



Sahyog College of Management Studies, Thane (W)

Affiliated to Mumbai University

Course : BSC (Information Technology)

Semester : V

Subject : ADVANCED WEB DEVELOPMENT

Lab Manual

Prepared By,
Prof. Deepa Mishra

Practical No : 1

1. Write the program for the following:

- a. Create an application to print on screen the output of adding, subtracting, multiplying and dividing two numbers entered by the user in C#.
 - b. Create an application to print Floyd's triangle till n rows in C#.
 - c. Create an application to demonstrate following operations
 - i. Generate Fibonacci series.
 - ii. Test for prime numbers.
-

- a. Create an application to print on screen the output of adding, subtracting, multiplying and dividing two numbers entered by the user in C#.

Solution :

```
using System;
namespace ConsoleApplication1
{
    class Program
    {
        static void Main(string[] args)
        {
            int num1, num2,a,s,m,d;
            Console.Write("Enter number 1: ");
            num1 = Int32.Parse(Console.ReadLine());
            Console.Write("Enter number 2: ");
            num2 = Convert.ToInt32(Console.ReadLine());

            a = num1 + num2 ;
            s = num1 - num2;
            m = num1 * num2;
            d = num1 / num2;
            Console.WriteLine("Addition="+a);
            Console.WriteLine("Subtraction=" + s);
            Console.WriteLine("Multiplication=" + m);
            Console.WriteLine("Division=" + d);
            Console.ReadKey();
        }
    }
}
```

Output:

```
Enter number 1: 15
Enter number 2: 6
Addition=21
Subtraction=9
Multiplication=90
Division=2
```

b. Create an application to print Floyd's triangle till n rows in C#.

Solution:

```
using System;
public class Exercise22
{
    public static void Main()
    {
        int i, j, n, p, q;
        Console.Write("Input number of rows : ");
        n = Convert.ToInt32(Console.ReadLine());
        for (i = 1; i <= n; i++)
        {
            if (i % 2 == 0)
            {
                p = 1;
                q = 0;
            }
            else
            {
                p = 0;
                q = 1;
            }

            for (j = 1; j <= i; j++)
            {
                if (j % 2 == 0)
                    Console.Write("{0}", p);
                else
                    Console.Write("{0}", q);
            }

            Console.Write("\n");
        }
    }
}
```

OUTPUT:

Input number of rows : 7

```
1
01
101
0101
10101
010101
1010101
```

c. Create an application to demonstrate following operations

i. Generate Fibonacci series. ii. Test for prime numbers

i. Generate Fibonacci series.

Solution:

```
using System;
namespace ConsoleApplication3
{
    class Program
    {
        static void Main(string[] args)
        {
            int num1 = 0, num2 = 1, num3, num;
            Console.Write("Upto how many number you want fibonacci series:");
            num = int.Parse(Console.ReadLine());

            Console.Write(num1 + "\t" + num2 + " ");
            for (int i=2;i<num;i++)
            {
                num3 = num1 + num2;

                Console.Write("\t" + num3);
                num1 = num2;
                num2 = num3;
            }
        }
    }
}
```

OUTPUT:

Upto how many number you want fibonacci series: 5
0 1 1 2 3

ii. Test for prime numbers

Solution:

```
using System;
public class PrimeNumberExample
{
    public static void Main(string[] args)
    {
        int n, i, m = 0, flag = 0;
        Console.Write("Enter the Number to check Prime: ");
        n = int.Parse(Console.ReadLine());
        m = n / 2;
        for (i = 2; i <= m; i++)
        {
            if (n % i == 0)
            {
```

```

        Console.WriteLine("Number is not Prime.");
        flag = 1;
        break;
    }
}
if (flag == 0)
    Console.WriteLine("Number is Prime.");
}
}

```

Output:

Enter the Number to check Prime: 17
Number is Prime.

Practical No : 2

2. Write the program for the following:

- a. Create a simple application to demonstrate the concepts boxing and unboxing.
- b. Create a simple application to perform addition and subtraction using delegate.
- c. Create a simple application to demonstrate use of the concepts of interfaces.

-
- a. Create a simple application to demonstrate the concepts boxing and unboxing.

Solution:

```

using System;
public class Exercise22
{
    public static void Main()
    {
        int num = 2020;
        // boxing
        object obj = num;
        num = 100;
        Console.WriteLine("BOXING");
        System.Console.WriteLine("value of num is "+num);
        System.Console.WriteLine("value of obj is " + obj);
        num = 23;
        obj = num;
        // unboxing
        int i = (int)obj;
        Console.WriteLine("UNBOXING");
        Console.WriteLine("Value of ob object is : " + obj);
        Console.WriteLine("Value of i is " + i);
    }
}

```

Output:

BOXING

value of num is 100

value of obj is 2020

UNBOXING

Value of ob object is :23

Value of i is 23

b. Create a simple application to perform addition and subtraction using delegate.

Solution:

using System;

delegate int NumberChanger(int n);

namespace example

```
{
    class Delegate
    {
        static int num = 10;
        public static int AddNum(int a)
        {
            num += a;
            return num;
        }

        public static int SubNum(int b)
        {
            num -= b;
            return num;
        }
        public static int getNum()
        {
            return num;
        }
    }

    static void Main(string[] args)
    {
        NumberChanger n1 = new NumberChanger(AddNum);
        NumberChanger n2 = new NumberChanger(SubNum);
        n1(25);
        Console.WriteLine("Addition= {0}", getNum());
        n2(5);
        Console.WriteLine("subtraction={0}", getNum());
        Console.ReadKey();
    }
}
```

Output:

Addition= 35

subtraction=30

c. Create a simple application to demonstrate use of the concepts of interfaces.

Solution:

```
using System;
public interface Drawable
{
    void draw();
}
public class Rectangle : Drawable
{
    public void draw()
    {
        Console.WriteLine("drawing rectangle...");
    }
}
public class Circle : Drawable
{
    public void draw()
    {
        Console.WriteLine("drawing circle...");
    }
}
public class TestInterface
{
    public static void Main()
    {
        Drawable d;
        d = new Rectangle();
        d.draw();
        d = new Circle();
        d.draw();
    }
}
```

Output:

```
drawing rectangle...
drawing circle...
```

Practical No : 3

- a. Create a simple web page with various server controls to demonstrate setting and use of their properties. (Example : AutoPostBack)
 - b. Create a simple application to demonstrate your vacation using calendar control.
 - c) Demonstrate the use of Treeview operations on the web form.
-

- a. Create a simple web page with various server controls to demonstrate setting and use of their properties. (Example : AutoPostBack)

Solution:

Source Code: - aspx file

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="WebForm1.aspx.cs" Inherits="WebApplication1.WebForm1" %>
```

```
<!DOCTYPE html>
```

```
<html xmlns="http://www.w3.org/1999/xhtml">
```

```
<head runat="server">
```

```
<title></title>
```

```
<style type="text/css">
```

```
.style1
```

```
{
```

```
width:100%;
```

```
}
```

```
.style2
```

```
{
```

```
width:153px;
```

```
}
```

```
.style3
```

```
{
```

```
width:153px; height:26px;
```

```
}
```

```
.style4
```

```
{
```

```
height:26px;
```

```
}
```

```
.auto-style1 { width: 153px; height: 28px;
```

```
}
```

```
.auto-style2
```

```
{
```



```

height: 28px;
}
</style>
</head>
<body>
<form id="form1" runat="server">
<div>
<table class="style1">
<tr>
<td class="style2">
<asp:Label ID="Label1" runat="server" Text="Roll No">
</asp:Label>
</td>
<td>
<asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>
</td>
</tr>
<tr>
<td class="style2">
<asp:Label ID="Label2" runat="server" Text="Name">
</asp:Label></td>
<td>
<asp:TextBox ID="TextBox2" runat="server"></asp:TextBox>
</td>
</tr>
<tr>
<td class="auto-style1">Class</td>
<td class="auto-style2">
<asp:RadioButton ID="RadioButton1" runat="server" Text="FY"/>
<asp:RadioButton ID="RadioButton2" runat="server" Text="SY"/>
<asp:RadioButton ID="RadioButton3" runat="server" Text="TY"/>
</td>
</tr>
<tr>
<td class="style3">
<asp:Label ID="Label4" runat="server" Text="Course"></asp:Label>
</td>
<td class="style4">
<asp:DropDownList ID="DropDownList1" runat="server">
<asp:ListItem>B.Com</asp:ListItem>
<asp:ListItem>BMS</asp:ListItem>
<asp:ListItem>B.SC(IT)</asp:ListItem>
<asp:ListItem>B.Sc(CS)</asp:ListItem>

```

[illegible]

Source code : cs file

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

namespace WebApplication1
{
    public partial class WebForm1 : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
        }

        protected void DropDownList1_SelectedIndexChanged1(object sender, EventArgs e)
        {
        }

        protected void Button1_Click(object sender, EventArgs e)
        {
            string s;
            if (RadioButton1.Checked == true)
            {
                s = RadioButton1.Text;
            }
            else if (RadioButton2.Checked == true)
            {
                s = RadioButton1.Text;
            }
        }
    }
}
```

```

}
else
{
s = RadioButton3.Text;

}
Label5.Text = "You have been enrolled in " + DropDownList1.SelectedItem; Label5.Text +=
" in " + s;
}
}
}
}

```

Output:-

localhost:50423/WebForm1.aspx

Roll No: 7122

Name: Harshal

Class: ☐ FY ☐ SY ☒ TY

Course: B.SC(IT)

Button

You have been enrolled in B.SC(IT) in TY

b. Create a simple application to demonstrate your vacation using calendar control.

