CA LAB-XII (B) Lab on Microsoft .Net Technologies

Assignment 1) Creation of Virtual Directory, Home directory, Home page, hosting of website

Assignment 2) Demonstrate Page Life Cycle of ASP.NET. Use important page events for your demonstration.

1. File-New Website
2. Take 1 Label and 1 Button
3. Write Following Code

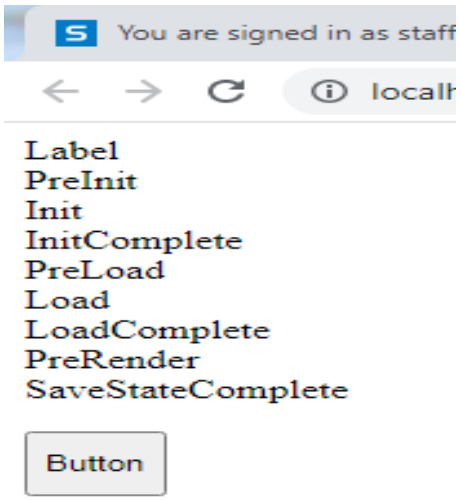
```
using System;
using System.Configuration;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.HtmlControls;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Xml.Linq;

public partial class _Default : System.Web.UI.Page
{
    protected void Page_PreInit(object sender, EventArgs e)
    {
        //Work and It will assign the values to label.
        Label1.Text = Label1.Text + "<br/>" + "PreInit";
    }
    protected void Page_Init(object sender, EventArgs e)
    {
        //Work and It will assign the values to label.
        Label1.Text = Label1.Text + "<br/>" + "Init";
    }
    protected void Page_InitComplete(object sender, EventArgs e)
    {
        //Work and It will assign the values to label.
        Label1.Text = Label1.Text + "<br/>" + "InitComplete";
    }
    protected override void OnPreLoad(EventArgs e)
    {
        //Work and It will assign the values to label.
        //If the page is post back, then label contrl values will be
loaded from view state.
        //E.g: If you string str = Label1.Text, then str will contain
viewstate values.
        Label1.Text = Label1.Text + "<br/>" + "PreLoad";
    }
    protected void Page_Load(object sender, EventArgs e)
    {
```

```
        //Work and It will assign the values to label.
        Label1.Text = Label1.Text + "<br/>" + "Load";
    }
    protected void Button1_Click(object sender, EventArgs e)
    {
        //Work and It will assign the values to label.
        Label1.Text = Label1.Text + "<br/>" + "btnSubmit_Click";
    }
    protected void Page_LoadComplete(object sender, EventArgs e)
    {
        //Work and It will assign the values to label.
        Label1.Text = Label1.Text + "<br/>" + "LoadComplete";
    }
    protected override void OnPreRender(EventArgs e)
    {
        //Work and It will assign the values to label.
        Label1.Text = Label1.Text + "<br/>" + "PreRender";
    }
    protected override void OnSaveStateComplete(EventArgs e)
    {
        //Work and It will assign the values to label.
        //But "SaveStateComplete" values will not be available during post
back. i.e. View state.
        Label1.Text = Label1.Text + "<br/>" + "SaveStateComplete";
    }
    protected void Page_UnLoad(object sender, EventArgs e)
    {
        //Work and it will not effect label contrl, view stae and post
back data.
        Label1.Text = Label1.Text + "<br/>" + "UnLoad";
    }

}
```

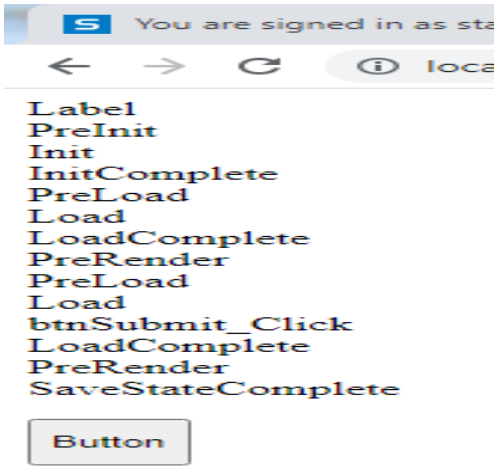
4. Run

OUTPUT:-

Before Button Click



After Button Click



Assignment 3) Write VB.Net/C# console applications to demonstrate: OO concepts: polymorphism, encapsulation, inheritance, interface inheritance, abstract classes/methods, overloading, overriding, collection classes, properties

1. File-New Project-Console Application
2. Write Following Code

```
using System;
using System.Collections;

    //Function Overloading
    /*class Program
    {
        public void add(int a, int b)
        {
            Console.WriteLine(a + b);
        }
        public void add(float x, float y)
        {
            Console.WriteLine(x + y);
        }
        public void add(string s1, string s2)
        {
            Console.WriteLine(s1 + s2);
        }
        static void Main(string[] args)
        {
            Program obj = new Program();
            obj.add(10, 20);
            obj.add(10.5f, 20.5f);
            obj.add("pranaya", "kumar");
            Console.WriteLine("Press any key to exist.");
            Console.ReadKey();
        }
    }*/
    //Function Overriding
    /*class Class1
        {
            public virtual void show()
            {
                Console.WriteLine("Super class show method");
            }
        }
      class Class2 : Class1
        {
            public override void show()
            {
                Console.WriteLine("Sub class override show method");
            }
        }
        class Program
        {
            static void Main(string[] args)
            {
                Class2 obj = new Class2();
```

```
                obj.show();
                Console.ReadKey();
            }
        }*/
    //Function Hiding
    /*class Class1
    {
        public void display()
        {
            Console.WriteLine("Super class display method");
        }
    }
    class Class2 : Class1
    {
        public new void display()
        {
            Console.WriteLine("Sub class display method");
        }
    }
    class Program
    {
        static void Main(string[] args)
        {
            Class2 obj = new Class2();
            obj.display();
            Console.ReadKey();
        }
    }*/
    //Encapsulation
    /*public class Bank
    {
        //hiding class data by declaring the variable as private
        private double balance;
        //creating public setter and getter methods
        public double getBalance()
        {
            //add validation logic if needed
            return balance;
        }
        public void setBalance(double balance)
        {
            // add validation logic to check whether data is correct or
not
            this.balance = balance;
        }
    }
    class BankUser
    {
        public static void Main()
        {
            Bank SBI = new Bank();
            SBI.setBalance(500);
            Console.WriteLine(SBI.getBalance());
            Console.WriteLine("Press any key to exist.");
            Console.ReadKey();
        }
    }*/
```

```
//inheritance
/*class Branch
{
    int BranchCode;
    string BranchName, BranchAddress;
    public void GetBranchData()
    {
        Console.WriteLine("ENTER BRANCH DETAILS:");
        Console.WriteLine("ENTER BRANCH CODE");
        BranchCode = int.Parse(Console.ReadLine());
        Console.WriteLine("ENTER BRANCH NAME");
        BranchName = Console.ReadLine();
        Console.WriteLine("ENTER BRANCH ADDRESS");
        BranchAddress = Console.ReadLine();
    }
    public void DisplayBranchData()
    {
        Console.WriteLine("BRANCH CODE IS : " + BranchCode);
        Console.WriteLine("BRANCH NAME IS : " + BranchName);
        Console.WriteLine("BRANCH ADDRESS IS : " + BranchAddress);
    }
}
class Employee : Branch
{
    int EmployeeId, EmployeeAge;
    string EmployeeName, EmployeeAddress;
    public void GetEmployeeData()
    {
        Console.WriteLine("ENTER EMPLYEE DETAILS:");
        Console.WriteLine("ENTER EMPLOYEE ID");
        EmployeeId = int.Parse(Console.ReadLine());
        Console.WriteLine("ENTER EMPLOYEE AGE");
        EmployeeAge = int.Parse(Console.ReadLine());
        Console.WriteLine("ENTER EMPLOYEE NAME");
        EmployeeName = Console.ReadLine();
        Console.WriteLine("ENTER EMPLOYEE ADDRESS");
        EmployeeAddress = Console.ReadLine();
    }
    public void DisplayEmployeeData()
    {
        Console.WriteLine("EMPLOYEE ID IS : " + EmployeeId);
        Console.WriteLine("EMPLOYEE NAME IS : " + EmployeeName);
        Console.WriteLine("EMPLOYEE ADDRESS IS : " + EmployeeAddress);
        Console.WriteLine("EMPLOYEE AGE IS : " + EmployeeAge);
    }
}
class Program
{
    static void Main(string[] args)
    {
        Employee obj1 = new Employee();
        obj1.GetBranchData();
        obj1.GetEmployeeData();
        obj1.DisplayBranchData();
        obj1.DisplayEmployeeData();
        Console.WriteLine("Press any key to exist.");
        Console.ReadKey();
```

```
    }
}*/
//interface inheritance
/*public interface A
{
    void method1();
    void method2();
}
interface B : A
{
    void method3();
}
class MyClass : B
{
    public void method1()
    {
        Console.WriteLine("implement method1");
    }
    public void method2()
    {
        Console.WriteLine("implement method2");
    }
    public void method3()
    {
        Console.WriteLine("implement method3");
    }
}
class Program
{
    static void Main(string[] args)
    {
        MyClass obj = new MyClass();
        obj.method1();
        obj.method2();
        obj.method3();
        Console.WriteLine("Press any key to exist.");
        Console.ReadKey();
    }
}*/
//Abstract classes/methods
/*public abstract class MyClass
{
    public abstract void calculate(double x);
    public void show()
    {
        Console.WriteLine("Show");
    }
}
class Sub1 : MyClass
{
    public override void calculate(double x)
    {
        Console.WriteLine("SQUARE ROOT IS " + Math.Sqrt(x));
    }
}
class Test
{
```

