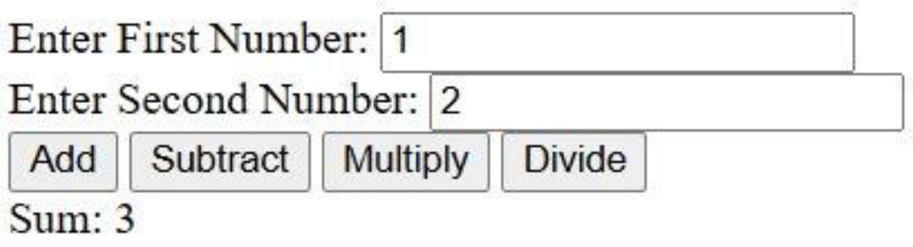
Prac 1a.: Aim: Create an application to print on screen the output of adding, subtracting, multiplying and dividing two numbers entered by the user in C#.


Enter First Number: 1

Enter Second Number: 2

Sum: 3

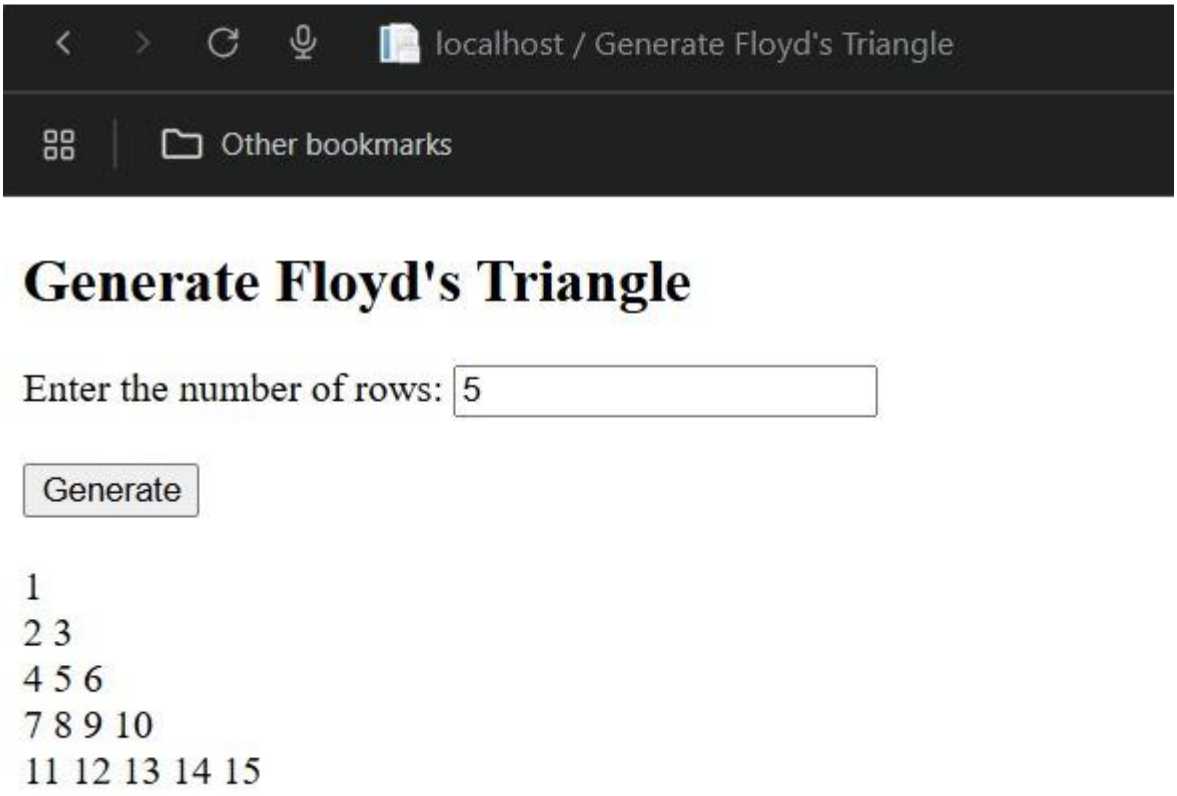
//prac1a webform.aspx

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="WebForm2.aspx.cs"
Inherits="WebApplication2.WebForm2" %>
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
  <title>Simple Calculator</title>
</head>
<body>
  <form id="form1" runat="server">
    <div>
      <label>Enter First Number:</label>
      <asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>
    </div>
    <div>
      <label>Enter Second Number:</label>
      <asp:TextBox ID="TextBox2" runat="server"></asp:TextBox>
    </div>
    <div>
      <asp:Button ID="Button1" runat="server" Text="Add" OnClick="Button1_Click" />
      <asp:Button ID="Button2" runat="server" Text="Subtract" OnClick="Button2_Click" />
      <asp:Button ID="Button3" runat="server" Text="Multiply" OnClick="Button3_Click" />
      <asp:Button ID="Button4" runat="server" Text="Divide" OnClick="Button4_Click" />
    </div>
    <div>
      <asp:Label ID="lblResult" runat="server" Text=""></asp:Label>
    </div>
  </form>
</body>
</html>
```

// webform.aspx.cs

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

namespace WebApplication2
{
  public partial class WebForm2 : Page
  {
    protected void Button1_Click(object sender, EventArgs e)
    {
      int num1 = int.Parse(TextBox1.Text);
      int num2 = int.Parse(TextBox2.Text);
      lblResult.Text = "Sum: " + (num1 + num2).ToString();
    }

    protected void Button2_Click(object sender, EventArgs e)
    {
      int num1 = int.Parse(TextBox1.Text);
      int num2 = int.Parse(TextBox2.Text);
      lblResult.Text = "Difference: " + (num1 - num2).ToString();
    }

    protected void Button3_Click(object sender, EventArgs e)
    {
      int num1 = int.Parse(TextBox1.Text);
      int num2 = int.Parse(TextBox2.Text);
      lblResult.Text = "Product: " + (num1 * num2).ToString();
    }

    protected void Button4_Click(object sender, EventArgs e)
    {
      int num1 = int.Parse(TextBox1.Text);
      int num2 = int.Parse(TextBox2.Text);
      if (num2 != 0)
        lblResult.Text = "Quotient: " + (num1 / num2).ToString();
      else
        lblResult.Text = "Cannot divide by zero!";
    }
  }
}
```

PRACTICAL 1B AIM: Create an application to print Floyd’s triangle till n rows in C#. Step 1: Add a Web Form



Generate Floyd's Triangle

Enter the number of rows: 5

```
1
2 3
4 5 6
7 8 9 10
11 12 13 14 15
```

//prac 1b webform2.aspx

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

<!DOCTYPE html>

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

<head runat="server">

  <title>Generate Floyd's Triangle</title>

</head>

<body>

  <form id="form1" runat="server">

    <div>

      <h2>Generate Floyd's Triangle</h2>

      <asp:Label ID="lblPrompt" runat="server" Text="Enter the number of rows:"></asp:Label>

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

      <br /><br />
```

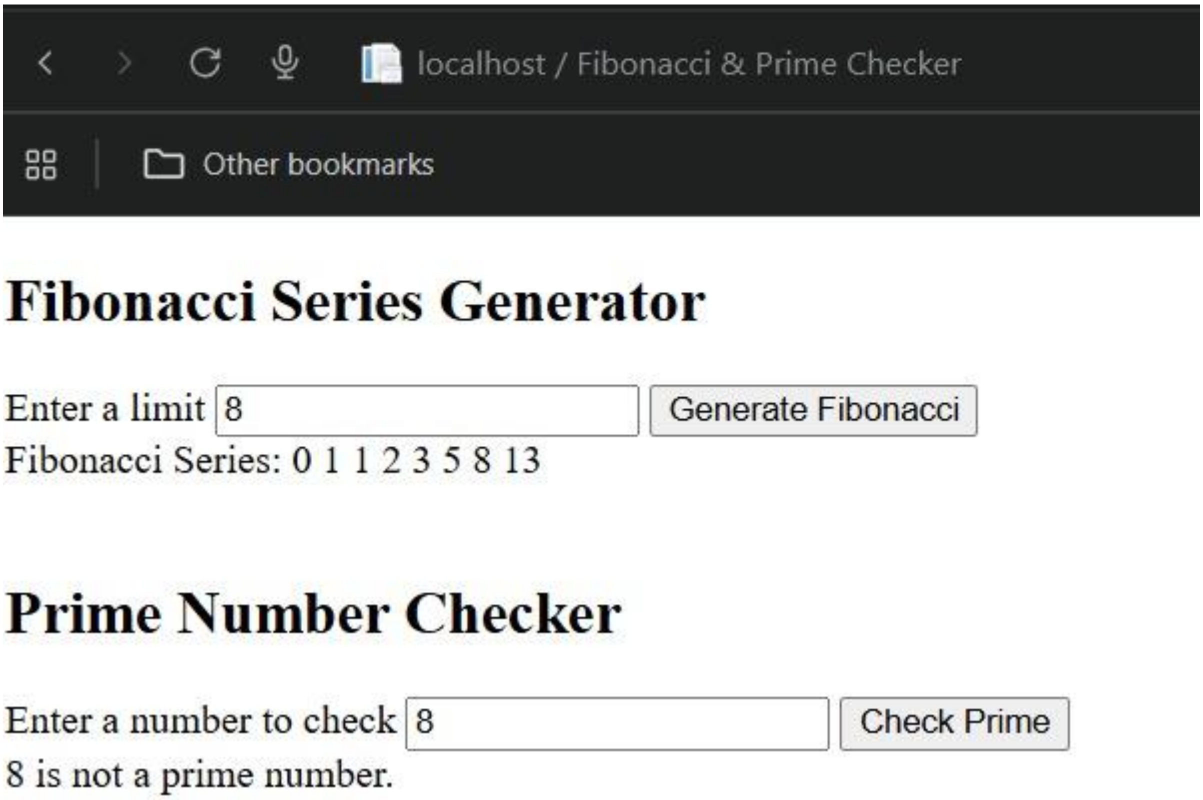
```
         <asp:Button ID="btnGenerate" runat="server" Text="Generate"
OnClick="btnGenerate_Click" />