Solution:

calndrCtrl.aspx

July 2018						Aug
Mon	Tu	We	Th	Fr	Sa	Su
25	26	27	28	29	30	1
2	3	4	5	6	7	8
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

Your selected date : Label
 Today's Date : Label
 Ganpati Vacation Start: Label
 Days Remaining For Ganpati Vacation : Label
 Days remeaning for new year : Label

Result Reset

Calender properties set for this example:

```

<asp:Calendar ID="Calendar1" runat="server" BackColor="#FFFFCC"
BorderColor="#FFCC66" BorderWidth="1px" DayNameFormat="Shortest"

```

```

Font-Names="Verdana" Font-Size="8pt" ForeColor="#663399" Height="200px"
NextPrevFormat="ShortMonth" OnDayRender="Calendar1_DayRender"
ShowGridLines="True" Width="300px"
OnSelectionChanged="Calendar1_SelectionChanged" >
<DayHeaderStyle BackColor="#FFCC66" Font-Bold="True" Height="1px" />
<NextPrevStyle BorderStyle="Solid" BorderWidth="2px" Font-Size="9pt"
ForeColor="#FFFFCC" />
<OtherMonthDayStyle BackColor="#FFCC99" BorderStyle="Solid"
ForeColor="#CC9966" />
<SelectedDayStyle BackColor="Red" Font-Bold="True" />
<SelectorStyle BackColor="#FFCC66" />
<TitleStyle BackColor="#990000" Font-Bold="True" Font-Size="9pt"
ForeColor="#FFFFCC" />

<TodayDayStyle BackColor="#FFCC66" ForeColor="White" />
<WeekendDayStyle Height="50px" />
</asp:Calendar>

```

calndrCtrl.aspx.cs

```

protected void btnResult_Click(object sender, EventArgs e)
{
    Calendar1.Caption = "SAMBARE";
    Calendar1.FirstDayOfWeek = FirstDayOfWeek.Sunday;
    Calendar1.NextPrevFormat = NextPrevFormat.ShortMonth;
    Calendar1.TitleFormat = TitleFormat.Month;

    Label2.Text = "Todays Date"+Calendar1.TodaysDate.ToShortDateString();
    Label3.Text = "Ganpati Vacation Start: 9-13-2018";

    if (Calendar1.SelectedDate.ToShortDateString() == "9-13-2018")
        Label3.Text = "<b>Ganpati Festival Start</b>";
    if (Calendar1.SelectedDate.ToShortDateString() == "9-23-2018")
        Label3.Text = "<b>Ganpati Festival End</b>";
}

protected void Calendar1_DayRender(object sender,
System.Web.UI.WebControls.DayRenderEventArgs e)
{
    if (e.Day.Date.Day == 5 && e.Day.Date.Month == 9)
    {
        e.Cell.BackColor = System.Drawing.Color.Yellow;
        Label lbl = new Label();

```

```

lbl.Text = "<br>Teachers Day!";
e.Cell.Controls.Add(lbl);
}
if (e.Day.Date.Day == 13 && e.Day.Date.Month == 9)
{
Calendar1.SelectedDate = new DateTime(2018, 9, 12);
Calendar1.SelectedDates.SelectRange(Calendar1.SelectedDate,
Calendar1.SelectedDate.AddDays(10));
Label lbl1 = new Label();
lbl1.Text = "<br>Ganpati!";
e.Cell.Controls.Add(lbl1);
}
}
protected void btnReset_Click(object sender, EventArgs e)
{
Label1.Text = "";
Label2.Text = "";
Label3.Text = "";
Label4.Text = "";
Label5.Text = "";
Calendar1.SelectedDates.Clear();
}

protected void Calendar1_SelectionChanged(object sender, EventArgs e)
{
Label1.Text = "Your Selected Date:" + Calendar1.SelectedDate.Date.ToString();
}

```

Output:

SAMBARE

September						
Su	Mo	Tu	We	Th	Fr	Sa
26	27	28	29	30	31	1
2	3	4	5 Teachers Day!	6	7	8
9	10	11	12	13 Ganpati!	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	1	2	3	4	5	6

Your selected date : 27-08-2018 00:00:00
 Today's Date : 15-07-2018
 Ganpati Vacation Start : 9-13-2018
 Days Remaining For Ganpati Vacation : 59
 Days remeaning for new year : 168

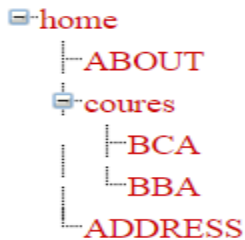
Result Reset

c. Demonstrate the use of Treeview operations on the web form.

Solution:

```
<asp:TreeView ID="TreeView1" runat="server" ShowLines="True" ForeColor="#CC0000"
PathSeparator="&lt;">
  <Nodes>
    <asp:TreeNode Text="home" Value="New Node" NavigateUrl="~/HOME.aspx">
    <asp:TreeNode NavigateUrl="~/ABOUT.aspx" Text="ABOUT"
Value="ABOUT"></asp:TreeNode>
    <asp:TreeNode Text="coures" Value="New Node" NavigateUrl="~/course.aspx">
    <asp:TreeNode Text="BCA" Value="New Node" NavigateUrl="~/BCA.aspx">
    </asp:TreeNode>
    <asp:TreeNode NavigateUrl="~/BBA.aspx" Text="BBA" Value="BBA"></asp:TreeNode>
    </asp:TreeNode>
    <asp:TreeNode Text="ADDRESS" Value="ADDRESS"
NavigateUrl="~/ADDRESS.aspx"></asp:TreeNode>
    </asp:TreeNode>
  </Nodes>
</asp:TreeView>
```

Output:



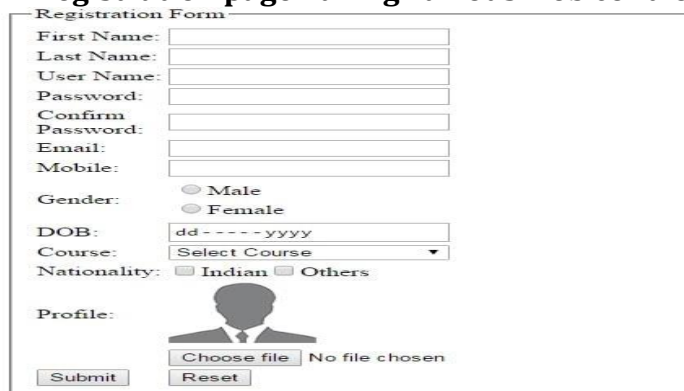
Practical No : 4

- Create a Registration form to demonstrate use of various Validation controls.
- Create Web Form to demonstrate use of Adrotator Control.
- Create Web Form to demonstrate use User Controls.

- Create a Registration form to demonstrate use of various Validation controls.

