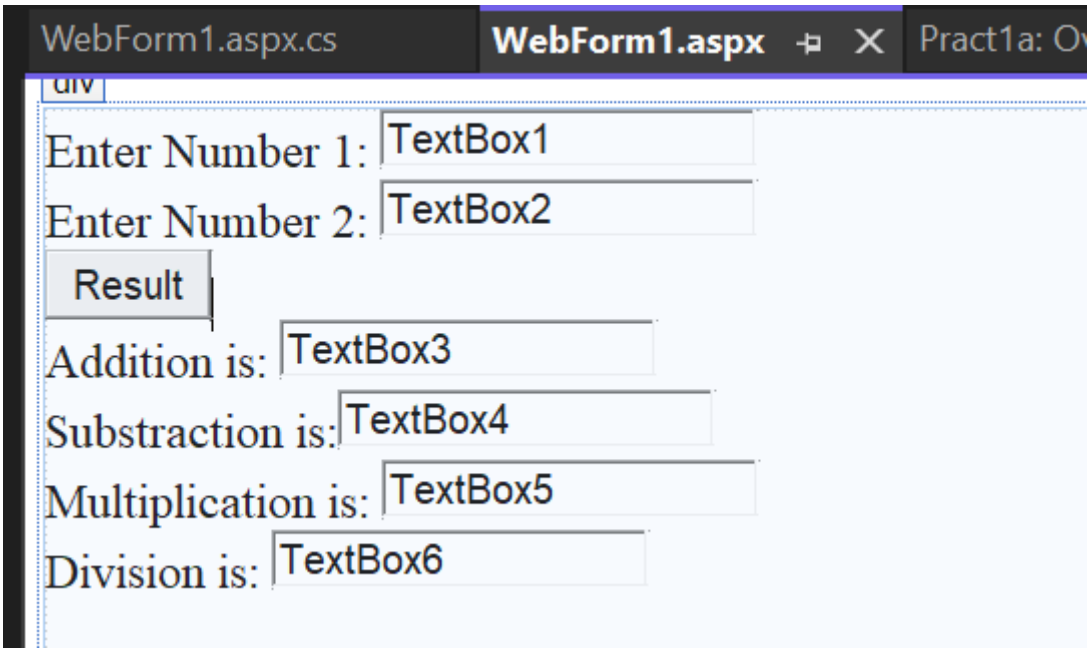
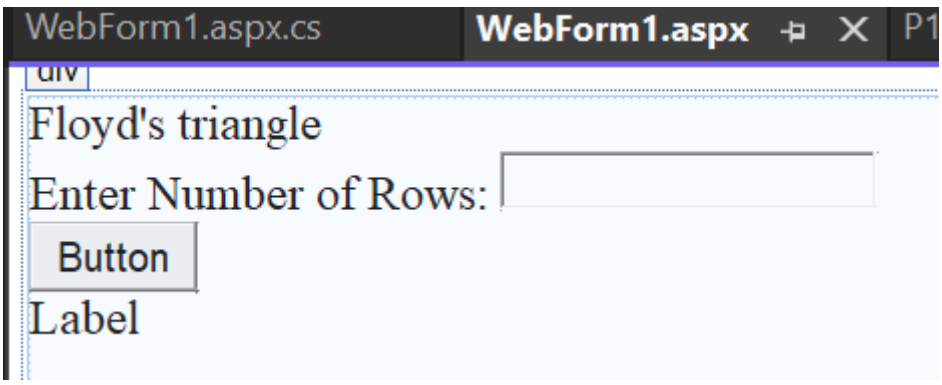
Q1a - Create an application to print on screen the output of adding, subtracting, multiplying and dividing two numbers entered by the user in C#.




```
protected void Button1_Click(object sender, EventArgs e)
{
    double number1 = Convert.ToDouble(TextBox1.Text);
    double number2 = Convert.ToDouble(TextBox2.Text);

    double add = number1 + number2;
    double sub = number1 - number2;
    double mul = number1 * number2;
    double div = number2 == 0 ? double.NaN : number1 / number2;

    TextBox3.Text = add.ToString();
    TextBox4.Text = sub.ToString();
    TextBox5.Text = mul.ToString();
    TextBox6.Text = div.ToString();
```