         <br /><br />

         <asp:Label ID="fTriangle" runat="server"></asp:Label>

      </div>

   </form>

</body>

</html>
```

//webform2.aspx.cs

```
using System;


namespace YourNamespace

{

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

   {

      protected void btnGenerate_Click(object sender, EventArgs e)

      {

         int n = Convert.ToInt32(txtRow.Text);

         string result = "";

         int number = 1;

         for (int i = 1; i <= n; i++)

         {

            for (int j = 1; j <= i; j++)

            {

               result += number + " ";

               number++;

            }

            result += "<br/>";

         }
```

```
            fTriangle.Text = result;
        }
    }
}
```

PRACTICAL 1C AIM: Create an application to demonstrate following operations i. Generate Fibonacci series. ii. Test for prime numbers.



Fibonacci Series Generator

Enter a limit 8

Fibonacci Series: 0 1 1 2 3 5 8 13

Prime Number Checker

Enter a number to check 8

8 is not a prime number.

//prac1c webform2.aspx

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="WebForm2.aspx.cs" Inherits="YourNamespace.WebForm2" %>
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
  <title>Fibonacci & Prime Checker</title>
</head>
<body>
  <form id="form1" runat="server">
    <div>
      <h2>Fibonacci Series Generator</h2>
      <asp:Label ID="lblFibPrompt" runat="server" Text="Enter a limit"></asp:Label>
      <asp:TextBox ID="txtFibonacci" runat="server"></asp:TextBox>
      <asp:Button ID="btnGenerateFibonacci" runat="server" Text="Generate Fibonacci" OnClick="btnGenerateFibonacci_Click" />
      <br />
      <asp:Label ID="lblFibonacci" runat="server" Text=""></asp:Label>
      <br /><br />

      <h2>Prime Number Checker</h2>
      <asp:Label ID="lblPrimePrompt" runat="server" Text="Enter a number to check"></asp:Label>
      <asp:TextBox ID="txtPrime" runat="server"></asp:TextBox>
      <asp:Button ID="btnCheckPrime" runat="server" Text="Check Prime" OnClick="btnCheckPrime_Click" />
      <br />
      <asp:Label ID="lblPrime" runat="server" Text=""></asp:Label>
    </div>
  </form>
```

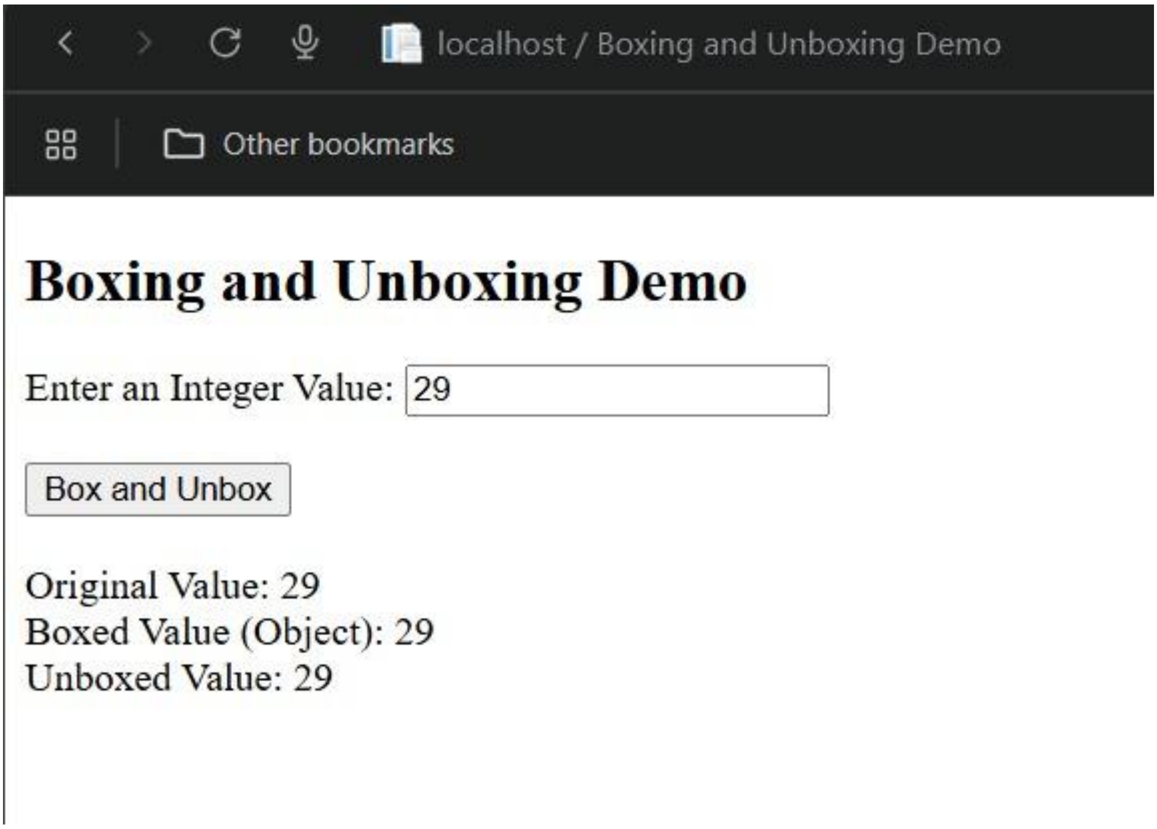
```
</body>
</html>
```

//webform2.aspx.cs

```
using System;

namespace YourNamespace
{
  public partial class WebForm2 : System.Web.UI.Page
  {
    protected void btnGenerateFibonacci_Click(object sender, EventArgs e)
    {
      int terms;
      if (int.TryParse(txtFibonacci.Text, out terms) && terms > 0)
      {
        string result = "";
        int a = 0, b = 1;

        for (int i = 0; i < terms; i++)
        {
          result += a + " ";
          int next = a + b;
          a = b;
          b = next;
        }
        lblFibonacci.Text = "Fibonacci Series: " + result;
      }
      else
      {
```

```
                lblFibonacci.Text = "Please enter a valid number greater than 0.";
            }
        }

        protected void btnCheckPrime_Click(object sender, EventArgs e)
        {
            int number;
            if (int.TryParse(txtPrime.Text, out number) && number > 1)
            {
                bool isPrime = true;
                for (int i = 2; i <= Math.Sqrt(number); i++)
                {
                    if (number % i == 0)
                    {
                        isPrime = false;
                        break;
                    }
                }
                lblPrime.Text = isPrime
                    ? $"{number} is a prime number."
                    : $"{number} is not a prime number.";
            }
            else
            {
                lblPrime.Text = "Please enter a valid number greater than 1.";
            }
        }
    }
}
```

PRACTICAL 2A: Aim: Create a simple application to demonstrate the concepts boxing and unboxing.



