Practical No. 1

Aim: Write C# programs for understanding C# basics involving:

a. Variables and Data Types **b. Object-Based Manipulation**

c. Conditional Logic **d. Loops** **e. Methods**

a. Variables and Data Types

Program:

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

namespace ConsoleApplication1
{
   class Program
   {
      static void Main(string[] args)
      {
         int a = 20;
         bool b = true;
         double c = 5.5D;
         float d = 5.5F;
         string val = "Hello World";
         Console.WriteLine("Integer: " + a);
         Console.WriteLine("Boolean Value: " + b);
         Console.WriteLine("Decimal Value: " + c);
         Console.WriteLine("Float Value: " + d);
         Console.WriteLine("String Value: " + val);
         Console.ReadKey();
      }
   }
}
```

---

b. Object-Based Manipulation

Program:

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

namespace ConsoleApplication1
{
   class Program
   {
      static void Main(string[] args)
      {
         string mystring;
         int a = 100;
         Console.WriteLine("Convert Number to String");
         mystring = a.ToString();
```

```
Console.WriteLine("String is " + mystring);

string s = "    This is test string";

Console.WriteLine("\nBefore Trim() Method: " + s);
s = s.Trim();
Console.WriteLine("\nAfter Trim() Method: " + s);

s = s.Substring(0, 4);
Console.WriteLine("\nSubstring() Method: " + s);

s = s.ToUpper();
Console.WriteLine("\nUppercase String: " + s);

s = s.Replace("IS", "AT");
Console.WriteLine("\nReplace String: " + s);

int length = s.Length;
Console.WriteLine("\nLength of String is: " + length);

Console.WriteLine("\n**********************************");

Console.WriteLine("\nDateTime Object");
DateTime myDate = DateTime.Now;
Console.WriteLine("Today's date is: " + myDate);

myDate = myDate.AddDays(100);
Console.WriteLine("\nAfter 100 Days the Date is: " + myDate);

string dateString = myDate.Year.ToString();
Console.WriteLine("\nYear in String is: " + dateString);

DateTime myDate1 = DateTime.Now;
DateTime myDate2 = DateTime.Now.AddHours(3000);
Console.WriteLine("\nDate 1 : " + myDate1);
Console.WriteLine("\nDate 2 : " + myDate2);

TimeSpan difference;
difference = myDate2.Subtract(myDate1);
Console.WriteLine("\nDifference between 2 Dates: " + difference.Days.ToString()+" Days");

double numberOfMinutes;
numberOfMinutes = difference.TotalMinutes;
Console.WriteLine("\nNumber of Minutes: " + numberOfMinutes);

Console.WriteLine("\n**********************************");

Console.WriteLine("\nThe Array Type:");
int[] myArray = { 1, 2, 3, 4, 5 };
int numberOfElements;
numberOfElements = myArray.Length;
Console.WriteLine("\nTotal Elements in array:" + numberOfElements);
```

```
        Console.ReadKey();
      }
   }
}
```

c. Conditional Logic

1. If…Else Condition

Program:

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

namespace ConsoleApplication1
{
   class Program
   {
      static void Main(string[] args)
      {
         double percentage;
         Console.WriteLine("Enter your Percetage: ");
         percentage = Convert.ToDouble(Console.ReadLine());

         if (percentage >= 80.00)
         {
            Console.WriteLine("You get 'O' Grade.");
         }
         else if(percentage <= 79.99 && percentage >= 75.00)
         {
            Console.WriteLine("You get 'A' Grade.");
         }
         else if (percentage <= 74.99 && percentage >= 70.00)
         {
            Console.WriteLine("You get 'B' Grade.");
         }
         else if (percentage <= 69.99 && percentage >= 60.00)
         {
            Console.WriteLine("You get 'C' Grade.");
         }
         else if (percentage <= 59.99 && percentage >= 50.00)
         {
            Console.WriteLine("You get 'D' Grade.");
         }
         else if (percentage <= 49.99 && percentage >= 40.00)
         {
            Console.WriteLine("You get 'E' Grade.");
         }
         else
         {
            Console.WriteLine("You get 'F' Grade.");
         }
         Console.ReadKey();
```

```
    }
  }
}
```

2. Switch Case

Program:

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

namespace ConsoleApplication1
{
  class Program
  {
    static void Main(string[] args)
    {
      char op;
      int first, second, result;

      Console.Write("Enter first number: ");
      first = Convert.ToInt32(Console.ReadLine());
      Console.Write("\nEnter second number: ");
      second = Convert.ToInt32(Console.ReadLine());
      Console.Write("\nEnter operator (+, -, *, /): ");
      op = (char)Console.Read();

      switch (op)
      {
        case '+':
          result = first + second;
          Console.WriteLine("\n" + first + " + " + second + " = " + result);
          break;

        case '-':
          result = first - second;
          Console.WriteLine("\n" + first + " - " + second + " = " + result);
          break;

        case '*':
          result = first * second;
          Console.WriteLine("\n" + first + " * " + second + " = " + result);
          break;

        case '/':
          result = first / second;
          Console.WriteLine("\n" + first + " / " + second + " = " + result);
          break;

        default:
          Console.WriteLine("Invalid Operator");
          break;
```

```
        }
        Console.ReadKey();
    }
  }
}
```

d. Loops

1. For Loop

Program:

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

namespace ConsoleApplication1
{
  class Program
  {
    static void Main(string[] args)
    {
      int n = 10, sum = 0;
      for (int i = 1; i <= n; i++)
      {
        sum = sum + i;
      }
      Console.WriteLine("Sum of first {0} natural numbers = {1}", n, sum);
      Console.ReadKey();
    }
  }
}
```

2. While Loop

Program:

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

namespace ConsoleApplication1
{
  class Program
  {
    static void Main(string[] args)
    {
      int n = 10, sum = 0, i = 1;

      while (i <= n)
      {
        sum = sum + i;
        i++;
      }
```

```
        Console.WriteLine("Sum of first {0} natural numbers = {1}", n, sum);
        Console.ReadKey();
    }
  }
}
```

3. Foreach Loop

Program:

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

namespace ConsoleApplication1
{
  class Program
  {
    static void Main(string[] args)
    {
      int sum = 0;
      int[] n = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 };

      foreach (int number in n)
      {
        sum = sum + number;
      }
      Console.WriteLine("Sum of first {0} natural numbers = {1}", n.Length.ToString(), sum);
      Console.ReadKey();
    }
  }
}
```

e. Methods

1. Method Overloading

Program:

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

namespace ConsoleApplication1
{
  class Program
  {
    void calculate(int r)
    {
      double area,pi=3.14;
      area = pi * r * r;
      Console.WriteLine("Area of Circle: " + area);
    }
    void calculate(int l, int b)
```

```
    {
        double area;
        area = l * b;
        Console.WriteLine("Area of Rectangle " + area);
    }
    static void Main(string[] args)
    {
        Program p = new Program();

        int r, l, b;

        Console.WriteLine("Enter radius: ");
        r = Convert.ToInt32(Console.ReadLine());
        p.calculate(r);
        Console.WriteLine("\n-----------------------------------\n");

        Console.WriteLine("Enter Length: ");
        l = Convert.ToInt32(Console.ReadLine());
        Console.WriteLine("Enter Breadth: ");
        b = Convert.ToInt32(Console.ReadLine());
        p.calculate(l, b);
        Console.WriteLine("\n-------------------------------------\n");

        Console.ReadKey();
    }
  }
}
```

Practical No. 2

Aim: Write C# programs for Object oriented concepts of C# such as:

a. Program using classes **b. Constructor and Function Overloading**

c. Inheritance **d. Namespaces**

a. Program using Classes

Program:

```
using System;
namespace sycs
{
   class Employee
   {
      public string name;
      public void work(string work)
      {
         Console.WriteLine("Work: " + work);
      }
   }

   class EmployeeDrive
   {
      static void Main(string[] args)
      {
         // create Employee object
         Employee e1 = new Employee();

         Console.WriteLine("Employee 1");

         // set name of the Employee
         e1.name = "Gloria";
         Console.WriteLine("Name: " + e1.name);

         //call method of the Employee
         e1.work("Coding");

         Console.ReadLine();

      }
   }
}
```

---

b. Constructor and Function Overloading

1. Constructor Overloading

Program:

```
using System;
namespace sycs
{
   class gamescore
   {
      string user;
      int age;
      //Default Constructor
```

```
    public gamescore()
    {
        user = "John";
        age = 25;
        Console.WriteLine("\nPrevious User {0} and he was {1} year old", user, age);
    }

    //Parameterized Constructor
    public gamescore(string name, int age1)
    {
        user = name;
        age = age1;
        Console.WriteLine("\nCurrent User {0} and he is {1} year old", user, age);
    }
  }

  class sycs
  {
    static void Main(string[] args)
    {
        //Default Constructor Called
        gamescore gs = new gamescore();


        //Overloaded Constructor.
        gamescore gs1 = new gamescore("Ram", 30);
        Console.ReadLine();
    }

  }
}
```

Output:

Previous User John and he was 25 year old

Current User Ram and he is 30 year old

2. Function Overloading

Program:

```
using System;
namespace sycs
{
  class shape
  {
    public void Area(int side)
    {
        int square_area = side * side;
        Console.WriteLine("\nThe Area of Square is :" + square_area);
    }
    public void Area(int length, int breadth)
    {
```

```
            int rect_area = length * breadth;
            Console.WriteLine("\nThe Area of Rectangle is :" + rect_area);
        }

        public void Area(double radius)
        {
            double circle_area = 3.14 * radius * radius;
            Console.WriteLine("\nThe Area of Circle is :" + circle_area);
        }
    }
    class sycs
    {
        static void Main(string[] args)
        {
            shape s = new shape();
            s.Area(10);
            s.Area(10, 20);
            s.Area(10.8);
            Console.ReadKey();
        }
    }
}
```

---

c. Inheritance:

Program:

```
using System;
namespace sycs
{
    class sycs
    {
        static void Main(string[] args)
        {
            Scooter sc = new Scooter();
            sc.ScooterType();

            Car c = new Car();
            c.CarType();

            Console.ReadKey();
        }
    }

    //Creating Base Class
    class Tyre
    {
        protected void TyreType()
        {
            Console.WriteLine("This is Tubeless Tyre");
        }
    }
    //Creating Child Class
    class Scooter : Tyre
```

```
    {
        public void ScooterType()
        {
            Console.WriteLine("\nScooter Color is Red");
            TyreType();
        }
    }
    //Creating Child Class
    class Car : Tyre
    {
        public void CarType()
        {
            Console.WriteLine("\n\nCar Type : Ferrari");
            TyreType();
        }
    }
}
```

Output:
Scooter Color is Red
This is Tubeless Tyre

Car Type : Ferrari
This is Tubeless Tyre

---

d. Namespacses:

1. Example of namespace in C# where one namespace program accesses another namespace program.

Program:

```
using System;
namespace First
{
    public class Hello
    {
        public void sayHello()
        {
            Console.WriteLine("Hello First Namespace");
        }
    }
}
namespace Second
{
    public class Hello
    {
        public void sayHello()
        {
            Console.WriteLine("Hello Second Namespace");
        }
    }
}
public class TestNamespace
```

```
{
   public static void Main(String[] args)
   {
      First.Hello h1 = new First.Hello();
      Second.Hello h2 = new Second.Hello();
      h1.sayHello();
      h2.sayHello();
   }
}
```

2. Example of namespace where we are using "using" keyword so that we don't have to use a complete name for accessing a namespace program.

Program:

```
using System;
using First;
using Second;
namespace First
{
   public class Hello
   {
      public void sayHello()
      {
         Console.WriteLine("Hello Namespace");
      }
   }
}
namespace Second
{
   public class Welcome
   {
      public void sayWelcome()
      {
         Console.WriteLine("Welcome Namespace");
      }
   }
}
public class TestNamespace
{
   public static void Main(String[] args)
   {
      Hello h1 = new Hello();
      Welcome w1 = new Welcome();
      h1.sayHello();
      w1.sayWelcome();
   }
}
```

Practical No. 3
Aim: Design ASP.NET Pages with Server Controls

Default.aspx Page:

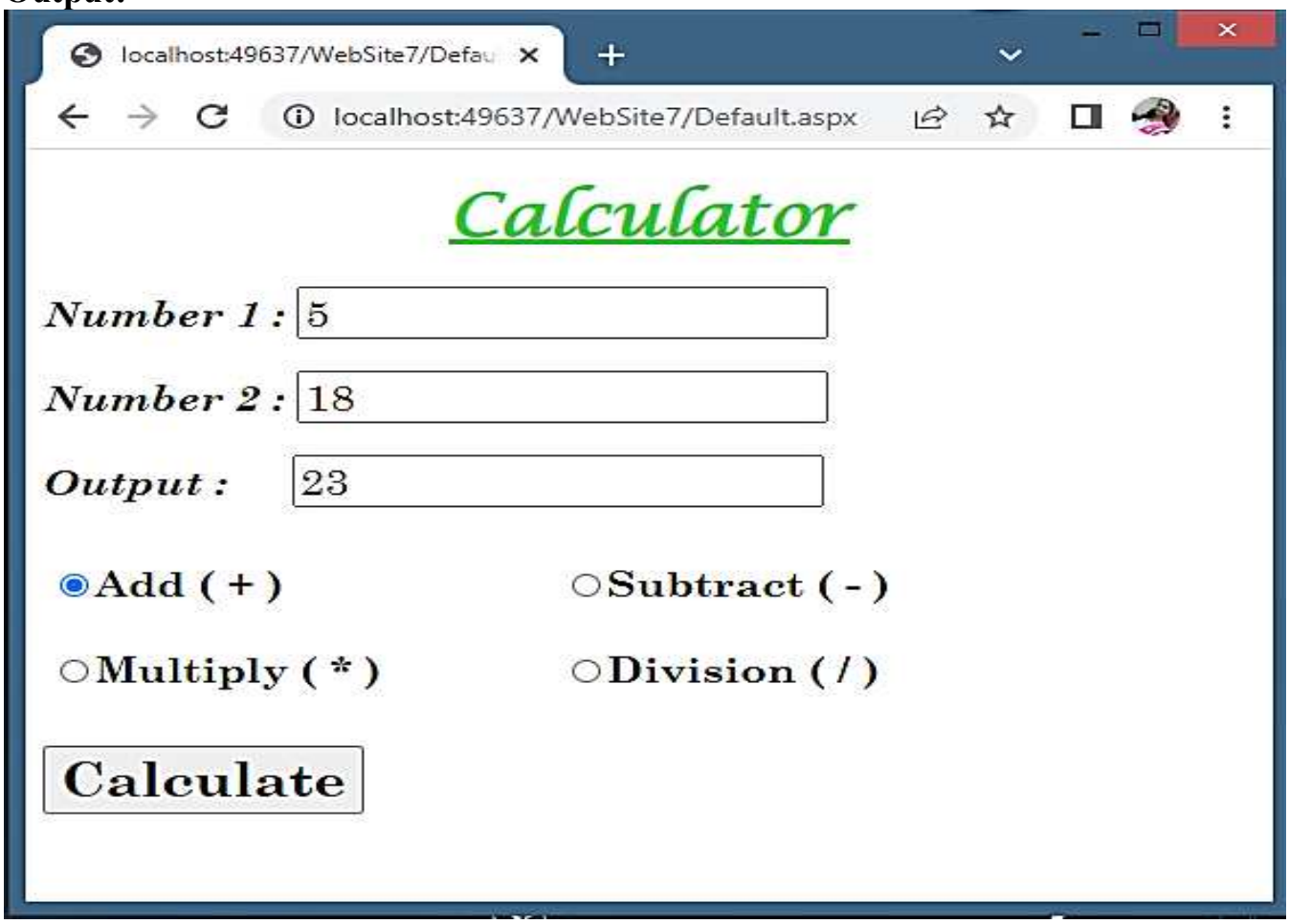
```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Default.aspx.cs" Inherits="_Default" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">            <title></title>            </head>
<body>
   <form id="form1" runat="server">
   <div align="left">
   <p align="center">
   <asp:Label ID="Label1" runat="server" Text="Calculator" Font-Bold="True" Font-Italic="True"
Font-Names="Lucida Calligraphy" Font-Size="30pt" Font-Underline="True" ForeColor="#33CC33"> </asp:Label>
   </p>
      <asp:Label ID="Label2" runat="server" Text="Number 1 : " Font-Bold="True" Font-Italic="True"
Font-Names="Century Schoolbook" Font-Size="20pt"> </asp:Label>
      <asp:TextBox ID="TextBox1" runat="server" Font-Names="Century Schoolbook" Font-Size="20pt">
</asp:TextBox>
      <br /><br />
      <asp:Label ID="Label3" runat="server" Text="Number 2 : " Font-Bold="True" Font-Italic="True"
Font-Names="Century Schoolbook" Font-Size="20pt"> </asp:Label>
      <asp:TextBox ID="TextBox2" runat="server" Font-Names="Century Schoolbook" Font-Size="20pt">
</asp:TextBox>
      <br /><br />
      <asp:Label ID="Label4" runat="server" Text="Output : " Font-Bold="True" Font-Italic="True"
Font-Names="Century Schoolbook" Font-Size="20pt"> </asp:Label>
             
      <asp:TextBox ID="TextBox3" runat="server" ReadOnly="True" Font-Names="Century Schoolbook"
Font-Size="20pt"> </asp:TextBox>
      <br /><br />
      <asp:RadioButtonList ID="RadioButtonList1" runat="server" Font-Bold="True"
         Font-Names="Century Schoolbook" Font-Size="20pt">
         <asp:ListItem Value="Add">Add ( + )</asp:ListItem>
         <asp:ListItem Value="Subtract">Subtract ( - )</asp:ListItem>
         <asp:ListItem Value="Multiply">Multiply ( * )</asp:ListItem>
         <asp:ListItem Value="Division">Division ( / )</asp:ListItem>
      </asp:RadioButtonList>
      <br />
      <asp:Button ID="Button1" runat="server" Text="Calculate" Font-Bold="True"
         Font-Names="Century Schoolbook" Font-Size="20pt" onclick="Button1_Click" />

   </div>
```

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

Default.aspx.cs Page:

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

      string s;
      s = RadioButtonList1.SelectedValue.ToString();
      if (s == "Add")
      {
         output = number1 + number2;
         TextBox3.Text = output.ToString();
      }
      else if(s == "Subtract")
      {
         output = number1 - number2;
         TextBox3.Text = output.ToString();
      }
      else if (s == "Multiply")
      {
         output = number1 * number2;
         TextBox3.Text = output.ToString();
      }
      else if (s == "Division")
      {
         output = number1 / number2;
         TextBox3.Text = output.ToString();
      }
   }
}
```

Output:


localhost:49637/WebSite7/Defau
localhost:49637/WebSite7/Default.aspx
Calculator
Number 1 : 5
Number 2 : 18
Output : 23
Add (+)
Subtract (-)
Multiply (*)
Division (/)
Calculate

Practical No. 4

Aim: Design ASP.NET Pages with Web controls and demonstrate the use of AutoPostBack.

Default.aspx Page:

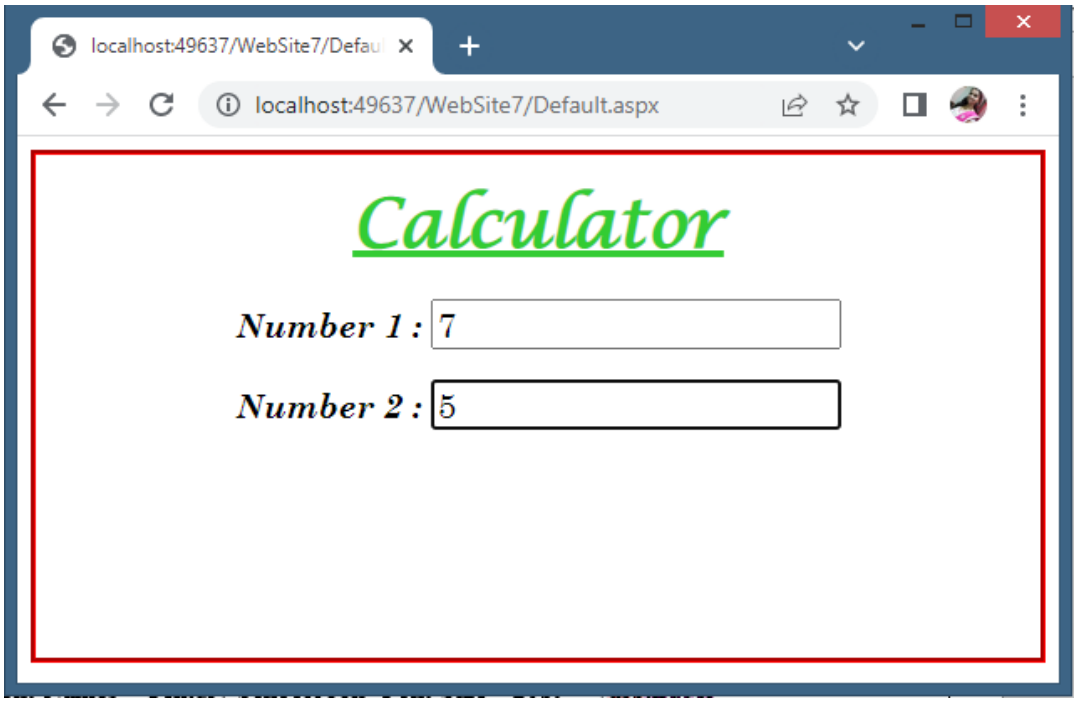
```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Default.aspx.cs" Inherits="_Default" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">      <title></title>         </head>
<body>
   <form id="form1" runat="server">
   <div align="center" style="border: medium groove #FF0000; width: 600px; height: 300px;">
   <p align="center">
   <asp:Label ID="Label1" runat="server" Text="Calculator" Font-Bold="True" Font-Italic="True" Font-
Names="Lucida Calligraphy" Font-Size="30pt" Font-Underline="True" ForeColor="#33CC33"></asp:Label>
   </p>
      <asp:Label ID="Label2" runat="server" Text="Number 1 : " Font-Bold="True"
         Font-Italic="True" Font-Names="Century Schoolbook" Font-Size="15pt"></asp:Label>
      <asp:TextBox ID="TextBox1" runat="server" Font-Names="Century Schoolbook"
         Font-Size="15pt"></asp:TextBox>
      <br /><br />
      <asp:Label ID="Label3" runat="server" Text="Number 2 : " Font-Bold="True"
         Font-Italic="True" Font-Names="Century Schoolbook" Font-Size="15pt"></asp:Label>
      <asp:TextBox ID="TextBox2" runat="server" Font-Names="Century Schoolbook"
         Font-Size="15pt" AutoPostBack="True" ontextchanged="TextBox2_TextChanged"></asp:TextBox>
      <br /><br />
      <asp:Label ID="Label4" runat="server" Text="" Font-Bold="True"
         Font-Italic="True" Font-Names="Century Schoolbook" Font-Size="15pt"></asp:Label>
   </div>
   </form>
</body>
</html>
```

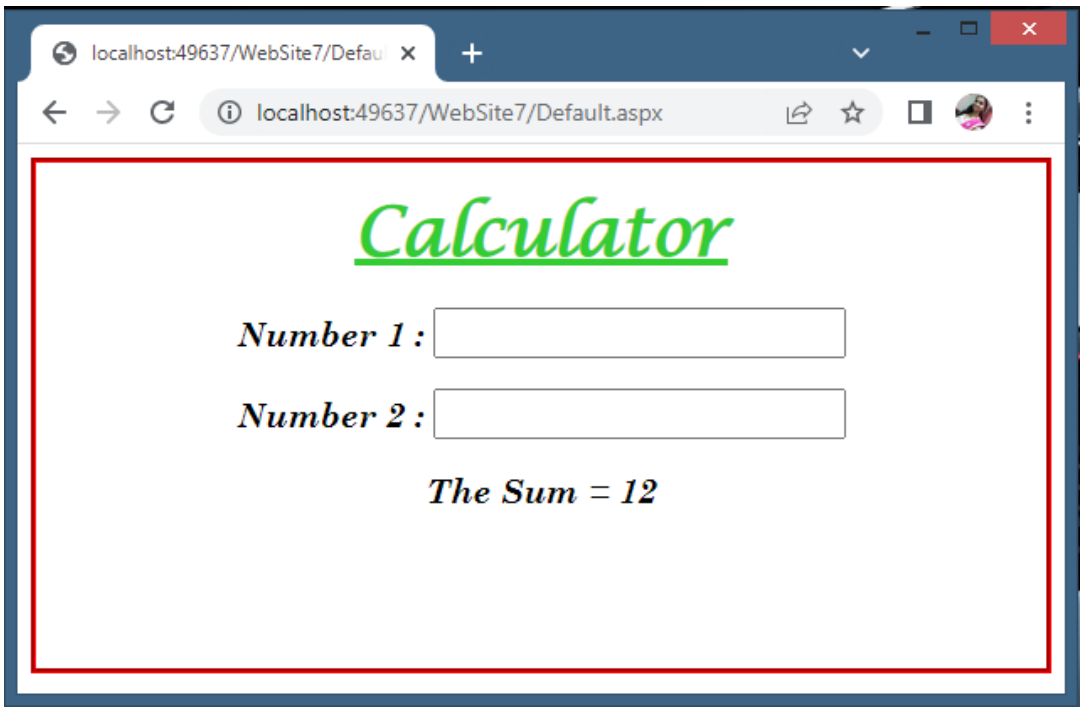
Default.aspx.CS Page:

```
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)
```

```
    {
    }
    protected void TextBox2_TextChanged(object sender, EventArgs e)
    {
        int sum = Convert.ToInt32(TextBox1.Text) + Convert.ToInt32(TextBox2.Text);
        Label4.Text = "The Sum = " + sum.ToString();
        TextBox1.Text = "";
        TextBox2.Text = "";
    }
}
```

Output:





Practical No. 5

Aim: Design ASP.NET Pages with Rich Controls (Calendar Control)

Default.aspx Page:

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Default2.aspx.cs" Inherits="Default2" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<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="Rich Controls (Calendar)"
         Font-Bold="True" Font-Italic="True" Font-Names="Century Schoolbook"
         Font-Size="20pt" Font-Underline="True" ForeColor="#336600"></asp:Label>
      <br /><br />
      <asp:Calendar ID="Calendar1" runat="server" BackColor="White"
         BorderColor="Black" BorderStyle="Solid" CellSpacing="1" Font-Names="Verdana"
         Font-Size="12pt" ForeColor="Black" Height="250px" NextPrevFormat="ShortMonth" Width="330px">
         <DayHeaderStyle Font-Bold="True" Font-Size="8pt" ForeColor="#333333" Height="8pt" />
         <DayStyle BackColor="#CCCCCC" />
         <NextPrevStyle Font-Bold="True" Font-Size="8pt" ForeColor="White" />
         <OtherMonthDayStyle ForeColor="#999999" />
         <SelectedDayStyle BackColor="#333399" ForeColor="White" />
         <TitleStyle BackColor="#333399" BorderStyle="Solid" Font-Bold="True"
            Font-Size="12pt" ForeColor="White" Height="12pt" />
         <TodayDayStyle BackColor="#999999" ForeColor="White" />
      </asp:Calendar>
      <br />
      <asp:Label ID="Label2" runat="server" Font-Names="Times New Roman" Font-Size="15pt" Text = "Todays
Date: "></asp:Label>
      <br /><br />
      <asp:Label ID="Label3" runat="server" Font-Names="Times New Roman" Font-Size="15pt"
Text="Select Your Birth Date: "></asp:Label>
      <br /><br />
      <asp:Label ID="Label4" runat="server" Font-Names="Times New Roman" Font-Size="15pt" Text="Days
remaining for Yor Birthday: "></asp:Label>
      <br /><br />
      <asp:Label ID="Label5" runat="server" Font-Names="Times New Roman" Font-Size="15pt" Text="Days
Remaining for NEW YEAR: "></asp:Label>
      <br /><br />
      <asp:Button ID="Button1" runat="server" Text="Result" Font-Italic="False"
         Font-Names="Times New Roman" Font-Size="15pt" Font-Bold="True"
```

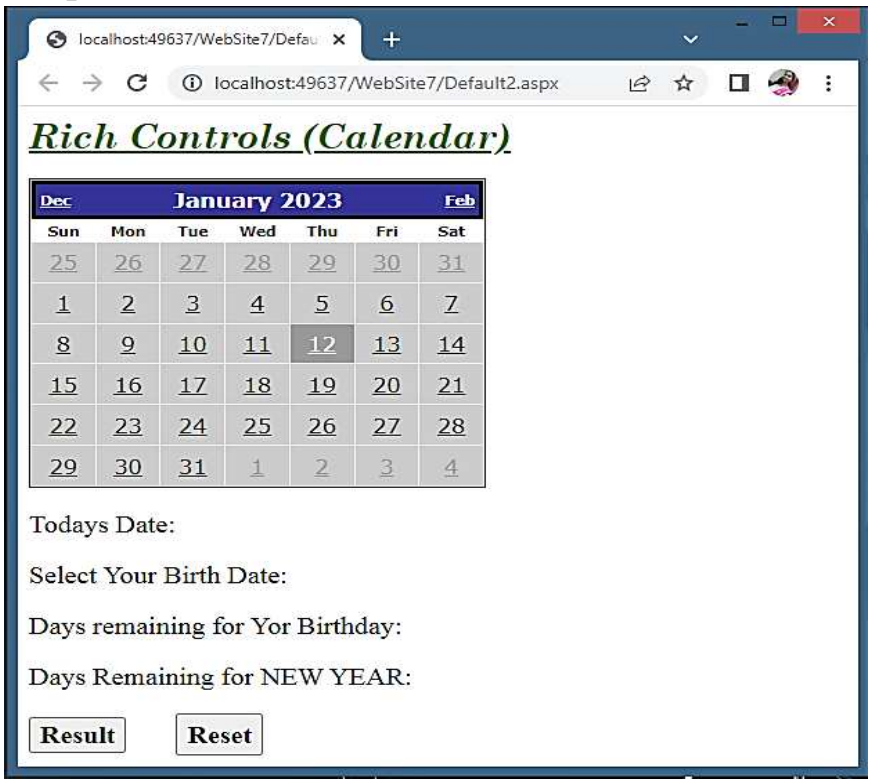
```
        onclick="Button1_Click" />
             
      <asp:Button ID="Button2" runat="server" Text="Reset" Font-Italic="False"
        Font-Names="Times New Roman" Font-Size="15pt" Font-Bold="True"
        Height="34px" onclick="Button2_Click" />
      <br />
   </div>
   </form>
</body>
</html>
```

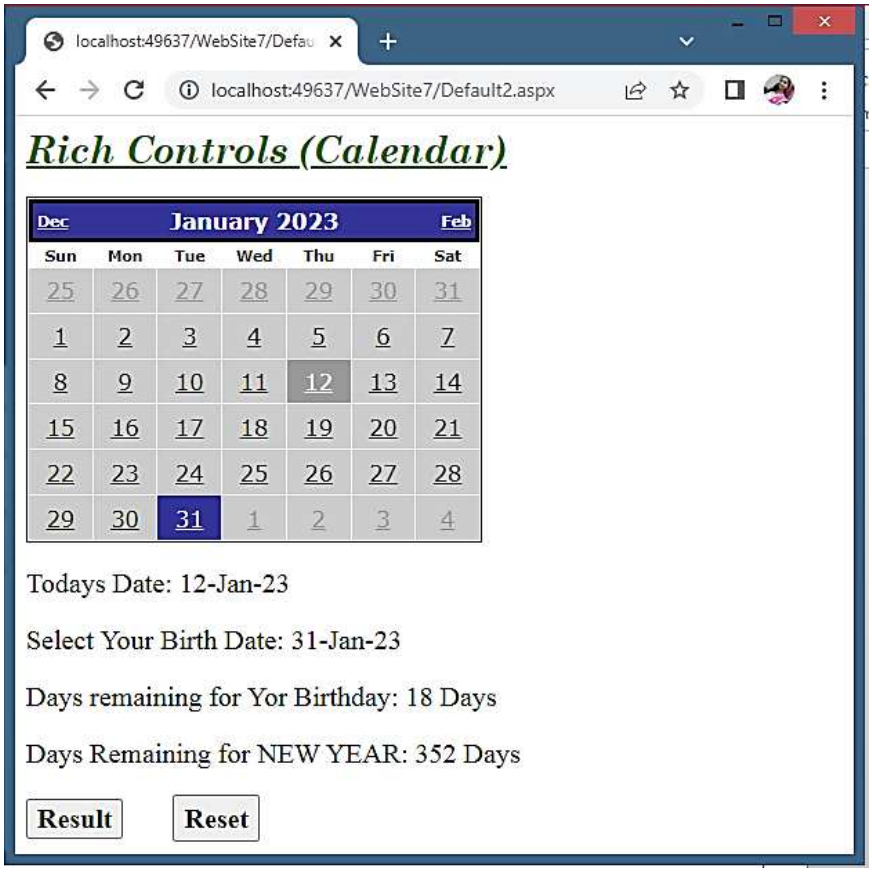
Default.aspx.cs Page:

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

public partial class Default2 : System.Web.UI.Page
{
   protected void Page_Load(object sender, EventArgs e)
   {  }
   protected void Button1_Click(object sender, EventArgs e)
   {
      //Current Date
      Label2.Text = Label2.Text + " " + Calendar1.TodaysDate.ToShortDateString();

      //Birthday
      Label3.Text = Label3.Text + " " + Calendar1.SelectedDate.Date.ToShortDateString();

      //Calculation
      int year = Calendar1.SelectedDate.Year;
      int month = Calendar1.SelectedDate.Month;
      int day = Calendar1.SelectedDate.Day;
      TimeSpan d = new DateTime(year,month,day) - DateTime.Now;
      Label4.Text = Label4.Text + " " + d.Days.ToString() + " Days";

      //New Year
      TimeSpan d1 = new DateTime(2023, 12, 31) - DateTime.Now;
      Label5.Text = Label5.Text + " " + d1.Days.ToString() + " Days";
   }
   protected void Button2_Click(object sender, EventArgs e)
   {
      Response.Redirect("http://localhost:49637/WebSite7/Default2.aspx");
```

```
    }
}
```

Output:


localhost:49637/WebSite7/Defau
localhost:49637/WebSite7/Default2.aspx
Rich Controls (Calendar)
Dec
January 2023
Feb
Sun Mon Tue Wed Thu Fri Sat
25 26 27 28 29 30 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 31 1 2 3 4
Todays Date:
Select Your Birth Date:
Days remaining for Yor Birthday:
Days Remaining for NEW YEAR:
Result
Reset


localhost:49637/WebSite7/Defau
localhost:49637/WebSite7/Default2.aspx
Rich Controls (Calendar)
Dec
January 2023
Feb
Sun Mon Tue Wed Thu Fri Sat
25 26 27 28 29 30 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 31 1 2 3 4
Todays Date: 12-Jan-23
Select Your Birth Date: 31-Jan-23
Days remaining for Yor Birthday: 18 Days
Days Remaining for NEW YEAR: 352 Days
Result
Reset