CMR UNIVERSITY

Practical Record

.NET Framwork and Applications Lab

8CSPL5261

Master of Computer Applications

[3rd Semester]

Name: Mahadev

Reg.no: 21DMMCA028

2022-2023

School of Science Studies

#5, Bhuvanagiri, OMBR Layout, Bengaluru-560 043, Karnataka-India

CMR UNIVERSITY

Certificate

This is to certify that ***Mahadev*** *belonging to* ***Ist Semester MCA*** *has successfully completed the project titled* ***laboratory exercises*** *in fulfilment of practical prescribed by the* ***School of Science Studies*** *for the subject Mobile Application Development lab during the academic year* ***2022-2023****.*

Faculty In-charge **ProgramCoordinator**

Name: *Mahadev*

Reg.no: *21DMMCA028*

Date of Practical Exam: _____

Examiners:

1.

2.

School of Science Studies

#5, Bhuvanagiri, OMBR Layout, Bengaluru- 560 043, Karnataka- India

Part-A

Part–A

1. Demonstration of Constructors : Define a class Student, which contains the following information about students: Full Name, Semester, Course_Code, Subject, e-mail and phone number. Declare several constructors for the class Student, which have different lists of parameters (for complete information about a student or part of it). Data, which has no initial value to be initialized with null. Add a method in the class Student, which displays complete information about the student.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace program1

{
   public class student
   {
      public string name;
      int sem, regno;
      string course_code;
      string email;
      string subject;
      long phoneno;
      public static int count = 0;

      public student()
      {
Console.WriteLine("default constructor is invoked");
Console.WriteLine();
         count++;
         name = "";
         regno = 0;
sem = 0;
course_code = "";
```

```
        email = "";
        subject = "";
phoneno = 0;
    }

    public student(string name, int regno, int sem, string course_code,string subject, string email, long phoneno)
    {
        count++;
Console.WriteLine("parameterised constructor is invoked");
Console.WriteLine();
        this.name = name;
this.regno = regno;
this.sem = sem;
this.course_code = course_code;
this.email = email;
this.subject = subject;
this.phoneno = phoneno;
    }
    public student(student s)
    {
        count++;
Console.WriteLine(" copy constructor is invoked");
Console.WriteLine();
        this.name = s.name;
this.regno = s.regno;
this.sem = s.sem;
this.course_code = s.course_code;
this.email = s.email;
this.subject = s.subject;
this.phoneno = s.phoneno;
```

```
    }
    public student(string name, long phoneno)
    {
       count++;
       this.name = name;
this.phoneno = phoneno;
Console.WriteLine("\nenter student details");
Console.WriteLine("enter the regno of " + name);
this.regno = Int32.Parse(Console.ReadLine());
Console.WriteLine("enter the semester");
this.sem = Int32.Parse(Console.ReadLine());
Console.WriteLine("enter the coursecode");
this.course_code=Console.ReadLine();
Console.WriteLine("enter the email");
this.email = Console.ReadLine();
Console.WriteLine("enter the coursename");
this.subject = Console.ReadLine();

    }

    public void printDetails()
    {
Console.WriteLine("student "+ count +" details");
Console.WriteLine("name :" + this.name);
Console.WriteLine("regno :" + this.regno);
Console.WriteLine("semester :" + this.sem);
Console.WriteLine("coursecode :" + this.course_code);
Console.WriteLine("subject :" + this.subject);
Console.WriteLine("email:" + this.email);
Console.WriteLine("phoneno:" + this.phoneno);
```

