GURU NANAK COLLEGE OF ARTS, SCIENCE & COMMERCE
G.T.B NAGAR, MUMBAI-400037

DEPARTMENT OF INFORMATION TECHNOLOGY

TY BSc.IT SEMESTER V

Practical Journal In

ADVANCED WEB PROGRAMMING

Submitted by

NAME- Neeraj Rastogi

ROLL NO- 99

Academic Year 2023-24

GURU NANAK COLLEGE OF ARTS, SCIENCE & COMMERCE
G.T.B NAGAR, MUMBAI-400037

DEPARTMENT OF INFORMATION TECHNOLOGY

CERTIFICATE

This is to certify that Mr.Neeraj Rastogi of **B.Sc.(I.T.) Semester-V** Roll No. 99 has successfully completed the practicals in the subject of ADVANCED WEB PROGRAMMING as per the requirement of the University Of Mumbai in part fulfillment for the completion of Degree of Bachelor of Science (INFORMATION TECHNOLOGY). It is also to certify that this is the original work of the candidate done during the academic year 2022-2023.

Subject In-Charge

In-Charge,BSc(IT)

Date

College Seat

Examiner

INDEX

Practical: 1

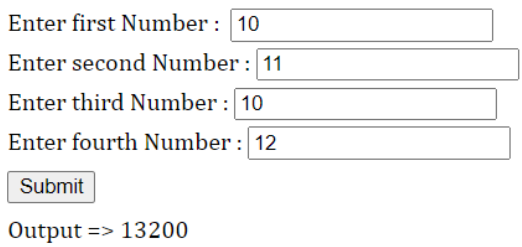
Working with basic C# and ASP .NET

1.A) Create an application that obtains four int values from the user and displays the product.

Source Code :

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

namespace practical1
{
   public partial class practical1a : System.Web.UI.Page
   {
      protected void Page_Load(object sender, EventArgs e)
      {

      }

      protected void btn_submit_Click(object sender, EventArgs e)
      {
         double num1, num2, num3, num4, ans;
         double.TryParse(txt_firstNum.Text, out num1);
         double.TryParse(txt_secondNum.Text, out num2);
         double.TryParse(txt_thirdNum.Text, out num3);
         double.TryParse(txt_fouthNum.Text, out num4);
         ans = num1*num2*num3*num4;
         lbl_ans.Text = ans.ToString();
      }
   }
}
```

Output:

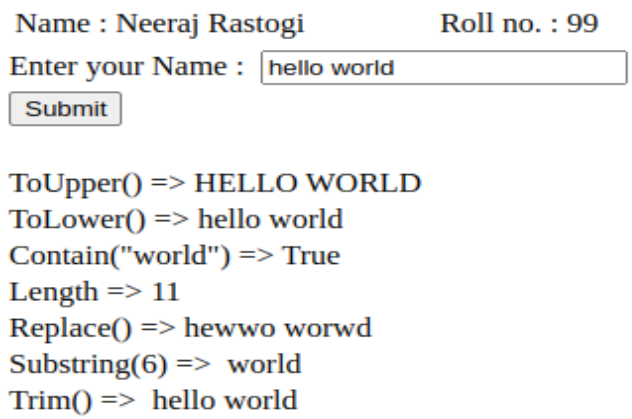


1.B) Create an application to demonstrate string operations.

Source Code :

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

namespace practical1
{
   public partial class practical1b : System.Web.UI.Page
   {
      protected void Page_Load(object sender, EventArgs e)
      {

      }

      protected void btn_submit_Click(object sender, EventArgs e)
      {
         String word = txt_name.Text;
         lbl_uppercase.Text = word.ToUpper();
         lbl_lowercase.Text = word.ToLower();
         lbl_contain.Text = word.Contains("world").ToString();
         lbl_length.Text = word.Length.ToString();
         lbl_Replace.Text = word.Replace("l","w");
         lbl_Substring.Text = word.Substring(6);
         lbl_trim.Text = word.Trim();
      }
   }
 }
```

Output:



1. **C) Create an application that receives the (Student Id, Student Name, Course Name, Date of Birth) information from a set of students. The application should also display the information of all the students once the data entered.**

Source Code :

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

namespace practical1
{
   struct student
   {
      public string id, name, cname, dob;
   }

   public partial class practical1c : System.Web.UI.Page
   {
      static student[] s = new student[2];
      static int i;
      protected void Page_Load(object sender, EventArgs e)
      {

      }

      protected void btn_submit_Click(object sender, EventArgs e)
      {
         Response.Write($"<h1>i={i}</h1>");
         s[i].id = txt_studentId.Text;
         s[i].name = txt_studentName.Text;
         s[i].cname = txt_courseName.Text;
         s[i].dob = txt_dob.Text;
         i++;
      }

      protected void btn_display_Click(object sender, EventArgs e)
      {
         for(int j = 0; j < i; j++)
         {


Response.Write("<br>==========================================
========<br>");
            Response.Write($"i={j} <br>\nstudent id : {s[j].id} <br>\nstudent name :
{s[j].name}<br>\ncourse name : {s[j].cname}<br>\nDate of Birth : {s[j].dob}");



Response.Write("<br>==========================================
========<br>");
```

```
        }
    }
  }
}
```

Output:

==========
i=0
student id : 123
student name : xyz
course name : bscit
Date of Birth : 01/05/2003
==========

==========
i=1
student id : 126
student name : abc
course name : bscit
Date of Birth : 02/05/2004
==========

Name : Neeraj Rastogi Roll no. : 99

Enter Student Id : 126
Enter Student Name : abc
Enter Course Name : bscit
Date of Birth : 02/05/2004
Display Submit

1. (D) Create an application to demonstrate following operations :

i. Generate Fibonacci series.
ii. Test for vowel.
iii. Test For Prime numbers.
iv.Use of foreach loop with arrays.
v. Reverse a number and find sum of digits of a number.

Source Code :

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

namespace practical1
{
   public partial class practical1d : System.Web.UI.Page
   {
      public bool checkPrime(int n)
      {
         bool prime = false;
         for (int i = 2; i < n / 2; i++)
         {
            if (n % i == 0)
            {
               prime = true;
               break;
            }
         }
         return prime;
      }
      protected void Page_Load(object sender, EventArgs e)
      {



      }


      protected void Button1_Click(object sender, EventArgs e)
      {
         int num1 = 0;
         int num2 = 1;
         lbl_fsAns.Text += "0 1";
         for (int i = 2; i < int.Parse(txt_input.Text); i++)
         {
            int ans = num1 + num2;
            lbl_fsAns.Text += " " + ans;
            num1 = num2;
            num2 = ans;
         }
      }
```

```
    protected void btn_PrimeNum_Click(object sender, EventArgs e)
    {
       practical1d obj = new practical1d();
        int m = int.Parse(txt_pNumInput.Text);
        bool isPrime = obj.checkPrime(m);
       if (isPrime == true)
            {
                lbl_pnAns.Text = $"No, {m} is Not Prime Number";
            }
            else
            {
                lbl_pnAns.Text = $"Yes, {m} is Prime Number";
            }
        }

protected void btn_checkVowel_Click(object sender, EventArgs e)
        {
            char c = txt_vowelInput.Text[0];
            if(c =='a' || c == 'e' || c == 'i' || c == 'o' || c == 'u' || c
== 'A' || c == 'E' || c == 'I' || c == 'O' || c == 'U')
            {
                lbl_vowelAns.Text = $"Yes, {c} letter is Vowel";
            }
            else
            {
                lbl_vowelAns.Text = $"No, {c} letter is Not Vowel";
            }
        }

protected void btn_foreach_Click(object sender, EventArgs e)
        {
            int[] num = new int[] { 27, 1, 2, 3, 44, 5 };
            foreach(int i in num)
            {
                lbl_foreach.Text += $" {i}";
            }
        }

        protected void btn_reverse_Click(object sender, EventArgs e)
        {
            int n = int.Parse(txt_reverse.Text);
            int reverse = 0, reminder = 0;
            while (n>0) {
                reminder = n % 10;
                reverse = reverse * 10 + reminder;
                n = n / 10;
            }
            lbl_reverse.Text = reverse.ToString();
        }

protected void btn_sumOfDigit_Click(object sender, EventArgs e)
        {
            int n = int.Parse(txt_sumOfDigit.Text);
            int sum = 0, reminder = 0;
            while (n > 0)
```

```
            {
                reminder = n % 10;
                sum = sum + reminder;
                n = n / 10;
            }
            lbl_sumOfDigit.Text = sum.ToString();
        }
    }
}
```

Output:

Name :Neeraj Rastogi Roll no. : 99

Fibonacci Series :

Enter Number : 10 [Fibonacci Series] => **0 1 1 2 3 5 8 13 21 34**

Prime Number :

Enter Number : 54 [Prime Number] => **No, 54 is Not Prime Number**

Vowels :

Enter Char : hello [Vowels] => **No, h letter is Not Vowel**

Foreach Loop :

[27, 1, 2, 3, 44, 5] [Foreach Loop] => **27 1 2 3 44 5**

Reverse :

Enter Number : 4321 [Reverse] => **1234**

Sum of Digits :

Enter Number : 16243 [Sum of Digits] => **16**

Practical: 2

Working with Object Oriented C# and ASP .NET

2 .A) Create simple application to perform following operations

Source Code :

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


class QuadraticEquationClass
{
   public double a, b, c, r1, r2;
   public double compute()
   {
      double d1;
      d1 = b*b-4*a*c;
      if (d1 == 0)
      {
         r1 = r2 = (-b) / (2 * a);
         return d1;
      }
      else if (d1 > 0)
      {
         r1 = (-b + Math.Sqrt(d1))/(2*a);
         return d1;
      }
      else
      {
         r1 = (-b) / (2 * a);
         r2 = Math.Sqrt(d1) / (2 * a);
         return d1;
      }
   }
}


namespace practical2
{
   public partial class practical2a : System.Web.UI.Page
   {
      protected void Page_Load(object sender, EventArgs e)
      {



      }

      public int factorialFunc(int num)
      {
```

```
        if (num == 0 || num == 1)
        {
            return 1;
        }
        else
        {
            return num * factorialFunc(num - 1);
        }
    }
    protected void Button1_Click(object sender, EventArgs e)
    {
        int factorialNumber = int.Parse(txt_takeFactorial.Text);
        lbl_factorialAnswer.Text = factorialFunc(factorialNumber).ToString();
    }

    protected void btn_inrToDoller_Click(object sender, EventArgs e)
    {
        lbl_doller.Text = Convert.ToDouble(double.Parse(txt_inr.Text)/83).ToString();
    }

    protected void btn_dollerToInr_Click(object sender, EventArgs e)
    {
        lbl_inr.Text = Convert.ToDouble(double.Parse(txt_doller.Text) * 83).ToString();
    }

    protected void btn_CtoF_Click(object sender, EventArgs e)
    {
        lbl_fahrenheit.Text = ((int.Parse(txt_celsius.Text) * 9 / 5) +32).ToString();
    }

    protected void btn_FtoC_Click(object sender, EventArgs e)
    {
        lbl_celsius.Text = ((int.Parse(txt_fahrenheit.Text) - 32) * 5 / 9).ToString();
    }

    protected void btn_qeAnswer_Click(object sender, EventArgs e)
    {
        QuadraticEquationClass q = new QuadraticEquationClass();
        q.a = int.Parse(txt_qe1.Text);
        q.b = int.Parse(txt_qe2.Text);
        q.c = int.Parse(txt_qe3.Text);
        double d = q.compute();
        if (d == 0)
        {
            lbl_qeAnswer.Text = $"roots are real and equal <br>first and second root is
{q.r1}";
        }
        else if (d < 0)
        {
            lbl_qeAnswer.Text = $"roots are real and distinct<br>First root is
{q.r1}<br>second root is {q.r2}";
        }
        else
        {
            lbl_qeAnswer.Text = $"roots are Imaginary<br>first root is {q.r1}<br>second
root is {q.r2}";
```