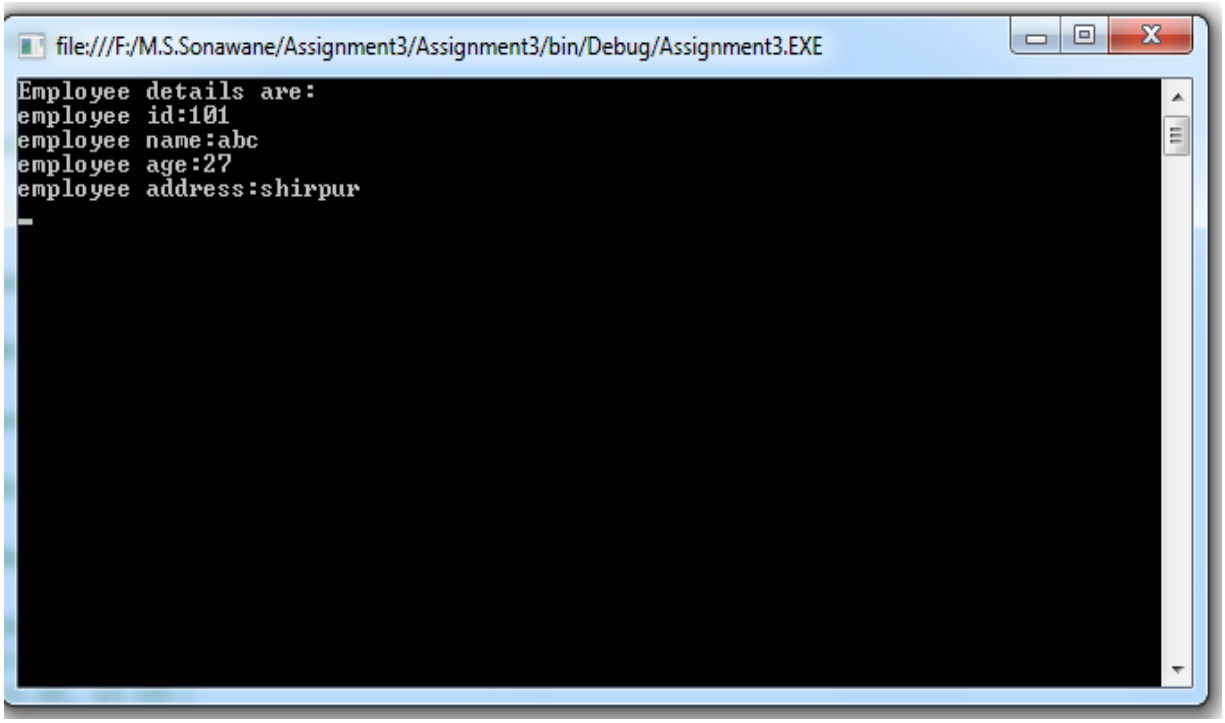
```
        static void Main(string[] args)
        {
            Sub1 obj1 = new Sub1();
            obj1.calculate(9);
            obj1.show();
            Console.ReadKey();
        }
    }*/
//Collection Classes  Stack, Queue, LinkedList, SortedList, ArrayList,
HashTable, etc.
/*class Program
{
    static void Main(string[] args)
    {
        //Createing ArrayList collection using default constructor
        ArrayList al = new ArrayList();
        Console.WriteLine("Initial Capacity: " + al.Capacity);
        al.Add(10);
        Console.WriteLine("Capacity after adding first item: " +
al.Capacity);
        al.Add("hello");
        al.Add(true);
        al.Add(3.14f);
        Console.WriteLine("Capacity after adding fourth item: " +
al.Capacity);
        al.Add('A');
        Console.WriteLine("Capacity after adding 5th element: " +
al.Capacity);
        //Printing the ArrayList elements using for loop
        for (int i = 0; i < al.Count; i++)
        {
            Console.Write(al[i] + "  ");
        }
        Console.WriteLine();
        //Removing the values from the middle of the array list
        //here we are removing by value
        al.Remove(true);
        //You can also remove element by using index position
        // al.RemoveAt(2);
        //Printing the ArrayList elements using foreach loop after
        // removing an element from the collection
        foreach (object obj in al)
        {
            Console.Write(obj + "  ");
        }
        Console.WriteLine();
        //inserting values into the middle of the array list collection
        al.Insert(2, false);
        // Printing the values of the collection using foreach loop after
        // inserting an element into the middle of the collection
        foreach (object obj in al)
        {
            Console.Write(obj + "  ");
        }
        Console.WriteLine();
        // creating new ArrayList collection by passing the old
        // array list as parameter
```

```
        ArrayList coll = new ArrayList(al);
        Console.WriteLine("Initial capacity of new array list collection:"
+ coll.Capacity);
        // Printing the values of the new array list collection using
foreach loop
        foreach (object obj in coll)
        {
            Console.Write(obj + "   ");
        }
        Console.ReadKey();
    }
}*/
//Properties
public class Example
{
    private int _empid, _eage;
    private string _ename, _eaddress;
    public int empid
    {
        set
        {
            _empid = value;
        }
        get
        {
            return _empid;
        }
    }
    public int eage
    {
        set
        {
            _eage = value;
        }
        get
        {
            return _eage;
        }
    }
    public string ename
    {
        set
        {
            _ename = value;
        }
        get
        {
            return _ename;
        }
    }
    public string eaddress
    {
        set
        {
            _eaddress = value;
        }
        get
```

```
        {
            return _eaddress;
        }
    }
}
class Program
{
    static void Main(string[] args)
    {
        Example obj1 = new Example();
        obj1.empid = 101;
        obj1.ename = "abc";
        obj1.eage = 27;
        obj1.eaddress = "shirpur";
        Console.WriteLine("Employee details are:");
        Console.WriteLine("employee id:" + obj1.empid);
        Console.WriteLine("employee name:" + obj1.ename);
        Console.WriteLine("employee age:" + obj1.eage);
        Console.WriteLine("employee address:" + obj1.eaddress);
        Console.ReadKey();
    }
}
```

3. Run

OUTPUT:-



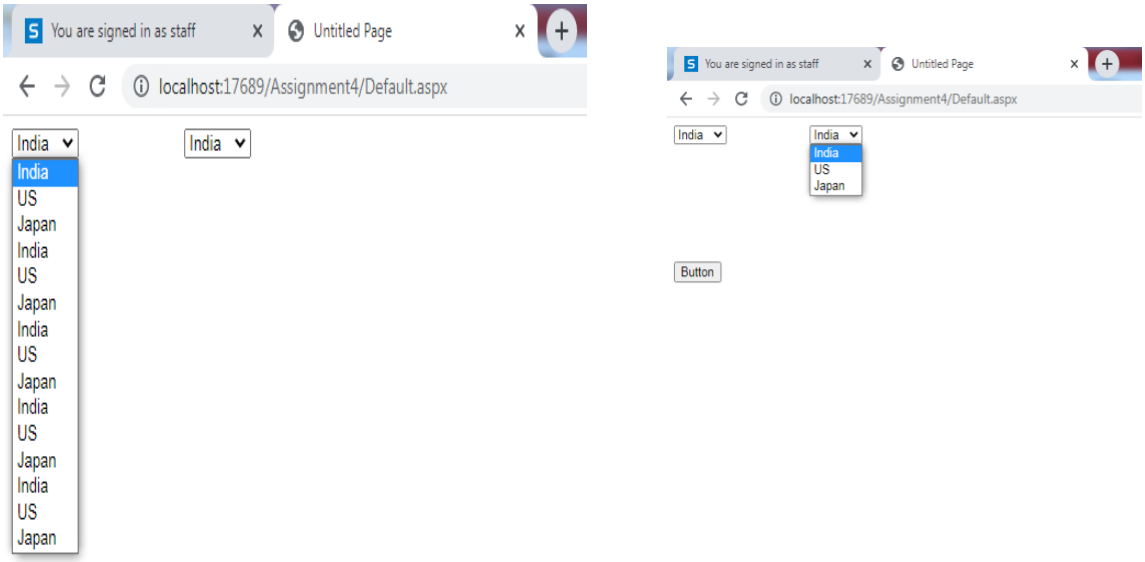
Assignment 4) Demonstrate concept of postback and viewstate using web form server controls of ASP.NET

1. File-New Website
2. Take 2 DropDownList2, 2nd having property `EnableViewState="false"`
3. Write Following Code

```
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)
    {
        bind1();
        bind2();
    }
    public void bind1()
    {
        ListItem l1 = new ListItem("India");
        ListItem l2 = new ListItem("US");
        ListItem l3 = new ListItem("Japan");
        DropDownList1.Items.Add(l1);
        DropDownList1.Items.Add(l2);
        DropDownList1.Items.Add(l3);
    }

    public void bind2()
    {
        ListItem l1 = new ListItem("India");
        ListItem l2 = new ListItem("US");
        ListItem l3 = new ListItem("Japan");
        DropDownList2.Items.Add(l1);
        DropDownList2.Items.Add(l2);
        DropDownList2.Items.Add(l3);
    }

    protected void Button1_Click(object sender, EventArgs e)
    {

    }
}
```

4. Run

OUTPUT:-



Assignment 5) Demonstrate various Web form server controls using sample data entry screen form for registering for a service on website. Also use validation controls to validate input data.

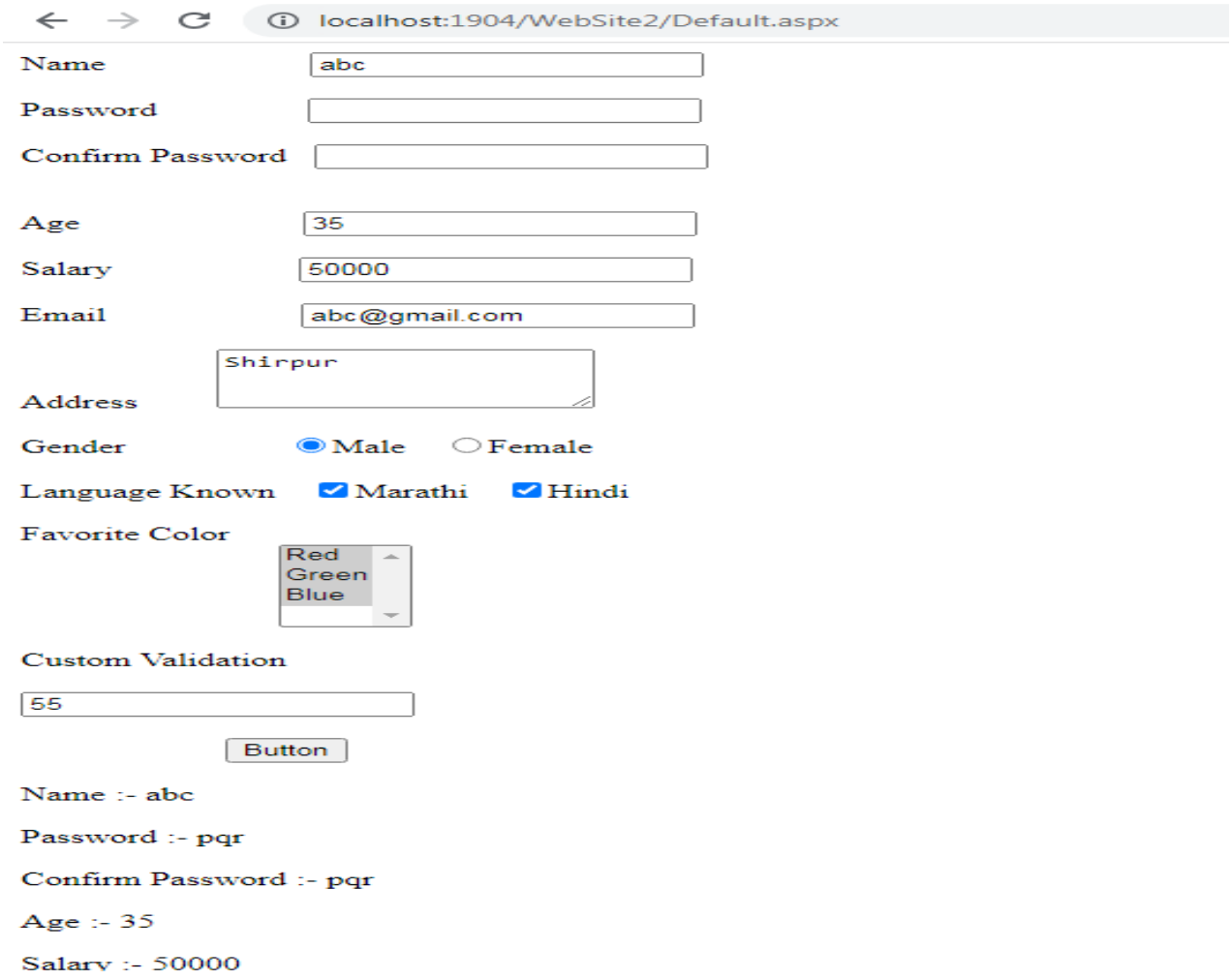
1. File-New Website
2. Write Following Code

```
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)
    {

    }
    protected void CustomValidator1_ServerValidate(object source,
ServerValidateEventArgs args)
    {
        try
```

```
        {
            int num = int.Parse(args.Value);
            args.IsValid = ((num % 5) == 0);
        }
        catch (Exception ex)
        {
            args.IsValid = false;

        }
    }
    protected void Button1_Click(object sender, EventArgs e)
    {
        Label1.Text = "Name :- " + TextBox8.Text;
        Label1.Visible = true;
Label2.Text = "Password :- " + TextBox10.Text;
Label2. Visible = true;
Label3.Text = "Confirm Password :- " + TextBox11.Text;
Label3.Visible = true;
Label4.Text = "Age :- " + TextBox12.Text;
Label4.Visible = true;
Label5.Text = "Salary :- " + TextBox13.Text;
Label5.Visible = true;
Label6.Text = "Email :- " + TextBox6.Text;
Label6.Visible = true;
Label7.Text = "Address :- " + TextBox9.Text;
Label7.Visible = true;
if (RadioButton1.Checked)
{
Label8.Text = " Gender :- Male";
Label8.Visible = true;
}
else
{
Label8.Text = " Gender :- Female";
Label8.Visible = true;
}
if (CheckBox1.Checked)
{
Label9.Visible = true;
Label9.Text = "Known Language :- " + CheckBox1.Text;
}
if (CheckBox2.Checked)
{
Label9.Visible = true;
Label9.Text += "," + CheckBox2.Text;
}
string msg = "";
foreach (ListItem li in ListBox1.Items)
{
    if (li.Selected == true)
    {
        msg += "<BR>" + li.Text + " is selected.";
    }
}
Label10.Text = msg;
 }
}
```