Solution:

1. Registration page having various web controls



Student.aspx:

```
<!DOCTYPE html>
<script runat="server">
    Protected Sub dob_SelectionChanged(sender As Object, e As EventArgs)
        End Sub
</script>
<head>
    <title>Registration Page</title>
</head>
<body>
    <form id="f1" method="post" runat="server">
        <fieldset style="width:280px">
            <legend>Registration Form</legend>
            <table>
                <tr>
                    <td>First Name:</td><td> <asp:textbox id="txt1" runat="server"
></asp:textbox></td>
                    <td> <asp:RequiredFieldValidator ID="validfname" runat="server"
ControlToValidate="txt1" ErrorMessage="Required!"
ForeColor="Red"></asp:RequiredFieldValidator></td>
                </tr>
```

```

        <tr>
            <td>Last Name:</td><td> <asp:textbox id="txt2" runat="server"
></asp:textbox></td>
            <td><asp:RequiredFieldValidator ID="validlname" runat="server"
ControlToValidate="txt2" ErrorMessage="Required!"
ForeColor="Red"></asp:RequiredFieldValidator></td>
        </tr>
        <tr>
            <td>User Name:</td><td> <asp:textbox id="user"
runat="server"></asp:textbox></td>
            <td><asp:RequiredFieldValidator ID="validuser" runat="server"
ControlToValidate="user" ErrorMessage="Required!"
ForeColor="Red"></asp:RequiredFieldValidator></td>
        </tr>
        <tr>
            <td>Password:</td><td><asp:textbox ID="pwd" runat="server"
TextMode="Password"></asp:textbox></td>
            <td><asp:RequiredFieldValidator ID="validpwd" runat="server"
ControlToValidate="pwd" ErrorMessage="Required!"
ForeColor="Red"></asp:RequiredFieldValidator></td>
        </tr>
        <tr>
            <td>Confirm Password:</td><td><asp:textbox ID="Textbox1"
runat="server" TextMode="Password"></asp:textbox></td>
        </tr>
        <tr>
            <td>Email:</td><td><asp:TextBox ID="email" runat="server"
TextMode="Email" ></asp:TextBox></td>
            <td><asp:RequiredFieldValidator ID="validemail" runat="server"
ControlToValidate="email" ErrorMessage="required!"
ForeColor="Red"></asp:RequiredFieldValidator></td>
        </tr>
        <tr>
            <td>Mobile:</td><td><asp:TextBox ID="mobile" runat="server" TextMode
="Phone"></asp:TextBox></td>
        </tr>
        <tr>
            <td>Gender:</td><td><asp:RadioButtonList ID="RadioButtonList1"
runat="server">
                <asp:ListItem Text="Male" Value="0"></asp:ListItem>
                <asp:ListItem Text="Female" Value="1"></asp:ListItem>
            </asp:RadioButtonList></td>
        </tr>

```

```


        <tr>
            <td>DOB: </td><td><asp:TextBox ID="dob" runat="server"
TextMode="Date" Width="168px"></asp:TextBox> </td>
            <td><asp:RequiredFieldValidator ID="validdob" runat="server"
ControlToValidate="dob" ErrorMessage="Required"
ForeColor="Red"></asp:RequiredFieldValidator></td>
        </tr>
        <tr>
            <td>Course: </td><td><asp:DropDownList ID="ddlCourse" runat="server"
datavaluefield="Course" Width="173px">
                <asp:ListItem text="Select Course" Value="-1"></asp:ListItem>
                <asp:ListItem Text ="BTech" Value ="0"></asp:ListItem>
                <asp:ListItem Text ="MCA" Value ="1"></asp:ListItem>
                <asp:ListItem Text ="MBA" Value="2"></asp:ListItem>
            </asp:DropDownList></td>
            <td><asp:RequiredFieldValidator InitialValue="-1" ID="validcourse"
runat="server" ControlToValidate="ddlCourse" ErrorMessage="Required!"
ForeColor="Red"></asp:RequiredFieldValidator></td>
        </tr>
        <tr>
            <td>Nationality:</td><td><asp:CheckBox ID="check" Text="Indian"
runat="server"/><asp:CheckBox id="checkNat" Text="Others" runat="server" /></td>
        </tr>
        <tr>
            <td>Profile: </td><td><asp:Image id="img" ImageUrl="images/new/new-
member.png" runat="server" /></td>
        </tr>
        <tr>
            <td></td><td><asp:FileUpload ID="imgupload" runat="server"
Enabled="true" /></td>
        </tr>
        <tr>
            <td><asp:Button ID="btn1" runat="server"
Text="Submit"></asp:Button></td>
            <td><asp:Button ID="btn2" runat="server"
Text="Reset"></asp:Button></td>
        </tr>
    </table>
</fieldset>
</form>
</body>

```

Output:

Validation: When we submit the page without filling any record.

The screenshot shows a 'Registration Form' with the following fields and validation messages:

- First Name: Required!
- Last Name: Required!
- User Name: Required!
- Password: Required!
- Confirm Password:
- Email: required!
- Mobile:
- Gender: ☐ Male ☐ Female
- DOB: Required
- Course: Required!
- Nationality: ☐ Indian ☐ Others
- Profile: 
 No file chosen
-

b. Create Web Form to demonstrate use of Adrotator Control.

Solution:

XML File

`<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>`

`<AlternateText>Order roses and flowers</AlternateText>`

`<Impressions>20</Impressions>`

`<Keyword>gifts</Keyword>`

`</Ad>`

`<Ad>`

`<ImageUrl>rose3.jpeg</ImageUrl>`

`<NavigateUrl>http://www.flowers2moscow.com</NavigateUrl>`

`<AlternateText>Send flowers to Russia</AlternateText>`

`<Impressions>20</Impressions>`

`<Keyword>russia</Keyword>`

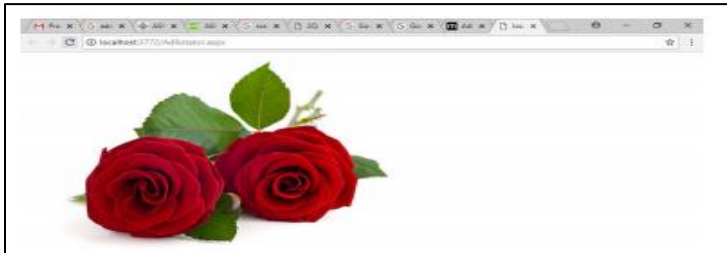
`</Ad>`

</Advertisements>

Default.aspx

```
<asp:AdRotator ID="AdRotator1" runat="server" DataSourceID="XmlDataSource1" />
<asp:XmlDataSource ID="XmlDataSource1" runat="server"
DataFile="~/ADFILE.xml"></asp:XmlDataSource>
```

OUTPUT:



c. Create web form to demonstrate use User Controls.

Solution:

MyUserControl.ascx

```
<%@ Control Language="C#" AutoEventWireup="true"
CodeFile="MyUserControl.ascx.cs" Inherits="MyUserControl" %>
<h3>This is User Contro1 </h3>
<table>
<tr>
<td>Name</td>
<td>
<asp:TextBox ID="txtName" runat="server"></asp:TextBox>
</td>
</tr>
<tr>
<td>City</td>
<td><asp:TextBox ID="txtcity" runat="server"></asp:TextBox></td>
</tr>
<tr>
<td></td>
<td>
<td>
</td>
</tr>
<tr>
<td></td>
<td>
<asp:Button ID="txtSave" runat="server" Text="Save" onclick="txtSave_Click" />
```

```

</td>
</tr>
</table><br />
<asp:Label ID="Label1" runat="server" ForeColor="White" Text=" "></asp:Label>

```

MyUserControl.ascx.cs

```

protected void txtSave_Click(object sender, EventArgs e)
{
    Label1.Text = "Your Name is " + txtName.Text + " and you are from " +
    txtcity.Text;
}

```

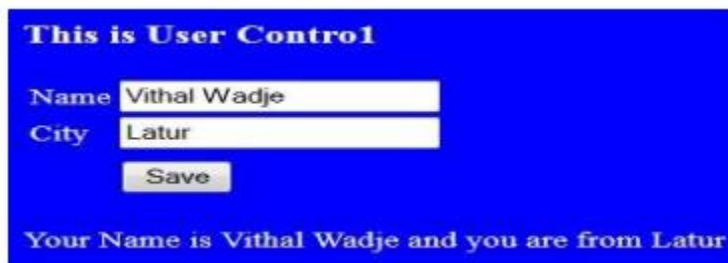
UserControlDisplay.aspx

```

<%@ Page Language="C#" AutoEventWireup="true"
CodeFile="UserControlDisplay.aspx.cs" Inherits="UserControlDisplay" %>
<%@ Register Src="~/MyUserControl.ascx" TagPrefix="uc"
TagName="Student"%>
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
<title></title>
</head>
<body>
<form id="form1" runat="server">
<div>
<uc:Student ID="studentcontrol" runat="server" />
</div>
</form>
</body>
</html>

```

OUTPUT :



This is User Control

Name

City

Your Name is Vithal Wadje and you are from Latur

Practical No :5

- a. Create Web Form to demonstrate use of Website Navigation controls and Site Map.
 - b. Create a web application to demonstrate use of Master Page and content page.
 - c. Create a web application to demonstrate various states of ASP.NET Pages.
-

- a. Create Web Form to demonstrate use of Website Navigation controls and Site Map.