```
    }
  }
  class Program
  {
    static void Main(string[] args)
    {
      student s1 = new student();
      student s2 = new student("Arjun",23,3,"8CSPL5231","C#","Arjun@gmail.com",5749264097);
      s2.printDetails();
      student s3 = new student("Krishna", 978598054);
      s3.printDetails();
      student s4 = new student(s3);
      s4.printDetails();
Console.WriteLine("Total no of student is created" +student.count);
Console.ReadLine();
    }
  }
}
```

OUTPUT:

default constructor is invoked parameterised constructor is invoked student 2 details name :Arjun regno :23 semester :3 coursecode :8CSPL5231 subject :C# email:Arjun@gmail.com phoneno:5749264097	enter student details enter the regno of Krishna 28 enter the semester 4 enter the coursecode 8CSPL5231 enter the email krishna@gmail.com enter the coursename C#	student 3 details name :Krishna regno :28 semester :4 coursecode :8CSPL5231 subject :C# email:krishna@gmail.com phoneno:978598054 copy constructor is invoked	student 4 details name :Krishna regno :28 semester :4 coursecode :8CSPL5231 subject :C# email:krishna@gmail.com phoneno:978598054 Total no of student is created4

2. **Demonstration of Delegates :** Create a delegate called strMyDel that takes one string parameter .Create a class named TestDelegate that contains two non-static methods changecase() and reverse() having following signature:

void changecase(String str); Changes the case of input characters ,

void Reverse(String str); - reverse the given string
Use multicast delegate

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace ConsoleApplication1
{
   class Program
   {
      public delegate void strMyDel(string str);
      public void changecase(string str)
      {
         int l = str.Length;
char[] arr1 = str.ToCharArray();

Console.Write("\nAfter conversion the string is :");
         foreach (char ch in arr1)
         {
            if (char.IsLower(ch))
Console.Write(char.ToUpper(ch));
            else
Console.Write(char.ToLower(ch));
         }
      }

      public void reverse(string myStr)
      {
         string rev = "";
Console.WriteLine("\nstringis :{0} \n", myStr);
```

```
        int len = myStr.Length - 1;
        while (len>= 0)
        {
           rev = rev + myStr[len];
len--;
        }
Console.WriteLine("reversed string is :{0}", rev);

    }

    static void Main(string[] args)

    {
        Program strobj = new Program();
strMyDeldeleobj = new strMyDel(strobj.changecase);
deleobj +=strobj.reverse;
Console.WriteLine("string conversion Program");
Console.WriteLine("Enter a sentence");
        string sentence =Console.ReadLine();
deleobj.Invoke(sentence);
Console.ReadLine();
    }
  }
}
```

OUTPUT:

ChangeCase: nAMASTEwORLD

Reverse: dlroWetsamaN

3. Demonstration of Exceptions: Write a method ReadNumber(int start, int end) that reads an integer from the console in the range [start…end]. In case the input integer is not valid or it is not in the required range throw appropriate exceptions. Using this method, write a program that takes 10 integers a1, a2,. a10 such that 1 < a1 < … < a10 <; 100.

```
using System;

public class Program

{

   public static int ReadNumber(int start, int end)

   {

      int number;

      while (true)

      {

Console.Write($"Enter an integer between {start} and {end}: ");

         string input = Console.ReadLine();

         try

         {

            number = int.Parse(input);

            if (number < start || number > end)

            {

               throw new ArgumentOutOfRangeException();

            }

            break;

         }

         catch (FormatException)

         {

Console.WriteLine("Invalid input. Please enter a valid integer.");

         }

         catch (ArgumentOutOfRangeException)

         {

Console.WriteLine($"The entered number is not in the required range [{start}...{end}].");

         }

      }


      return number;

   }
```

```
   public static void Main(string[] args)
   {
int[] numbers = new int[10];
      int start = 1;
      int end = 100;

      try
      {
         for (int i = 0; i<numbers.Length; i++)
         {
            numbers[i] = ReadNumber(start + i, end);
         }

Console.WriteLine("Entered numbers:");
         foreach (int number in numbers)
         {
Console.WriteLine(number);
         }
      }
      catch (Exception ex)
      {
Console.WriteLine("An error occurred: " + ex.Message);
      }
   }}
}
```

OUTPUT:

Enter an integer between 1 and 100: 2

Enter an integer between 2 and 100: 4

Enter an integer between 3 and 100: 3

Enter an integer between 4 and 100: 66

Enter an integer between 5 and 100: 34

Enter an integer between 6 and 100: 98

Enter an integer between 7 and 100: 98

Enter an integer between 8 and 100: 78

Enter an integer between 9 and 100: 99

Enter an integer between 10 and 100: 100

Entered numbers:

2

4

3

66

34

98

98

78

99

100

4. ***Design a Windows Forms application containing the following controls:***

- **A ListBox**
- **A Button**
- **An Image**
- **A Label**

The listbox is used to list items available in a store. When the user clicks on an item in the listbox, its image is displayed in the image control. When the user clicks the button, the cost of the selected item is displayed in the control.

