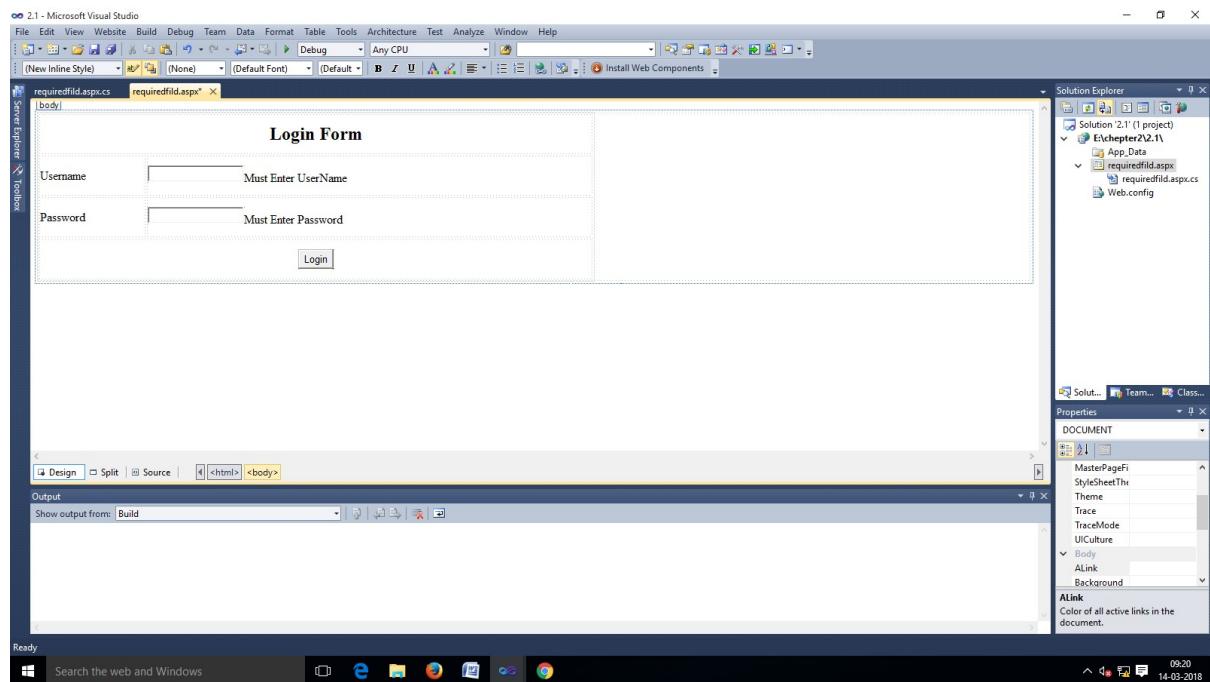


July 2020						
Sun	Mon	Tue	Wed	Thu	Fri	Sat
<u>28</u>	<u>29</u>	<u>30</u>	<u>1</u>	<u>2</u>	<u>3</u>	<u>4</u>
<u>5</u>	<b>6</b>	<u>7</u>	<u>8</u>	<u>9</u>	<u>10</u>	<u>11</u>
<u>12</u>	<u>13</u>	<u>14</u>	<u>15</u>	<u>16</u>	<u>17</u>	<u>18</u>
<u>19</u>	<u>20</u>	<u>21</u>	<u>22</u>	<u>23</u>	<u>24</u>	<u>25</u>
<u>26</u>	<u>27</u>	<u>28</u>	<u>29</u>	<u>30</u>	<u>31</u>	<u>1</u>
<u>2</u>	<u>3</u>	<u>4</u>	<u>5</u>	<u>6</u>	<u>7</u>	<u>8</u>

You Selected: Monday, July 6, 2020

## **1.10) Create a login page using required field validation control. [requiredfield.aspx]**

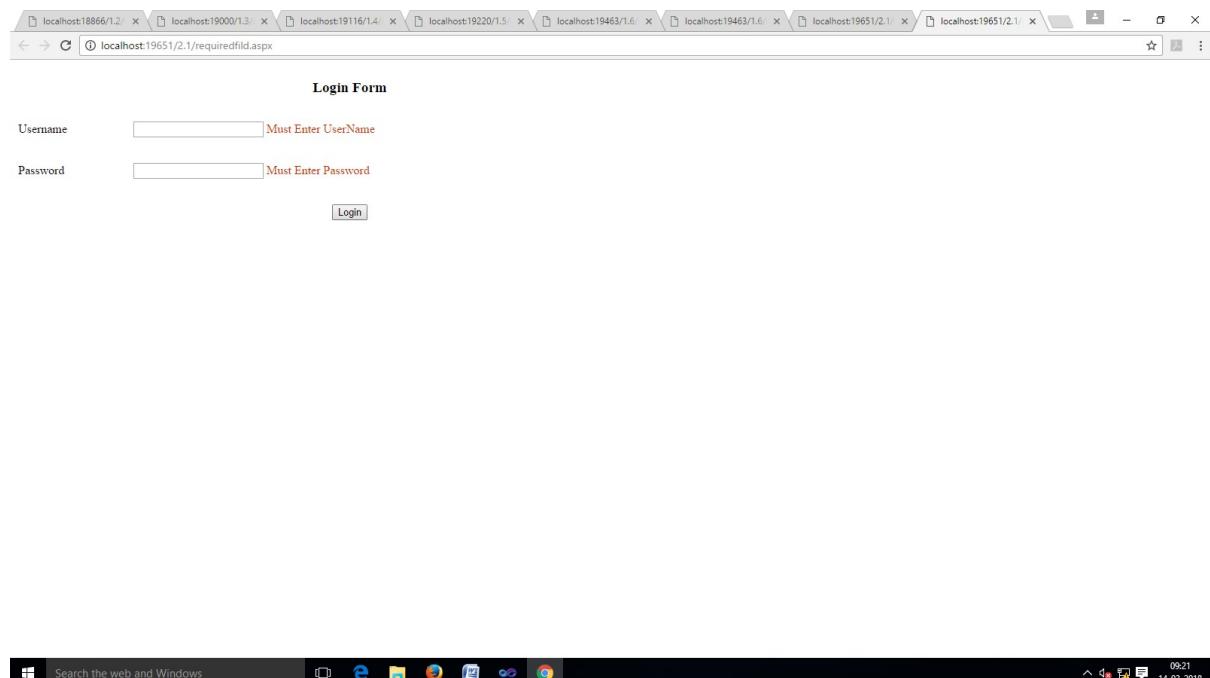
### **At Design Time:-**



- 1) **Required Field Validator**:- By this we can specify to user that this field is Compulsory. We have to set its properties.

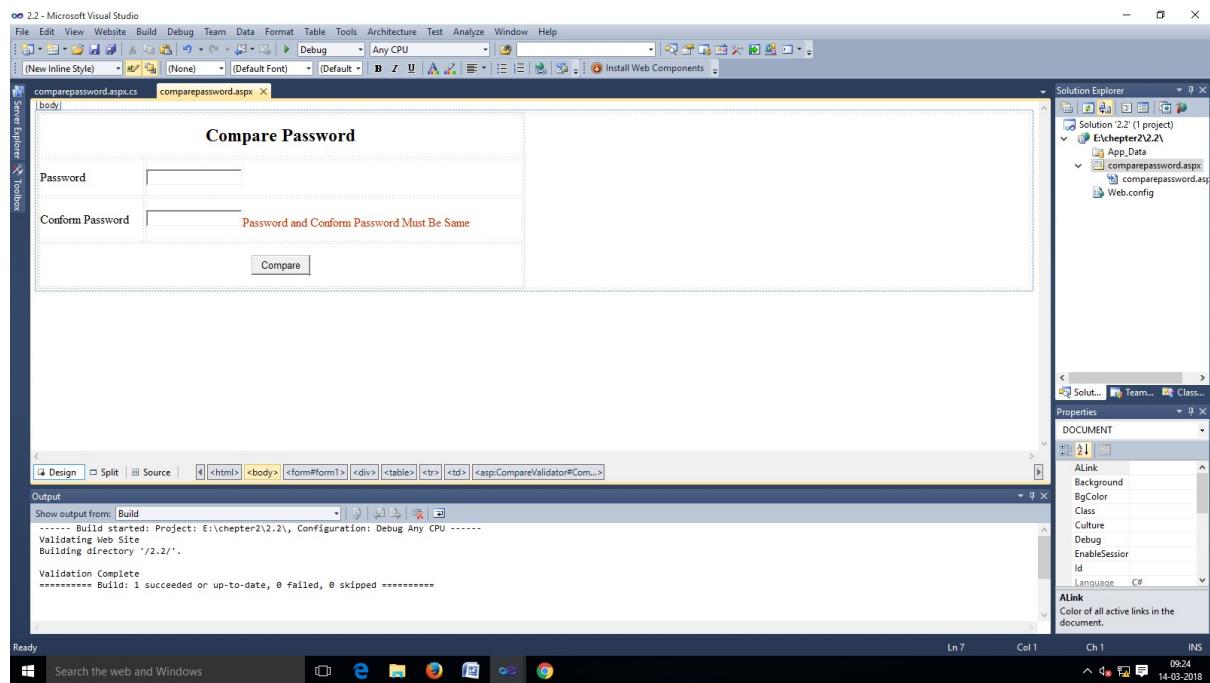
- ErrorMessage: Which string wants to generate as an error message.
- Text: It is simple text of validator. Generally there is only \* to points an error.
- ControlToValidate: On which standard control you want to validate.

### **At Run Time:-**



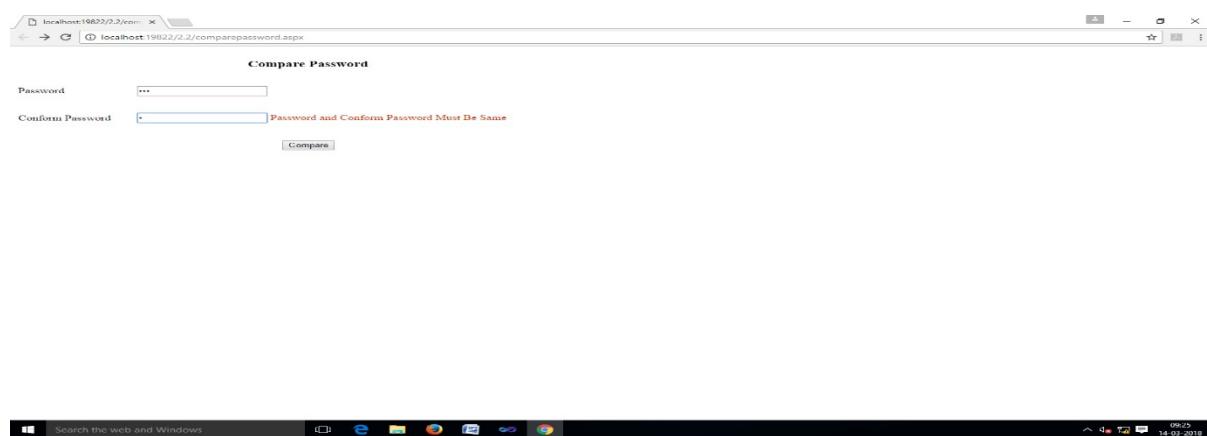
## 1.11) Create page with 2 labels and 2 textbox using compare validation control. [comparepassword.aspx]

At Design Time:-



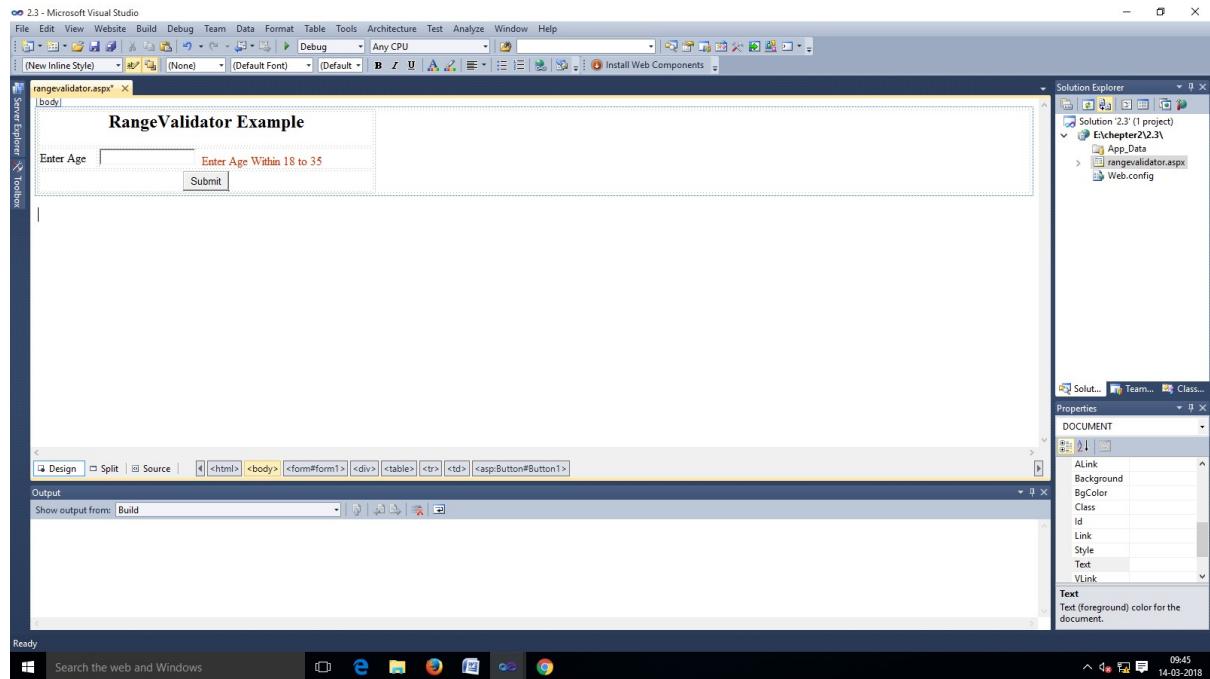
- 1) Compare Validator:- This validator compare input string to another control. If strings Are not same then it shows error. Properties are
  - ControlToCompare:- From which control u want to compare.
  - Other properties are same as Required Field validator.
  - We put this validator on Confirm Password field

At Run Time:-



## **1.12) Create page using range validation control(enter date in date textbox). [rangevalidator.aspx]**

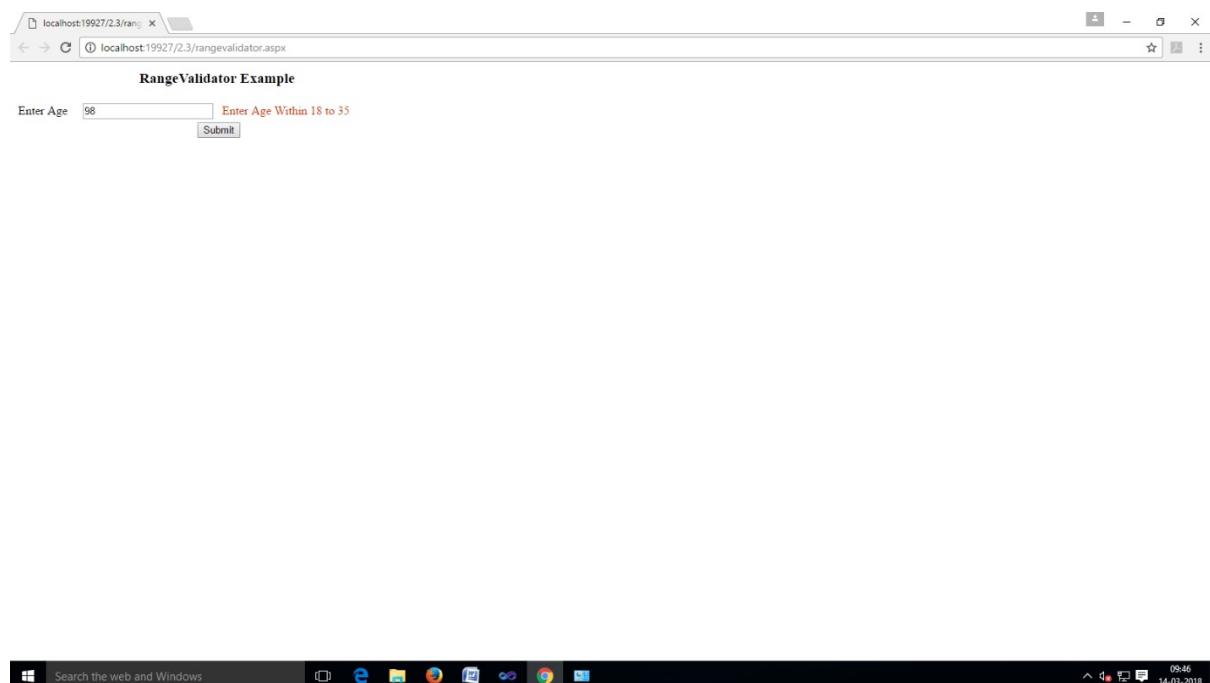
**At Design Time:-**



**Range Field Validator**:- This validator checks that whether input is in between range or Not.  
Properties are:

- Max value: - In this we have to define its max range.
- Min Value: - In this we have to define its min range.
- Other properties are same as Required Field Validator.

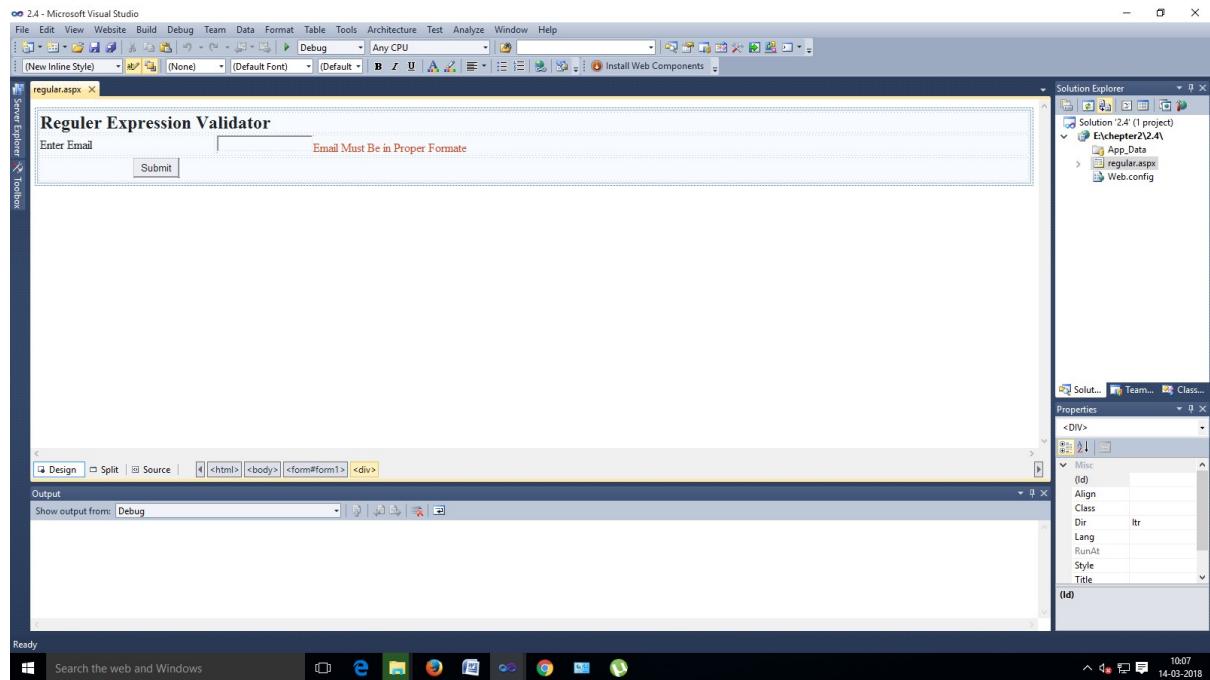
**At Run Time:-**





## **1.13) Create page using regular expression control(enter email in email textbox). [regular.aspx]**

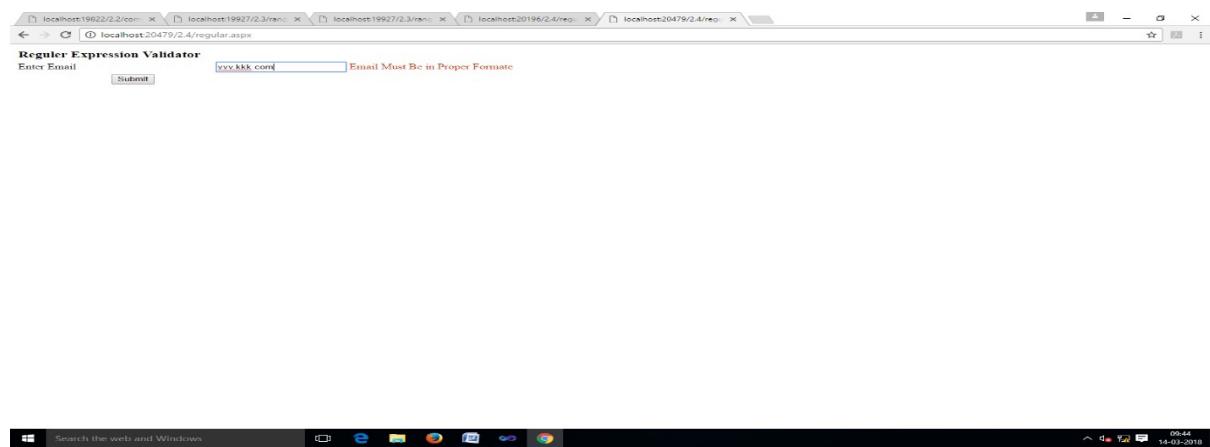
### **At Design Time:-**



**RegularExpression Validator**: This validator checks input string whether it is in Format or not. Properties are

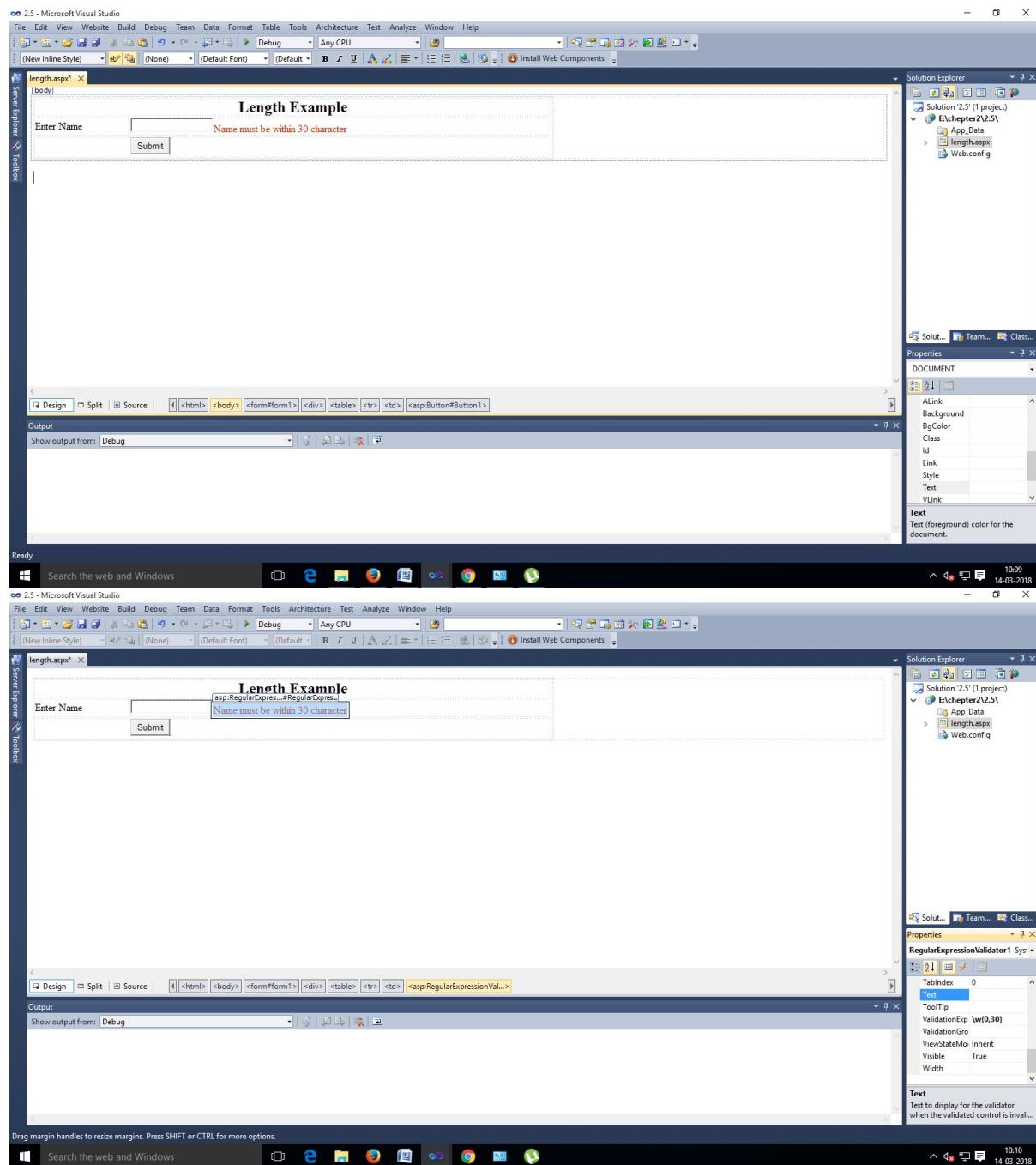
- Validation Expression: - you can see some format of strings. You can set from this property.

### **At Run Time:-**



## **1.14) Create page using regular expression control(enter email in email textbox). [length.aspx]**

### **At Design Time:-**

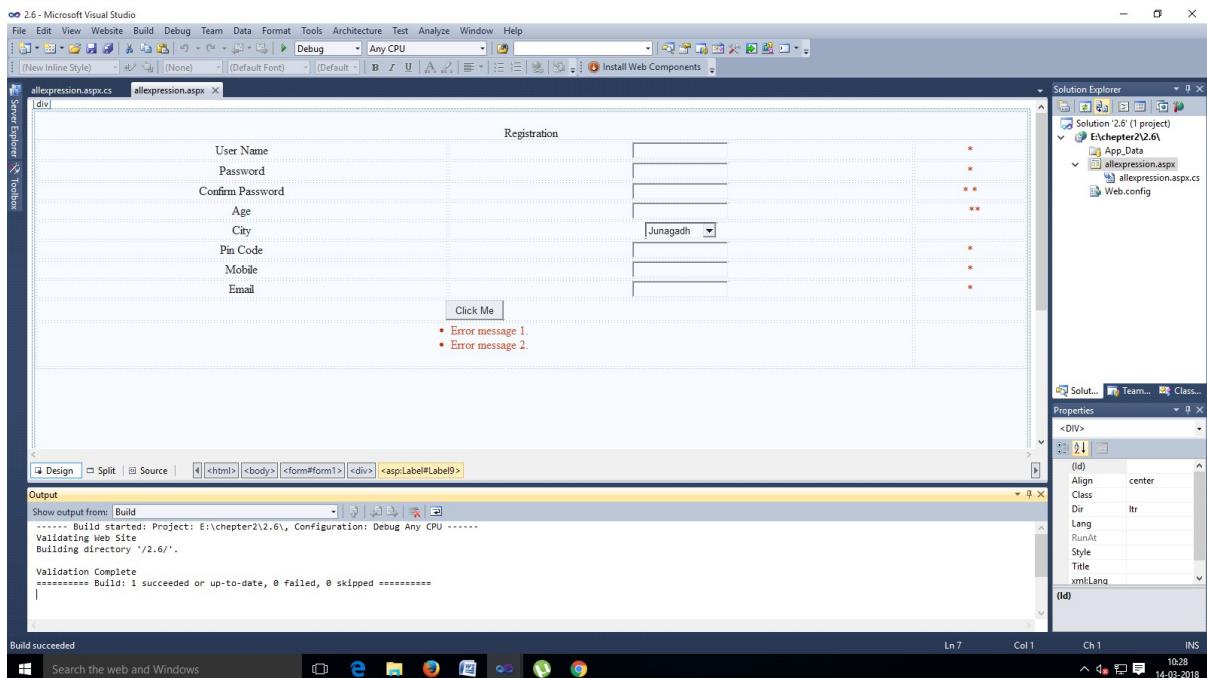


### **At Run Time:-**

## Subject :- ASP.NET



## **1.15) Create registration form using all validation control and also use validation summary control.[ allexpression.aspx]**



Here we use these validators. Required Field Validator, Range Validator, Regular Expression Validator & Validation Summary.

1) **Required Field Validator**:- By this we can specify to user that this field is Compulsory. We have to set its properties.

- ErrorMessage: Which string wants to generate as an error message.
- Text: It is simple text of validator. Generally there is only \* to points an error.
- ControlToValidate: On which standard control you want to validate.

In this application we put validators on UserName, Password, Confirm Password, & on Age field.

2) **Range Field Validator**:- This validator checks that whether input is in between range or Not. Properties are:

- Max value: - In this we have to define its max range.
- Min Value: - In this we have to define its min range.
- Other properties are same as Required Field Validator.

In this application we put this validator on Password & Age.

3) **Compare Validator**:- This validator compare input string to another control. If strings Are not same then it shows error. Properties are

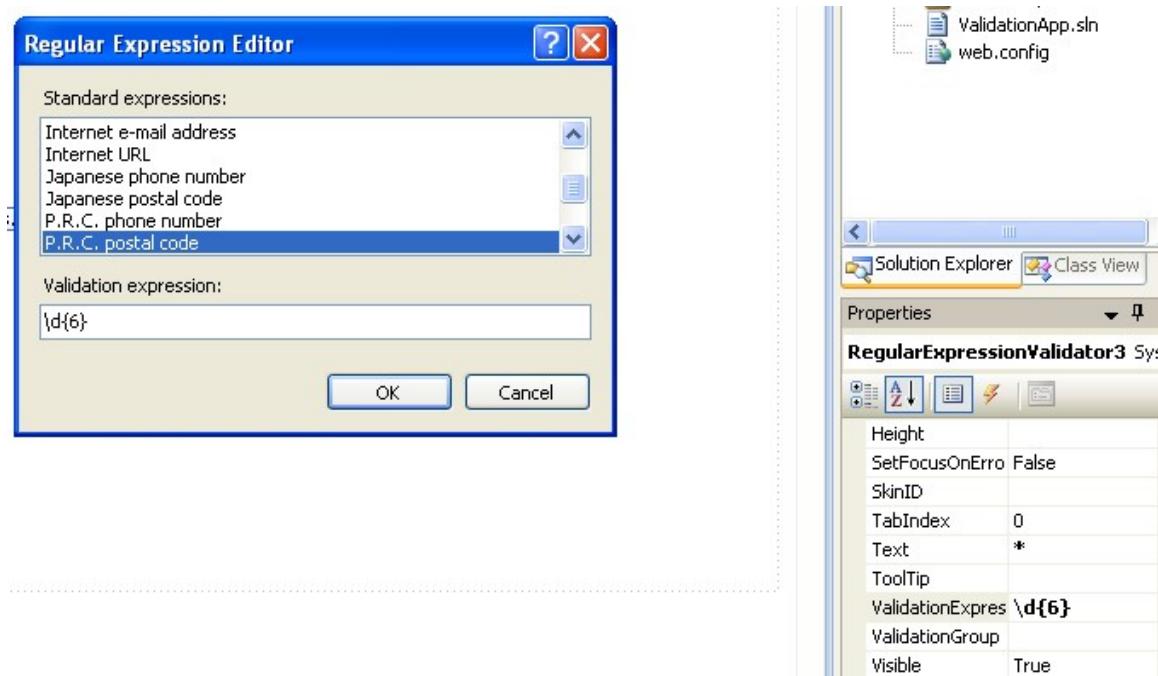
-ControlToCompare:- From which control u want to compare.

Other properties are same as Required Field validator.

We put this validator on Confirm Password field

4) RegularExpression Validator:- This validator checks input string whether it is in Format or not. Properties are

- Validation Expression: - you can see some format of strings. You can set from this property.



This is set on Pincode, Email & MobileNo field.

5) Validation Summary: - This is the control you have no need to set properties on this. By this control Error Messages will show.