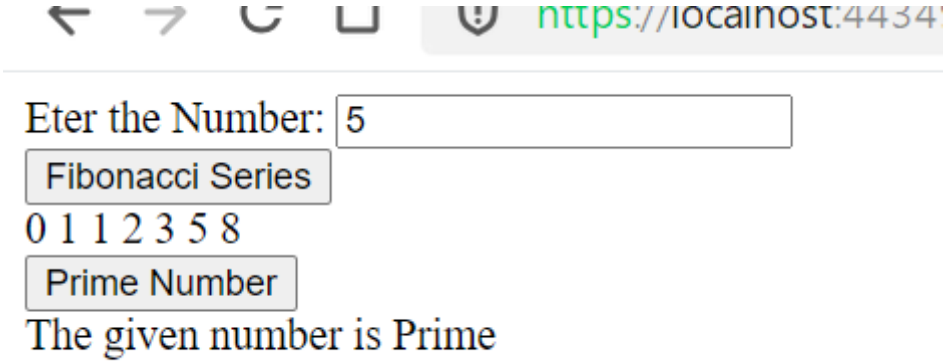
Q1b - Create an application to print Floyd's triangle till n rows in C#




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

```
{
    int rows = Convert.ToInt32(TextBox1.Text);
    int number = 1;
    StringBuilder sb = new StringBuilder();

    for (int i = 1; i <= rows; i++)
    {
        for (int j = 1; j <= i; j++)
        {
            sb.Append(number + " ");
            number++;
        }
        sb.AppendLine();
    }

    Label1.Text = sb.ToString();
```

Q1c- Create an application to demonstrate following operations i. Generate Fibonacci series. ii. Test for prime numbers.




```
    protected void Button1_Click(object sender, EventArgs e)
    {
        int a = 0, b = 1, c, i, n;
        Label1.Text = a.ToString() + " " + b.ToString();
        n = Convert.ToInt32(TextBox1.Text);

        for (i = 1; i <= n; ++i)
        {
            c = a + b;
            Label1.Text = Label1.Text + " " + c.ToString();
            a = b;
            b = c;
        }
    }

    protected void Button2_Click(object sender, EventArgs e)
    {
        int n, i, s = 0;
        n = Convert.ToInt32(TextBox1.Text);
```

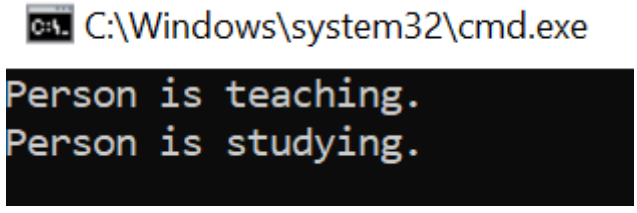
```
        if (n == 0 || n == 1)
            s = 1;

        for (i = 2; i <= n / 2; ++i)
        {
            if (n % i == 0)
            {
                s = 1;
                break;
            }
        }

        if (s == 0)
            Label2.Text = "The given number is Prime";
        else
            Label2.Text = "The given number is Not Prime";
    }
}
}
```

Slip- (a) Write a program in C# to demonstrate multiple inheritance using interfaces

Console App (.NET Framework)

```
using System;


// First Interface
interface ITeacher
{
    void Teach();
}


// Second Interface
interface IStudent
{
    void Study();
}


// Class implementing multiple interfaces
class Person : ITeacher, IStudent
```