3. Run

OUTPUT:-


localhost:1904/WebSite2/Default.aspx
Name
abc
Password
Confirm Password
Age
35
Salary
50000
Email
abc@gmail.com
Address
Shirpur
Gender
Male
Female
Language Known
Marathi
Hindi
Favorite Color
Red
Green
Blue
Custom Validation
55
Button
Name :- abc
Password :- pqr
Confirm Password :- pqr
Age :- 35
Salary :- 50000

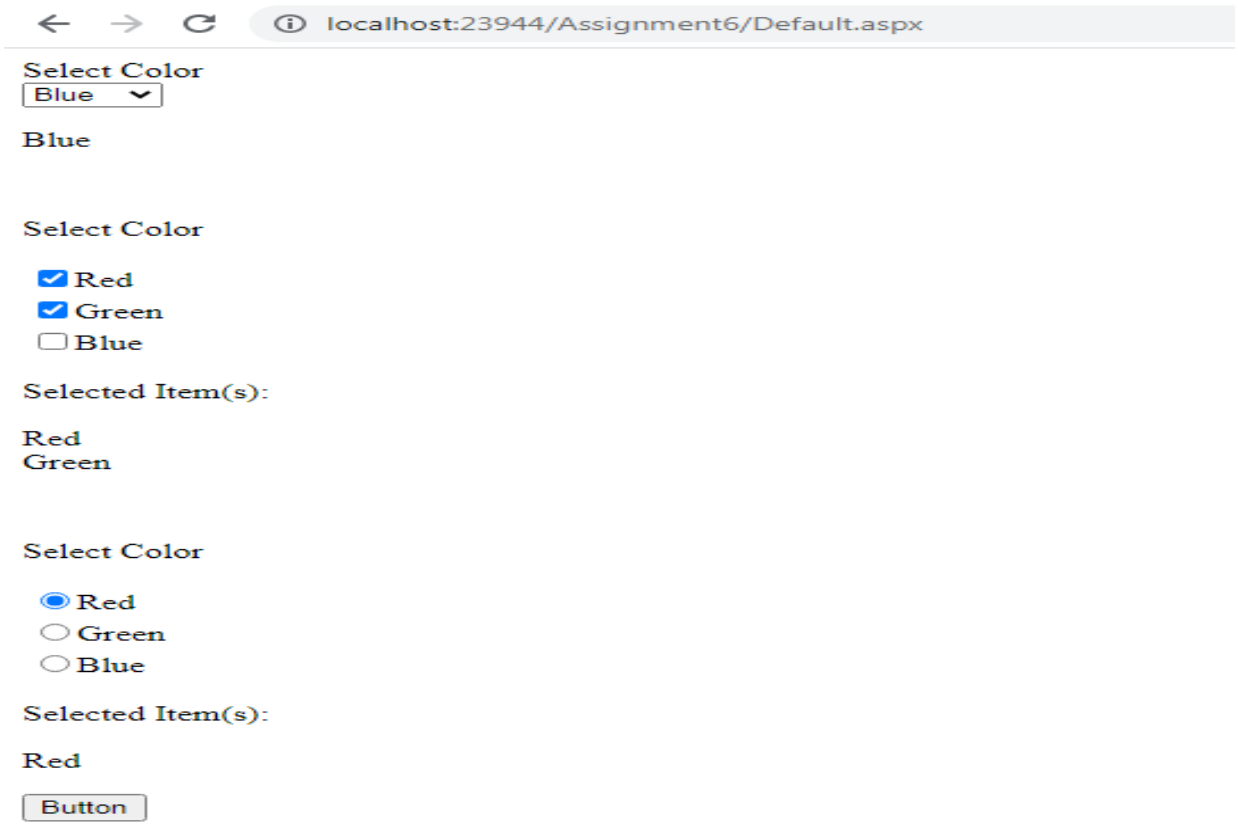
Assignment 6) Demonstrate DropDownList box, CheckButtonList, RadioButtonList controls.

1. File-New Website
2. Write Following Code

```
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)
    {

    }
    protected void Button1_Click(object sender, EventArgs e)
    {
        Label1.Text = DropDownList1.SelectedValue.ToString();

        Label2.Text = "Selected Item(s):<br /><br />";
        for (int i = 0; i < CheckBoxList2.Items.Count; i++)
        {

            if (CheckBoxList2.Items[i].Selected)
            {

                Label2.Text += CheckBoxList2.Items[i].Text + "<br />";

            }

        }

        Label3.Text = "Selected Item(s):<br /><br />";
        for (int i = 0; i < RadioButtonList1.Items.Count; i++)
        {

            if (RadioButtonList1.Items[i].Selected)
            {

                Label3.Text += RadioButtonList1.Items[i].Text + "<br />";

            }

        }
    }
}
```

3. Run

OUTPUT:-



Assignment 7) Demonstrate Databinding using Hashtable, ArraryList, DataTable data sources.

1. Write Following Code in Default.aspx.cs

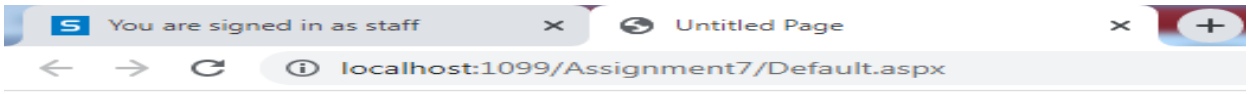
```
using System.Data;
using System.Data.SqlClient;
using System.Collections;
using System.Collections.Generic;
public partial class _Default : System.Web.UI.Page
{

    protected void Page_Load(object sender, EventArgs e)
    {
        SqlConnection con = new SqlConnection("Data
Source=.\\SQLEXPRESS;AttachDbFilename=I:\\2021-22\\Subjects\\CA-305(B)
Microsoft .Net Technologies\\CA-LAB-XII (B) Lab on Microsoft .Net
Technologies\\Assignment7\\App_Data\\Database.mdf;Integrated
Security=True;User Instance=True");
        DataTable ds = new DataTable();
```

```
        SqlDataAdapter sde = new SqlDataAdapter("select * from student",
con);
        sde.Fill(ds);
        GridView1.DataSource = ds;
        GridView1.DataBind();

        ArrayList list = new ArrayList();
        list.Add("AAAA");
        list.Add("BBBB");
        list.Add("CCCC");
        GridView2.DataSource =list;
        GridView2.DataBind();


        Hashtable hashItems = new Hashtable();
        hashItems.Add("4","GGGG");
        hashItems.Add("5","HHHH");
        hashItems.Add("6","IIII");
        hashItems.Add("7","JJJJ");
        GridView3.DataSource = hashItems;
        GridView3.DataBind();
    }

}
```

2. For `Hashtable` Write Following Code in Default.aspx

```
<asp:GridView ID="GridView3" runat="server"AutoGenerateColumns="false">
    <Columns>
                <asp:BoundField DataField="key" HeaderText="Roll No" />
                <asp:BoundField DataField="value" HeaderText="Name" />
     </Columns>
    </asp:GridView>
```

OUTPUT:-

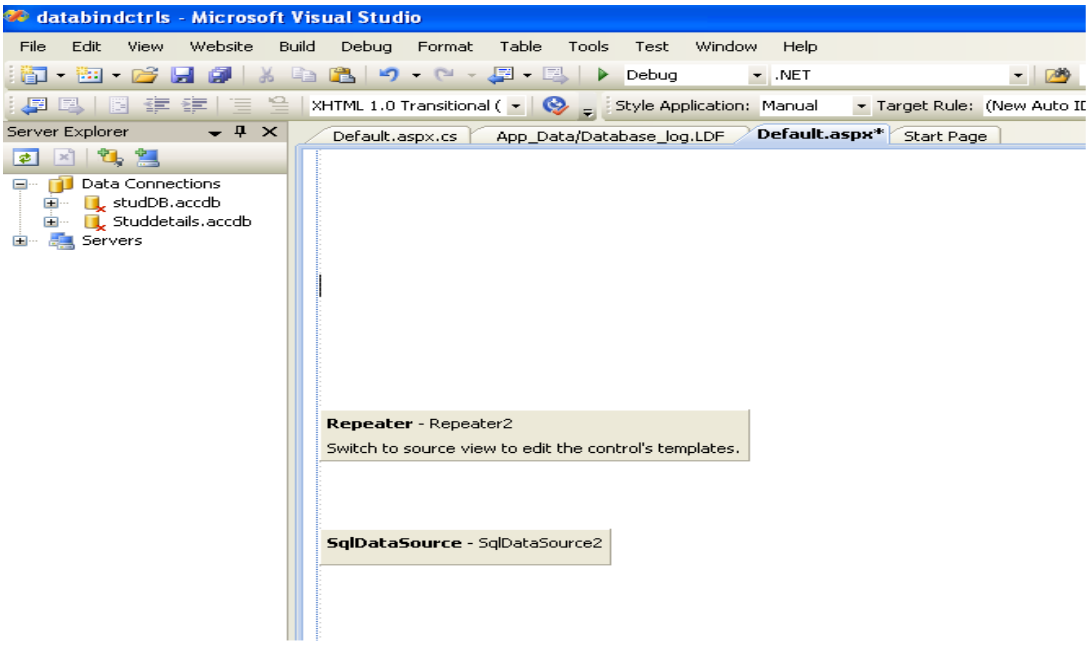


rno	name
1	a
2	b
3	c
4	d
5	e

Item
AAAA
BBBB
CCCC

Roll No	Name
4	GGGG
5	HHHH
6	IIII
7	JJJJ

Assignment 8) Demonstrate Repeater control with the help of various templates.