```
        }
    }
  }
}
```

Output:

Name : Neeraj Rastogi Roll no. : 99

Factorial :

Enter the Number : 4 Factorial => **24**

Money Conversion :

INR : 100 ₹ to $ => **$1.20481927710843**

DOLLER : 234.23 $ to ₹ => **₹19441.09**

Quadratic Equation :

12 34 3 Answ

roots are Imaginary
first root is -0.0911688566177612
second root is 0

Temperature Conversion :

Enter celsius : 30 C to F => **86 °F**

Enter fahrenheit : 86 F to C => **30 °C**

2. B) Create simple application to demonstrate use of following concepts.

i. **Function Overloading**
ii. **Inheritance (all types)**
iii. **Constructor overloading**
iv. **Interfaces**

Source Code :

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

public class BaseClass
{
   public int b;
   public string baseMethod()
   {
      return "This is Base Class<br>";
   }
}

public class DerivedClass : BaseClass
{
   public int d;
   public string derivedMethod()
   {
      return "This is Derived Class";
   }
}




public class A  //base class
{
   public string funcA()
   {
      return "class A of Method";
   }
}
public class B : A
{
   public string funcB()
   {
      return "class B of Method";
   }
}
public class C : A
{
   public string funcC()
   {
```

```
        return "class C of Method";
    }
}



public class Class1
{
    public string cls1Func()
    {
        return "1st Class";
    }
}
public class Class2 : Class1
{
    public string cls2Func()
    {
        return "2nd Class";
    }
}

public class Class3 : Class2
{
    public string cls3Func()
    {
        return "3rd Class";
    }
}



public class Contructor
{
    public string str;
    public Contructor()
    {
        str = "default Contructor";
    }

    public Contructor(int a, int b)
    {
        str = Convert.ToString(a + b);
    }

    public Contructor(double a, double b, double c)
    {
        str = Convert.ToString(a + b + c);
    }

    public Contructor(string a, string b)
    {
        str = a+b;
    }
}
public interface IA //ineterface  1
{
```

```
   string setImgs(string a);
}
public interface IB  //Interface 2
{
   int getAmount(int Amt);
}
public class ICar : IA, IB //implementatin
{
   public int getAmount(int Amt)
   {
      return 100+Amt;
   }
   public string setImgs(string a)
   {
      return "this is the car name "+a;
   }
}


namespace practical2
{
   public partial class practical2b : System.Web.UI.Page
   {
      protected void Page_Load(object sender, EventArgs e)
      {

      }

      public int sum(int a, int b)
      {
         return a + b;
      }

      public double sum(double a,double b)
      {
         return a + b;
      }

      public String sum(String a, String b)
      {
         return a + b;
      }
      protected void btn_Answer_Click(object sender, EventArgs e)
      {
         lbl_sumInt.Text = Convert.ToString(sum(2,4));
         lbl_sumDouble.Text = Convert.ToString(sum(3.5,5.5));
         lbl_concat.Text = Convert.ToString(sum("Hello","World"));
      }

      protected void btn_singlelevel_Click(object sender, EventArgs e)
      {
         DerivedClass d = new DerivedClass();
         lbl_singlelevel.Text += d.baseMethod();
         lbl_singlelevel.Text += d.derivedMethod();
      }
```

```
        protected void btn_hirearchical_Click(object sender, EventArgs e)
        {
            C objc = new C();
            B objb = new B();
            lbl_HierarchicalInheritance.Text += "From object of class b to Calling
"+objb.funcA() + "<br>";
            lbl_HierarchicalInheritance.Text += "From object of class c to Calling " +
objc.funcA() + "<br>";
            lbl_HierarchicalInheritance.Text += "From object of class b to Calling " +
objb.funcB() + "<br>";
            lbl_HierarchicalInheritance.Text += "From object of class c to Calling " +
objc.funcC() + "<br>";
        }

        protected void btn_multilevelInheritance_Click(object sender, EventArgs e)
        {
            Class3 c3 = new Class3();
            lbl_multilevelInheritance.Text += "This is " + c3.cls1Func() + "<br>";
            lbl_multilevelInheritance.Text += "This is " + c3.cls2Func() + "<br>";
            lbl_multilevelInheritance.Text += "This is " + c3.cls3Func() + "<br>";
        }

        protected void btn_constructorOverloading_Click(object sender, EventArgs e)
        {
            Contructor c1 = new Contructor(2,4);
            Label1.Text = Convert.ToString(c1.str);

            Contructor c2 = new Contructor(12.5, 5.5, 3.2);
            Label2.Text = Convert.ToString(c2.str);

            Contructor c3 = new Contructor("hello", "world");
            Label3.Text = Convert.ToString(c3.str);
        }

        protected void btn_interface_Click(object sender, EventArgs e)
        {
            IA obj1 = new ICar();
            IB obj2 = new ICar();
            lbl_interface.Text += "interface1 => "+ Convert.ToString(obj1.setImgs("BMW"))
+"<br>";
            lbl_interface.Text += "interface2 => " + Convert.ToString(obj2.getAmount(10));
        }
    }
}
```

Output:

Name : Neeraj Rastogi Roll no. : 99

==========

Function Overloading :

sum(2,4) => **6**

sum(3.5,5.5) => **9**

sum("Hello","World") => **HelloWorld**

Answer

==========

Inheritance (all types) :

Single level inheritance => show

=> This is Base Class
This is Derived Class

Hierarchical inheritance => show

=> From object of class b to Calling class A of Method
From object of class c to Calling class A of Method
From object of class b to Calling class B of Method
From object of class c to Calling class C of Method

Multilevel inheritance => show

=> This is 1st Class
This is 2nd Class
This is 3rd Class
This is 1st Class
This is 2nd Class
This is 3rd Class

==========

Constructor Overloading :

Constructor(2, 4) => **6**

Constructor(12.5, 5.5, 3.2) => **21.2**

Constructor("hello", "world") => **helloworld**

Answer

==========

Interfaces :

show

=> interface1 => this is the car name BMW
interface2 => 110

2. C) Create simple application to demonstrate use of following concepts

i. **Using Delegates and events**
ii. **Exception handling**

Source Code :

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



delegate int NumberChanger(int n);
namespace practical2
{
   public partial class practical2c : System.Web.UI.Page
   {
      static int num = 10;

      public static int AddNum(int p)
      {
         num += p;
         return num;
      }
      public static int MultNum(int q)
      {
         num *= q;
         return num;
      }
      public static int getNum()
      {
         return num;
      }
      protected void Page_Load(object sender, EventArgs e)
      {

      }

      protected void Button1_Click(object sender, EventArgs e)
      {
         NumberChanger nc1 = new NumberChanger(AddNum);
         NumberChanger nc2 = new NumberChanger(MultNum);

         //calling the methods using the delegate objects
         try
         {
            nc1(25);
            lbl_delegateAnswer1.Text = "Value of Num: {0} => " +
Convert.ToString(getNum());
            nc2(5);
```

```
            lbl_delegateAnswer2.Text = "Value of Num: {0} => " +
Convert.ToString(getNum());
        }
        catch(Exception ex)
        {
            Response.Write(ex.Message);
        }
    }
  }
}
```

Output:

Name : Neeraj Rastogi Roll no. : 99

Delegates :

Value of Num: {0} => 35
Value of Num: {0} => 175

Button

Practical: 3

Working with Web Forms and Controls

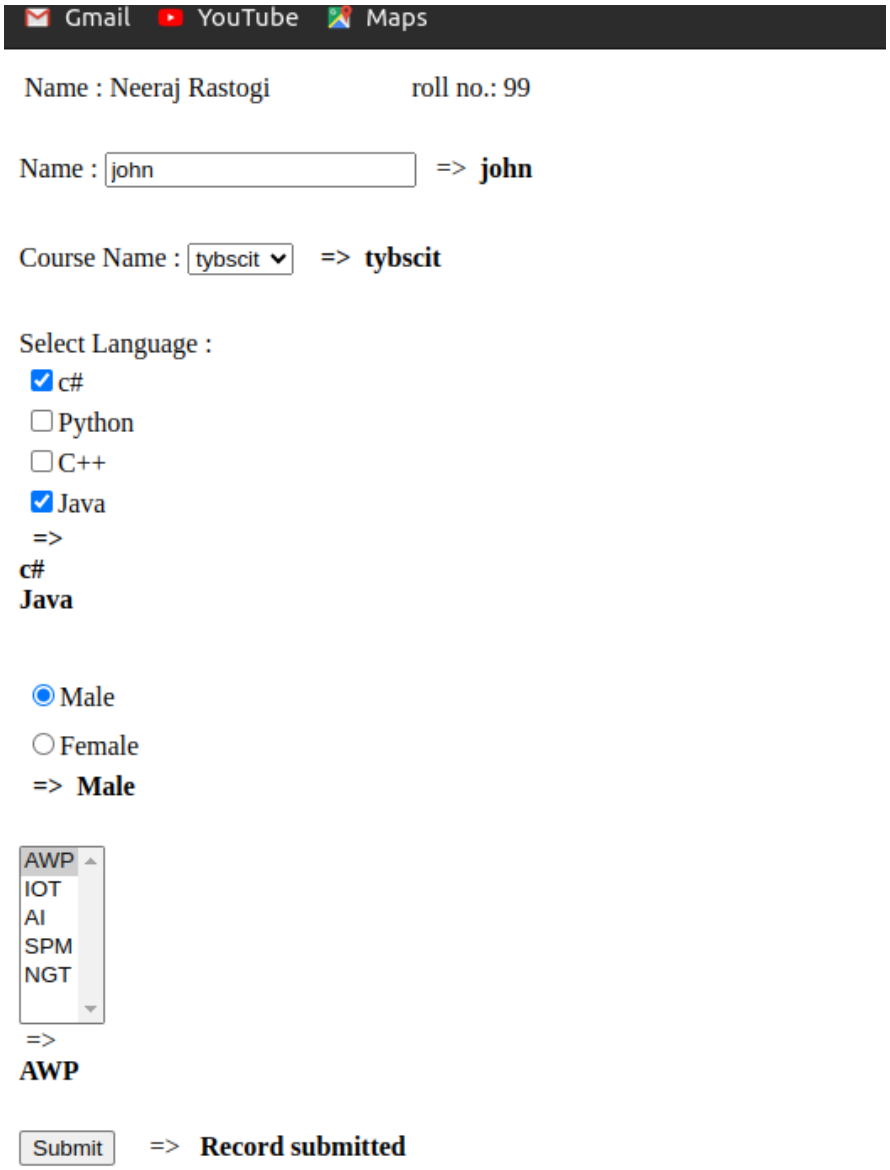
3. A) Create a simple web page with various sever controls to demonstrate setting and use of their properties. (Example : AutoPostBack).

Source Code :

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

namespace practical3
{
   public partial class practical3a : System.Web.UI.Page
   {
      protected void Page_Load(object sender, EventArgs e)
      {

      }

      protected void btn_submit_Click(object sender, EventArgs e)
      {
         lbl_submit.Text = "Record submitted";
      }

      protected void TextBox1_TextChanged(object sender, EventArgs e)
      {
         String str;
         str = txt_Name.Text;
         lbl_Name.Text = str;
      }

      protected void DropDownList1_SelectedIndexChanged(object sender, EventArgs e)
      {
         lbl_courseName.Text = DropDownList1.SelectedValue;
      }

      protected void CheckBoxList1_SelectedIndexChanged(object sender, EventArgs e)
      {
         lbl_checkboxList.Text = "";
         foreach (ListItem x in CheckBoxList1.Items)
         {
            if (x.Selected)
            {
               lbl_checkboxList.Text += $"<br> {x.Value}";
            }
         }
      }

      protected void RadioButtonList1_SelectedIndexChanged(object sender, EventArgs e)
```

```
        {
            lbl_radioButtonList.Text = RadioButtonList1.SelectedValue;
        }

        protected void ListBox1_SelectedIndexChanged(object sender, EventArgs e)
        {
            lbl_listBox.Text = "";
            foreach (ListItem x in ListBox1.Items)
            {
                if (x.Selected)
                {
                    lbl_listBox.Text += $"<br> {x.Value}";
                }
            }
        }
    }
}
```

Output:



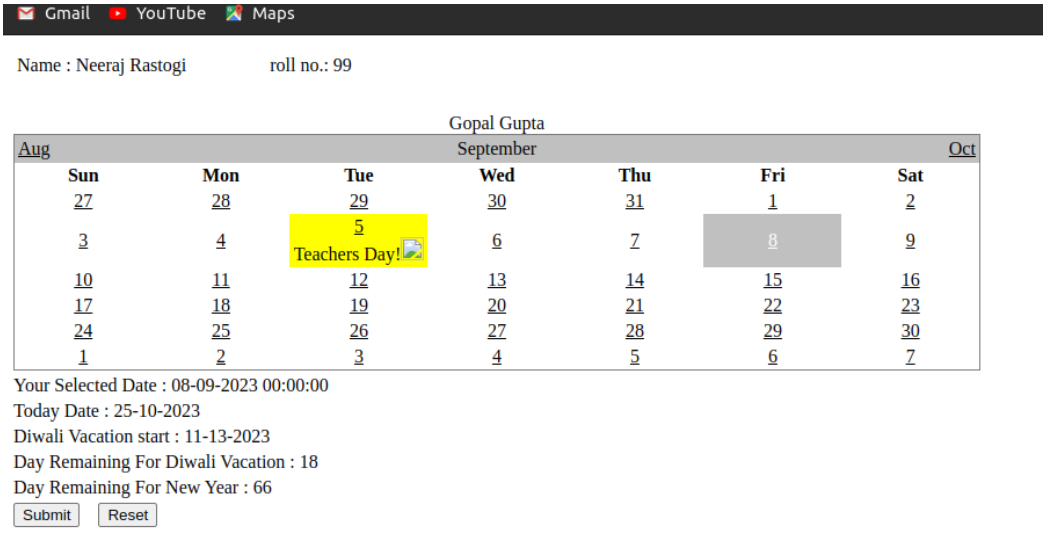
3. B)Demonstrate the use of Calendar control to perform following operations.
a) Display messages in a calendar control
b) Display vacation in a calendar control
c) Selected day in a calendar control using style
d) Difference between two calendar dates

Source Code :

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

namespace practical3
{
   public partial class practical3b : System.Web.UI.Page
   {
      protected void Page_Load(object sender, EventArgs e)
      {

