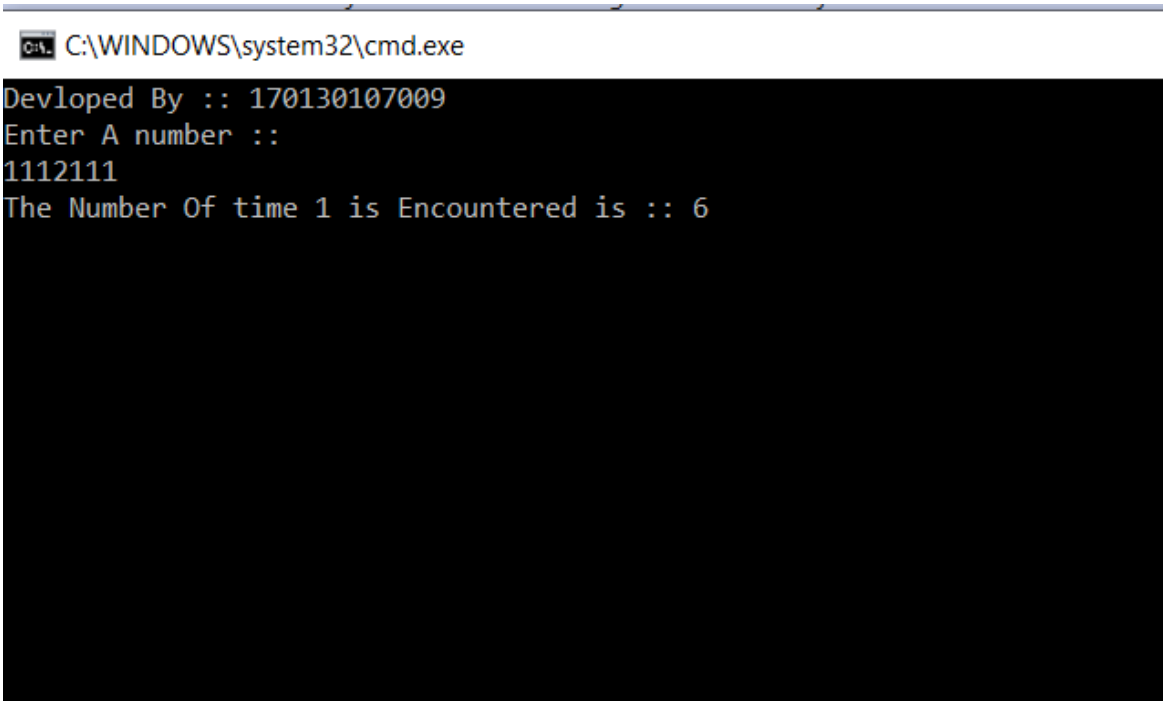
Practical 1 :

AIM :: Create C#.net console application to count the number of 1's in the entered number. (as an input only integer numbers are allowed)

```
using System;


namespace Practicle_
{
   class Class1
   {
      static int result = 0;
      string developer = "170130107009";
      static void Main(String []args)
      {
         Console.WriteLine("Enter A number :: ");
         int n = Convert.ToInt32(Console.ReadLine());
         int x;
         while(n>0)
         {
            x = n % 10;
            checkNumber(x);
            n = n / 10;
```

```
        }

        Console.WriteLine("The Number Of time 1 is Encountered is :: {0}",result);
        Console.ReadKey();

    }
    static void checkNumber(int n)
    {
        if(n==1)
        {
            result++;
        }
    }
  }
}
```

Output ::


C:\WINDOWS\system32\cmd.exe
Devloped By :: 170130107009
Enter A number ::
1112111
The Number Of time 1 is Encountered is :: 6

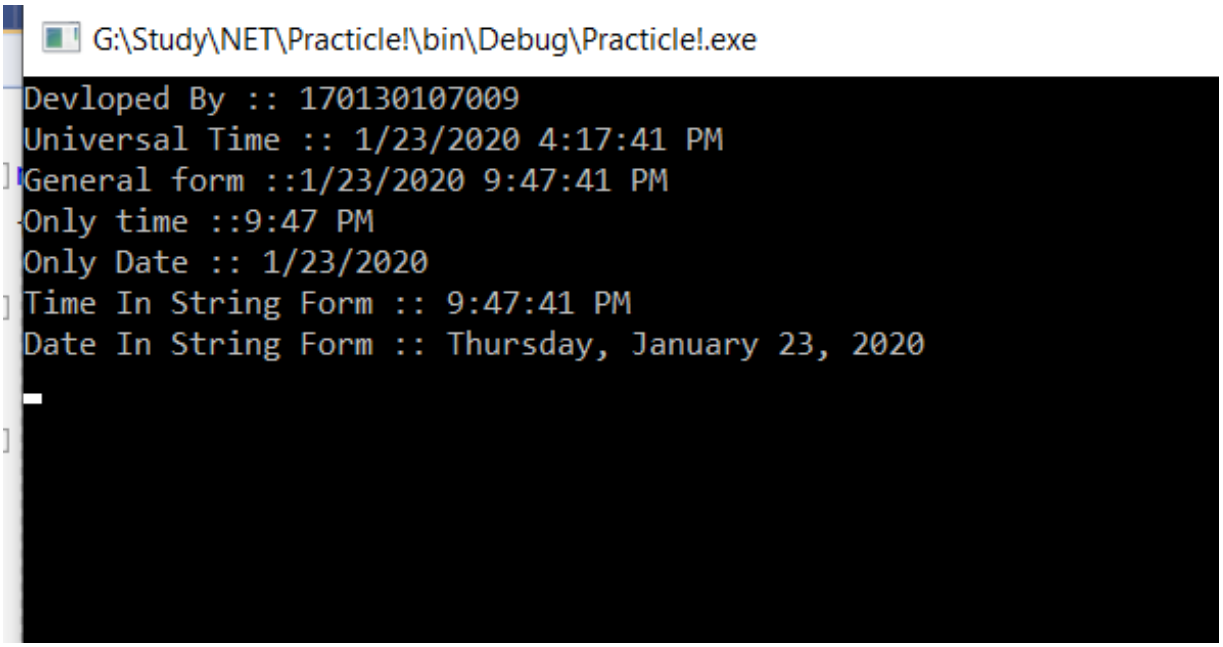
Practical 2 :

AIM :: Create C#.net console application to display the date in various formats using DateTime class.

```
using System;
namespace Practicle_
{
   class Practicle2
   {
      static void Main(String []args)
      {
         Console.WriteLine("Devloped By :: 170130107009");
         DateTime dt = DateTime.Now;

         string dates = dt.ToShortDateString();

         string times = dt.ToShortTimeString();
         string timel = dt.ToLongTimeString();
         string datel = dt.ToLongDateString();
         Console.WriteLine("Universal Time :: "+dt.ToUniversalTime());
         Console.WriteLine("General form ::"+dt);
         Console.WriteLine("Only time ::" + times);
```

```
        Console.WriteLine("Only Date :: "+dates);
        Console.WriteLine("Time In String Form :: "+timel);
        Console.WriteLine("Date In String Form :: " + datel);
        Console.ReadKey();
    }
  }
}
```

Output ::



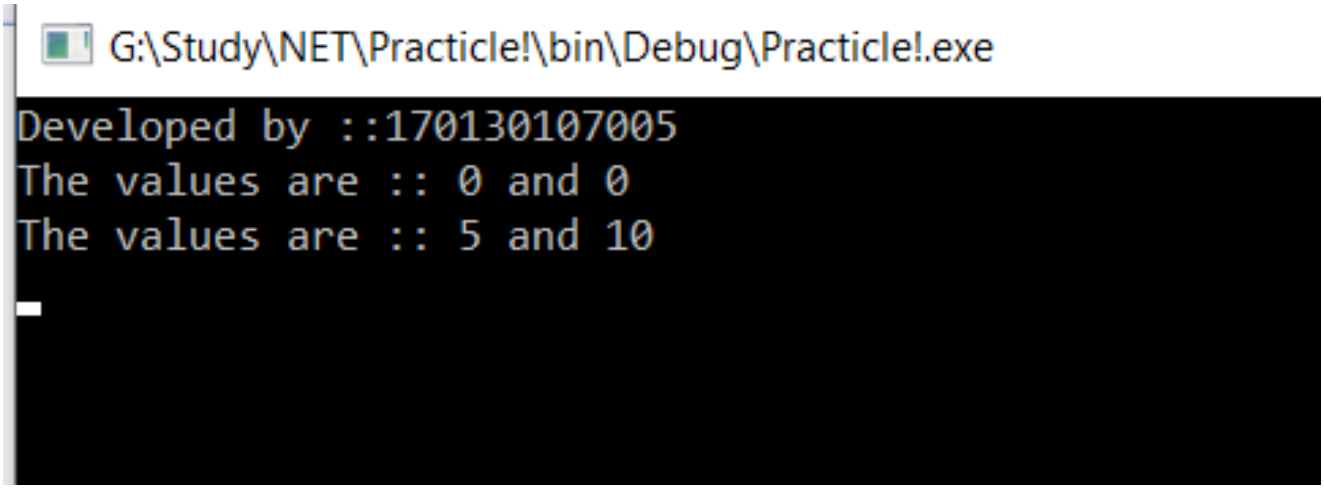
Practical 3 :

AIM :: Create C#.Net console application which demonstrates the usage of constructor and destructor.

```
using System;

namespace Practicle_

{

   class Demo

   {

      public int x, y;


      public Demo()

      {

         this.x = 0;

         this.y = 0;

      }

      public Demo(int x,int y)

      {

         this.x = x;

         this.y = y;

      }


      public void Display()
```

```
    {
        Console.WriteLine("The values are :: "+x+" and "+y);
    }

    ~Demo()
    {
        Console.WriteLine("The Destructor is Called..");
    }
  }
  class Practicle3
  {
    static void Main(String []args)
    {
        Demo d1 = new Demo();
        Demo d2 = new Demo(5, 10);
        Console.WriteLine("Developed by ::170130107009");
        d1.Display();
        d2.Display();
        Console.ReadLine();
    }
  }
 }
```

Output ::


G:\Study\NET\Practicle!\bin\Debug\Practicle!.exe
Developed by ::170130107005
The values are :: 0 and 0
The values are :: 5 and 10

Practical 4 :

AIM :: Create C#.Net console application which demonstrates the usage of inheritance in C#.Net.

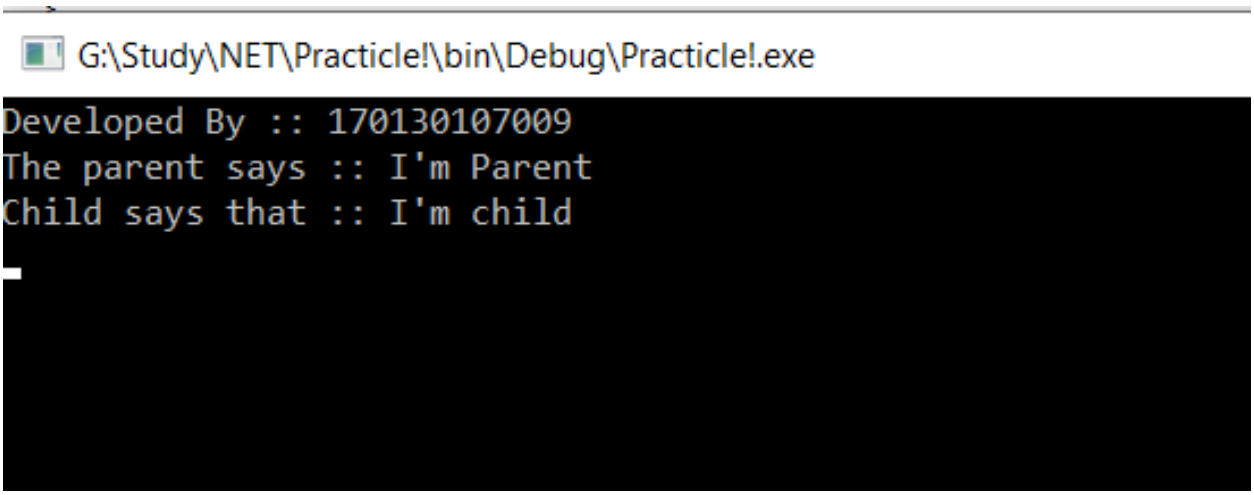
```
using System;



namespace Practicle_
{
   class Parent
   {
      private string msg;

      public Parent(string msg)
      {
         this.msg = msg;
      }

      public string Msg
      {
         get
         {
            return msg;
```

```
        }
        set
        {
            msg = value;
        }
    }

    public void show_parent()
    {
        Console.WriteLine("The parent says :: "+msg);
    }
}
class Practicle4 : Parent
{
    private string msg1;
    Practicle4(string msg1,string msg):base(msg)
    {
        this.msg1 = msg1;
    }

    public string Msg1
    {
```

```
        get
        {
            return msg1;
        }
        set
        {
            this.msg1 = value;
        }
    }
    public void show_child()
    {
        Console.WriteLine("Child says that :: "+msg1);
    }

    static void Main(string []args)
    {
        Practicle4 obj = new Practicle4("I'm child", "I'm Parent");

        Console.WriteLine("Developed By :: 170130107009");
        obj.show_parent();
        obj.show_child();
```

```
            Console.ReadLine();
        }
    }
}
```

Output ::



Practical 5 :

AIM :: Create C#.Net application which demonstrates the usage of properties and indexer.