Solution:

STEP 1:

Create 7 web forms

Home.aspx,
about.aspx,
contact.aspx,
course.aspx,
bca.aspx,
bba.aspx

STEP 2:

Add a sitemap file add following code:

```
<?xml version="1.0" encoding="utf-8" ?>
<siteMap xmlns="http://schemas.microsoft.com/AspNet/SiteMap-File-1.0" >

  <siteMapNode url="~/home.aspx" title="HOME" description="">

    <siteMapNode url="~/about.aspx" title="about" description="course">
</siteMapNode>

    <siteMapNode url="~/course.aspx" title="course" description="">
      <siteMapNode url="~/bca.aspx" title="bca"/>
      <siteMapNode url="~/bba.aspx" title="bba"/>
</siteMapNode>

    <siteMapNode url="~/ADDRESS.aspx" title="ADDRESS" description="Address">
</siteMapNode>
    <siteMapNode url="~/contact.aspx" title="contact" description="contact">
</siteMapNode>
  </siteMapNode>
```

</siteMap>

Step3:

Drag sitemappath control in each form.

Home.aspx:

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="home.aspx.cs"
Inherits="sitemapp.WebForm1" %>
<!DOCTYPE html>
```

```
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            <asp:SiteMapPath ID="SiteMapPath1" runat="server"
RenderCurrentNodeAsLink="True" ParentLevelsDisplayed="3">
</asp:SiteMapPath>
            <h1>home page</h1>

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

Contact.aspx:

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="contact.aspx.cs"
Inherits="sitemapp.contact" %>
```

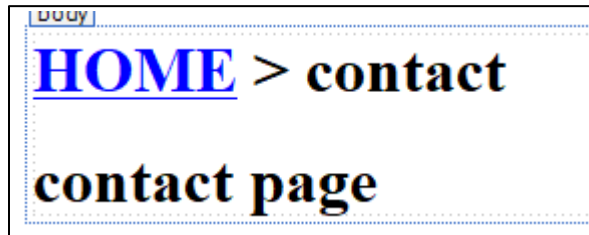
```
<!DOCTYPE html>
```

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

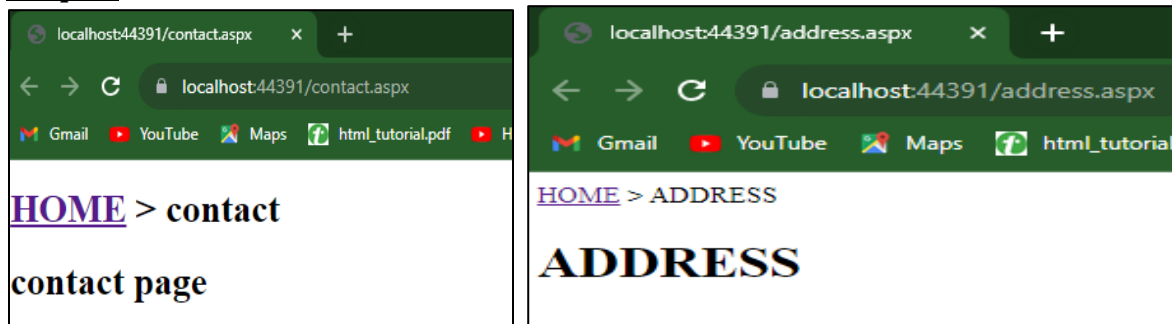
```

        </h1>
        <h1>contact page</h1>
    </div>
</form>
</body>
</html>

```



Output:



- b. Create a web application to demonstrate use of Master Page with applying Styles and Themes for page beautification.**

Solution:

MasterPage.master

```

<%@ Master Language="C#" AutoEventWireup="true"
CodeFile="MasterPage.master.cs"
Inherits="MasterPage" %>
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">

<title>Master Page Demo</title>
<link href="css/my.css" rel="stylesheet" />
<asp:ContentPlaceHolder ID="head" runat="server">
</asp:ContentPlaceHolder>

```

```

<style type="text/css">
.auto-style1 {
position: absolute;
top: 373px;
left: 1028px;
bottom: 303px;
}
.auto-style2 {
position: absolute;
top: 537px;
left: 1016px;
z-index: 1;
}
</style>
</head>
<body>
<!DOCTYPE html>
<form id="form1" runat="server">
<html>
<head>
<title>Master</title>
<link rel="stylesheet" type="text/css" href="StyleSheet.css">
</head>
<body>
<header id="header">
<h1>Demo Of Master Page</h1>
</header>
<nav id="nav">
<ul>
<li><a href="home.aspx">Insight</a></li>
<li><a href="#">Products</a></li>
<li><a href="#">Downloads</a></li>
<li><a href="#">Contact Us</a></li>
</ul>
</nav>
<aside id="side">
<h1>Info</h1>
<a href="#"><p>Product Type 1</p></a>
<a href="#"><p>Product Type 2</p></a>
<a href="#"><p>Product Type 3<a href="#"><asp:ScriptManager
ID="ScriptManager1"
runat="server">
</asp:ScriptManager>

```



```

</a>
</p>
<asp:Button ID="Button2" runat="server" CssClass="auto-style1" style="z-index: 1"
Text="Button" />
<asp:Button ID="Button1" runat="server" CssClass="auto-style2" Text="Button" />
</aside>
<div id="con">

<asp:ContentPlaceHolder ID="ContentPlaceHolder1" runat="server">
</asp:ContentPlaceHolder>
</div>
<footer id="footer">
copyright @Sambare
</footer>
</body>
</html>
</form>
</body>
</html>

```

MasterDisplay.aspx

```

<%@ Page Title="" Language="C#" MasterPageFile="~/MasterPage.master"
AutoEventWireup="true" CodeFile="MasterDisplay.aspx.cs" Inherits="MasterDisplay"
%>
<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1"
runat="server">
<h1>Home page</h1>
</asp:Content>

```

StyleSheet.css

```

#header{
color: blueviolet;
text-align: center;
font-size: 20px;
}
#nav{
background-color:darkseagreen;
padding: 5px;
}
ul{
list-style-type: none;

```

```

}
li a {
color:crimson ;
font-size: 30px;
column-width: 5%;
}
li
{
display: inline;
padding-left: 2px;
column-width: 20px;
}
a{
text-decoration: none;
margin-left:20px
}

li a:hover{
background-color: aqua;
color:coral ;
padding:1%;
}
#side{
text-align: center;
float: right;
width: 15%;
padding-bottom: 79%;
background-color: #F1FAEE;
}
#article{
background-color: burlywood;
padding: 10px;
padding-bottom: 75%;
}
#footer{
background-color: #C7EFCF;
text-align:center;
padding-bottom: 5%;
font-size: 20px;
}
#con{
border:double;
border-color:burlywood;

```

}

Output:



c. Create a web application to demonstrate various states of ASP.NET Pages.

Solution:

Cookies Program:

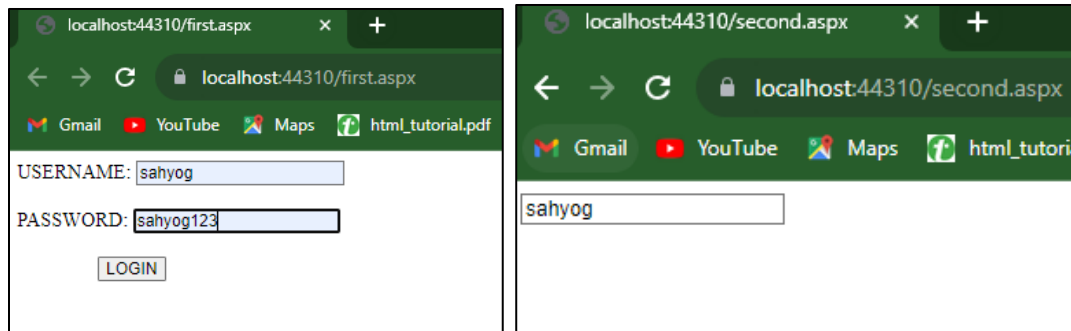
FIRST.ASPX:

```
HttpCookie hp = new HttpCookie("LOgin Details");  
hp["username"] = TextBox1.Text;  
hp["password"] = TextBox2.Text;
```

```
Response.Cookies.Add(hp);  
hp.Expires = DateTime.Now.AddSeconds(100);  
Response.Write("cookies added");  
Response.Redirect("second.aspx");
```

SECOND.ASPX:

```
protected void Page_Load(object sender, EventArgs e)  
{  
    HttpCookie c = Request.Cookies["LOgin Details"];  
    TextBox1.Text = c["username"].ToString();  
}
```



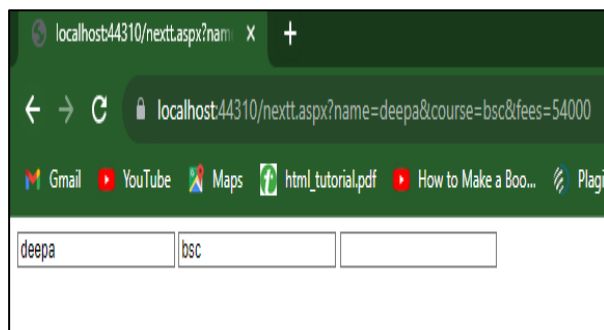
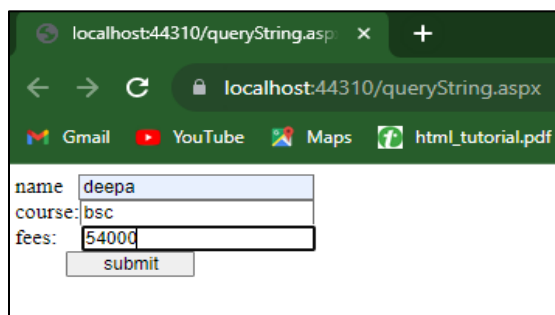
Query String Program:

First.aspx