**At Run Time:-**

1)



Registration

User Name

Password

Confirm Password

Age

City

Pin Code

Mobile

Email

Click Me

2)



Registration

User Name

Password

Confirm Password

Age

City

Pin Code

Mobile

Email

Click Me

\*

\*

\*

\*

•

•

•

User Name is Required

Password is Required

Confirm Password is required

Age is Required

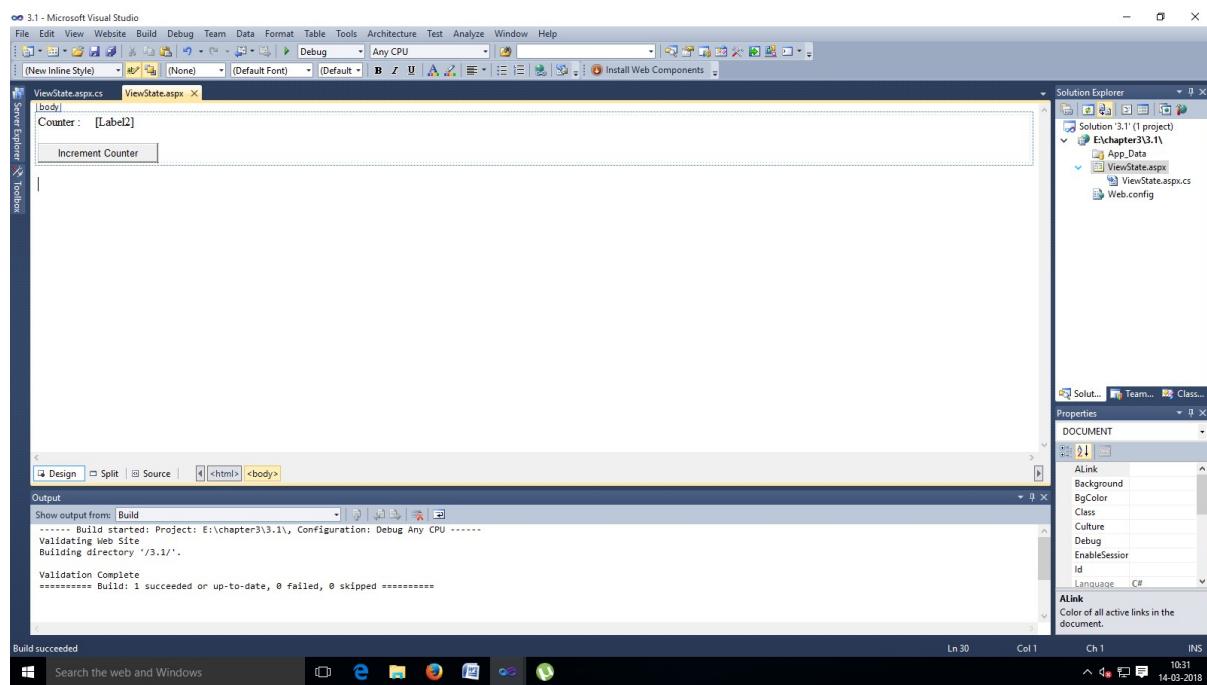


# Chapter-2

# State Management

**2.1) Application for the use of ViewState:**

- At Design Time:



For viewstate.Aspx.cs page:-

```
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 btnIncrCounter_Click(object sender, EventArgs e)
    {
        int c;
        if (ViewState["Counter"] == null)
        {
            c = 1;
            ViewState["Counter"] = c;
        }
        else
        {
            c = Convert.ToInt32(ViewState["Counter"]) + 1;
            ViewState["Counter"] = c;
        }
        Label2.Text = c.ToString();
    }
}
```

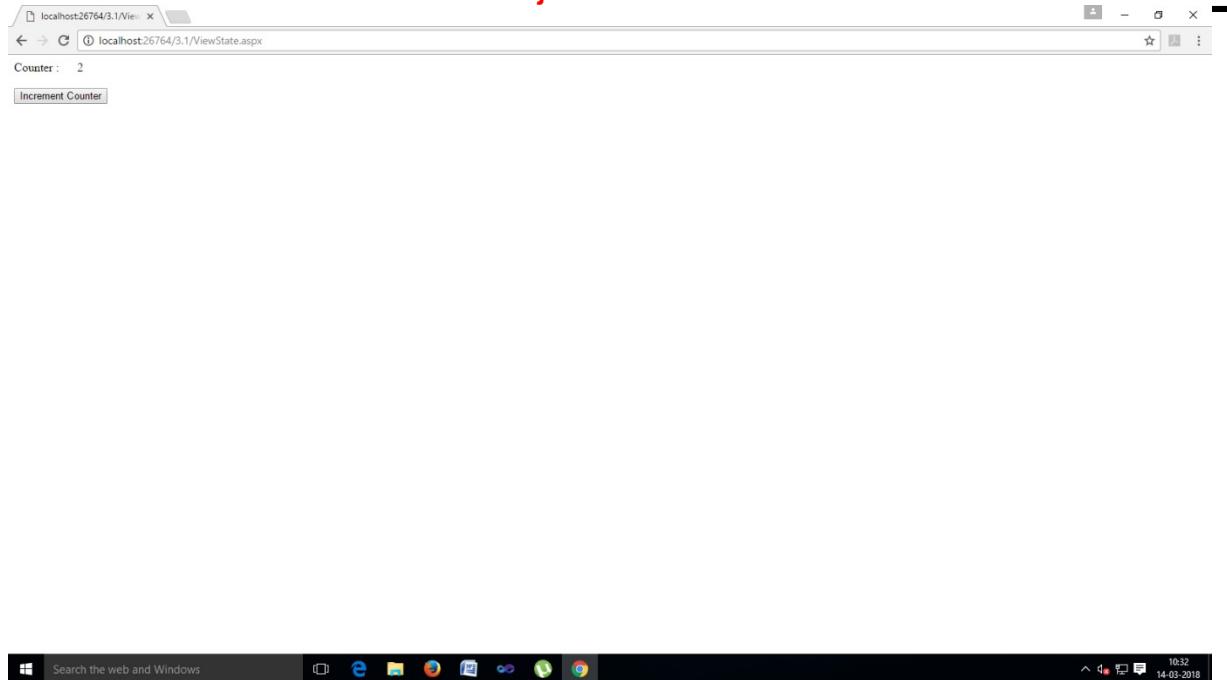
At Run Time:-



**After Clicking Button**



## Subject :- ASP.NET



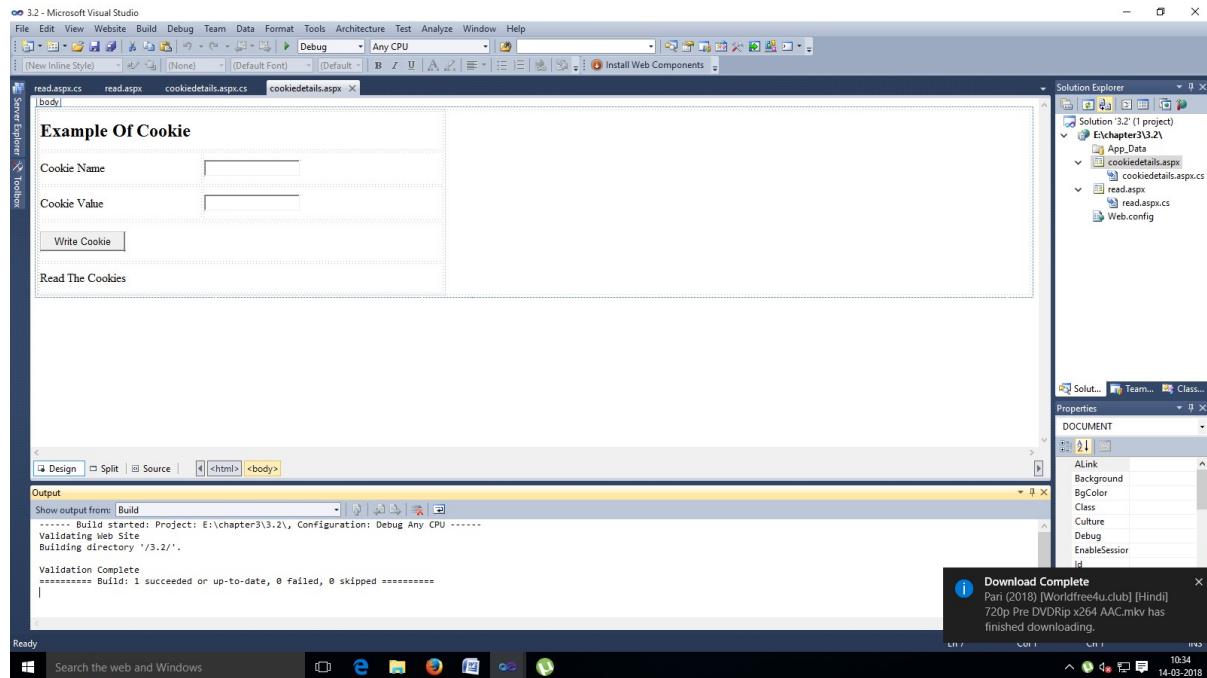
The counter increases as you clicking the button.

## 2.2) Application for the use of Cookies:-

For this take two forms 1) cookiedetails.aspx 2) read.aspx

At Design Time:

### Cookiedetail.aspx



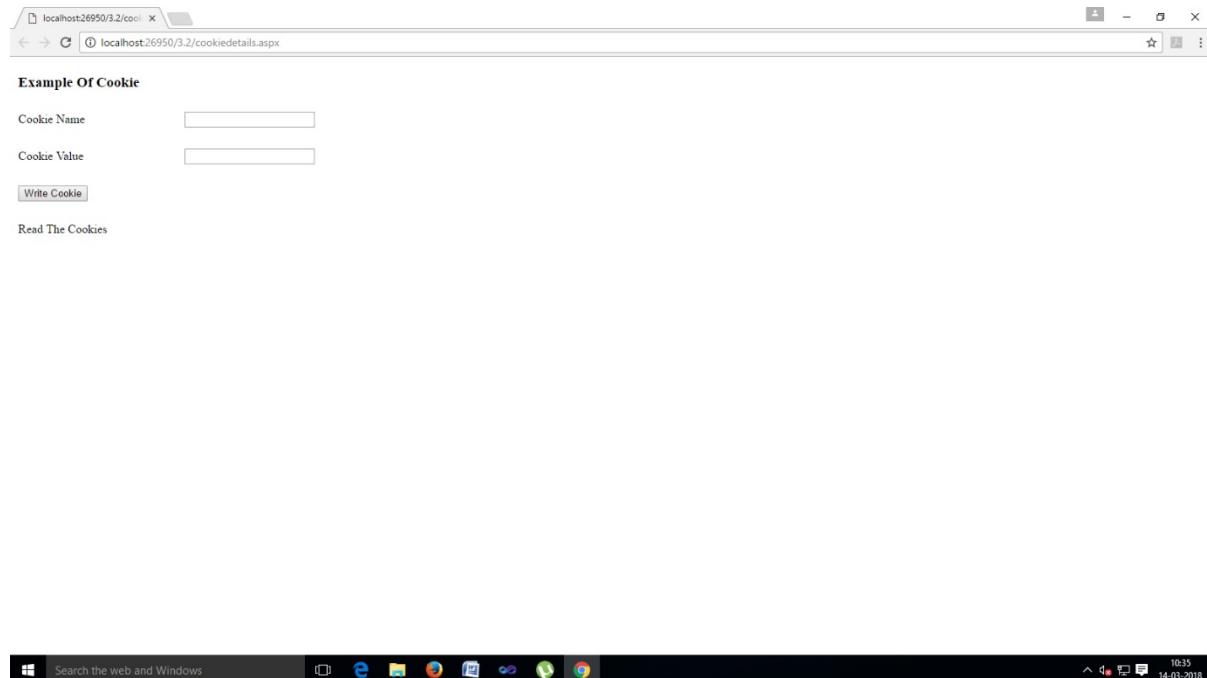
For cookiedetails.aspx.cs page:-

```
using System;
using System.Collections;
using System.Configuration;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.HtmlControls;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Xml.Linq;

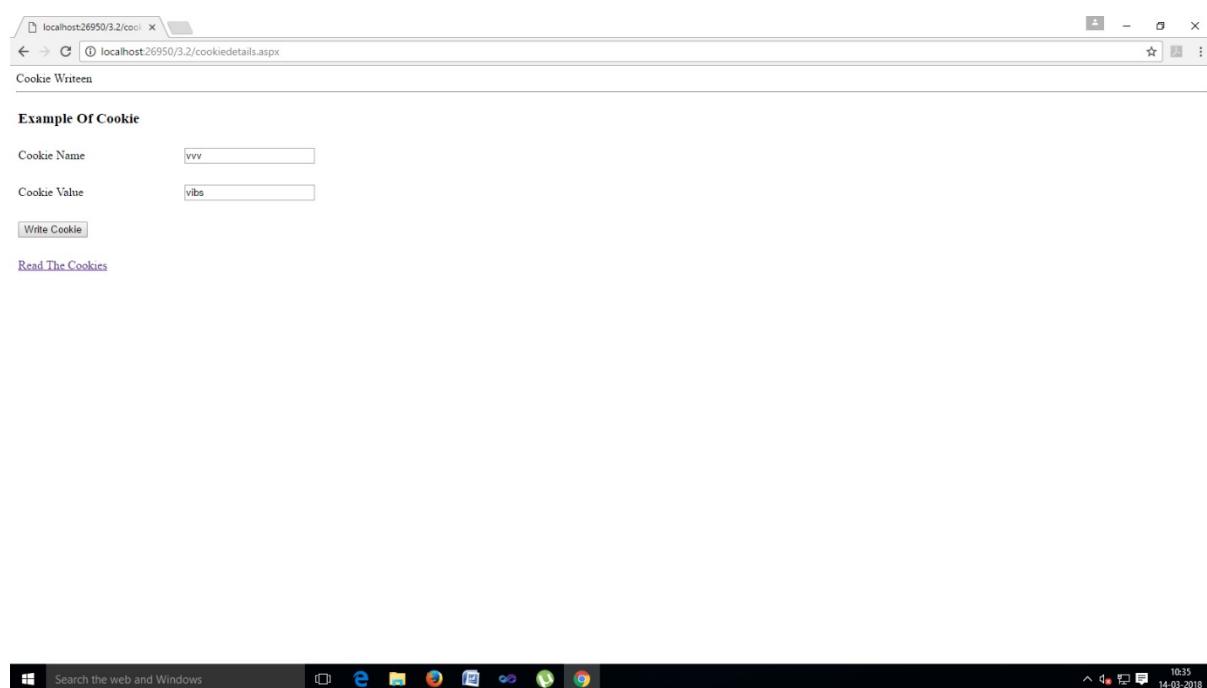
public partial class cookiedetails : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        if (IsPostBack)
        {
            lnkread.Visible = true;
            lnkread.NavigateUrl = "read.aspx?cookie=" + txtcname.Text.ToString();
        }
    }
    protected void btnwrite_Click(object sender, EventArgs e)
    {
        HttpCookie cookie = new HttpCookie(txtcname.Text);
        cookie.Value = txtcvalue.Text;
        DateTime dtnow = DateTime.Now;
    }
}
```

```
TimeSpan tsmiunte = new TimeSpan(0, 0, 1, 0);
cookie.Expires = dtnow + tsmiunte;
Response.Cookies.Add(cookie);
Response.Write("Cookie Written <br><hr>");
}
}
```

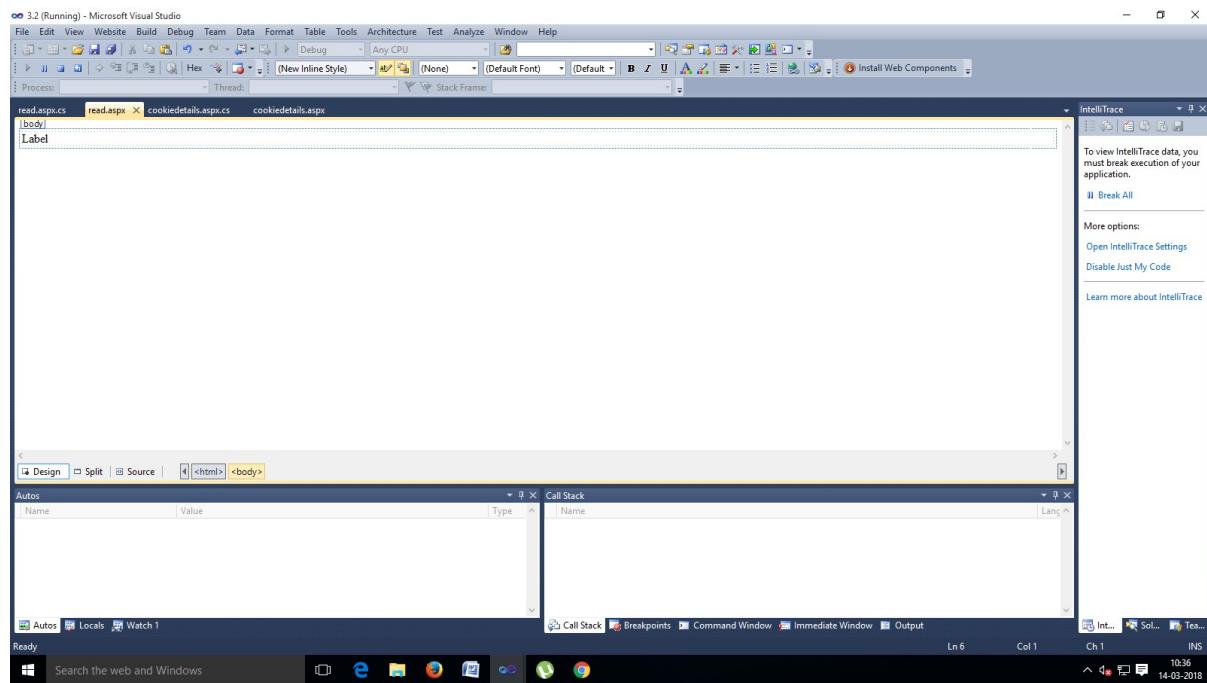
### **At Run Time:-**



### **After Clicking Write Cookies button.**



## Read.aspx



## Read.aspx.cs

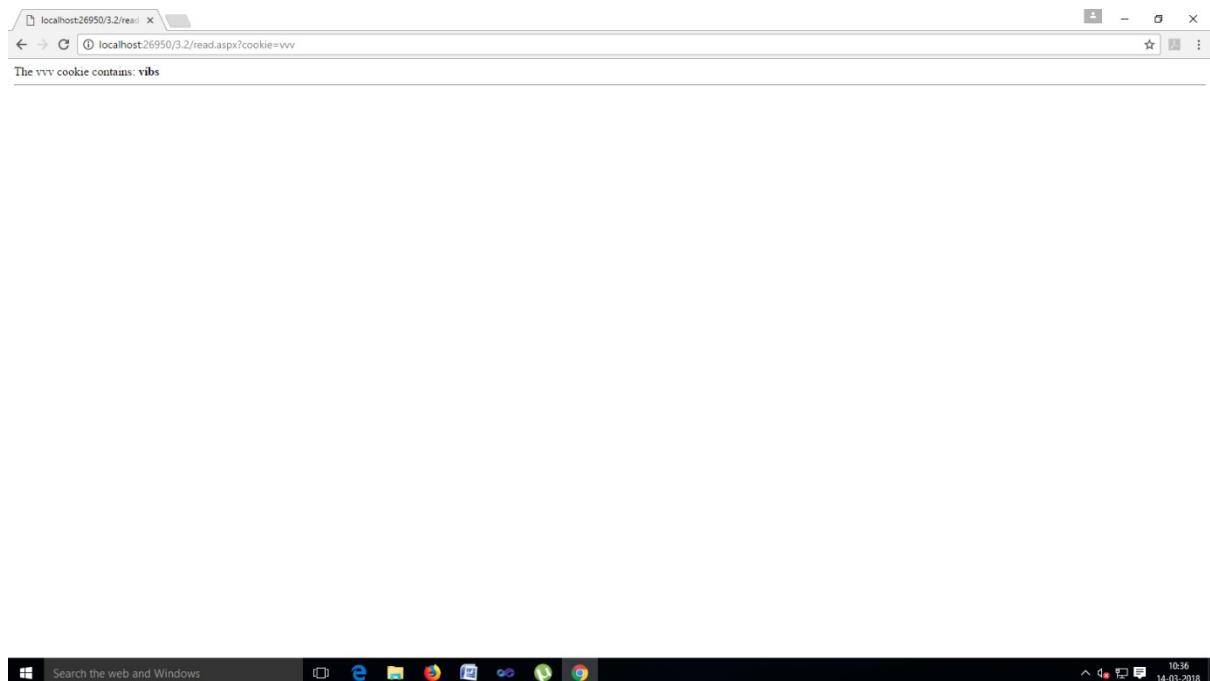
```
using System;
using System.Collections;
using System.Configuration;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.HtmlControls;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Xml.Linq;

public partial class read : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        if (Request.QueryString["cookie"] != null)
        {
            String strcookiename = Request.QueryString["cookie"].ToString();

            HttpCookie cookie = Request.Cookies[strcookiename];
            if (cookie == null)
            {
                lblcookie.Text = "Cookie not found <br><hr>";
            }
            else
            {
                String strcookievalue = cookie.Value.ToString();
                lblcookie.Text = "The " + strcookiename + " cookie contains: <b>" + strcookievalue +
                "</b><br><hr>";
            }
        }
    }
}
```

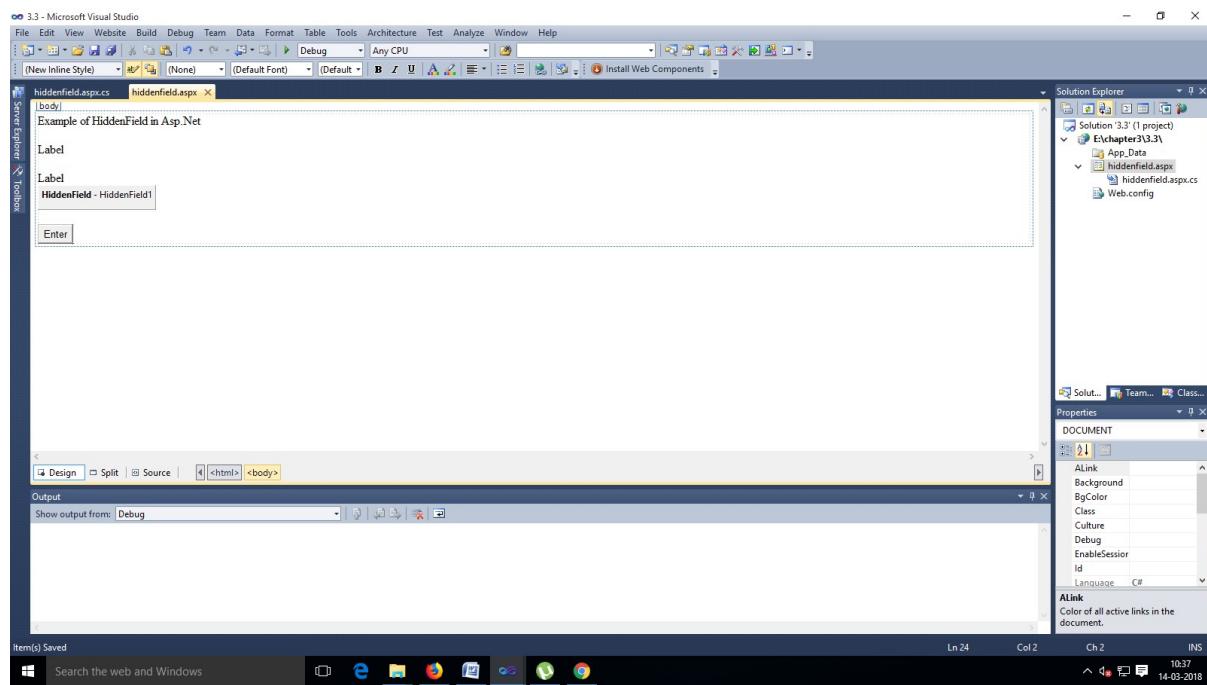
}

**At Run Time:-**



## 2.3) WAP to get value of one page to another page using Hidden Fields

**At Design Time:**



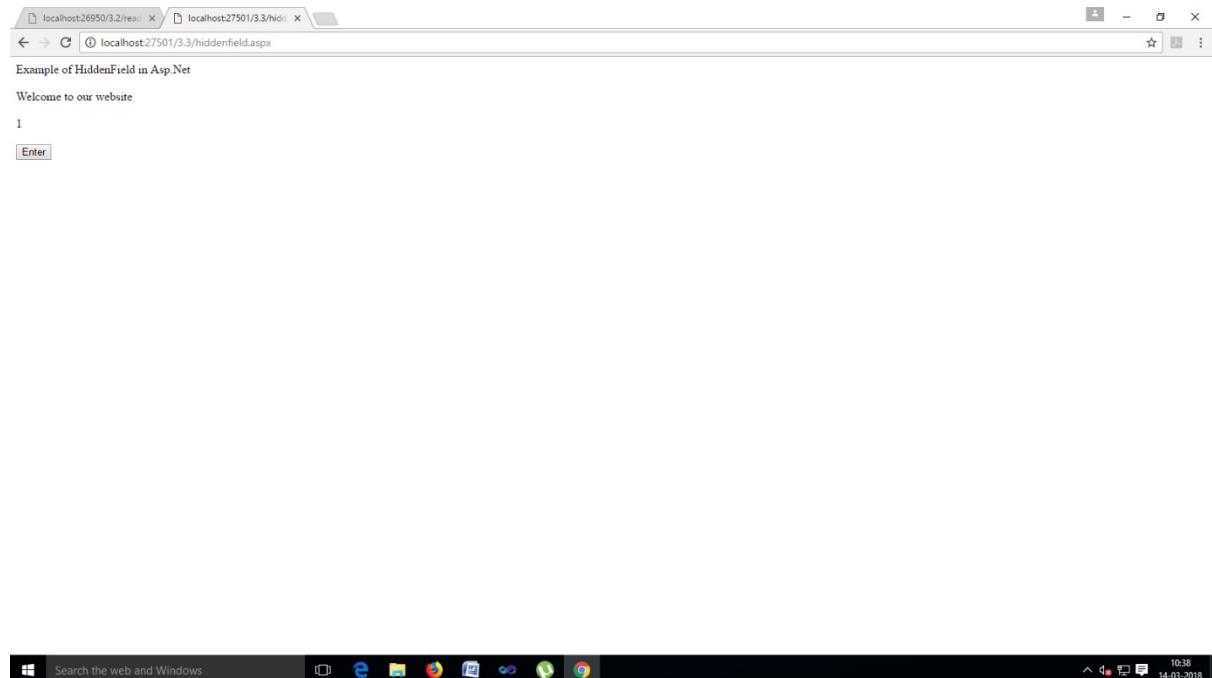
### Hiddenfiled.aspx.cs

```
using System;
using System.Configuration;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.HtmlControls;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Xml.Linq;

public partial class _Default : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        Page.EnableViewState = false;
        HiddenField1.Value = "Welcome to our website";
        Label1.Text = HiddenField1.Value;
    }
    protected void btnentr_Click(object sender, EventArgs e)
    {
        HiddenField1.Value = "0";
        int i = 0;
        i = (int.Parse(HiddenField1.Value) + 1);
        Label2.Text = i.ToString();
        HiddenField1.Value = i.ToString();
    }
}
```

---

## **At Run Time**

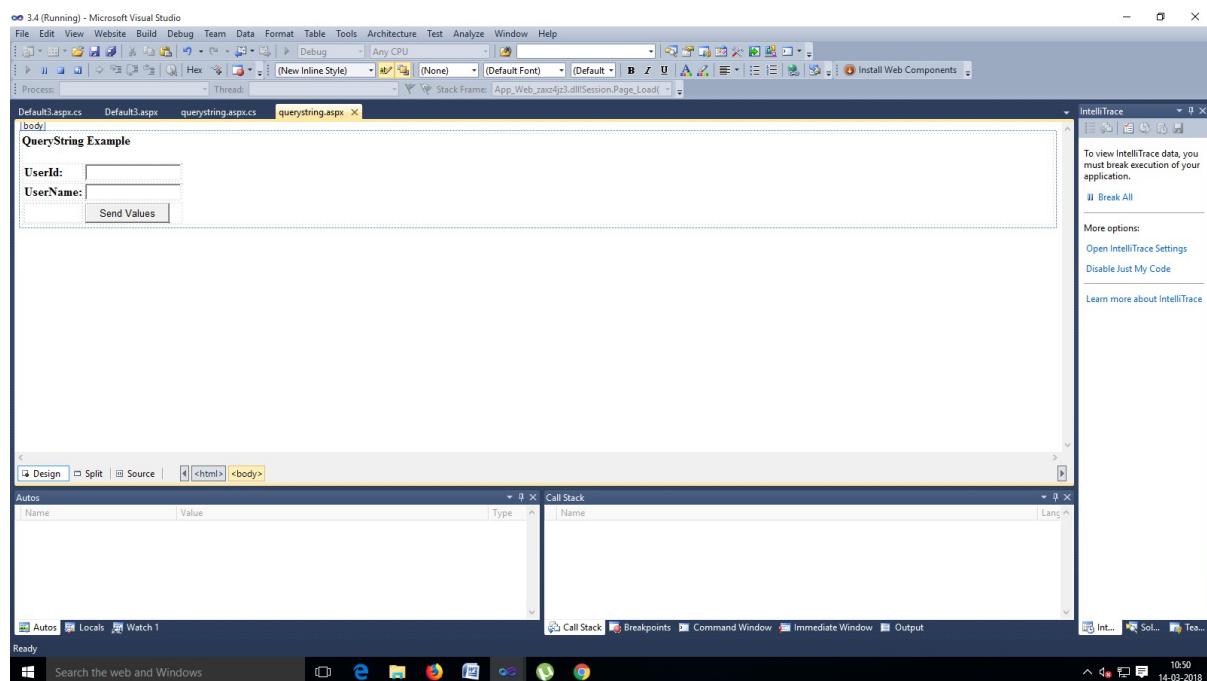


## 2.4) WAP to get value of one page to another page using Query String

For this take two forms 1) default3.aspx 2) querystring.aspx

At Design Time:

### Querystring.aspx



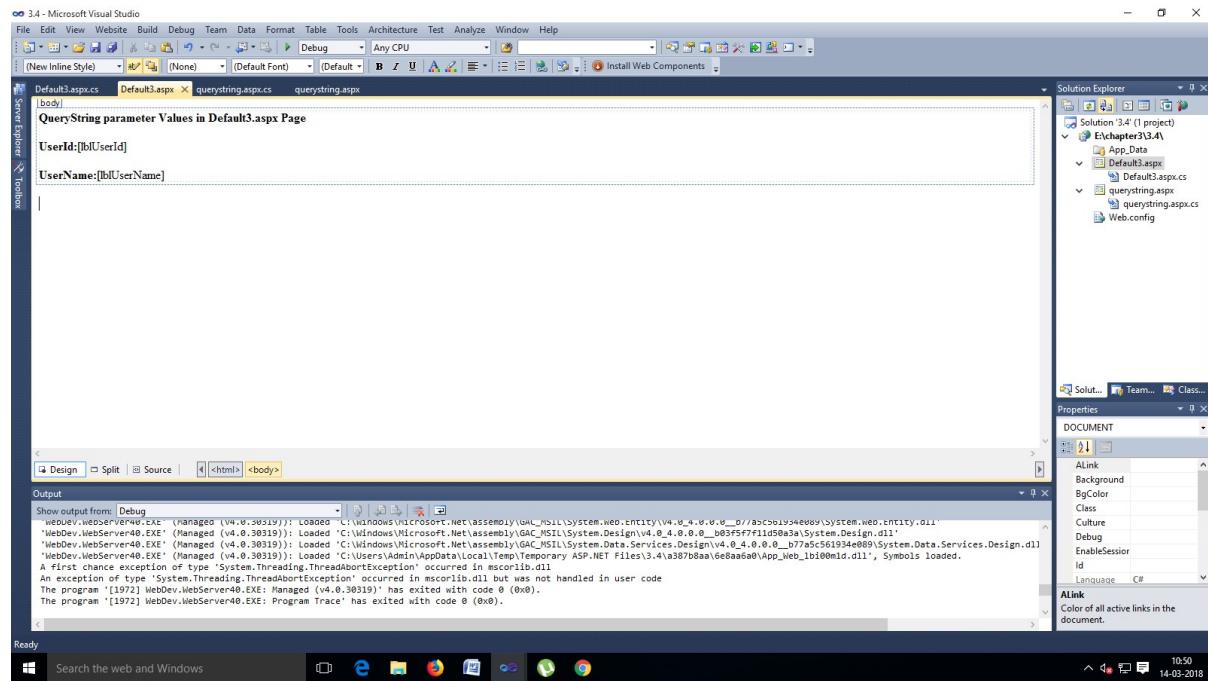
### Querystring.aspx.cs

```
using System;
using System.Collections;
using System.Configuration;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.HtmlControls;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Xml.Linq;

public partial class Default2 : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
    }

    protected void btnSend_Click(object sender, EventArgs e)
    {
        Response.Redirect("Default3.aspx?UserId=" + txtUserId.Text + "&UserName=" + txtUserName.Text);
    }
}
```

## Default3.aspx

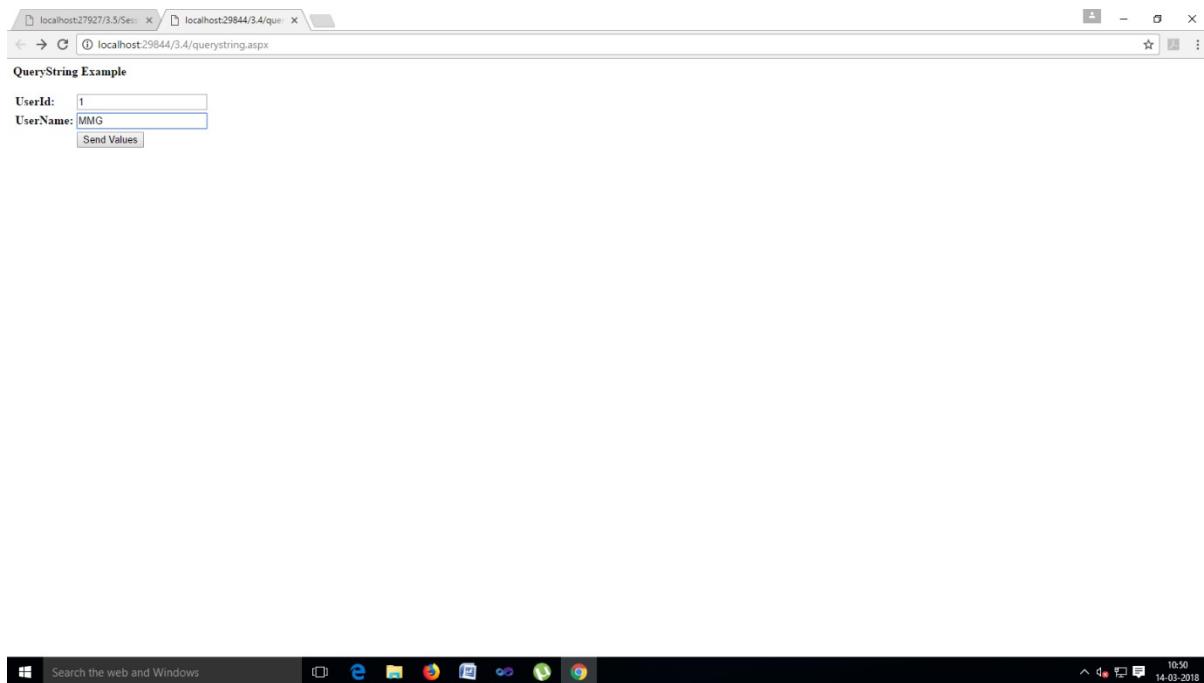


## Default3.aspx.cs

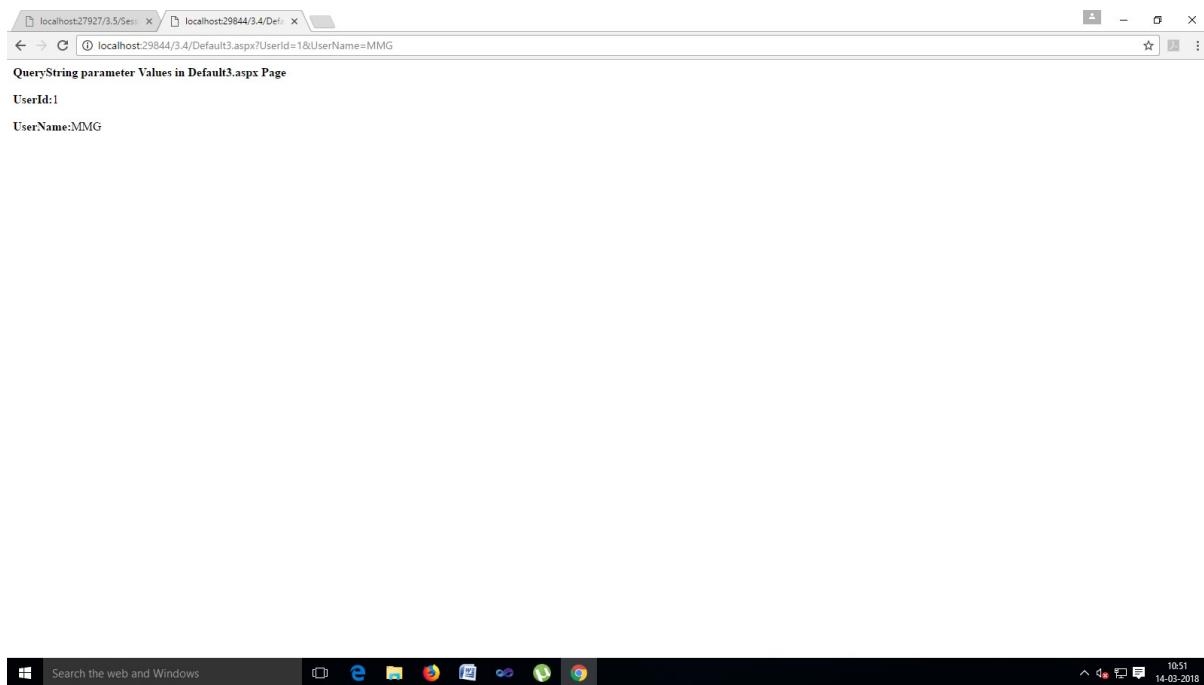
```
using System;
using System.Collections;
using System.Configuration;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.HtmlControls;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Xml.Linq;

public partial class Default3 : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        if (!IsPostBack)
        {
            lblUserId.Text = Request.QueryString["UserId"];
            lblUserName.Text = Request.QueryString["UserName"];
        }
    }
}
```