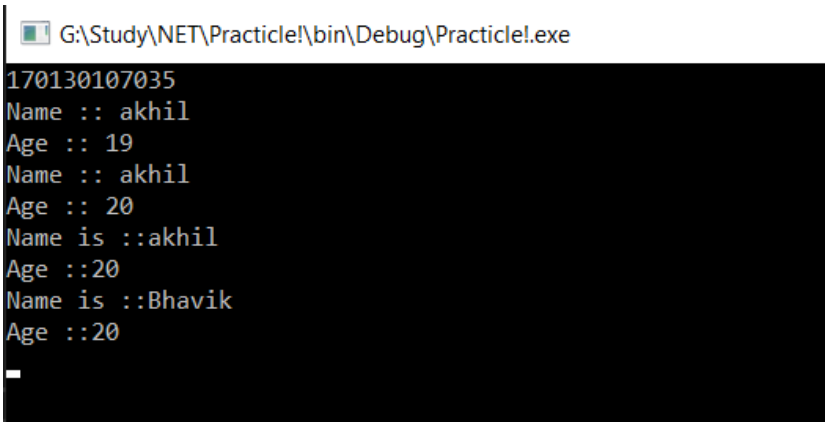
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;


namespace Practicle_
{
   class practical5
   {
      string name;
      int no;
      string city;
      string mobile;
      public practical5(string name,int no,string city,string mobile)
      {
         this.name = name;
         this.no = no;
         this.city = city;
```

```
        this.mobile = mobile;
    }
    public string Name
    {
        get { return name; }
        set { name = value; }
    }
    public int No
    {
        get { return no; }
        set { no = value; }
    }
    public object this[int index]
    {
        get
        {
            if (index == 1)
            { return name; }
            else if (index == 2)
            {
                return no;
            }
```

```
            return null;
        }

        set
        {
            if (index == 1)
            { name = (string)value; }
            else if (index == 2)
            {
                no = (int)value;
            }
        }
    }
    static void Main(string []args)
    {
        practical5 p = new practical5("akhil",19,"Una","170130107035");
        Console.WriteLine("170130107035");
        Console.WriteLine("Name :: " + p.Name);
        Console.WriteLine("Age :: " + p.No);
        p.No = 20;
```

```
        Console.WriteLine("Name :: " + p.Name);
        Console.WriteLine("Age :: " + p.No);

        Console.WriteLine("Name is ::" + p[1]);
        Console.WriteLine("Age ::" + p[2]);

        p[1] ="Bhavik";
        Console.WriteLine("Name is ::" + p[1]);
        Console.WriteLine("Age ::" + p[2]);

        Console.ReadLine();
    }
  }
}
```

Output ::



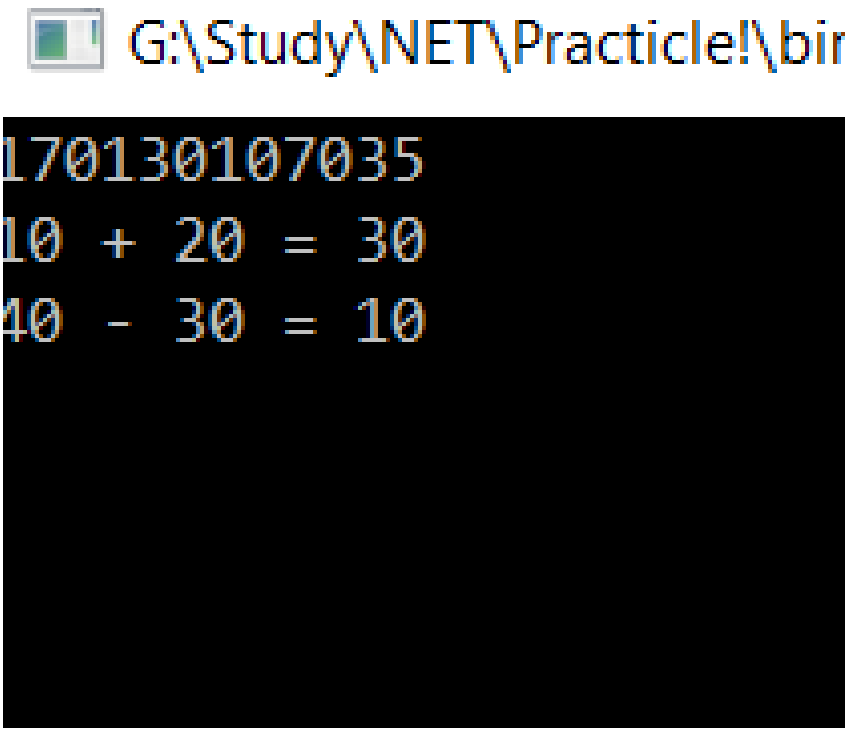
Practical 6 :

AIM :: Create C#.Net console application which demonstrates the usage of Delegates in C#.Net.

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


namespace Practicle_
{
   class Practical6
   {
      int x, y;
      public delegate void addnum(int a, int b);
      public delegate void subnum(int a, int b);
      public Practical6(int x,int y)
      {
         this.x = x;
         this.y = y;


      }
```

```
    public void sum(int a,int b)
    {
       Console.WriteLine("{0} + {1} = {2}", a, b, a + b);
    }
    public void sub(int a, int b)
    {
       Console.WriteLine("{0} - {1} = {2}", a, b, a - b);
    }
    static void Main(string []args)
    {
       Practical6 obj = new Practical6(5, 7);
       Console.WriteLine("170130107035");
       addnum del_obj1 = new addnum(obj.sum);
       subnum del_obj2 = new subnum(obj.sub);

       del_obj1(10, 20);
       del_obj2(40, 30);
       Console.ReadLine();
    }
  }
}
```

Output ::



Practical 7 :

AIM :: Create a web application that will make connection with SQL Server express and perform operations of addition, updating and deletion of data on Login Form.

```
using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.SqlClient;


namespace Practical8._1

{

   public partial class Form1 : Form

   {

      SqlConnection con = new SqlConnection(@"Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\user\Documents\
demo.mdf;Integrated Security=True;Connect Timeout=30");

      public Form1()
```

```
    {
        InitializeComponent();
    }

    private void Form1_Load(object sender, EventArgs e)
    {
        display();

    }

    private void Label1_Click(object sender, EventArgs e)
    {

    }
    private void Button1_Click(object sender, EventArgs e)
    {
        con.Open();
        SqlCommand cmd = con.CreateCommand();
        cmd.CommandType = CommandType.Text;
        cmd.CommandText = "insert into log
values('"+textBox1.Text+"','"+textBox2.Text+"')";
        cmd.ExecuteNonQuery();
```

```
        con.Close();
        MessageBox.Show("Data Inserted Successfully");
        display();
        textBox1.Text = "";
        textBox2.Text = "";



    }
    public void display()
    {
        con.Open();
        SqlCommand cmd = con.CreateCommand();
        cmd.CommandType = CommandType.Text;
        cmd.CommandText = "select * from log";
        DataTable dt = new DataTable();
        SqlDataAdapter da = new SqlDataAdapter(cmd);
        da.Fill(dt);
        dataGridView1.DataSource = dt;
        cmd.ExecuteNonQuery();

        con.Close();
```

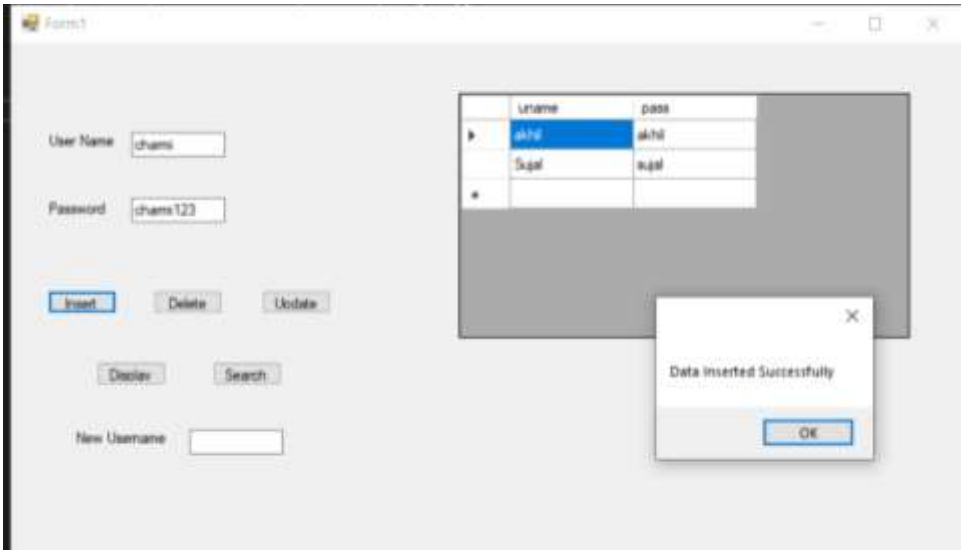
```
}
private void Button2_Click(object sender, EventArgs e)
{
    con.Open();
   SqlCommand cmd = con.CreateCommand();
   cmd.CommandType = CommandType.Text;
   cmd.CommandText = "delete from log where uname='"+textBox1.Text+"'";
   cmd.ExecuteNonQuery();

   con.Close();
   MessageBox.Show("Data deleted Successfully");
   display();
   textBox1.Text = "";

}

private void Button3_Click(object sender, EventArgs e)
{
   con.Open();
   SqlCommand cmd = con.CreateCommand();
   cmd.CommandType = CommandType.Text;
```

```
        cmd.CommandText = "update log set uname='"+textBox3.Text+"' where
uname='"+textBox1.Text+"'";
        cmd.ExecuteNonQuery();

        con.Close();
        MessageBox.Show("Data updated Successfully");
        display();
        textBox1.Text = "";
        textBox3.Text = "";
    }

    private void Button4_Click(object sender, EventArgs e)
    {

    }

    private void Button5_Click(object sender, EventArgs e)
    {
        con.Open();
        SqlCommand cmd = con.CreateCommand();
        cmd.CommandType = CommandType.Text;
```