Boxing and Unboxing Demo

Enter an Integer Value: 29

Box and Unbox

```
Original Value: 29
Boxed Value (Object): 29
Unboxed Value: 29
```

//prac2a webform2.aspx

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

<!DOCTYPE html>

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

<head runat="server">

  <title>Boxing and Unboxing Demo</title>

</head>

<body>

  <form id="form1" runat="server">
```

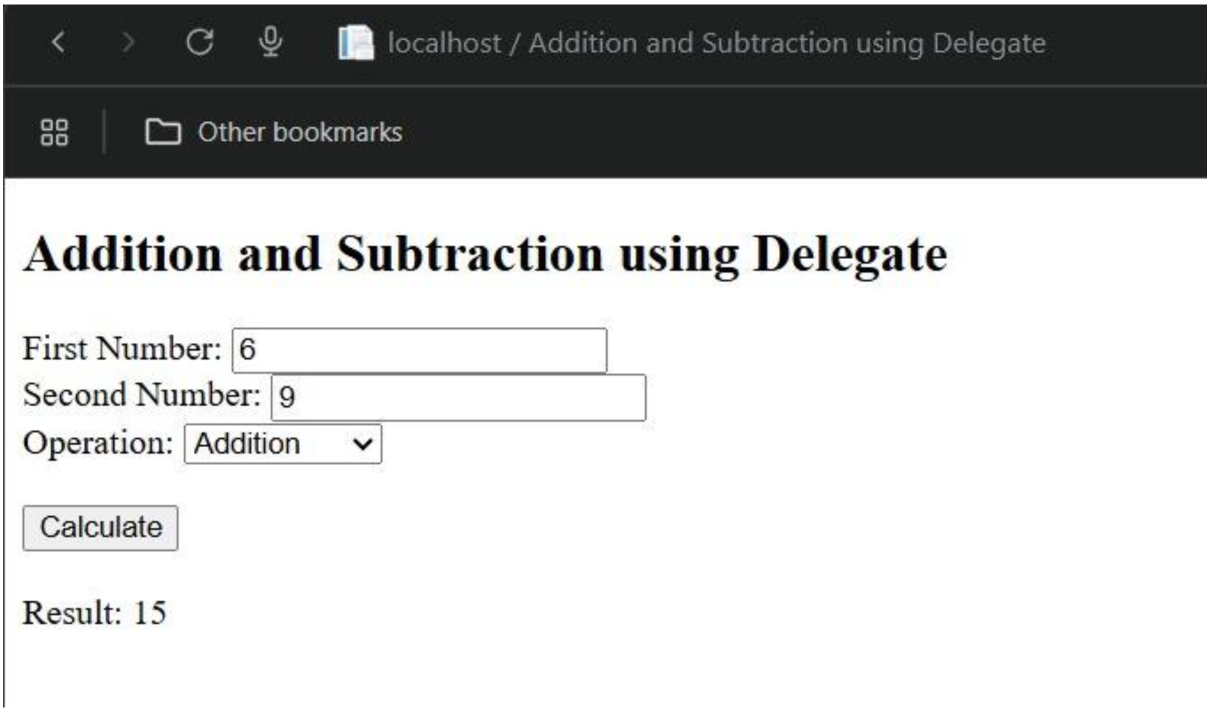
```
    <div>
      <h2>Boxing and Unboxing Demo</h2>
      <asp:Label ID="LblPrompt" runat="server" Text="Enter an Integer Value:"></asp:Label>
      <asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>
      <br /><br />
      <asp:Button ID="Button1" runat="server" Text="Box and Unbox" OnClick="Button1_Click" />
      <br /><br />
      <asp:Label ID="ResultLabel" runat="server" Text=""></asp:Label>
    </div>
  </form>
</body>
</html>
```

//webform.aspx.cs

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

namespace YourNamespace
{
  public partial class WebForm4 : System.Web.UI.Page
  {
    protected void Button1_Click(object sender, EventArgs e)
    {
      // Read the input value from the textbox
      int intValue = int.Parse(TextBox1.Text);

      // Boxing: Converting value type (int) to reference type (object)
```

```
            object boxedValue = intValue;

            // Unboxing: Converting reference type (object) back to value type (int)
            int unboxedValue = (int)boxedValue;

            // Display the results
            ResultLabel.Text = $"Original Value: {intValue}<br />" +
                               $"Boxed Value (Object): {boxedValue}<br />" +
                               $"Unboxed Value: {unboxedValue}";
        }
    }
}
```

PRACTICAL 2B: Aim: Create a simple application to perform addition and subtraction using delegate.



Addition and Subtraction using Delegate

First Number: 6
Second Number: 9
Operation: Addition

Result: 15

//prac2b webform2.aspx

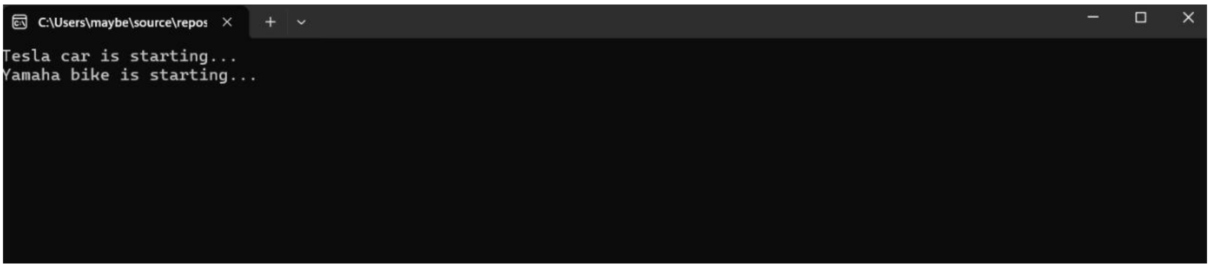
```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="WebForm2.aspx.cs"
Inherits="Devang.WebForm2" %>

<!DOCTYPE html>

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

<head runat="server">

  <title>Addition and Subtraction using Delegate</title>

</head>

<body>

  <form id="form1" runat="server">

    <div>

      <h2>Addition and Subtraction using Delegate</h2>

      <asp:Label ID="LblFirst" runat="server" Text="First Number:"></asp:Label>

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

      <br />

      <asp:Label ID="LblSecond" runat="server" Text="Second Number:"></asp:Label>

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

      <br />

      <asp:Label ID="LblOp" runat="server" Text="Operation:" />

      <asp:DropDownList ID="ddlOperation" runat="server">

        <asp:ListItem>Addition</asp:ListItem>

        <asp:ListItem>Subtraction</asp:ListItem>

      </asp:DropDownList>

      <br /><br />

      <asp:Button ID="btnCalculate" runat="server" Text="Calculate"
OnClick="btnCalculate_Click" />
```

```
            <br /><br />
            <asp:Label ID="lblResult" runat="server" Text=""></asp:Label>
        </div>
    </form>
</body>
</html>
```

//webform2.aspx.cs

```
using System;

namespace Devang
{
    public partial class WebForm2 : System.Web.UI.Page
    {
        public delegate int OperationDelegate(int x, int y);

        // Addition method
        public int Add(int x, int y)
        {
            return x + y;
        }

        // Subtraction method
        public int Subtract(int x, int y)
        {
            return x - y;
        }
    protected void btnCalculate_Click(object sender, EventArgs e)
    {
      int firstNumber = Convert.ToInt32(txtFirstNumber.Text);
      int secondNumber = Convert.ToInt32(txtSecondNumber.Text);

      OperationDelegate operation;
      if (ddlOperation.SelectedValue == "Addition")
        operation = new OperationDelegate(Add);
      else
        operation = new OperationDelegate(Subtract);

      int result = operation(firstNumber, secondNumber);
      lblResult.Text = "Result: " + result.ToString();
    }
  }
}
```

PRACTICAL 2C: Aim: Create a simple application to demonstrate use of the concepts of interfaces



```
Tesla car is starting...
Yamaha bike is starting...
```

isme webform mat create karna .net ka consoleapp create karna and usme upar left mai 3 files create karne ka option ayega vo create karna code niche diya hai filename match krna chaiye exact classes se.