Property Table

Sno	Control	Property	Value
1	Label1	Name	label1
		Text	Choose an item from the below list
		Font	
2	ListBox1	Name	Store
		Items	
		Font	
3	Label2	Name	Cost
		Text	
4	PictureBox1	Name	PictureBox1
		SizeMode	CenterImage

```
using System.Windows.Forms;
using static System.Windows.Forms.AxHost;
namespace Windows FormsApp4
{
3 references
```

```
public partial class Form1 : Form
{
1 reference
public Form1()
{
}
InitializeComponent();
1 reference
private void Form1_Load(object sender, EventArgs e)
{
}

Private void button_Click(object sender, EventArgs e)

{

If(Store.SelectedIndex> -1)

{

Int I = Store.SelectedIndex;

Cost.Text = “You have chosen”  +Store.GetItemIndex(Store.selectedItem) + “and its cost is Rs.”;

If(i == 0)

        Cost.Text += “90”;

else if(i == 1)

        Cost.Text+=”300”;

else if(i == 2)

        Cost.Text+=”70”;

else if(i == 3)

        Cost.Text+=”100”;

else if(i == 4)

        Cost.Text+=”55”;

Else if(i == 5)

        Cost.Text+=”80”;

}

}

        Private void Store_SelectedIndexChanged(object sender, EventArgs e)

        {

                Picturebox1.Refresh();

                Int i = Store .SelectedIndex;
```

```
if(i == 0)
{
        pictureBox1.Image = Image.FormFile("D:\\img\\grapes.jpg");
}
else if(i == 1)
{
        pictureBox1.Image = Image.FormFile("D:\\img\\apple.jpg");
}
else if(i == 2)
{
        pictureBox1.Image = Image.FormFile("D:\\img\\chikoo.jpg");
}
else if(i == 3)
{
        pictureBox1.Image = Image.FormFile("D:\\img\\strawberry.jpg");
}
else if(i == 4)
{
        pictureBox1.Image = Image.FormFile("D:\\img\\banana.jpg");
}
else if(i == 5)
{
        pictureBox1.Image = Image.FormFile("D:\\img\\orange.jpg");
}
}
}}
```

5.Write a Program in ASP that has a form taking the user's name as input. Store this name in a permanent cookie & whenever the page is opened again, then the value of the name field should be attached with the cookie's content.

WebForm1.aspx.cx

```
using System;
```

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

namespace labprogram5
{
   public partial class WebForm1 :System.Web.UI.Page
   {
      protected void Page_Load(object sender, EventArgs e)
      {
HttpCookie c = Request.Cookies["mycookie"];
         if (c != null)
         {
            Label1.Text = c["username"];
         }
      }

      protected void Button1_Click(object sender, EventArgs e)
      {
         if (TextBox1.Text == "" || TextBox2.Text == "")
         {
            Label2.Visible = true;
            Label2.Text = "enter valid details";
         }
         else
         {