      }

      protected void Calender1_DayRender(object sender, 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);
            Image g1 = new Image();
            g1.ImageUrl = "./images/teacherDay.jpg";
            g1.Height = 20;
            g1.Width = 20;
            e.Cell.Controls.Add(g1);
         }
         if (e.Day.Date.Day == 13 && e.Day.Date.Month == 11)
         {
            Calendar1.SelectedDate = new DateTime(2023, 11, 13);
            Calendar1.SelectedDates.SelectRange(Calendar1.SelectedDate,
            Calendar1.SelectedDate.AddDays(5));
            Label lbl1 = new Label();
            lbl1.Text = "<br>Diwali";
            e.Cell.Controls.Add(lbl1);
         }
      }
```

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

    protected void btn_submit_Click(object sender, EventArgs e)
    {
       Calendar1.Caption = "Gopal Gupta";
       Calendar1.FirstDayOfWeek = FirstDayOfWeek.Sunday;
       Calendar1.NextPrevFormat = NextPrevFormat.ShortMonth;
       Calendar1.TitleFormat= TitleFormat.Month;
       Label2.Text = $"Today Date : {Calendar1.TodaysDate.ToShortDateString()}";
       Label3.Text = "Diwali Vacation start : 11-13-2023 ";
       TimeSpan d = new DateTime(2023, 11, 13) - DateTime.Now;
       Label4.Text = $"Day Remaining For Diwali Vacation : {d.Days.ToString()} ";
       TimeSpan d1 = new DateTime(2023, 12, 31) - DateTime.Now;
       Label5.Text = $"Day Remaining For New Year : {d1.Days.ToString()} ";
       if (Calendar1.SelectedDate.ToShortDateString() == "11-13-2023")
       {
          Label3.Text = "<b>Diwali festival start</b>";
       }
       if (Calendar1.SelectedDate.ToShortDateString() == "11-18-2023")
       {
          Label3.Text = "<b>Diwali festival End</b>";
       }
    }
  }
}
```

Output:



3.C) Demonstrate the use of Treeview control perform following operations.

a) Treeview control and datalist
b) Treeview operations

Source Code :

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

namespace practical3
{
   public partial class practical3c : System.Web.UI.Page
   {
      protected void Page_Load(object sender, EventArgs e)
      {

      }

      protected void TreeView1_SelectedNodeChanged(object sender, EventArgs e)
      {
         lbl_result.Text = TreeView1.SelectedNode.Text;
      }
   }
}
```

Output:

Name : advait roll no.: 90

Asp.net practicals
Practical 3
Practical 3(A)
Practical 3(B)
Practical 3(C)
Java Practicals

=> **Java Practicals**

sid: 1
sname: abc
age: 20

sid: 2
sname: xyz
age: 22

sid: 3
sname: opq
age: 19

Practical : 4

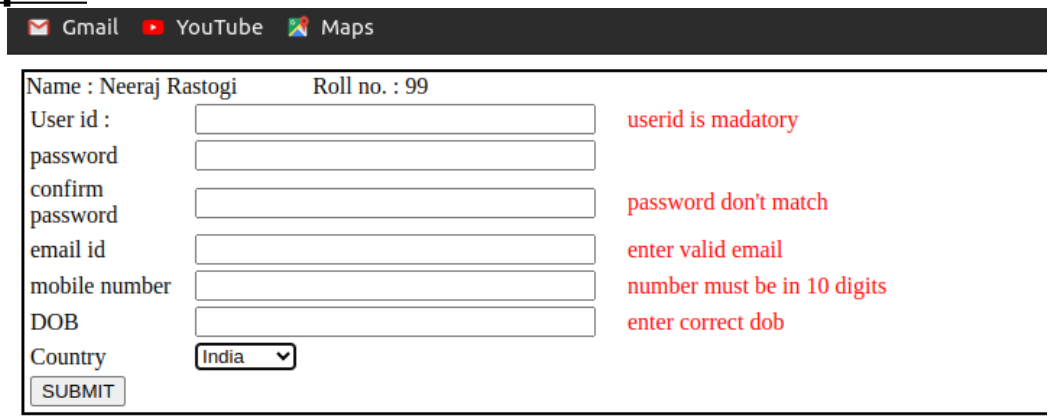
Working with Form Controls

4.A) Create a Registration form to demonstrate use of various Validation controls.

WebForm.aspx:

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

<%@ Register src="webusercontrol/Login.ascx" tagname="Login" tagprefix="uc1" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
    <style type="text/css">
        .auto-style1 {
            width: 100%;
        }
    </style>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            <table class="auto-style1">
                <tr>
                    <td style="border : 2px solid black">
                        Name : Gopal Gupta             Roll no. : 27<br />
                        <uc1:Login ID="Login1" runat="server"/>
                    </td>
                </tr>
                <tr>
                    <td></td>
                </tr>
                <tr>
                    <td style="border : 2px solid black">
                        Name : Gopal Gupta             Roll no. : 27<br />
                        <uc1:Login ID="Login2" runat="server"/>
                    </td>
                </tr>
            </table>
        </div>
    </form>
</body>
</html>
```

Output:


Gmail
YouTube
Maps
Name : Neeraj Rastogi Roll no. : 99
User id :
userid is madatory
password
confirm password
password don't match
email id
enter valid email
mobile number
number must be in 10 digits
DOB
enter correct dob
Country
India
SUBMIT

4.B) Create Web Form to demonstrate use of Adrotator Control.

Source Code :

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

namespace practical4
{
   public partial class practical4b : System.Web.UI.Page
   {
      protected void Page_Load(object sender, EventArgs e)
      {

      }

      protected void btn_next_Click(object sender, EventArgs e)
      {
         Page.Response.Redirect(Page.Request.Url.ToString(), true);
      }
   }
}
```

Output:

4.C) Create Web Form to demonstrate use User Controls.

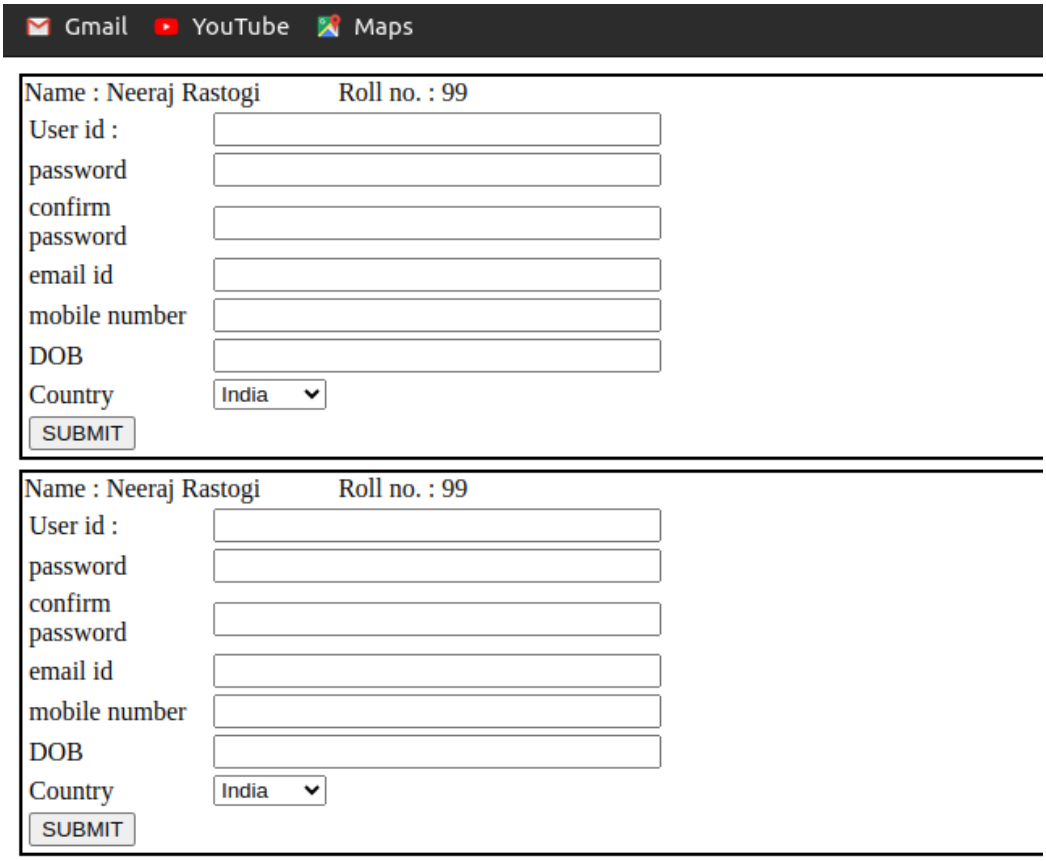
WebForm.ascx:

```
<%@ Control Language="C#" AutoEventWireup="true" CodeBehind="Login.ascx.cs"
Inherits="validator.Login" %>
<style type="text/css">
   .auto-style1 {
      width: 100%;
   }
   .auto-style2 {
      text-align: left;
   }
   .auto-style3 {
      width: 295px;
   }
   .auto-style4 {
      width: 110px;
   }
</style>

<table class="auto-style1">
   <tr>
      <td class="auto-style4">User id : </td>
      <td class="auto-style3">
         <asp:TextBox ID="txt_userId" runat="server" Width="277px"></asp:TextBox>
      </td>
      <td>
         <asp:RequiredFieldValidator ID="RequiredFieldValidator1" runat="server"
ControlToValidate="txt_userId" Display="Dynamic" ErrorMessage="user id is madatory"
ForeColor="Red" SetFocusOnError="True">userid is
madatory</asp:RequiredFieldValidator>
      </td>
   </tr>
   <tr>
      <td class="auto-style4">password</td>
      <td class="auto-style3">
         <asp:TextBox ID="txt_password" runat="server"
Width="277px"></asp:TextBox>
      </td>
      <td>
         <asp:RequiredFieldValidator ID="RequiredFieldValidator2" runat="server"
ControlToValidate="txt_password" Display="Dynamic" ErrorMessage="password is
madatory" ForeColor="Red" SetFocusOnError="True">password is
madatory</asp:RequiredFieldValidator>
      </td>
   </tr>
   <tr>
      <td class="auto-style4">confirm password</td>
      <td class="auto-style3">
         <asp:TextBox ID="txt_confirmpassword" runat="server"
Width="277px"></asp:TextBox>
      </td>
      <td>
```

```
            <asp:CompareValidator ID="CompareValidator1" runat="server"
ControlToCompare="txt_password" ControlToValidate="txt_confirmpassword"
Display="Dynamic" ErrorMessage="password don't match" ForeColor="Red"
SetFocusOnError="True" ValidateRequestMode="Enabled"
ViewStateMode="Enabled">password don&amp;#39;t match</asp:CompareValidator>
        </td>
    </tr>
    <tr>
        <td class="auto-style4">email id</td>
        <td class="auto-style3">
            <asp:TextBox ID="txt_email" runat="server" Width="277px"></asp:TextBox>
        </td>
        <td>
            <asp:RegularExpressionValidator ID="RegularExpressionValidator1"
runat="server" ControlToValidate="txt_email" Display="Dynamic" ErrorMessage="enter
valid email" ForeColor="Red" SetFocusOnError="True" ValidationExpression="\w+([-+.']\
w+)*@\w+([-.]\w+)*\.\w+([-.]\w+)*">enter valid
email</asp:RegularExpressionValidator>
        </td>
    </tr>
    <tr>
        <td class="auto-style4">mobile number</td>
        <td class="auto-style3">
            <asp:TextBox ID="txt_number" runat="server" Width="277px"></asp:TextBox>
        </td>
        <td>
            <asp:RegularExpressionValidator ID="RegularExpressionValidator2"
runat="server" ControlToValidate="txt_number" Display="Dynamic"
ErrorMessage="number must be in 10 digits" ForeColor="Red" SetFocusOnError="True"
ValidationExpression="\d{10}">number must be in 10
digits</asp:RegularExpressionValidator>
        </td>
    </tr>
    <tr>
        <td class="auto-style4">DOB</td>
        <td class="auto-style3">
            <asp:TextBox ID="txt_dateofbirth" runat="server"
Width="277px"></asp:TextBox>
        </td>
        <td>
            <asp:RangeValidator ID="RangeValidator1" runat="server"
ControlToValidate="txt_dateofbirth" Display="Dynamic" ErrorMessage="enter correct
dob" ForeColor="Red" SetFocusOnError="True">enter correct
dob</asp:RangeValidator>
        </td>
    </tr>
    <tr>
        <td class="auto-style4">Country</td>
        <td class="auto-style3">
            <asp:DropDownList ID="ddl_country" runat="server">
                <asp:ListItem>India</asp:ListItem>
                <asp:ListItem>america</asp:ListItem>
                <asp:ListItem>usa</asp:ListItem>
                <asp:ListItem>dubai</asp:ListItem>
            </asp:DropDownList>
        </td>
```