```
protected void Button1_Click(object sender, EventArgs e)
{
    Response.Redirect("nextt.aspx?name=" + TextBox1.Text + "&course="
    + TextBox2.Text + "&fees=" + TextBox3.Text);
}
```

Next.aspx

```
protected void Page_Load(object sender, EventArgs e)
{
    string n = Request.QueryString["name"];
    TextBox1.Text = n;
    TextBox2.Text = Request.QueryString["course"];
}
```



Session Program:

LOGIN PAGE

namespace sessions

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

```

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

}
protected void Button1_Click(object sender, EventArgs e)
{

    if(TextBox1.Text=="sahyog" && TextBox2.Text=="sahyog123")
    {
        Session["user"] = TextBox1.Text;
        Session["pswd"] = TextBox2.Text;
        Response.Redirect("enroll.aspx");
    }
    else
    {
        TextBox1.Focus();
    }
}
}
}

```

ENROLL PAGE

namespace sessions

```

{
public partial class enroll : System.Web.UI.Page
{
protected void Page_Load(object sender, EventArgs e)
{
    if(Session["user"] ==null)
    {
        Response.Redirect("Login.aspx");
    }
    else
    {
        Label1.Text = "HELLO " + Session["user"];
    }
}

protected void Button1_Click(object sender, EventArgs e)
{
    Session["course"] = DropDownList1.SelectedItem;
    Response.Redirect("third.aspx");
}
}

```

```
}  
}  
}
```

THIRD.ASPX

```
protected void PREV_Click(object sender, EventArgs e)  
{  
    if(Session["user"]==null || Session["pswd"]==null)  
    {  
        Response.Redirect("LOGIN.ASPX");  
    }  
    else  
    {  
        //Label1.Text = Session["course"].ToString();  
        Response.Redirect("enroll.ASPX");  
    }  
}  
  
protected void LOGOUT_Click(object sender, EventArgs e)  
{  
    Response.Redirect("logout.aspx");  
}
```

LOGOUT.ASPX

```
namespace sessions  
{  
    public partial class logout : System.Web.UI.Page  
    {  
        protected void Page_Load(object sender, EventArgs e)  
        {  
            if(Session["user"]!=null)  
            {  
                Session.Abandon();  
                Response.Redirect("login.aspx");  
            }  
        }  
    }  
}
```

Practical No : 6

- Create a web application for inserting and deleting records from a database.
- Create a web application to display Using Disconnected Data Access and Databinding using GridView.

a. Create a web application for inserting and deleting records from a database.
Solution:

Web Form and employee table Design:

	eid	ename	desig	addrs	salary
1	101	Nitesh	Manager	kurla	54000
2	102	priyanka	developer	thane	74000
3	103	Kriti	designer	mumbai	59000
4	110	Nihanka	CEO	Pune	89000

Solution:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data.SqlClient;
namespace WebApplication1
{
    public partial class WebForm1 : System.Web.UI.Page
    {
        string cs = "data source=DESKTOP-8QHF2UM;initial catalog=college;integrated
security=true";
        SqlConnection con;
        protected void Page_Load(object sender, EventArgs e)
        {
            con = new SqlConnection(cs);
            if(IsPostBack==false)
            {
```

```

        string q = "select eid from employee";
        SqlCommand cmd = new SqlCommand(q, con);
        con.Open();
        SqlDataReader dr = cmd.ExecuteReader();
        dr.Read();

        DropDownList1.DataSource = dr;
        DropDownList1.DataTextField = "eid";
        DropDownList1.DataBind();
        con.Close();
    }
}
protected void btn_insert_Click(object sender, EventArgs e)
{
    string query = "insert into employee
values('" + eid.Text + "','" + ename.Text + "','" + desig.Text + "','" + address.Text + "','" + salary.Text +
"')";
    SqlCommand cmd = new SqlCommand(query, con);
    con.Open();
    cmd.ExecuteNonQuery();
    Response.Write("Data Inserted");
    con.Close();
}

protected void DropDownList1_SelectedIndexChanged(object sender, EventArgs e)
{
    string q1 = "select ename from employee where eid='" +
DropDownList1.SelectedValue + "'";
    SqlCommand cmd1 = new SqlCommand(q1, con);
    con.Open();
    SqlDataReader dr1 = cmd1.ExecuteReader();
    dr1.Read();
    emp_name.Text = dr1[0].ToString();
    con.Close();
}

protected void btn_del_Click(object sender, EventArgs e)
{
    string q2 = "delete from employee where eid='" + DropDownList1.SelectedValue + "'";
    SqlCommand cmd2 = new SqlCommand(q2, con);
    con.Open();
    cmd2.ExecuteNonQuery();
    Response.Write("record deleted");
}

```



```

        con.Close();
    }
}

```

Output:

Data Inserted

Employee Id:

Name:

designation:

Address:

Salary:

Select ID:

Employee Name:

	eid	ename	desig	addrs	salary
1	101	Nitesh	Manager	kurla	54000
2	102	priyanka	developer	thane	74000
3	103	Kriti	designer	mumbai	59000
4	110	Niharika	CEO	Pune	89000
5	115	Rohan	Developer	Mumbra	32000

record deleted

Employee Id:

Name:

designation:

Address:

Salary:

Select ID:

Employee Name:

	eid	ename	desig	addrs	salary
1	101	Nitesh	Manager	kurla	54000
2	102	priyanka	developer	thane	74000
3	110	Niharika	CEO	Pune	89000
4	115	Rohan	Developer	Mumbra	32000

b. Create a web application to display Using Disconnected Data Access and Databinding using GridView.

Solution:

```

using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data.SqlClient;
using System.Data;
namespace database1
{
    public partial class dataset : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            string cs = "data source=DESKTOP-16RE47P\\SQLEXPRESS;initial
catalog=college1;uid=sa;pwd=deepa123";

```

```

SqlConnection conn = new SqlConnection(cs);
SqlCommand cmd = new SqlCommand("select * from student", conn);
SqlDataAdapter da = new SqlDataAdapter(cmd);
DataSet ds = new DataSet();
da.Fill(ds);
GridView1.DataSource = ds;
GridView1.DataBind();
    }
}
}

```

Output:

roll	name	course
101	deepa	bca
12	sahyog	MBA
54	deepa	MBA
32	poonam	bba
13	neha	mbbs
13	riya	bams
54	giya	ba
12	priya	bba
100	xyz	bca

Practical No : 7

- Create a web application to demonstrate the use of different types of Cookies.
- Create a web application to demonstrate Form Security and Windows Security with proper Authentication and Authorization properties.

-
- Create a web application to demonstrate the use of different types of Cookies.

Solution:

FIRST.ASPX:

```

HttpCookie hp = new HttpCookie("Login Details");
hp["username"] = TextBox1.Text;
hp["password"] = TextBox2.Text;

```

```

Response.Cookies.Add(hp);

```

```

hp.Expires = DateTime.Now.AddSeconds(100);

```

```

Response.Write("cookies added");

```

```

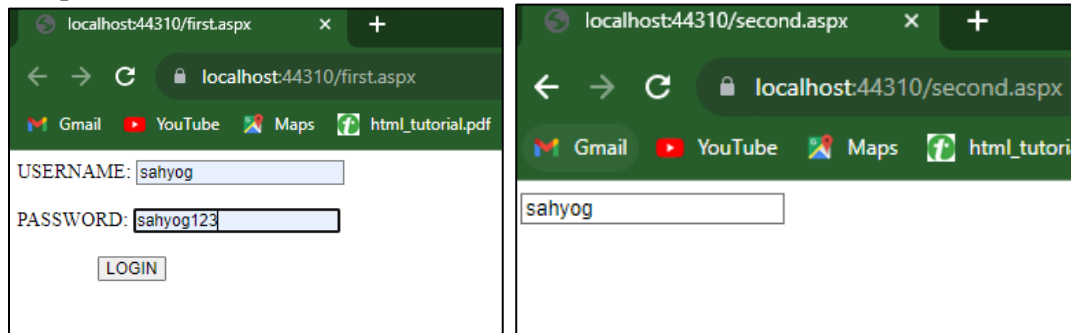
Response.Redirect("second.aspx");