HttpCookie cookie = new HttpCookie("mycookie");
            cookie["username"] = TextBox1.Text;
```

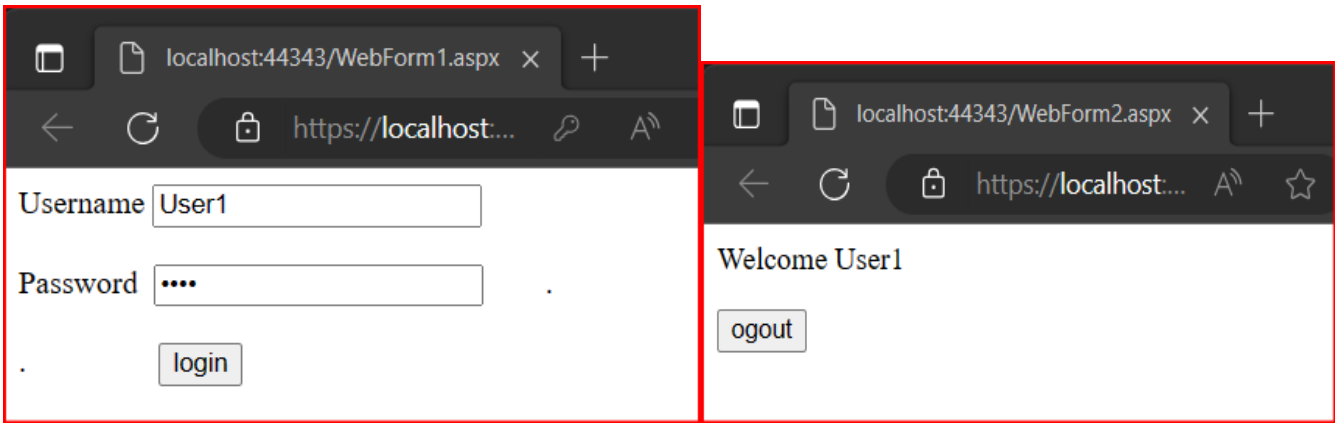
```
            cookie["password"] = TextBox2.Text;
Response.Cookies.Add(cookie);
cookie.Expires = DateTime.Now.AddMinutes(2);
Response.Redirect("logout.aspx");
        }
      }
   }
}
```

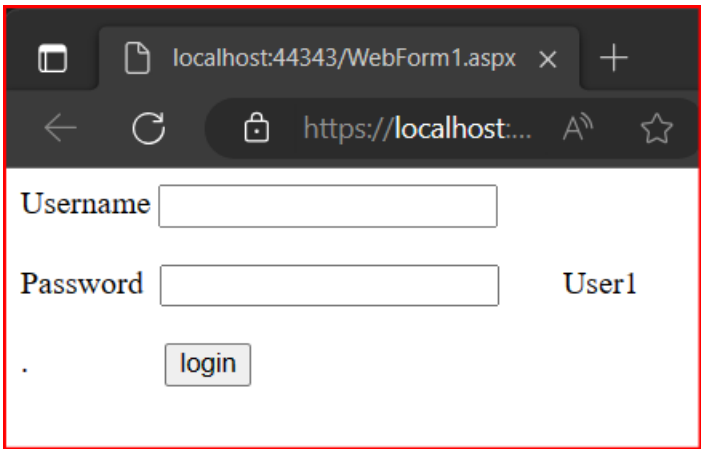
WebForm2.aspx.cs

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

namespace labprogram5
{
   public partial class logout :System.Web.UI.Page
   {
      protected void Page_Load(object sender, EventArgs e)
      {
HttpCookie c = Request.Cookies["mycookie"];
         if (c != null)
         {
            Label1.Text = c["username"];

         }
      }
```

```
    protected void Button1_Click(object sender, EventArgs e)
    {
Response.Redirect("cookie.aspx");
    }
  }
}
```

OUTPUT:





6. Create an ASP.NET web application to insert 3 records inside the SQL database table 'Department' having following fields(DeptId, DeptName, EmpName, Salary). Update the salary for any one employee and increment it to 15% of the present salary. Perform delete operation on 1 row of the database table.

```
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 labprgm6
{
   public partial class dept :System.Web.UI.Page
   {
SqlConnection con;
SqlCommandcmd;
SqlDataReaderdr;
      protected void Page_Load(object sender, EventArgs e)
      {
         con=new SqlConnection("Data Source=.\\sqlexpress;InitialCatalog=cmrdata;Integrated Security=True");
con.Open();
         if (!IsPostBack)
         {

cmd = new SqlCommand("select * from dept", con);
dr = cmd.ExecuteReader();
            while (dr.Read())
DropDownList1.Items.Add(dr.GetValue(0).ToString());
dr.Close();
         }
      }

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

```
cmd=new SqlCommand("delete from dept where empid="+DropDownList1.Text,con);
cmd.ExecuteNonQuery();
Response.Write("record deleted");
        TextBox2.Text="";
        TextBox3.Text="";
        TextBox4.Text="";
        TextBox5.Text="";
DropDownList1.Items.Remove(DropDownList1.SelectedItem);

    }

    protected void DropDownList1_SelectedIndexChanged(object sender, EventArgs e)
    {
try{
cmd=new SqlCommand("select * from dept where empid="+DropDownList1.Text,con);
dr=cmd.ExecuteReader();
            if(dr.Read())
            {
            TextBox2.Text=dr.GetString(1).ToString();
               TextBox3.Text=dr.GetString(2).ToString();
               TextBox4.Text=dr.GetString(3).ToString();

               TextBox5.Text=dr.GetString(4).ToString();
            }
            else
Response.Write("record not found");
        }
catch(Exception ex)
        {
Response.Write(ex.Message);
        }
```

```
dr.Close();
    }

    protected void Button2_Click(object sender, EventArgs e)
    {
cmd=new SqlCommand("update  dept set salary =(salary*0.15+salary) where empid="+DropDownList1.Text,con);
cmd.ExecuteNonQuery();
Response.Write("record updated");
        TextBox2.Text="";
        TextBox3.Text="";
        TextBox4.Text="";
        TextBox5.Text="";