```
      <td> </td>
    </tr>
    <tr>
      <td class="auto-style2" colspan="3">
        <asp:Button ID="Button1" runat="server" Text="SUBMIT" />
      </td>
    </tr>
</table>
```

Output:



Practical : 5

Working with Navigation, Beautification and Master page.

5.A) Create Web Form to demonstrate use of Website Navigation controls and Site Map.

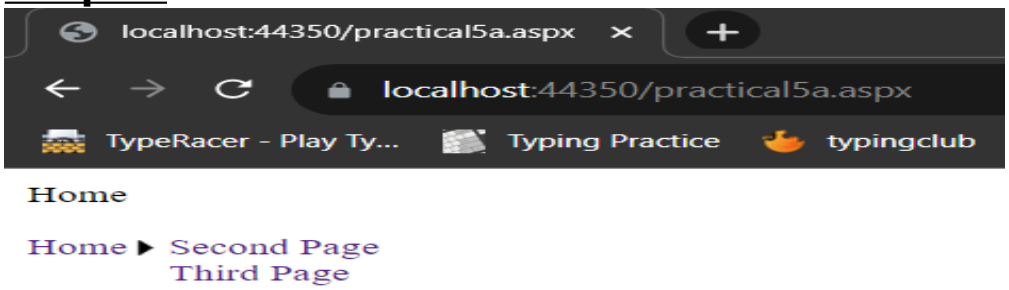
Web.sitemap:

```
<?xml version="1.0" encoding="utf-8" ?>
<siteMap xmlns="http://schemas.microsoft.com/AspNet/SiteMap-File-1.0" >
  <siteMapNode url="practical5a.aspx" title="Home"  description="Home">
    <siteMapNode url="WebForm2.aspx" title="Second Page"  description="Second Page"
/>
    <siteMapNode url="WebForm3.aspx" title="Third Page"  description="Third Page" />
  </siteMapNode>
</siteMap>
```

Home.aspx:

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="practical5a.aspx.cs"
Inherits="practical5.practical5a" %>

<!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">
            </asp:SiteMapPath>
            <br />
            <br />
            <asp:Menu ID="Menu1" runat="server" DataSourceID="SiteMapDataSource1">
            </asp:Menu>
            <br />
            <br />
            <asp:SiteMapDataSource ID="SiteMapDataSource1" runat="server" />

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

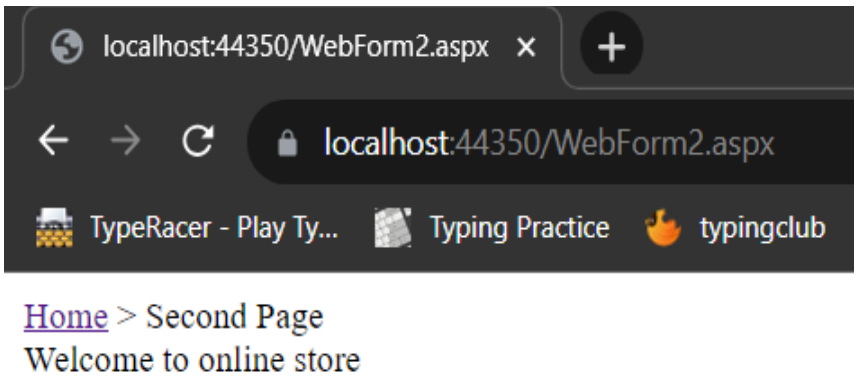
Output:



WebForm2.aspx:

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="WebForm2.aspx.cs"
Inherits="practical5.WebForm2" %>

<!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">
            </asp:SiteMapPath>
            <br/>Welcome to online store
        </div>
    </form>
</body>
</html>
```

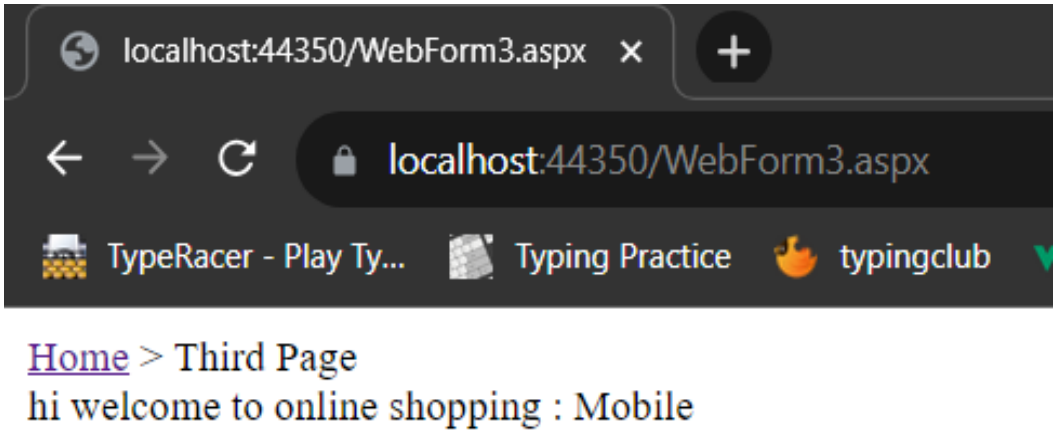
Output:



WebForm3.aspx:

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="WebForm3.aspx.cs"
Inherits="practical5.WebForm3" %>

<!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">
            </asp:SiteMapPath>
            <br/>hi welcome to online shopping : Mobile
        </div>
    </form>
</body>
</html>
```

Output:



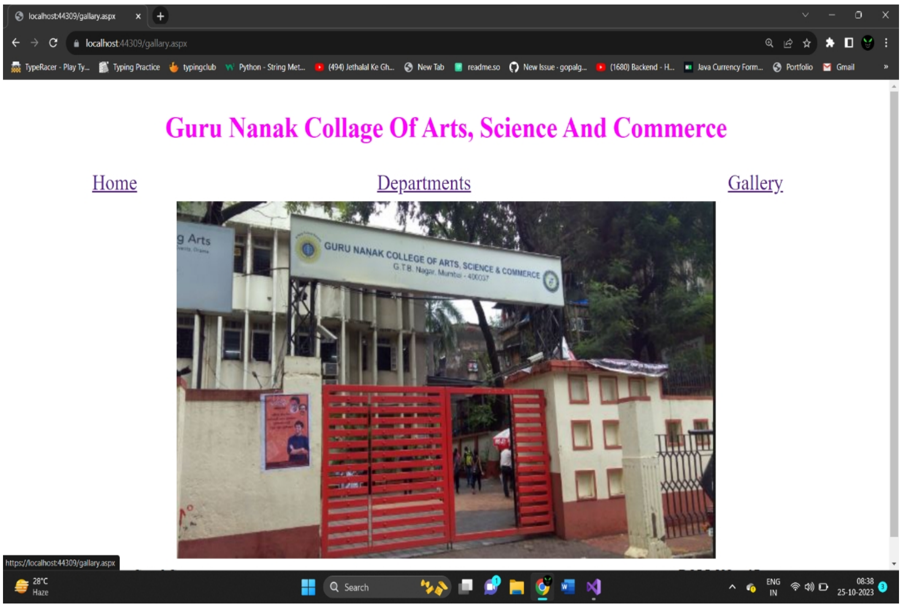
5.B) Create a web application to demonstrate use of Master Page with applying Styles and Themes for page beautification.

MasterPage.master:

```
<%@ Master Language="C#" AutoEventWireup="true"
CodeBehind="masterPage.master.cs" Inherits="masterPagePractical2.masterPage" %>

<!DOCTYPE html>

<html>
<head runat="server">
   <title></title>
   <asp:ContentPlaceHolder ID="head" runat="server">
   </asp:ContentPlaceHolder>
   <style type="text/css">
      .auto-style1 {
         width: 100%;
      }
      .auto-style2 {
         text-align: center;
      }
      .auto-style3 {
         color: #FF00FF;
         font-size: xx-large;
      }
      .auto-style4 {
         font-size: x-large;
      }
   </style>
</head>
<body>
   <form id="form1" runat="server">

      <div>
            <table class="auto-style1">
               <tr>
                  <td class="auto-style2">
                     <h1 class="auto-style3">Guru Nanak Collage Of Arts, Science And
Commerce</h1>
                  </td>
               </tr>
               <tr>
                  <td>
                     <table class="auto-style1">
                        <tr>
                           <td class="auto-style2">
                              <asp:HyperLink ID="HyperLink1" runat="server"
NavigateUrl="~/Home.aspx" CssClass="auto-style4">Home</asp:HyperLink>
                           </td>
                           <td class="auto-style2">
                              <asp:HyperLink ID="HyperLink2" runat="server"
NavigateUrl="~/department.aspx"
CssClass="auto-style4">Departments</asp:HyperLink>
```

```
                                    </td>
                                    <td class="auto-style2">
                                        <asp:HyperLink ID="HyperLink3" runat="server"
NavigateUrl="~/gallary.aspx" CssClass="auto-style4">Gallery</asp:HyperLink>
                                    </td>
                                </tr>
                            </table>
                        </td>
                    </tr>
                    <tr>
                        <td><asp:ContentPlaceHolder ID="ContentPlaceHolder1"
runat="server"></asp:ContentPlaceHolder></td>
                    </tr>
                    <tr>
                        <td class="auto-style2" ><strong>Gopal Gutpa</strong>
<strong>           &
nbsp;            
;            &nb
sp;            &
nbsp;            
;            &nb
sp;            &
nbsp;            
;            &nb
sp;            &
nbsp;            
;            &nb
sp;    ROLL NO. : 27</strong></td>
                    </tr>
                </table>
        </div>
    </form>
</body>
</html>
```

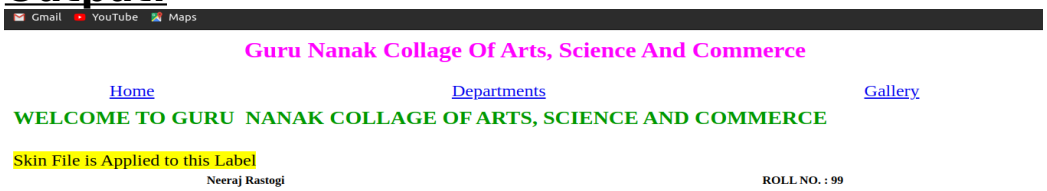
Output:



SkinTheme.skin:

```
<asp:Label runat="server" BackColor="Yellow" Font-Size="X-Large"/>
```

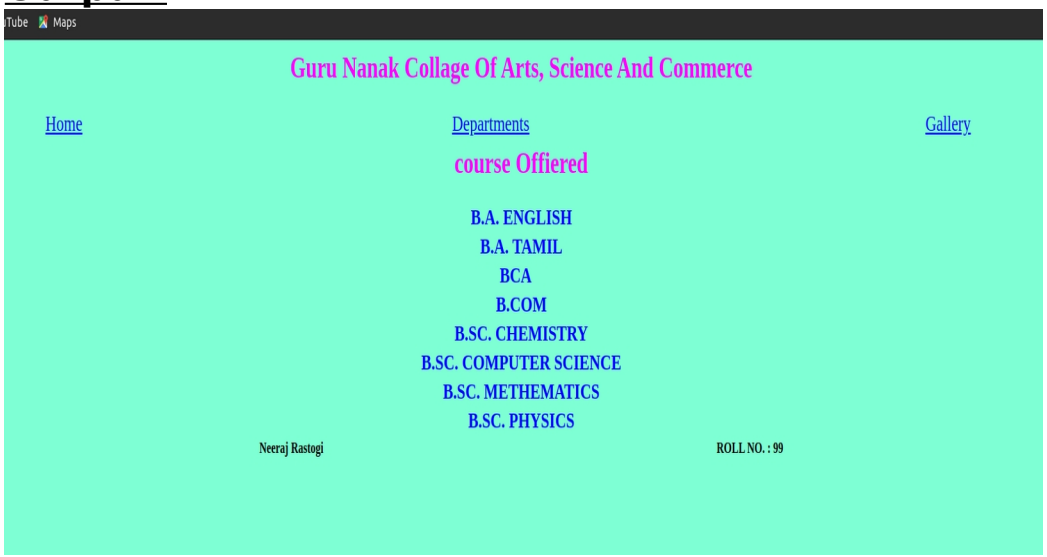
Output:



StyleSheet1.css:

```
body {
    background-color: aquamarine;
    font-family: 'Monotype Corsiva';
    font-size: larger;
}
```

Output:



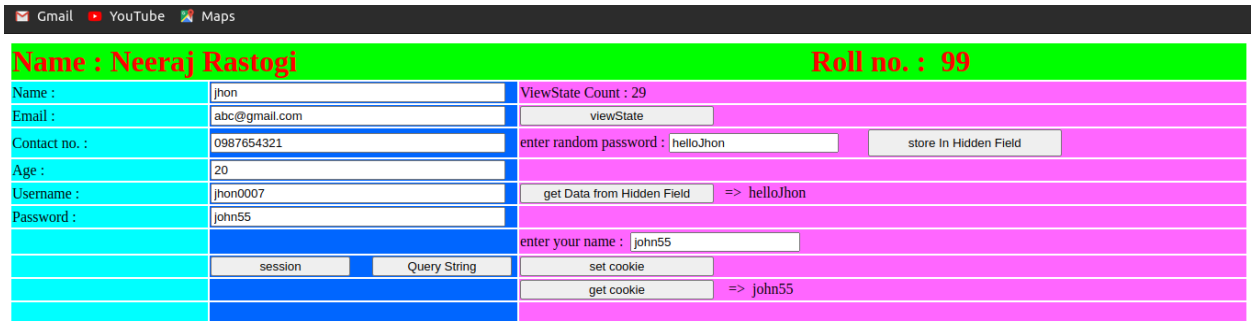
5.C) Create a web application to demonstrate various states of ASP.NET Pages.

Source Code :

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

namespace awptest1
{
   public partial class signup : System.Web.UI.Page
   {
      protected void Page_Load(object sender, EventArgs e)
      {

      }

      protected void btn_viewstate_Click(object sender, EventArgs e)
      {
         if (ViewState["Gopal"]!=null) {
            ViewState["Gopal"] = Convert.ToInt32(ViewState["Gopal"])+1;
         }else
         {
            ViewState.Add("Gopal", 27);
         }
         lbl_viewstatecounter.Text = ViewState["Gopal"].ToString();
      }

      protected void btn_session_Click(object sender, EventArgs e)
      {
         Session["name"] =txt_name.Text;
         Session["email"] =txt_email.Text;
         Session["contact"] =txt_contact.Text;
         Session["age"] =txt_age.Text;
         Session["username"] =txt_username.Text;
         Session["password"] =txt_password.Text;
         Response.Redirect("showDataThroughSession.aspx");
      }

      protected void btn_querystring_Click(object sender, EventArgs e)
      {
         Response.Redirect("showDataThroughQS.aspx?name=" + txt_name.Text +
"&email=" + txt_email.Text + "&contact=" + txt_contact.Text + "&age=" +txt_age.Text
+ "&username=" + txt_username.Text +"&password=" + txt_password.Text);
      }

      protected void btn_setHiddenField_Click(object sender, EventArgs e)
      {
         HiddenField1.Value=txt_hiddenfield.Text;
      }
```

```
    protected void btn_getHiddenField_Click(object sender, EventArgs e)
    {
        lbl_displayHiddenField.Text = HiddenField1.Value;
    }

    protected void btn_setCookie_Click(object sender, EventArgs e)
    {
        Response.Cookies["name"].Value = txt_cookie.Text;
    }

    protected void btn_getCookie_Click(object sender, EventArgs e)
    {
        lbl_getCookie.Text=Request.Cookies["name"].Value;
    }
  }
}
```

Output:



Practical : 6

Working with Database

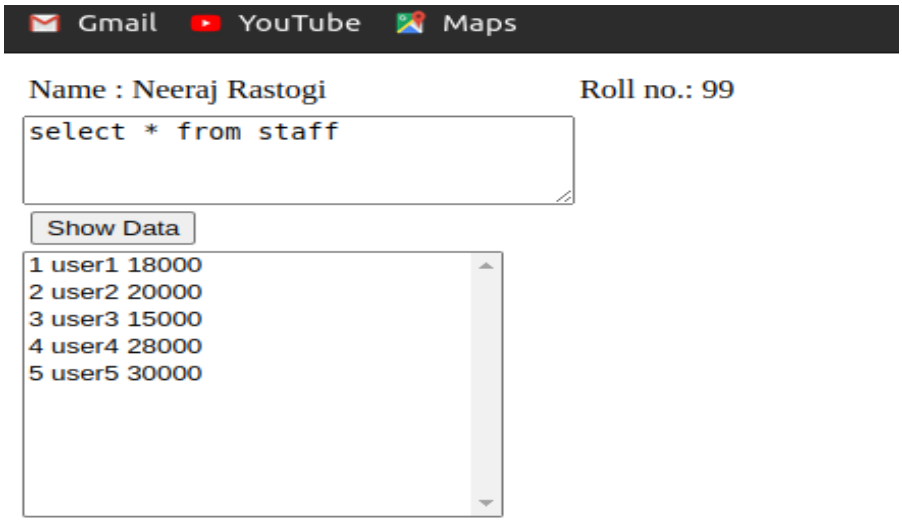
6.A) Create a web application bind data in a multiline textbox by querying in another textbox.

Source Code :

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data;
using System.Data.SqlClient;
using System.Web.Configuration;

namespace practical6
{
   public partial class practical6a : System.Web.UI.Page
   {
      protected void Page_Load(object sender, EventArgs e)
      {

      }

      protected void btn_showData_Click(object sender, EventArgs e)
      {
         try
         {
            //establish connection
            String str =
WebConfigurationManager.ConnectionStrings["ConnectionString"].ConnectionString;
            SqlConnection con = new SqlConnection(str);
            con.Open();
            //create sqlcommand
            SqlCommand cmd = new SqlCommand(txt_takeval.Text, con);
            //execute query
            SqlDataReader dr = cmd.ExecuteReader();
            //clear a listbox
            lb_showData.Items.Clear();
            while(dr.Read())
            {
               String itemstr = "";
               for (int i=0;i<dr.FieldCount;i++)
               {
                  itemstr = itemstr + " " + dr[i].ToString();
               }
               lb_showData.Items.Add(itemstr);
            }
            con.Close();
         }
         catch (Exception ex)
         {
            lb_showData.Items.Clear();
```

```
            lb_showData.Items.Add("Invalid Query"+ ex.Message);
        }
    }
  }
}
```

Output:

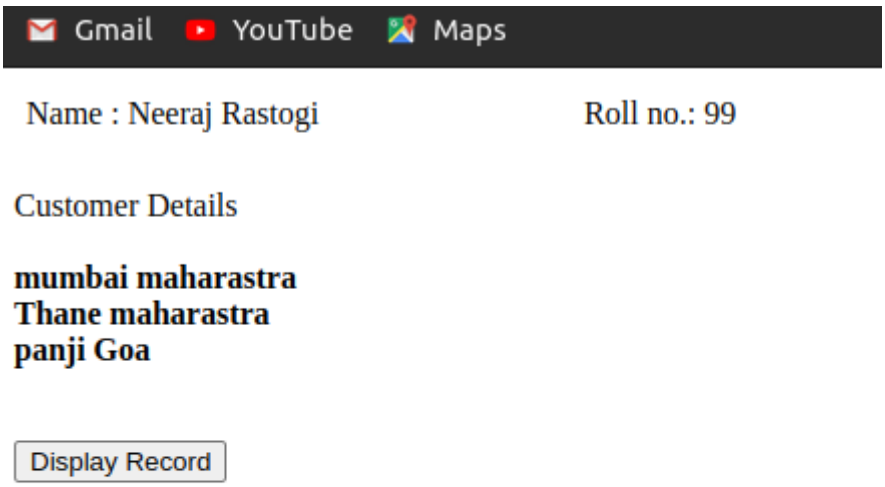


6.B) Create a web application to display records by using database.

Source Code :

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data;
using System.Data.SqlClient;
using System.Web.Configuration;

namespace practical6
{
   public partial class practical6b : System.Web.UI.Page
   {
      protected void Page_Load(object sender, EventArgs e)
      {

      }

      protected void Button1_Click(object sender, EventArgs e)
      {
         try
         {
            //establish connection
            String str =
WebConfigurationManager.ConnectionStrings["ConnectionString"].ConnectionString;
            SqlConnection con = new SqlConnection(str);
            con.Open();
            SqlCommand cmd = new SqlCommand("select city,state from customer", con);
            SqlDataReader dr = cmd.ExecuteReader();
            lbl_output.Text = "No record found";
            string output = "";
            while (dr.Read())
            {
               output += dr[0] + " " + dr[1] + "<br>";
               if (output == "")
               {
                  lbl_output.Text = "Record Not found";
               }else{
                  lbl_output.Text = output;
               }
            }
         }
         catch (Exception ex)
         {
            lbl_output.Text = $"Contact Admin {ex.Message}";
         }
      }
   }
}
```

Output:


Gmail
YouTube
Maps
Name : Neeraj Rastogi
Roll no.: 99
Customer Details
mumbai maharastra
Thane maharastra
panji Goa
Display Record

6.C) Demonstrate the use of Datalist link control.

DataList.aspx:

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="practical6c.aspx.cs"
Inherits="practical6.practical6c" %>

<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
   <title></title>
   <style type="text/css">
      .auto-style1 {
         width: 100%;
      }
      .auto-style2 {
         width: 429px;
      height: 20px;
   }
   .auto-style3 {
      height: 20px;
   }
   </style>
</head>
<body>
   <form id="form1" runat="server">
      <table class="auto-style1">
         <tr>
            <td>
               <table class="auto-style1">
                  <tr>
                     <td class="auto-style2">Name : Gopal
Gupta           &nbs
p;            &n
bsp;        Roll no.: 27</td>
                     <td class="auto-style3"></td>
                  </tr>
               </table>
            </td>
         </tr>
         <tr>
            <td> </td>
         </tr>
         <tr>
            <td>
               <asp:DataList ID="DataList1" runat="server"
BackColor="LightGoldenrodYellow" BorderColor="Tan" BorderWidth="1px"
CellPadding="2" DataKeyField="Id" DataSourceID="SqlDataSource1"
ForeColor="Black">
                  <AlternatingItemStyle BackColor="PaleGoldenrod" />
                  <FooterStyle BackColor="Tan" />
                  <HeaderStyle BackColor="Tan" Font-Bold="True" />
                  <ItemTemplate>
                     Id:
                     <asp:Label ID="IdLabel" runat="server" Text='<%# Eval("Id") %>' />
```

```
                    <br />
                    cname:
                    <asp:Label ID="cnameLabel" runat="server" Text='<%#
Eval("cname") %>' />
                    <br />
                    cemail:
                    <asp:Label ID="cemailLabel" runat="server" Text='<%#
Eval("cemail") %>' />
                    <br />
                    city:
                    <asp:Label ID="cityLabel" runat="server" Text='<%# Eval("city")
%>' />
                    <br />
                    state:
                    <asp:Label ID="stateLabel" runat="server" Text='<%# Eval("state")
%>' />
                    <br />
<br />
                </ItemTemplate>
                <SelectedItemStyle BackColor="DarkSlateBlue"
ForeColor="GhostWhite" />
            </asp:DataList>
        </td>
    </tr>
    <tr>
        <td> </td>
    </tr>
    <tr>
        <td>
            <asp:SqlDataSource ID="SqlDataSource1" runat="server"
ConnectionString="<%$ ConnectionStrings:ConnectionString %>"
SelectCommand="SELECT * FROM [customer]"></asp:SqlDataSource>
        </td>
    </tr>
</table>
<div>
</div>
</form>
</body>
</html>
```

Output:


Gmail
YouTube
Maps
Name : Neeraj Rastogi
Roll no.: 99
Id: 1
cname: abc
cemail: abc@gmail.com
city: mumbai
state: maharastra
Id: 2
cname: xyz
cemail: xyz@gamil.com
city: Thane
state: maharastra
Id: 3
cname: mno
cemail: mon@gmail.com
city: panji
state: Goa

Practical : 7

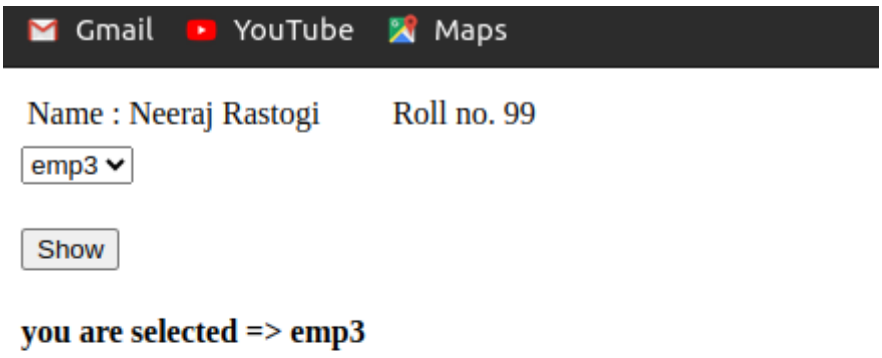
Working with Database

7.A) Create a web application to display Databinding using dropdownlist control.

Source Code :

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data;
using System.Data.SqlClient;
using System.Web.Configuration;

namespace practical7
{
   public partial class practical7a : System.Web.UI.Page
   {
      protected void Page_Load(object sender, EventArgs e)
      {
         if (!IsPostBack)
         {
            try
            {
               string conStr =
WebConfigurationManager.ConnectionStrings["ConnectionString"].ConnectionString;
               SqlConnection con = new SqlConnection(conStr);
               con.Open();
               SqlCommand cmd = new SqlCommand("select * from emp", con);
               SqlDataReader dr = cmd.ExecuteReader();
               DropDownList1.DataSource = dr;
               DropDownList1.DataTextField = "Ename";
               DropDownList1.DataBind();
               con.Close();
            }
            catch(Exception ex)
            {
               lbl_output.Text = ex.Message;
            }
         }
      }

      protected void btn_show_Click(object sender, EventArgs e)
      {
         lbl_output.Text = "you are selected => "+DropDownList1.SelectedValue;
      }
   }
}
```

Output:

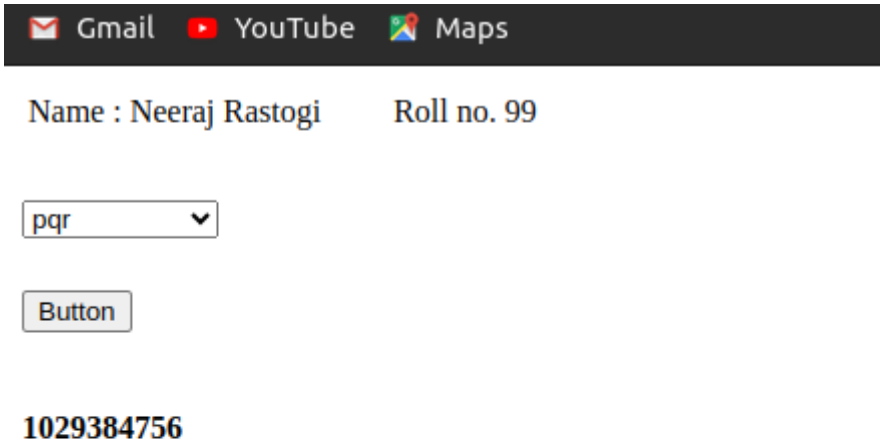

Gmail
YouTube
Maps
Name : Neeraj Rastogi Roll no. 99
emp3
Show
you are selected => emp3

7.B) Create a web application for to display the phone no of an author using database.