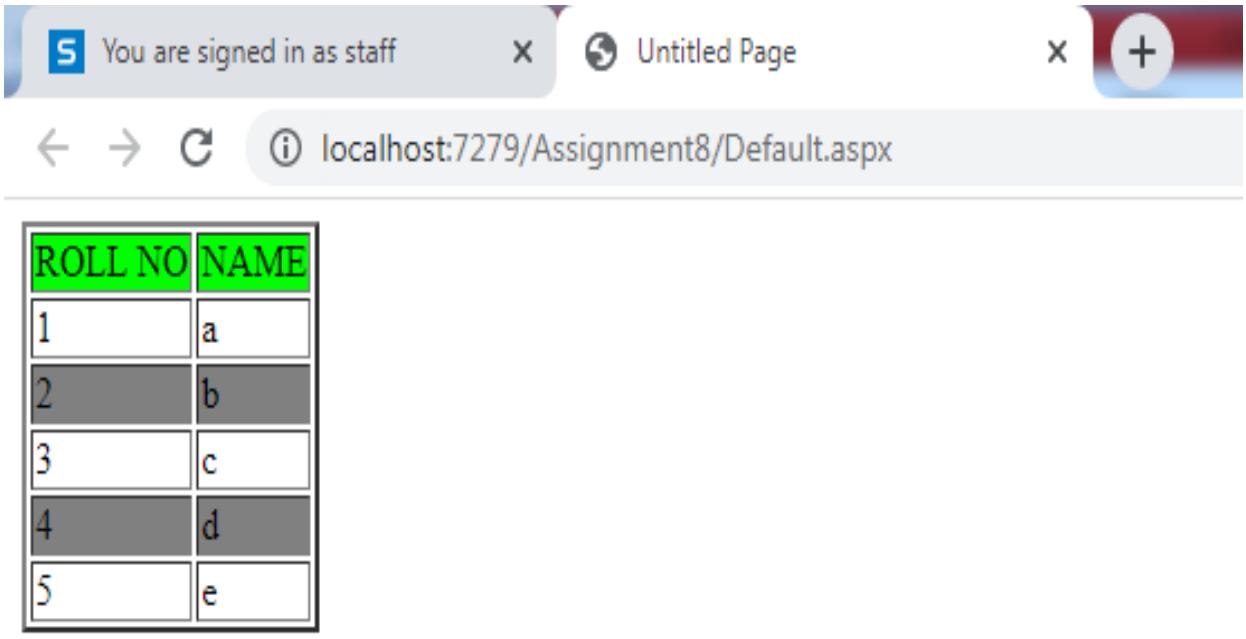
1. Right Click on **App_Data** in **Solution Explorer**
2. Add New Item select **SQL Server Database** and give database name
3. In **Server Explorer** right click on **Tables,** select **Add New Table**
4. Right click on your table and select **Show Table Data** and insert records
5. Take **GridView** from **Data** click on **Choose Data Source, Database,** select your database/table/fields
6. Click on **Test Query** and **Finish**
7. 

Add Following code in Source view

```
<table border =2>
<asp:Repeater ID="Repeater1" runat="server"
DataSourceID="SqlDataSource1">
<HeaderTemplate >
<tr ><td style ="background-color:Lime ;">ROLL NO</td>
<td style ="background-color:Lime ;">NAME</td></tr>
</HeaderTemplate>
<ItemTemplate >
<tr><td><%#Eval("rno") %></td>
<td><%#Eval("name") %></td></tr></ItemTemplate>
<AlternatingItemTemplate >
<tr><td style ="background-color :Gray ;"><%#Eval("rno") %></td>
<td style ="background-color :Gray;"><%#Eval("name")%> </td></tr>
</AlternatingItemTemplate>
   </asp:Repeater> </table>
```

8.Save , BUILT & RUN .

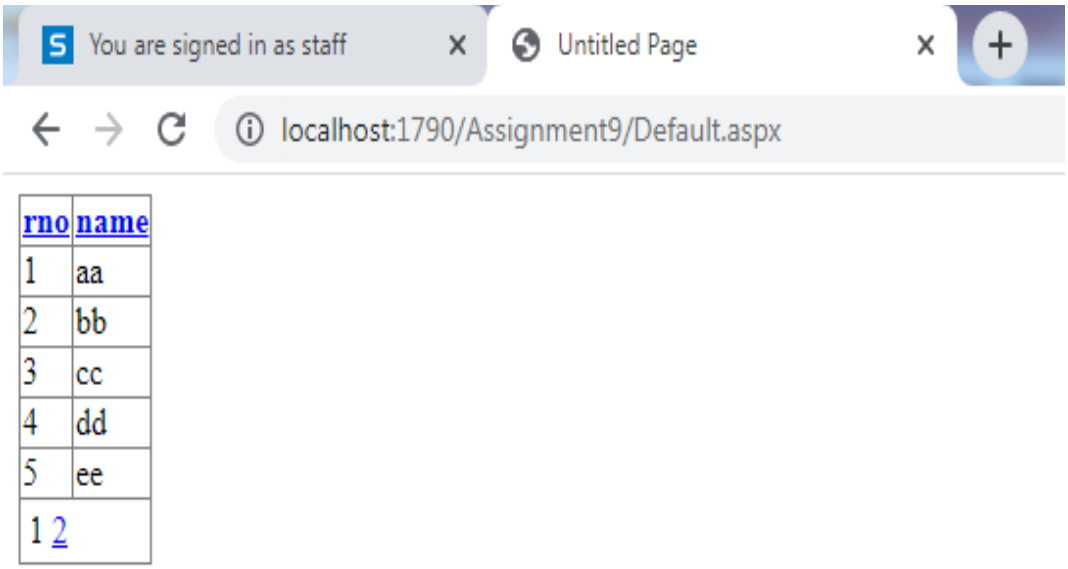
OUTPUT:-



Assignment 9) Demonstrate paging, sorting, filtering of data in asp:DataGrid/DataGridView.

1. Right Click on **App_Data** in **Solution Explorer**
2. Add New Item select **SQL Server Database** and give database name
3. In **Server Explorer** right click on **Tables,** select **Add New Table**
4. Right click on your table and select **Show Table Data** and insert records
5. Take **GridView** from **Data** click on **Choose Data Source, Database,** select your database/table/fields
6. Click on **Test Query** and **Finish**
7. Click on **GridView** and **Tick Enable Paging, Enable Sorting** and set **PageSize** property.
8. Save and Run

OUTPUT:-



rno	name
1	aa
2	bb
3	cc
4	dd
5	ee
1 2	

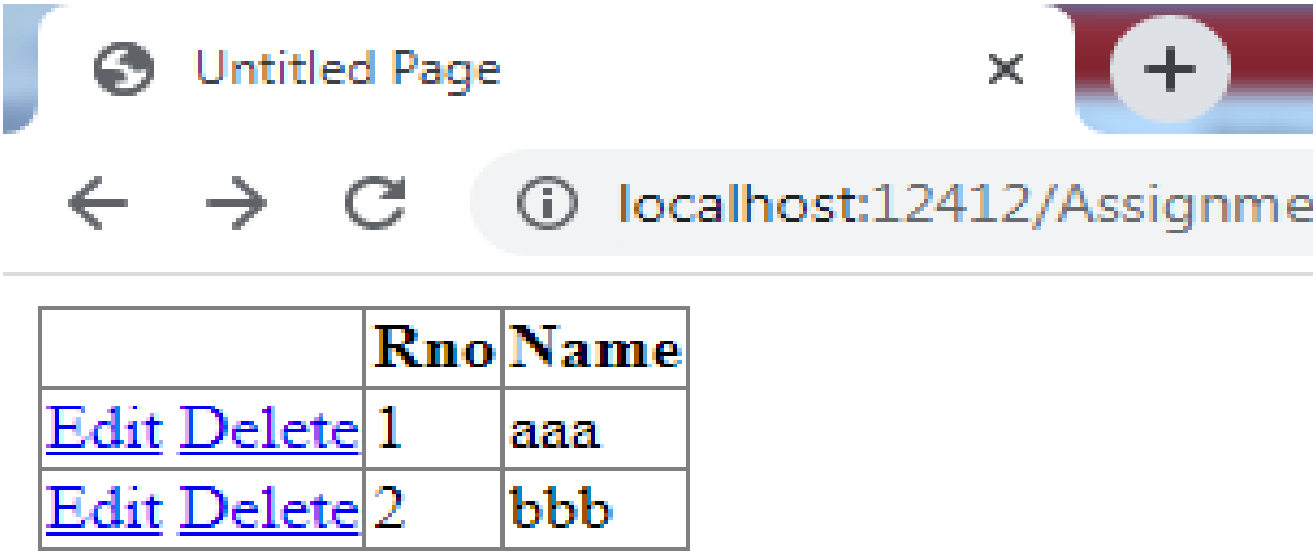
Assignment 10) Demonstrate editing process in DataGrid and DataList controls. Make use of necessary templates for proper visual appearance.

Editing process in DataGrid

1. Right Click on **App_Data** in **Solution Explorer**
2. Add New Item select **SQL Server Database** and give database name
3. In **Server Explorer** right click on **Tables,** select **Add New Table**
4. Right click on your table and select **Show Table Data** and insert records
5. Take **GridView** from **Data** click on **Choose Data Source, Database,** select your database/table/fields
6. Click on **Test Query** and **Finish**
7. Click on **GridView** and set **AutoGenerateDeleteButton="True", AutoGenerateEditButton="True**
8. Write Following Code in Default.aspx

```
9. <asp:SqlDataSource ID="SqlDataSource1" runat="server"
   ConnectionString="<%$ ConnectionStrings:ConnectionString %>"
10. SelectCommand="SELECT * FROM [stud]"
11. UpdateCommand="update [stud] set [name]=@name where [rno]=@rno"
   DeleteCommand="delete from [stud] where [rno]=@rno">
12. </asp:SqlDataSource>
```

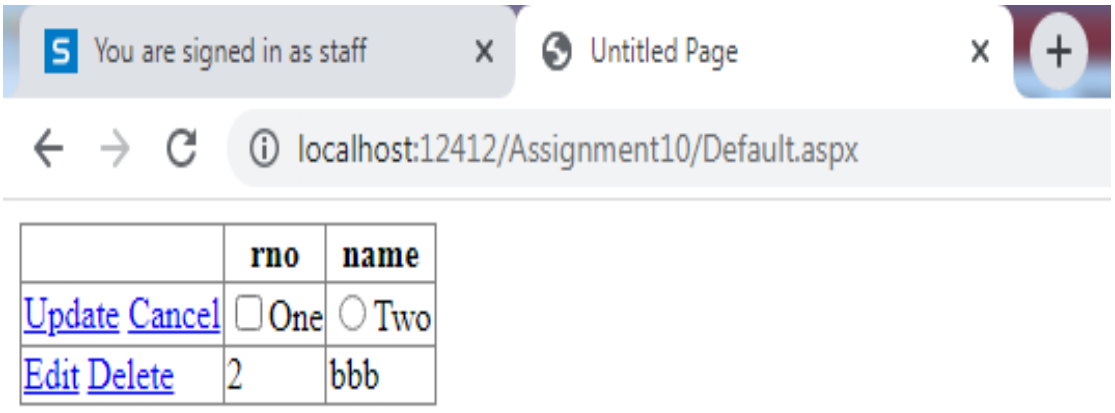
OUTPUT:-



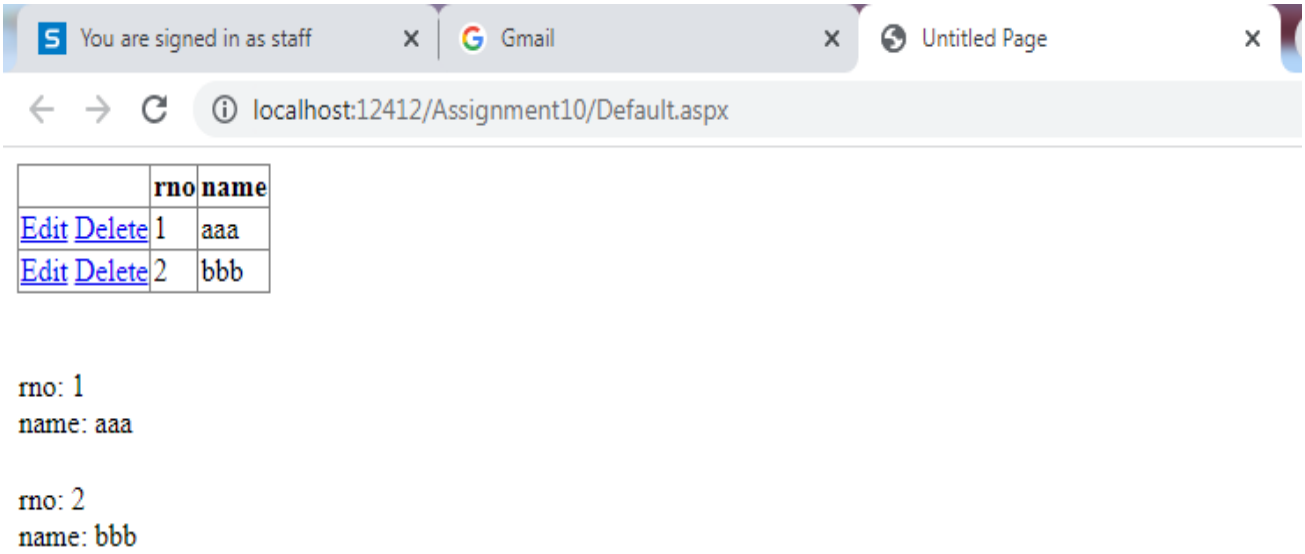
Use of Templates

1. Click on **GridView** and **Edit Columns**
2. Select field and click on **Convert this field into a TemplateField** and observe the code.
3. Click on **GridView** and **Edit Templates** and select template that you want to change (Example:- change **EditItemTemplate** of rollnumber to **CheckBox**(Set its Properties), change **EditItemTemplate** of name to **RadioButton**(Set its Properties)).
4. Save and Run