    }

  }
}
```

OUTPUT:

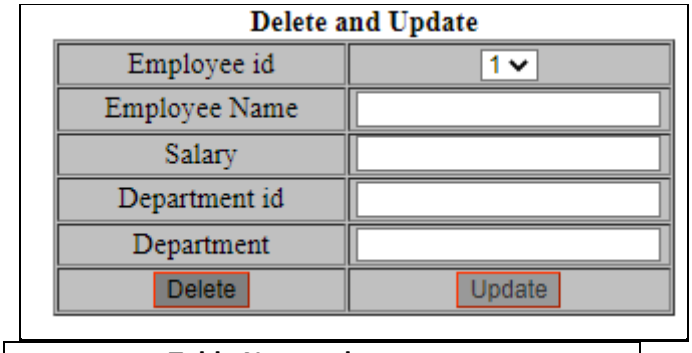


Table Name : department			
SNo	column name	Datatype	Constraints
1	empid	numeric(10,0)	primary key
2	ename	varchar(15)	
3	salary	numeric(10,0)	
4	deptid	varchar(15)	
5	deptname	varchar(15)	

Property Table

SNo	Control Name	Property	Value
1	DropDownList1	id	lstempid
2	TextBox2	id	txtname
3	TextBox3	id	txtsal
4	TextBox4	id	txtdeptid
5	TextBox5	id	txtdeptname
6	Button1	Text	Delete
		id	btndelete
7	Button2	Text	Update Salary
		id	btnupdate

7. Create an ASP.NET web application to display all the Empname and Deptid of the employee from the 'Department' table and bind it to GridView . Database fields are(DeptId, DeptName, EmpName, Salary).

```
using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;
```

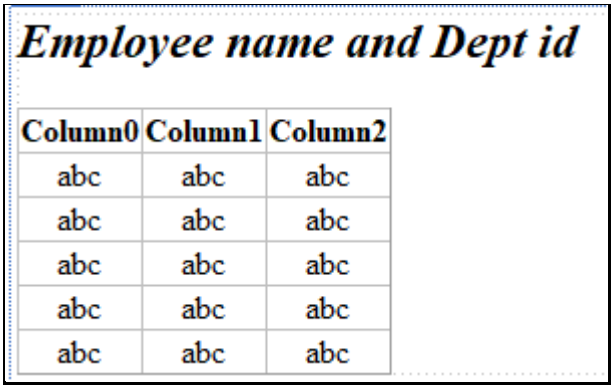
```
using System.Web.UI.WebControls;
using System.Data.SqlClient;
using System.Configuration;
using System.Data;

namespace WebApplication4
{
   public partial class WebForm1 :System.Web.UI.Page
   {
SqlConnection con;

SqlDataAdapter da;
DataSet ds;
      protected void Page_Load(object sender, EventArgs e)
      {