Source Code :

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data;
using System.Data.SqlClient;
using System.Web.Configuration;

namespace practical7
{
   public partial class pratical7b : System.Web.UI.Page
   {
      protected void Page_Load(object sender, EventArgs e)
      {
         if (!IsPostBack)
         {
            try
            {
               string conStr =
WebConfigurationManager.ConnectionStrings["ConnectionString"].ConnectionString;
               SqlConnection con = new SqlConnection(conStr);
               con.Open();
               SqlCommand cmd = new SqlCommand("select * from authers", con);
               SqlDataReader dr = cmd.ExecuteReader();
               DropDownList1.DataSource = dr;
               DropDownList1.DataTextField = "authname";
               DropDownList1.DataValueField = "phoneno";
               DropDownList1.DataBind();
               con.Close();
            }
            catch (Exception ex)
            {
               lbl_show.Text = ex.Message;
            }
         }
      }

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

Output:


Gmail
YouTube
Maps
Name : Neeraj Rastogi Roll no. 99
pqr
Button
1029384756

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

Source Code :

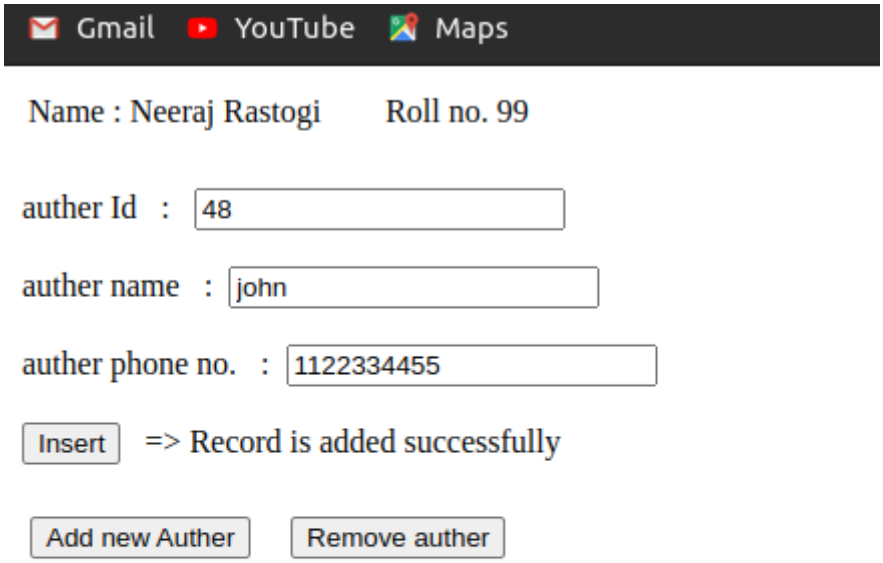
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data;
using System.Data.SqlClient;
using System.Web.Configuration;

namespace practical7
{
   public partial class practical7c : System.Web.UI.Page
   {
      protected void Page_Load(object sender, EventArgs e)
      {

      }

      protected void btn_add_Click(object sender, EventArgs e)
      {
         MultiView1.SetActiveView(View1);

      }

      protected void btn_remove_Click(object sender, EventArgs e)
      {
         MultiView1.SetActiveView(View2);
         updateDropDownList();
         lbl_delete.Text = "";
         lbl_name.Text = "";
      }

      protected void btn_Insert_Click(object sender, EventArgs e)
      {
         try
         {
            string conStr =
WebConfigurationManager.ConnectionStrings["ConnectionString"].ConnectionString;
            SqlConnection con = new SqlConnection(conStr);
            con.Open();
            SqlCommand cmd = new SqlCommand("insert into authers values(@authId,
@authname, @phoneno)", con);
            cmd.Parameters.AddWithValue("@authId", txt_autherId.Text);
            cmd.Parameters.AddWithValue("@authname", txt_name.Text);
            cmd.Parameters.AddWithValue("@phoneno", txt_phoneno.Text);
            cmd.ExecuteNonQuery();
            lbl_message.Text = "Record is added successfully";
            txt_autherId.Text = "";
            txt_name.Text = "";
```

```
            txt_phoneno.Text = "";
            con.Close();
            updateDropDownList();
        }
        catch (Exception ex)
        {
            lbl_message.Text = ex.Message;
        }
    }

    protected void btn_delete_Click(object sender, EventArgs e)
    {
        try
        {
            string conStr =
WebConfigurationManager.ConnectionStrings["ConnectionString"].ConnectionString;
            SqlConnection con = new SqlConnection(conStr);
            con.Open();
            SqlCommand cmd = new SqlCommand("delete from authers where
authId=@authId", con);
            cmd.Parameters.AddWithValue("@authId", DropDownList1.SelectedItem.Text);
            cmd.ExecuteNonQuery();
            con.Close();
            lbl_delete.Text = "record is delete successfully";
            updateDropDownList();
        }
        catch (Exception ex)
        {
            lbl_delete.Text = ex.Message;
        }
    }

    public void updateDropDownList()
    {
        try
        {
            string conStr =
WebConfigurationManager.ConnectionStrings["ConnectionString"].ConnectionString;
            SqlConnection con = new SqlConnection(conStr);
            con.Open();
            SqlCommand cmd = new SqlCommand("select * from authers", con);
            SqlDataReader dr = cmd.ExecuteReader();
            DropDownList1.DataSource = dr;
            DropDownList1.DataTextField = "authId";
            DropDownList1.DataValueField = "authname";
            DropDownList1.DataBind();
            con.Close();
        }
        catch (Exception ex)
        {
            lbl_delete.Text = ex.Message;
        }
    }

    protected void DropDownList1_SelectedIndexChanged(object sender, EventArgs e)
    {
```

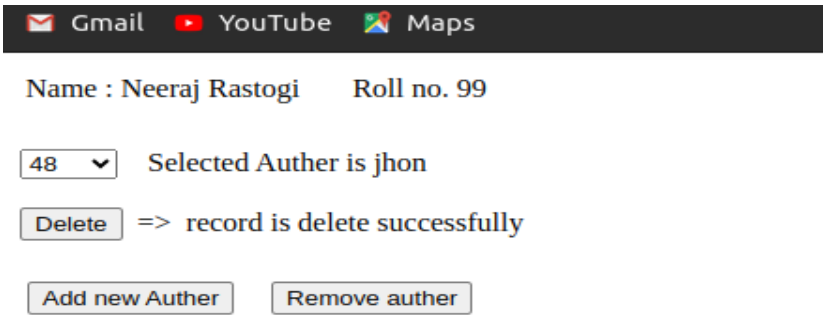
```
        lbl_dropdownname.Text = $"Selected Auther is {DropDownList1.SelectedValue}";
    }
  }
}
```

Output:

Insert Data into the Database .



Remove Data from the Database.



Practical : 8

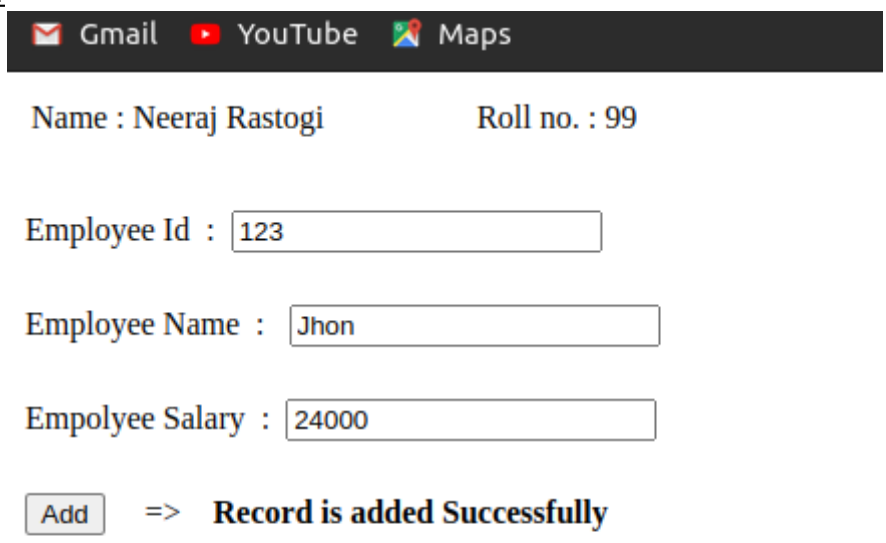
Working with data controls

8.A) Create a web application to demonstrate various uses and properties of SqlDataSource.

Source Code :

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

namespace practical8
{
   public partial class practical9a : System.Web.UI.Page
   {
      protected void Page_Load(object sender, EventArgs e)
      {

      }

      protected void btn_add_Click(object sender, EventArgs e)
      {
         SqlDataSource1.InsertParameters["Eid"].DefaultValue = txt_eid.Text;
         SqlDataSource1.InsertParameters["Ename"].DefaultValue = txt_ename.Text;
         SqlDataSource1.InsertParameters["Esalary"].DefaultValue = txt_esalary.Text;
         SqlDataSource1.Insert();
         lbl_result.Text = "Record is added Successfully";
         txt_eid.Text = "";
         txt_ename.Text = "";
         txt_esalary.Text = "";
      }
   }
}
```

Output:



8.B) Create a web application to demonstrate data binding using DetailsView and FormView Control.

FormView:

```
<asp:FormView ID="FormView1" runat="server" AllowPaging="True"
DataKeyNames="EId" DataSourceID="SqlDataSource1" Height="198px" Width="274px">
                            <EditItemTemplate>
                                EId:
                                <asp:Label ID="EIdLabel1" runat="server"
Text='<%# Eval("EId") %>' />
                                <br />
                                Ename:
                                <asp:TextBox ID="EnameTextBox" runat="server"
Text='<%# Bind("Ename") %>' />
                                <br />
                                Esalary:
                                <asp:TextBox ID="EsalaryTextBox" runat="server"
Text='<%# Bind("Esalary") %>' />
                                <br />
                                <asp:LinkButton ID="UpdateButton"
runat="server" CausesValidation="True" CommandName="Update" Text="Update" />
                                 <asp:LinkButton ID="UpdateCancelButton"
runat="server" CausesValidation="False" CommandName="Cancel" Text="Cancel" />
                            </EditItemTemplate>
                            <HeaderTemplate>
                                My empolyee information
                            </HeaderTemplate>
                            <InsertItemTemplate>
                                EId:
                                <asp:TextBox ID="EIdTextBox" runat="server"
Text='<%# Bind("EId") %>' />
                                <br />
                                Ename:
                                <asp:TextBox ID="EnameTextBox" runat="server"
Text='<%# Bind("Ename") %>' />
                                <br />
                                Esalary:
                                <asp:TextBox ID="EsalaryTextBox" runat="server"
Text='<%# Bind("Esalary") %>' />
                                <br />
                                <asp:LinkButton ID="InsertButton"
runat="server" CausesValidation="True" CommandName="Insert" Text="Insert" />
                                 <asp:LinkButton ID="InsertCancelButton"
runat="server" CausesValidation="False" CommandName="Cancel" Text="Cancel" />
                            </InsertItemTemplate>
                            <ItemTemplate>
                                EId:
                                <asp:Label ID="EIdLabel" runat="server" Text='<
%# Eval("EId") %>' />
                                <br />
                                Ename:
                                <asp:Label ID="EnameLabel" runat="server"
Text='<%# Bind("Ename") %>' />
                                <br />
                                Esalary:
                                <asp:Label ID="EsalaryLabel" runat="server"
Text='<%# Bind("Esalary") %>' />
                                <br />
```

```
                                <asp:LinkButton ID="EditButton" runat="server"
CausesValidation="False" CommandName="Edit" Text="Edit" />
                                 <asp:LinkButton ID="DeleteButton"
runat="server" CausesValidation="False" CommandName="Delete" Text="Delete" />
                                 <asp:LinkButton ID="NewButton"
runat="server" CausesValidation="False" CommandName="New" Text="New" />
                            </ItemTemplate>
                        </asp:FormView>
```