## **At Run Time**



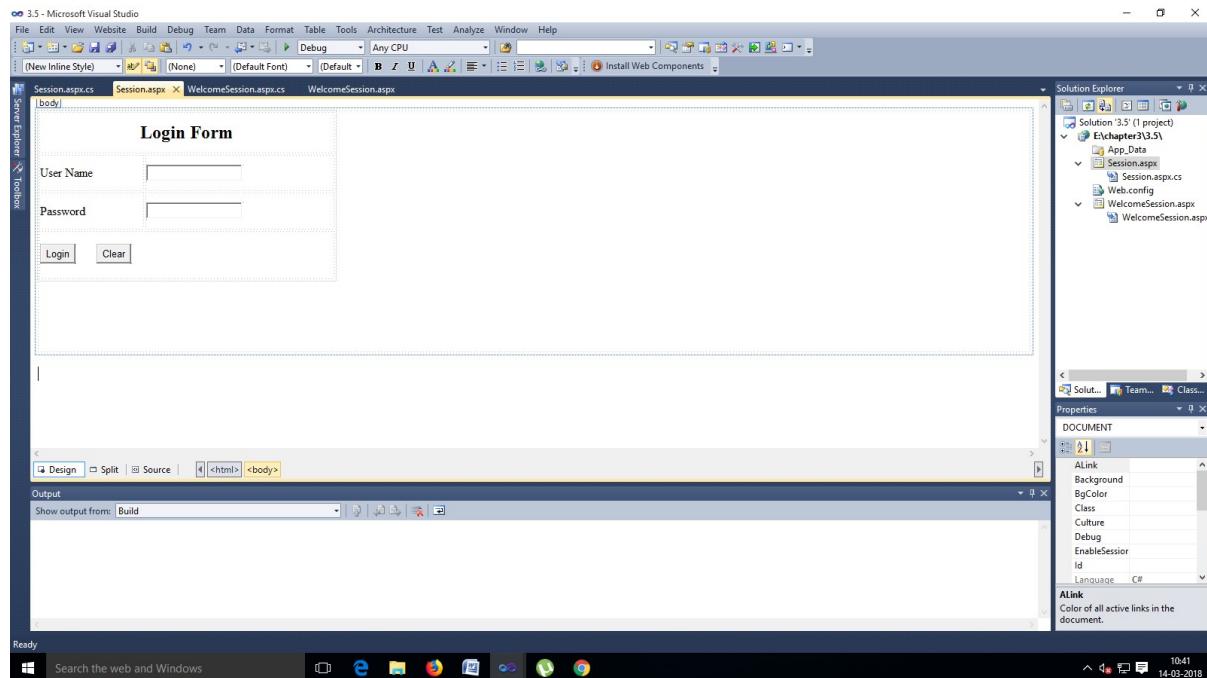
**After clicking on SendValue Button**



## 2.5) Create login page using session state

For this application we take two forms. 1) Session.aspx 2) WelcomeSession.aspx

### Session



### For session.Aspx.cs page:-

```
using System;
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)
    {

    }

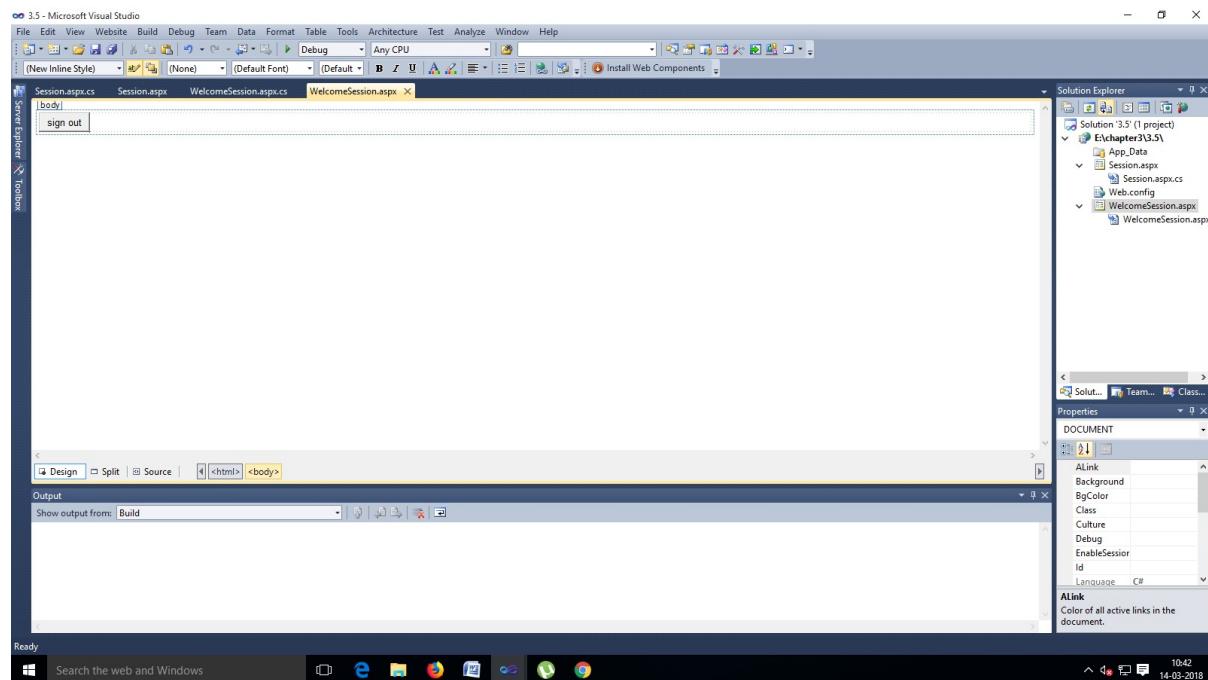
    protected void btnclear_Click(object sender, EventArgs e)
    {
        TextBox1.Text = "";
        TextBox2.Text = "";
        TextBox1.Focus();
    }

    protected void btnlogin_Click(object sender, EventArgs e)
    {
        if (TextBox1.Text == "admin" && TextBox2.Text == "admin")
        {
            Session["User"] = TextBox1.Text;
            Response.Redirect("WelcomeSession.aspx");
        }
        else
        {
            Response.Redirect("WelcomeSession.aspx");
        }
    }
}
```

```
}
```

### Design of welcomesession.aspx page:-

There is only one sign out button of aspx page.



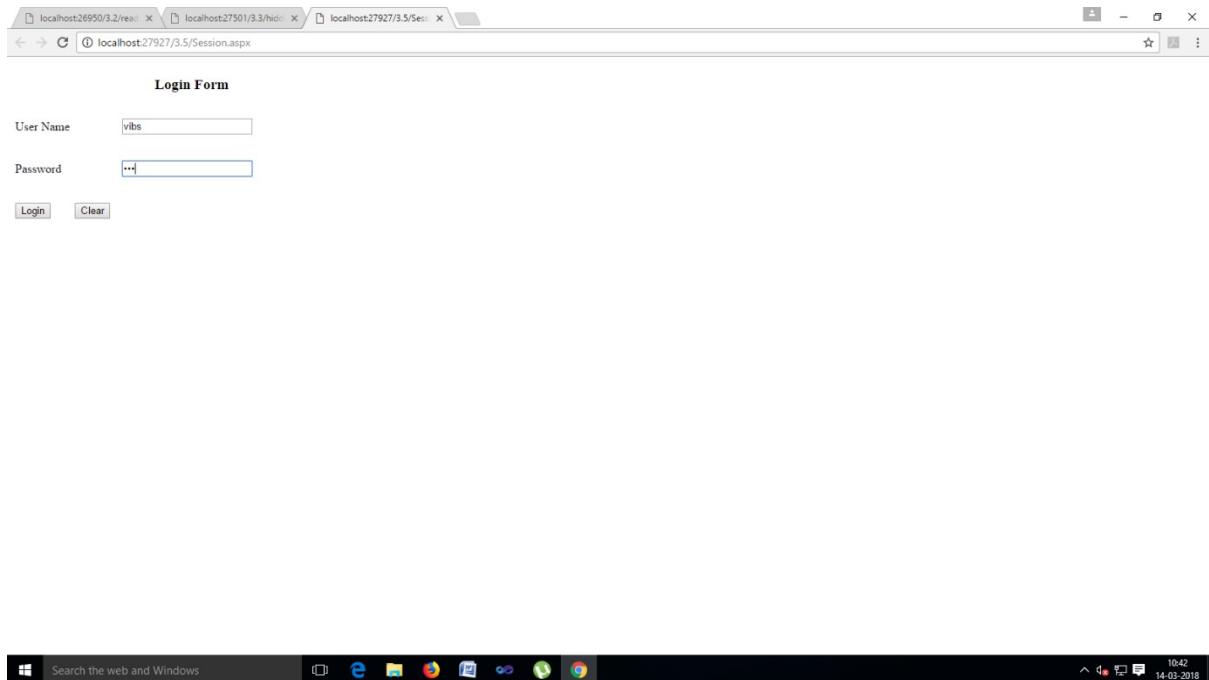
### For welcomesession.aspx.cs page

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

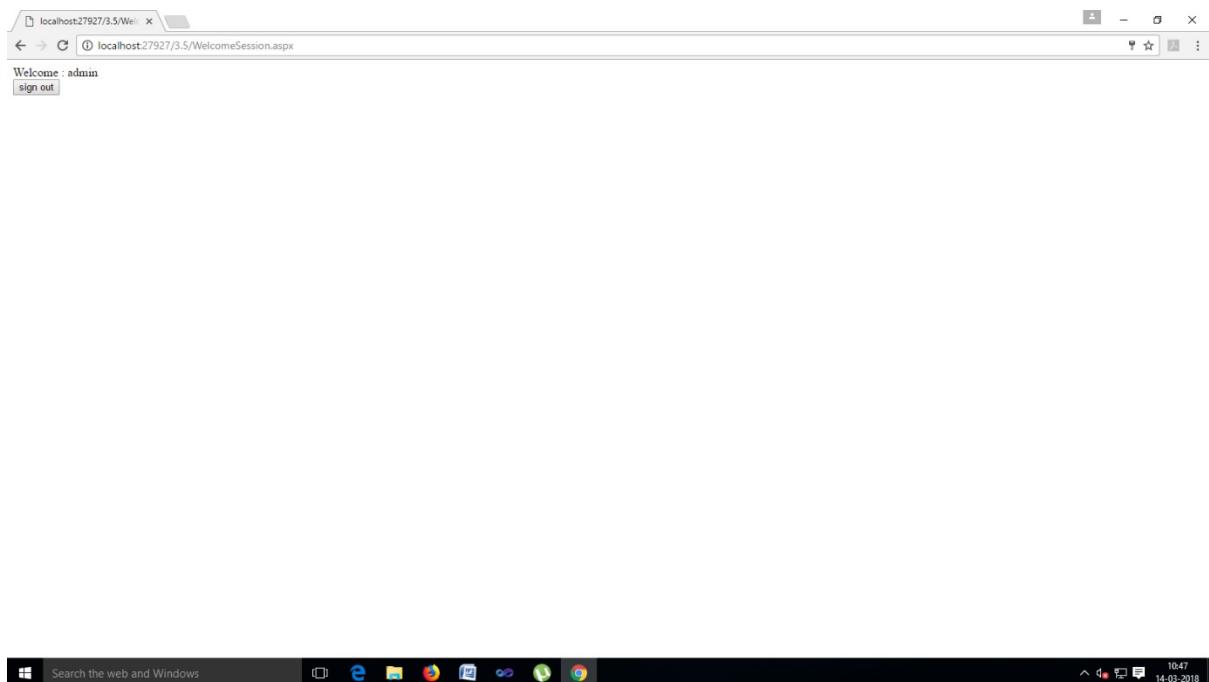
public partial class WelcomeSession : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        if (Session["User"] == null)
        {
            Response.Write("Welcome Guest");
        }
        else
        {
            Response.Write("Welcome : " + Session["User"].ToString());
        }
    }

    protected void Button1_Click(object sender, EventArgs e)
    {
        Session["User"] = null;
        Response.Redirect("~/Session.aspx");
    }
}
```

**At Run Time:-**

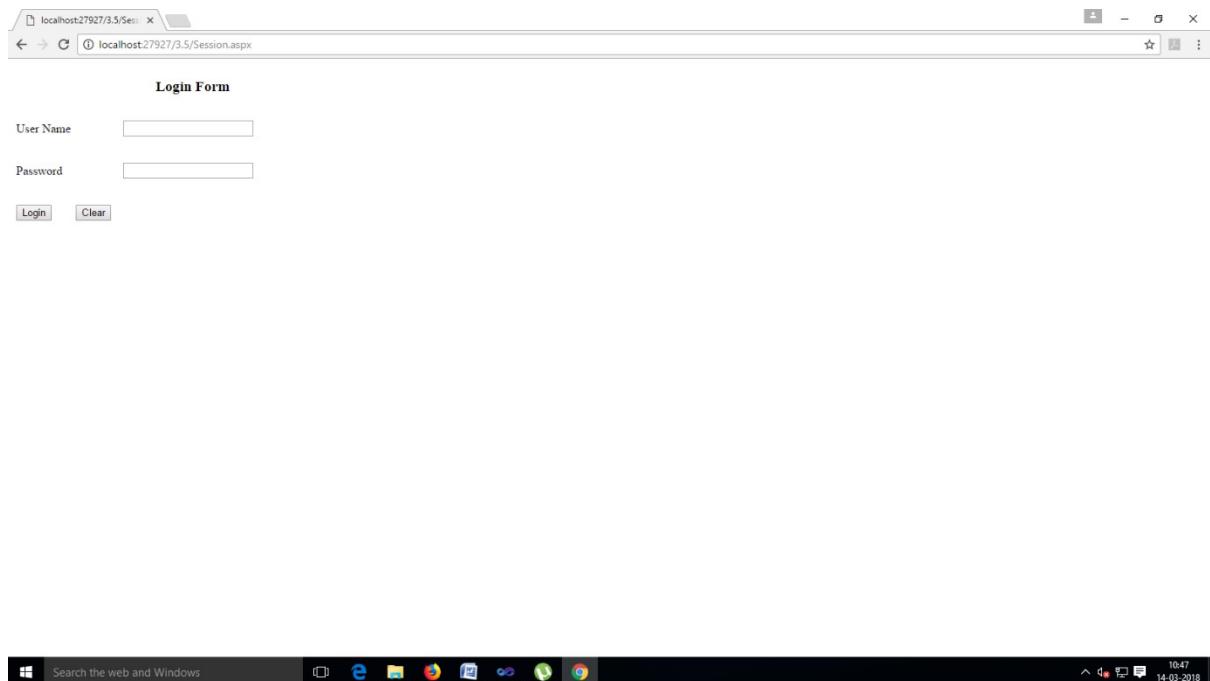


**After clicking Login button**



**It will be redirected to Session page**

**After clicking sign out button**



## **2.6) WAP to get visitors count using application state**

- **At Design Time:**

Take Global.asax file in your project & below there is code for it.

```
<%@ApplicationLanguage="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
    int counter = Convert.ToInt32(Application["visitor"]) + 1;
    Application["visitor"] = counter;
}

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.
}

</script>
```

**Take another form named VisitorCounterExample.aspx& put one label in it.**

### **Visitorcounter.aspx.cs Page**

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

public partial class VisitorCounterExample : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        Label1.Text = "Visitor No: " + Application["visitor"].ToString();
    }
}
```

### **At Run Time:-**



**When you start this application again then o/p will be**



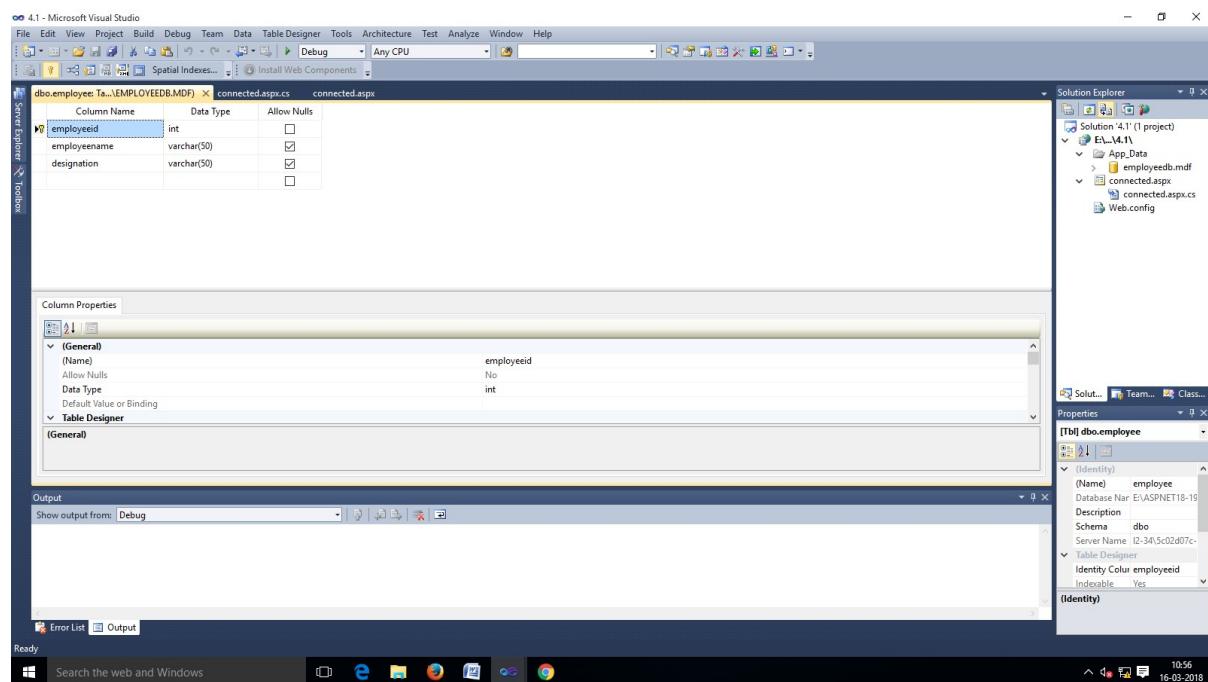
# Chapter-3

# ADO.NET And

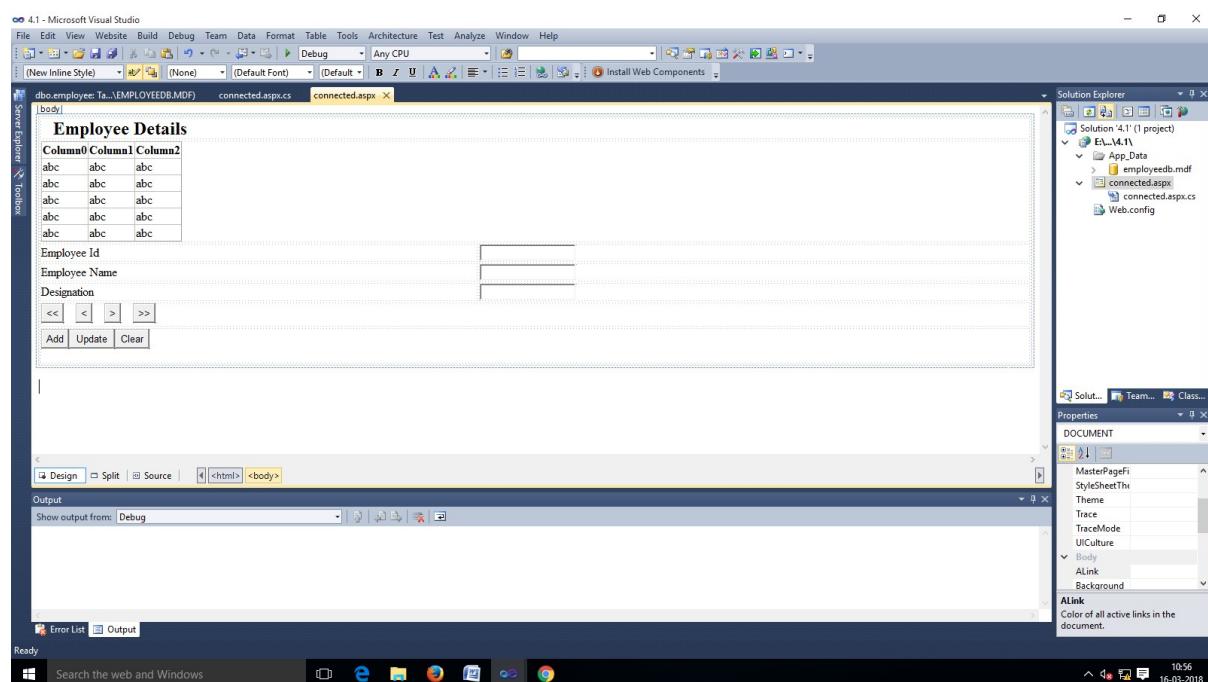
# Database

**3.1)Connected architecture example for inserting record into employee table (EmpId, Empname, Designation) and display in gridview and also edit record.**

## Table



## At Design Time:-



## Connected.aspx.cs

```
using System;
using System.Collections;
using System.Configuration;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.HtmlControls;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Xml.Linq;
using System.Data.SqlClient;
using System.Web.Configuration;

public partial class connected : System.Web.UI.Page
{
    string constring;
    SqlConnection connect;
    SqlCommand cmd;
    StaticSqlDataAdapter da;
    StaticDataTable dt = new DataTable();
    DataSet ds = new DataSet();
    Staticint recptr = 0;
    void showdata()
    {
        dt.Clear();
        da = new SqlDataAdapter("select * from employee", connect);
        da.Fill(dt);
        txtid.Text = dt.Rows[recptr][0].ToString();
        txtname.Text = dt.Rows[recptr][1].ToString();
        txtdesignation.Text = dt.Rows[recptr][2].ToString();

        GridView1.DataSource = dt;
        GridView1.DataBind();
    }
    void refreshdata()
    {
        ds.Clear();
        da = new SqlDataAdapter("select*from employee", connect);
        da.Fill(ds);

    }
    protected void Page_Load(object sender, EventArgs e)
    {
        constring = WebConfigurationManager.ConnectionStrings["constring"].ConnectionString;
        connect = new SqlConnection(constring);
        if (!IsPostBack)
        {
            refreshdata();
            showdata();
        }
        connect.Open();
    }

    protected void btnfirst_Click(object sender, EventArgs e)
    {
        recptr = 0;
        showdata();
    }
    protected void btnprevious_Click(object sender, EventArgs e)
    {
        if (recptr > 0)
        {
```

```
    reciptr--;
    showdata();
}
}
protectedvoid btnnext_Click(object sender, EventArgs e)
{
if(recptr < dt.Rows.Count - 1)
{
    reciptr++;
    showdata();
}
}
protectedvoid btnlast_Click(object sender, EventArgs e)
{
    reciptr = dt.Rows.Count - 1;
    showdata();
}

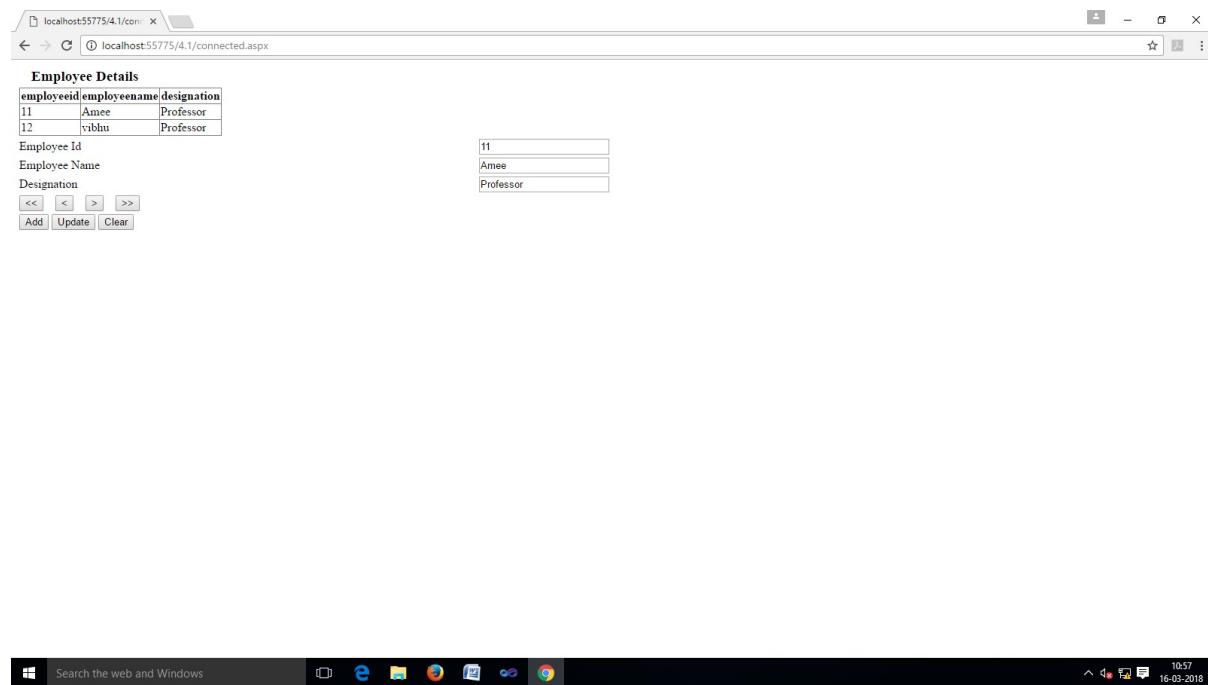
protectedvoid btnclear_Click(object sender, EventArgs e)
{
    txtid.Text = "";
    txtname.Text = "";
    txtdesignation.Text = "";
}
protectedvoid btnadd_Click(object sender, EventArgs e)
{
    string query = "insert into employee values('" + txtname.Text + "','" + txtdesignation.Text + "')";
    cmd = new SqlCommand(query, connect);
    int rows = cmd.ExecuteNonQuery();

    refreshdata();
    showdata();
    GridView1.DataBind();
}

protectedvoid btnupdate_Click(object sender, EventArgs e)
{
    cmd = new SqlCommand("update employee set employeeename='" + txtname.Text + " , designation='" +
    txtdesignation.Text + "' where employeeid=" + txtid.Text, connect);
    cmd.ExecuteNonQuery();
    refreshdata();
    showdata();
    GridView1.DataBind();
}
}
```

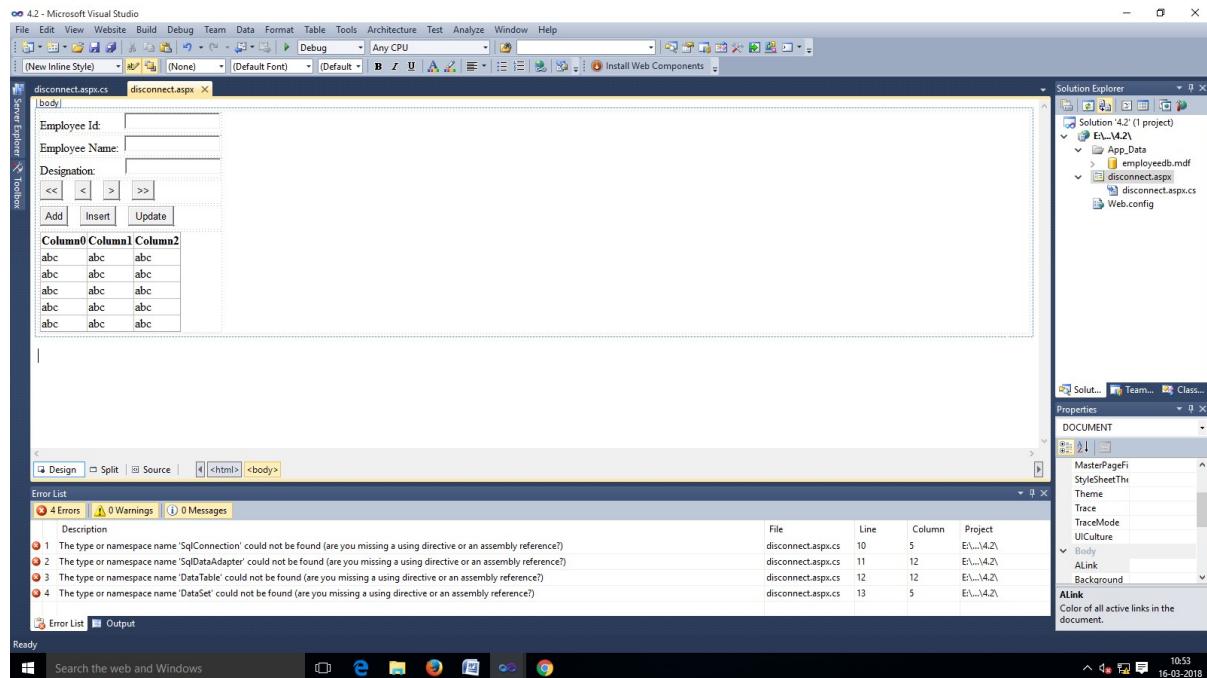
---

## **At Run Time**



### 3.2) Dis-Connected architecture example for inserting record into employee table (EmpId, Empname, Designation) and also display in gridview and also edit record and next, previous, first, and last navigation.

#### At Design Time:



#### Disconnect.aspx.cs

```

using System;
using System.Collections;
using System.Configuration;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.HtmlControls;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Xml.Linq;
using System.Data.SqlClient;

public partial class disconnect : System.Web.UI.Page
{
    SqlConnection con = new SqlConnection(@"Data
Source=.\SQLEXPRESS;AttachDbFilename=|DataDirectory|\employeedb.mdf;Integrated Security=True;User
Instance=True");
    static SqlDataAdapter da;
    static DataTable dt = new DataTable();
    DataSet ds = new DataSet();
    static int recptr = 0;

    void showdata()
    {
        dt.Clear();
        da = new SqlDataAdapter("select * from employee", con);
        da.Fill(dt);
    }
}

```

```
txtid.Text = dt.Rows[recptr][0].ToString();
txtnm.Text = dt.Rows[recptr][1].ToString();
txtdesignation.Text = dt.Rows[recptr][2].ToString();

GridView1.DataSource = dt;
GridView1.DataBind();
}

protected void Page_Load(object sender, EventArgs e)
{
if (!IsPostBack)
{
    showdata();
}
}

protected void btnfirst_Click(object sender, EventArgs e)
{
    recptr = 0;
    showdata();
}

protected void btnprevious_Click(object sender, EventArgs e)
{
if (recptr > 0)
{
    recptr--;
    showdata();
}
}

protected void btnnext_Click(object sender, EventArgs e)
{
if (recptr < dt.Rows.Count - 1)
{
    recptr++;
    showdata();
}
}

protected void btnlast_Click(object sender, EventArgs e)
{
    recptr = dt.Rows.Count - 1;
    showdata();
}

protected void btnadd_Click(object sender, EventArgs e)
{
    txtid.Text = "";
    txtnm.Text = "";
    txtdesignation.Text = "";
}

protected void btninsert_Click(object sender, EventArgs e)
{
    DataRow dr = dt.NewRow();

    dr[1] = txtnm.Text;
    dr[2] = txtdesignation.Text;
    dt.Rows.Add(dr);
}

SqlCommandBuilder cb = new SqlCommandBuilder(da);
da.Update(dt);
showdata();
}

protected void btnupdate_Click(object sender, EventArgs e)
{
    DataRow dr = dt.Rows[recptr];
    dr.BeginEdit();
    dr[0] = txtid.Text;
    dr[1] = txtnm.Text;
    dr[2] = txtdesignation.Text;
```

```
dr.EndEdit();
SqlCommandBuilder cb = new SqlCommandBuilder(da);
da.Update(dt);
showdata();
}

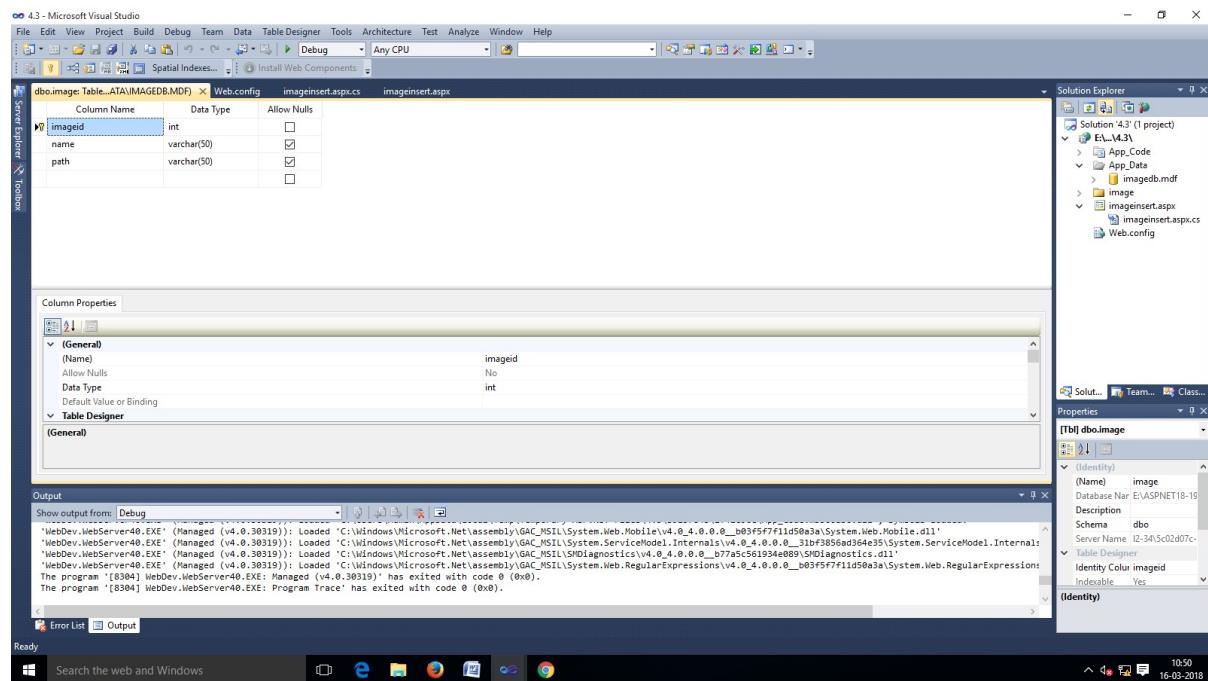
}
```

## At Run Time

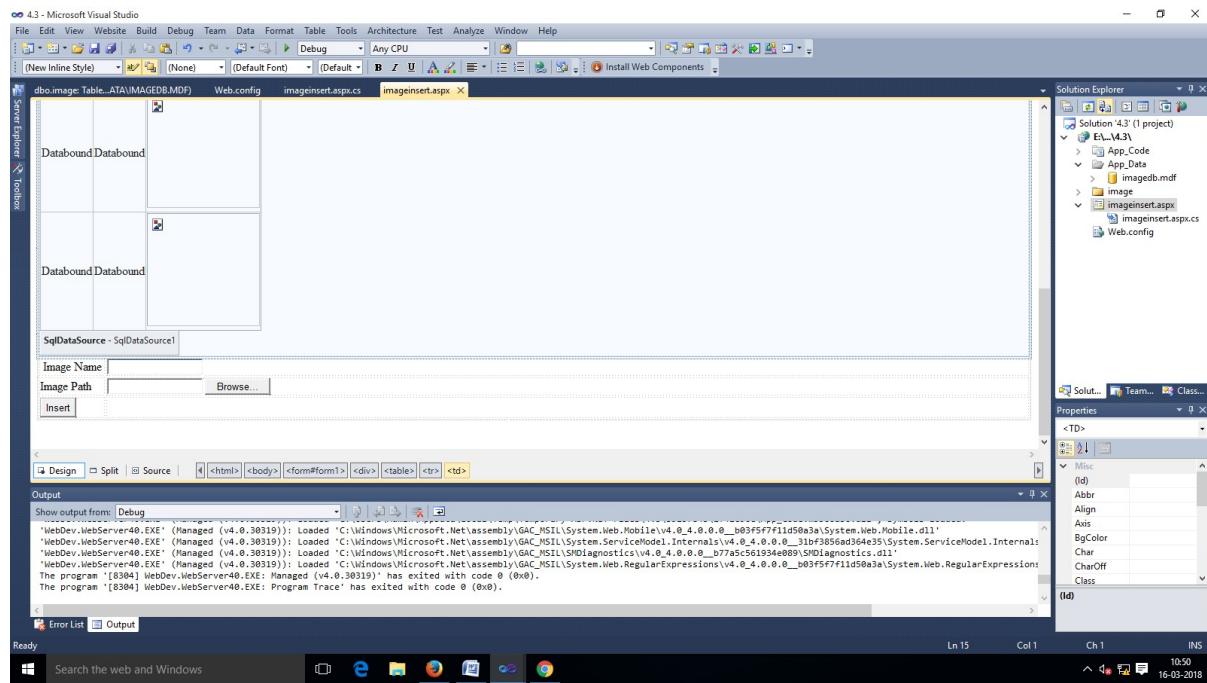


### 3.3) Insert (imageid, name, path) in image database and Display images in gridview

#### Table



## At Design Time:



## Imageinsert.aspx

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

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

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
<title>Untitled Page</title>
</head>
<body>
<form id="form1" runat="server">
<div>
<table style="width: 100%;">
<tr>
<td colspan="2">
<br/>
<br/>
<br/>
<asp:Label ID="Label4" runat="server" Font-Bold="True" Font-Size="Larger" Text="Image Example"></asp:Label>
</td>
</tr>
<tr>
<td colspan="2">
<asp:GridView ID="GridView1" runat="server" AutoGenerateColumns="False" DataKeyNames="imageid" DataSourceID="SqlDataSource1">
<Columns>
<asp:BoundField DataField="imageid" HeaderText="imageid" InsertVisible="False" ReadOnly="True" SortExpression="imageid"/>
<asp:BoundField DataField="name" HeaderText="name" SortExpression="name"/>
<asp:TemplateField HeaderText="pimage">
<ItemTemplate>
<asp:Image ID="image1" runat="server" Height="150" Width="150" ImageUrl='<%#Eval("path")%>'/>
</ItemTemplate>

```

```
</asp:TemplateField>
</Columns>
</asp:GridView>
<asp:SqlDataSourceID="SqlDataSource1" runat="server"
ConnectionString="<%$ ConnectionStrings:ConnectionString %>">
SelectCommand="SELECT * FROM [image]"</asp:SqlDataSource>
</td>
</tr>+
<tr>
<td>
 <asp:LabelID="Label2" runat="server" Text="Image Name"></asp:Label>
</td>
<td>
<asp:TextBoxID="txtname" runat="server"></asp:TextBox>
</td>
</tr>
<tr>
<td>
<asp:LabelID="Label3" runat="server" Text="Image Path"></asp:Label>
</td>
<td>
<asp:FileUploadID="flupath" runat="server"/>
</td>
</tr>
<tr>
<td>
<asp:ButtonID="btninsert" runat="server" onclick="btninsert_Click"
Text="Insert"/>
</td>
<td>
 </td>
</tr>
</table>
</div>
</form>
</body>
</html>
```

## Imageinsert.aspx.cs

```
using System;
using System.Collections;
using System.Configuration;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.HtmlControls;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Xml.Linq;
using System.Data.SqlClient;
using System.IO;

public partial class Imageinsert : System.Web.UI.Page
{
    SqlConnection con;
    SqlCommand cmd;
    connect c = new connect();
    static SqlDataAdapter da;
```

```
staticDataSet ds = newDataSet();
void refreshdata()
{
    ds.Clear();
    da = newSqlDataAdapter("select*from image", con);
    da.Fill(ds);

}
protectedvoid Page_Load(object sender, EventArgs e)
{
    con = newSqlConnection(c.getconstr());
    con.Open();
if (!IsPostBack)
{
    refreshdata();
}

}
protectedvoid btninsert_Click(object sender, EventArgs e)
{
string filename = Path.GetFileName(fupath.PostedFile.FileName);
    fupath.SaveAs(Server.MapPath("~/image/" + filename));
string img = "~/image/" + filename;
cmd = newSqlCommand("insert into image values(@name,@path)", con);
    cmd.Parameters.AddWithValue("@name", txtname.Text);
    cmd.Parameters.AddWithValue("@path", img);

    cmd.ExecuteNonQuery();
    refreshdata();

    GridView1.DataBind();
}
}
```

## At Run Time

## Subject :- ASP.NET

Screenshot of a Microsoft Edge browser window showing an ASP.NET application. The title bar reads "localhost:54617/4.3/imageinsert.aspx". The main content area displays a table titled "Image Example" with two rows:

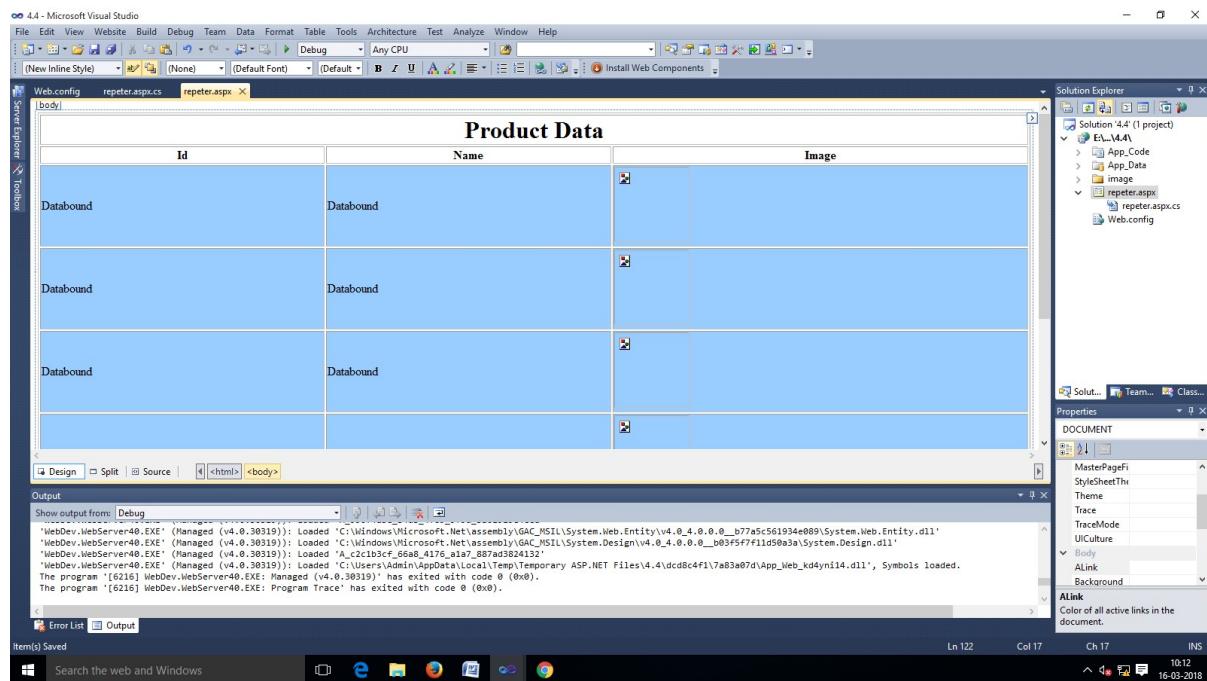
imageid	name	pimage
1	Blue Hill	
2	Wniter	

Below the table are input fields for "Image Name" (with placeholder "Enter name") and "Image Path" (with a "Choose File" button and a message "No file chosen"). A "Insert" button is at the bottom.



### 3.4) Example of repeater control on image database

**At Design Time:-**



### Repeater.aspx

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

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

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
<title>Untitled Page</title>
</head>
<body>
<form id="form1" runat="server">
<div>

<asp:Repeater ID="Repeater1" runat="server" DataSourceID="SqlDataSource1">
<HeaderTemplate>
<table width="100%" border="1">
<tr><th colspan="5">Product Data</th></tr>
<tr>
<th>Id</th>
<th>Name</th>
<th>Image</th>
</tr>
<tr><td>%#DataBinder.Eval(Container.DataItem, "imageid")%</td>
<td>%#DataBinder.Eval(Container.DataItem, "name")%</td>
</tr>
</table>
</HeaderTemplate>
<ItemTemplate>

<tr bgcolor="#99ccff">
<td>%#DataBinder.Eval(Container.DataItem, "imageid")%</td>
<td>%#DataBinder.Eval(Container.DataItem, "name")%</td>
</tr>
</ItemTemplate>
<FooterTemplate>
</FooterTemplate>
</asp:Repeater>
</div>
</form>

```

```
<td>
<asp:ImageID="Image1" runat="server" ImageUrl='<%# DataBinder.Eval(Container.DataItem,
"path")%>' Height="100" Width="100"/>
</td>

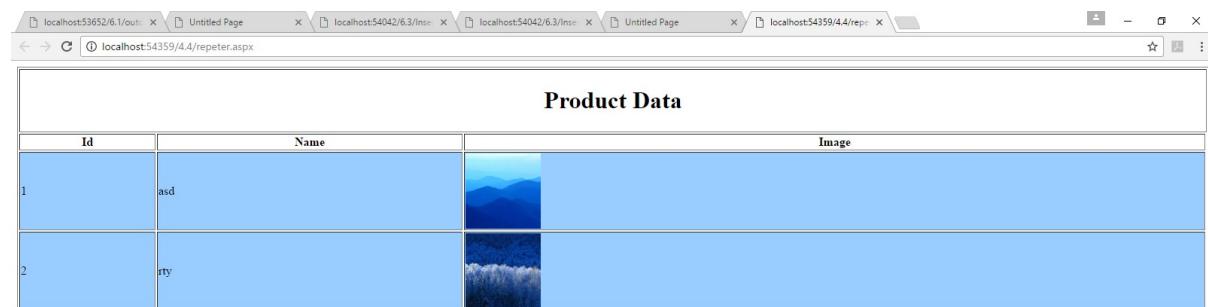
</tr>

</ItemTemplate>

<FooterTemplate>
</table>
</FooterTemplate>
</asp:Repeater>
<asp:SqlDataSourceID="SqlDataSource1" runat="server"
ConnectionString="<%$ ConnectionStrings:ConnectionString %>">
SelectCommand="SELECT * FROM [image]"></asp:SqlDataSource>
<br/>

</div>
</form>
</body>
</html>
```

### At Run Time:-



The screenshot shows a web browser window with multiple tabs open. The active tab displays a table titled "Product Data" with three columns: "Id", "Name", and "Image". There are two rows of data. The first row has an Id of 1 and a Name of "asd", with an image thumbnail showing a blue landscape. The second row has an Id of 2 and a Name of "try", with an image thumbnail showing a forest scene.

Id	Name	Image
1	asd	
2	try	



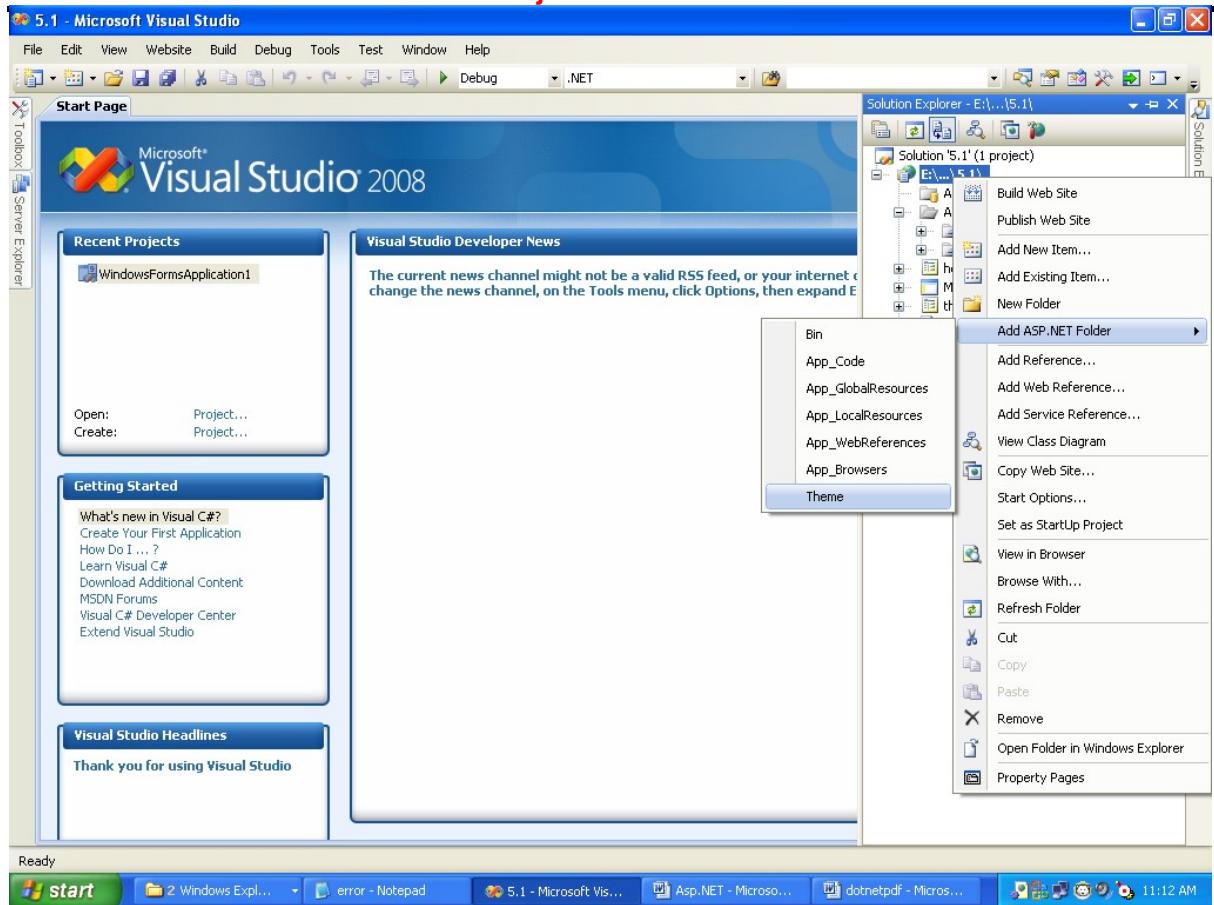
# Chapter-4

## Master Pages and Theme caching, Application Pages and Data

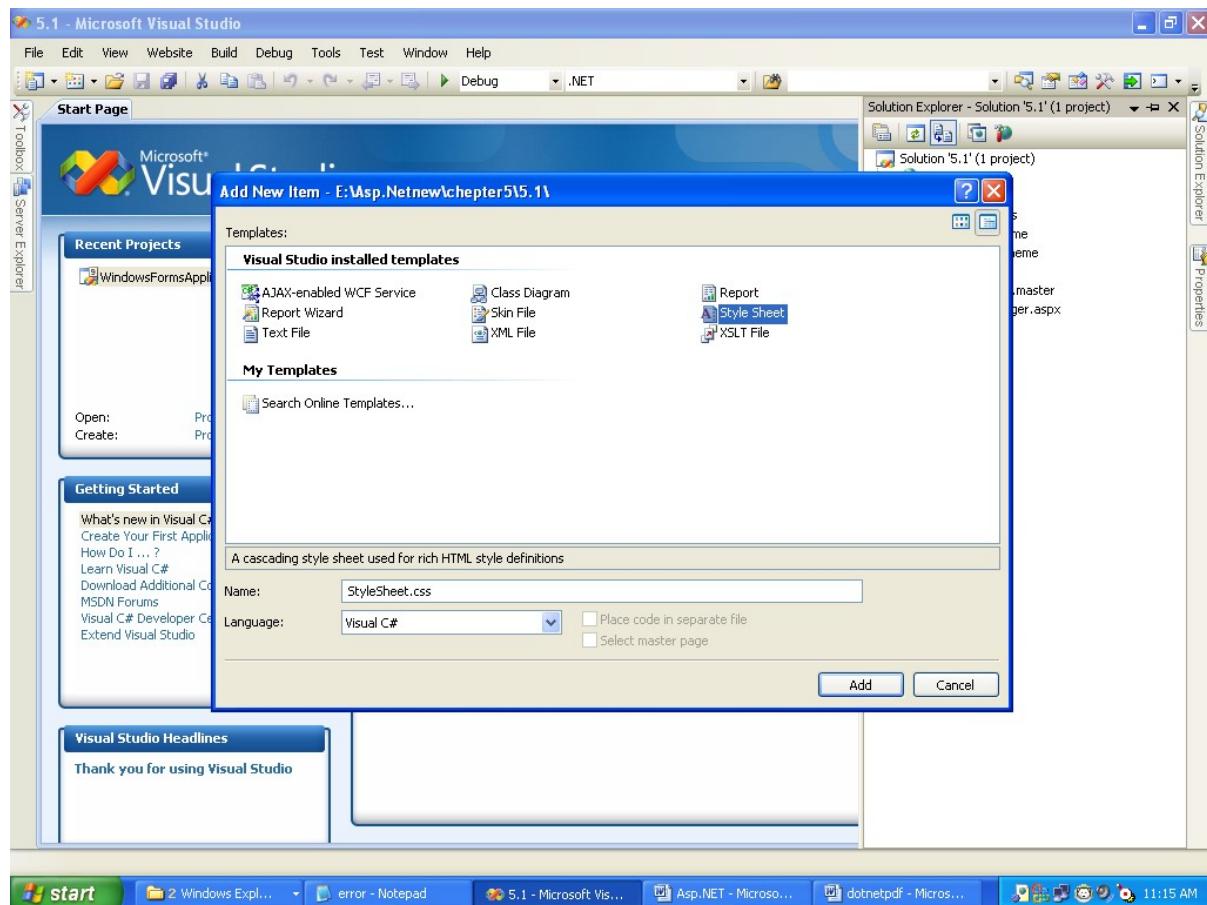
### 4.1) Example of master page using theme and css

First Add Theme BlueTheme

## Subject :- ASP.NET



## Add StyleSheet BlueStylesheet



### 1)BlueStyleSheet

```

body
{
    margin:20px20px20px20px;
    padding: 0;
    font-family:Georgia,TimesNewRoman,Serif;
    color:Black;
    width:960px;
    border-right:1pxsolidblack;
    background-color:#98B3F6;
}
#header
{
    background-color:#1B7DCE;
    margin:10px10px0px10px;
    height :120px;
    overflow:hidden;
}
#content
{
    padding-top:1em;
    margin: 012px0180px;
    background-color:White;
    overflow:hidden;
}
#footer
{
    clear: both;
}

```

```
background-color:#62645C;
padding-bottom:1em;
border-top: 1px solid #333;
padding-left: 200px;
}

#header.Title
{
    color:White;
}
#hedaerh1
{
    margin:10px 40px 10px 40px;
}
#Container
{
    background-color:#F5F5F5;
    overflow:hidden;
}
a
{
    color:#1177cc;
}
a:visited
{
    color:Gray;
}
.MenuLi
{
    display:inline;
    margin:0px;
}
.Menua
{
    text-decoration:none;
    background-color:Blue;
    padding:5px;
    color:White;
    border-right:4px solid Maroon;
    margin:0px;
    border-left:none;
}
.Menua:hover
{
    background-color:Maroon;
    padding:5px;
    border-right:4px solid Blue;
    margin:0px;
}
.welcomespan
{
    color:#98B3F6;
}
```

**First Add Theme GreenTheme****Add StyleSheet GreenStylesheet**

```

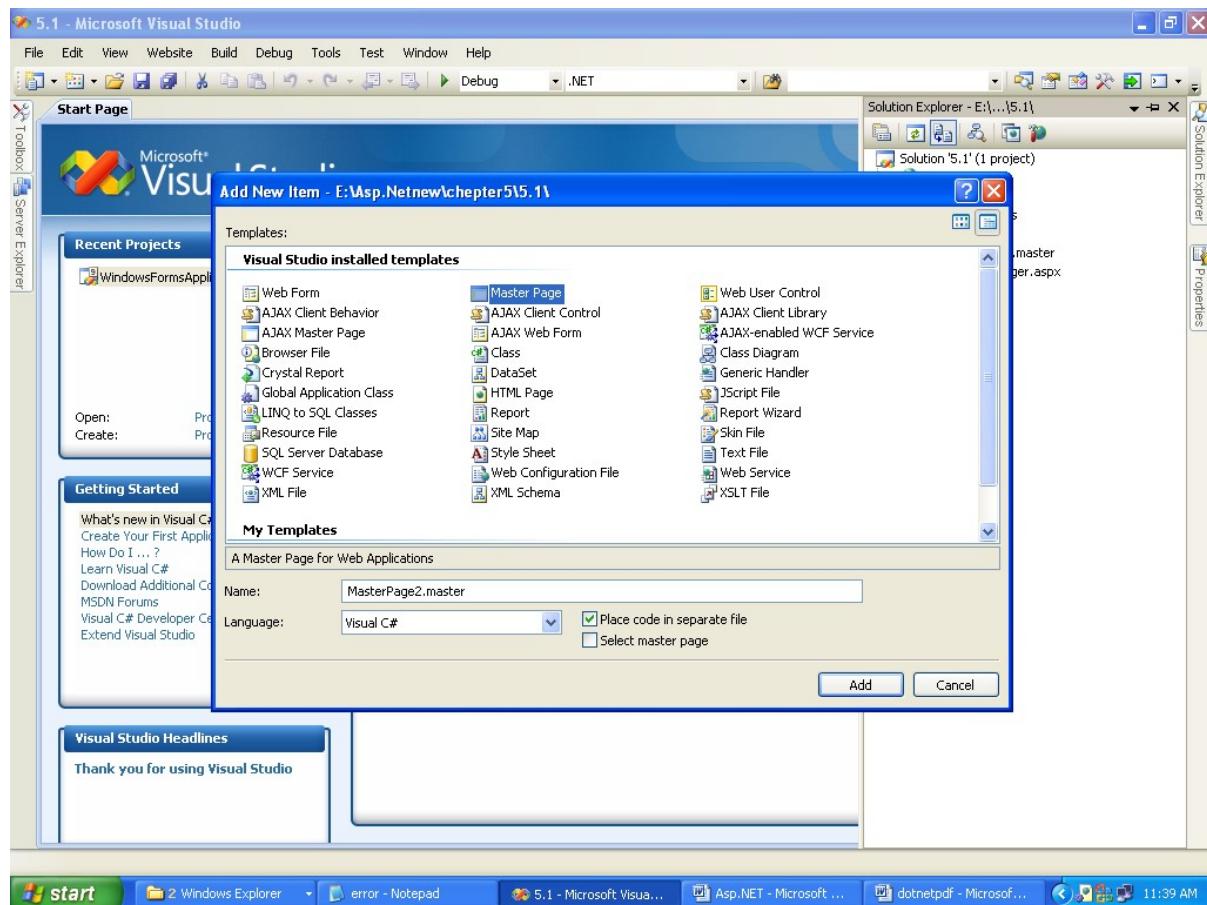
body
{
    margin:20px20px20px20px;
    padding: 0;
    font-family:Georgia,TimesNewRoman,Serif;
    color:Black;
    width:960px;
    border-right:1pxsolidblack;
    background-color:Green;
}
#header
{
    background-color:#CE6E1E;
    margin:10px10px0px10px;
    height :120px;
    overflow:hidden;
}
#content
{
    padding-top:1em;
    margin: 012px0180px;
    background-color:White;
    overflow:hidden;
}
#footer
{
    clear: both;
    background-color:#446476;
    padding-bottom:1em;
    border-top: 1pxsolid#333;
    padding-left: 200px;
}

#header.Title
{
    color:White;
}
#hedaerh1
{
    margin:10px40px10px40px;
}
#container
{
    background-color:#F5F5F5;
    overflow:hidden;
}
a
{
    color:#BC3455;
}
a:visited
{
    color:Gray;
}

```

```
.Menuli
{
    display:inline;
    margin:0px;
}
.Menua
{
    text-decoration:none;
    background-color:#FFA284;
    padding:5px;
    color:Black;
    border-right:4px solid #E7FBFD;
    margin:0px;
    border-left:none;
}
.Menua:hover
{
    background-color:#E07F93;
    padding:5px;
    border-right:4px solid #FFA2284;
    margin:0px;
}
.welcomespan
{
    color:#CE6E1E;
}
```

## Add MasterPage



```
<%@MasterLanguage="C#" AutoEventWireup="true" CodeFile="MasterPage.master.cs" Inherits="MasterPage"
%>

<!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 id="Head1" runat="server">
<title>Untitled Page</title>

</head>
<body>
<form id="form1" runat="server">
<div id="container">
<div id="header">
<h1 class="Title">MasterPage With Theme Website</h1>
<ul class="menu">

<li><a href="home.aspx">Home</a></li>
<li><a href="thememanager.aspx">Setting</a></li>
</ul>
</div>
<div id="contect">
<asp:ContentPlaceHolder id="ContentPlaceHolder1" runat="server">

</asp:ContentPlaceHolder>
</div>
<div id="footer">
<h3 class="Title">This is footer</h3>
</div>
```

```
</div>
</form>
</body>
</html>
```

---

## **Add Two WebForms (1)ThemeManager (2) Home**

### **Home.aspx**

```
<%@PageLanguage="C#" MasterPageFile="~/MasterPage.master" AutoEventWireup="true" CodeFile="home.aspx.cs" Inherits="home" Title="Untitled Page"%>

<asp:Content ID="Content1" ContentPlaceHolderID="ContentPlaceHolder1" Runat="Server">
<div class="welcome">
<h2><span>Welcome</span>to our website</h2>
<p>This is the welcome page</p>
</div>
</asp:Content>
```

### **ThemeManager.aspx**

```
<%@PageLanguage="C#" MasterPageFile="~/MasterPage.master" AutoEventWireup="true" CodeFile="thememanager.aspx.cs" Inherits="thememanager" Title="Untitled Page"%>

<asp:Content ID="Content1" ContentPlaceHolderID="ContentPlaceHolder1" Runat="Server">
<asp:DropDownList ID="DropDownList1" runat="server"
onselectedindexchanged="DropDownList1_SelectedIndexChanged"
AutoPostBack="True">
<asp:ListItem>Select Any Item</asp:ListItem>
<asp:ListItem>BlueTheme</asp:ListItem>
<asp:ListItem>GreenTheme</asp:ListItem>
</asp:DropDownList>
</asp:Content>
```

---

**ThemeManager.cs**

```

using System;
using System.Collections;
using System.Configuration;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.HtmlControls;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Xml.Linq;
using System.Data.SqlClient;
using System.Web.Configuration;

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

    }

    protected void DropDownList1_SelectedIndexChanged(object sender, EventArgs e)
    {
        Configuration config = WebConfigurationManager.OpenWebConfiguration("~/");
        PagesSection pages = (PagesSection)config.GetSection("system.web/pages");
        pages.Theme = DropDownList1.SelectedItem.Text.ToString();
        if (!pages.SectionInformation.IsLocked)
        {
            config.Save();
            Response.Redirect("home.aspx");
        }
        else
        {
            Response.Write("<script>alert('configuration is not saved')</script>");
        }
    }
}

```

**In Webconfig file set**

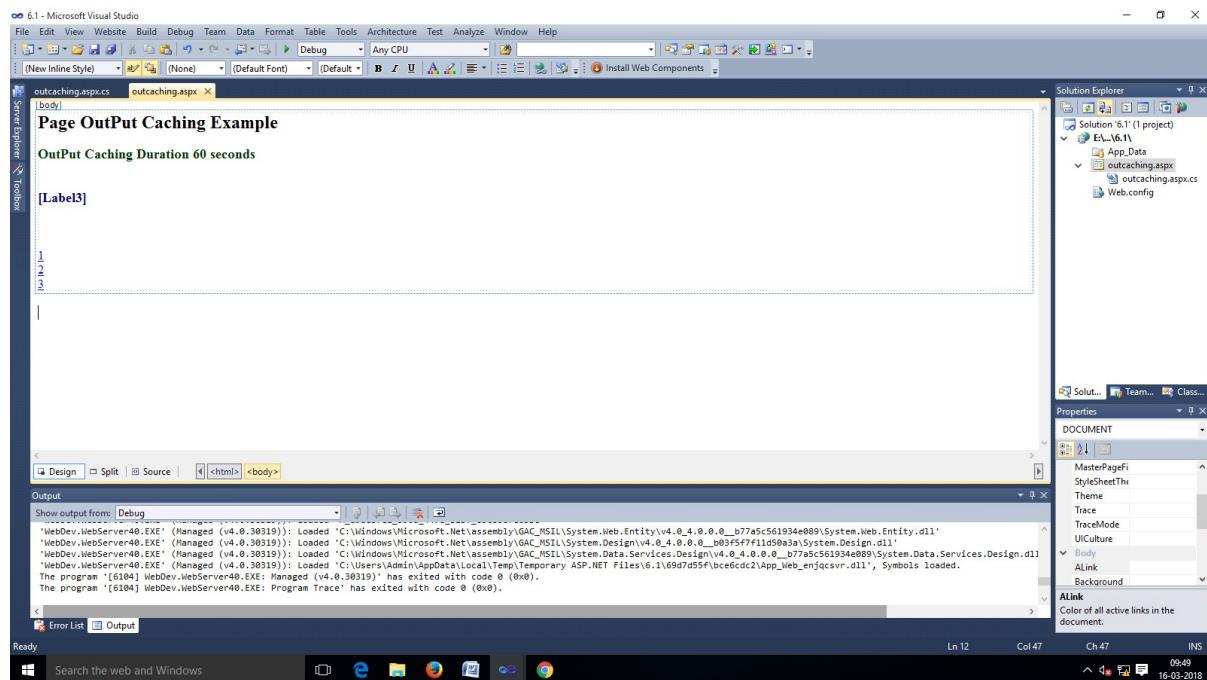
```
<pagestheme="GreenTheme">
```

**At Run Time:-**



## 4.2) WAP of Page output caching

### At Design Time:



### Outcaching.aspx

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

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<%@OutputCache Duration="60" VaryByParam="p" %>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
<title>Untitled Page</title>
</head>
<body>
<form id="form1" runat="server">
<div>

<asp:Label ID="Label1" runat="server" Font-Bold="True" Font-Size="Larger" Text="Page OutPut Caching Example"></asp:Label>
<br/>
<br/>
<asp:Label ID="Label2" runat="server" Font-Bold="True" Font-Size="Large" ForeColor="#003300" Text="OutPut Caching Duration 60 seconds"></asp:Label>
<br/>
<br/>
<br/>
<asp:Label ID="Label3" runat="server" Font-Bold="True" Font-Size="Large" ForeColor="#000066"></asp:Label><br/>
<br/><br/><br/>

<a href="?p=1">1</a><br/>
<a href="?p=2">2</a><br/>
<a href="?p=3">3</a>
</div>
```

</form>

</body>  
</html>

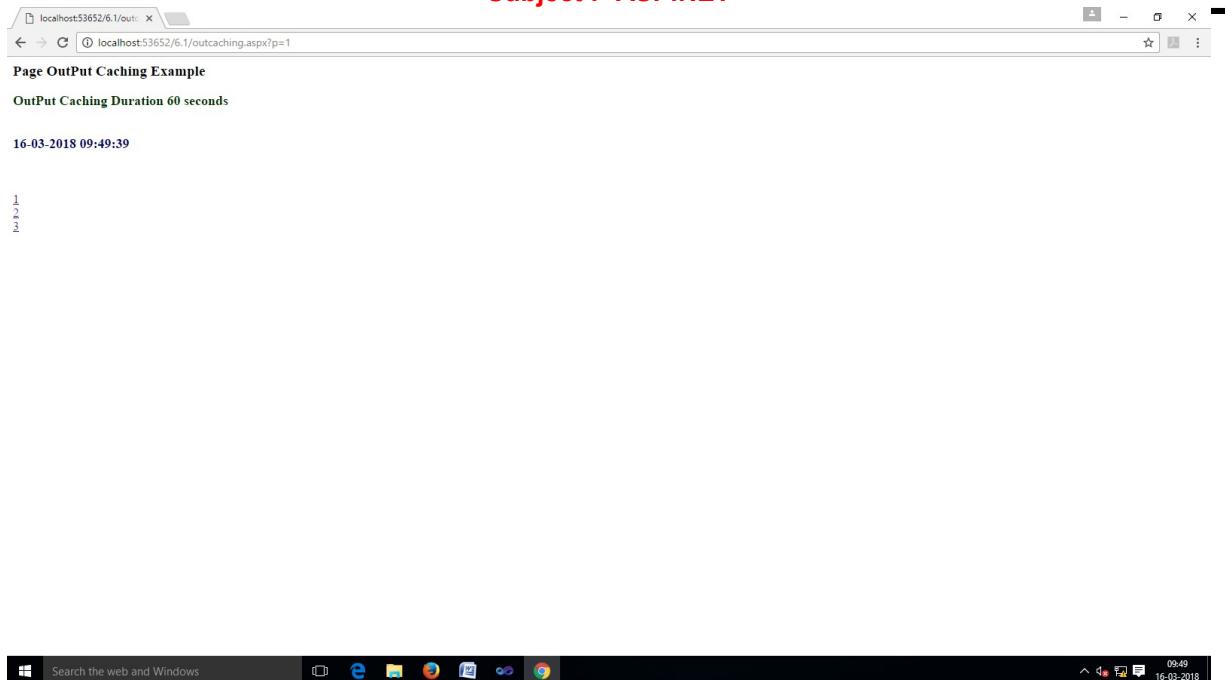
## **Outcaching.aspx.cs**

```
using System;
using System.Configuration;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.HtmlControls;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Xml.Linq;

public partial class _outcaching : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        Label3.Text = DateTime.Now.ToString();
    }
}
```

**At Run Time:-**

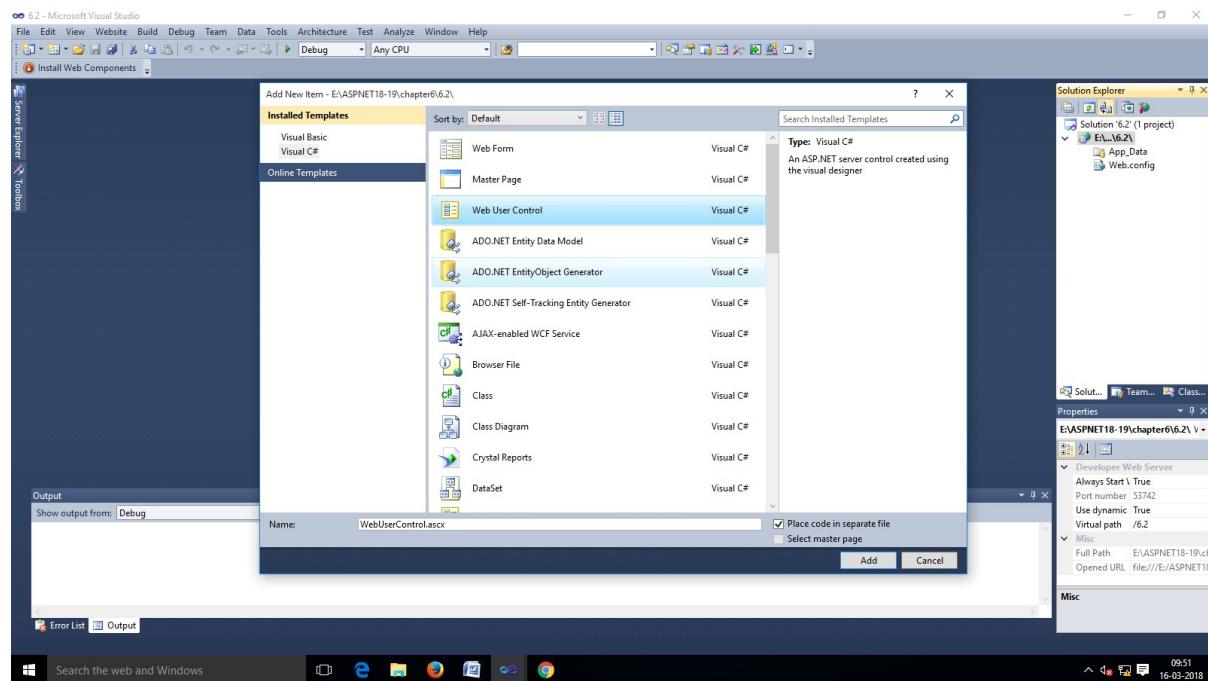
## Subject :- ASP.NET



## 4.3) WAP of partial page caching

- At Design Time

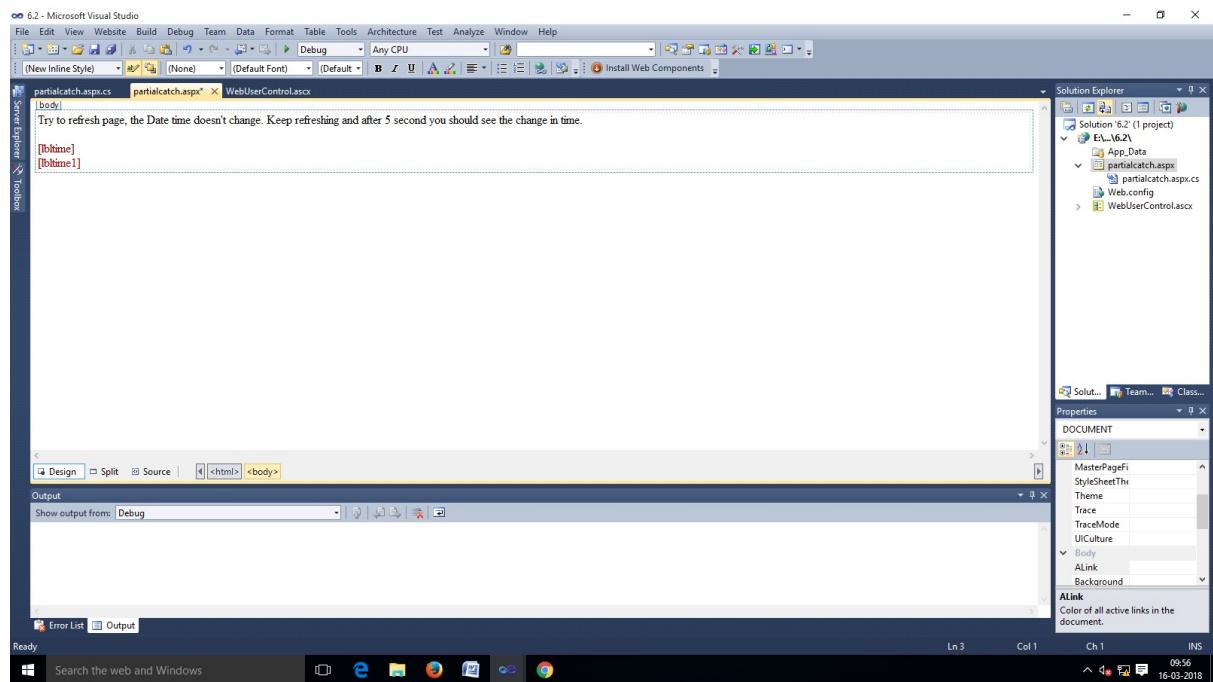
### Add WebUserControl



### WebUserControl.ascx