      }

      protected void Button1_Click(object sender, EventArgs e)
      {
         if (!this.IsPostBack)
         {
            con = new SqlConnection("Data Source=.\\sqlexpress;InitialCatalog=cmrdata;Integrated Security=True");
            da = new SqlDataAdapter("select * from dept", con);
            ds = new DataSet();
da.Fill(ds, "dept");
            GridView1.DataSource = ds;
            GridView1.DataBind();
         }
      }
      }
```

}

OUTPUT:

Table Name : department			
SNo	column name	Datatype	Constraints
1	empid	numeric(10,0)	primary key
2	ename	varchar(15)	
3	salary	numeric(10,0)	
4	deptid	varchar(15)	
5	deptname	varchar(15)	

Property Table			
SNo	Control Name	Property	Value
1	Label	id	Label1
2	GridView1	id	GridView1

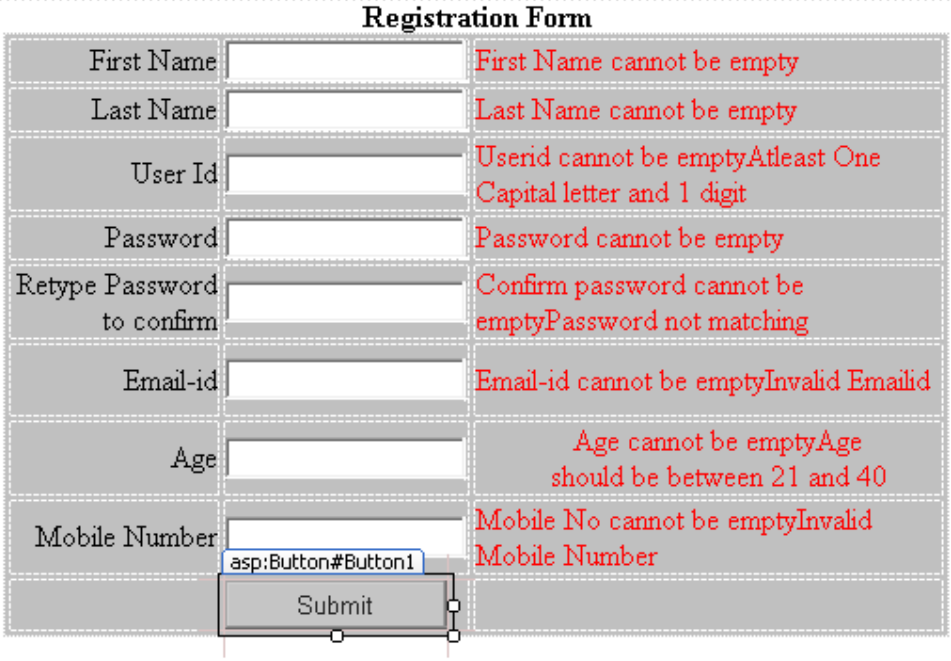


8. Design an ASP.NET registration web page that demonstrates all Validation controls.

Property Table			
S.No	**Control Name**	**Property**	**textfn**
1	TextBox1	id	txtfn
2	TextBox2	id	txtln
3	TextBox3	id	txtuser

4	TextBox4	id	txtpwd
		TextMode	Password
5	TextBox5	id	txtrepwd
		TextMode	Password
6	TextBox6	id	txtemail
7	TextBox7	id	txtage
8	TextBox8	id	txtmobile
9	Button1	Text	Submit
10	RequiredFieldValidator1	ControlToValidate	txtfn
		ForeColor	Red
		ErrorMessage	First Name cannot be empty
11	RequiredFieldValidator2	ControlToValidate	txtln
		ForeColor	Red
		ErrorMessage	Last Name cannot be empty
12	RequiredFieldValidator3	ControlToValidate	txtuser
		ForeColor	Red
		ErrorMessage	Userid cannot be empty
13	RequiredFieldValidator4	ControlToValidate	txtpwd
		ForeColor	Red
		ErrorMessage	Password cannot be empty
14	RequiredFieldValidator5	ControlToValidate	txtrepwd
		ForeColor	Red
		ErrorMessage	Confirm password cannot be empty
15	RequiredFieldValidator6	ControlToValidate	txtemail
		ForeColor	Red
		ErrorMessage	emailid cannot be empty
16	RequiredFieldValidator7	ControlToValidate	txtage
		ForeColor	Red
		ErrorMessage	Age cannot be empty
17	RequiredFieldValidator8	ControlToValidate	txtmobile
		ForeColor	Red
		ErrorMessage	Mobileno cannot be empty

18	CustomValidator1	ControlToValidate	txtuser
		ForeColor	Red
		ErrorMessage	Atleast One Capital letter and 1 digit
19	RangeValidator1	ControlToValidate	txtage
		ForeColor	Red
		ErrorMessage	Age should be between 21 and 40
		Maximum Value	40
		Minimum Value	21
20	CompareValidator1	ControlToValidate	txtrepwd
		ForeColor	Red
		ErrorMessage	Password not matching
		ControlToCompare	txtpwd
21	RegularExpressionValidator1	ControlToValidate	txtemail
		ForeColor	Red
		ErrorMessage	Invalid Emailid
		ValidationExpression	\w+([-+.']\w+)*@\w+([-.]\w+)*\.\w+([-.]\w+)*
22	RegularExpressionValidator2	ControlToValidate	txtmobile
		ForeColor	Red
		ErrorMessage	Invalid Mobile Number
		ValidationExpression	\d{10}

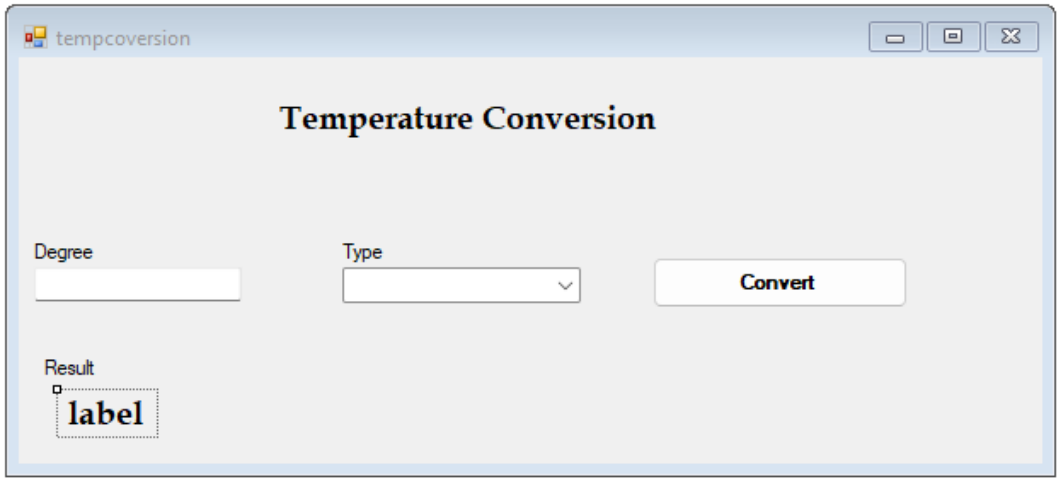