```

SECOND.ASPX:

```
protected void Page_Load(object sender, EventArgs e)
{
    HttpCookie c = Request.Cookies["Login Details"];
    TextBox1.Text = c["username"].ToString();
}
```

Output:



- b. Create a web application to demonstrate Form Security and Windows Security with proper Authentication and Authorization properties.**

Solution:

Form Security:

WEB.CONFIG:

```
<system.web>
<authentication mode="Forms">
<forms name="demo_cookie" loginUrl="form1.aspx" defaultUrl="welcome.aspx"
timeout="1">
    <credentials passwordFormat="Clear">
    <user name="deepa" password="deepa123"/>
    <user name="priya" password="125"/>
    </credentials>
</forms>
</authentication>
<authorization>
    <deny users="?" />
</authorization>
</system.web>
```

LOGIN.ASPX

USERNAME:

PASSWORD:

☐ REMEMBER ME

Label

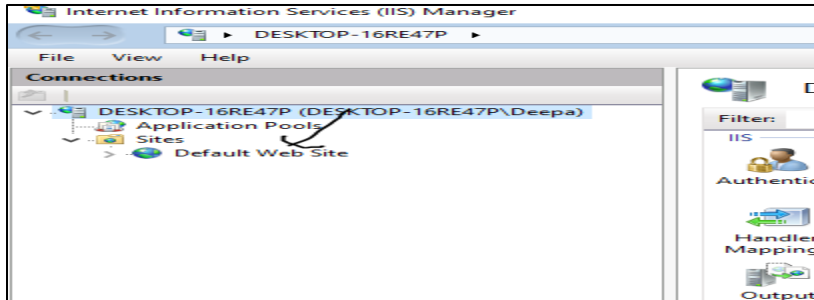
```
using System.Web.Security;
namespace Authentication
{
    public partial class form1 : System.Web.UI.Page
    {
        protected void Button1_Click(object sender, EventArgs e)
        {
            if(FormsAuthentication.Authenticate(TextBox1.Text,TextBox2.Text))
            {
                FormsAuthentication.RedirectFromLoginPage(TextBox1.Text, true);
            }
            else
            {
                Label3.Text = "invalid username or password";
            }
        }
    }
}
```

Windows Authentication:

Add IIS MANAGER

1. Click on windows features on/off
2. Select internet information service
3. Select world wide web services
4. Click on security and select windows authentication

Open iis manager

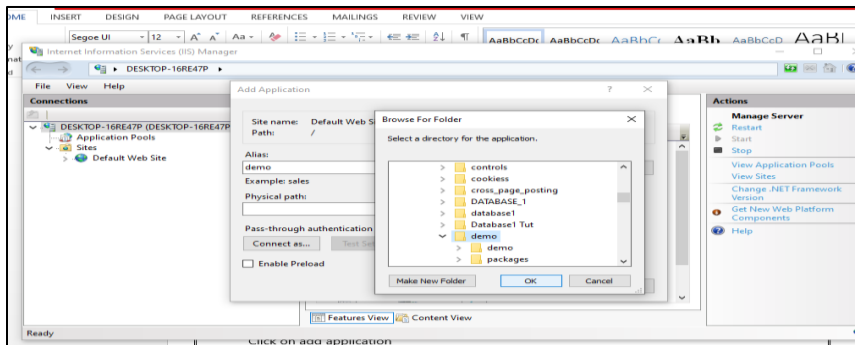


Right click on default web site

Click on add application

->select the application on which you want to apply windows authentication

->set an alias name



Click on add

After adding double click on demo , click on authentication

➔ Disable anonymous

➔ Enable windows

Now on web.config write following code

```
<system.web>
  <authentication mode="Windows"></authentication>
  <authorization>
    <allow users="*" />
    <deny users="?" />
  </authorization>
```

Practical No : 8

- a. Create a web application for inserting and deleting records from a database.
(Using Execute Non Query).
 - b. Create a web application for user defined exception handling.
-

- a. Create a web application for inserting and deleting records from a database.
(Using Execute Non Query).

Web Form and employee table Design:

	eid	ename	desig	addr	salary
1	101	Nitesh	Manager	kurla	54000
2	102	priyanka	developer	thane	74000
3	103	Kriti	designer	mumbai	59000
4	110	Nihanka	CEO	Pune	89000

Solution:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Data.SqlClient;
namespace WebApplication1
{
    public partial class WebForm1 : System.Web.UI.Page
    {
        string cs = "data source=DESKTOP-8QHF2UM;initial catalog=college;integrated
security=true";
        SqlConnection con;
        protected void Page_Load(object sender, EventArgs e)
        {
            con = new SqlConnection(cs);
            if(IsPostBack==false)
            {

                string q = "select eid from employee";
                SqlCommand cmd = new SqlCommand(q, con);
                con.Open();
                SqlDataReader dr = cmd.ExecuteReader();
```

```

        dr.Read();

        DropDownList1.DataSource = dr;
        DropDownList1.DataTextField = "eid";
        DropDownList1.DataBind();
        con.Close();
    }
}
protected void btn_insert_Click(object sender, EventArgs e)
{
    string query = "insert into employee
values('"+eid.Text+"','"+ename.Text+"','"+desig.Text+"','"+address.Text+"','"+salary.Text+
"')";
    SqlCommand cmd = new SqlCommand(query, con);
    con.Open();
    cmd.ExecuteNonQuery();
    Response.Write("Data Inserted");
    con.Close();
}

protected void DropDownList1_SelectedIndexChanged(object sender, EventArgs e)
{
    string q1 = "select ename from employee where eid='" +
DropDownList1.SelectedValue + "'";
    SqlCommand cmd1 = new SqlCommand(q1, con);
    con.Open();
    SqlDataReader dr1 = cmd1.ExecuteReader();
    dr1.Read();
    emp_name.Text = dr1[0].ToString();
    con.Close();
}

protected void btn_del_Click(object sender, EventArgs e)
{
    string q2 = "delete from employee where eid='" + DropDownList1.SelectedValue + "'";
    SqlCommand cmd2 = new SqlCommand(q2, con);
    con.Open();
    cmd2.ExecuteNonQuery();
    Response.Write("record deleted");
    con.Close();
}
}
}

```

Output:

Data Inserted

Employee Id:

Name:

designation:

Address:

Salary:

Select ID:

Employee Name:

	eid	ename	desig	addrs	salary
1	101	Nitesh	Manager	kurla	54000
2	102	priyanka	developer	thane	74000
3	103	Kriti	designer	mumbai	59000
4	110	Niharika	CEO	Pune	89000
5	115	Rohan	Developer	Mumbra	32000

record deleted

Employee Id:

Name:

designation:

Address:

Salary:

Select ID:

Employee Name:

	eid	ename	desig	addrs	salary
1	101	Nitesh	Manager	kurla	54000
2	102	priyanka	developer	thane	74000
3	110	Niharika	CEO	Pune	89000
4	115	Rohan	Developer	Mumbra	32000

b. Create a web application for user defined exception handling.

Solution:

using System;

using System.Web.UI.WebControls;

namespace exception_handling

{

public class invalidAge:Exception

{

public invalidAge(string m):base(m)

{

}

}

public partial class WebForm2 : System.Web.UI.Page

{

protected void Page_Load(object sender, EventArgs e)

{

}

protected void Button1_Click(object sender, EventArgs e)

{

int a;

try

{

a = Convert.ToInt32(TextBox1.Text);

if (a < 18)

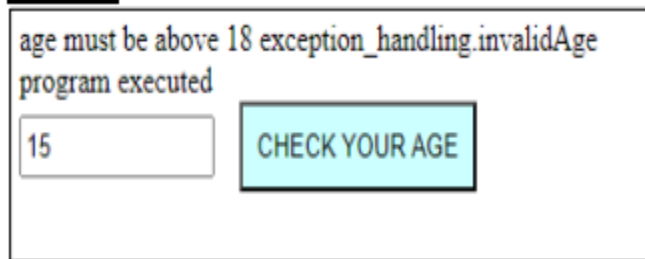

```

throw new invalidAge("age must be above 18");
else
Response.Write("valid age");
}
catch (Exception e1)
{
Response.Write(e1.Message + " " + e1.GetType());