Filename: Interface1.cs

```
public interface IVehicle

{

    string Model { get; set; }

    void Start(); // Method signature

}
```

Filename: Vehicle.cs

```
using System;


public class Car : IVehicle

{

    public string Model { get; set; }


    public Car(string model)

    {
```

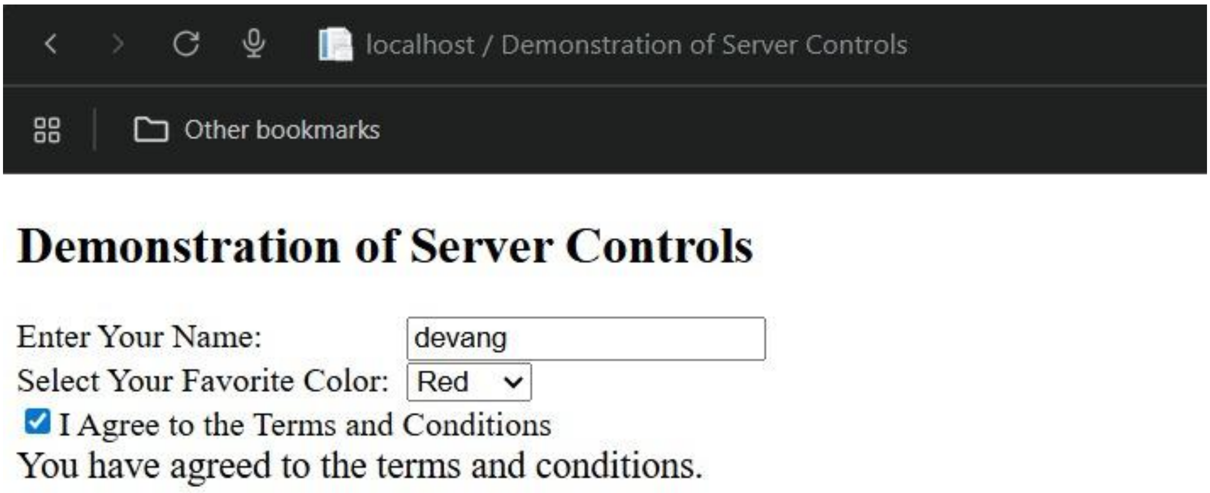
```
    Model = model;
  }

  public void Start()
  {
    Console.WriteLine($"{Model} car is starting...");
  }
}

public class Bike : IVehicle
{
  public string Model { get; set; }

  public Bike(string model)
  {
    Model = model;
  }

  public void Start()
  {
    Console.WriteLine($"{Model} bike is starting...");
  }
}
```

Filename: Program.cs

```
using System;

class Program
```

```
{
  static void Main(string[] args)
  {
    // Create instances of Car and Bike
    IVehicle car = new Car("Tesla");
    IVehicle bike = new Bike("Yamaha");

    // Use the interface methods
    car.Start();    // Output: Tesla car is starting...
    bike.Start();   // Output: Yamaha bike is starting...

    Console.ReadKey();
  }
}
```

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



Demonstration of Server Controls

Enter Your Name: devang

Select Your Favorite Color: Red

☑ I Agree to the Terms and Conditions

You have agreed to the terms and conditions.

//prac3a webform1.aspx

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="WebForm1.aspx.cs"
Inherits="WebApplication5.WebForm1" %>

<!DOCTYPE html>

<script runat="server">


  protected void Page_Load(object sender, EventArgs e)

  {


  }


  protected void txtName_TextChanged(object sender, EventArgs e)
  {


  }
</script>


<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
  <title>Demonstration of Server Controls</title>
  <style>
    .lbl { width: 180px; display: inline-block; }
  </style>
</head>
<body>
  <form id="form1" runat="server">
    <h2>Demonstration of Server Controls</h2>
    <div>
      <span class="lbl">Enter Your Name:</span>
      <asp:TextBox ID="txtName" runat="server" AutoPostBack="true"
OnTextChanged="txtName_TextChanged"></asp:TextBox>
    </div>
    <div>
      <span class="lbl">Select Your Favorite Color:</span>
      <asp:DropDownList ID="ddlColor" runat="server" AutoPostBack="true"
OnSelectedIndexChanged="ddlColor_SelectedIndexChanged">
        <asp:ListItem Text="Red" Value="Red"></asp:ListItem>
        <asp:ListItem Text="Green" Value="Green"></asp:ListItem>
        <asp:ListItem Text="Blue" Value="Blue"></asp:ListItem>
      </asp:DropDownList>
    </div>
    <div>
```

```
      <asp:CheckBox ID="chkAgree" runat="server" Text="I Agree to the Terms and Conditions" AutoPostBack="true" OnCheckedChanged="chkAgree_CheckedChanged" />
    </div>
    <div>
      <asp:Label ID="lblOutput" runat="server" Font-Size="Large" ForeColor="Black"></asp:Label>
    </div>
  </form>
</body>
</html>
```

//webform1.aspx.cs

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

namespace WebApplication5
{
  public partial class WebForm1 : Page
  {
    protected void txtName_TextChanged(object sender, EventArgs e)
    {
      lblOutput.Text = "Hello, " + txtName.Text + "!";
    }

    protected void ddlColor_SelectedIndexChanged(object sender, EventArgs e)
    {
      lblOutput.Text = "Your favorite color is " + ddlColor.SelectedItem.Text + ".";
    }

    protected void chkAgree_CheckedChanged(object sender, EventArgs e)
    {
      if (chkAgree.Checked)
        lblOutput.Text = "You have agreed to the terms and conditions.";
      else
        lblOutput.Text = "You have not agreed to the terms and conditions.";
    }
  }
}
```

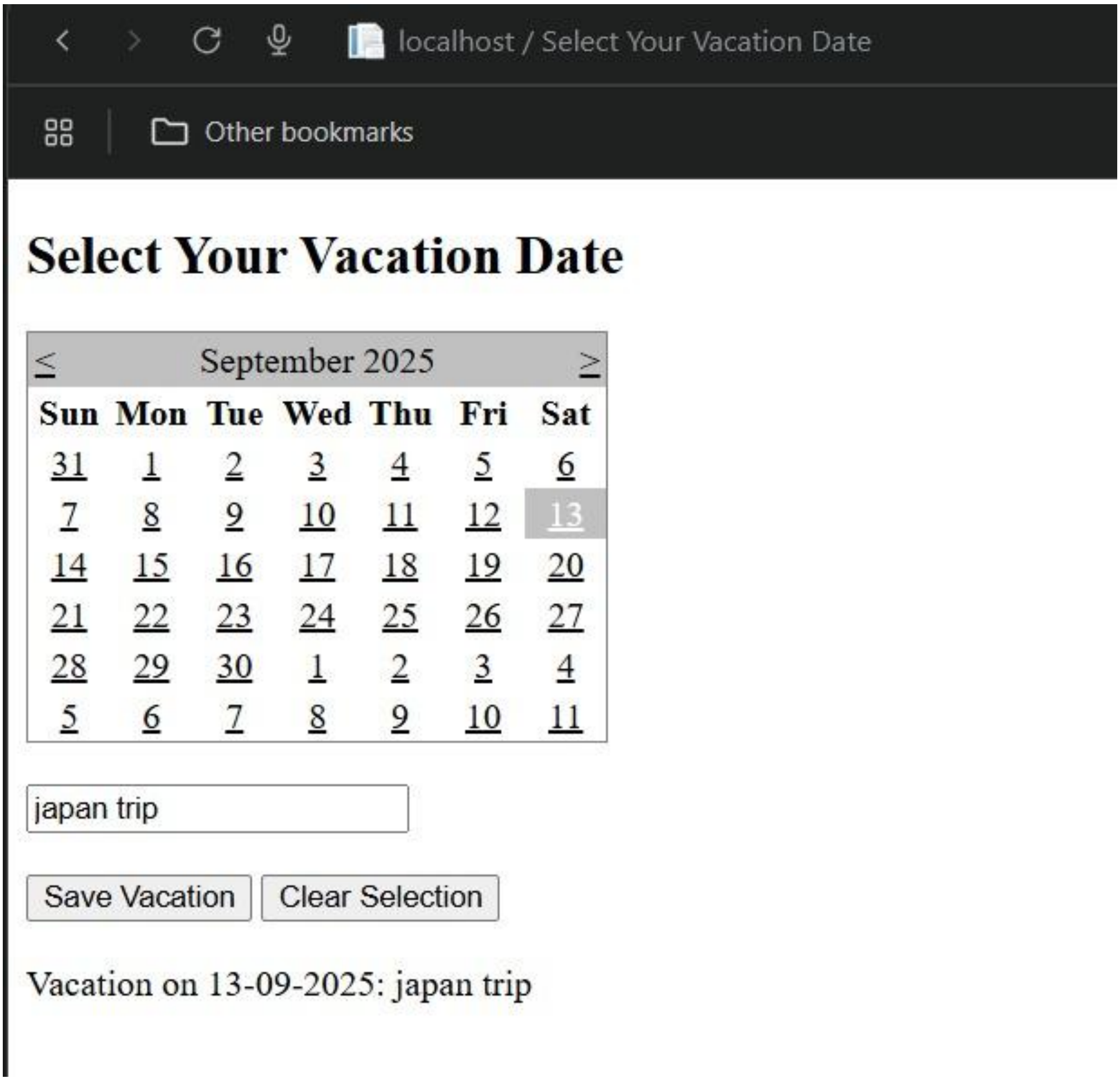
//webform.aspx.designer.cs

```
namespace WebApplication5
{


  public partial class WebForm1
  {

    /// <summary>
    /// form1 control.
    /// </summary>
    /// <remarks>
    /// Auto-generated field.
    /// To modify move field declaration from designer file to code-behind file.
    /// </remarks>
    protected global::System.Web.UI.HtmlControls.HtmlForm form1;
```

```
/// <summary>
/// txtName control.
/// </summary>
/// <remarks>
/// Auto-generated field.
/// To modify move field declaration from designer file to code-behind file.
/// </remarks>
protected global::System.Web.UI.WebControls.TextBox txtName;

/// <summary>
/// ddlColor control.
/// </summary>
/// <remarks>
/// Auto-generated field.
/// To modify move field declaration from designer file to code-behind file.
/// </remarks>
protected global::System.Web.UI.WebControls.DropDownList ddlColor;

/// <summary>
/// chkAgree control.
/// </summary>
/// <remarks>
/// Auto-generated field.
/// To modify move field declaration from designer file to code-behind file.
/// </remarks>
protected global::System.Web.UI.WebControls.CheckBox chkAgree;

/// <summary>
```

```
    /// lblOutput control.
    /// </summary>
    /// <remarks>
    /// Auto-generated field.
    /// To modify move field declaration from designer file to code-behind file.
    /// </remarks>
    protected global::System.Web.UI.WebControls.Label lblOutput;
  }
}
```