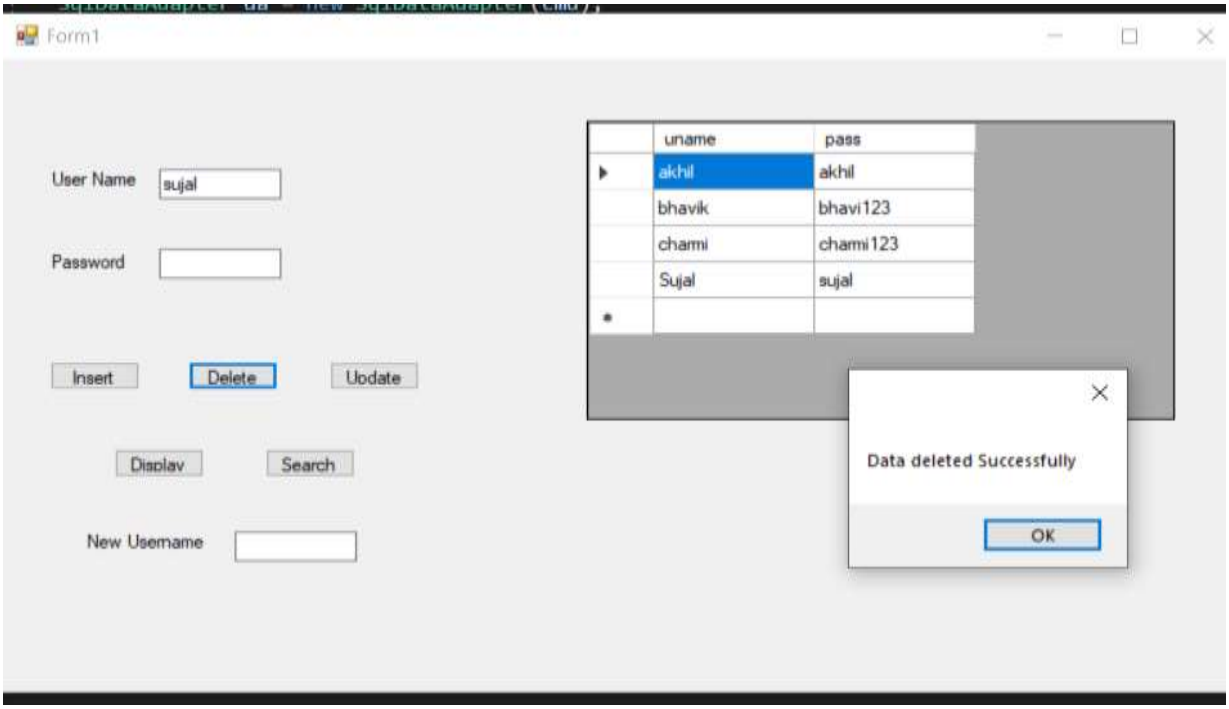
```
        cmd.CommandText = "select * from log where
uname='"+textBox1.Text+"'";

        DataTable dt = new DataTable();

        SqlDataAdapter da = new SqlDataAdapter(cmd);

        da.Fill(dt);

        dataGridView1.DataSource = dt;

        cmd.ExecuteNonQuery();


        con.Close();

        textBox1.Text = "";

      }

   }

}
```

Output ::

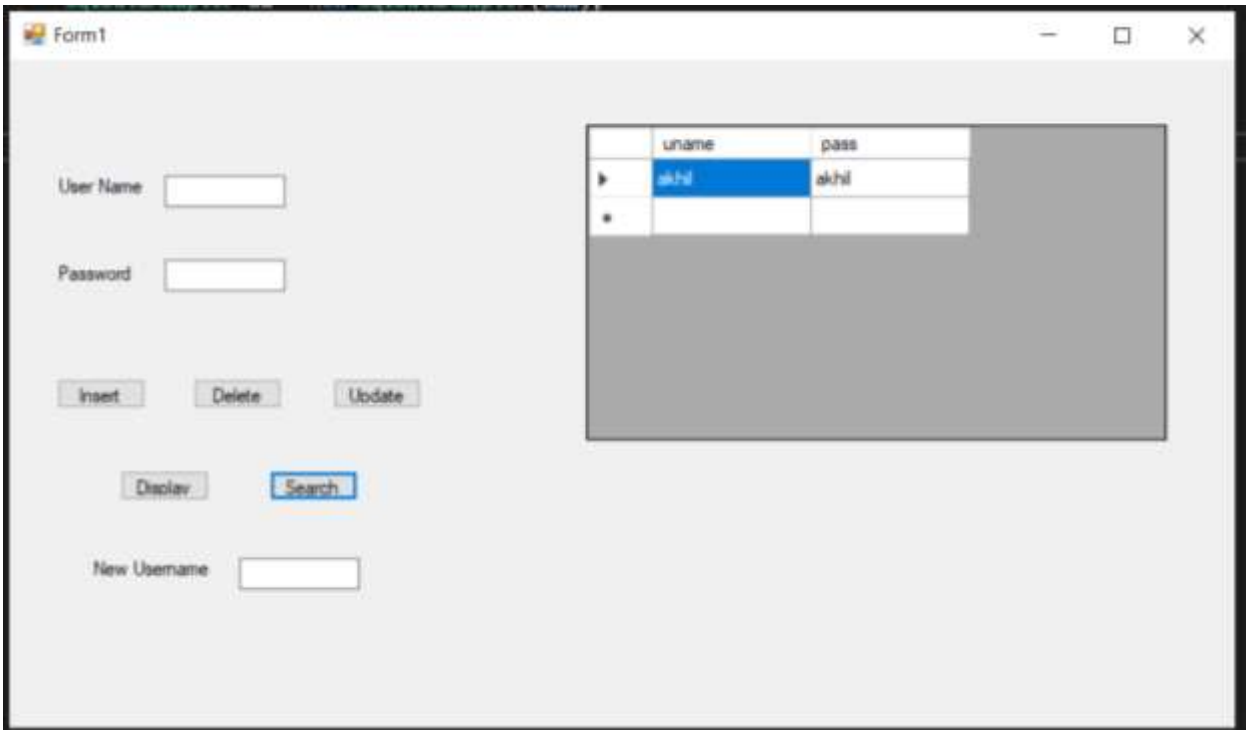
***Onclick_Insert**



***Onclick_delete**



***Onclick_Search**



Practical 8 :

AIM :: Create a webpage to bind the user data from database into a GridView dynamically.

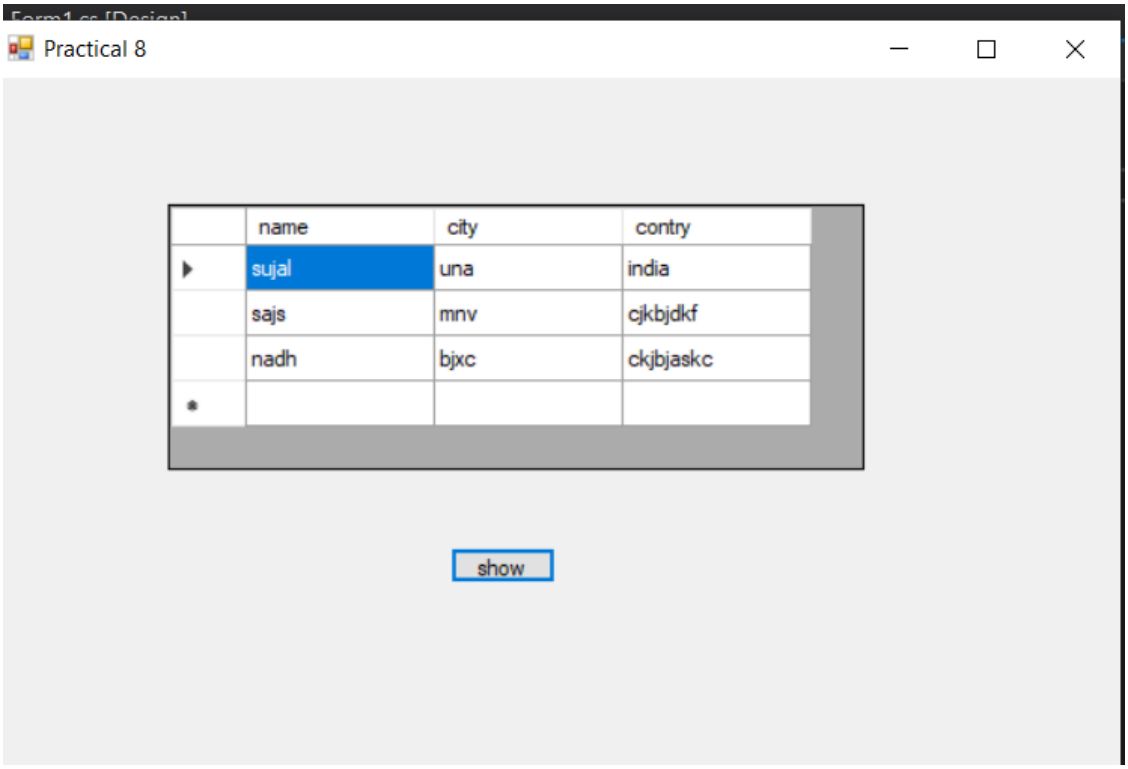
```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Data.SqlClient;


namespace practical8
{
   public partial class Form1 : Form
   {
      SqlConnection con = new SqlConnection(@"Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\user\Documents\
demo.mdf;Integrated Security=True;Connect Timeout=30");
      public Form1()
      {
```

```
        InitializeComponent();
    }

    private void DataGridView1_CellContentClick(object sender,
DataGridViewCellEventArgs e)
    {

    }
    private void Button1_Click(object sender, EventArgs e)
    {
        con.Open();
        SqlCommand cmd = con.CreateCommand();
        cmd.CommandType = CommandType.Text;
        cmd.CommandText = "select *from person";
        cmd.ExecuteNonQuery();
        DataTable dt = new DataTable();
        SqlDataAdapter da = new SqlDataAdapter(cmd);
        da.Fill(dt);
        dataGridView1.DataSource = dt;
        con.Close();
```

```
        }
    }
}
```

Output ::

*Onclick_Search



Practical 9 :

AIM :: To study about different windows control available in .net framework 3.5.

1) **Button** : fire a command when a mouse click occur or the enter.

Properties:

1. **BackColor:** Gets or sets the background color of the button control.
2. **Image :** Gets or sets the image that is displayed on a button control.

Events:

1. **Click :** Occurs when user clicks the Button.
2. **Validated :** Occurs when the button control is finished validating.
3. **GotFocus:** Occurs when the button control receives focus.
4. **TextChanged:** Occurs when the Text property value changes.
5. **FontChange:** Occurs when font is changed.

2) **TextBox:** Allows to display text and to allow the user to enter information. A textbox control is used to display, or accept input,asingle line of text.

Properties:

1. **TextLength:** Gets the length of text in the TextBoxcontrol.
2. **PasswordChar:** Gets or sets the character used to mask characters of a password in a single-line TextBoxcontrol.
3. **Multiline :** Gets or sets a value indicating whether this is a multiline TextBoxcontrol.
4. **ReadOnly:** Gets or sets a value indicating whether text in the text box is read-only.

Events:

1. **TextChanged:** Occurs when the text in the textbox changes.
2. **TextAlignChanged:** Occurs when the TextAlignproperty value changes.

3) Label: Allows to display text to the user. The label class is defined in the System.window.Formsnamespace.

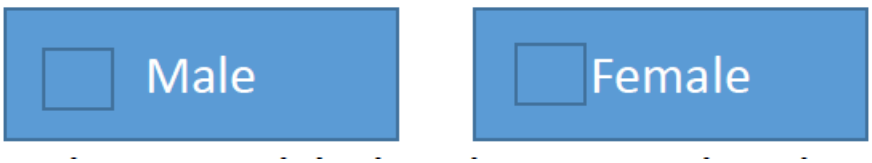
Properties;

1. **TextAlign:** Gets or sets how text is aligned in a Label control.
2. **Visible :** Gets or sets a value indicating whether the control and all its child controls are displayed.
3. **Autosize:** Gets or sets a value specifying if the label control should be automatically resized to display all its contents.

Events:

1. **TextChanged:** Occur when the text property value changed.
2. **Click:** Occur when the user click the label.
3. **Leave:** Occur when the input focus leave the label.
4. **LostFocus:** Occur when the control loses focus.

4) CheckBox: CheckBox gives the option to the user such as true/false or yes/no. allows the user to select multiple option at a time.



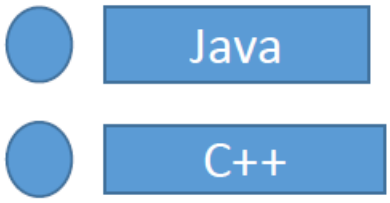
Properties:

1. **CheckAlign:** Gets or Sets the location of the check box portion of the checkbox.
2. **CheckState:** Gets or sets the state of the checkbox.
3. **BackColor:** Gets or sets the background color of the checkbox control.
4. **Image:** Gets or sets the image that is displayed on a checkbox control.