Name : Neeraj Rastogi Roll no. : 99

My empolyee information

EId: 1
Ename: user1
Esalary: 18000
Edit Delete New

1 2 3 4 5 6 7 8 9 10

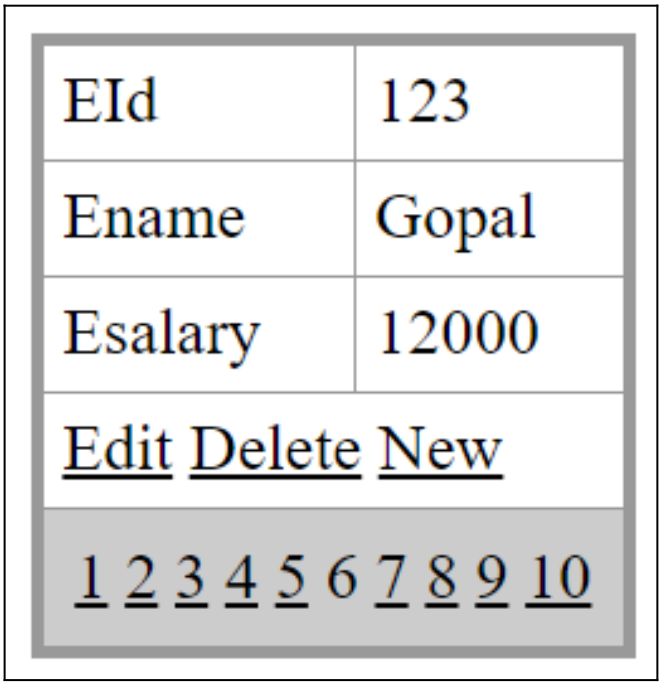
EId	1
Ename	user1
Esalary	18000
Edit Delete New	
1 2 3 4 5 6 7 8 9 10	

Details View:

```
<asp:DetailsView ID="DetailsView1" runat="server" AllowPaging="True"
AutoGenerateRows="False" BackColor="#CCCCCC" BorderColor="#999999"
BorderStyle="Solid" BorderWidth="3px" CellPadding="4" CellSpacing="2"
DataKeyNames="EId" DataSourceID="SqlDataSource1" ForeColor="Black"
Height="50px" Width="125px">
                <EditRowStyle BackColor="#000099" Font-Bold="True"
ForeColor="White" />
                <Fields>
```

```
                    <asp:BoundField DataField="EId" HeaderText="EId"
ReadOnly="True" SortExpression="EId" />
                    <asp:BoundField DataField="Ename" HeaderText="Ename"
SortExpression="Ename" />
                    <asp:BoundField DataField="Esalary" HeaderText="Esalary"
SortExpression="Esalary" />
                    <asp:CommandField ShowDeleteButton="True"
ShowEditButton="True" ShowInsertButton="True" />
                </Fields>
                <FooterStyle BackColor="#CCCCCC" />
                <HeaderStyle BackColor="Black" Font-Bold="True"
ForeColor="White" />
                <PagerStyle BackColor="#CCCCCC" ForeColor="Black"
HorizontalAlign="Left" />
                <RowStyle BackColor="White" />
            </asp:DetailsView>
```

Output:

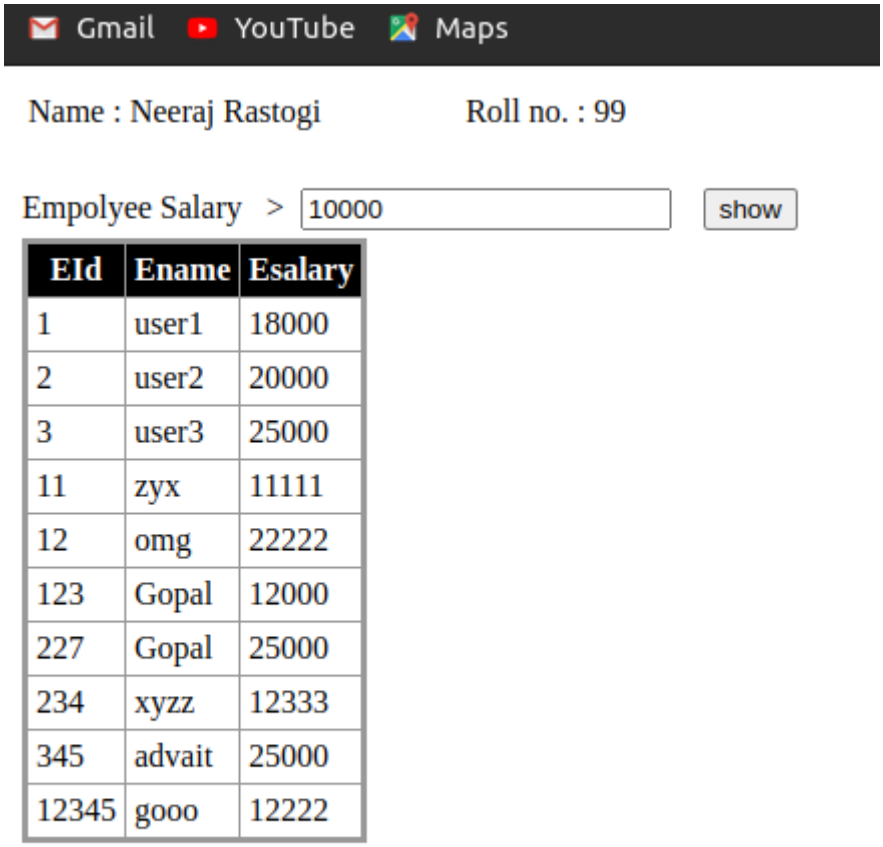


8.C) Create a web application to display Using Disconnected Data Access and Databinding using GridView.

Source Code :

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data;
using System.Data.SqlClient;
using System.Web.Configuration;

namespace practical8
{
   public partial class practical8c : System.Web.UI.Page
   {
      protected void Page_Load(object sender, EventArgs e)
      {

      }

      protected void btn_show_Click(object sender, EventArgs e)
      {
         try
         {
            String constr =
WebConfigurationManager.ConnectionStrings["ConnectionString"].ConnectionString;
            SqlConnection con = new SqlConnection(constr);
            con.Open();
            String sql = "select * from emp where Esalary >" + txt_getdata.Text;
            SqlDataAdapter da = new SqlDataAdapter(sql, con);
            DataSet ds = new DataSet();
            da.Fill(ds,"emp");
            GridView1.DataSource = ds;
            GridView1.DataBind();
            con.Close();
         }
         catch (Exception ex)
         {
            Response.Write(ex);
         }
      }
   }
}
```

Output:


Gmail
YouTube
Maps
Name : Neeraj Rastogi
Roll no. : 99
Empolyee Salary > 10000
show
EId Ename Esalary
1 user1 18000
2 user2 20000
3 user3 25000
11 zyx 11111
12 omg 22222
123 Gopal 12000
227 Gopal 25000
234 xyzz 12333
345 advait 25000
12345 gooo 12222

Practical : 9

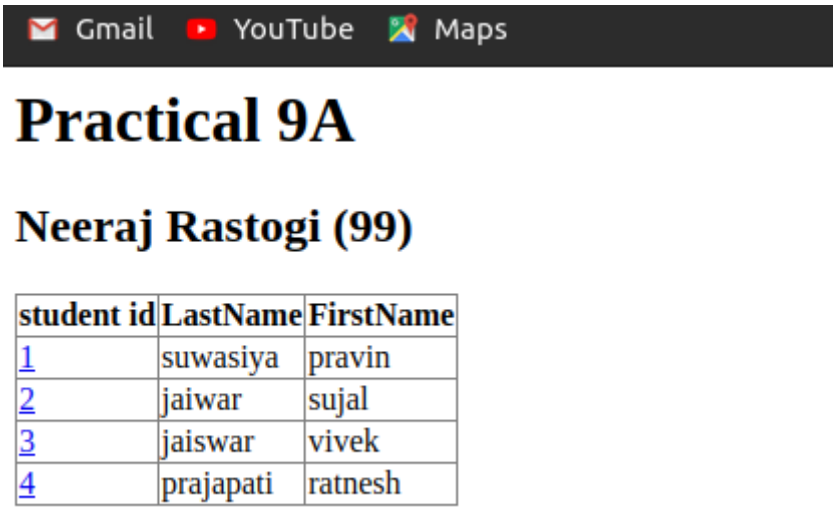
Working with GridView control

9.A) Create a web application to demonstrate use of GridView control template and GridView hyperlink.

Source Code :

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="WebForm2.aspx.cs"
Inherits="_9A.WebForm2" %>

<!DOCTYPE html>

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

      <h1>Practical 9A</h1>
      <h2>Pravin Suwasiya (118)</h2>
      <div>
         <asp:GridView ID="GridView1" runat="server" AutoGenerateColumns="False"
DataSourceID="SqlDataSource1">
            <Columns>
               <asp:HyperLinkField DataNavigateUrlFields="sid"
DataNavigateUrlFormatString="~/WebForm3.aspx?sid={0}" DataTextField="sid"
DataTextFormatString="{0}" HeaderText="student id" NavigateUrl="~/WebForm3.aspx"
/>
               <asp:BoundField DataField="LastName" HeaderText="LastName"
SortExpression="LastName" />
               <asp:BoundField DataField="FirstName" HeaderText="FirstName"
SortExpression="FirstName" />
            </Columns>
            <EmptyDataTemplate>
               no data found
            </EmptyDataTemplate>
         </asp:GridView>
         <asp:SqlDataSource ID="SqlDataSource1" runat="server" ConnectionString="<
%$ ConnectionStrings:pravindbConnectionString2 %>" SelectCommand="SELECT *
FROM [students]"></asp:SqlDataSource>
      </div>
   </form>
</body>
</html>
```

Output:



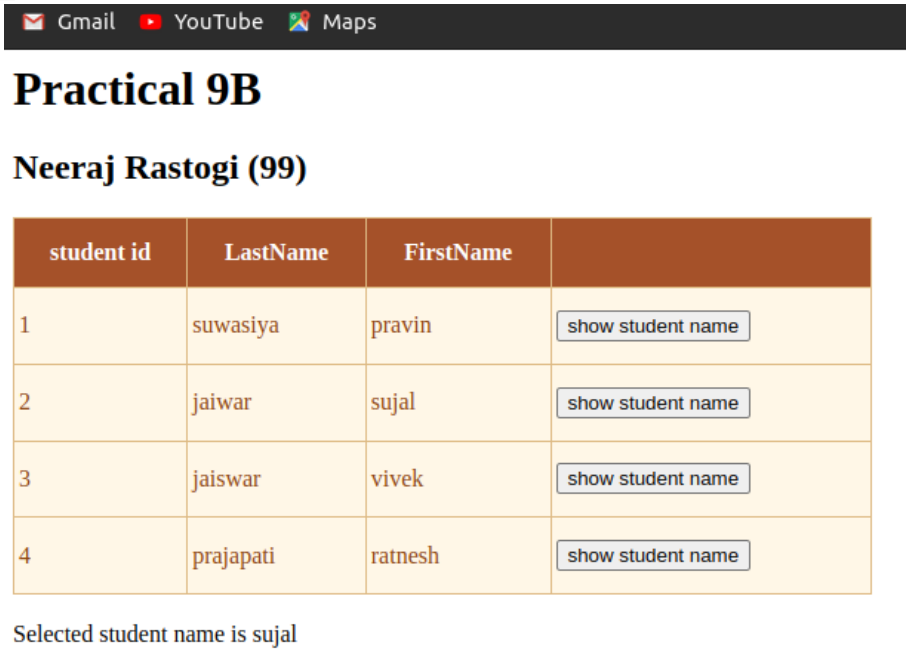
sid	1
LastName	suwasiya
FirstName	pravin

9.B) Create a web application to demonstrate use of GridView button column and GridView events.