OUTPUT:-



DataList:-



Assignment 11) Demonstrate State Management features of ASP.NET using sample shopping cart application

1.Start->Programs->Visual studio 2008
2.Create new web site.
3. Take Label1, Button1
4. Write following code

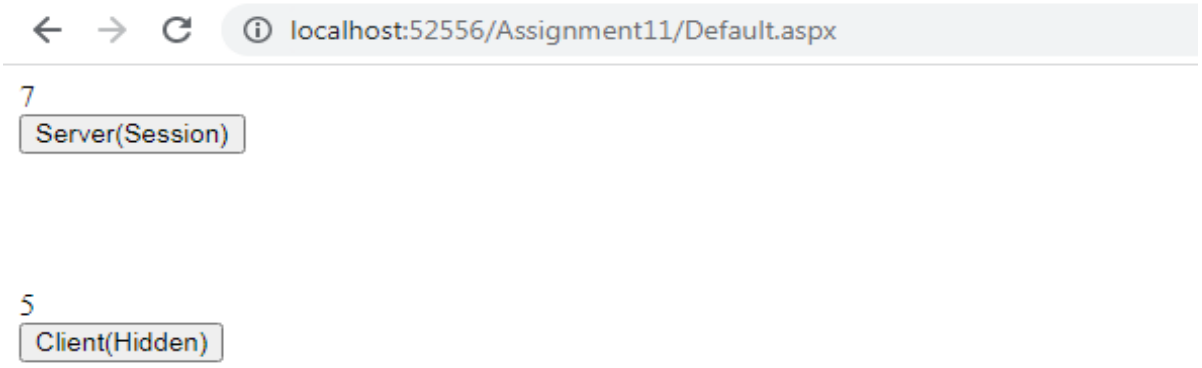
```
protected void Button1_Click(object sender, EventArgs e)
{
Session["Count"] = Convert.ToInt32(Session["Count"]) + 1;//Set Value to
                                                          The Session
Label1.Text = Session["Count"].ToString(); //Get Value from the Sesion
}
```

5. Take Label2, Button2, HiddentField1, Set HiddentField1.Value=0
6. Write following code

```
protected void Button2_Click(object sender, EventArgs e)
{
  if (HiddenField1.Value != null)
  {
      try
      {
          int val = Convert.ToInt32(HiddenField1.Value) + 1;
          HiddenField1.Value = val.ToString();
          Label2.Text = val.ToString();
      }
      catch (Exception exc)
      {
          Response.Write(exc);
```

```
        }
    }
}
```

OUTPUT:-



Assignment 12) Create sample website for demonstrating use of Profiles/Themes using skin files.

Profiles

1. Add the profile details in the web.config file within the <system.web> element.

```
<configuration>
        <system.web>
                <profile>
                   <properties>
                        <add name="Name" type="String"/>
                        <add name="Class" type="String"/>
                   </properties>
                </profile>
        </system.web>
</configuration>
```

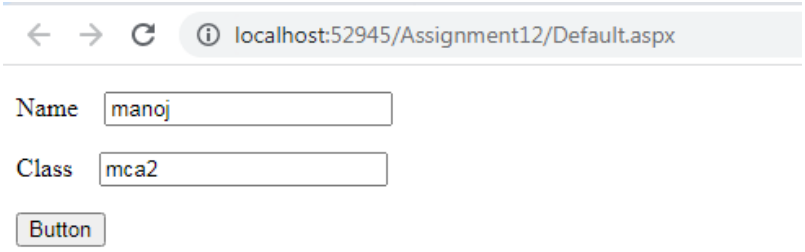
2. Add 2 labels and 2 text boxes to take the user input as defined in the profile and add a button for submitting the data.
3. Update Page_load to display profile information and Write the following handler for the Submit button, for saving the user data into the profile:

```
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)
{
  if (!this.IsPostBack)
  {
      ProfileCommon pc = this.Profile.GetProfile(Profile.UserName);

      if (pc != null)
      {
          this.TextBox1.Text = pc.Name;
          this.TextBox2.Text = pc.Class;
      }
  }
}
protected void Button1_Click(object sender, EventArgs e)
{
  ProfileCommon pc = this.Profile.GetProfile(Profile.UserName);

  if (pc != null)
  {
      pc.Name = this.TextBox1.Text;
      pc.Class= this.TextBox2.Text;
     pc.Save();
  }
}
}
```

4. When the page is executed for the first time, the user needs to enter the information. However, next time the user details would be automatically loaded.

OUTPUT:-



Themes/Skins

1.Start->Programs->Visual studio 2008
2. Create new web site.
3. Add one Theme Folder in your Website.
i) Right Click > Add New ASP.NET Folder >Choose Theme folder.
4. Add one Skin File in your Website.
ii)Right Click > Add New Item > Skin file.
5. Write a following code in Skin File.

```
<asp:Label runat="server" ForeColor="orange"></asp:Label>

<asp:TextBox runat="server" BackColor="orange" ForeColor="black"
BorderColor="Red" BorderStyle="Solid"></asp:TextBox>

<asp:RadioButton runat="server" BackColor="orange"
ForeColor="black" BorderColor="Red"
BorderStyle="Solid"></asp:RadioButton>

<asp:Button runat="server" BackColor="orange" ForeColor="black"
BorderColor="Red" BorderStyle="Solid"></asp:Button>

<asp:TextBox runat="server" skinID="s1" BackColor="Red"
ForeColor="black" BorderColor="Red"
BorderStyle="Solid"></asp:TextBox>
```

6. Apply this skin file in thoruh out the website then **Goto web.config** file and set the Page attribute value give the theme name.

```
<pages theme="Theme1">
```

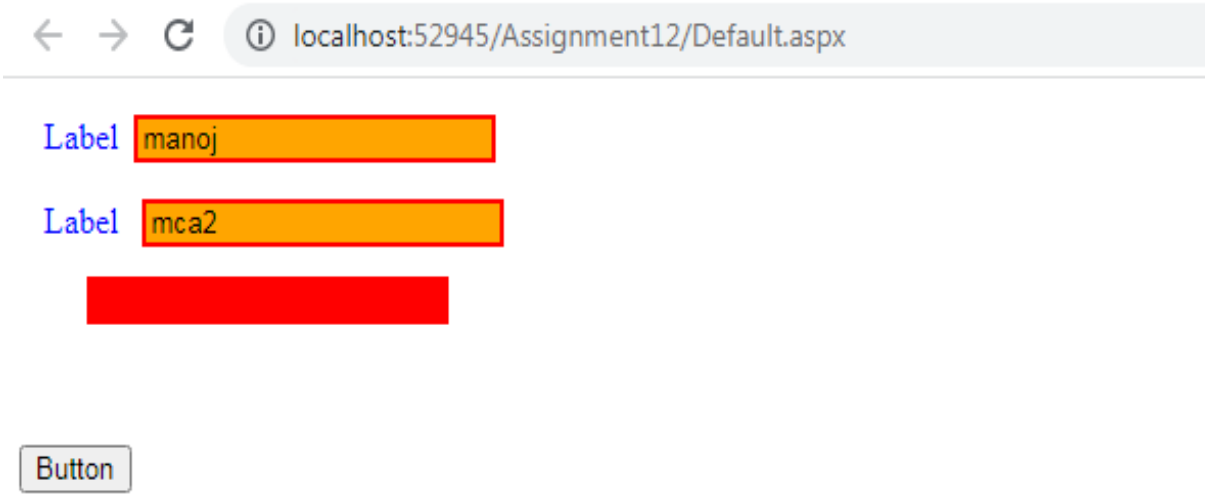
7. Apply this skin file on page level controls then Goto Source Code set the Page attribute value give the theme name.

```
<%@ Page Theme="Theme1" Language="C#" AutoEventWireup="true"
CodeFile="Default.aspx.cs" Inherits="_Default" %>
```

8. Apply this skin on Particular Control level then Goto Source Code set the skinID.

```
<asp:TextBox ID="TextBox3" runat="server"
SkinID="s1"></asp:TextBox>
```

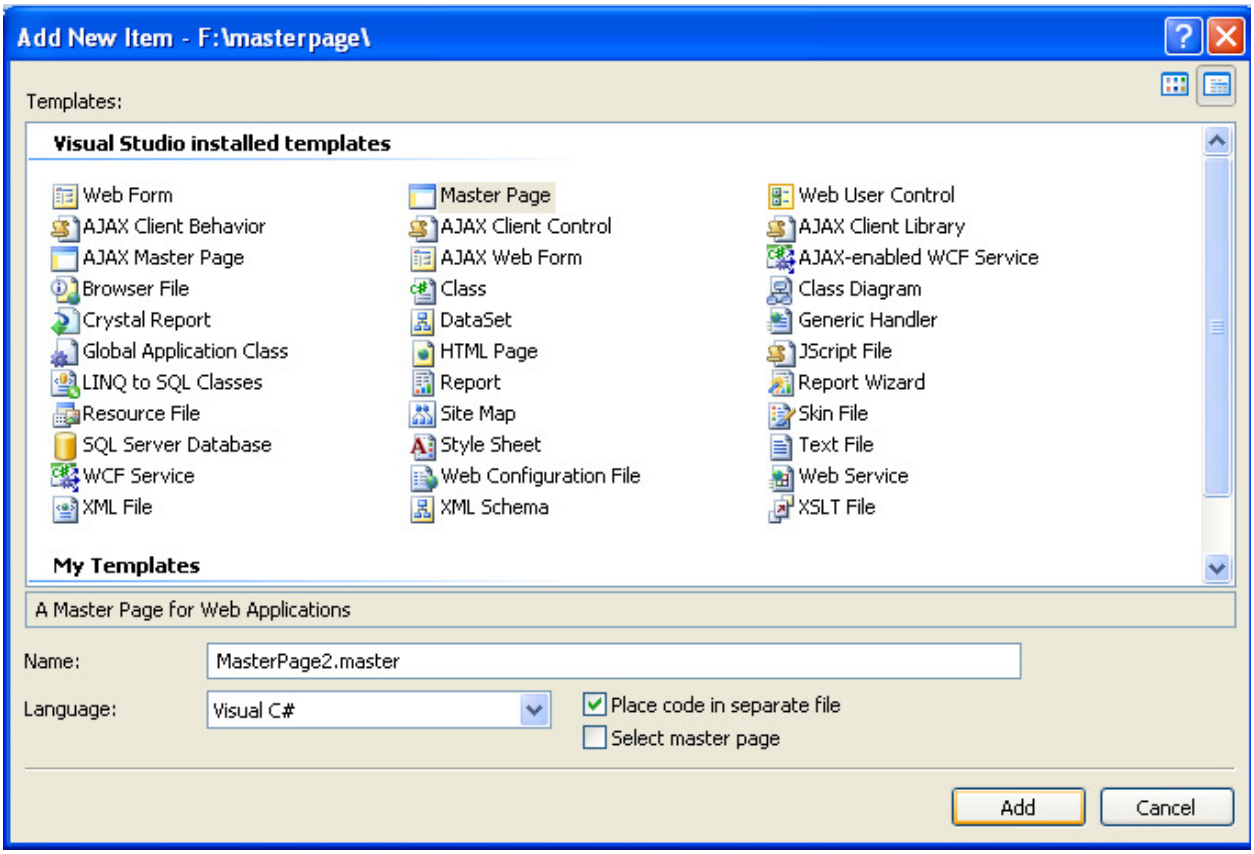
OUTPUT:-



Assignment 13) Demonstrate Master Pages and website navigation controls (sitemap path, treeview, menu) using SiteMap file.