Events:

1. **CheckedChanged:** Occur when the value of the checked property changed.
2. **CheckStateChanged:** occur when the value of Checkstate property changed.

5) RadioButton: Radio button are also called option button. These are similar to checkboxes because the user can select and deselect them. Radion button works in a group.



Properties:

1. **Checked:** Gets or sets a value indicating whether the radio button is checked.
2. **CheckAlign:** Gets or sets the location of the check box portion of the radio button control.
3. **Image:** Gets or sets the image that is displayed on a radio button control.
4. **Font:** Gets or sets the font of the text displayed by radio button control.

Events:

1. **CheckedChanged:** Occurs when the value of the checked property of the radio button control changed.
2. **AppearanceChanged:** Occurs when the value of the Appearance property of the radio button control changed.

6) ListBox: A ListBox control provides a user interface to display list of items.

Users can select one or more items from the list. If the items exceed a specified limit, a scrollbar automatically appears to let the user to scroll through the list.

Properties:

1. **Items:** Gets the items of the listbox.
2. **SelectdItem:** gets or sets the currently selected item in the listbox.
3. **SelectedIndex:** gets or sets the zero-based index of the currently selected item in a listbox.

Events:

1. **SelectedIndexchanged:** Occur when the selectedIndex property has changed.
2. **SizeChanged:** Occur when the size property value changed.

7) ComboBox: A ComboBox control is a combination of the a textbox and a listbox control. Only one list item is displayed at one time in a combobox and othe available items are loaded in a combobox list.

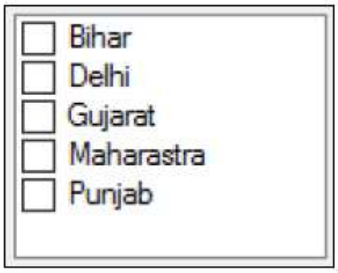
Properties:

1. **Items:** Gets the item of the combobox.
2. **DisplayMember:** gets of sets the property to display for this listcontrol.
3. **Enable:** gets or sets a value indicating whether the the control can respond to user interaction.

Events:

1. **SelectedIndexchanged:** Occurs when the selectedIndex property has changed .
2. **SizeChanged:** occur when the size property value changes.
3. **GotFocus:** occurs when the control receives focus.

8) CheckedListBox: the window forms checkedlistbox control display a list of item, like the listbox control, and also can display a check mark next to items in the list.



Properties:

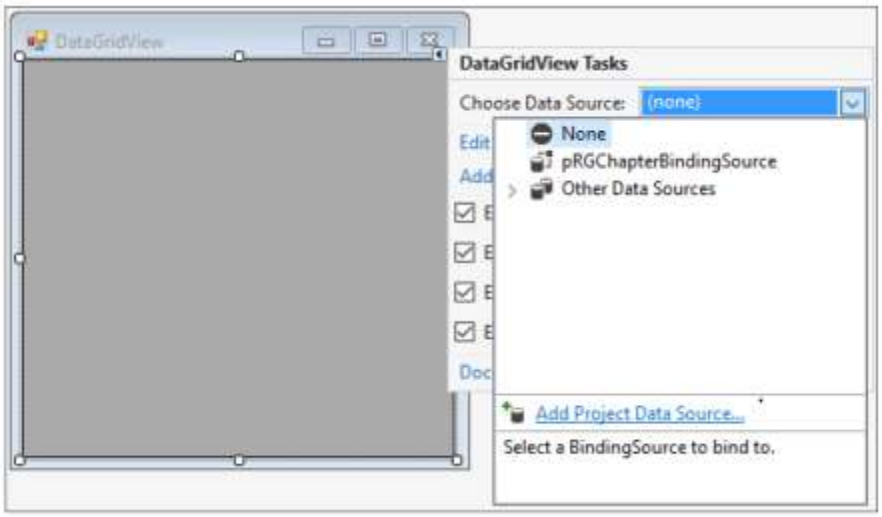
1. **CheckedItems:** Collection of checked item in this checkedlistbox.
2. **CheckedIndices:** Collection of checked indexes in checkedlistbox.
3. **Items:** gets the collection of items in this checkedlistbox.

4. **Sorted:** gets or sets a value indicating wheter the item is sorted alphabetically.

Events:

1. **Click:** Occurs when the user clicks the CheckedListBox control.
2. **ItemCheck:** occur when the checked state of an item changed.
3. **SelectedIndexChanged:** Occur when the selectedindex property.

9) DataGridView: Displaying data in tabular format like rows and columns with the help of DataGridViewControl. The DataGridViewControl is designed to displaying tabular data in windows forms.
It makes easy to define the basic appearance of cells and the display formatting of cell values.All cells derive from the DataGridViewCell base class.



Properties:

1. **CellBorderStyle:** Gets the cell border style for the DataGridView.
2. **ColumnCount:** gets or sets the number of column displayed int the datagridview.
3. **Columns:** gets a collection that contains all the column in the control.
4. **CurrentRow:** get the current row containg the current cell.

Events:

1. **CellClick:** occur when any part of cell is clicked.
2. **CellLeave:** occur when a cell loses input focus and is no longer the current cell.
3. **Click:** occur when control is clicked.

Practical 10 :

AIM :: Create a Simple calculator.

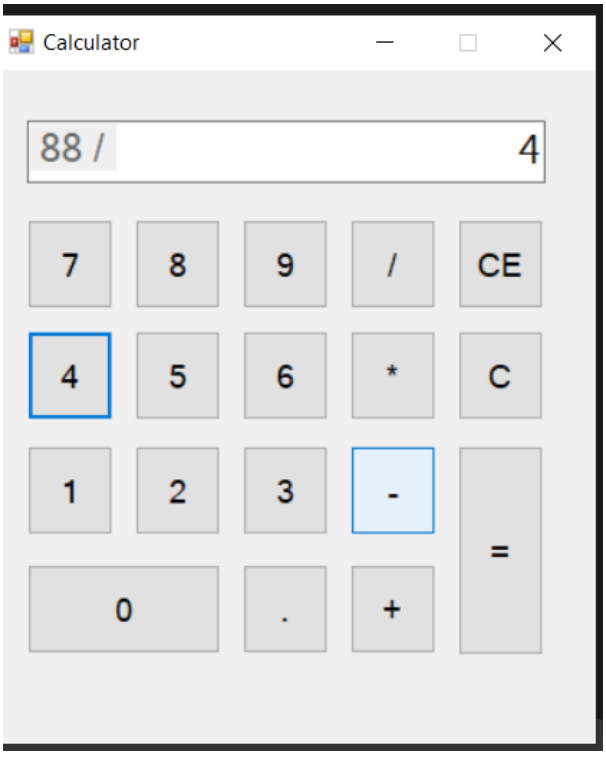
```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace Practical10
{
   public partial class Calculator : Form
   {
      Double value = 0;
      String operation = "";
      bool operation_pressed = false;
      public Calculator()
      {
         InitializeComponent();
```

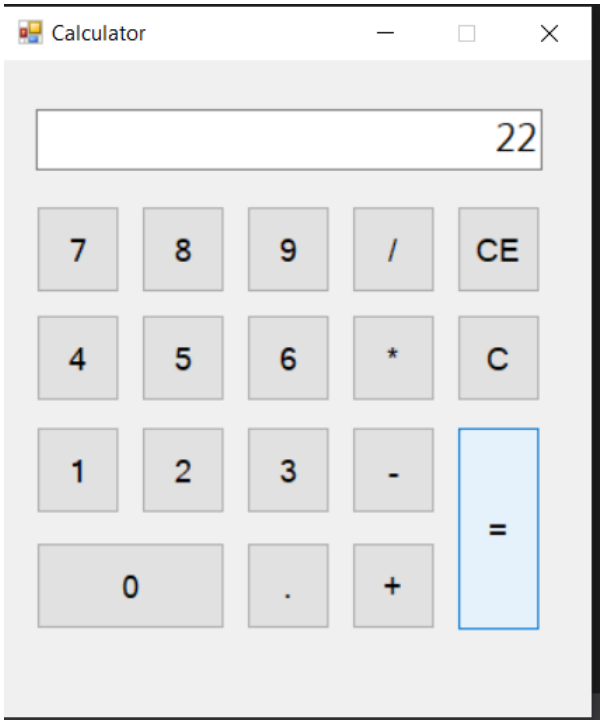
```
}

private void Calculator_Load(object sender, EventArgs e)
{

}

private void TextBox1_TextChanged(object sender, EventArgs e)
{

}

private void Button_Click(object sender, EventArgs e)
{
   if((result.Text == "0")|| operation_pressed)
   {
      result.Clear();
   }
   operation_pressed = false;
   Button b = (Button)sender;
   result.Text = result.Text + b.Text;
}
```

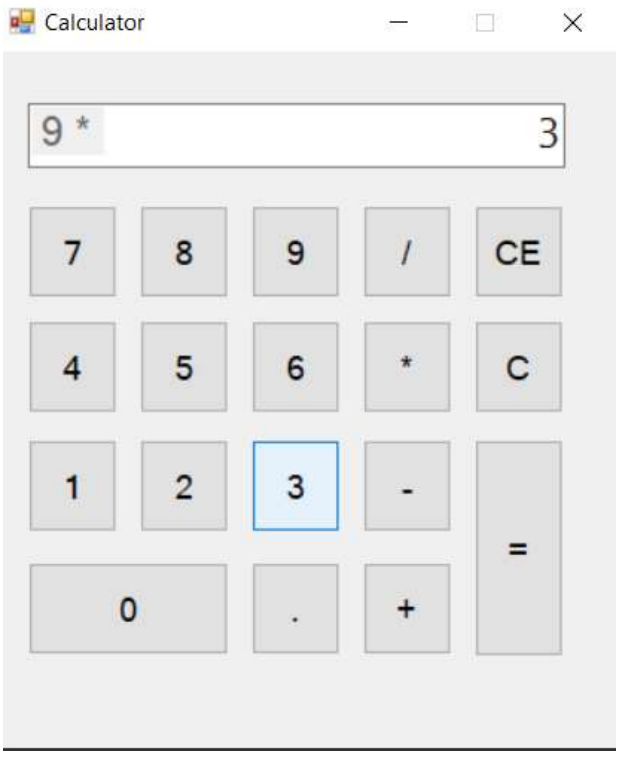
```
private void Button16_Click(object sender, EventArgs e)
{
    result.Text = "0";
}

private void operator_click(object sender, EventArgs e)
{
    Button b = (Button)sender;
    operation = b.Text;
    value = Double.Parse(result.Text);
    operation_pressed = true;
    equation.Text = value + " " + operation;
}

private void Button18_Click(object sender, EventArgs e)
{
    operation_pressed = false;
    equation.Text = "";
    switch(operation)
    {
        case "+":
```

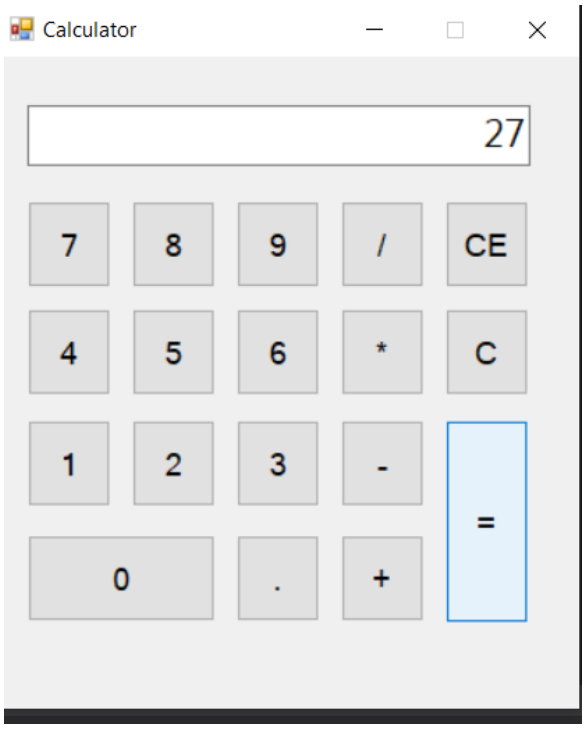
```
            result.Text = (value + Double.Parse(result.Text)).ToString(); break;
        case "-":
            result.Text = (value - Double.Parse(result.Text)).ToString(); break;
        case "*":
            result.Text = (value * Double.Parse(result.Text)).ToString(); break;
        case "/":
            result.Text = (value / Double.Parse(result.Text)).ToString(); break;
        default:
            break;
      }

    }

    private void Button17_Click(object sender, EventArgs e)
    {
      result.Clear();
      value = 0;
    }
  }
}
```

Output ::


Calculator
88 /
4
7 8 9 / CE
4 5 6 * C
1 2 3 - =
0 . +


Calculator
22
7 8 9 / CE
4 5 6 * C
1 2 3 - =
0 . +


Calculator
9 *
3
7 8 9 / CE
4 5 6 * C
1 2 3 - =
0 . +


Calculator
27
7 8 9 / CE
4 5 6 * C
1 2 3 - =
0 . +

Practical 11 :

AIM :: Create web application in ASP.NET to provide input validations using Input Valuators.