```
<%@ControlLanguage="C#" AutoEventWireup="true" CodeFile="WebUserControl.ascx.cs" Inherits="WebUserControl"%>
<%@OutputCacheDuration="5" VaryByParam="None"%>
<asp:LabelID="lbltime" runat="server" EnableViewState="False" ForeColor="Maroon"></asp:Label>
```

## Partialcatch.aspx



## Partialcatch.aspx

```
<%@PageLanguage="C#" AutoEventWireup="true" CodeFile="partialcatch.aspx.cs" Inherits="partialcatch"%>
<%@RegisterSrc="~/WebUserControl.ascx" TagPrefix="uc" TagName="usercontrol"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN""http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

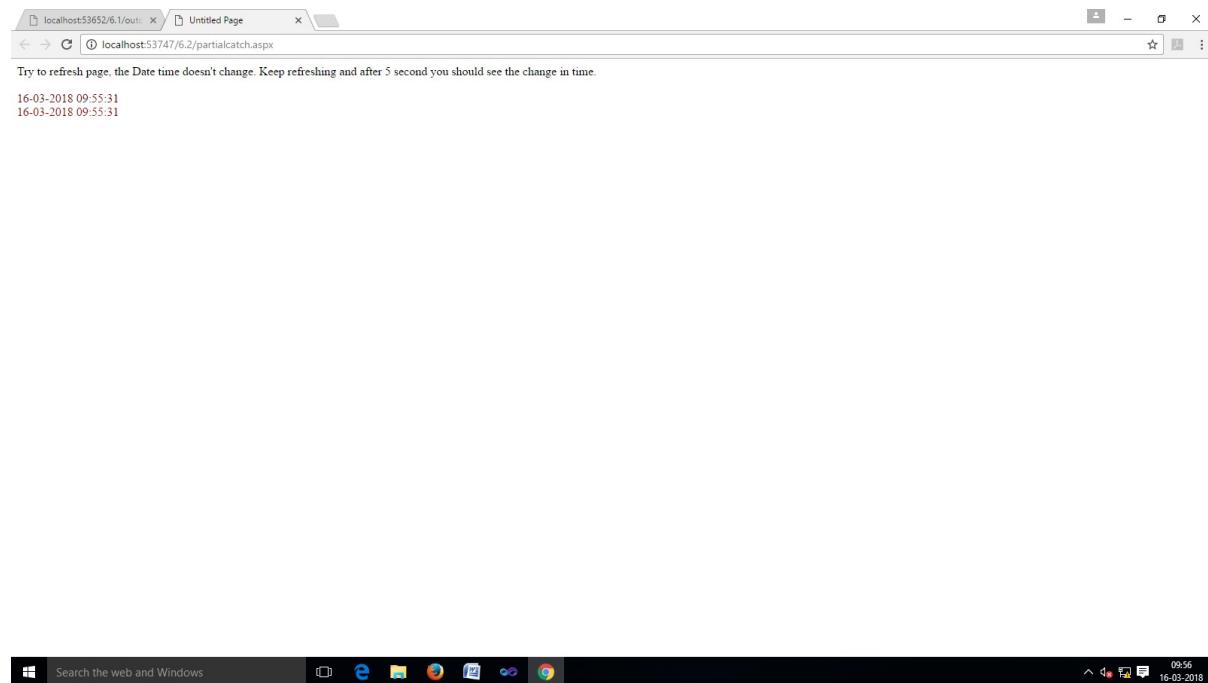
<htmlxmlns="http://www.w3.org/1999/xhtml">
<headrunat="server">
<title>Untitled Page</title>
</head>
<body>
<formid="form1"runat="server">
<div>
    Try to refresh page, the Date time doesn't change. Keep refreshing and after 5 second you should see the
    change in time.
    <br/><br/><uc:usercontrolrunat="server"ID="uc1"/>
    <br/>
    <asp:LabelID="lbltime1"runat="server"EnableViewState="False"
    ForeColor="#990000"></asp:Label>
</div>
</form>
</body>
</html>
```

## Partialcatch.aspx.cs

```
using System;
using System.Collections;
using System.Configuration;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.HtmlControls;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Xml.Linq;

public partial class partialcatch : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        lbltime1.Text = DateTime.Now.ToString();
    }
}
```

### At RunTime:-

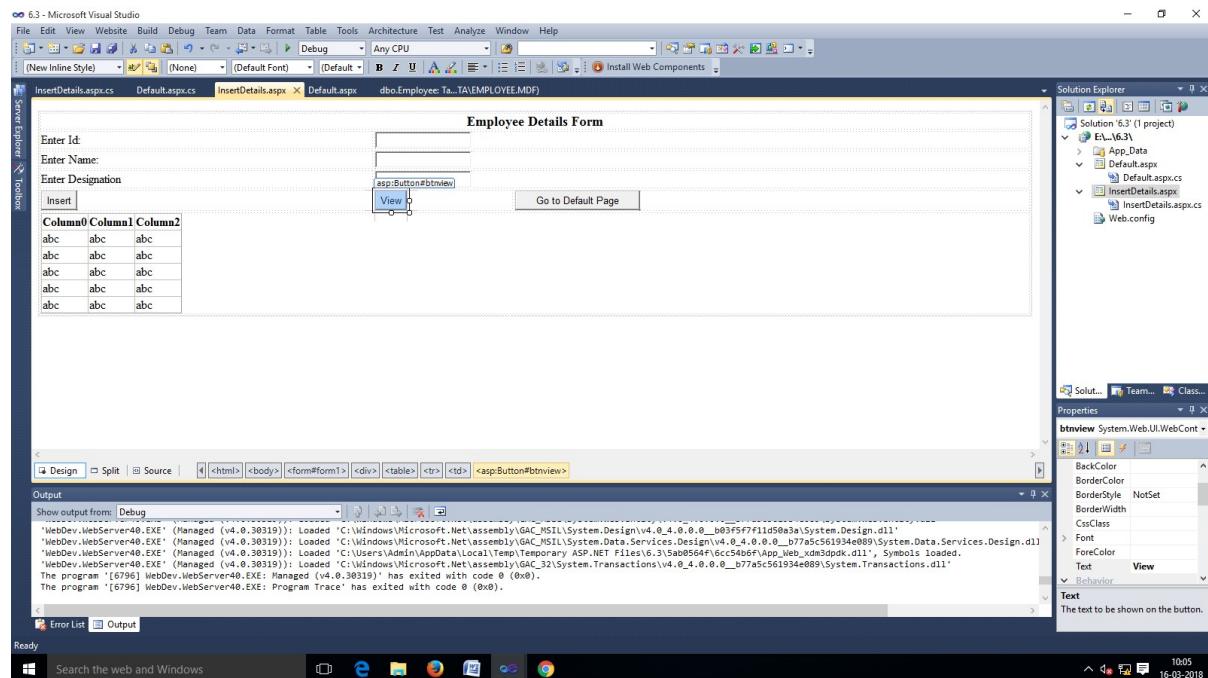


## 4.4) WAP of data caching

For this application we take two forms. 1) InsertDetails 2) Default.aspx

### - At Design Time

#### InsertDetails.aspx

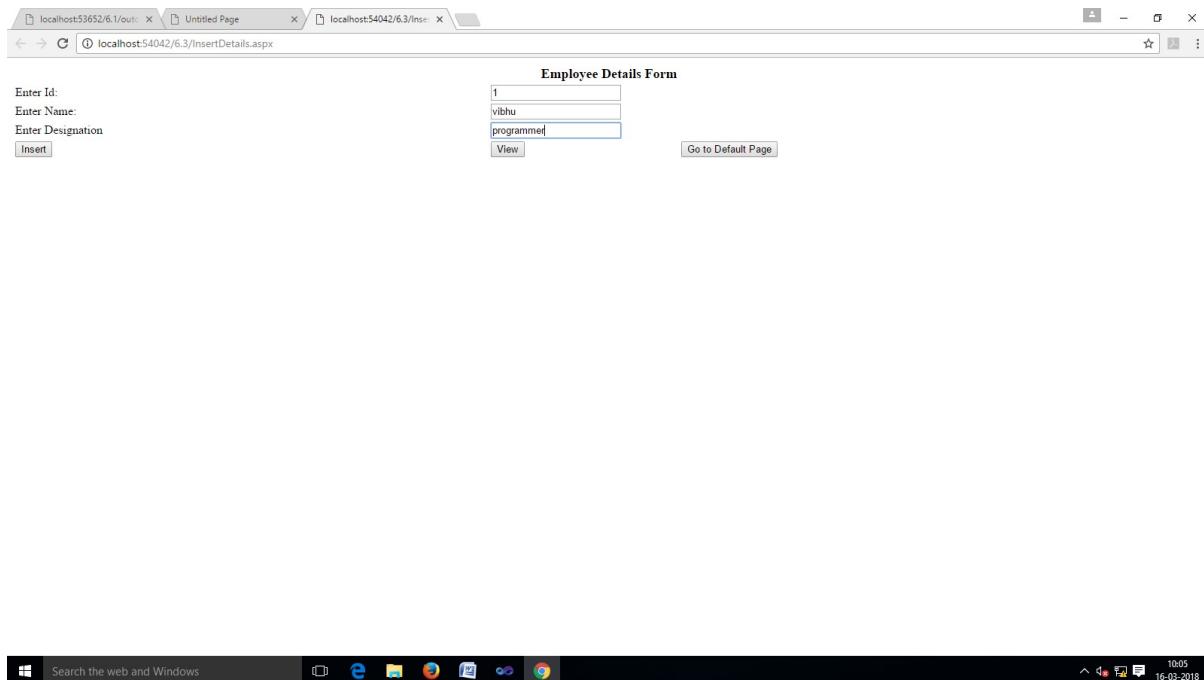


#### InsertDetails.aspx.cs

```
using System;
using System.Collections;
using System.Configuration;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.HtmlControls;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Xml.Linq;
using System.Data.SqlClient;
public partial class InsertDetails : System.Web.UI.Page
{
    SqlConnection con;
    protected void Page_Load(object sender, EventArgs e)
    {
        con = new SqlConnection(@"Data
Source=.\SQLEXPRESS;AttachDbFilename=|DataDirectory|\Employee.mdf;Integrated Security=True;User
Instance=True");
        con.Open();
    }
    protected void btninsert_Click(object sender, EventArgs e)
    {
        string query = "insert into Employee values('" + txtid.Text + "','" + txtname.Text + "','" + txtdesignation.Text + "')";
        SqlCommand cmd = new SqlCommand(query, con);
        cmd.ExecuteNonQuery();
    }
}
```

```
        }
protectedvoid btnview_Click(object sender, EventArgs e)
{
    SqlDataAdapter sda = new SqlDataAdapter("select * from Employee",con);
    DataTable dt = new DataTable();
    sda.Fill(dt);
    GridView1.DataSource = dt;
    GridView1.DataBind();
}
protectedvoid Btngotopage_Click(object sender, EventArgs e)
{
    Response.Redirect("Default.aspx");
}
}
```

**At RunTime:**



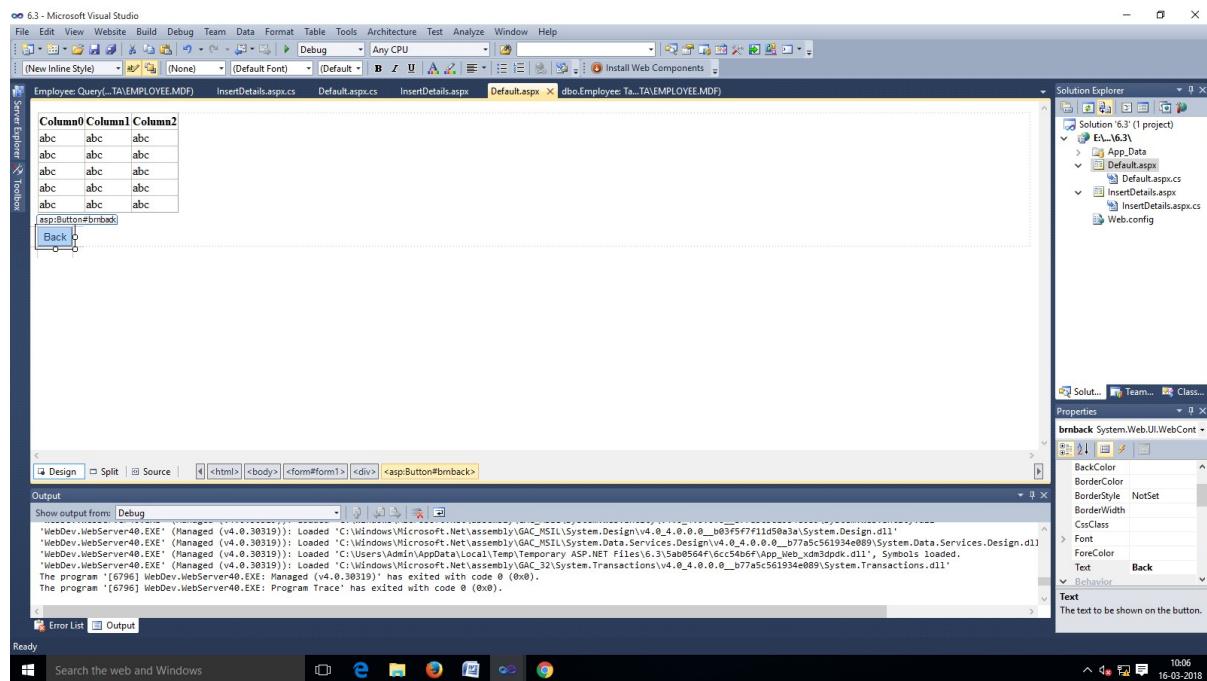
**After Clicking on insert Button**

## Subject :- ASP.NET

The screenshot shows a Windows desktop environment with a web browser window open to an ASP.NET page. The browser tabs include 'localhost:53652/6.1/out...', 'Untitled Page', and 'localhost:54042/6.3/InsertDetails.aspx'. The main content area displays an 'Employee Details Form' with fields for 'Enter Id:', 'Enter Name:', and 'Enter Designation'. Below the form is a table with columns 'id', 'name', and 'designation'. The table contains two rows of data: (1, vibhuti, programmer) and (2, vbs, programmer). A 'View' button is located next to the table, and a 'Go to Default Page' link is at the bottom right.

id	name	designation
1	vibhuti	programmer
2	vbs	programmer

## **Default.aspx**



## **Default.aspx.cs**

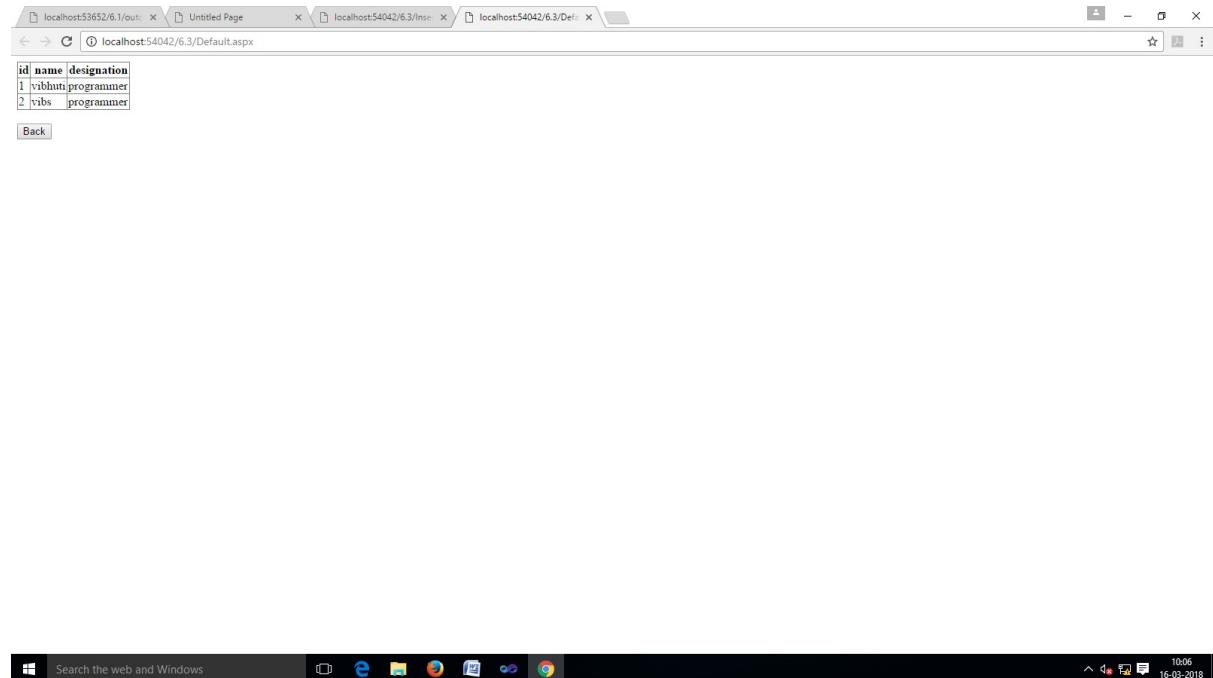
```

using System;
using System.Collections;
using System.Configuration;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.HtmlControls;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Xml.Linq;
using System.Data.SqlClient;
public partial class _Default : System.Web.UI.Page
{
    SqlConnection cn;
    protected void Page_Load(object sender, EventArgs e)
    {
        cn = new SqlConnection(@"Data
Source=.\SQLEXPRESS;AttachDbFilename=|DataDirectory|\Employee.mdf;Integrated Security=True;User
Instance=True");
        cn.Open();
        DataSet dsEmployee;
        if (Cache["Employee"] == null)
        {
            dsEmployee = new DataSet();
            SqlDataAdapter daEmployee;
            daEmployee = new SqlDataAdapter("select * from Employee", cn);
            daEmployee.Fill(dsEmployee, "Employee");
            Cache.Insert("Employee", dsEmployee, new
System.Web.Caching.CacheDependency(Server.MapPath("Master.xml")), DateTime.Now.AddSeconds(45),
TimeSpan.Zero);
        }
    }
}

```

```
        cn.Close();
        cn.Dispose();
    }
else
{
    dsEmployee=(DataSet)Cache["Employee"];
}
GridView1.DataSource = dsEmployee;
GridView1.DataBind();
}
protected void brnback_Click(object sender, EventArgs e)
{
    Response.Redirect("InsertDetails.aspx");
}
}
```

## At RunTime



# Chapter-5

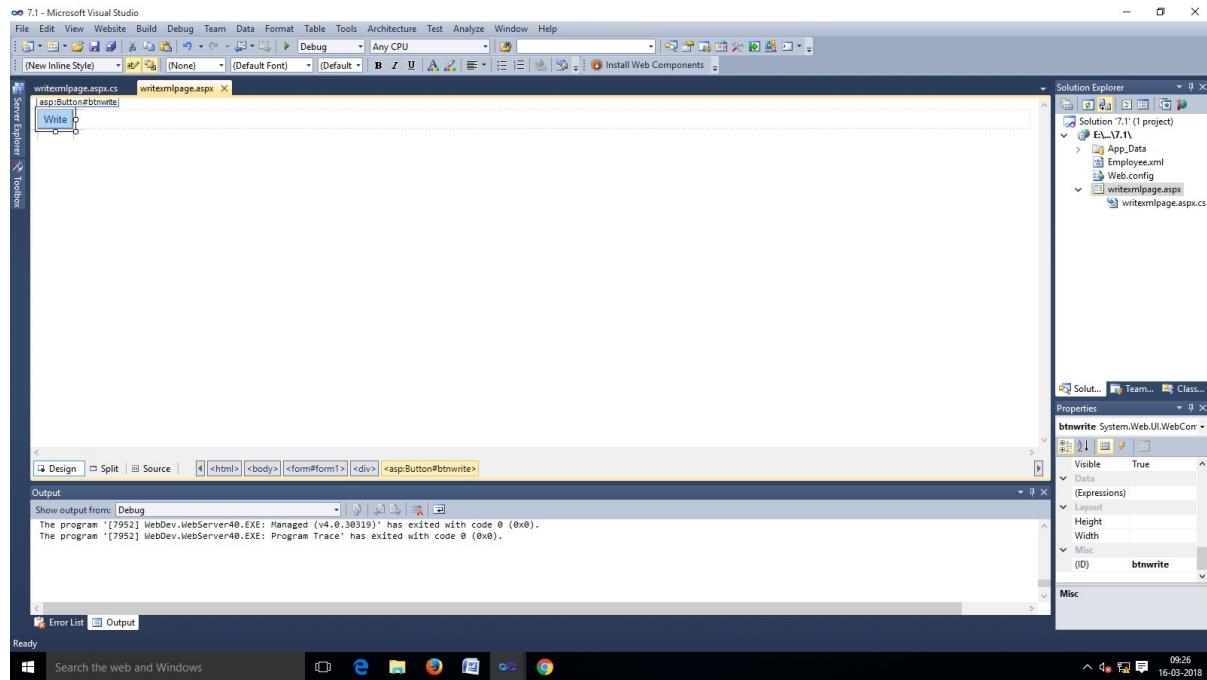
## Working With XML

### Asp.Net Application configuration and Deployment of Application

## **5.1) WAP to fetch record from database and write data in XML file**

For this application we take two forms.1) writexmlpage.aspx 2) Employee.xml

### **At Design Time:**



### **WriteXmlpage.aspx.cs**

```
using System;
using System.Collections;
using System.Configuration;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.HtmlControls;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Xml.Linq;
using System.Data.SqlClient;

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

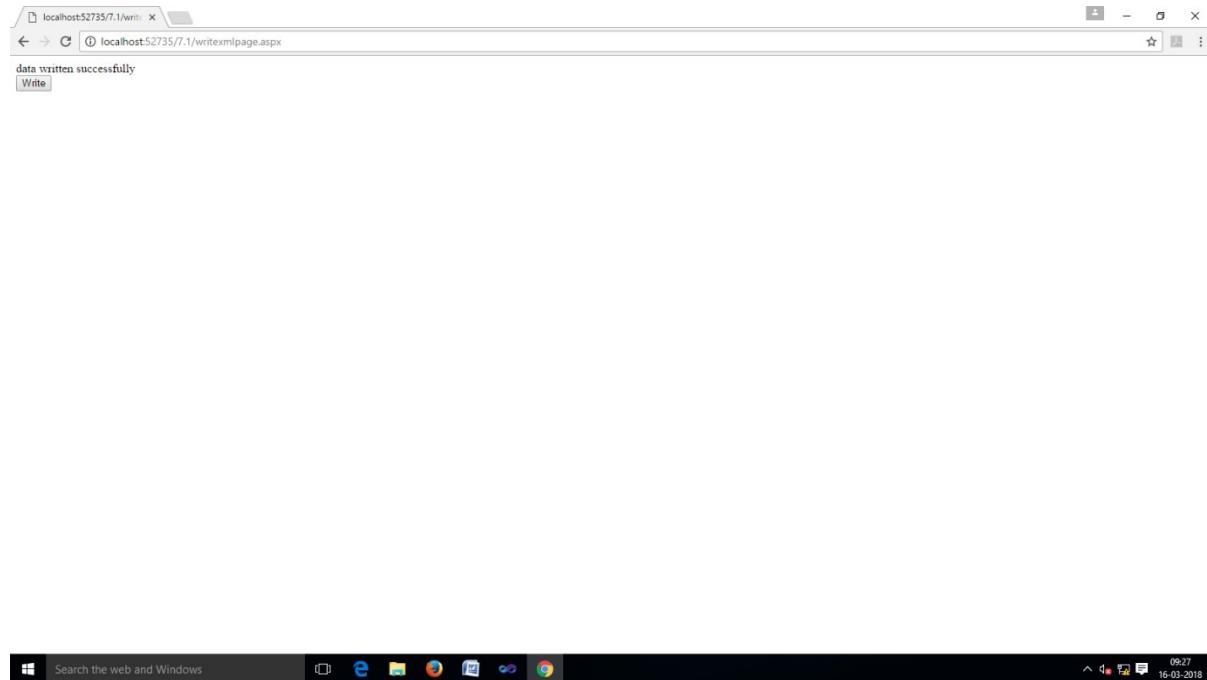
    }
    protected void btnwrite_Click(object sender, EventArgs e)
    {
        SqlConnection con = new SqlConnection(@"Data
Source=.\SQLEXPRESS;AttachDbFilename=|DataDirectory|\Employee.mdf;Integrated Security=True;User
Instance=True");
        con.Open();
        string commandString = "Select * from Employee";
        SqlDataAdapter da = new SqlDataAdapter(commandString, con);
        DataSet ds = new DataSet();
        da.Fill(ds);
        string st = Server.MapPath("~/");
    }
}
```

```
ds.WriteXml(st + "\\Employee.xml");
Response.Write("data written successfully");
}
}
```

## **Employee.xml**

```
<?xmlversion="1.0"standalone="yes"?>
<NewDataSet>
<Table>
<id>1    </id>
<name>Bhavin</name>
<designation>Professor</designation>
</Table>
<Table>
<id>2    </id>
<name>Khushal</name>
<designation>Professor</designation>
</Table>
<Table>
<id>3    </id>
<name>Harshit</name>
<designation>Professor</designation>
</Table>
</NewDataSet>
```

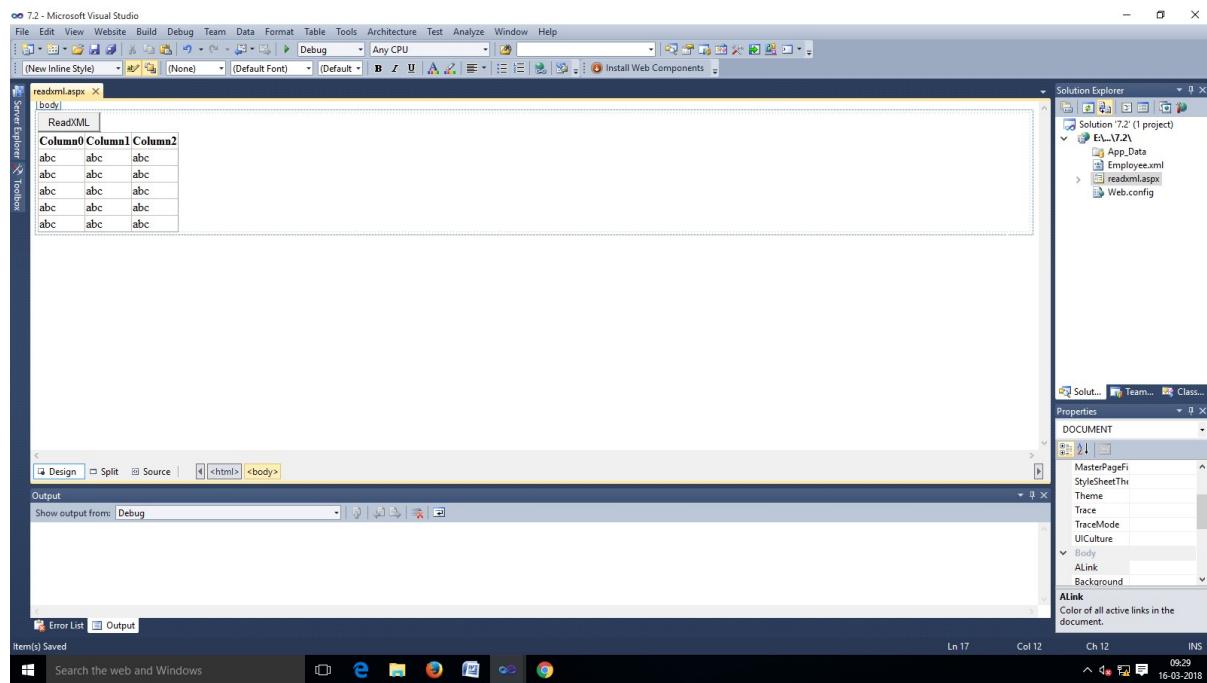
## **At RunTime:**



## **5.2) WAP to fetch record from XML file and display in gridview**

For this application we take two forms.1) readxml.aspx 2) Employee.xml

### **At Design Time:**



### **Readxml.aspx.cs**

```
using System;
using System.Collections;
using System.Configuration;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.HtmlControls;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Xml.Linq;

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

    }
    protected void btnreadxml_Click(object sender, EventArgs e)
    {
        DataSet ds = new DataSet();
        ds.ReadXml(MapPath("Employee.xml"));
        GridView1.DataSource = ds;
        GridView1.DataBind();
    }
}
```

### **Employee.xml**

```
<?xmlversion="1.0"encoding="utf-8" ?>
<NewDataSet>
```

```
<Table>
```

```
<EmployeeId>1</EmployeeId>
<Name>vibhuti</Name>
<Designation>programmer</Designation>
</Table>
```

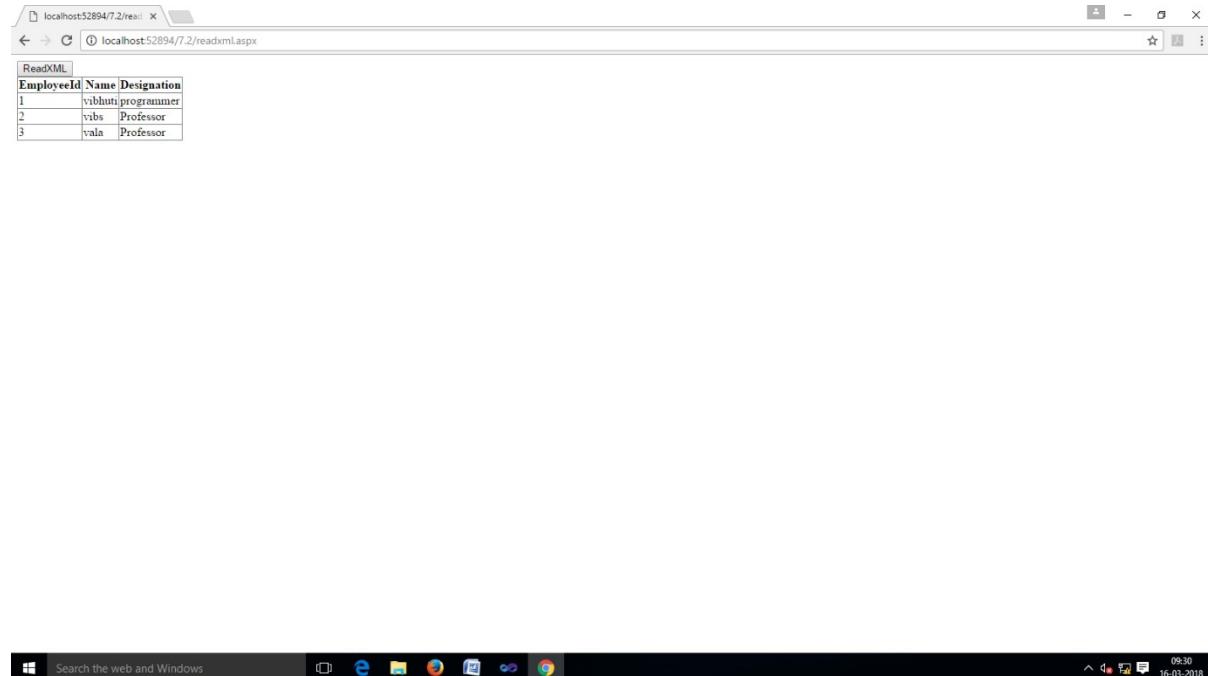
```
<Table>
```

```
<EmployeeId>2</EmployeeId>
<Name>vibs</Name>
<Designation>Professor</Designation>
</Table>
```

```
<Table>
```

```
<EmployeeId>3</EmployeeId>
<Name>vala</Name>
<Designation>Professor</Designation>
</Table>
</NewDataSet>
```

## At RunTime

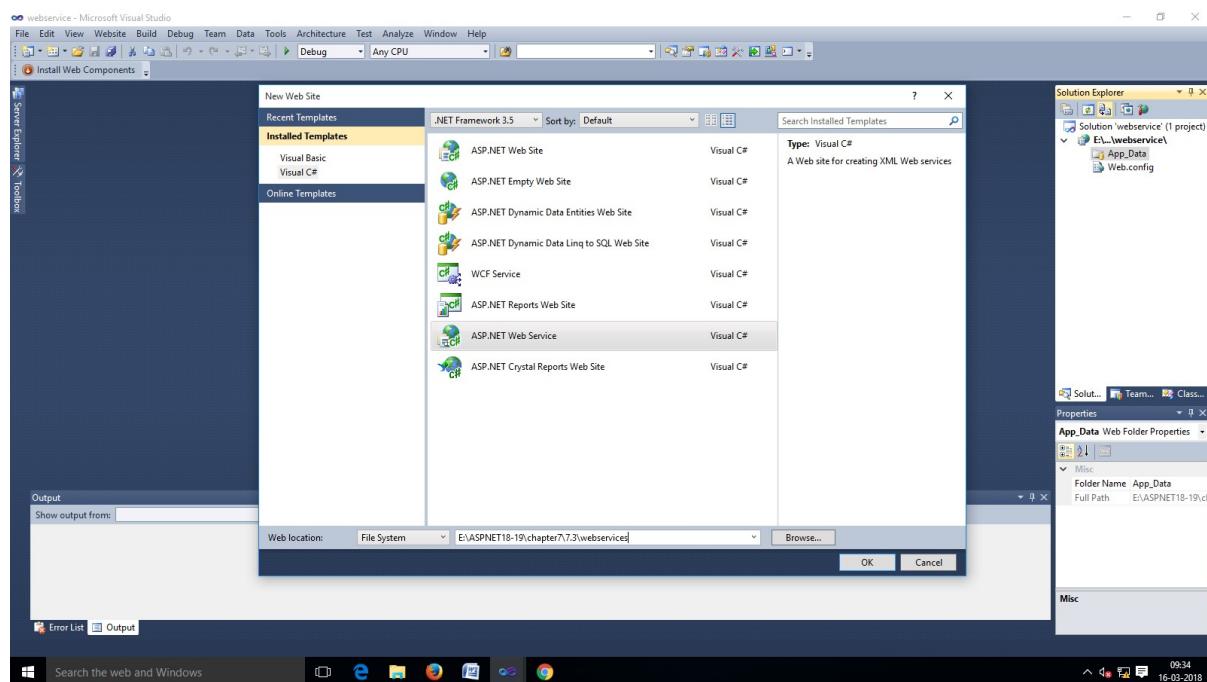


### 5.3) Example of web service

For this First WE Needs To create webservice file in one application

Insert a WEbservice File at the time of creating new application as below

#### At Design Time:



#### Service.cs

```

using System;
using System.Linq;
using System.Web;
using System.Web.Services;
using System.Web.Services.Protocols;
using System.Xml.Linq;

[WebService(Namespace = "http://tempuri.org/")]
[WebServiceBinding(ConformsTo = WsiProfiles.BasicProfile1_1)]
// To allow this Web Service to be called from script, using ASP.NET AJAX, uncomment the following line.
// [System.Web.Script.Services.ScriptService]
public class Service : System.Web.Services.WebService
{
    public Service()
    {
        //Uncomment the following line if using designed components
        //InitializeComponent();
    }

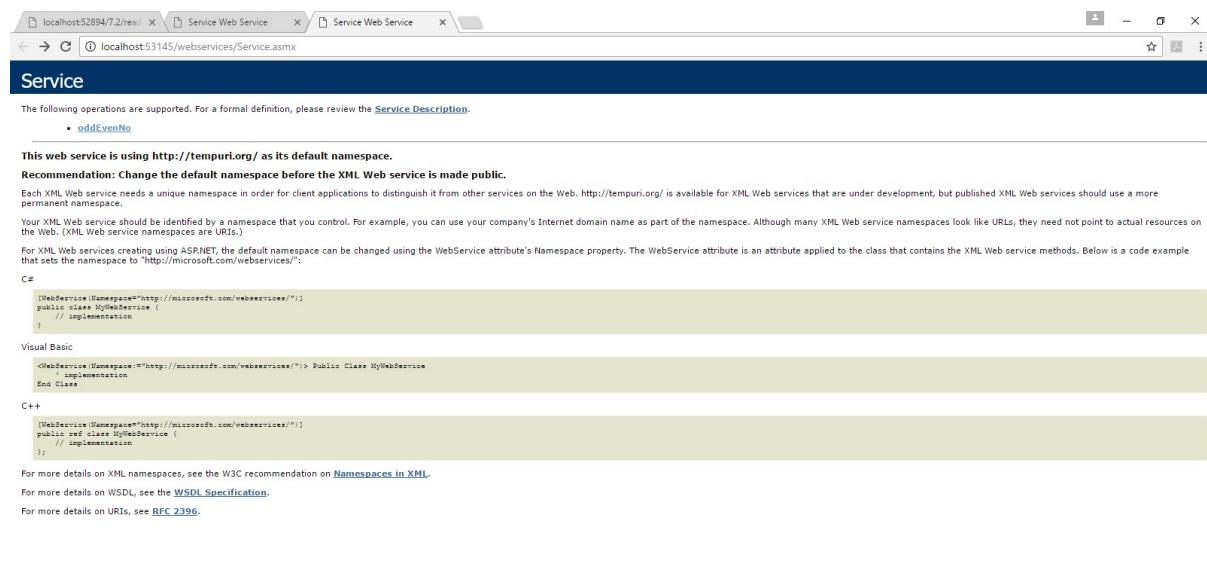
    [WebMethod]
    public string oddEvenNo(int no)
    {
        string str;
        if (no % 2 == 0)
        {
            str = "No is even";
        }
    }
}

```

```
else
{
    str="No is odd";
}
return str;

}
```

## At Run Time: Click On OddEvenNo



## Click On Invoke

## Subject :- ASP.NET

The screenshot shows a web browser window with the title "Service Web Service". The URL is "localhost:53145/webservices/Service.asmx?op=oddEvenNo". The page displays a "Test" section for the "oddEvenNo" operation. It includes a parameter input field for "no:" and an "Invoke" button. Below this is a "SOAP 1.1" section containing sample request and response XML. The response XML is highlighted in green. At the bottom, there is another "SOAP 1.2" section with its own sample XML.

```
POST /webservices/Service.asmx HTTP/1.1
Host: localhost
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://tempuri.org/oddEvenNo"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
<soap:Body>
<oddEvenNo xmlns="http://tempuri.org/">
<no>int</no>
</oddEvenNo>
</soap:Body>
</soap:Envelope>

HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
<soap:Body>
<oddEvenNoResponse xmlns="http://tempuri.org/">
<oddEvenNoResult>
</oddEvenNoResult>
</oddEvenNoResponse>
</soap:Body>
</soap:Envelope>
```

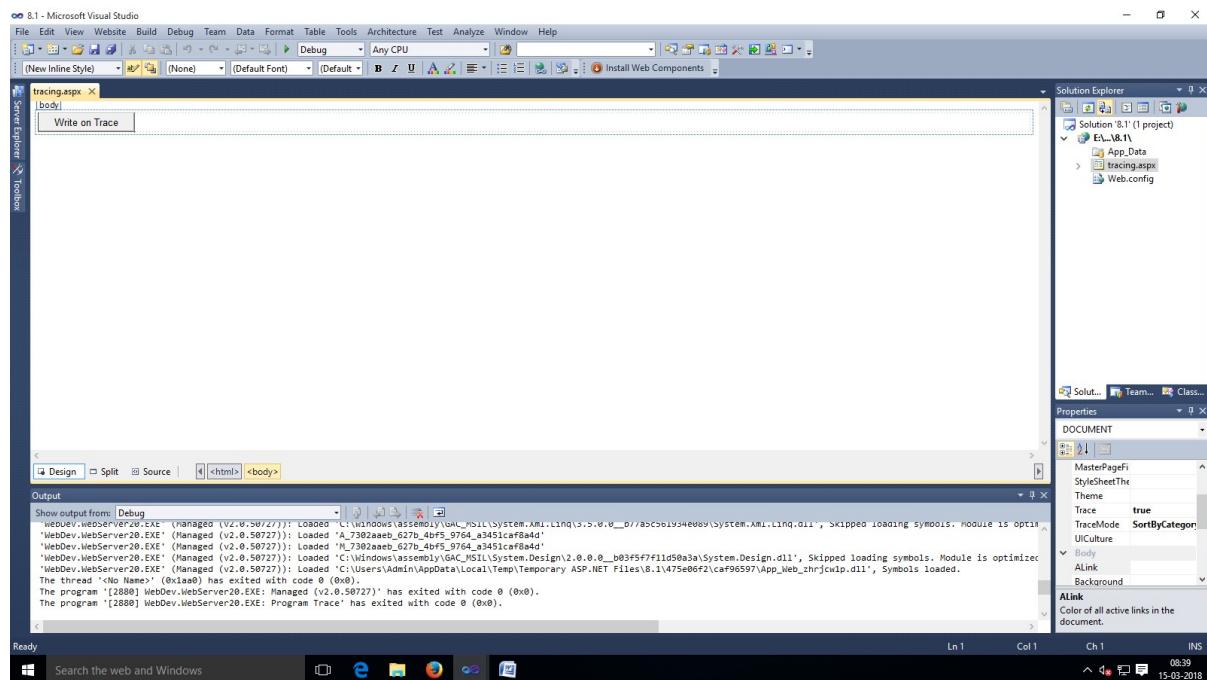
The screenshot shows a web browser window with the title "Service Web Service". The URL is "localhost:53145/webservices/Service.asmx/oddEvenNo". The page displays a message stating "This XML file does not appear to have any style information associated with it. The document tree is shown below." Below this is the XML response from the previous screenshot.

```
<string xmlns="http://tempuri.org/">No is even</string>
```

## 5.4) Example of tracing

**At Design Time:**

### Tracing.aspx



### Tracing.aspx.cs

```
using System;
using System.Collections;
using System.Configuration;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.HtmlControls;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Xml.Linq;

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

    }

    protected void btnwriteontrace_Click(object sender, EventArgs e)
    {
        if(Trace.IsEnabled)
        {
            Trace.Write("Trace Write Method Called");
            Trace.Warn("trace warn method called");
            Response.Write("trace example page called");
        }
    }
}
```

}

## OutPut:

The screenshot shows a browser window with the URL `localhost:1593/8.1/tracing.aspx`. The page displays tracing information for a request. At the top, there's a 'Request Details' section with session ID, time of request, and encoding information. Below it is a 'Trace Information' table showing a list of events with their start and end times. At the bottom is a 'Control Tree' table showing the hierarchy and state of various controls on the page. The status bar at the bottom right indicates the date and time as 15-03-2018 08:39.

Category	Message	From First(s)	From Last(s)
aspx.page	Begin PreInit	5.97332211941661E-05	0.000060
aspx.page	End PreInit	0.000104106471224118	0.000044
aspx.page	Begin Init	0.000158178369327675	0.000011
aspx.page	End Init	0.000244052875164736	0.000023
aspx.page	Begin InitComplete	0.000267263498257326	0.000023
aspx.page	End InitComplete	0.0002880847925021	0.000023
aspx.page	Begin PreLoad	0.000310271417517126	0.000021
aspx.page	End PreLoad	0.000335330320500000	0.000022
aspx.page	Begin Load	0.000353100000000000	0.000022
aspx.page	End Load	0.000363100000000000	0.000022
aspx.page	Begin LoadComplete	0.000376831292562053	0.000021
aspx.page	End LoadComplete	0.0003933525219153	0.000020
aspx.page	Begin PreRender	0.000450217821457743	0.000022
aspx.page	End PreRender	0.000478207102245867	0.000028
aspx.page	Begin PreRenderComplete	0.000499711061875766	0.000022
aspx.page	End PreRenderComplete	0.000519930000000000	0.000026
aspx.page	Begin SaveState	0.0009233048933115538	0.000013
aspx.page	End SaveState	0.0009642548564201	0.000041
aspx.page	Begin SaveStateComplete	0.0703828012013545	0.069419
aspx.page	End SaveStateComplete	0.0704179584646866	0.000035
aspx.page	Begin Render	0.0706132007688178	0.000195
aspx.page	End Render	0.0706132007688178	0.000195

Control UniqueID	Type	Render Size Bytes (including children)	ViewState Size Bytes (excluding children)	ControlState Size Bytes (excluding children)
__Page	ASP-tracing_aspx	840	0	0
ct02	System.Web.UI.LiteralControl	174	0	0
ct00	System.Web.UI.HtmlControls.HtmlHead	46	0	0
ct01	System.Web.UI.HtmlControls.HtmlTitle	33	0	0
ct03	System.Web.UI.LiteralControl	14	0	0
form1	System.Web.UI.HtmlControls.HtmlForm	586	0	0
btwnwriteontrace	System.Web.UI.LiteralControl	21	0	0
ct05	System.Web.UI.WebControls.Button	90	0	0
ct06	System.Web.UI.LiteralControl	18	0	0
		20	0	0

---

## 5.5) Example of error handling using custom error

---

**For this application we take two forms.1) errorhandling.aspx 2) pagenotfound.aspx 3)global.asax**

### Global.asax

```
<%@ApplicationLanguage="C#"%>

<script runat="server">

void Application_Start(object sender, EventArgs e)
{
// 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
Exception err = Server.GetLastError();
string errorMsg=<h1>Error in page:</h1>"+Request.Url.ToString()+"<h1>Error
Message:</h1>"+err.Message.ToString()+"Stack trace:" +err.StackTrace.ToString();
Response.Write(errorMsg);
System.Diagnostics.EventLog.WriteEntry("Customer Errors Generated",errorMsg);

}

void Session_Start(object sender, EventArgs e)
{
// Code that runs when a new session is started

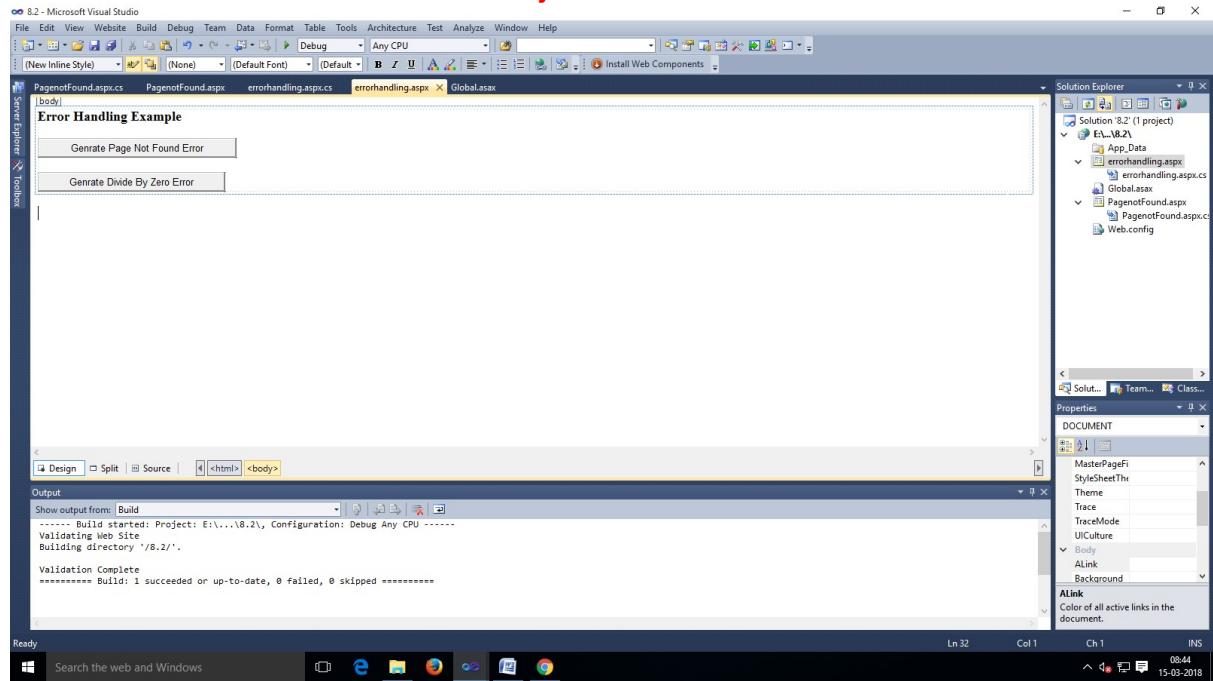
}

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.

}

</script>
```

### Errorhandling



## Errorhandling.aspx.cs

```
using System;
using System.Collections;
using System.Configuration;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.HtmlControls;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Xml.Linq;

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

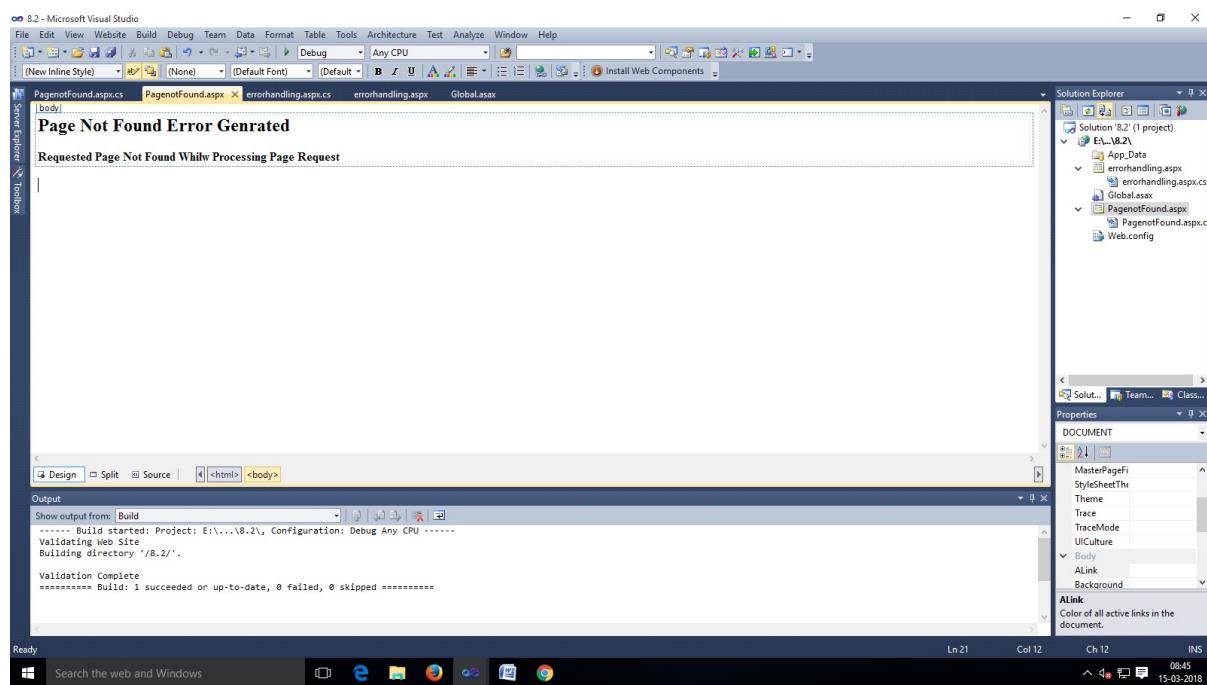
    }

    protected void btndividebyzero_Click(object sender, EventArgs e)
    {
        try
        {
            int a = 10;
            int b = 0;
            int c = a / b;
        }
        catch (Exception ex)
        {
            Response.Write(ex.Message);
        }
    }

    protected void btnpagenotfound_Click(object sender, EventArgs e)
    {
        Response.Redirect("~/aaa.aspx");
    }
}
```

}

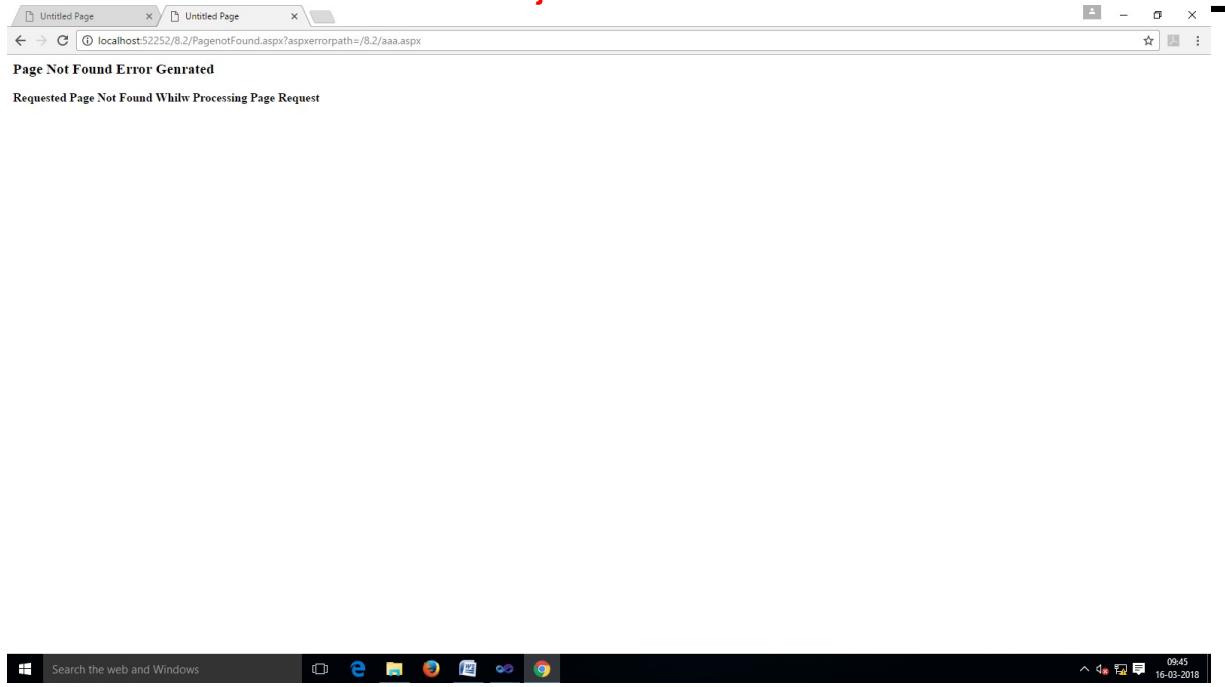
## Pagenotfound



## OutPut

After clicking on generate pagenot found error button

## Subject :- ASP.NET



---

## **After clicking on devideby zero error**



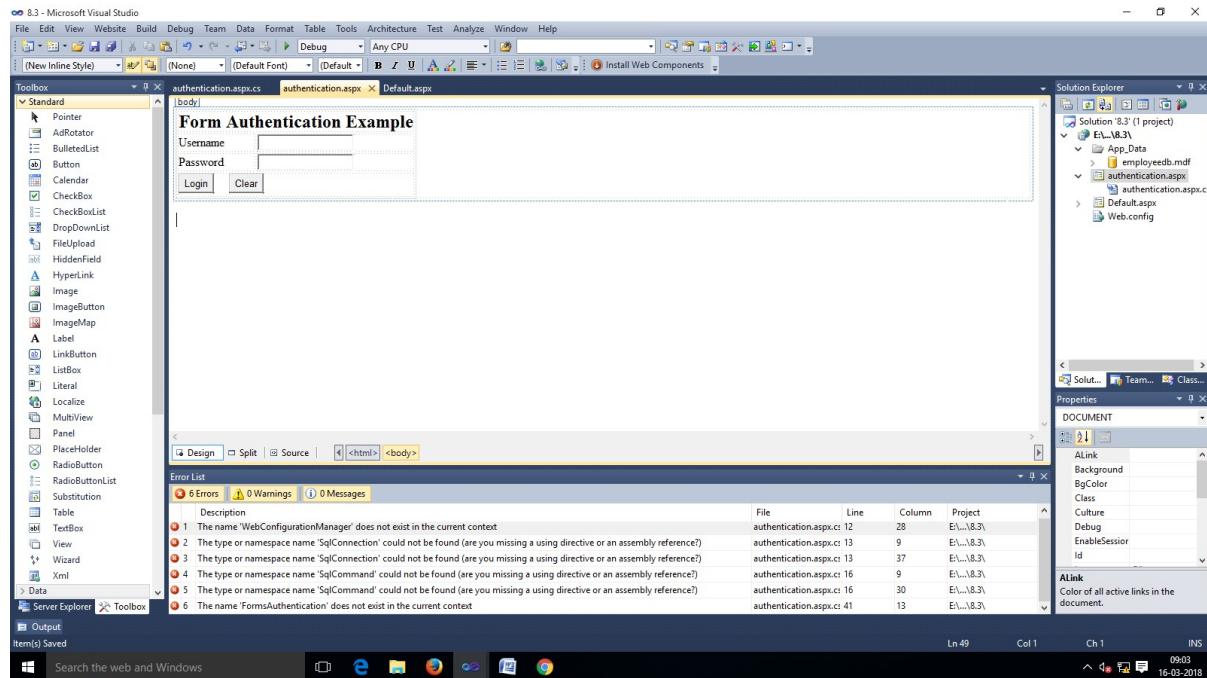
## 5.6) WAP for authentication and authorization

- a. Form authentication
- b. Window authentication

Take two forms (1) authentication.aspx (2) default.aspx

There is no any codeing or designing in default page

At Design Time:



### Authentication.aspx.cs

```
using System;
using System.Collections;
using System.Configuration;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.HtmlControls;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Xml.Linq;
using System.Web.Configuration;
using System.Data.SqlClient;
```

```
public partial class authentication : System.Web.UI.Page
{

```

```
Publicbool chkuser()
{
    string constring =
WebConfigurationManager.ConnectionStrings["ConnectionString"].ConnectionString;
SqlConnection connect = new SqlConnection(constring);
string query = "select * from login where username=@user and password=@password";
connect.Open();
SqlCommand cmd = new SqlCommand(query, connect);
cmd.Parameters.Add("@user",txtusername.Text);
cmd.Parameters.Add("@password",txtpassword.Text);

Object obj = cmd.ExecuteScalar();
connect.Close();

if (obj != null)
{
    Returntrue;
}

else
{
    Returnfalse;
}
Protectedvoid Page_Load(object sender, EventArgs e)
{

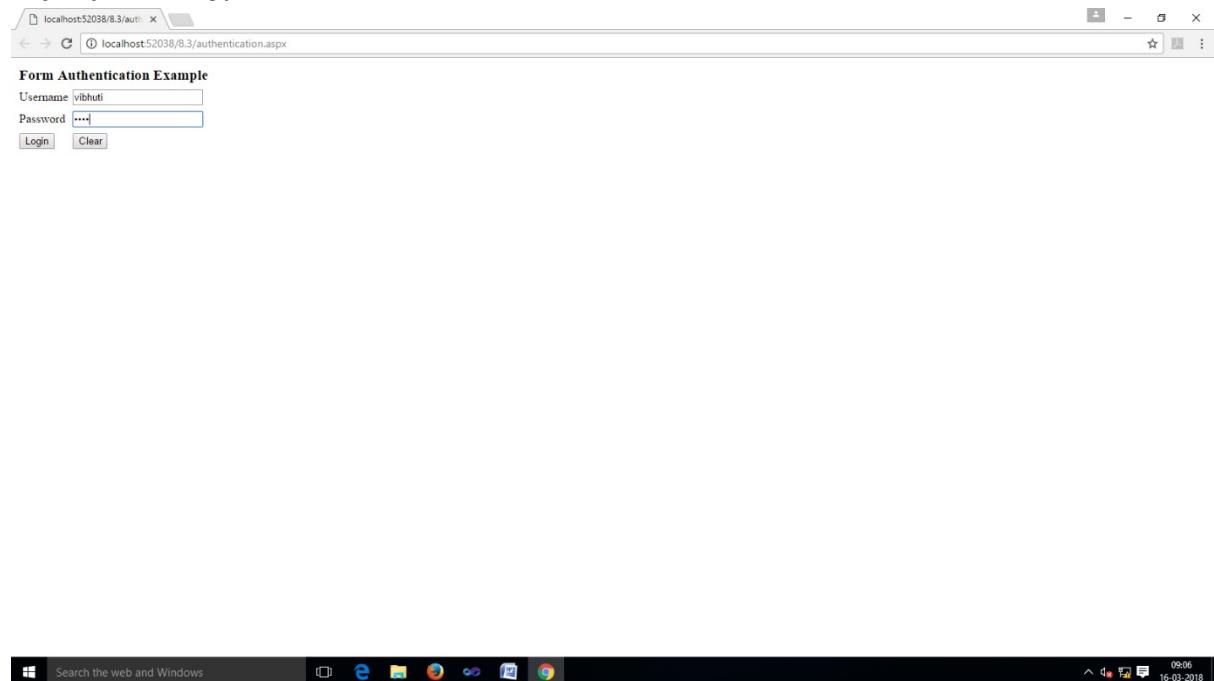
}
protectedvoid btnlogin_Click(object sender, EventArgs e)
{
if (chkuser())
{
FormsAuthentication.RedirectFromLoginPage(txtusername.Text, false);
}
}
protectedvoid btnclear_Click(object sender, EventArgs e)
{
}
}
```

### **Make some changes in web.config file**

```
<authenticationmode="Forms">
    <formsname="Login"loginUrl="authentication.aspx"timeout="120">
</forms>
    </authentication>
    <authorization>
<allowusers="vibhuti"/>
    <denyusers="?">
    </authorization>
```



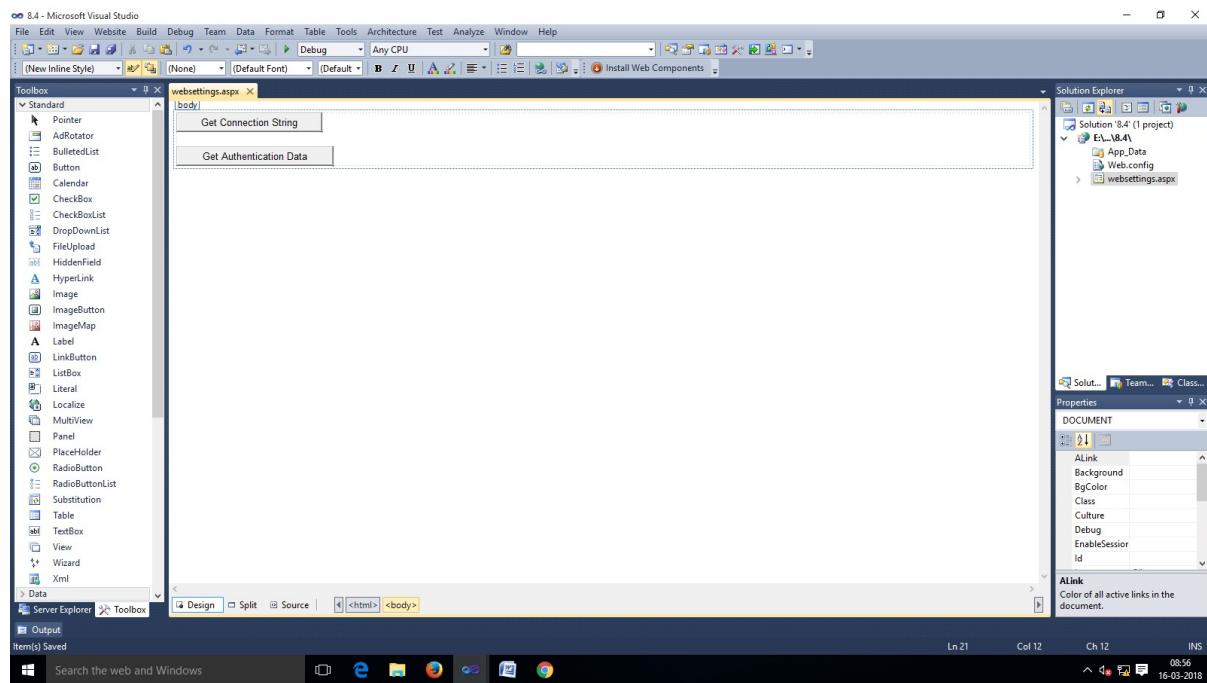
**At Run Time:**



**It Will Redirect In default page if user name and password is same**

## 5.7) Reading settings from web config file.

### At Design Time:



### Websetting.aspx.cs

```

using System;
using System.Collections;
using System.Configuration;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.HtmlControls;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Xml.Linq;
using System.Web.Configuration;

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

    protected void btngtcon_Click(object sender, EventArgs e)
    {
        string constr = WebConfigurationManager.ConnectionStrings["constr"].ConnectionString;
        Response.Write("<p>Connection String is:" + constr);
        Response.Write("<hr>");
    }

    protected void btngtdata_Click(object sender, EventArgs e)
    {
        AuthenticationSection auth =
        (AuthenticationSection)WebConfigurationManager.GetWebApplicationSection("system.web/authentication");
    }
}