PRACTICAL 3B: Aim: Create a simple application to demonstrate your vacation using calendar control.



Select Your Vacation Date

<			September 2025			>
Sun	Mon	Tue	Wed	Thu	Fri	Sat
31	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	1	2	3	4
5	6	7	8	9	10	11

japan trip

Vacation on 13-09-2025: japan trip

//prac3b webform1.aspx

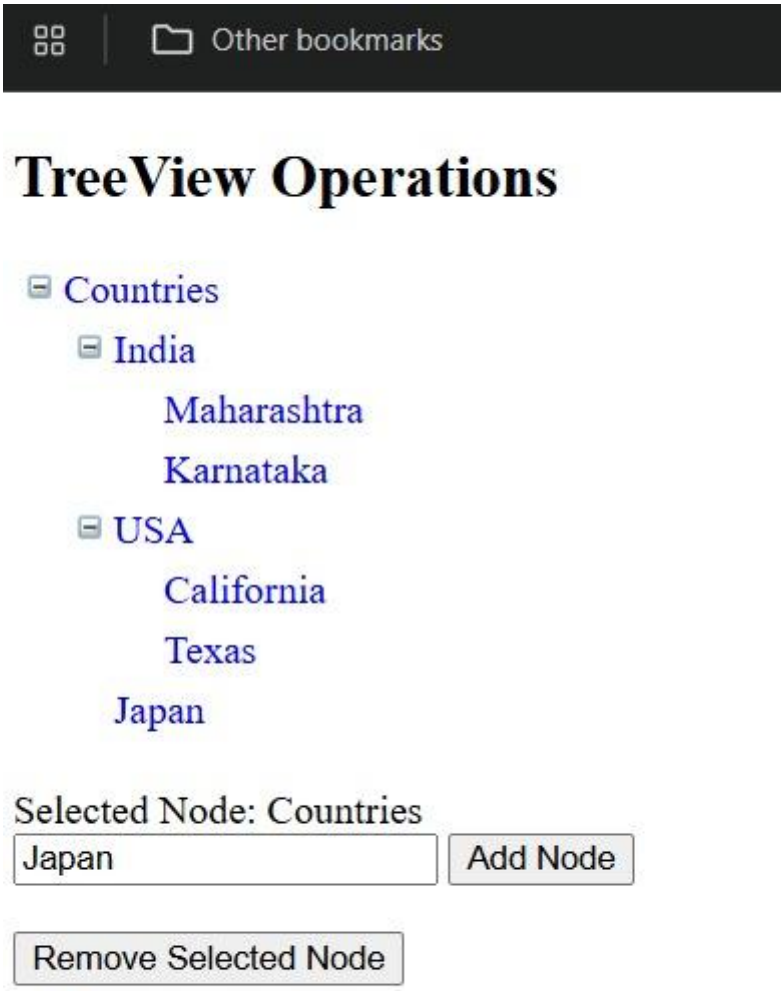
```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="WebForm1.aspx.cs"
Inherits="YourNamespace.WebForm1" %>

<!DOCTYPE html>

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

<head runat="server">

  <title>Select Your Vacation Date</title>

</head>
```

```
<body>
  <form id="form1" runat="server">
    <div>
      <h2>Select Your Vacation Date</h2>
      <asp:Calendar ID="vacationCalendar" runat="server"
OnSelectionChanged="vacationCalendar_SelectionChanged"></asp:Calendar>
      <br />
      <asp:TextBox ID="TextBoxDetails" runat="server" Placeholder="Enter vacation
details"></asp:TextBox>
      <br /><br />
      <asp:Button ID="ButtonSave" runat="server" Text="Save Vacation"
OnClick="ButtonSave_Click" />
      <asp:Button ID="btnClearSelection" runat="server" Text="Clear Selection"
OnClick="btnClearSelection_Click" />
      <br /><br />
      <asp:Label ID="lblResult" runat="server" Text="" />
    </div>
  </form>
</body>
</html>
```

//webfor1.aspx.cs

```
using System;

namespace YourNamespace
{
  public partial class WebForm1 : System.Web.UI.Page
  {
    protected void btnClearSelection_Click(object sender, EventArgs e)
    {
        vacationCalendar.SelectedDates.Clear();
        lblResult.Text = "Selected Date: ";
    }

    protected void vacationCalendar_SelectionChanged(object sender, EventArgs e)
    {
        lblResult.Text = "Selected Date: " +
vacationCalendar.SelectedDate.ToShortDateString();
    }

    protected void ButtonSave_Click(object sender, EventArgs e)
    {
        string vacationDate = vacationCalendar.SelectedDate.ToShortDateString();
        string vacationDetails = TextBoxDetails.Text;
        if (string.IsNullOrEmpty(vacationDetails))
        {
            lblResult.Text = "Please enter vacation details!";
        }
        else
        {
            lblResult.Text = "Vacation on " + vacationDate + ": " + vacationDetails;
        }
    }
  }
}
```

PRACTICAL 3C: Aim: Demonstrate the use of Treeview operations on the web form.



TreeView Operations

- Countries
  - India
    - Maharashtra
    - Karnataka
  - USA
    - California
    - Texas
  - Japan

Selected Node: Countries

Japan

// prac3c webform1.aspx

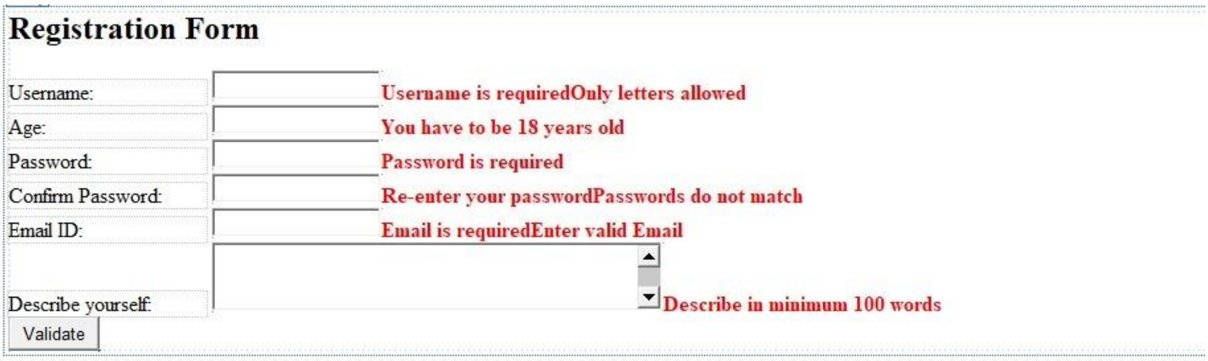
```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="WebForm1.aspx.cs"
Inherits="Devang.WebForm1" %>

<!DOCTYPE html>

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

<head runat="server">

  <title>TreeView Operations</title>

</head>
```

```
<body>
  <form id="form1" runat="server">
    <div>
      <h2>TreeView Operations</h2>
      <asp:TreeView ID="treeView1" runat="server" OnSelectedNodeChanged="treeView1_SelectedNodeChanged">
        <Nodes>
          <asp:TreeNode Text="Countries" Value="Countries">
            <asp:TreeNode Text="India" Value="India">
              <asp:TreeNode Text="Maharashtra" Value="Maharashtra"></asp:TreeNode>
              <asp:TreeNode Text="Karnataka" Value="Karnataka"></asp:TreeNode>
            </asp:TreeNode>
            <asp:TreeNode Text="USA" Value="USA">
              <asp:TreeNode Text="California" Value="California"></asp:TreeNode>
              <asp:TreeNode Text="Texas" Value="Texas"></asp:TreeNode>
            </asp:TreeNode>
          </asp:TreeNode>
        </Nodes>
      </asp:TreeView>
      <br />
      <asp:Label ID="lblSelectedNode" runat="server" Text="Selected Node:" />
      <br />
      <asp:TextBox ID="txtNodeText" runat="server" />
      <asp:Button ID="btnAddNode" runat="server" Text="Add Node" OnClick="btnAddNode_Click" />
      <br /><br />
      <asp:Button ID="btnRemoveNode" runat="server" Text="Remove Selected Node" OnClick="btnRemoveNode_Click" />
    </div>
  </form>
</body>
</html>
```