->Index.aspx

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

<!DOCTYPE html>

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

<head runat="server">

    <title></title>

    <style type="text/css">

        .auto-style1 {

            width: 70%;

            background-color: #808000;

            height: 45px;  }

        .auto-style2 { width: 261px; }

        td{ padding: 10px; }

        .auto-style4 { text-align: left; }

        .auto-style7 { margin-left: 120px;   }

        .auto-style8 {

            width: 261px;

            text-align: left;
```

```
        }
        .auto-style9 {
            text-align: left;
            margin-left: 120px;
        }
        .auto-style10 {color: #000000; }
        .auto-style11 {  background-color: #66FFCC; }
    </style>
</head>
<body style="width: 1251px; height: 545px">
    <form id="form1" runat="server">
        <div>
            <center>
                <h2 class="auto-style10"><strong><span class="auto-
style11">Registration form with validation</span></strong></h2>
                <br />
            </center>
                <table align="center" class="auto-style1">
                    <tr>
                        <td class="auto-style8">First Name</td>
            <center>
                        <td class="auto-style4">
```

```
                <asp:TextBox ID="txtFn" runat="server"></asp:TextBox>

                <asp:RequiredFieldValidator ID="RequiredFieldValidator1"
runat="server" ControlToValidate="txtFn" ErrorMessage="First Name is Empty"
ForeColor="Yellow"></asp:RequiredFieldValidator>

          </td>

            </tr>

        </center>

            <tr>

              <td class="auto-style8">Last Name</td>

        <center>

              <td class="auto-style9">

                <asp:TextBox ID="txtLn" runat="server"></asp:TextBox>

                <asp:RequiredFieldValidator ID="RequiredFieldValidator2"
runat="server" ControlToValidate="txtLn" ErrorMessage="Last Name is Empty"
ForeColor="Yellow"></asp:RequiredFieldValidator>

          </td>

            </tr>

            <tr>

              <td class="auto-style8">Email ID</td>

        <center>

              <td class="auto-style9">

                <asp:TextBox ID="txtId" runat="server"></asp:TextBox>
```

```
                <asp:RequiredFieldValidator ID="RequiredFieldValidator3"
runat="server" ControlToValidate="txtId" ErrorMessage="Email is Empty"
ForeColor="Yellow"></asp:RequiredFieldValidator>

                <asp:RegularExpressionValidator
ID="RegularExpressionValidator1" runat="server" ControlToValidate="txtId"
ErrorMessage="Ivalid Email" ForeColor="Blue" ValidationExpression="\w+([-
+.']\w+)*@\w+([-.]\w+)*\.\w+([-.]\w+)*"></asp:RegularExpressionValidator>

          </td>

            </tr>

            <tr>

              <td class="auto-style8">Password</td>

        <center>

              <td class="auto-style9">

                <asp:TextBox ID="txtPs" runat="server"
TextMode="Password"></asp:TextBox>

                <asp:RequiredFieldValidator ID="RequiredFieldValidator4"
runat="server" ControlToValidate="txtPs" ErrorMessage="Password is Blank"
ForeColor="Yellow"></asp:RequiredFieldValidator>

          </td>

            </tr>

            <tr>

              <td class="auto-style8">RE-Type Password</td>

        <center>

              <td class="auto-style9">
```

```
                <asp:TextBox ID="txtRps" runat="server"
TextMode="Password"></asp:TextBox>

                <asp:RequiredFieldValidator ID="RequiredFieldValidator5"
runat="server" ControlToValidate="txtRps" ErrorMessage="Re-type password is
blank" ForeColor="Yellow"></asp:RequiredFieldValidator>

                <asp:CompareValidator ID="CompareValidator1" runat="server"
ControlToCompare="txtPs" ControlToValidate="txtRps" ErrorMessage="Password
must be same" ForeColor="#0066FF"></asp:CompareValidator>

            </td>

              </tr>

              <tr>

                <td class="auto-style8">Age</td>

          <center>

                <td class="auto-style9">

                  <asp:TextBox ID="txtAge" runat="server"></asp:TextBox>

                  <asp:RequiredFieldValidator ID="RequiredFieldValidator6"
runat="server" ControlToValidate="txtAge" ErrorMessage="Enter Age"
ForeColor="Yellow"></asp:RequiredFieldValidator>

                  <asp:RangeValidator ID="RangeValidator1" runat="server"
ControlToValidate="txtAge" ErrorMessage="&gt;18 and &lt;30"
ForeColor="#0066FF" MaximumValue="30" MinimumValue="18"
Type="Integer"></asp:RangeValidator>

            </td>

              </tr>

              <tr>
```

```
                <td class="auto-style8">Mobile</td>
        <center>
                <td class="auto-style9">
                  <asp:TextBox ID="txtMobile" runat="server"></asp:TextBox>
                  <asp:RequiredFieldValidator ID="RequiredFieldValidator7"
runat="server" ControlToValidate="txtMobile" ErrorMessage="Mobile No is
empty" ForeColor="Yellow"></asp:RequiredFieldValidator>
                  <asp:RegularExpressionValidator
ID="RegularExpressionValidator2" runat="server" ControlToValidate="txtMobile"
ErrorMessage="Ivalid Number" ForeColor="Blue"
ValidationExpression="\d{10}"></asp:RegularExpressionValidator>
            </td>
              </tr>
              <tr>
                <td class="auto-style8">User ID</td>
        <center>
                <td class="auto-style9">
                  <asp:TextBox ID="txtUser" runat="server"></asp:TextBox>
                  <asp:RequiredFieldValidator ID="RequiredFieldValidator8"
runat="server" ControlToValidate="txtUser" ErrorMessage="user is empty"
ForeColor="Yellow"></asp:RequiredFieldValidator>
            </td>
              </tr>
              <tr>
```

```
            <td class="auto-style2"> </td>
            <td class="auto-style7"> </td>
        </tr>
        <tr>
            <td class="auto-style2"> </td>
            <td class="auto-style7">
                <asp:Button ID="btnS" runat="server" Text="Register Now" />
            </td>
        </tr>
    </table>
   
        <tr>
            <td class="auto-style2"> </td>
            <td class="auto-style7"> </td>
        </tr>
        <tr>
            <td class="auto-style2"> </td>
            <td class="auto-style7"> </td>
        </tr>
        <tr>
            <td class="auto-style2"> </td>
            <td class="auto-style7"> </td>
```

```
</tr>
<tr>
   <td class="auto-style2"> </td>
   <td class="auto-style7"> </td>
</tr>
<tr>
   <td class="auto-style2"> </td>
   <td class="auto-style7"> </td>
</tr>
<tr>
   <td class="auto-style2"> </td>
   <td class="auto-style7"> </td>
</tr>
<tr>
   <td class="auto-style2"> </td>
   <td class="auto-style7"> </td>
</tr>
<tr>
   <td class="auto-style2"> </td>
   <td class="auto-style7"> </td>
</tr>
<tr>
```

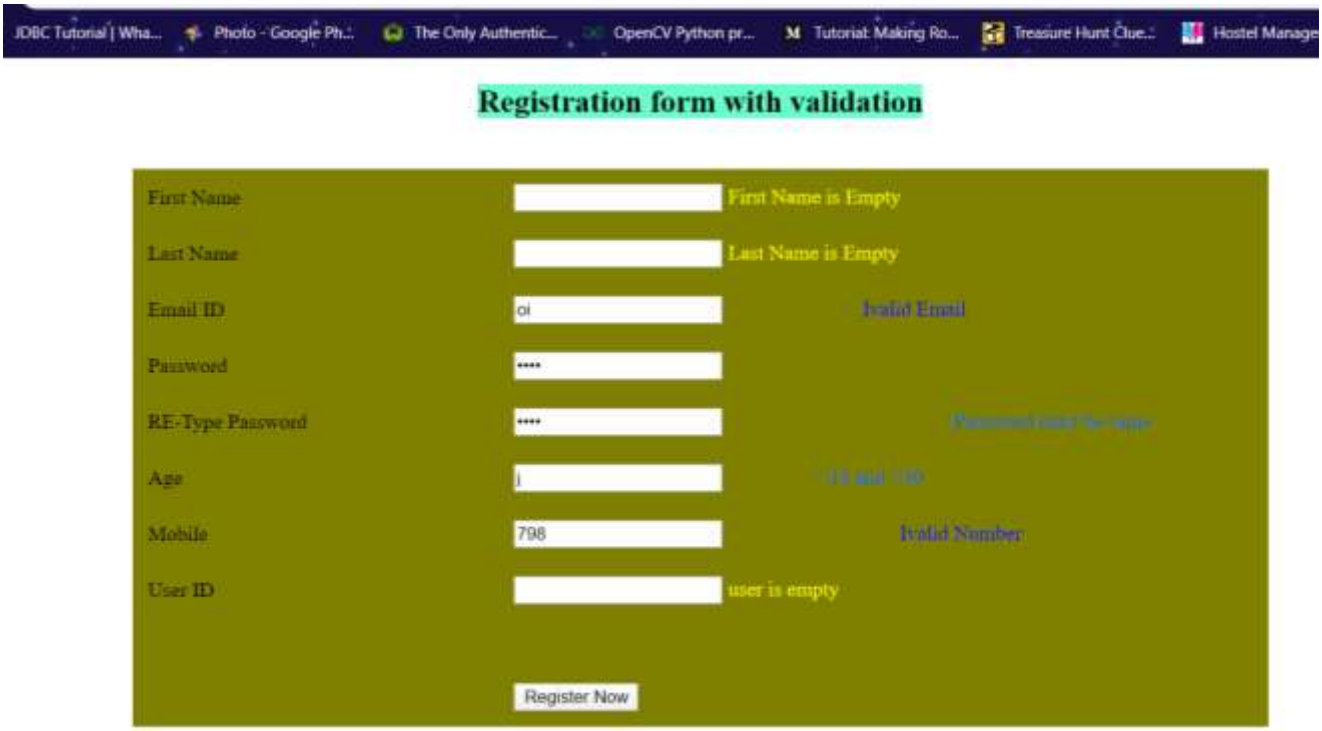
```
            <td class="auto-style2"> </td>
            <td class="auto-style7"> </td>
        </tr>
        <tr>
            <td class="auto-style2"> </td>
            <td class="auto-style7"> </td>
        </tr>
        <tr>
            <td class="auto-style2"> </td>
            <td class="auto-style7"> </td>
        </tr>
        <tr>
            <td class="auto-style2"> </td>
            <td class="auto-style7"> </td>
        </tr>
        <tr>
            <td class="auto-style2"> </td>
            <td class="auto-style6"> </td>
        </tr>
        <tr>
            <td class="auto-style2"> </td>
            <td class="auto-style5"> </td>
```

```
            </tr>
            <tr>
               <td class="auto-style2"> </td>
               <td> </td>
            </tr>
         </center></center></center></center></center></center></center>
      </div>
   </form>
</body>
</html>
```

->Index.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
public partial class _Default : System.Web.UI.Page
{    protected void Page_Load(object sender, EventArgs e)
     {   }
}
```

Output ::


JDBC Tutorial | Wha...
Photo - Google Ph...
The Only Authentic...
OpenCV Python pr...
Tutorial: Making Ro...
Treasure Hunt Clue...
Hostel Manage
Registration form with validation
First Name
First Name is Empty
Last Name
Last Name is Empty
Email ID
oi
Invalid Email
Password
RE-Type Password
Age
Mobile
798
Invalid Number
User ID
user is empty
Register Now

Practical 12 :

AIM :: Design a web form to allow user to enter following details in his Resume using Web Server Controls. Set validations using properties. When data is submitted it must be viewed in the panel below the form. Fields of Resume are Name, Address, City, Pin code, Phone, Gender, Qualification, Specialization subject, Percentage.

->Index.aspx

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


<!DOCTYPE html>


<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
    <style type="text/css">
        .auto-style1 {text-align: center;}
        .auto-style2 {
            width: 89%;
            height: 500px;
            margin-left: 66px;
            background-color: #C0C0C0;
        }
```

```
        .auto-style3 { width: 421px; }
        .auto-style4 { width: 421px;
            height: 36px; }
        .auto-style5 {height: 36px; }
        .auto-style6 { margin-left: 40px;
            background-color: #FFFFCC; }
    </style>
</head>
<body style="height: 51px">
    <form id="form1" runat="server">
    <h1 class="auto-style1">Detail Form</h1>
        <table class="auto-style2">
            <tr>
                <td class="auto-style3">  Name</td>
                <td>
                    <asp:TextBox ID="txtName" runat="server"
Width="352px"></asp:TextBox>
                    <asp:RequiredFieldValidator ID="RequiredFieldValidator1"
runat="server" ControlToValidate="txtName" ErrorMessage="Name is required"
ForeColor="Red"></asp:RequiredFieldValidator>
                </td>
            </tr>
            <tr>
```

```
            <td class="auto-style3">  Address</td>
            <td>
                <asp:TextBox ID="txtAddr" runat="server" Height="100px"
Width="353px"></asp:TextBox>
                <asp:RequiredFieldValidator ID="RequiredFieldValidator2"
runat="server" ControlToValidate="txtAddr" ErrorMessage="Fill Address"
ForeColor="Red"></asp:RequiredFieldValidator>
            </td>
        </tr>
        <tr>
            <td class="auto-style3">  City</td>
            <td>
                <asp:TextBox ID="txtCity" runat="server"
Width="350px"></asp:TextBox>
                <asp:RequiredFieldValidator ID="RequiredFieldValidator3"
runat="server" ControlToValidate="txtCity" ErrorMessage="City required"
ForeColor="Red"></asp:RequiredFieldValidator>
            </td>
        </tr>
        <tr>
            <td class="auto-style3">  Pin Code</td>
            <td>
                <asp:TextBox ID="txtPin" runat="server"
Width="348px"></asp:TextBox>
```

```
            <asp:RequiredFieldValidator ID="RequiredFieldValidator4"
runat="server" ControlToValidate="txtPin" ErrorMessage="pin missing"
ForeColor="Red"></asp:RequiredFieldValidator>

          </td>

        </tr>

        <tr>

          <td class="auto-style3">  Phone</td>

          <td>

            <asp:TextBox ID="txtPhone" runat="server"
Width="348px"></asp:TextBox>

            <asp:RegularExpressionValidator ID="RegularExpressionValidator1"
runat="server" ControlToValidate="txtPhone" ErrorMessage="Enter number in
proper format" ForeColor="Red"
ValidationExpression="\d{10}"></asp:RegularExpressionValidator>

          </td>

        </tr>

        <tr>

          <td class="auto-style3">  Gender</td>

          <td>

            <asp:TextBox ID="txtGender" runat="server"
Width="345px"></asp:TextBox>

            <asp:RequiredFieldValidator ID="RequiredFieldValidator6"
runat="server" ControlToValidate="txtGender" ErrorMessage="gender reqiured"
ForeColor="Red"></asp:RequiredFieldValidator>

          </td>
```

```
        </tr>
        <tr>
            <td class="auto-style3">  Qualification</td>
            <td>
                <asp:TextBox ID="txtQual" runat="server"
Width="349px"></asp:TextBox>
                <asp:RequiredFieldValidator ID="RequiredFieldValidator7"
runat="server" ControlToValidate="txtQual" ErrorMessage="qulification needed"
ForeColor="Red"></asp:RequiredFieldValidator>
            </td>
        </tr>
        <tr>
            <td class="auto-style4">  Specialisation Subject</td>
            <td class="auto-style5">
                <asp:TextBox ID="txtSub" runat="server"
Width="349px"></asp:TextBox>
                <asp:RequiredFieldValidator ID="RequiredFieldValidator8"
runat="server" ControlToValidate="txtSub" ErrorMessage="atleast on sub
required" ForeColor="Red"></asp:RequiredFieldValidator>
            </td>
        </tr>
        <tr>
            <td class="auto-style3">  Percentage(%)</td>
            <td>
```

```
            <asp:TextBox ID="txtPr" runat="server"
Width="351px"></asp:TextBox>

            <asp:RangeValidator ID="RangeValidator1" runat="server"
ControlToValidate="txtPr" ErrorMessage="pr must be &lt;=0  &amp; &gt;=100"
ForeColor="Red" MaximumValue="100" MinimumValue="0"
Type="Integer"></asp:RangeValidator>

          </td>

        </tr>

        <tr>

          <td class="auto-style3"> </td>

          <td> </td>

        </tr>

        <tr>

          <td class="auto-style3"> </td>

          <td>

            <asp:Button ID="Button1" runat="server" OnClick="Button1_Click"
Text="Submit" />

          </td>

        </tr>

    </table>

 

    <br />

 
```

```
<div class="auto-style6">
    Your Resume Details You Entered Are ::
    <br /><br /><br />
    Name :: <asp:Label ID="Label1" runat="server"></asp:Label>
    <br />
    <br />
    Address::<asp:Label ID="Label2" runat="server"></asp:Label>
    <br />
    <br />
    City ::<asp:Label ID="Label3" runat="server"></asp:Label>
    <br />
    <br />
    Pin Code::<asp:Label ID="Label4" runat="server"></asp:Label>
    <br />
    <br />
    Phone :::<asp:Label ID="Label5" runat="server"></asp:Label>
    <br />
    <br />
    Gender ::<asp:Label ID="Label6" runat="server"></asp:Label>
    <br />
    <br />
    Qualification ::<asp:Label ID="Label7" runat="server"></asp:Label>
```

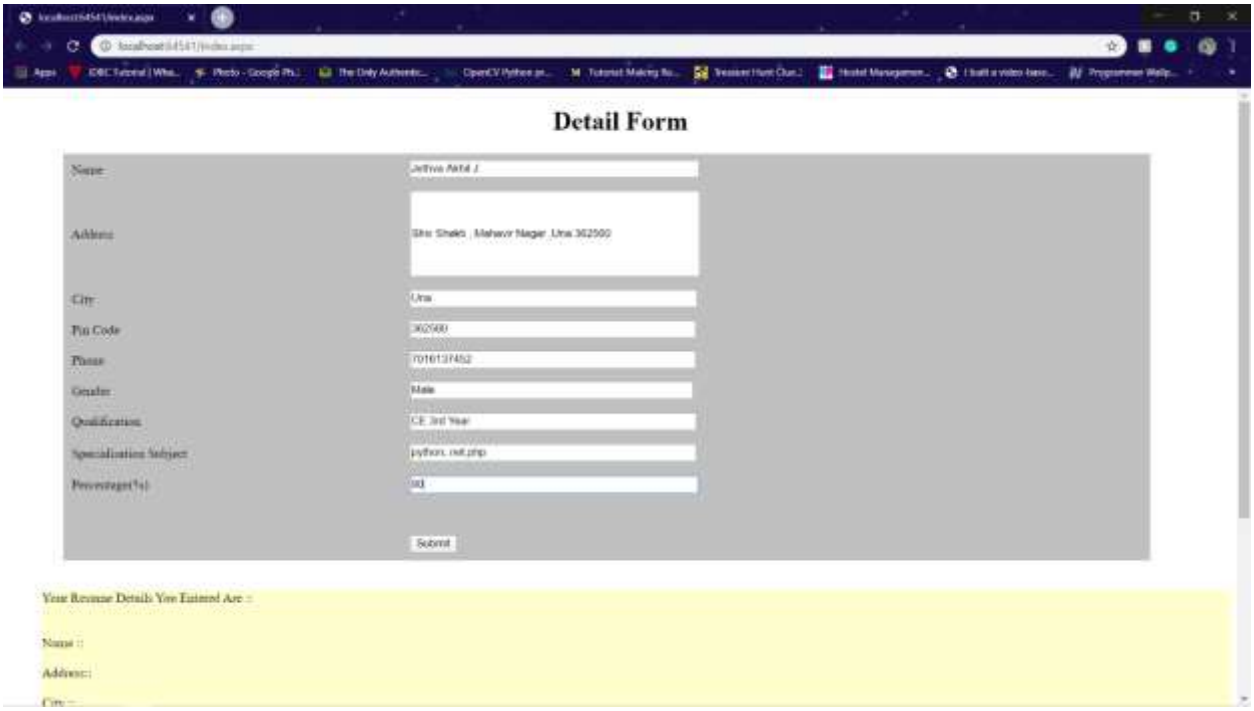
```
        <br />
 <br />
        Specialisation Subject ::<asp:Label ID="Label8"
runat="server"></asp:Label>
        <br />
        <br />
        Percentage ::<asp:Label ID="Label9" runat="server"></asp:Label>
        <br /><br /><br /><br /><br />
      </div>
   </form>
</body>
</html>
```

->index.aspx.cs

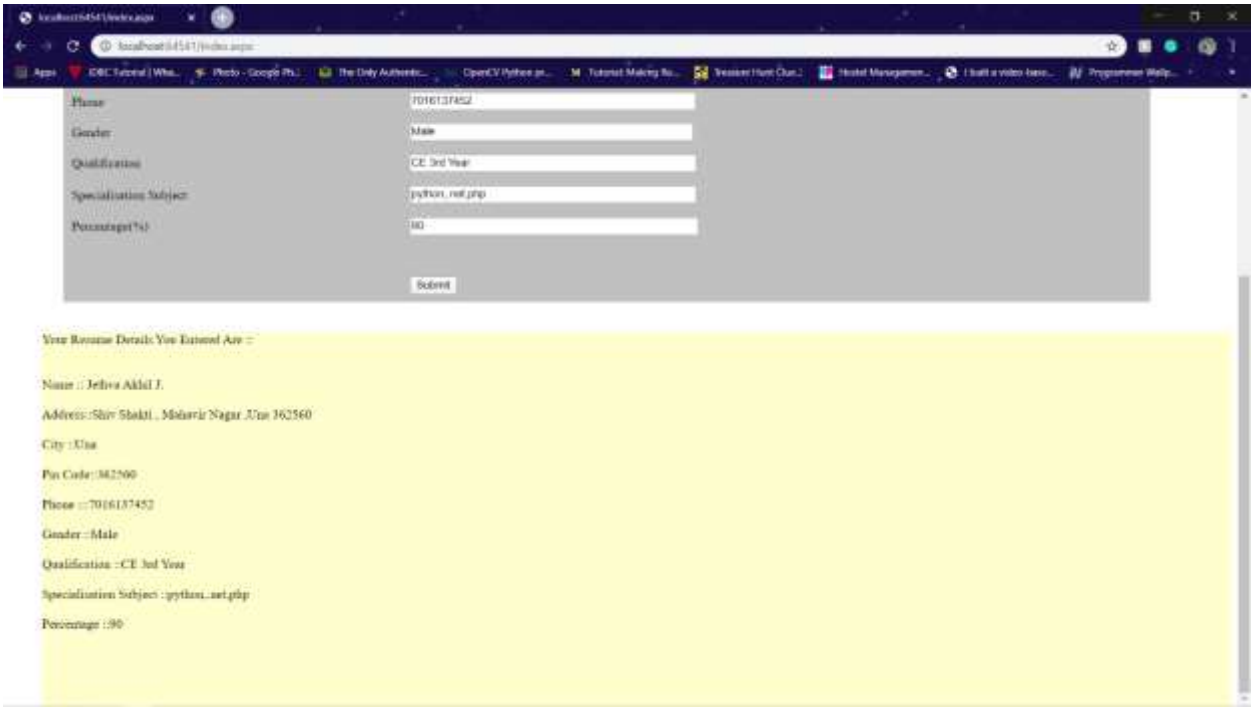
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
```