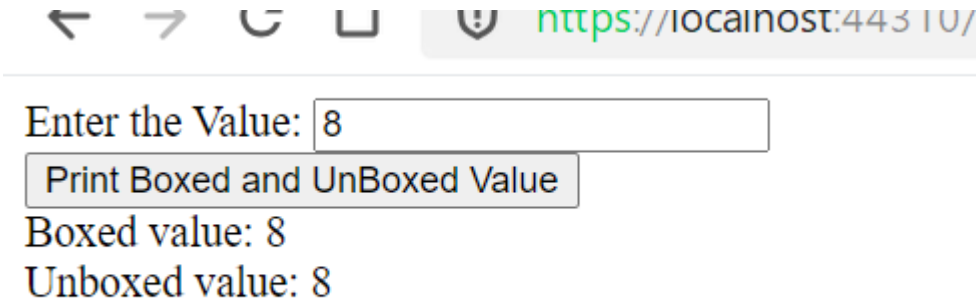
```
{
    public void Teach()
    {
        Console.WriteLine("Person is teaching.");
    }

    public void Study()
    {
        Console.WriteLine("Person is studying.");
    }
}

class Program
{
    static void Main(string[] args)
    {
        // Object of Person class
        Person p = new Person();

        // Calling methods of both interfaces
        p.Teach();
        p.Study();

        Console.ReadLine();
    }
}
```



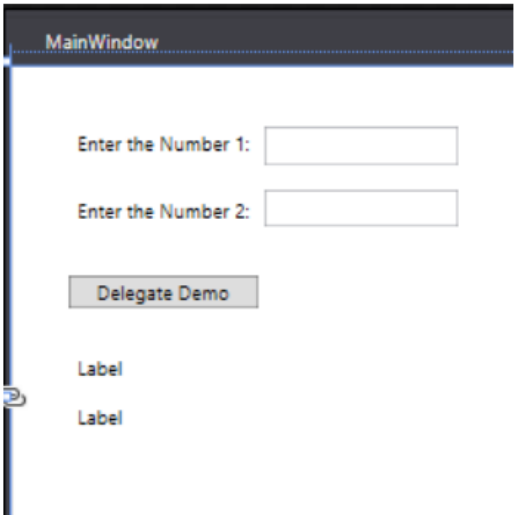
(b) Create a Registration form to demonstrate use of various validation control

Q2(a) - Create a simple application to demonstrate the concepts boxing and unboxing.

```
protected void Button1_Click(object sender, EventArgs e)
{
   int valueType = Convert.ToInt32(TextBox1.Text);
   object boxed = valueType;
   Label1.Text = "Boxed value: " + boxed;
   int unboxed = (int)boxed;
   Label2.Text = "Unboxed value: " + unboxed;
}
```



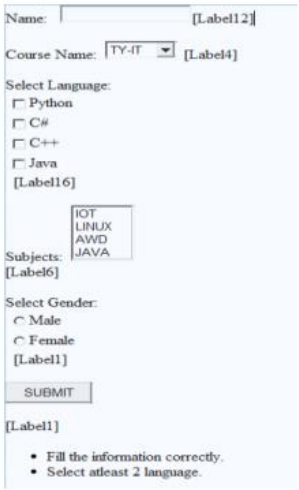
Q2(b)- Create a simple application to perform addition and subtraction using delegate.




```
public delegate int MathOperation(int a, int b);

public partial class MainWindow : Window

{

public MainWindow()

{

InitializeComponent();

}
```

```
int Add(int a, int b)
{
return a + b;
}
int Subtract(int a, int b)
{
return a - b;
}
private void Button_Click(object sender, RoutedEventArgs e)
{
int x = Convert.ToInt32(TextBox1.Text);
int y = Convert.ToInt32(TextBox2.Text);
MathOperation add = new MathOperation(Add);
MathOperation subtract = new MathOperation(Subtract);
Label1.Content = add(x, y).ToString();
Label2.Content = subtract(x, y).ToString();
}
}
```

Q3(a)- Create a simple web page with various server controls to demonstrate setting and use of their properties. (Example : AutoPostBack)

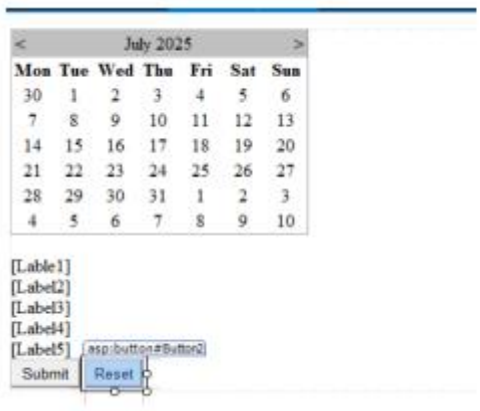



```
namespace WebApplication1
{
public partial class WebForm1 : System.Web.UI.Page
```

```
{
protected void Page_Load(object sender, EventArgs e)
{
}
protected void DropDownList1_SelectedIndexChanged(object sender, EventArgs e)
{
Label4.Text = DropDownList1.SelectedValue;
}
protected void TextBox1_TextChanged(object sender, EventArgs e)
{
String str;
str = TextBox1.Text;
Label2.Text = str;
}
protected void CheckBoxList1_SelectedIndexChanged(object sender, EventArgs e)
{
Label6.Text = "";
foreach(ListItem x in CheckBoxList1.Items)
{
if(x.Selected)
{
Label6.Text += "<br>" + x.Value;
}
}
}
protected void RadioButtonList1_SelectedIndexChanged(object sender, EventArgs e)
{
Label10.Text = RadioButtonList1.SelectedValue;
}
protected void Button1_Click(object sender, EventArgs e)
{
```

```
 Label11.Text = "Record Submitted";
 }
 protected void ListBox1_SelectedIndexChanged(object sender, EventArgs e)
{
 Label8.Text = "";
 foreach(ListItem x in ListBox1.Items)
 {
 if(x.Selected)
 {
 Label8.Text += x.Value;
 }
 }
 }
 }
}
```

Q3(b)- Create a simple application to demonstrate your vacation using calendar control.




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

```
protected void Button1_Click(object sender, EventArgs e)
{
{
Calendar1.Caption = "DISHA KUNDAR";
Calendar1.FirstDayOfWeek = FirstDayOfWeek.Sunday;
Calendar1.NextPrevFormat = NextPrevFormat.FullMonth;
Calendar1.TitleFormat = TitleFormat.Month;
Label2.Text = "Todays Date" + Calendar1.SelectedDate.ToShortDateString();
Label3.Text = "Diwali Vacation Start: 10-23-2025";
TimeSpan d = new DateTime(2025, 10, 17) - Calendar1.SelectedDate;
Label4.Text = "Days Remaining For Diwali Vacation:" + d.Days.ToString();
TimeSpan d1 = new DateTime(2025, 12, 31) - Calendar1.SelectedDate;
Label5.Text = "Days Remaining for New Year:" + d1.Days.ToString();
if (Calendar1.SelectedDate.ToShortDateString() == "10-17-2025")
Label3.Text = "<b>Diwali Festival Start</b>";
if (Calendar1.SelectedDate.ToShortDateString() == "10-23-2025")
Label3.Text = "<b>Diwali Festival End</b>";
}
}
protected void Button2_Click(object sender, EventArgs e)
{
{
Label1.Text = "";
Label2.Text = "";
Label3.Text = "";
Label4.Text = "";
Label5.Text = "";
Calendar1.SelectedDates.Clear();
}
}
```

Q3(c)- Demonstrate the use of Treeview operations on the web form.

Xmal file:

```
<?xml version="1.0" encoding="utf-8" ?>
<students>
<student sid="1" sname="Disha" age="19" ></student>
<student sid="2" sname="Swapnali" age="19" ></student>
<student sid="3" sname="Shaista" age="21" ></student>
</students>
```


```
namespace Practical3C
{
 public partial class WebForm1 : System.Web.UI.Page
 {
 protected void Page_Load(object sender, EventArgs e)
 {
 }

 protected void DataList1_SelectedIndexChanged(object sender, EventArgs e)
 {

 }
 protected void TreeView1_SelectedNodeChanged(object sender, EventArgs e)
 {
 Label1.Text = TreeView1.SelectedNode.Text;
```

```
    }
    }
}
```