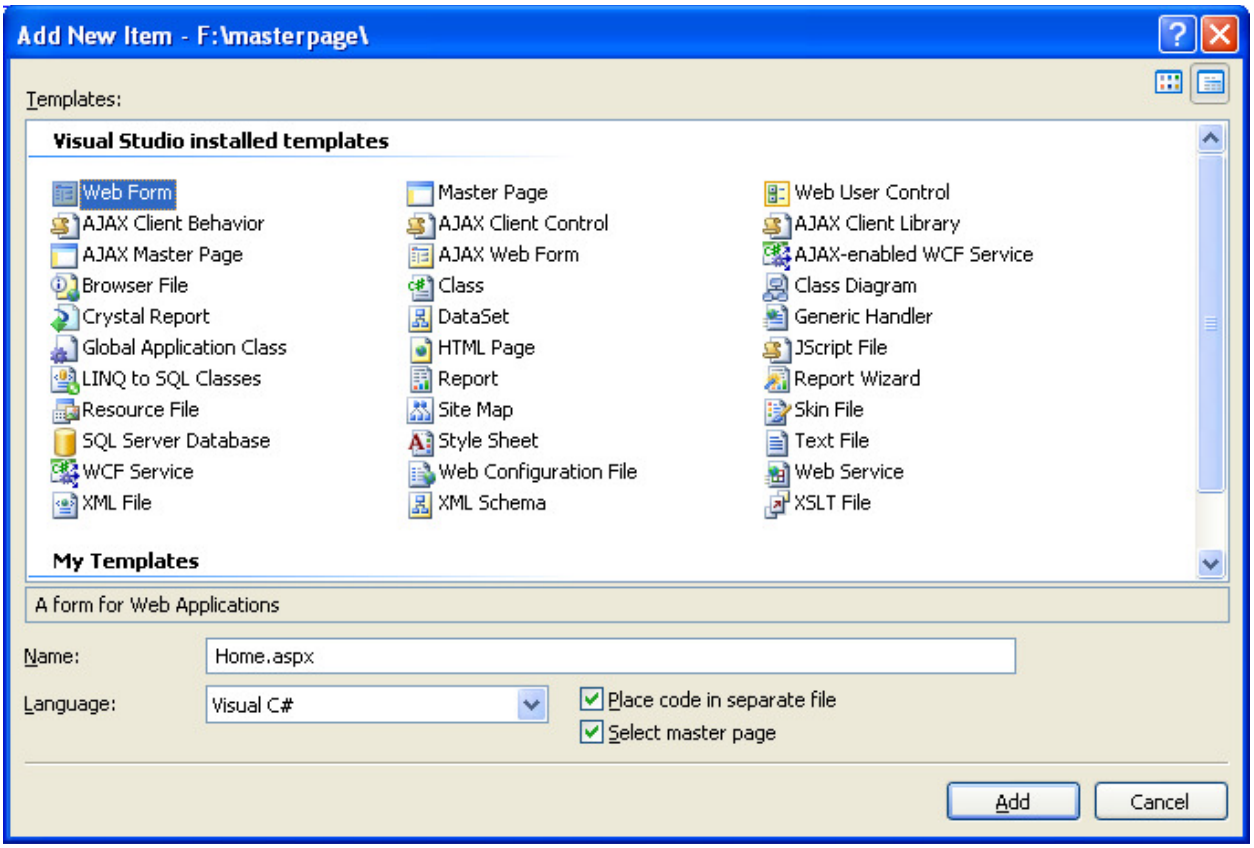
Assignment 14) Demonstrate Properties of website navigation controls.

1.Start->Programs->Visual studio 2008
2.Create new web site.
3.Right click on solution Explorer->Add New Items ->Select MasterPage and add MasterPage.master



4.Add Html Table in Master Page.
5. Drag and Drop ContentPlaceHolder into Middle Table Cell.
6. .Right click on solution Explorer->Add New Items ->Select Web Form and Add Home.aspx

Do not forget to Check Select Master Page CheckBox .

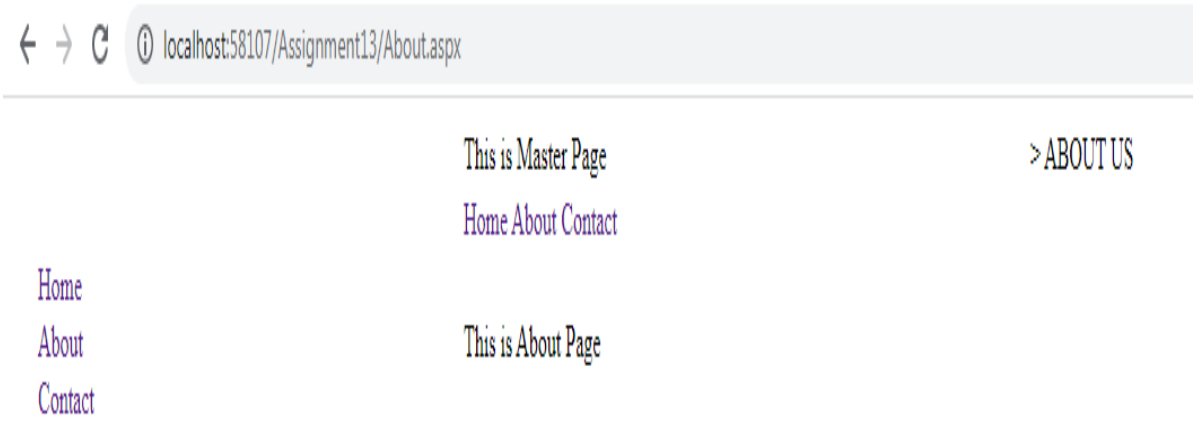


7. Similarly add Two More Content Pages Aboutus.aspx and Contactus.aspx
8. Add Menu Control in Top Most Cell of Table & Click on Smart tag -> Edit Menu Items and Add 3 menu items(Home,AboutUs,ContactUs). And set the Text and Navigate Url Properties of Menu Items.
9. Drag SiteMapDataSource Control from ToolBox and Drop it on Master Page.
10. Drag TreeView Control from ToolBox and Drop it on Master Page.
11. Write a following Code in SiteMapDataSource Control file.

```
<?xml version="1.0" encoding="utf-8" ?>
<siteMap xmlns="http://schemas.microsoft.com/AspNet/SiteMap-File-1.0" >
<siteMapNode url="" title="" description="">
<siteMapNode url="home.aspx" title="HOME" description="" />
<siteMapNode url="aboutus.aspx" title="ABOUT US" description="" />
<siteMapNode url="contact.aspx" title="CONTACT US" description="" />
</siteMapNode>
</siteMap>
```

12. Click on SmartTag on TreeView Control and Choose the DataSource->SiteMapDataSource1
13 . Drag SiteMapPath Control from ToolBox and Drop it on Master Page.
14. Save Built and Run.

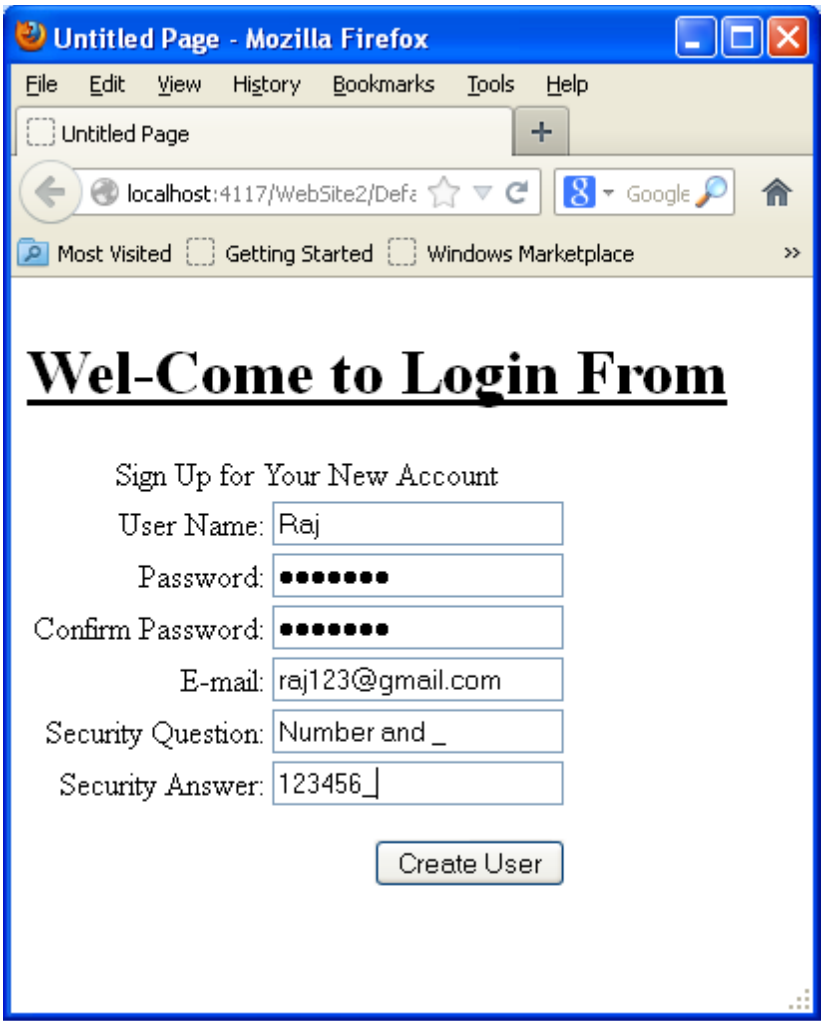
OUTPUT:-

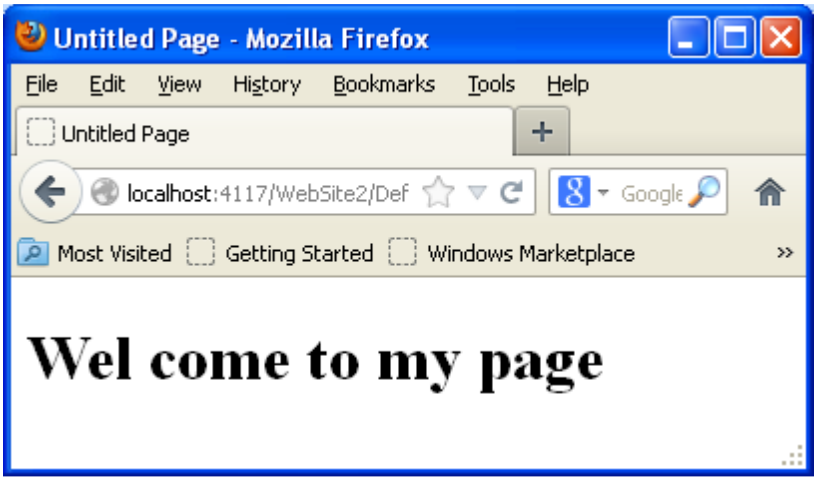


Assignment 15) Demonstrate Authorization/Authentication using Login controls and Roles/Membership/AccessRules

1) Create Microsoft Visual Studio 2008
2) Select File->New->WebSite
3) Select ASP.NET Web site template from the Templates pane and name it “UserRoleDemo”