}
finally
{
Response.Write("<br> program executed");
}}}}

```

Output:



The screenshot shows a web page with the following content:

- Text: "age must be above 18 exception_handling.invalidAge"
- Text: "program executed"
- A text input field containing the value "15".
- A blue button labeled "CHECK YOUR AGE".

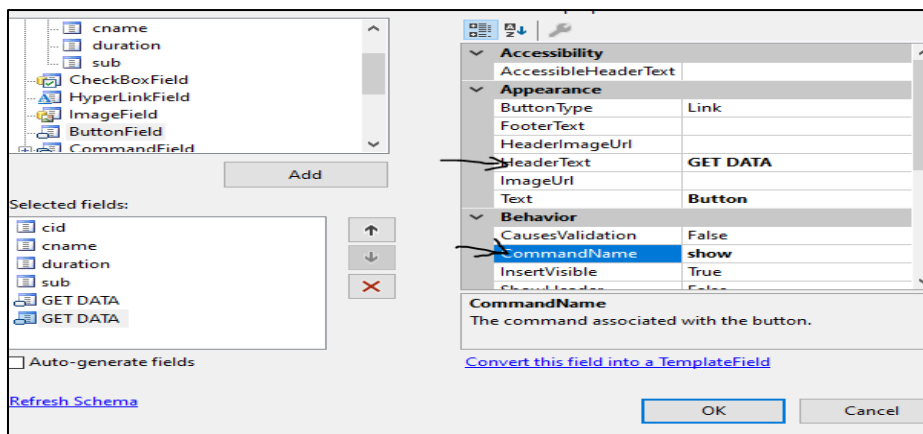
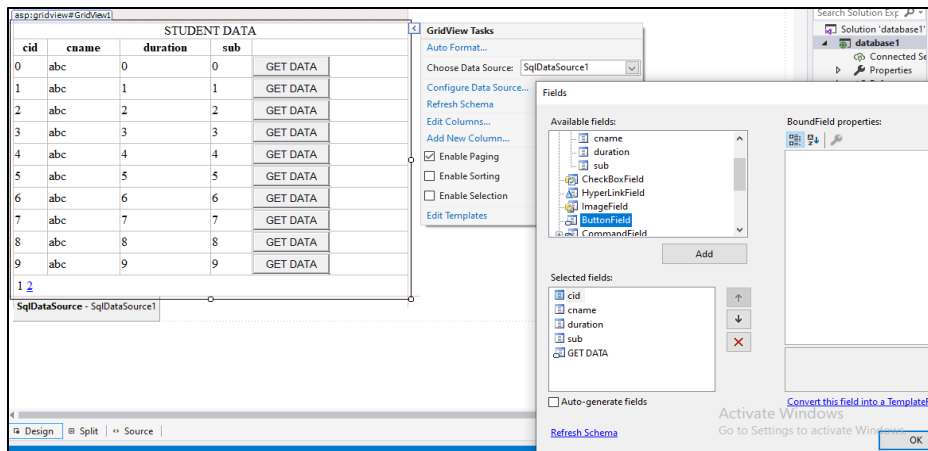
Practical no.9

- Create a web application to demonstrate use of GridView button column and GridView events along with paging and sorting.
- Create a web application to demonstrate data binding using DetailsView and FormView Control.

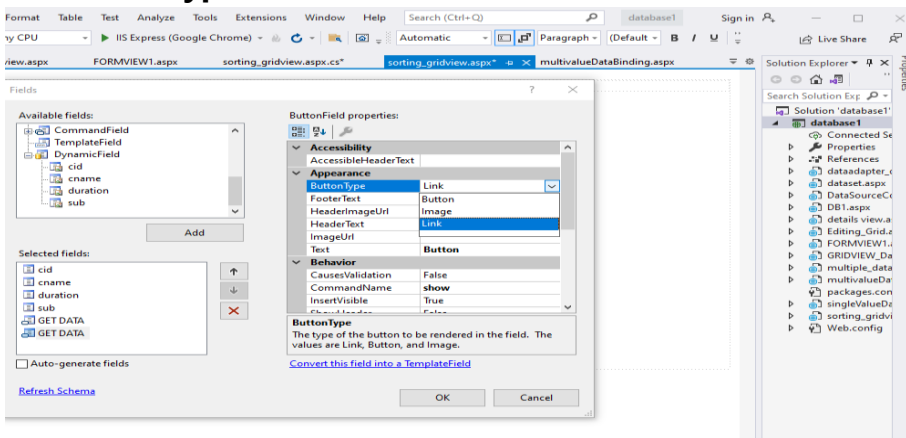
- Create a web application to demonstrate use of GridView button column and GridView events.

Solution:

Click on edit columns->button field->Add->Set header text and Command name->ok



Set ButtonType to Button



On RowCommand write down following code:

protected void GridView1_RowCommand(object sender, GridViewCommandEventArgs e)

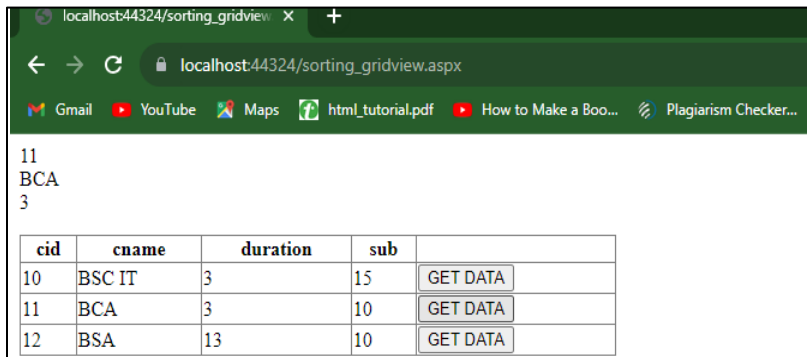
```

{
    if(e.CommandName=="show")
    {
        int i = Convert.ToInt32(e.CommandArgument);

        Response.Write(GridView1.Rows[i].Cells[0].Text);
        Response.Write(" <BR> " + GridView1.Rows[i].Cells[1].Text);
        Response.Write(" <BR> " + GridView1.Rows[i].Cells[2].Text);
    }
}

```

OUTPUT



cid	cname	duration	sub	
10	BSC IT	3	15	GET DATA
11	BCA	3	10	GET DATA
12	BSA	13	10	GET DATA

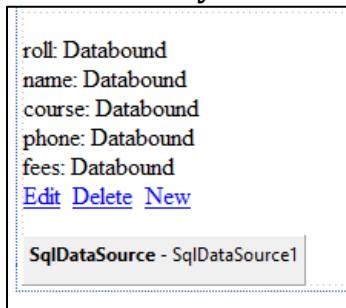
b. Create a web application to demonstrate data binding using DetailsView and FormView Control.

Solution:

(Note: Primary Key is must)

Drag form view control->configure it-> configure Database->new connection->put server name->select database->next->select column from table -> advanced option
->select both the checkboxes ->ok

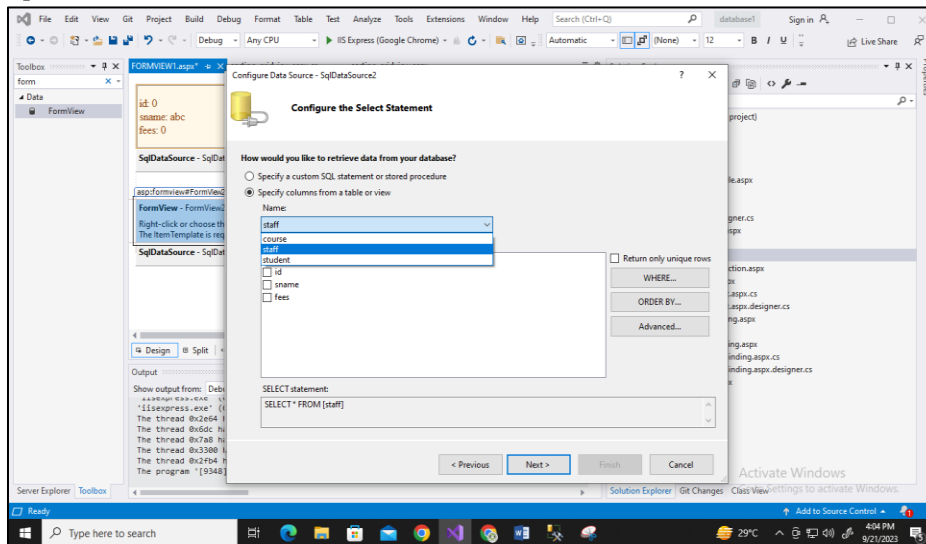
FormView Layout after configuration:



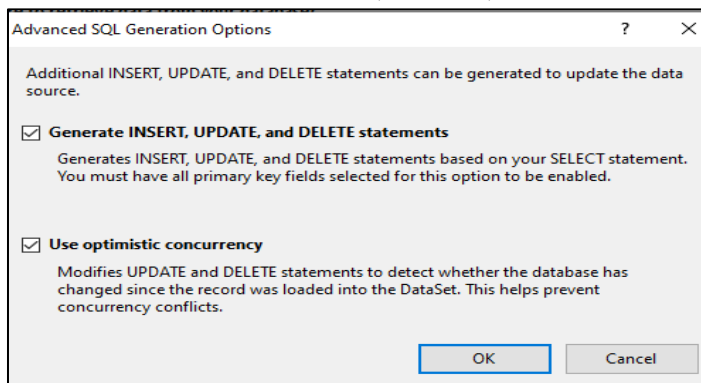
roll: Databound
name: Databound
course: Databound
phone: Databound
fees: Databound
[Edit](#) [Delete](#) [New](#)

SqlDataSource - SqlDataSource1

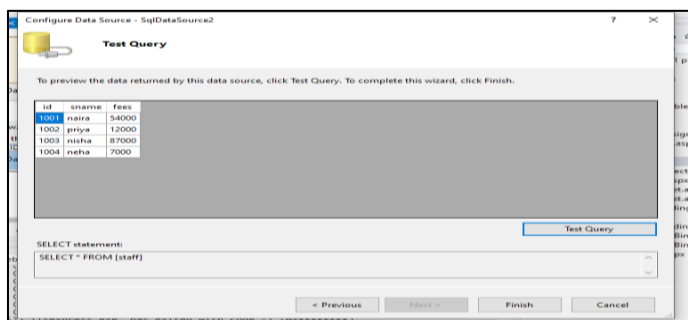
1. Drag formview and datasource control.
2. Configure datasource,select the table having primary key and click on advanced option.



3. Select both the checkboxes ,click ok,then click on next



4. Click test query then finish



5. Set datasource of form view control and run the code.

id: 1001
sname: naira
fees: 54000
[Edit](#) [Delete](#) [New](#)

- On edit button click will get 2 options update and cancel

id: 1001
sname:
fees:
[Update](#) [Cancel](#)

Here we can only change name and fees.

- On new button click will get 2 options insert and cancel

id:
sname:
fees:
[Insert](#) [Cancel](#)

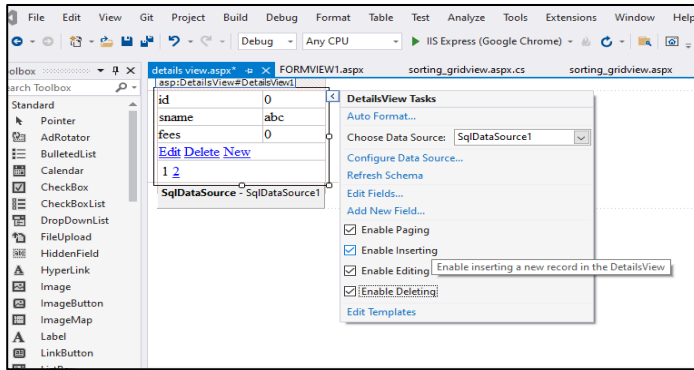
- On delete click.record will be deleted from the table

On formView Control->enable paging->click on edit template->choose header template/footer template etc. from display option

Details View:

Solution:

- ✚ Drag a details view and sqldatasource control.
- ✚ There must a primary key in details to do insertion, updation and deletion.
- ✚ Configure the datasource same as we configured for formview control.



Select all 4 options

Output:

id	1003
sname	nisha
fees	87000
Edit Delete New	
1 2 3 4	

Practical no.10

- a. Create a web application to demonstrate JS Bootstrap Button.
- b. Create a web application to demonstrate use of various Ajax controls.

a. Create a web application to demonstrate JS Bootstrap Button.

Solution:

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="WebForm4.aspx.cs"
Inherits="WebApplication34.WebForm4" %>
```

```
<!DOCTYPE html>
```