```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

namespace labprg8
{
   public partial class validation :System.Web.UI.Page
   {
      protected void Page_Load(object sender, EventArgs e)
      {

      }

      protected void TextBox8_TextChanged(object sender, EventArgs e)
      {
```

```
    }

    protected void CustomValidator1_ServerValidate(object source, ServerValidateEventArgsargs)
    {
      {
        string str = args.Value;
args.IsValid = false;
        if (str.Length< 7 || str.Length> 20)
{ return; }
        bool capital = false;


       foreach (char ch in str)
        {
          if (ch>= 'A' &&ch<= 'Z')
          {
            capital = true;
            break;
          }
        }
        if (!capital)
          return;
        bool digit = false;
        foreach (char ch in str)
        {
          if (ch>= '0' &&ch<= '9')
{ digit = true; break; }
        }
        if (!digit)
          return;
```

```
args.IsValid = true;
        }

    }

    protected void Button1_Click(object sender, EventArgs e)
    {
Response.Write("user registration for" + TextBox1.Text + " submitted succesfully");
    }
  }
}
```

9. Create a simple Web Service that converts the temperature from Fahrenheit to Celsius, and vice versa. Also write an ASP program to consume this web service.



Form1.cs

```
using System;
 using System.Collections.Generic;
using System.ComponentMcdel;
 using System. Data;
 using System. Drawing;
 using System. Ling;
 using System. Text;
 using Systemmindcws.Forms;
 using windcmsFormsApplicationl.ServiceReferencel;
namespace windcwsFormsApplicationl
public partial class Forml : Form{
        webServicelSoapClientobj;
        public Forml(){
                InitializeComponent();
                }
        private void buttonl_Click(object sender, EventArgs e) {
                double res=0.0;
                 if (comboBoxl.Text == "Farenheit") res = obj.tofaren(Convert.ToDouble(textBoxl.Text));
```

```
        else if (comboBoxl.Text == "Celsius") res = obj.tocelsius(Convert.ToDouble(textBoxl.Text));
        resultlabel.Visible = true;
        resultlabel.Text = res.ToString(); //textBoxl.Text = "";
        textBoxl.Focus();
}
private void Forml_Load(object sender, EventArgs e){
        obj = new webServicelSoapClient();
        }
        }
}
```

```
using System;
using System.Collecticas.Generic;
 using System.Linq; using System.Web;
 using System.Web.Services;
-namespace Conversion
- /// <summary>
/// Summary description for WebServicel ///
</summary>
[webService(Namespace = "htto://temouri.org/")]
[webService8inding(ConformsTo = wsiProfiles.BasicProfile1_1)]
[System.Componenthodel.ToolboxItem(false)]
 // To allow this Web Service to be called from script, using ASP.NET A)AX, uncomment the following line.
 // [System.Web.Script.Services.ScriptService]
public class WebServicel :System.Web.Services.webService{
        [wletNethod] public double tofaren(double c) {
                return ( c*1.8 + 32);
                }
        [WebMethod] public double tocelsius(double f){
                return ((f- 32)/1.8);
```