//webform1.aspx.cs

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

namespace Devang
{
  public partial class WebForm1 : System.Web.UI.Page
  {
    protected void treeView1_SelectedNodeChanged(object sender, EventArgs e)
    {
      lblSelectedNode.Text = "Selected Node: " + treeView1.SelectedNode.Text;
    }

    protected void btnAddNode_Click(object sender, EventArgs e)
    {
      if (treeView1.SelectedNode != null && !string.IsNullOrEmpty(txtNodeText.Text))
      {
        TreeNode newNode = new TreeNode(txtNodeText.Text);
        treeView1.SelectedNode.ChildNodes.Add(newNode);
        treeView1.SelectedNode.Expand();
      }
    }

    protected void btnRemoveNode_Click(object sender, EventArgs e)
    {
      if (treeView1.SelectedNode != null && treeView1.SelectedNode.Parent != null)
      {

treeView1.SelectedNode.Parent.ChildNodes.Remove(treeView1.SelectedNode);
        lblSelectedNode.Text = "Selected Node:";
      }
    }
  }
}
```

PRACTICAL 4A: Aim: Create a Registration form to demonstrate use of various Validation controls.



Registration Form

Username:		Username is requiredOnly letters allowed
Age:		You have to be 18 years old
Password:		Password is required
Confirm Password:		Re-enter your passwordPasswords do not match
Email ID:		Email is requiredEnter valid Email
Describe yourself:		Describe in minimum 100 words

// prac4a webform.1.aspx

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

<!DOCTYPE html>

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

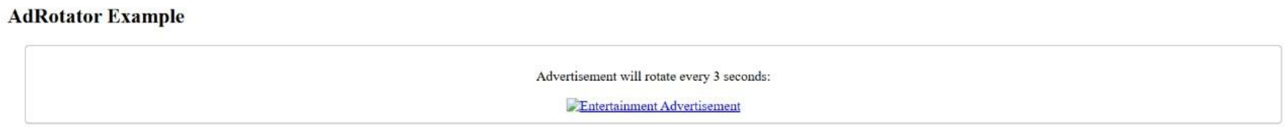
```
<head runat="server">
  <title>Registration Form</title>
  <style>
    .error { color: red; font-weight: bold; }
    .lbl { width: 150px; display: inline-block; }
  </style>
</head>
<body>
  <form id="form1" runat="server">
    <h2>Registration Form</h2>
    <div>
      <span class="lbl">Username:</span>
      <asp:TextBox ID="uname" runat="server"></asp:TextBox>
      <asp:RequiredFieldValidator ID="rfvUsername" runat="server"
        ControlToValidate="uname"
        ErrorMessage="Username is required"
        ForeColor="Red" CssClass="error"></asp:RequiredFieldValidator>
      <asp:RegularExpressionValidator ID="revUsername" runat="server"
        ControlToValidate="uname"
        ErrorMessage="Only letters allowed"
        ValidationExpression="^[a-zA-Z]+$"
        ForeColor="Red" CssClass="error"></asp:RegularExpressionValidator>
    </div>
    <div>
      <span class="lbl">Age:</span>
      <asp:TextBox ID="age" runat="server"></asp:TextBox>
      <asp:RangeValidator ID="rvAge" runat="server"
        ControlToValidate="age"
        ErrorMessage="You have to be 18 years old"
        MinimumValue="18" MaximumValue="150"
        Type="Integer"
        ForeColor="Red" CssClass="error"></asp:RangeValidator>
    </div>
    <div>
      <span class="lbl">Password:</span>
      <asp:TextBox ID="pswd" runat="server" TextMode="Password"></asp:TextBox>
      <asp:RequiredFieldValidator ID="rfvPassword" runat="server"
        ControlToValidate="pswd"
        ErrorMessage="Password is required"
        ForeColor="Red" CssClass="error"></asp:RequiredFieldValidator>
    </div>
    <div>
      <span class="lbl">Confirm Password:</span>
      <asp:TextBox ID="cpwd" runat="server" TextMode="Password"></asp:TextBox>
      <asp:RequiredFieldValidator ID="rfvConfirmPassword" runat="server"
        ControlToValidate="cpwd"
        ErrorMessage="Re-enter your password"
        ForeColor="Red" CssClass="error"></asp:RequiredFieldValidator>
      <asp:CompareValidator ID="cvPasswords" runat="server"
        ControlToValidate="cpwd"
        ControlToCompare="pswd"
        ErrorMessage="Passwords do not match"
        Operator="Equal"
        ForeColor="Red" CssClass="error"></asp:CompareValidator>
    </div>
    <div>
      <span class="lbl">Email ID:</span>
      <asp:TextBox ID="email" runat="server"></asp:TextBox>
```

```
      <asp:RequiredFieldValidator ID="rfvEmail" runat="server"
        ControlToValidate="email"
        ErrorMessage="Email is required"
        ForeColor="Red" CssClass="error"></asp:RequiredFieldValidator>
      <asp:RegularExpressionValidator ID="revEmail" runat="server"
        ControlToValidate="email"
        ErrorMessage="Enter valid Email"
        ValidationExpression="^[a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}$"
        ForeColor="Red" CssClass="error"></asp:RegularExpressionValidator>
    </div>
    <div>
      <span class="lbl">Describe yourself:</span>
      <asp:TextBox ID="desc" runat="server" TextMode="MultiLine" Rows="3" Columns="40"></asp:TextBox>
      <asp:CustomValidator ID="cvDesc" runat="server"
        ControlToValidate="desc"
        ErrorMessage="Describe in minimum 100 words"
        OnServerValidate="cvDesc_ServerValidate"
        ForeColor="Red" CssClass="error"></asp:CustomValidator>
    </div>
    <div>
      <asp:Button ID="btnValidate" runat="server" Text="Validate" />
    </div>
  </form>
</body>
</html>
```

// webform1.aspx.cs

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

public partial class Registration : Page
{
    protected void cvDesc_ServerValidate(object source, ServerValidateEventArgs args)
    {
        if (args.Value == null)
        {
            args.IsValid = false;
            return;
        }
        // Count words (splitting by whitespace)
        string[] words = args.Value.Trim().Split(new char[] { ' ', '\t', '\n', '\r' }, StringSplitOptions.RemoveEmptyEntries);
        args.IsValid = words.Length >= 100;
    }

    protected void Page_Load(object sender, EventArgs e)
    {

    }
}
```

PRACTICAL 4B: Aim: Create Web Form to demonstrate use of Adrotator Control.