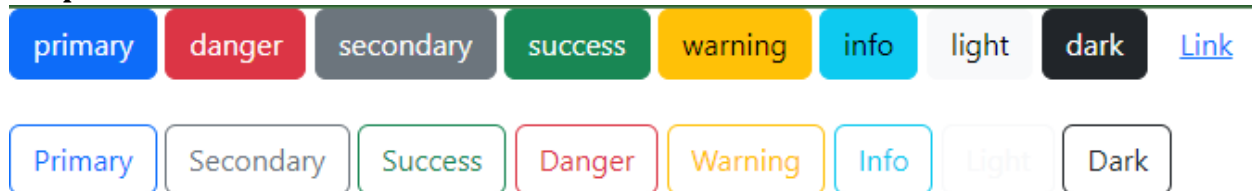
```
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <link href="Content/bootstrap.min.css" rel="stylesheet" />
    <script src="Scripts/bootstrap.bundle.js"></script>
    <script src="Scripts/bootstrap.min.js"></script>
    <title></title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            <asp:Button ID="Button1" runat="server" Text="primary" class="btn btn-primary"/>
            <asp:Button ID="Button2" runat="server" Text="danger" class="btn btn-danger"/>
            <asp:Button ID="Button3" runat="server" Text="secondary" class="btn btn-
secondary"/>
            <asp:Button ID="Button4" runat="server" Text="success" class="btn btn-success"/>
            <asp:Button ID="Button5" runat="server" Text="warning" class="btn btn-
warning"/>
            <asp:Button ID="Button6" runat="server" Text="info" class="btn btn-info"/>
            <asp:Button ID="Button7" runat="server" Text="light" class="btn btn-light"/>
            <asp:Button ID="Button8" runat="server" Text="dark" class="btn btn-dark"/>
            <button type="button" class="btn btn-link">Link</button>
            <br /><br />
            <button type="button" class="btn btn-outline-primary">Primary</button>
            <button type="button" class="btn btn-outline-secondary">Secondary</button>
            <button type="button" class="btn btn-outline-success">Success</button>
            <button type="button" class="btn btn-outline-danger">Danger</button>
            <button type="button" class="btn btn-outline-warning">Warning</button>
            <button type="button" class="btn btn-outline-info">Info</button>
            <button type="button" class="btn btn-outline-light">Light</button>
```

```

<button type="button" class="btn btn-outline-dark">Dark</button>
  </div>
</form>
</body>
</html>

```

Output:



b. Create a web application to demonstrate use of various Ajax controls.

Solution:

1. Timer Control

Step 1: drag a script manager control

Step2: drag update panel control

Step3: drag a label and timer control inside update panel.

```

<asp:UpdatePanel ID="UpdatePanel1" runat="server" UpdateMode="Always">

  <ContentTemplate>
    <asp:Timer ID="Timer1" runat="server" Interval="1000">
      </asp:Timer>

    <asp:Label ID="Label1" runat="server" Height="101px" style="width:304px" >
      </asp:Label>
    </ContentTemplate>

  </asp:UpdatePanel>

```

On page load write down following code:

```
Label1.Text = DateTime.Now.ToString("hh:mm:ss")
```

Output:



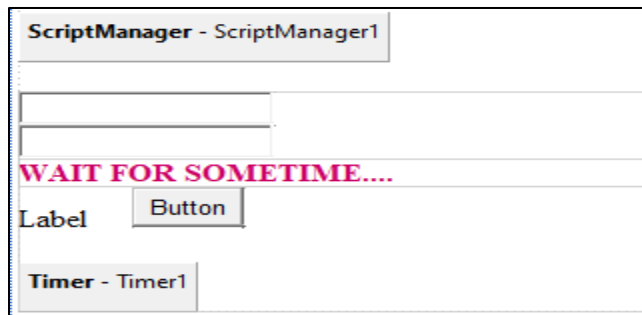
UpdateProgress Control:


```

<body>
  <form id="form1" runat="server">
    <div>
      <asp:Label ID="Label1" runat="server" Text="Label"></asp:Label>
    </div>
    <p>
      <asp:Label ID="Label3" runat="server" Text="Label"></asp:Label>
      <asp:ScriptManager ID="ScriptManager1" runat="server">
        </asp:ScriptManager>
      </p>
      <asp:UpdatePanel ID="UpdatePanel1" runat="server">

        <ContentTemplate>
          <asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>
          <br />
          <asp:TextBox ID="TextBox2" runat="server"></asp:TextBox>
          <br />
          <asp:UpdateProgress ID="UpdateProgress1" runat="server">
            <ProgressTemplate>
              <asp:Label ID="Label4" runat="server" Text="WAIT FOR SOMETIME...." Font-
Bold="True" ForeColor="#CC0066"></asp:Label>
            </ProgressTemplate>
          </asp:UpdateProgress>
          <asp:Label ID="Label2" runat="server" Text="Label"></asp:Label>
          <asp:Button ID="Button1" runat="server" CssClass="auto-style1"
OnClick="Button1_Click1" Text="Button" />
          <br />
          <br />
          <asp:Timer ID="Timer1" runat="server" Interval="1000"
OnTick="Timer1_Tick"></asp:Timer>
        </ContentTemplate>
      </asp:UpdatePanel>
    </form>
  </body>
</html>

```



```
protected void Button1_Click1(object sender, EventArgs e)
{
    System.Threading.Thread.Sleep(2000);
    Label2.Text = TextBox2.Text + TextBox1.Text;
}
```