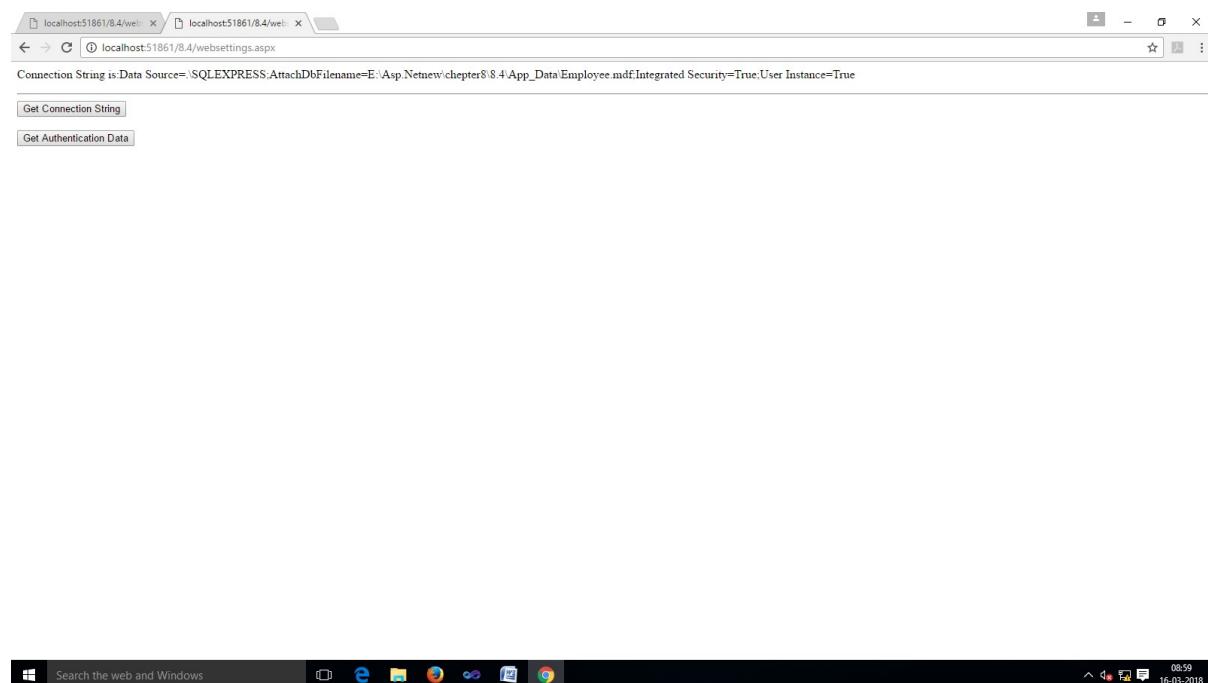
```

## Subject :- ASP.NET

```
Response.Write("<p>Authentication Mode:" + auth.Mode.ToString());
if(auth.Mode.ToString() == "Forms")
{
    Response.Write("<p>Login URL:" + auth.Forms.LoginUrl.ToString());
    Response.Write("<p>Time Out:" + auth.Forms.Timeout.ToString());
    Response.Write("<p>Form Name:" + auth.Forms.Name);
    Response.Write("<hr/>");
}
}
```

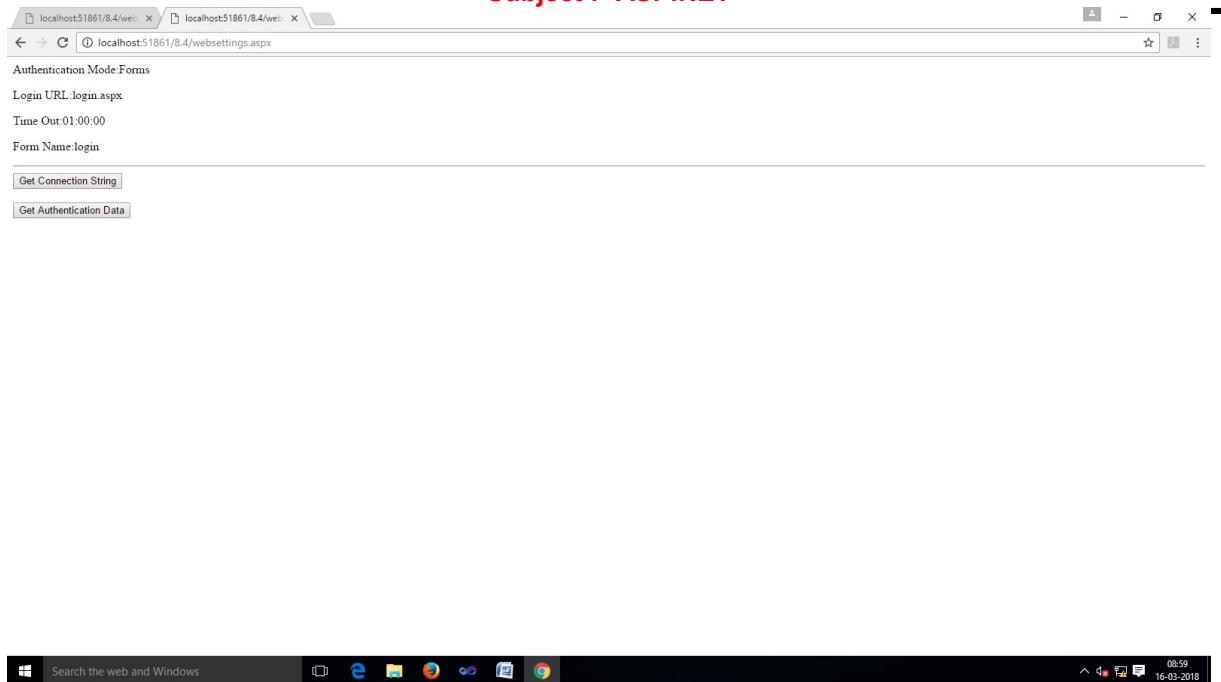
## OutPut

### After clicking on Get Connection String Button



### After clicking on Get Authentication Data Button

## Subject :- ASP.NET

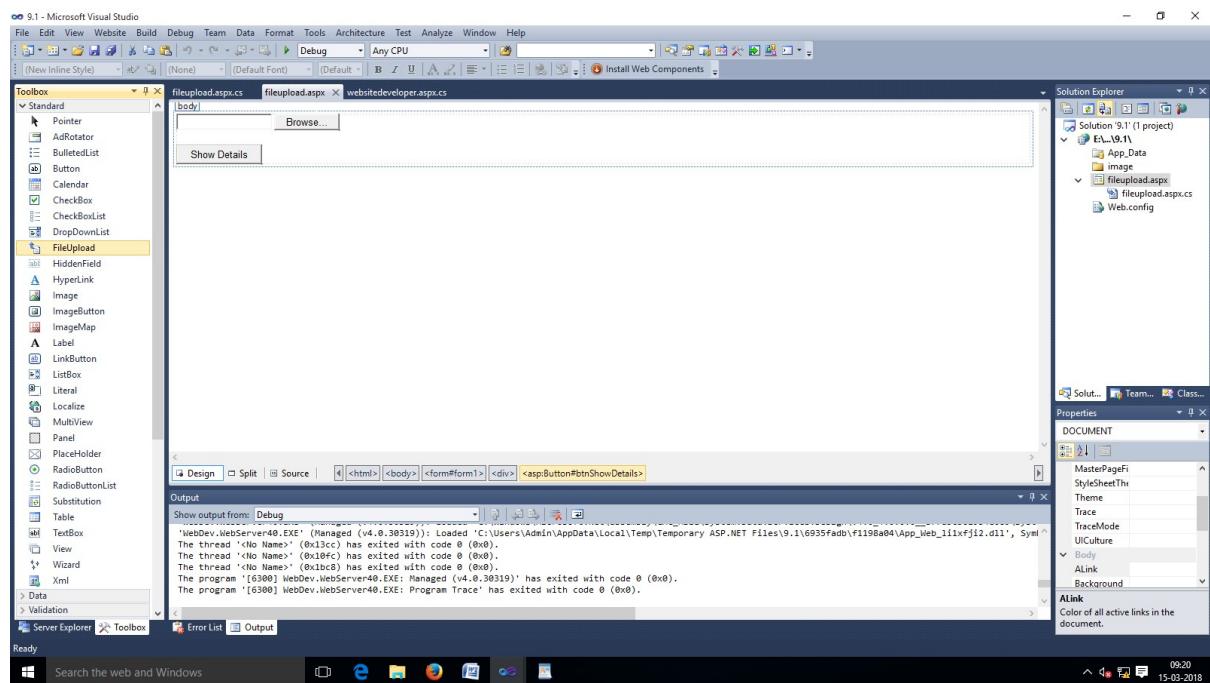


# **Extra Program**

## 1.1) File Upload Control

-At Design Time:-

ASPX File:-



ASPX.CS File:-

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

    }
    protected void btnShowDetails_Click(object sender, EventArgs e)
    {
        Response.Write("<u>Information About Selected File<u>");
        Response.Write("<br> - File Name : " + FileUpload1.FileName);
        Response.Write("<br> - Has File : " + FileUpload1.HasFile);
        Response.Write("<br> - Length : " + FileUpload1.FileBytes.Length);
        Response.Write("<br> - ReadOnly : " + FileUpload1.FileBytes.IsReadOnly);
        Response.Write("<br> - Get Length : " + FileUpload1.FileBytes.GetLength(0));
        Response.Write("<br> - Content Length : " + FileUpload1.FileContent.Length);
        Response.Write("<br> - Can Read : " + FileUpload1.FileContent.CanRead);

        Response.Write("<br> - Can Write : " + FileUpload1.FileContent.CanWrite);
        Response.Write("<br> - Can Seek : " + FileUpload1.FileContent.CanSeek);
    }
}
```

```
HttpPostedFile pf = FileUpload1.PostedFile;
    Response.Write("<br> - Posted File Name : " + pf.FileName);
    Response.Write("<br> - Posted Content Type : " + pf.ContentType);
    Response.Write("<br> - Posted Content Length : " + pf.ContentLength);

string fn;
string ct = pf.ContentType;
if(ct == "image/jpeg" || ct == "image/gif" || ct == "image/pjpeg")
{
if(ct == "image/jpeg" || ct == "image/pjpeg")
{
    fn = MapPath("extra.jpg");
}
else
{
    fn = MapPath("extra.gif");
}
Response.Write("<p> - File Path : " + fn);
FileUpload1.SaveAs(fn);
Response.Write("<p> - Selected File is an Image");
Response.Write("<p><image src = '" + fn + "' width = 500 height = 300/>");
}
elseif(pf.ContentType == "text/plain")
{
    fn = MapPath("extra.txt");
    Response.Write("<p> - File Path : " + fn);
    pf.SaveAs(fn);
    Response.Write("<p> - File Copied");
    Response.Write("<p> - Selected File Contents Are");
StreamReader sr = new StreamReader(fn);
string l;
int i = 1;
    Response.Write("<H2>File Contents Are as Follows</H2>");

while ((l = sr.ReadLine()) != null)
{
    Response.Write("<p> Line - " + i + " : " + l);
    i++;
}
sr.Close();
}
else
{
    Response.Write("<p> - Selected file is not image or text file");
}

}
```

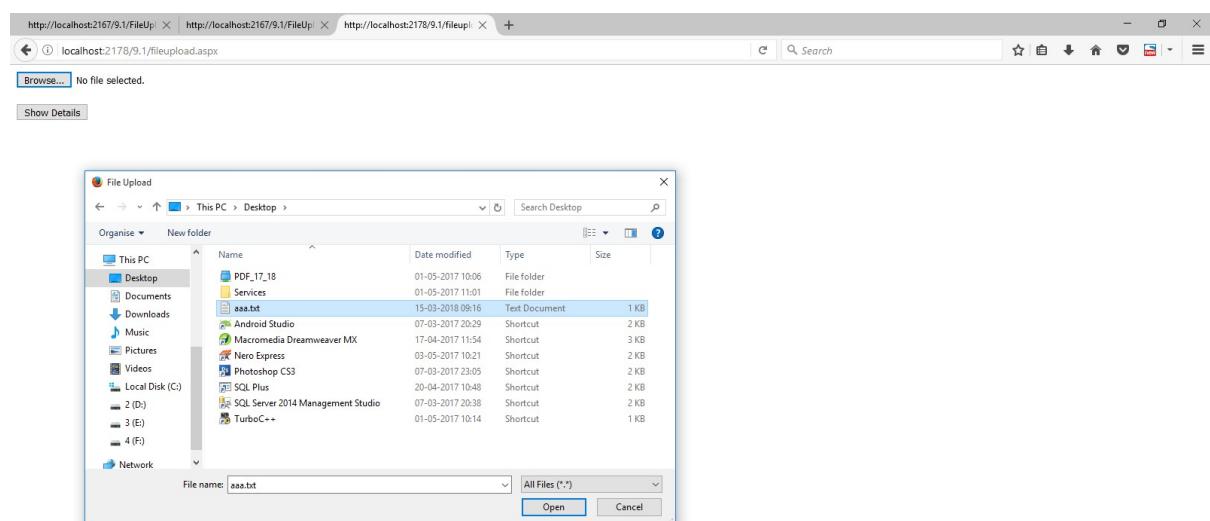
### **At Run Time:-**

1)

## Subject :- ASP.NET



2)



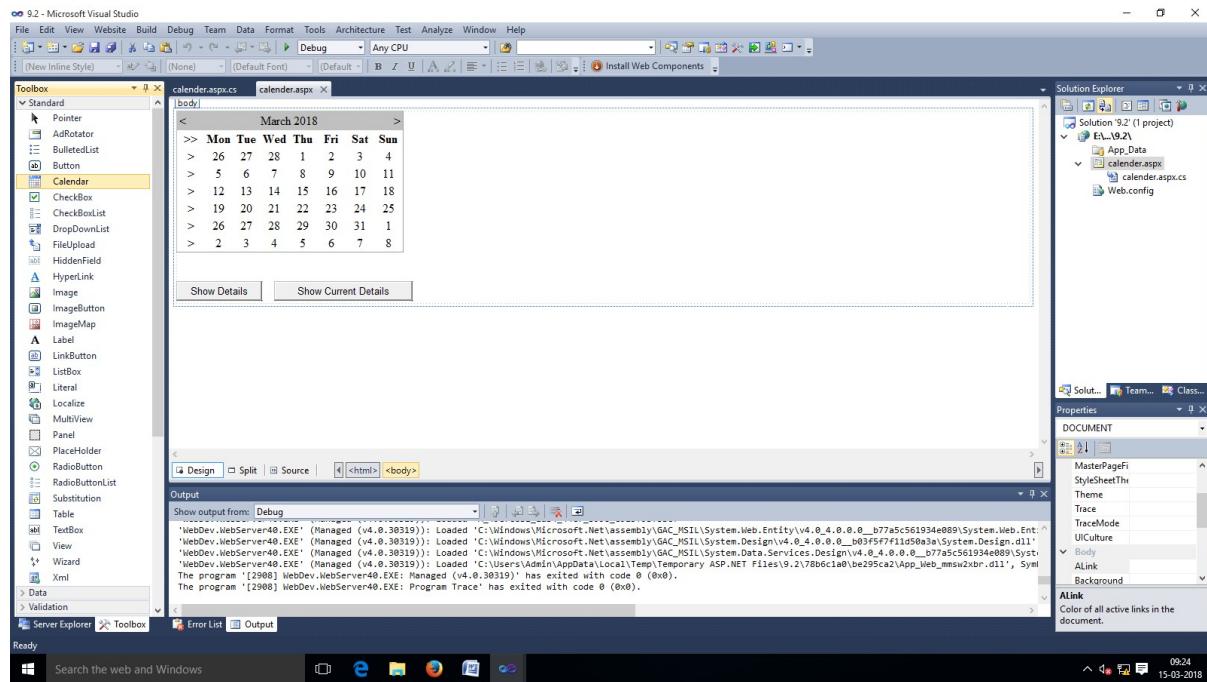
**3) After Clicking Show Details Button**

The screenshot shows a web browser window with three tabs open. The active tab displays "Information About Selected File" for a file named "aaa.txt". The file details include: File Name : aaa.txt, Has File : True, Length : 21, ReadOnly : False, GetLength : 21, ContentLength : 21, CanRead : True, CanWrite : False, CanSeek : True, PostedFileName : aaa.txt, PostedContentType : text/plain, PostedContentLength : 21, FilePath : E:\ASPNET18-19\chapter9\9.1\extra.txt, FileCopied, and SelectedFileContentsAre. Below this, a section titled "File Contents Are as Follows" shows the content "Line - 1 : vala vibhuti vikubhai". A "Show Details" button is visible. At the bottom of the browser window, there is a message: "Browse... No file selected.". The taskbar at the bottom of the screen shows the Windows Start button, a search bar with "Search the web and Windows", and several pinned icons for applications like File Explorer, Edge, and File History. The date and time on the taskbar are 15-03-2018 and 09:21.

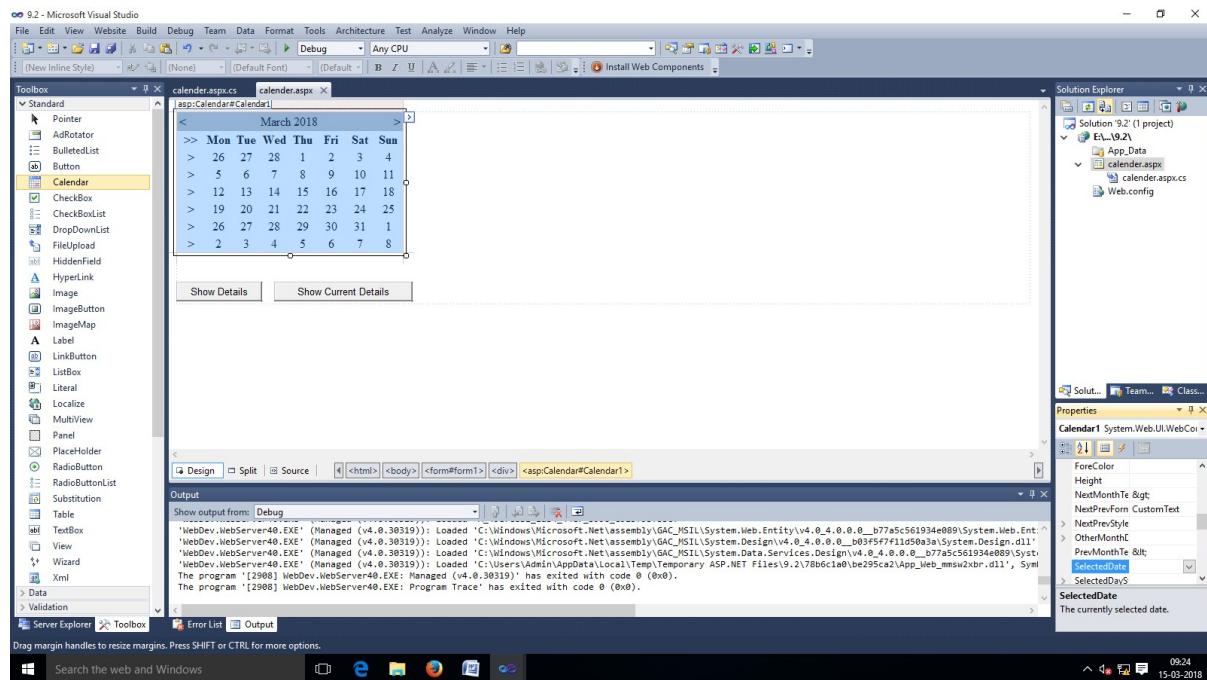
## 1.2) Calendar Control

**At Design Time:-**

**ASPX Page:-**



**Set its property Selection Mode:**



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

    }

    public void ShowDateInformation(DateTime d)
    {
        Response.Write("<p> - Date : " + d.ToString());
        Response.Write("<p> - Short Date : " + d.ToShortDateString());
        Response.Write("<p> - Long Date : " + d.ToString("yyyy-MM-dd"));
        Response.Write("<p> - Day : " + d.Day);
        Response.Write("<p> - Month : " + d.Month);
        Response.Write("<p> - Year : " + d.Year);
        Response.Write("<p> - Day of Week : " + d.DayOfWeek);
        Response.Write("<p> - Day of Year : " + d.DayOfYear);
    }

    protected void btnShowDetails_Click(object sender, EventArgs e)
    {
        //this.ShowDateInformation(Calendar1.SelectedDate);
        if (Calendar1.SelectedDates.Count == 0)
        {
            Response.Write("<p><b><u>No Date Selected</u></b></p>");
        }
        else if (Calendar1.SelectedDates.Count == 1)
        {
            Response.Write("Single Dates Selected");
        }
        else if (Calendar1.SelectedDates.Count > 1)
        {
            Response.Write("Multiple Date Selected");
        }
        DateTime d;
        for (int i = 0; i < Calendar1.SelectedDates.Count; i++)
        {
            d = Calendar1.SelectedDates[i];
            this.ShowDateInformation(d);
        }
    }

    protected void btnShowCurrent_Click(object sender, EventArgs e)
    {
        DateTime d = DateTime.Now;
        Response.Write("<p> - Current Time " + d.Hour + ":" + d.Minute + ":" + d.Second);
        ShowDateInformation(d);
    }
}
```

}

### **At Run Time:-**

For clicking Show Details button you have to select date from calendar. You can select single or multiple dates.

**After clicking Show Detail button O/p will be like this**

The screenshot shows a web browser window with the URL <http://localhost:9108/9.2/calender.aspx>. The page displays a calendar for March 2018. A date, March 14, 2018, is highlighted with a gray background. Below the calendar, there is a list of selected date details:

- Date : 14-03-2018 00:00:00
- Short Date : 14-03-2018
- Long Date : 14 March 2018
- Day : 14
- Month : 3
- Year : 2018
- Day of Week : Wednesday
- Day of Year : 73

At the bottom of the calendar, there are two buttons: "Show Details" and "Show Current Details".



**After clicking Show Current Detail button**

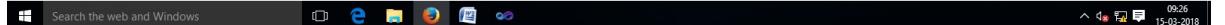
## Subject :- ASP.NET

http://localhost:9108/9.2/calender javascript\_doPostBack('Calend... X +

- Current Time 9:26:18  
- Date : 15-03-2018 09:26:18  
- Short Date : 15-03-2018  
- Long Date : 15 March 2018  
- Day : 15  
- Month : 3  
- Year : 2018  
- Day of Week : Thursday  
- Day of Year : 74

March 2018						
<	>>	Mon	Tue	Wed	Thu	Fri
>	<<	26	27	28	1	2
>	<	3	4	5	6	7
>	<	9	10	11	12	13
>	<	14	15	16	17	18
>	<	19	20	21	22	23
>	<	24	25	26	27	28
>	<	29	30	31	1	2
>	<	3	4	5	6	7
>	<	8				

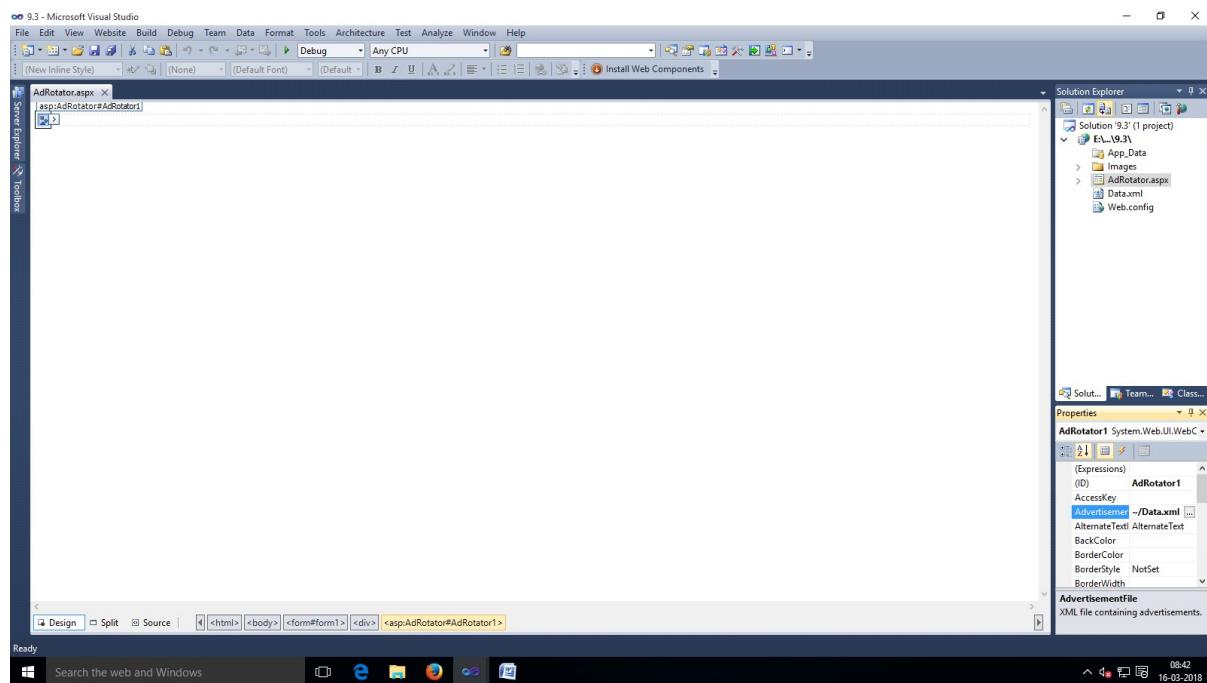
[Show Details](#) [Show Current Details](#)



## 1.3) AdRotator Control

**At Design Time:-**

ASPX Page:-



By this control user can set images. We have to create one XML file & set Advertisement file of AdRotator. Create Image folder in root directory of this application & put images which you want to display. In xml file you have to set path in ImageUrl tag.

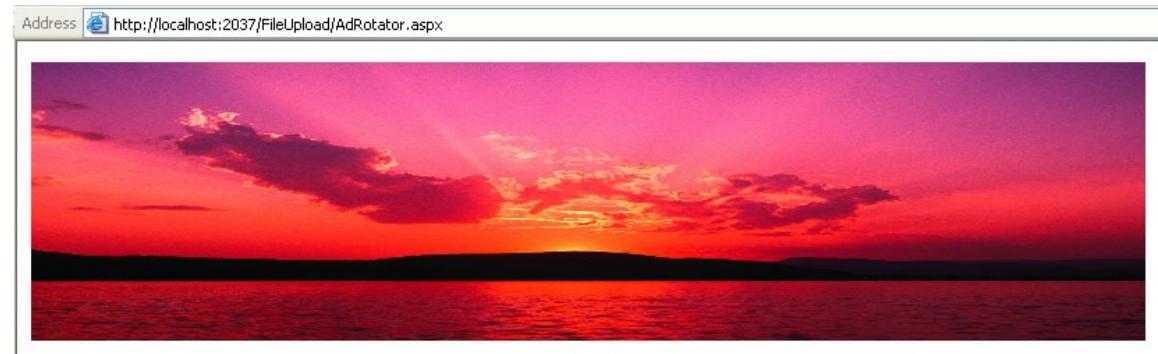
Data.xml file:-

```
<?xml version="1.0" encoding="utf-8" ?>
<Advertisements>

<Ad>
<ImageUrl>Images/Sunset.jpg</ImageUrl>
<Width>800</Width>
<Height>200</Height>
<AlternateText>Windows Logo</AlternateText>
<NavigateUrl>http://support.microsoft.com</NavigateUrl>
<keyword>Microsoft</keyword>
<Impressions>5</Impressions>
</Ad>
<Ad>
<ImageUrl>Images/Winter.jpg</ImageUrl>
<Width>800</Width>
<Height>200</Height>
<AlternateText>Sun Logo</AlternateText>
<NavigateUrl>http://www.sun.java.com</NavigateUrl>
<keyword>Microsoft</keyword>
<Impressions>5</Impressions>
</Ad>

</Advertisements>
```

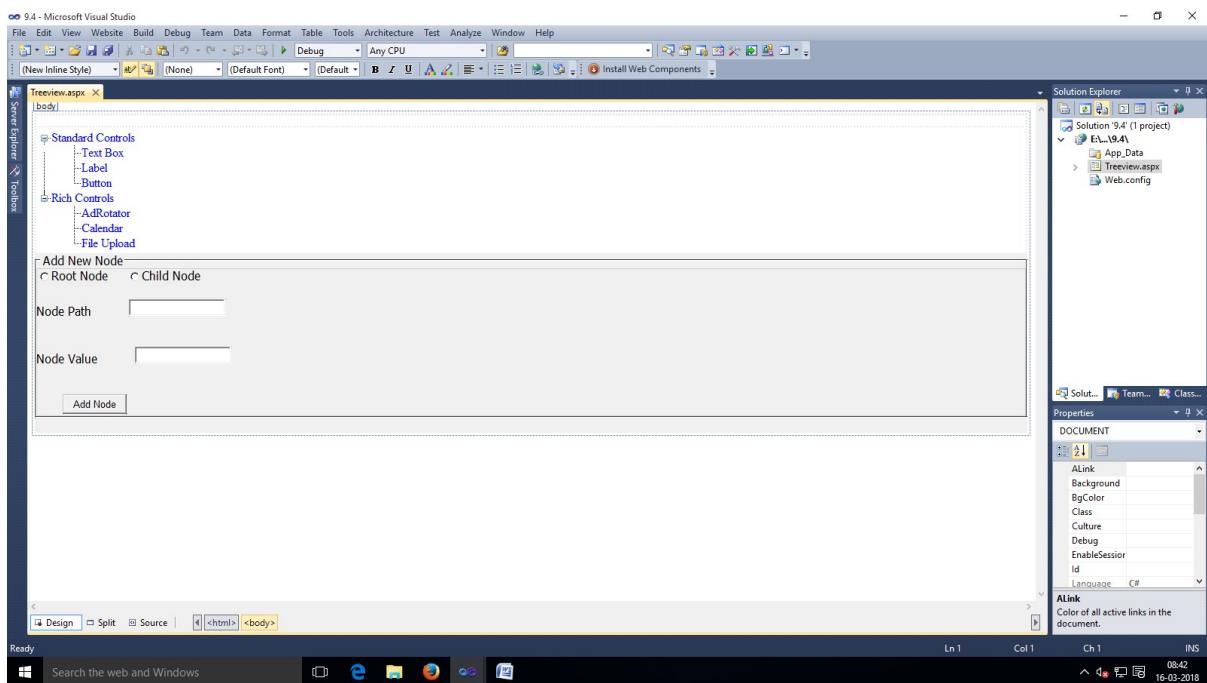
**At Run Time:-**



## 1.4)TreeView Control:-

At Design Time

ASPX Page:-



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)
    {
    }

    protected void TV_SelectedNodeChanged(object sender, EventArgs e)
    {
        if (TV.SelectedNode == null)
        {
            Response.Write("<p>Select Any Node to view Node Details");
            Response.Write("<hr>");
        }
    }

    else
    {
        Response.Write("<b>Selected Node Details </b><br>");
        Response.Write("<br> - Node Text : " + TV.SelectedNode.Text);
        Response.Write("<br> - Node Value : " + TV.SelectedNode.Value);
        Response.Write("<br> - Node Selected : " + TV.SelectedNode.Selected);
        Response.Write("<br> - Node Expanded : " + TV.SelectedNode.Expanded);
    }
}
```

## Subject :- ASP.NET

---

```
Response.Write("<br> - No. of Child Nodes : " + TV.SelectedNode.ChildNodes.Count);
for (int i = 0; i < TV.SelectedNode.ChildNodes.Count; i++)
{
    Response.Write("<br> Child : " + TV.SelectedNode.ChildNodes[i].Text);
}
Response.Write("<hr>");
}

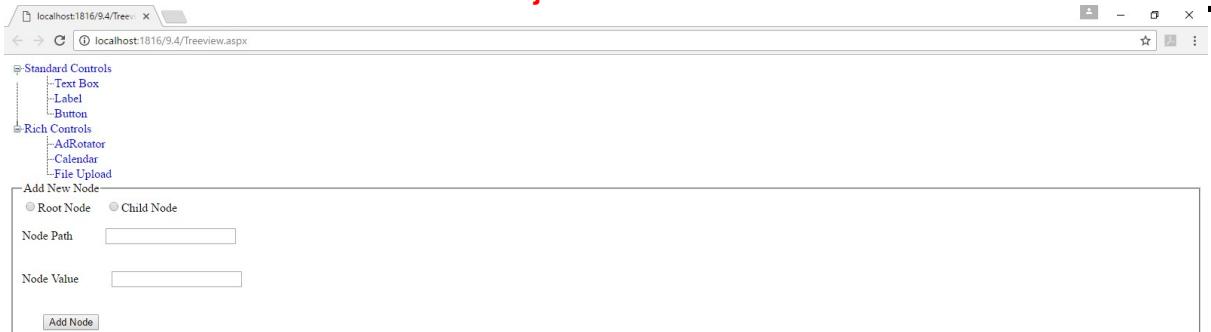
protectedvoid btn_RootNode_CheckedChanged(object sender, EventArgs e)
{
if(btn_RootNode.Checked == true)
    txt_NodePath.Enabled = false;
}

protectedvoid btn_ChildNode_CheckedChanged(object sender, EventArgs e)
{
if(btn_ChildNode.Checked == true)
    txt_NodePath.Enabled = true;
}

protectedvoid btn_AddNode_Click(object sender, EventArgs e)
{
if(btn_RootNode.Checked==false&& btn_ChildNode.Checked==false)
{
    Response.Write("<p> You must select ROOT NODE or CHILD NODE type");
    Response.Write("<hr>");
}
else
{
    TreeNode t=newTreeNode(txt_NodeValue.Text,txt_NodeValue.Text);
    if(btn_RootNode.Checked==true)
    {
        TV.Nodes.Add(t);
        Response.Write("<p>Root Node Added");
        Response.Write("<hr>");
    }
    elseif(TV.FindNode(txt_NodePath.Text)==null)
    {
        Response.Write("<p>Invalid Node Path");
        Response.Write("<hr>");
    }
    elseif(btn_ChildNode.Checked==true)
    {
        TV.FindNode(txt_NodePath.Text).ChildNodes.Add(t);
        Response.Write("<p>Child Node Added under "+txt_NodePath.Text);
        Response.Write("<hr>");
    }
}
}
}
```

**At Run Time:-**

## Subject :- ASP.NET



To add Node you have given two options Root or Child. Select as you need. If you want to add Root Node there is no need to write Node Path.

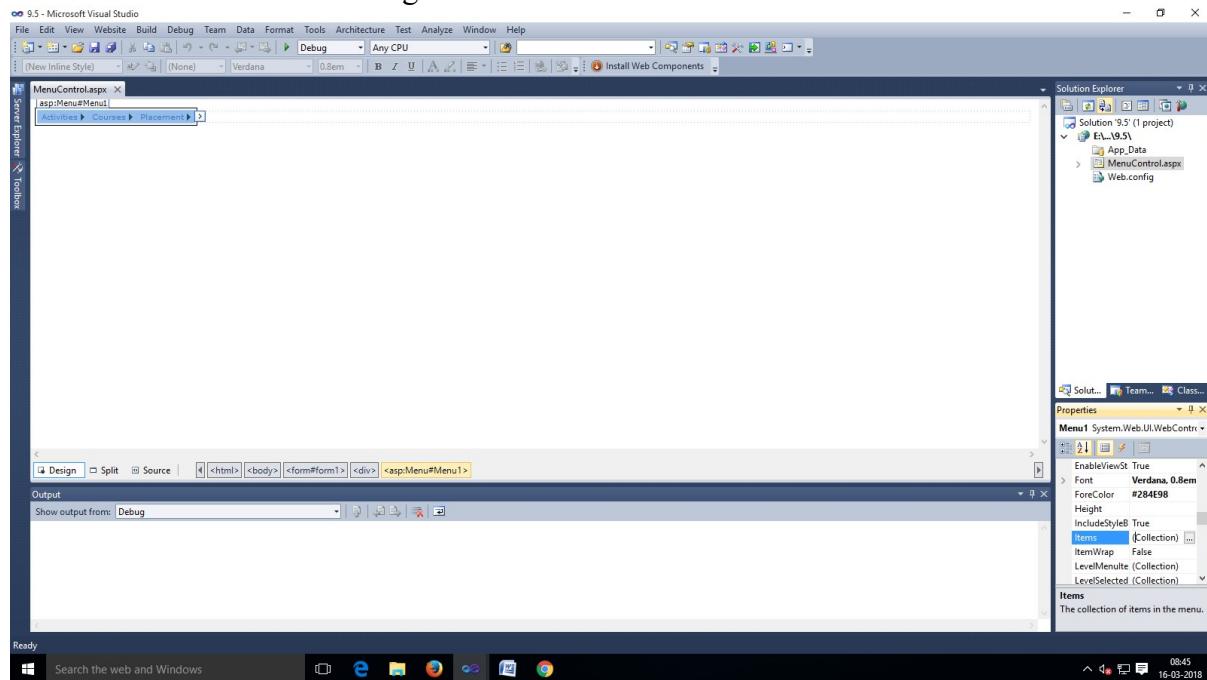


## 1.5) Menu Control

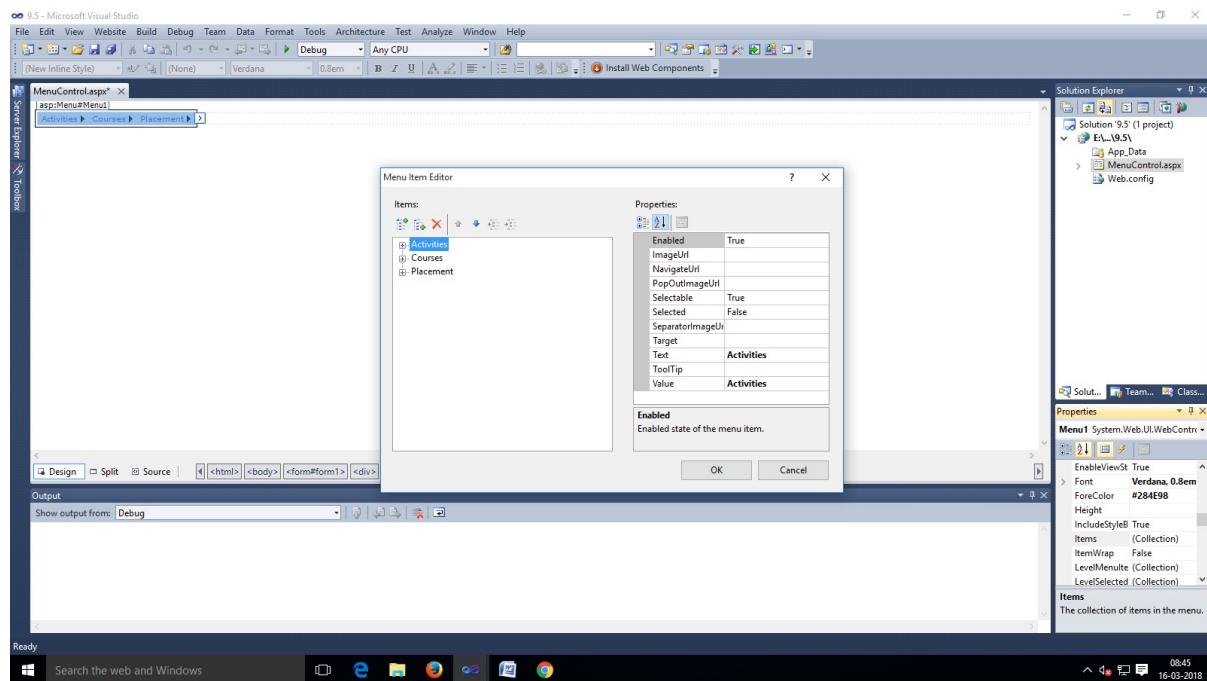
-At Design Time

ASPX Page:-

Take Menu Control from Navigation Control.



Here you see menu control is vertical you can change its orientation horizontal from its properties. To add Items from its properties you can add, delete Parent & child items.