AdRotator Example

Advertisement will rotate every 3 seconds:

Entertainment Advertisement

//prac4b webform1.aspx

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

<!DOCTYPE html>

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

<head runat="server">

  <title>AdRotator Example</title>

  <style>

    .ad-container {

      text-align: center;

      margin: 20px;

      border: 2px solid #ccc;

      padding: 10px;

      border-radius: 5px;

    }

  </style>

</head>

<body>

  <form id="form1" runat="server">

    <div>

      <h2>AdRotator Example</h2>

      <asp:ScriptManager ID="ScriptManager1" runat="server" />
      <div class="ad-container">
        <p>Advertisement will rotate every 3 seconds:</p>
        <asp:UpdatePanel ID="UpdatePanel1" runat="server"
UpdateMode="Conditional">
          <ContentTemplate>
            <asp:AdRotator ID="AdRotator1" runat="server"
                           AdvertisementFile="~/Advertisements.xml"
                           OnAdCreated="AdRotator1_AdCreated"
                           Target="_blank" />
          </ContentTemplate>
          <Triggers>
            <asp:AsyncPostBackTrigger ControlID="Timer2" EventName="Tick" />
          </Triggers>
        </asp:UpdatePanel>

        <asp:Timer ID="Timer2" runat="server" Interval="3000" Enabled="true"
OnTick="Timer2_Tick" />
      </div>
    </div>
  </form>
</body>
</html>
```

//   webform1.aspx.cs

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

namespace Devang
{
    public partial class WebForm1 : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            if (!IsPostBack)
            {
                // Set up the AdRotator control
                AdRotator1.AdvertisementFile = "~/Advertisements.xml";
                AdRotator1.AdCreated += AdRotator1_AdCreated;
            }
        }

        protected void AdRotator1_AdCreated(object sender, AdCreatedEventArgs e)
        {
            // Event handler for when an ad is created
            // You can add custom logic here if needed
            // For example, logging which ad was displayed
            if (!string.IsNullOrEmpty(e.AlternateText))
            {
                // Optional: Add any custom processing here
                System.Diagnostics.Debug.WriteLine($"Ad displayed: {e.AlternateText}");
            }
        }
        protected void Timer2_Tick(object sender, EventArgs e)
        {
            // Force the UpdatePanel to refresh, which will cause the AdRotator to display a
new ad
            UpdatePanel1.Update();
        }
    }
}
```

//   webform1.designer.aspx.cs

```
//------------------------------------------------------------------------------
// <auto-generated>
//     This code was generated by a tool.
//
//     Changes to this file may cause incorrect behavior and will be lost if
//     the code is regenerated.
// </auto-generated>
//------------------------------------------------------------------------------

namespace Devang
{


    public partial class WebForm1
```

```
{

    /// <summary>
    /// form1 control.
    /// </summary>
    /// <remarks>
    /// Auto-generated field.
    /// To modify move field declaration from designer file to code-behind file.
    /// </remarks>
    protected global::System.Web.UI.HtmlControls.HtmlForm form1;

    /// <summary>
    /// ScriptManager1 control.
    /// </summary>
    /// <remarks>
    /// Auto-generated field.
    /// To modify move field declaration from designer file to code-behind file.
    /// </remarks>
    protected global::System.Web.UI.ScriptManager ScriptManager1;

    /// <summary>
    /// UpdatePanel1 control.
    /// </summary>
    /// <remarks>
    /// Auto-generated field.
    /// To modify move field declaration from designer file to code-behind file.
    /// </remarks>
    protected global::System.Web.UI.UpdatePanel UpdatePanel1;
```

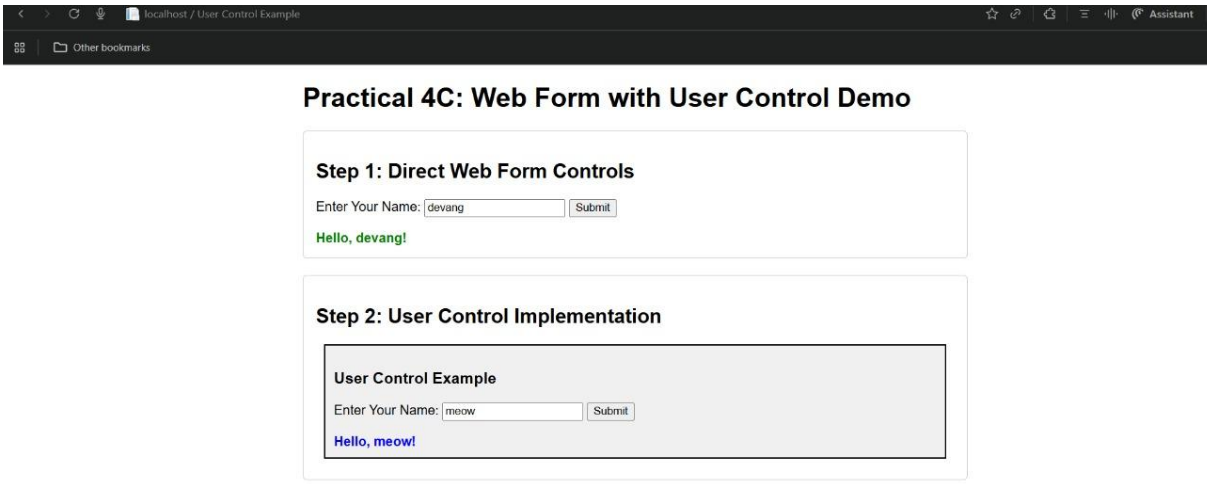
```
        /// <summary>
        /// AdRotator1 control.
        /// </summary>
        /// <remarks>
        /// Auto-generated field.
        /// To modify move field declaration from designer file to code-behind file.
        /// </remarks>
        protected global::System.Web.UI.WebControls.AdRotator AdRotator1;
    }
}
```

// advertisement.xml isme jo pasand ho vo image dalo rotate kregi har 3 sec.

```
<?xml version="1.0" encoding="utf-8"?>
<Advertisements>
  <Ad>

<ImageUrl>https://via.placeholder.com/300x100/FF6B6B/FFFFFF?text=Advertisement+1</ImageUrl>
    <NavigateUrl>https://www.example1.com</NavigateUrl>
    <AlternateText>Technology Advertisement</AlternateText>
    <Keyword>Technology</Keyword>
    <Impressions>80</Impressions>
  </Ad>
```

```
 <Ad>
  <ImageUrl>https://via.placeholder.com/300x100/4ECDC4/FFFFFF?text=Advertisement+2</ImageUrl>
  <NavigateUrl>https://www.example2.com</NavigateUrl>
  <AlternateText>Business Advertisement</AlternateText>
  <Keyword>Business</Keyword>
  <Impressions>50</Impressions>
 </Ad>
 <Ad>
  <ImageUrl>https://via.placeholder.com/300x100/45B7D1/FFFFFF?text=Advertisement+3</ImageUrl>
  <NavigateUrl>https://www.example3.com</NavigateUrl>
  <AlternateText>Entertainment Advertisement</AlternateText>
  <Keyword>Entertainment</Keyword>
  <Impressions>70</Impressions>
 </Ad>
 <Ad>
  <ImageUrl>https://via.placeholder.com/300x100/96CEB4/FFFFFF?text=Advertisement+4</ImageUrl>
  <NavigateUrl>https://www.example4.com</NavigateUrl>
  <AlternateText>Sports Advertisement</AlternateText>
  <Keyword>Sports</Keyword>
  <Impressions>60</Impressions>
 </Ad>
</Advertisements>
```

PRACTICAL 4C: Aim: Create Web Form to demonstrate use User Control



Practical 4C: Web Form with User Control Demo

Step 1: Direct Web Form Controls

Enter Your Name: devang Submit

Hello, devang!

Step 2: User Control Implementation

User Control Example

Enter Your Name: meow Submit

Hello, meow!

//prac4c webform1.aspx

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

<%@ Register Src="~/MyUserControl.ascx" TagPrefix="uc1" TagName="MyUserControl"
%>


<!DOCTYPE html>

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

<head runat="server">

  <title>User Control Example - Practical 4C</title>

  <style>

    body {

      font-family: Arial, sans-serif;

      margin: 20px;
```

```
        }
        .container {
          max-width: 800px;
          margin: 0 auto;
        }
        .form-section {
          margin: 20px 0;
          padding: 15px;
          border: 1px solid #ccc;
          border-radius: 5px;
        }
    </style>
</head>
<body>
    <form id="form1" runat="server">
        <div class="container">
          <h1>Practical 4C: Web Form with User Control Demo</h1>

          <div class="form-section">
            <h2>Step 1: Direct Web Form Controls</h2>
            <p>Enter Your Name:
              <asp:TextBox ID="txtName" runat="server"></asp:TextBox>
              <asp:Button ID="btnSubmit" runat="server" Text="Submit"
OnClick="btnSubmit_Click" />
            </p>
            <asp:Label ID="lblGreeting" runat="server" Font-Bold="true"
ForeColor="Green"></asp:Label>
          </div>

          <div class="form-section">
        <h2>Step 2: User Control Implementation</h2>
        <uc1:MyUserControl ID="MyUserControl1" runat="server" />
      </div>
    </div>
  </form>
</body>
</html>
```

// webform1.aspx.cs