```
        }
    }
}
```

10. Demonstrate how part of the web page can be updated using AJAX UpdatePanel control

Steps:

Create a page with two independently updating regions.

a) Create a web form

b) From the AJAX Extensions tab of the toolbox, drag and drop the ScriptManagercontrol[ie. add it to the page.]

c) Then add UpdatePanelcontrol two times in the form. [ie, add two UpdatePanel controls to the page.]

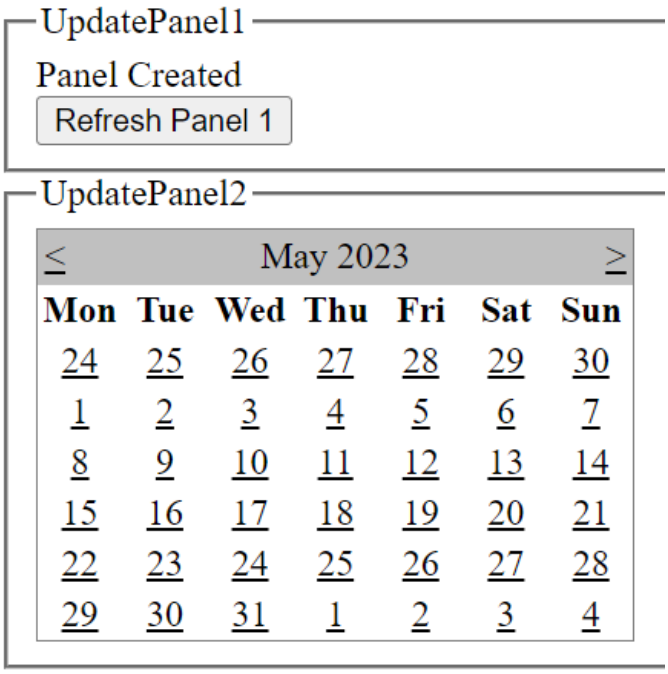
d)In one of the UpdatePanel controls, add a Label control and set its Text property to Panel Created. In the same UpdatePanel control, add a Button control and set its Text property to Refresh Panel1.
In the other UpdatePanel control, add a Calendar control.
And add a Calendar control outside the updatePanel.

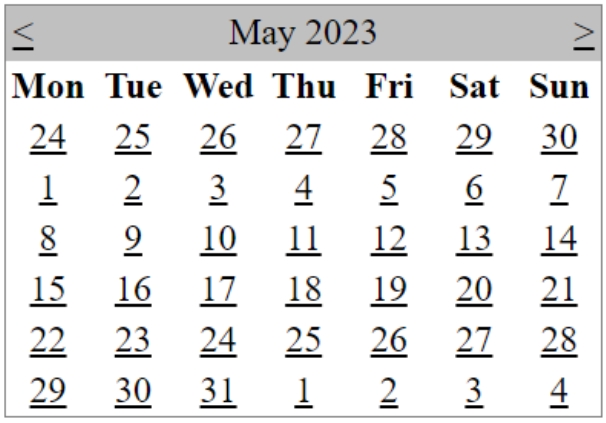
e)Double-click the button to add an event handler for its Click event.
Add code to the handler, which sets the Label control to the current time.

f)Save your changes and press CTRL+F5 to view the page in a browser.

g)Click the button.
The text in the panel changes to display the last time that the panel's content was refreshed.

h)In the calendar, move to a different month.





Program10.aspx

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="prgm10.aspx.cs" Inherits="AjaxPractise.prgm10" %>
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
   <title></title>
</head>
<body>
   <form id="form1" runat="server">
      <div>
        <asp:ScriptManager id="ScriptManager1" runat="server">
      </asp:ScriptManager>
```

```
    <asp:UpdatePanelid="UpdatePanel1" runat="server">
      <ContentTemplate>
        <fieldset>
        <legend>UpdatePanel1</legend>
        <asp:Label ID="Label1" runat="server" Text="Panel Created"></asp:Label><br />
        <asp:Button ID="Button1" runat="server" Text="Refresh Panel 1" OnClick="Button1_Click" />
        </fieldset>
      </ContentTemplate>
    </asp:UpdatePanel>
    <asp:UpdatePanel ID="UpdatePanel2" runat="server">
      <ContentTemplate>
        <fieldset>
        <legend>UpdatePanel2</legend>
        <asp:Calendar ID="Calendar1" runat="server"></asp:Calendar>
        </fieldset>
       </ContentTemplate>
    </asp:UpdatePanel>
      <br />
      <br />
      <br />
    <asp:Calendar ID="Calendar2" runat="server"></asp:Calendar>

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

Program10.aspx.cs

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

namespace WebApplication6
{
  public partial class WebForm1 :System.Web.UI.Page
  {
    protected void Page_Load(object sender, EventArgs e)
    {

    }

    protected void Button1_Click(object sender, EventArgs e)
    {
      Label1.Text = "panel refreshed at" + DateTime.Now.ToString();
    }
  }}
```