ASPX.CS Page:-

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

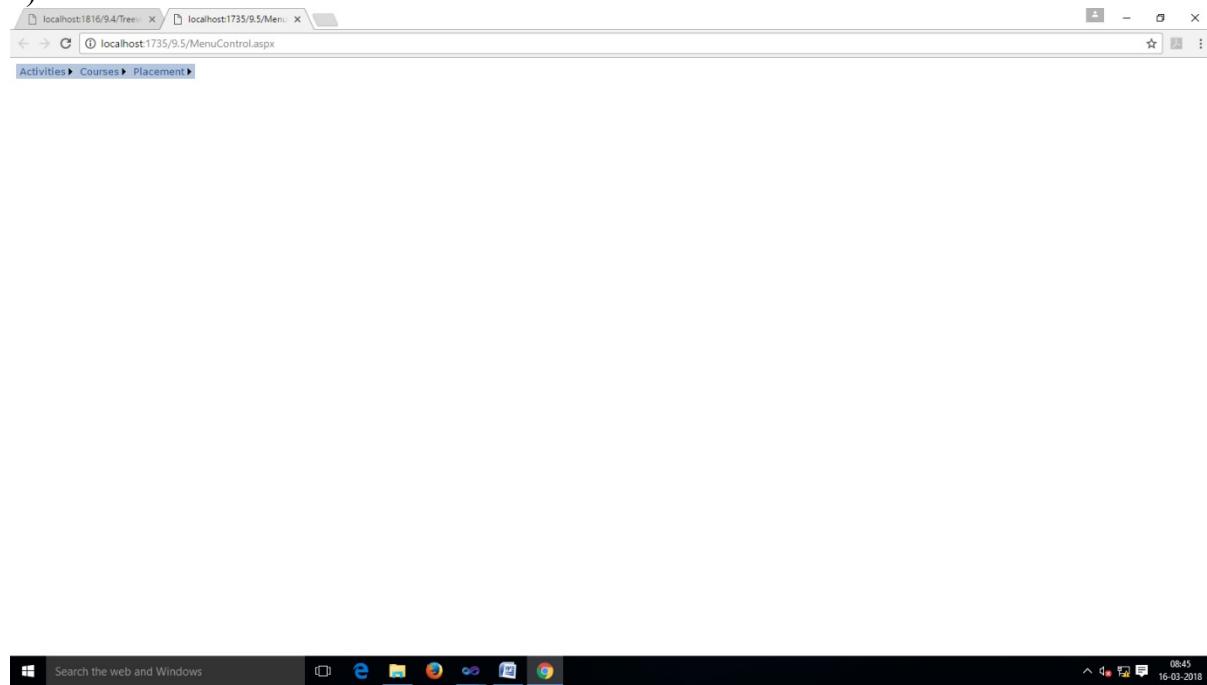
public partial class _Default : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {

    }

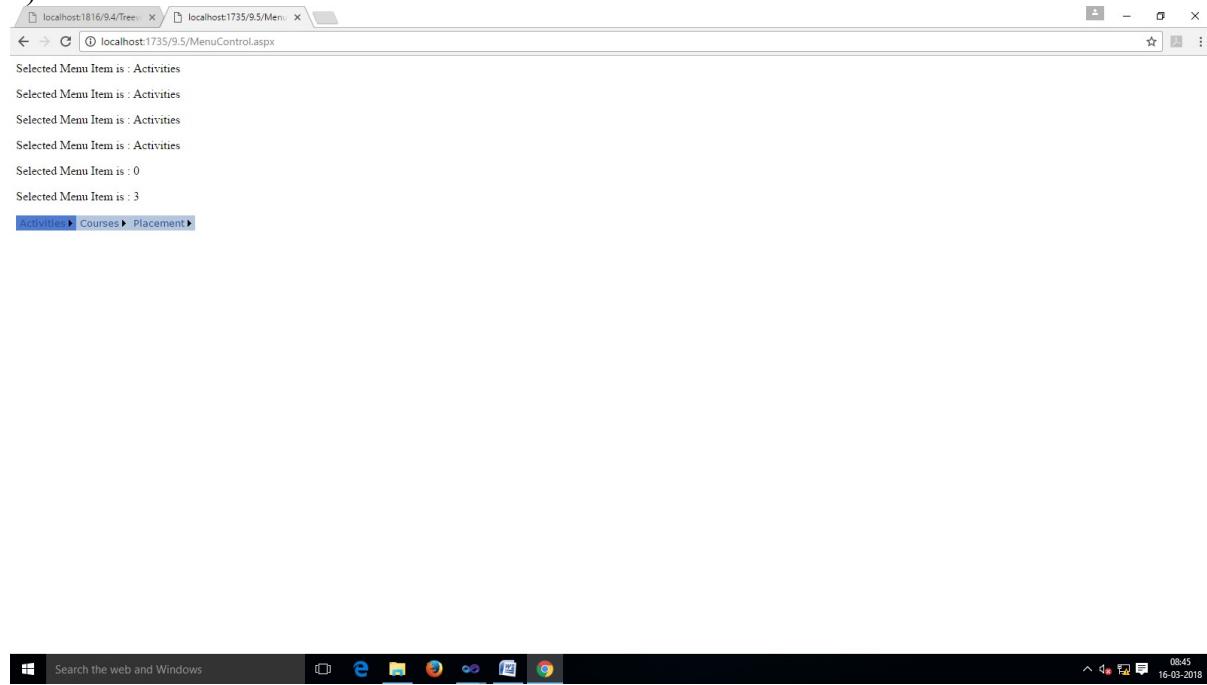
    protected void Menu1_MenuItemClick(object sender, MenuEventArgs e)
    {
        Response.Write("<p>Selected Menu Item is:" + e.Item.Text);
        Response.Write("<p>Selected Menu Item is:" + Menu1.SelectedItem.Text);
        Response.Write("<p>Selected Value:" + Menu1.SelectedValue);
        Response.Write("<p>Selected Value Path:" + Menu1.SelectedItem.ValuePath);
        Response.Write("<p>Selected Menu Depth:" + Menu1.SelectedItem.Depth);
        Response.Write("<p>No. of child Items:" + Menu1.SelectedItem.ChildItems.Count);
        if (Menu1.SelectedItem.Parent == null)
            Response.Write("<p>Its Root Menu");
        else
            Response.Write("<p>Parent Menu Item:" + Menu1.SelectedItem.Parent.Text);
        Menu1.Items.Add(newMenuItem("ABC"));
        if (Menu1.SelectedItem.ValuePath == "Courses/IT")
        {
            if (Menu1.FindItem("Courses/IT/MCA") == null)
            {
                MenuItem m = newMenuItem("MCA", "MCA");
                Menu1.SelectedItem.ChildItems.Add(m);
            }
        }
    }
}
```

**At Run Time:-**

1)



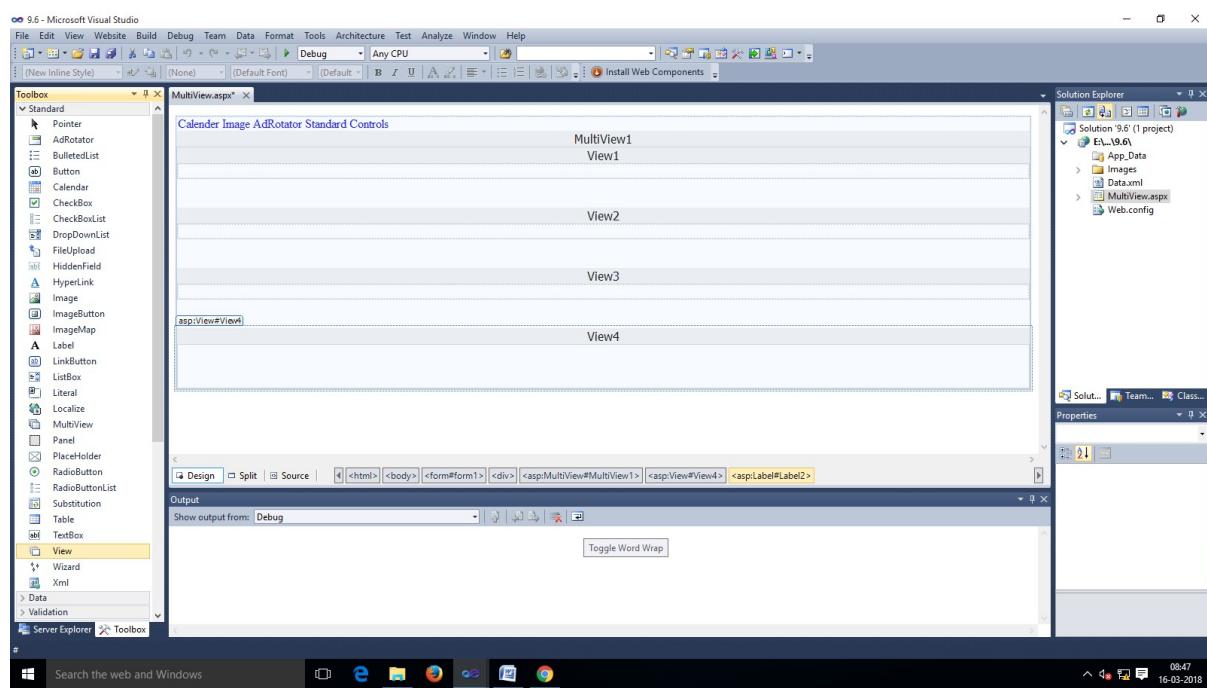
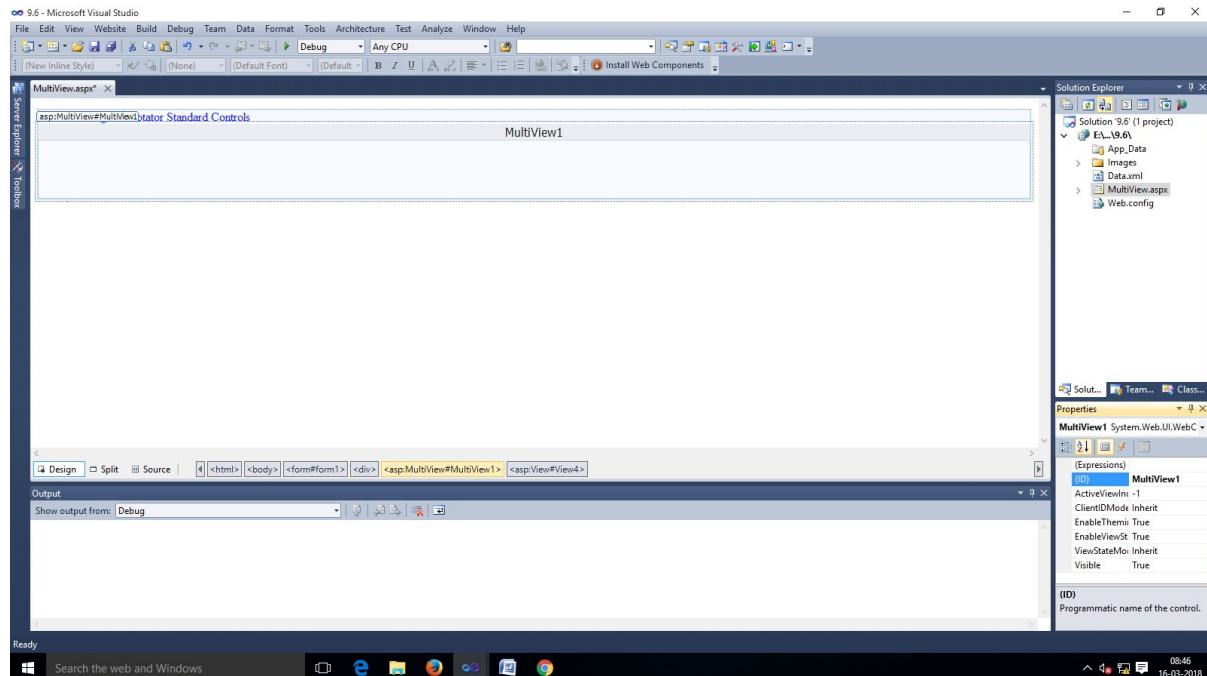
2)



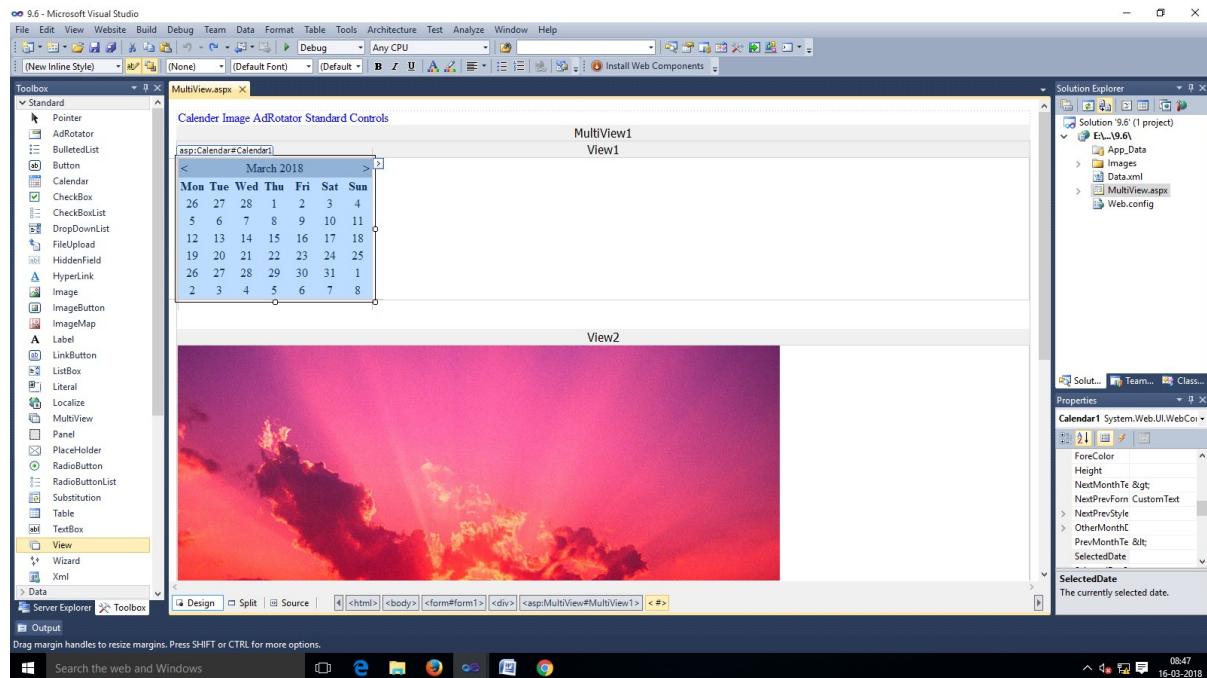
## **1.6) Multiview Control:-**

- At Design Time

### ASPX Page:-



Place Menu at the top & put different controls in Views. We have place calendar, Image, Adrotator & Standard Controls in views respectively.

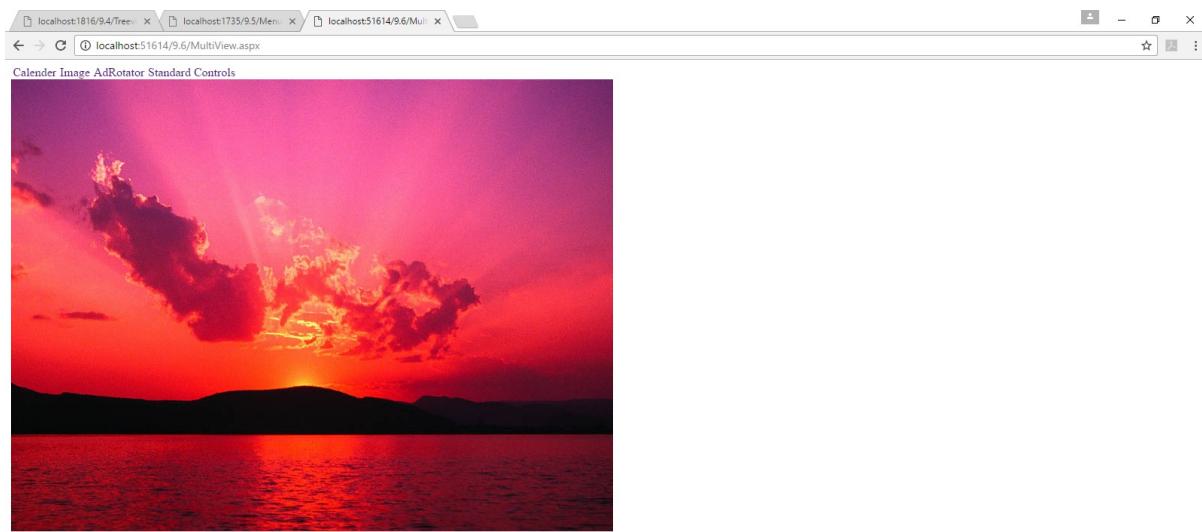


### ASPX.CS Page:-

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

public partial class MultiView : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        MultiView1.ActiveViewIndex = 0;
    }
    protected void Menu1_MenuItemClick(object sender, MenuEventArgs e)
    {
        MultiView1.ActiveViewIndex = Convert.ToInt32(e.Item.Value);
    }
}
```

**At Run Time:-**



## **1.7) Wizard Control:-**

- At Design Time:-

### ASPX (Source) File:-

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

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

<htmlxmlns="http://www.w3.org/1999/xhtml">
<headrunat="server">
<title></title>
<styletype="text/css">
#form1
{
height: 259px;
}
</style>
</head>
<body>
<formid="form1"runat="server">
<div>

<asp:WizardID="Wizard1"runat="server"ActiveStepIndex="0"BackColor="#F7F6F3"
BorderColor="#CCCCCC"BorderStyle="Solid"BorderWidth="1px"
Font-Names="Verdana"Font-Size="0.8em"Height="225px"
onfinishbuttonclick="Wizard1_FinishButtonClick"Width="391px">
<StepStyleBorderWidth="0px"ForeColor="#5D7B9D"/>
<WizardSteps>
<asp:WizardSteprunat="server"StepType="Start"title="Personal Information">
<asp:LabelID="Label1"runat="server"Text="Enter Your Personal Information"></asp:Label>
<br/>
<br/>
<asp:LabelID="lblname"runat="server"Text="Name : "></asp:Label>
&nbsp;&nbsp;
<asp:TextBoxID="txtname"runat="server"></asp:TextBox>
<br/>
<br/>
<asp:LabelID="lblage"runat="server"Text="Age : "></asp:Label>
&nbsp;&nbsp;&nbsp;&nbsp;
<asp:TextBoxID="txtage"runat="server"></asp:TextBox>
<br/>
<br/>
<asp:LabelID="lblsex"runat="server"Text="Sex : "></asp:Label>
&nbsp;&nbsp;&nbsp;&nbsp;
<asp:TextBoxID="txtsex"runat="server"OnTextChanged="TextBox3_TextChanged"></asp:TextBox>
<br/>
<br/>
<asp:LabelID="lblcourse"runat="server"Text="Course : "></asp:Label>
&nbsp;<asp:TextBoxID="txtcourse"runat="server"
OnTextChanged="TextBox4_TextChanged"></asp:TextBox>
</asp:WizardStep>

<asp:WizardSteprunat="server"StepType="Step"title="Question-1">
<asp:LabelID="lblque1"runat="server"
Text="Question 1: Which of the following control works on XML File?"></asp:Label>
<br/>
<br/>
<asp:RadioButtonID="rbupload"runat="server"Text="File Upload"/>
```

```
<br/>
<br/>
<asp:RadioButtonID="rbcal"runat="server"Text="Calendar"/>
<br/>
<br/>
<asp:RadioButtonID="rbad"runat="server"
OnCheckedChanged="RadioButton3_CheckedChanged"Text="AdRotator"/>
<br/>
<br/>
<asp:RadioButtonID="rbmulti"runat="server"Text="MultiView"/>
</asp:WizardStep>
<asp:WizardStep runat="server" StepType="Step" Title="Question-2">
<asp:LabelID="lblque2"runat="server"
Text="Question 2 : Which is the default view of Menu Control in ASP.NET?"></asp:Label>
<br/>
<br/>
<asp:RadioButtonID="rbhor"runat="server"Text="Horizontal"/>
<br/>
<br/>
<asp:RadioButtonID="rbvert"runat="server"Text="Vertical"/>
<br/>
<br/>
<asp:RadioButtonID="rbdef"runat="server"Text="Default"/>
<br/>
<br/>
<asp:RadioButtonID="rbnone"runat="server"Text="None of the above"/>
</asp:WizardStep>
<asp:WizardStep runat="server" StepType="Step" Title="Question-3">
<asp:LabelID="lblque3"runat="server"
Text="Question 3 : Which of the following is NOT SelectionMode of Calendar?"></asp:Label>
<br/>
<br/>
<asp:RadioButtonID="rbdy"runat="server"Text="Day"/>
<br/>
<br/>
<asp:RadioButtonID="rbdyweek"runat="server"Text="DayWeek"/>
<br/>
<br/>
<asp:RadioButtonID="rbdwm"runat="server"Text="DayWeekMonth"/>
<br/>
<br/>

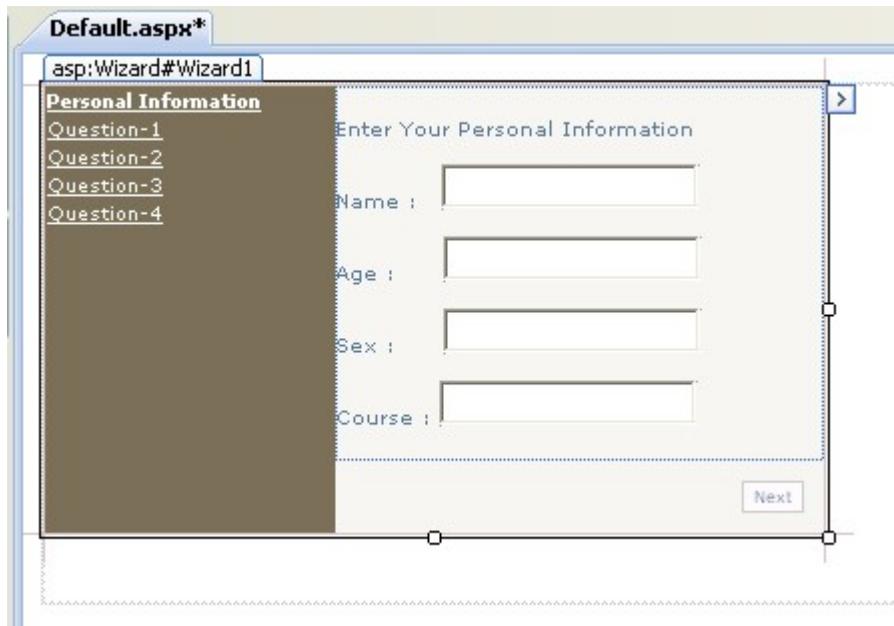
<asp:RadioButtonID="rbdwmy"runat="server"Text="DayWeekMonthYear"/>
</asp:WizardStep>
<asp:WizardStep runat="server" StepType="Finish" Title="Question-4">
<asp:LabelID="lblque4"runat="server"
Text="Question 4 : In TreeView, the node which is at last level is known as?"></asp:Label>
<br/>
<br/>
<asp:RadioButtonID="rbroot"runat="server"Text="Root Node"/>
<br/>
<br/>
<asp:RadioButtonID="rbparent"runat="server"Text="Parent Node"/>
<br/>
<br/>
<asp:RadioButtonID="rbleaf"runat="server"Text="Leaf Node"/>
<br/>
<br/>
<asp:RadioButtonID="rblast"runat="server"Text="Last Node"/>
</asp:WizardStep>
</WizardSteps>
<SideBarButtonStyle BorderWidth="0px" Font-Names="Verdana" ForeColor="White"/>
```

## Subject :- ASP.NET

```
<NavigationButtonStyleBackColor="#FFFFBF"BorderColor="#CCCCCC"
BorderStyle="Solid"BorderWidth="1px"Font-Names="Verdana"Font-Size="0.8em"
ForeColor="#284775"/>
<SideBarStyleBackColor="#7C6F57"BorderWidth="0px"Font-Size="0.9em"
VerticalAlign="Top"/>
<HeaderStyleBackColor="#5D7B9D"BorderStyle="Solid"Font-Bold="True"
Font-Size="0.9em"ForeColor="White"HorizontalAlign="Left"/>
</asp:Wizard>

</div>
</form>
</body>
</html>
```

### ASPX Page:-



Default.aspx\*

asp:Wizard#Wizard1

Personal Information

**Question-1**

Question 1: Which of the following control works on XML File?

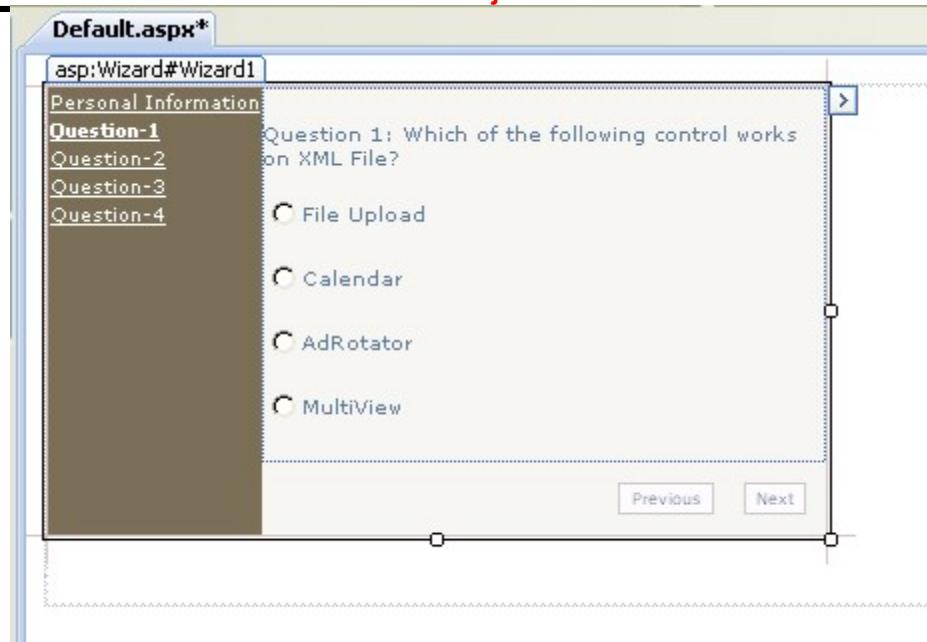
File Upload

Calendar

AdRotator

MultiView

Previous      Next



Default.aspx.cs Default.aspx\*

asp:Wizard#Wizard1

Personal Information

**Question-2**

Question 2 : Which is the default view of Menu Control in ASP.NET?

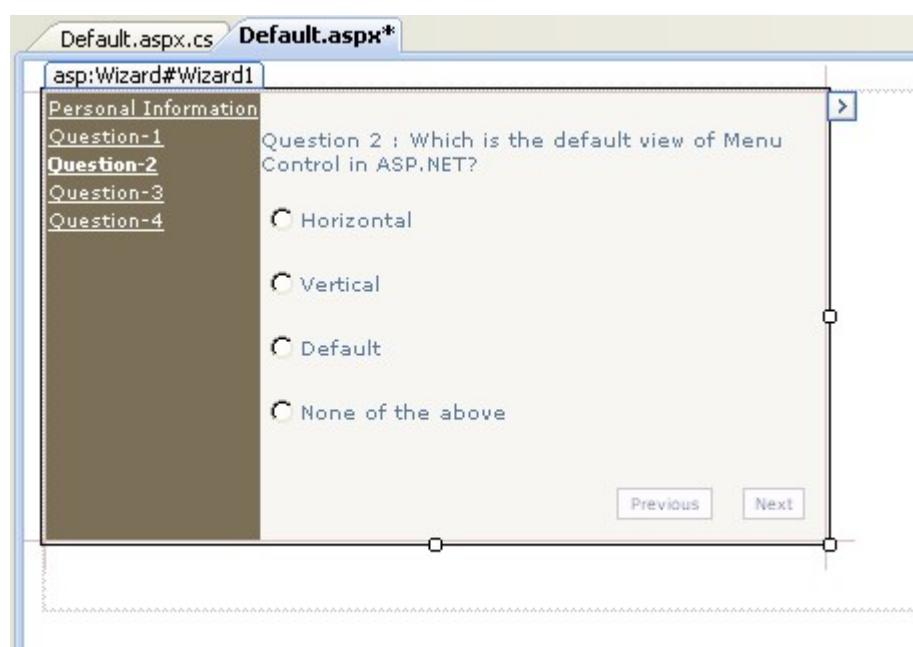
Horizontal

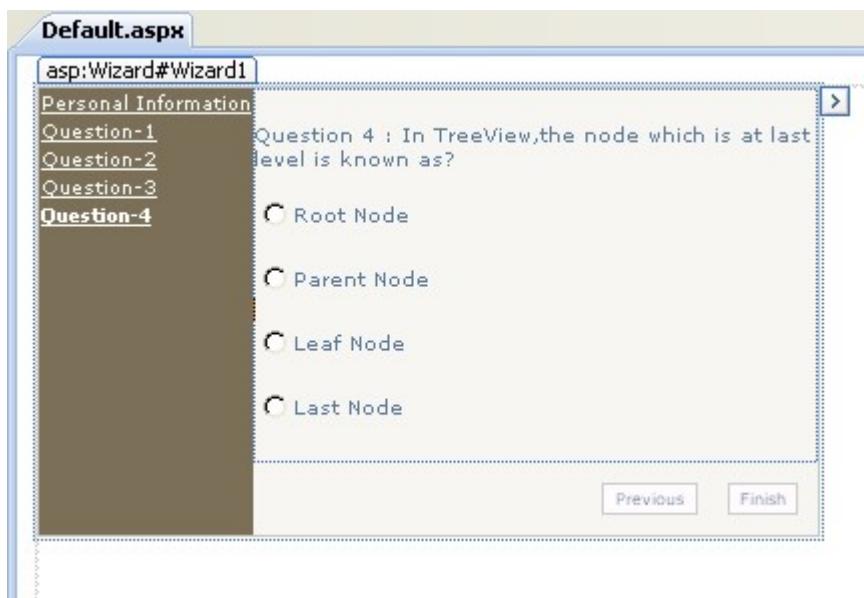
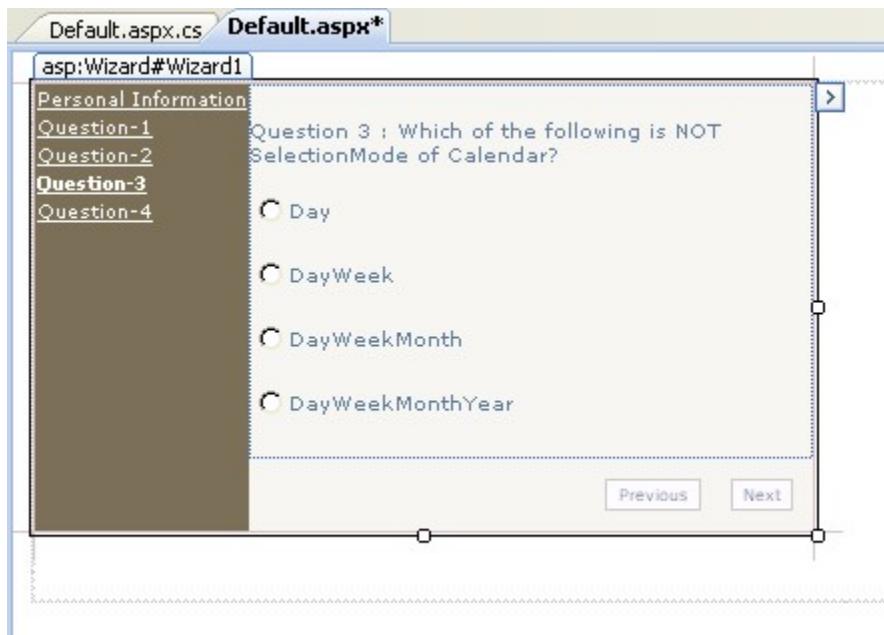
Vertical

Default

None of the above

Previous      Next





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)
    {

    }

    protected void TextBox3_TextChanged(object sender, EventArgs e)
    {

    }

    protected void TextBox4_TextChanged(object sender, EventArgs e)
    {

    }

    protected void RadioButton3_CheckedChanged(object sender, EventArgs e)
    {

    }

    protected void Wizard1_FinishButtonClick(object sender, WizardNavigationEventArgs e)
    {
        Response.Write("<h2>Personal Information : </h2><br/>");
        Response.Write("Name : " + txtname.Text + "<br/>");
        Response.Write("Age : " + txtage.Text + "<br/>");
        Response.Write("Sex : " + txtsex.Text + "<br/>");
        Response.Write("Course: " + txtcourse.Text + "<br/>");
        Response.Write("<hr>");

        Response.Write("<h2>Ans1 :</h2>");
        if (rbupload.Checked == true)
            Response.Write("File Upload");
        if (rbcal.Checked == true)
            Response.Write("Calendar");

        if (rbad.Checked == true)
            Response.Write("AdRotator");
        if (rbmulti.Checked == true)
            Response.Write("MultiView");
        Response.Write("<br/><hr>");

        Response.Write("<h2>Ans2 :</h2>");
        if (rbhor.Checked == true)
            Response.Write("Horizontal");
        if (rbvert.Checked == true)
            Response.Write("Vertical");
        if (rbdef.Checked == true)
            Response.Write("Default");
        if (rbnone.Checked == true)
            Response.Write("None of the above");
        Response.Write("<br/><hr>");

        Response.Write("<h2>Ans3 :</h2>");
        if (rbdy.Checked == true)
            Response.Write("Day");
        if (rbdyweek.Checked == true)
            Response.Write("DayWeek");
```

```
if(rbdwm.Checked == true)
    Response.Write("DayWeekMonth");
if(rbdwmy.Checked == true)
    Response.Write("DayWeekMonthYear");
    Response.Write("<br/><hr>");

    Response.Write("<h2>Ans4 :</h2>");
if(rbroot.Checked == true)
    Response.Write("Root Node");
if(rbparent.Checked == true)
    Response.Write("Parent Node");
if(rbleaf.Checked == true)
    Response.Write("Leaf Node");
if(rblast.Checked == true)
    Response.Write("Last Node");
    Response.Write("<br/><hr>");

}
}
```

**At Run Time:-**

**Personal Information**

Question-1      Enter Your Personal Information

Question-2

Question-3

Question-4

Name :	Shivani
Age :	22
Sex :	F
Course :	B.E.

**Next**



Personal  
Information  
**Question-1**  
Question-2  
Question-3  
Question-4

Question 1: Which of the following control works on XML File?

- File Upload
- Calendar
- AdRotator
- MultiView

[Previous](#)

[Next](#)

Personal  
Information  
**Question-1**  
**Question-2**  
Question-3  
Question-4

Question 2 : Which is the default view of Menu Control in ASP.NET?

- Horizontal
- Vertical
- Default
- None of the above

[Previous](#)

[Next](#)

[Personal Information](#)  
[Question-1](#)  
[Question-2](#)  
**[Question-3](#)**  
[Question-4](#)

Question 3 : Which of the following is NOT SelectionMode of Calendar?

- Day
- DayWeek
- DayWeekMonth
- DayWeekMonthYear

[Previous](#)

[Next](#)

[Personal Information](#)  
[Question-1](#)  
[Question-2](#)  
[Question-3](#)  
**[Question-4](#)**

Question 4 : In TreeView, the node which is at last level is known as?

- Root Node
- Parent Node
- Leaf Node
- Last Node

[Previous](#)

[Finish](#)

## Personal Information :

Name : Shivani

Age : 22

Sex : F

Course: B.E.

---

### Ans1 :

AdRotator

---

### Ans2 :

Vertical

---

### Ans3 :

Day Week Month

---

### Ans4 :

Leaf Node

---

**Lab Manual  
Developed at  
Computer Science Laboratory  
Of  
Shree M.M.Ghodasara Mahila College  
Under guidance of**

<b>Departmental Head</b>	<b>Dr. Raksha Bathani</b>
<b>Lab Incharge</b>	<b>Mr. Rajesh Makwana</b>
<b>Faculty</b>	<b>Prof. Bhavisha Viramgama</b>

*~ There is no alteration of Hard work*