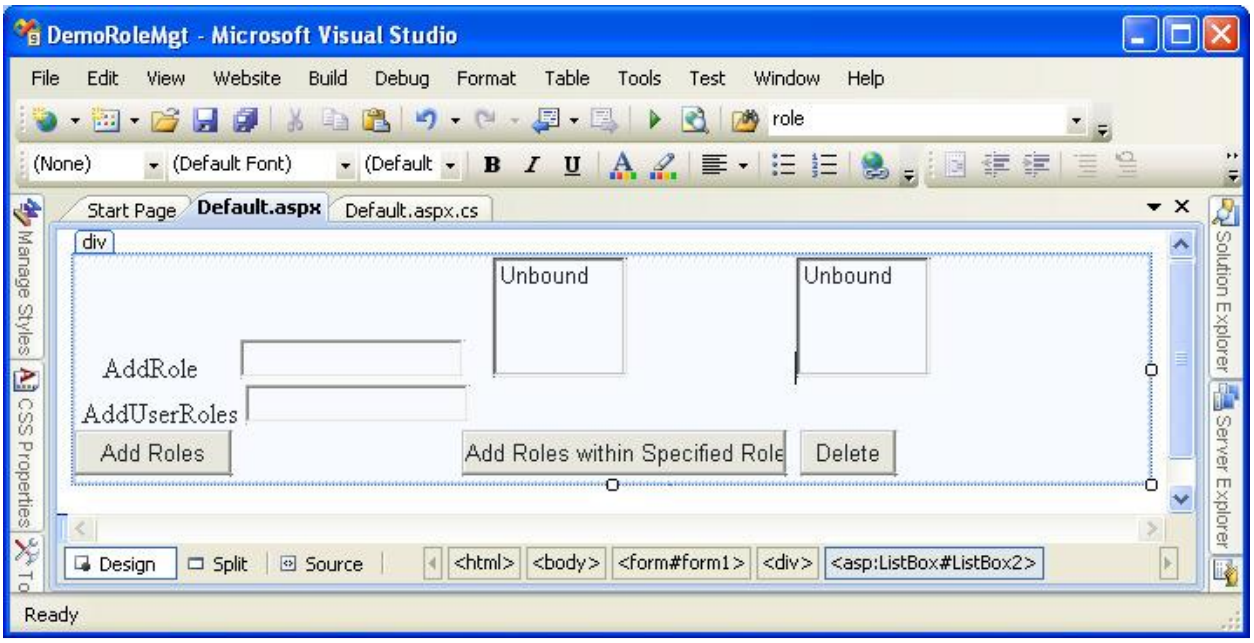
```
<form id="form1" runat="server">
<div>
   
   
<br />
<span class="style1">Wel-Come to Login From</span><br />
<br />
<asp:CreateUserWizard ID="CreateUserWizard1" runat="server"
OnContinueButtonClick="CreateUserWizard1_ContinueButtonClick">
<WizardSteps>
<asp:CreateUserWizardStep runat="server" />
<asp:CompleteWizardStep runat="server" />
</WizardSteps>
</asp:CreateUserWizard>
<br />
</div>
</form>
protected void CreateUserWizard1_ContinueButtonClick(object sender, EventArgs
e)
{
Response.Redirect("Default2.aspx");
}
```


Untitled Page - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Untitled Page
localhost:4117/WebSite2/Defa
Google
Most Visited
Getting Started
Windows Marketplace
Wel-Come to Login From
Sign Up for Your New Account
User Name: Raj
Password:
Confirm Password:
E-mail: raj123@gmail.com
Security Question: Number and _
Security Answer: 123456_
Create User



Role Management:-

7) Create **Microsoft Visual Studio 2008**
8) Select **File->New->WebSite**
9) Select ASP.NET Web site template from the Templates pane and name it "UserRoleDemo"
10) Select Website from Menu -> ASP.NET configuration
11) Go into web browser of ASP.NET configuration & select security & enable it.
12) Switch to the Web.`Config click on "Ok" Role are enable in your appliction`
13) `Select Default.aspx add following control`

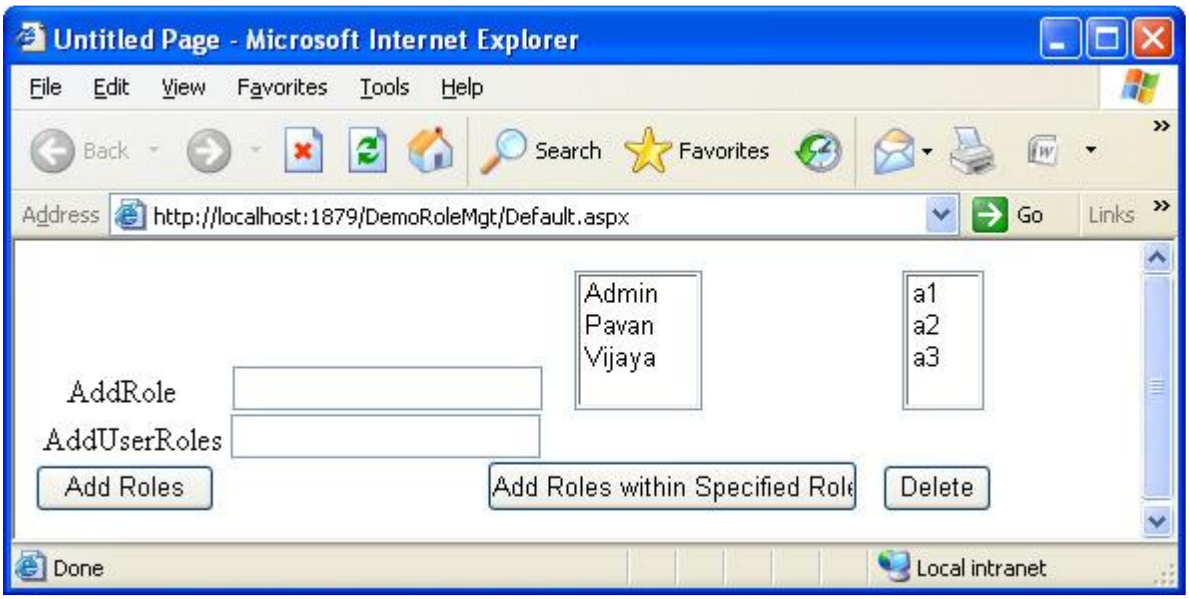


14) Write following code on Page_load & buttons click

```
public partial class _Default : System.Web.UI.Page
{
public string str;
protected void Page_Load(object sender, EventArgs e)
{
ListBox1.DataSource = Roles.GetAllRoles();
ListDataBind();
ListBox2.DataSource = Roles.GetUsersInRole("Admin");
ListBox2.DataBind();
}
protected void Button1_Click(object sender, EventArgs e)
{
Roles.CreateRole(TextBox1.Text);
ListDataBind();
}
protected void ListDataBind()
{
ListBox1.DataSource = Roles.GetAllRoles();
ListBox1.DataBind();
ListBox2.DataSource = Roles.GetUsersInRole("Admin");
ListBox2.DataBind();
}
protected void Button2_Click(object sender, EventArgs e)
{
Roles.AddUserToRole(TextBox2.Text, TextBox1.Text);
ListBox2.Items.Add(TextBox2.Text);
ListDataBind();
}
protected void Button3_Click(object sender, EventArgs e)
{
Roles.RemoveUserFromRole(TextBox2.Text, TextBox1.Text);
Response.Write("Role is deleted");
```

37

}
}
15) Save the program run it.



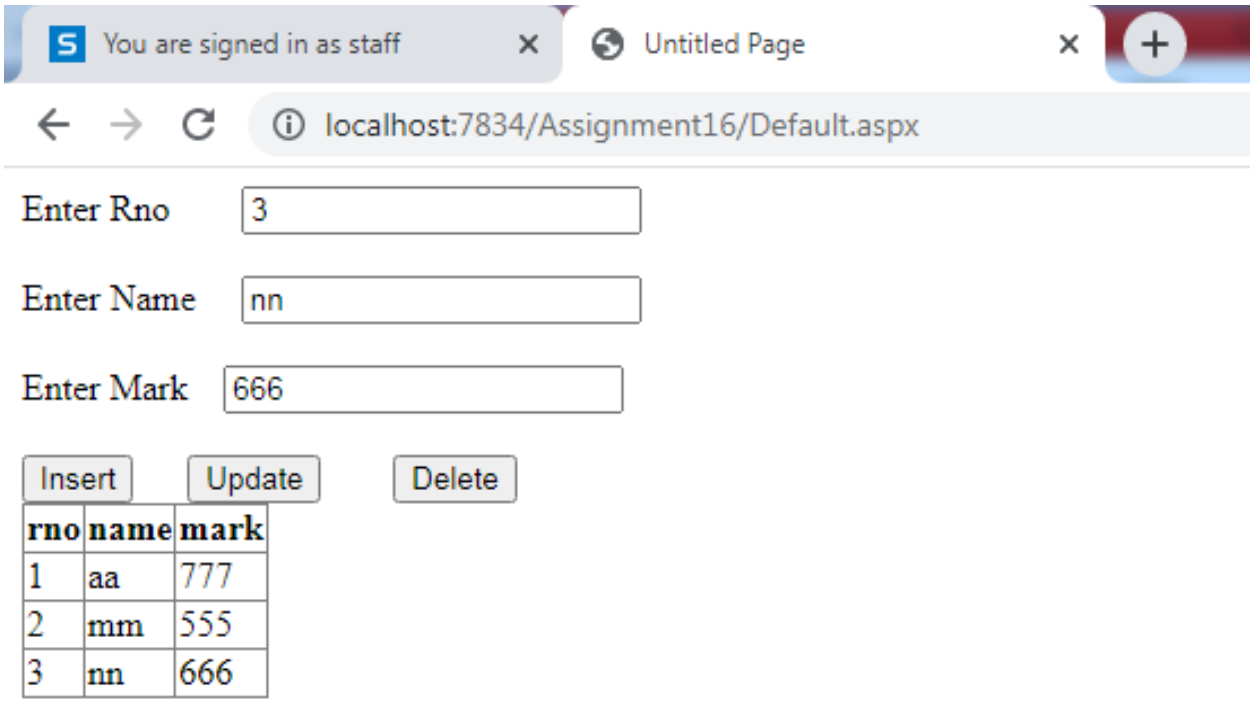
Assignment 16) Demonstrate creation of simple/complex DataReader/DataSet Objects.