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


    }
    protected void Button1_Click(object sender, EventArgs e)
    {
        Label1.Text = txtName.Text;
        Label2.Text = txtAddr.Text;
        Label3.Text = txtCity.Text;
        Label4.Text = txtPin.Text;
        Label5.Text = txtPhone.Text;
        Label6.Text = txtGender.Text;
        Label7.Text = txtQual.Text;
        Label8.Text = txtSub.Text;
        Label9.Text = txtPr.Text;
    }
}
```

Output ::



After Submit::



Practical 13 :

AIM :: Create a web form where user enters following marks. DOT NET, Advanced JAVA, TOC, Web Technology, Software Engineering (All out of 100). When user submits the marks, numeric value validation must be done. On entering marks, the grade should be displayed in message box. IF % > 90 and <<=100 AA > 80 and <<=90 AB > 70 and <<=80 BB > 60 and <<=67 BC >50 and <<=60 CC >40 and <= 50 DD Else Fail

->Index.aspx

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


<!DOCTYPE html>


<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
   <title></title>
   <style type="text/css">
      .auto-style1 {
         text-align: center;
         height: 361px;
      }

      .auto-style2 {
         text-align: center;
```

```
    height: 50px;
}
.auto-style3 {
    width: 68%;
    margin-left: 133px;
    background-color: #C0C0C0;
}
.auto-style4 {
    width: 340px;
}
td,table,tr{
    margin: 7px;
}
.auto-style5 {
    text-align: left;
}
.auto-style6 {
    width: 340px;
    text-align: left;
}
.auto-style7 {
    width: 340px;
```

```
            height: 26px;
        }
        .auto-style8 {
            height: 26px;
        }
    </style>
</head>
<body>
    <form id="form1" runat="server">
        <div>
        </div>
            <h2 class="auto-style2"><strong>Student MarkSheet</strong></h2>
    <table class="auto-style3">
        <tr>
            <th class="auto-style6">   Subject Name</th>
            <th class="auto-style5">      
Marks(0-100)</th>
        </tr>
        <tr>
            <td class="auto-style4"> </td>
            <td> </td>
        </tr>
```

```
<tr>
    <td class="auto-style4">  DOT Net</td>
    <td>     
        <asp:TextBox ID="txtSub1" runat="server"></asp:TextBox>
        <asp:RangeValidator ID="RangeValidator1" runat="server"
ControlToValidate="txtSub1" ErrorMessage="Number Must Be &gt;=0 and
&lt;=100" ForeColor="Red" MaximumValue="100" MinimumValue="0"
Type="Integer"></asp:RangeValidator>
    </td>
</tr>
<tr>
    <td class="auto-style4">  Advance JAVA</td>
    <td>     
        <asp:TextBox ID="txtSub2" runat="server"></asp:TextBox>
        <asp:RangeValidator ID="RangeValidator2" runat="server"
ControlToValidate="txtSub2" ErrorMessage="Number Must Be &gt;=0 and
&lt;=100" ForeColor="Red" MaximumValue="100" MinimumValue="0"
Type="Integer"></asp:RangeValidator>
    </td>
</tr>
<tr>
    <td class="auto-style4">  TOC</td>
    <td>     
        <asp:TextBox ID="txtSub3" runat="server"></asp:TextBox>
```

```
            <asp:RangeValidator ID="RangeValidator3" runat="server"
ControlToValidate="txtSub3" ErrorMessage="Number Must Be &gt;=0 and
&lt;=100" ForeColor="Red" MaximumValue="100" MinimumValue="0"
Type="Integer"></asp:RangeValidator>

        </td>

      </tr>

      <tr>

        <td class="auto-style4">  Web Technology </td>

        <td>     

          <asp:TextBox ID="txtSub4" runat="server"></asp:TextBox>

          <asp:RangeValidator ID="RangeValidator4" runat="server"
ControlToValidate="txtSub4" ErrorMessage="Number Must Be &gt;=0 and
&lt;=100" ForeColor="Red" MaximumValue="100" MinimumValue="0"
Type="Integer"></asp:RangeValidator>

        </td>

      </tr>

      <tr>

        <td class="auto-style4">  Software Engineering</td>

        <td>     

          <asp:TextBox ID="txtSub5" runat="server"></asp:TextBox>

          <asp:RangeValidator ID="RangeValidator5" runat="server"
ControlToValidate="txtSub5" ErrorMessage="Number Must Be &gt;=0 and
&lt;=100" ForeColor="Red" MaximumValue="100" MinimumValue="0"
Type="Integer"></asp:RangeValidator>

        </td>
```

```
        </tr>
        <tr>
            <td class="auto-style4"> </td>
            <td> </td>
        </tr>
        <tr>
            <td class="auto-style4"> </td>
            <td>     
                <asp:Button ID="Button1" runat="server" Text="Submit"
OnClick="Button1_Click" />
            </td>
        </tr>
        <tr>
            <td class="auto-style4"> </td>
            <td> </td>
        </tr>
        <tr>
            <td class="auto-style7">Strudent Grade Is</td>
            <td class="auto-style8">       
                <asp:Label ID="Label1" runat="server"></asp:Label>
            </td>
        </tr>
```

```
    </table>
    </form>
    <p class="auto-style1"> </p>
</body>
</html>
```

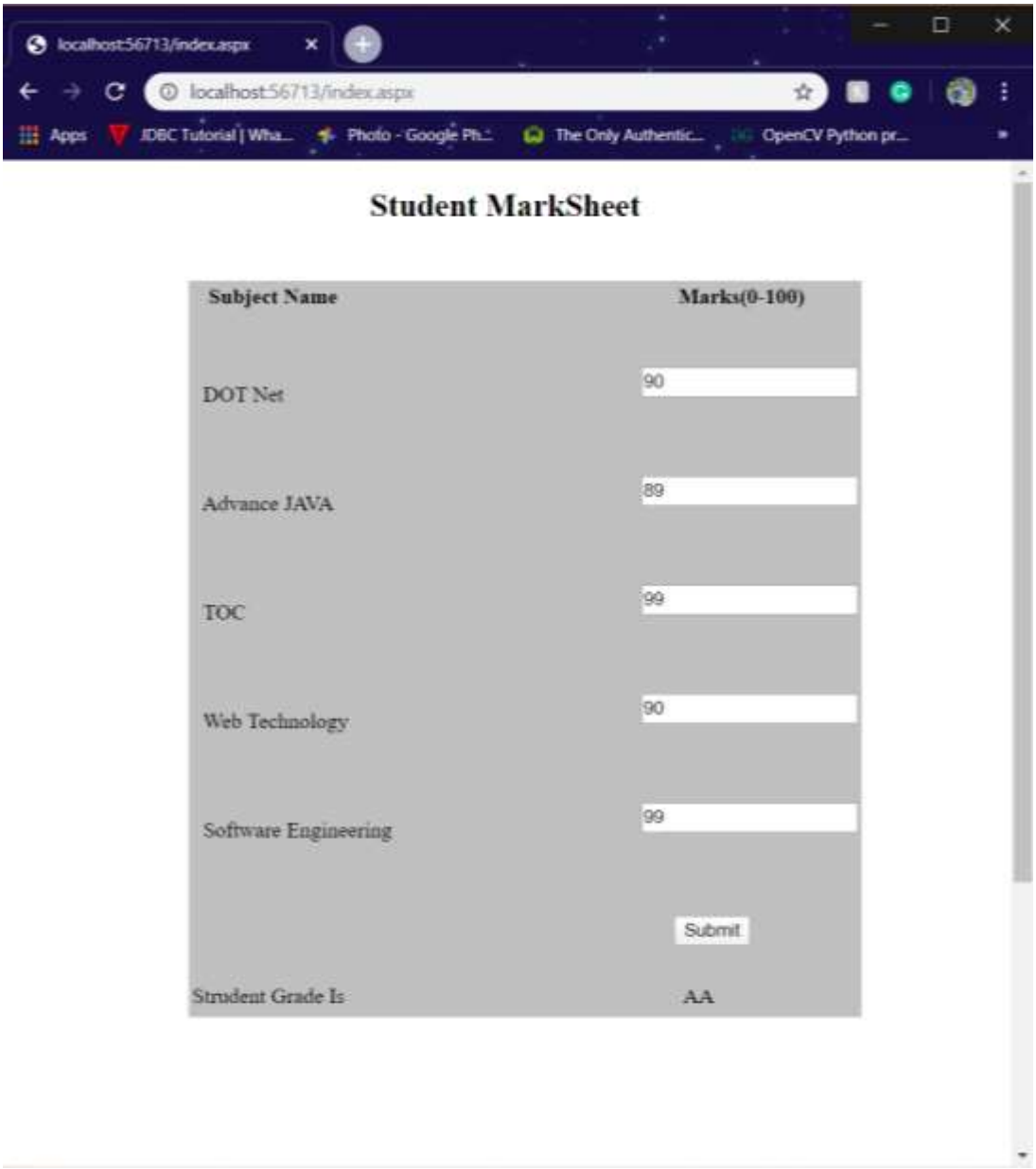
->index.aspx.cs

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

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

    }

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

```
        float sum = 0;
        sum = float.Parse(txtSub1.Text) + float.Parse(txtSub2.Text) +
float.Parse(txtSub3.Text) + float.Parse(txtSub4.Text) + float.Parse(txtSub5.Text);
        float pr = sum / 5;


        if(pr > 90 && pr <= 100)
        {
           Label1.Text = "AA";
        }
        else if(pr > 80 && pr <= 90)
        {
           Label1.Text = "AB";
        }
        else if (pr > 70 && pr <= 80)
        {
           Label1.Text = "BB";
        }
        else if (pr > 60 && pr <= 70)
        {
           Label1.Text = "BC";
        }
        else if (pr > 50 && pr <= 60)
```

```
        {
            Label1.Text = "CC";
        }
        else if (pr > 40 && pr <= 50)
        {
            Label1.Text = "CD";
        }
        else if (pr > 30 && pr <= 40)
        {
            Label1.Text = "DD";
        }
        else
        {
            Label1.Text = "Fail";

        }
    }
}
```

Output ::


localhost:56713/index.aspx
localhost:56713/index.aspx
Apps
JDBC Tutorial | Wha...
Photo - Google Ph...
The Only Authentic...
OpenCV Python pr...
Student MarkSheet
Subject Name
Marks(0-100)
DOT Net
90
Advance JAVA
89
TOC
99
Web Technology
90
Software Engineering
99
Submit
Strudent Grade Is
AA

Practical 14 :

AIM :: Create a web application that will use concepts of master page and theme concept

->Masterpage.master

```
<%@ Master Language="C#" AutoEventWireup="true"
CodeFile="MasterPage1.master.cs" Inherits="MasterPage1" %>


<!DOCTYPE html>


<html>
<head runat="server">
   <title>
      <asp:ContentPlaceHolder id="title" runat="server">
      </asp:ContentPlaceHolder>
   </title>
   <asp:ContentPlaceHolder id="head" runat="server">
   </asp:ContentPlaceHolder>
   <style type="text/css">
      .auto-style1 {
         padding-top:20px;
         text-align: center;
         height: 52px;
```

```
        background-color: #CC33FF;
    }
    .auto-style3 {
        color: #003300;
        background-color: #FFFF66;
    }
    body {
        background-image: url(http://localhost:64998/images/im1.png)
    }
  </style>
</head>
<body>
  <form id="form1" runat="server">
  <div>
    <h2 class="auto-style1">Web Applicxation Name for All Web pages</h2>
    <asp:ContentPlaceHolder id="body" runat="server">
    </asp:ContentPlaceHolder>

    <br />
    <br /><br /><br /><br /><br /><br /><br /><br /><br /><br /><br /><br
/><br /><br /><br />
    <br /><br /><br /><br /><br /><br /><br />
```

```
        <h4 class="auto-style3">@copyRight for all web pages</h4>
    </div>
    </form>
</body>
</html>
```