```
using System;
using System.Web.UI;

namespace Devang
{
  public partial class WebForm1 : Page
  {
    protected void Page_Load(object sender, EventArgs e)
    {
      // Page initialization code
      if (!IsPostBack)
      {
        // Any initialization logic can go here
      }
    }

    protected void btnSubmit_Click(object sender, EventArgs e)
    {
            // Code for the main form's button click event
            string name = txtName.Text;
            lblGreeting.Text = "Hello, " + name + "!";
        }
    }
}
```

//   MyUserControl.designer.aspx.cs

```
//------------------------------------------------------------------------------
// <auto-generated>
//     This code was generated by a tool.
//
//     Changes to this file may cause incorrect behavior and will be lost if
//     the code is regenerated.
// </auto-generated>
//------------------------------------------------------------------------------

namespace Devang
{
    public partial class MyUserControl
    {
        /// <summary>
        /// txtName control.
        /// </summary>
        /// <remarks>
        /// Auto-generated field.
        /// To modify move field declaration from designer file to code-behind file.
        /// </remarks>
        protected global::System.Web.UI.WebControls.TextBox txtName;

        /// <summary>
        /// btnSubmit control.
        /// </summary>
        /// <remarks>
        /// Auto-generated field.
        /// To modify move field declaration from designer file to code-behind file.
        /// </remarks>
        protected global::System.Web.UI.WebControls.Button btnSubmit;

        /// <summary>
        /// lblGreeting control.
        /// </summary>
        /// <remarks>
        /// Auto-generated field.
        /// To modify move field declaration from designer file to code-behind file.
        /// </remarks>
        protected global::System.Web.UI.WebControls.Label lblGreeting;
    }
}
```

//MyUserControl.ascx.cs

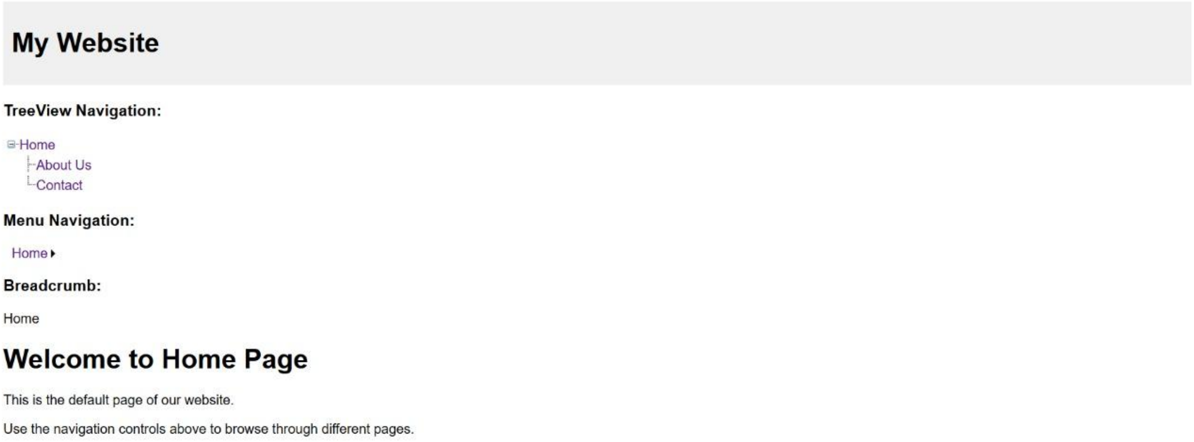
```
//------------------------------------------------------------------------------
// <auto-generated>
//     This code was generated by a tool.
//
```

```
//    Changes to this file may cause incorrect behavior and will be lost if
//    the code is regenerated.
// </auto-generated>
//------------------------------------------------------------------------------

namespace Devang
{
    public partial class MyUserControl
    {
        /// <summary>
        /// txtName control.
        /// </summary>
        /// <remarks>
        /// Auto-generated field.
        /// To modify move field declaration from designer file to code-behind file.
        /// </remarks>
        protected global::System.Web.UI.WebControls.TextBox txtName;

        /// <summary>
        /// btnSubmit control.
        /// </summary>
        /// <remarks>
        /// Auto-generated field.
        /// To modify move field declaration from designer file to code-behind file.
        /// </remarks>
        protected global::System.Web.UI.WebControls.Button btnSubmit;

        /// <summary>
        /// lblGreeting control.
        /// </summary>
        /// <remarks>
        /// Auto-generated field.
        /// To modify move field declaration from designer file to code-behind file.
        /// </remarks>
        protected global::System.Web.UI.WebControls.Label lblGreeting;
    }
}
```

/// MyUserControl.ascx

```
<%@ Control Language="C#" AutoEventWireup="true"
CodeBehind="MyUserControl.ascx.cs" Inherits="Devang.MyUserControl" %>


<div style="border: 2px solid #000; padding: 10px; margin: 10px; background-color:
#f0f0f0;">
    <h3>User Control Example</h3>
    <p>Enter Your Name:
        <asp:TextBox ID="txtName" runat="server"></asp:TextBox>
        <asp:Button ID="btnSubmit" runat="server" Text="Submit"
OnClick="btnSubmit_Click" />
    </p>
    <asp:Label ID="lblGreeting" runat="server" Font-Bold="true"
ForeColor="Blue"></asp:Label>
</div>
```

PRACTICAL 5A AIM: Create Web Form to demonstrate use of Website Navigation controls.



My Website

TreeView Navigation:

⊟Home
├About Us
└Contact

Menu Navigation:

Home ▸

Breadcrumb:

Home

Welcome to Home Page

This is the default page of our website.

Use the navigation controls above to browse through different pages.

//web.sitemap

```
<?xml version="1.0" encoding="utf-8" ?>

<siteMap xmlns="http://schemas.microsoft.com/AspNet/SiteMap-File-1.0" >

        <siteMapNode url="~/Default.aspx" title="Home" description="Home Page">

                <siteMapNode url="~/About.aspx" title="About Us" description="About Us
Page" />

                <siteMapNode url="~/Contact.aspx" title="Contact"
description="Contact Page" />

        </siteMapNode>

</siteMap>
```

//Site.master

```
<%@ Master Language="C#" AutoEventWireup="true" CodeBehind="Site.master.cs"
Inherits="WebApplication7.Site" %>

<!DOCTYPE html>

<html>
<head runat="server">
    <title></title>
    <asp:ContentPlaceHolder ID="head" runat="server">
    </asp:ContentPlaceHolder>
    <style>
        body { font-family: Arial, sans-serif; margin: 0; padding: 20px; }
        .header { background-color: #f0f0f0; padding: 10px; margin-bottom: 20px; }
        .navigation { margin-bottom: 20px; }
        .content { min-height: 400px; }
    </style>
</head>
<body>
    <form id="form1" runat="server">
        <div class="header">
            <h1>My Website</h1>
        </div>

        <div class="navigation">
            <asp:SiteMapDataSource ID="SiteMapDataSource1" runat="server" />

            <h3>TreeView Navigation:</h3>
            <asp:TreeView ID="TreeView1" runat="server"
                DataSourceID="SiteMapDataSource1"
                ShowLines="true">
            </asp:TreeView>

            <h3>Menu Navigation:</h3>
            <asp:Menu ID="Menu1" runat="server"
                DataSourceID="SiteMapDataSource1"
                Orientation="Horizontal"
                StaticMenuItemStyle-HorizontalPadding="10px">
            </asp:Menu>

            <h3>Breadcrumb:</h3>
            <asp:SiteMapPath ID="SiteMapPath1" runat="server" />
        </div>

        <div class="content">
            <asp:ContentPlaceHolder ID="MainContent" runat="server">
            </asp:ContentPlaceHolder>
        </div>
    </form>
</body>
</html>
```

//site.master.cs

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


namespace WebApplication7
{
   public partial class Site : System.Web.UI.MasterPage
   {
      protected void Page_Load(object sender, EventArgs e)
      {


      }
   }
}
```

//default.aspx

```
<%@ Page Title="Home" Language="C#" MasterPageFile="~/Site.Master"
AutoEventWireup="true" CodeBehind="Default.aspx.cs"
Inherits="WebApplication7._Default" %>


<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">
</asp:Content>


<asp:Content ID="Content2" ContentPlaceHolderID="MainContent" runat="server">
   <h1>Welcome to Home Page</h1>
   <p>This is the default page of our website.</p>
   <p>Use the navigation controls above to browse through different pages.</p>
</asp:Content>
```