1. Right Click on **App_Data** in **Solution Explorer**
2. Add New Item select **SQL Server Database** and give database name
3. In **Server Explorer** right click on **Tables,** select **Add New Table**
4. Right click on your table and select **Show Table Data** and insert records
5. Add 3 TextBoxes , 3 Buttons(INSERT,UPDATE,DELETE) & 1 GridView control to Default.aspx Page
6. Add Following Code
7.
8. **DataReader:-**

```
9. using System.Data;
10.      using System.Data.SqlClient;
11.
12.      public partial class _Default : System.Web.UI.Page
13.      {
14.
15.          SqlConnection con = new SqlConnection("Data
   Source=.\\SQLEXPRESS;AttachDbFilename=I:\\2021-22\\Subjects\\CA-305(B)
   Microsoft .Net Technologies\\CA-LAB-XII (B) Lab on Microsoft .Net
   Technologies\\Assignment16\\App_Data\\Database.mdf;Integrated
   Security=True;User Instance=True");
16.          SqlDataReader dr;
17.          protected void Page_Load(object sender, EventArgs e)
18.          {
```

```
19.             if (!IsPostBack)
20.                 getdata();
21.         }
22.         void getdata()
23.         {
24.             SqlCommand cmd = new SqlCommand("select * from student",
    con);
25.             con.Open();
26.             dr = cmd.ExecuteReader();
27.             GridView1.DataSource = dr;
28.             GridView1.DataBind();
29.             con.Close();
30.         }
31.
32.         protected void Button1_Click(object sender, EventArgs e)
33.         {
34.             SqlCommand cmd1 = new SqlCommand("insert into student
    values(" + Convert
35.       .ToInt32(TextBox1.Text) + ",'" + TextBox2.Text + "'," + Convert
36.       .ToInt32(TextBox3.Text) + ")", con);
37.             con.Open();
38.             cmd1.ExecuteNonQuery();
39.             con.Close();
40.             getdata();
41.         }
42.         protected void Button2_Click(object sender, EventArgs e)
43.         {
44.             SqlCommand cmd2 = new SqlCommand("Update student set
    name='" + TextBox2.Text + "',mark=" +
45.       Convert.ToInt32(TextBox3.Text) + "where rno=" +
    Convert.ToInt32(TextBox1.Text), con);
46.             con.Open();
47.             cmd2.ExecuteNonQuery();
48.             con.Close();
49.             getdata();
50.         }
51.
52.         protected void Button3_Click(object sender, EventArgs e)
53.         {
54.             SqlCommand cmd3 = new SqlCommand("Delete from student
    Where rno=" +
55.       Convert.ToInt32(TextBox1.Text), con);
56.             con.Open();
57.             cmd3.ExecuteNonQuery();
58.             con.Close();
59.             getdata();
60.         }
61.       }
```

OUTPUT:-



DataSet:-

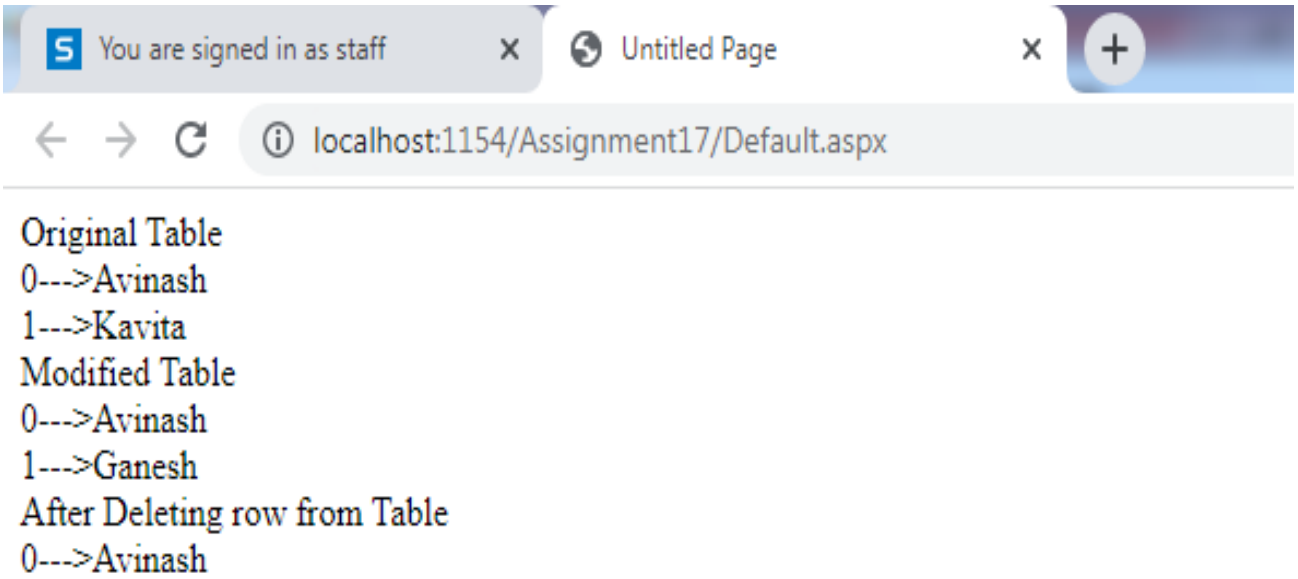
```
using System.Data;
using System.Data.SqlClient;
public partial class _Default : System.Web.UI.Page
{
    SqlConnection con = new SqlConnection("Data
Source=.\\SQLEXPRESS;AttachDbFilename=I:\\2021-22\\Subjects\\CA-305(B)
Microsoft .Net Technologies\\CA-LAB-XII (B) Lab on Microsoft .Net
Technologies\\Assignment16A\\App_Data\\Database.mdf;Integrated
Security=True;User Instance=True");
    protected void Page_Load(object sender, EventArgs e)
    {
        if (!IsPostBack)
            getdata();

    }
   void getdata()
    {
       DataSet ds = new DataSet();
       SqlDataAdapter sde = new SqlDataAdapter("select * from student", con);
               sde.Fill(ds);
                GridView1.DataSource = ds;
                GridView1.DataBind();

    }
}
```

Assignment 17) Demonstrate editing in DataTable objects.

```
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)
    {
        // Initializes a new instance of the System.Data.DataTable class with
the specified table name.
            DataTable tempTable = new DataTable();
           // Add columns in DataTable
            tempTable.Columns.Add("ID");
           tempTable.Columns.Add("NAME");
           // Add New Row in dataTable
         DataRow newrow = tempTable.NewRow();
          newrow[0] = 0;
           newrow[1] = "Avinash";
           tempTable.Rows.Add(newrow);
          // Add Another New Row in dataTable
            newrow = tempTable.NewRow();
            newrow[0] = 1;
           newrow[1] = "Kavita";
           tempTable.Rows.Add(newrow);
           //Now table is ready
           Response.Write("Original Table"+"<br>");

        for (int i = 0; i < tempTable.Rows.Count; i++)
            {
                Response.Write(tempTable.Rows[i][0].ToString() + "--->" +
tempTable.Rows[i][1].ToString()+"<br>");
            }
            // if you want edit data from DataTable
        Response.Write("Modified Table" + "<br>");
            for (int i = 0; i < tempTable.Rows.Count; i++)
           {
          if (tempTable.Rows[i][1].ToString() == "Kavita")
                {
                    tempTable.Rows[i][1] = "Ganesh";
                }
          Response.Write(tempTable.Rows[i][0].ToString() + "--->" +
tempTable.Rows[i][1].ToString() + "<br>");
            }
           // if you want delete row from DataTable
            Response.Write("After Deleting row from Table" + "<br>");
            for (int i = 0; i < tempTable.Rows.Count; i++)
```

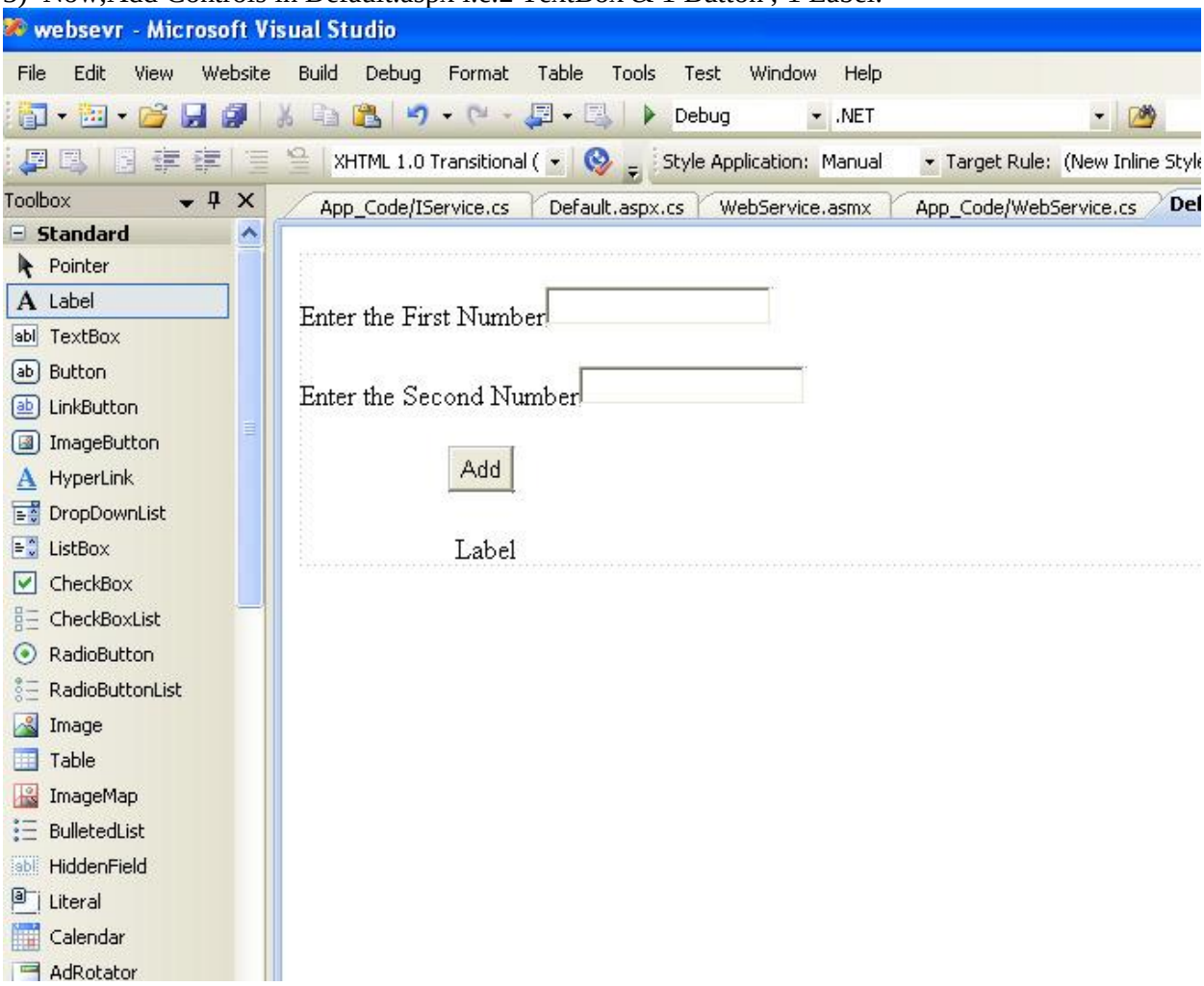
```
            {
                if (tempTable.Rows[i][1].ToString() == "Ganesh")
               {
                            tempTable.Rows[i].Delete();
                 }
                   }
            for (int i = 0; i < tempTable.Rows.Count; i++)
            {
                Response.Write(tempTable.Rows[i][0].ToString() + "--->" +
tempTable.Rows[i][1].ToString() + "<br>");
            }

    }
}
```

OUTPUT-



Assignment 18) Demonstrate Web Service hosting, access in ASP.NET

1) Create Microsoft Visual Studio 2008
2) Select File->New->WebSite
3) Now,Add Controls in Default.aspx i.e.2 TextBox & 1 Button , 1 Label.



5) Right-click web application's name in the Solution Explorer window and then select the Add New Item option from the context menu
6) Select **Web Service** template from the Templates pane and click the Add button.
7) Now Code for add() in WebService.cs is as below

```
[WebMethod]
public int add(int a,int b)
{
return a+b;
}
```

8) Go to Build Menu, Select Build Web Site and then run this code and test in Browser.
9) Right-click web application's name in the Solution Explorer window and then select the **Add Web Reference** option from the context menu.
10) Select **Web Services in this Solution** ,Give Web Reference name and then Add Reference.
11) Give **ws** name as reference.

12) Below Code write on Button Click in Default.aspx Page is

```
protected void Button1_Click1(object sender, EventArgs e)
{
int a = Convert.ToInt32(TextBox1.Text);
int b = Convert.ToInt32(TextBox2.Text);
ws.WebService w = new ws.WebService();
int c= w.add(a, b);
Label1.Text = c.ToString();
}
```

13) Save the program and run it.

OUTPUT-