->index.aspx

```
<%@ Page Title="" Language="C#" MasterPageFile="~/MasterPage1.master"
AutoEventWireup="true" CodeFile="index.aspx.cs" Inherits="index" %>


<asp:Content ID="Content1" ContentPlaceHolderID="title" Runat="Server">
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="head" Runat="Server">
    <style type="text/css">
        .auto-style4 {
            color: #FFFFFF;
        }
        .auto-style6 {
            color: #FFFFFF;
            text-align: right;
        }
    .auto-style7 {
        width: 1111px;
```

```
        height: 134px;
        margin-left: 53px;
    }
        .newStyle1 {
            font-family: Castellar;
        }
        .auto-style8 {
            text-align: center;
        }
        .newStyle2 {
            font-family: "Yu Gothic UI Semibold";
        }
        .auto-style9 {
            font-family: "Yu Gothic UI Semibold";
            color: #0000CC;
            text-decoration: underline;
        }
        .auto-style10 {
            color: #0000CC;
            text-decoration: underline;
        }
        .auto-style11 {
```

```
            color: #FF3300;
            background-color: #99FF66;
        }
        .auto-style12 {
            text-decoration: underline;
            color: #66FFCC;
            background-color: #2F3934;
        }
        .auto-style13 {
            color: #FF0000;
            background-color: #2F3934;
        }
        .auto-style14 {
            background-color: #99FF66;
        }
    </style>
</asp:Content>
<asp:Content ID="Content3" ContentPlaceHolderID="body" Runat="Server">
    <div class="auto-style7">
    <h1>
        <span class="auto-style4"><em class="newStyle1">Hello Everyone this is the
Home Page</em></span>
```

```
    </h1>
    <p class="auto-style4">
         </p>
    <h2 class="auto-style8">
        <span class="auto-style10"><em>W</em></span><span class="auto-style9"><em>elcome to the first page of out web site .</em></span></h2>
    <p class="auto-style6">
         </p>
        <p class="auto-style6">
             </p>
        <h2>
            <span class="auto-style14"><span class="auto-style12">Click Here to next page ::  </span></span>
            <span class="auto-style11"><a href="second.aspx"><span class="auto-style13">Next</span></a></span></h2>
    </div>
</asp:Content>
```

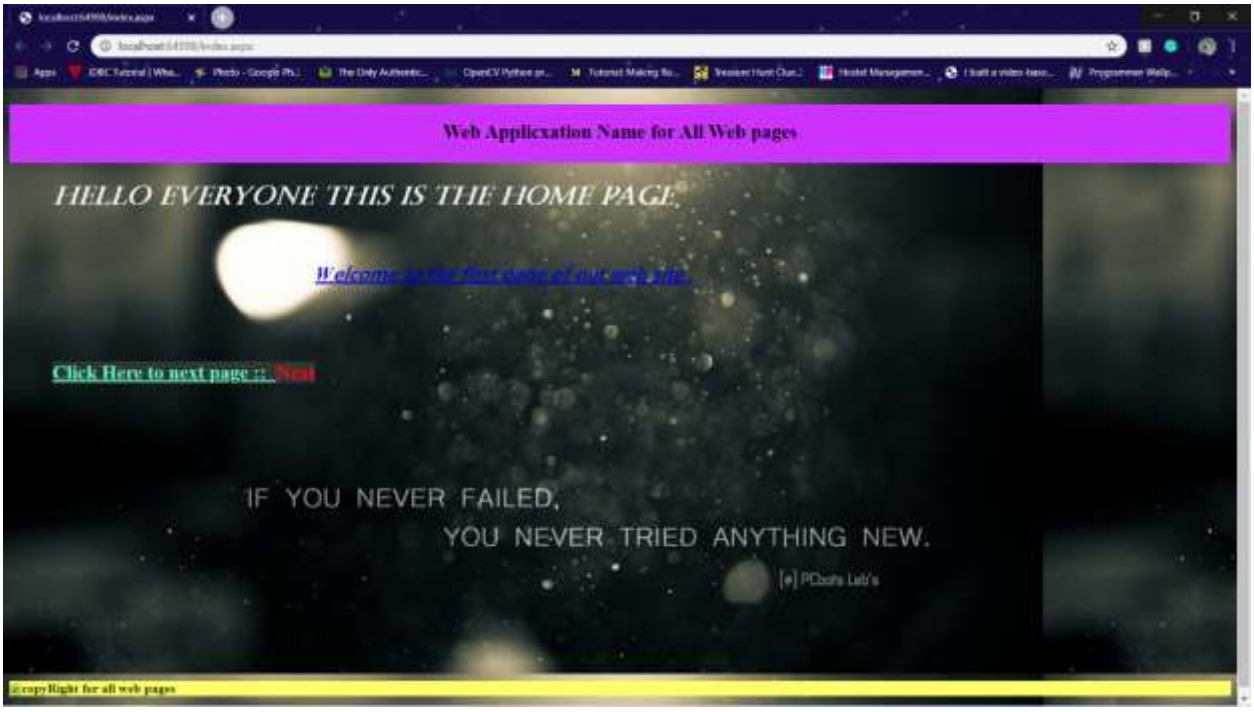
->second.aspx

```
<%@ Page Title="" Language="C#" MasterPageFile="~/MasterPage1.master" AutoEventWireup="true" CodeFile="second.aspx.cs" Inherits="second" %>


<asp:Content ID="Content1" ContentPlaceHolderID="title" Runat="Server">
```

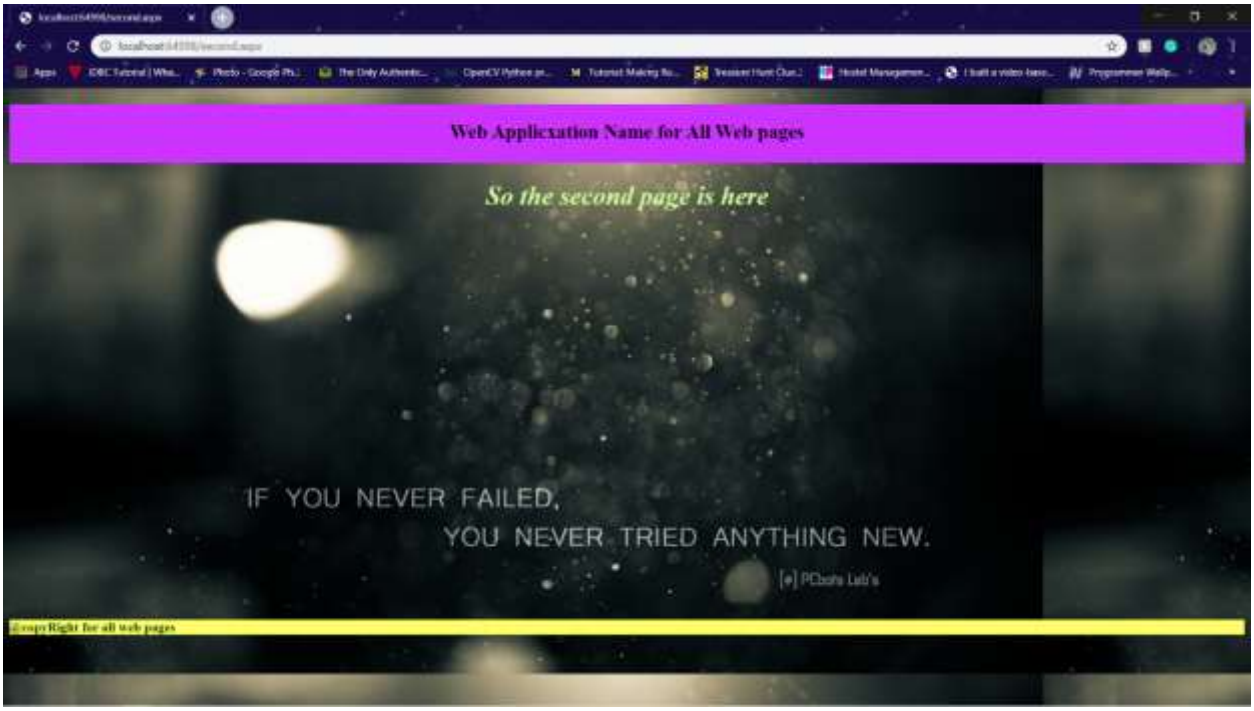
```
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="head" Runat="Server">
   <style type="text/css">
      .auto-style4 {
         color: #CCFF99;
         text-align: center;
      }
   </style>
</asp:Content>
<asp:Content ID="Content3" ContentPlaceHolderID="body" Runat="Server">
   <h1 class="auto-style4"><em>So the second page is here </em></h1>
</asp:Content>
```

Output ::

Index page



Second page



Practical 15 :

AIM :: Create a web application that will use the concept of State Management (I) View State (II) Query String (III) Cookies (IV) Session.

->Login.aspx:

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="login.aspx.cs"
Inherits="PRAC15.login" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
        <title></title>
</head>
<body>
        <form id="form1" runat="server">
        <div>
        <asp:Label ID="Label1" runat="server" Text="Enter name :"></asp:Label>
        <asp:TextBox ID="txtMail" runat="server"
        Width="219px"></asp:TextBox><br/><br/>
        <asp:Button ID="Button2" runat="server" Text="ViewState"
        Width="308px" onclick="Button2_Click" /><br/><br />
        <asp:Button ID="btnQuery" runat="server" Text="Next Page"
        Width="308px" onclick="btnQuery_Click" /><br/><br/>
        <asp:Label ID="lblView" runat="server"
        Text=""></asp:Label><br/><br/>
        </div>
        </form>
</body>
</html>
```

->Login.aspx.cs:

```
using System;
using System.Collections.Generic;
```

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

namespace PRAC15
{
   public partial class login :System.Web.UI.Page
   {
      protected void Button2_Click(object sender, EventArgs e)
      {
                  string user = txtMail.Text;
                  ViewState["Name"] = user;
                  lblView.Text = "Value of ViewStateis : " +
                  ViewState["Name"];
      }
      protected void btnQuery_Click(object sender, EventArgs e)
{
                  string user = txtMail.Text;
                  Session["user"] = user;
                  HttpCookiepcookie = new HttpCookie("Persistance");
                  pcookie.Value = "This is Persistence cookie";
                  pcookie.Expires = DateTime.Now.AddSeconds(10);
                  Response.Cookies.Add(pcookie);
                  Response.Redirect("welcome.aspx?name=" + user);
}
   }
}
```

***->Welcome.aspx*:**

```
<%@ Page Language="C#" AutoEventWireup="true"
CodeBehind="welcome.aspx.cs" Inherits="PRAC15.welcome" %>

<!DOCTYPE html>

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

```
<head runat="server">
        <title></title>
</head>
<body>
        <form id="form1" runat="server">
        <div>
        Welcome
        <asp:Label ID="lblQuery" runat="server" ></asp:Label>
        <asp:Label ID="lblSession" runat="server" ></asp:Label>
        <asp:Label ID="lblCookie" runat="server" ></asp:Label>
        </div>
        </form>
</body>
</html>
```

->Welcome.aspx.cs:

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

namespace PRAC15
{
   public partial class welcome :System.Web.UI.Page
   {
      protected void Page_Load(object sender, EventArgs e)
      {
         lblQuery.Text = "<br/>Value of QueryStringis : " +
         Request.QueryString["name"];
         lblSession.Text = "<br/>Value of Sessoin variable is: "
         + Session["user"];
         lblCookie.Text = "<br/>Value of Cookie is: " +
         Request.Cookies["Persistance"].Value;
      }
```

```
    }
}
```

Output ::


Enter name : abc@mail.com
ViewState
Next Page
Value of ViewState is : abc@mail.com
Welcome
Value of QueryString is : abc@mail.com
Value of Sessoin variable is: abc@mail.com
Value of Cookie is: This is Persistence cookie