Source Code :

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

namespace _9B
{
   public partial class WebForm1 : System.Web.UI.Page
   {
      protected void Page_Load(object sender, EventArgs e)
      {

      }
      protected void GridView1_RowCommand(object sender,
GridViewCommandEventArgs e)
      {
         if (e.CommandName == "Show")
         {
            int index = int.Parse(e.CommandArgument.ToString());
            GridViewRow row = GridView1.Rows[index];
            String name = row.Cells[2].Text;
            Label1.Text = "Selected student name is " + name;
         }
      }
   }
}
```

Output:



9.C) Create a web application to demonstrate GridView paging and Creating own table format using GridView.

Source Code :

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="WebForm1.aspx.cs"
Inherits="_9C.WebForm1" %>
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
</head>
<body>
    <form id="form1" runat="server">
        <h1>Practical 9B</h1>
        <h2>Pravin Suwasiya (118)</h2>
        <div>
            <asp:GridView ID="GridView1" runat="server" AllowPaging="True"
AutoGenerateColumns="False" BackColor="White" BorderColor="#CCCCCC"
BorderStyle="None" BorderWidth="1px" CellPadding="4"
DataSourceID="SqlDataSource1" ForeColor="Black" GridLines="Horizontal"
Height="206px" PageSize="2" Width="434px">
                <Columns>
                    <asp:BoundField DataField="sid" HeaderText="sid" SortExpression="sid" />
                    <asp:BoundField DataField="LastName" HeaderText="LastName"
SortExpression="LastName" />
                    <asp:BoundField DataField="FirstName" HeaderText="FirstName"
SortExpression="FirstName" />
                </Columns>
                <FooterStyle BackColor="#CCCC99" ForeColor="Black" />
                <HeaderStyle BackColor="#333333" Font-Bold="True" ForeColor="White" />
                <PagerStyle BackColor="White" ForeColor="Black" HorizontalAlign="Right" />
                <SelectedRowStyle BackColor="#CC3333" Font-Bold="True"
ForeColor="White" />
                <SortedAscendingCellStyle BackColor="#F7F7F7" />
                <SortedAscendingHeaderStyle BackColor="#4B4B4B" />
                <SortedDescendingCellStyle BackColor="#E5E5E5" />
                <SortedDescendingHeaderStyle BackColor="#242121" />
            </asp:GridView>
            <asp:SqlDataSource ID="SqlDataSource1" runat="server" ConnectionString="<
%$ ConnectionStrings:pravindbConnectionString %>" SelectCommand="SELECT * FROM
[students]"></asp:SqlDataSource>
        </div>
    </form>
</body>
</html>
```

Output:

Gmail YouTube Maps

Practical 9C

Neeraj Rastogi (99)

sid	LastName	FirstName
1	suwasiya	pravin
2	jaiwar	sujal

1 2

Practical : 10

Working with AJAX and XML

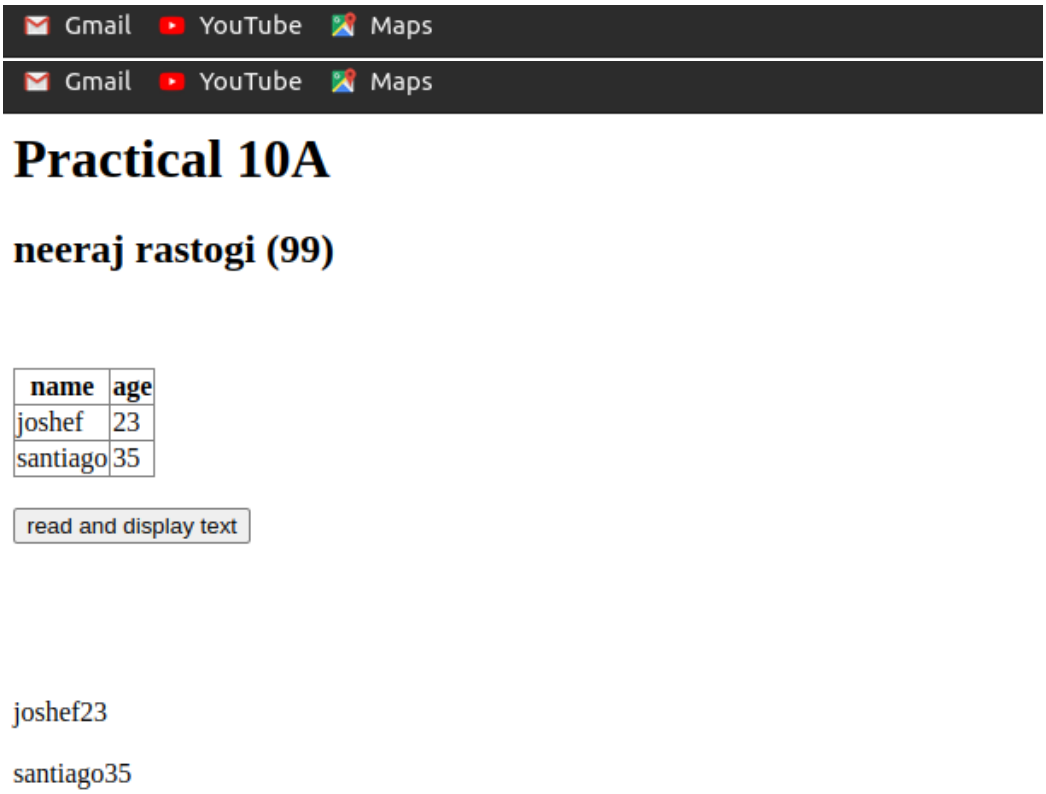
10.A) Create a web application to demonstrate reading and writing operation with XML.

Source Code :

```
using System;

namespace _10A
{
   public partial class WebForm1 : System.Web.UI.Page
   {
      protected void Page_Load(object sender, EventArgs e)
      {

      }

      protected void Button1_Click(object sender, EventArgs e)
      {
         FileStream fs = new FileStream("d:\\pravin.xml",FileMode.Create);
         XmlTextWriter writer = new XmlTextWriter(fs, null);
         writer.Formatting = Formatting.Indented;
         writer.WriteStartDocument();
         writer.WriteStartElement("students");

         writer.WriteStartElement("student");
         writer.WriteAttributeString("name", "user1");
         writer.WriteAttributeString("age", "27");
         writer.WriteEndElement();

         writer.WriteStartElement("student");
         writer.WriteAttributeString("name", "user2");
         writer.WriteAttributeString("age", "28");
         writer.WriteEndElement();

         writer.WriteStartElement("student");
         writer.WriteAttributeString("name", "user3");
         writer.WriteAttributeString("age", "21");
         writer.WriteEndElement();

         writer.WriteStartElement("student");
         writer.WriteAttributeString("name", "user4");
         writer.WriteAttributeString("age", "20");
         writer.WriteEndElement();

         writer.WriteStartElement("student");
         writer.WriteAttributeString("name", "user5");
         writer.WriteAttributeString("age", "26");
         writer.WriteEndElement();
```

```
            writer.WriteEndElement();
            writer.WriteEndDocument();
            writer.Close();
            fs.Close();
        }
    }
```

Output:



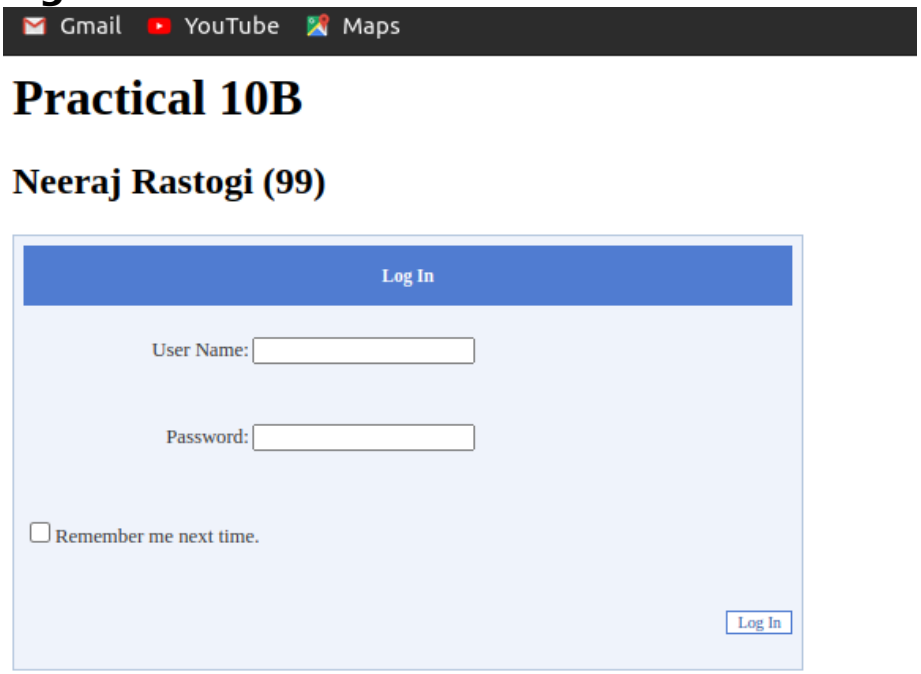
10.B) Create a web application to demonstrate Form Security and Windows Security with proper Authentication and Authorization properties..

Source Code :

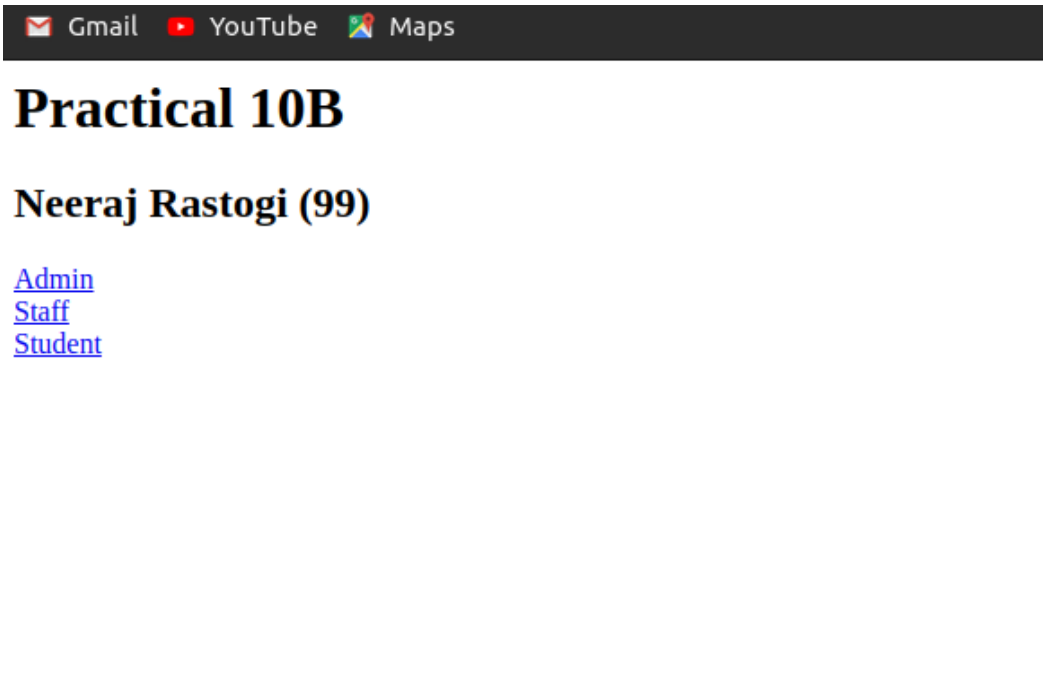
```
using System;
namespace _10B
{
   public partial class WebForm2 : System.Web.UI.Page
   {
      protected void Page_Load(object sender, EventArgs e)
      {

      }
      protected void Button1_Click(object sender, EventArgs e)
      {
         XmlDataSource ds = new XmlDataSource();
         ds.DataFile = "d:\\pravin.xml";
         GridView1.DataSource = ds;
         GridView1.DataBind();
         Label1.Text = "";
         XmlTextReader reader = new XmlTextReader("d:\\pravin.xml");
         while (reader.Read())
         {
            Label1.Text += reader.GetAttribute("name");
            Label1.Text += reader.GetAttribute("age");
            Label1.Text += "<br>";

         }
         reader.Close();
      }
   }
}
```

Output:

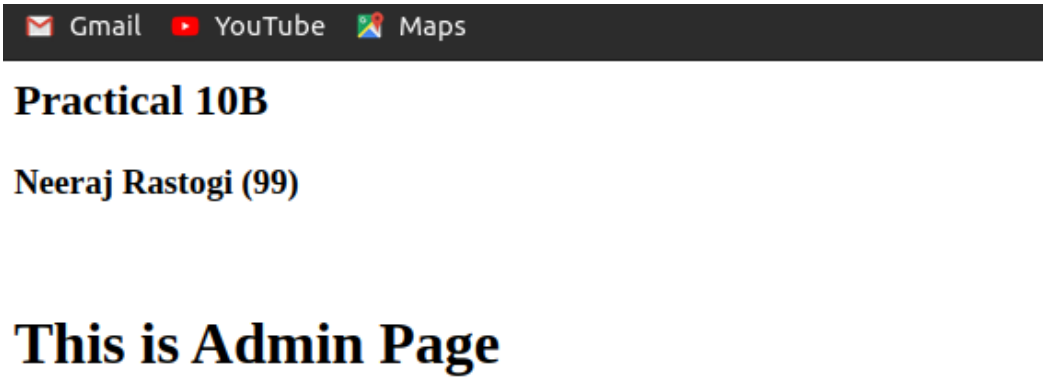
Login Page :



Home Page :



Admin Page :



10.C) Create a web application to demonstrate use of various Ajax controls.

Source Code :

```
using System;

namespace _10C
{
   public partial class WebForm1 : System.Web.UI.Page
   {
      protected void Page_Load(object sender, EventArgs e)
      {

      }
      protected void Button1_Click(object sender, EventArgs e)
      {
         System.Threading.Thread.Sleep(4000);

         Label1.Text = TextBox1.Text + TextBox2.Text;

      }

      protected void Button2_Click(object sender, EventArgs e)
      {
         System.Threading.Thread.Sleep(4000);
         Label2.Text = TextBox3.Text + TextBox4.Text;

      }
   }
}
```

Output: