B.N.BANDODKAR

COLLEGE OF SCEINCE

Department of Information Technology

Practical Journal

Name : Pallavi .B. Deshmukh

Class : T.Y.BSc.IT

PRN NO : 2019430123

Subject : Advanced Web Programming

Course Code : USIT5P3

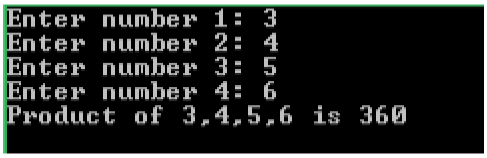
Practical No. - 1

A) Aim : Create an application that obtains 4 inputs and show product of the same.

Program :

```
using System;

namespace ConsoleApplication4

{

class Program

{ public static void Main(string[] args)

{

int a, b, c, d, prod;

Console.Write("Enter number 1: "); a =

Int32.Parse(Console.ReadLine());

Console.Write("Enter number 2: "); b =

Convert.ToInt32(Console.ReadLine());

Console.Write("Enter number 3: "); c =

Convert.ToInt32(Console.ReadLine());

Console.Write("Enter number 4: "); d =

Convert.ToInt32(Console.ReadLine());

prod = a * b * c * d;

Console.WriteLine("Product of {0},{1},{2},{3} is {4}",a,b,c,d,prod);

Console.ReadLine();

}

}

}
```

Output :



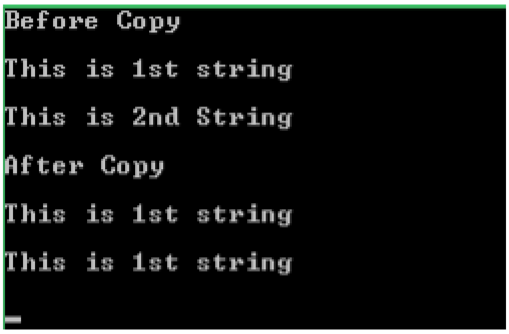
```
Enter number 1: 3
Enter number 2: 4
Enter number 3: 5
Enter number 4: 6
Product of 3,4,5,6 is 360
```

B) Aim : Create an application to demonstrate string operations.

Program :

```
using System; namespace
ConsoleApplication4 {
class Program {
 public static void Main(string[] args) {
 string str1 = "This is 1st string ";
string str2 = "This is 2nd String";
Console.WriteLine("Before Copy\n");
Console.WriteLine(str1+"\n");
Console.WriteLine(str2 + "\n");
str2=string.Copy(str1); Console.WriteLine("After
Copy\n");
Console.WriteLine(str1 + "\n");
Console.WriteLine(str2 + "\n");
Console.ReadLine();
}
}
}
```

Output :



```
Before Copy
This is 1st string
This is 2nd String
After Copy
This is 1st string
This is 1st string
_
```

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

Program :

Student Id:

Student Name:

Course Name:

Date of Birth:

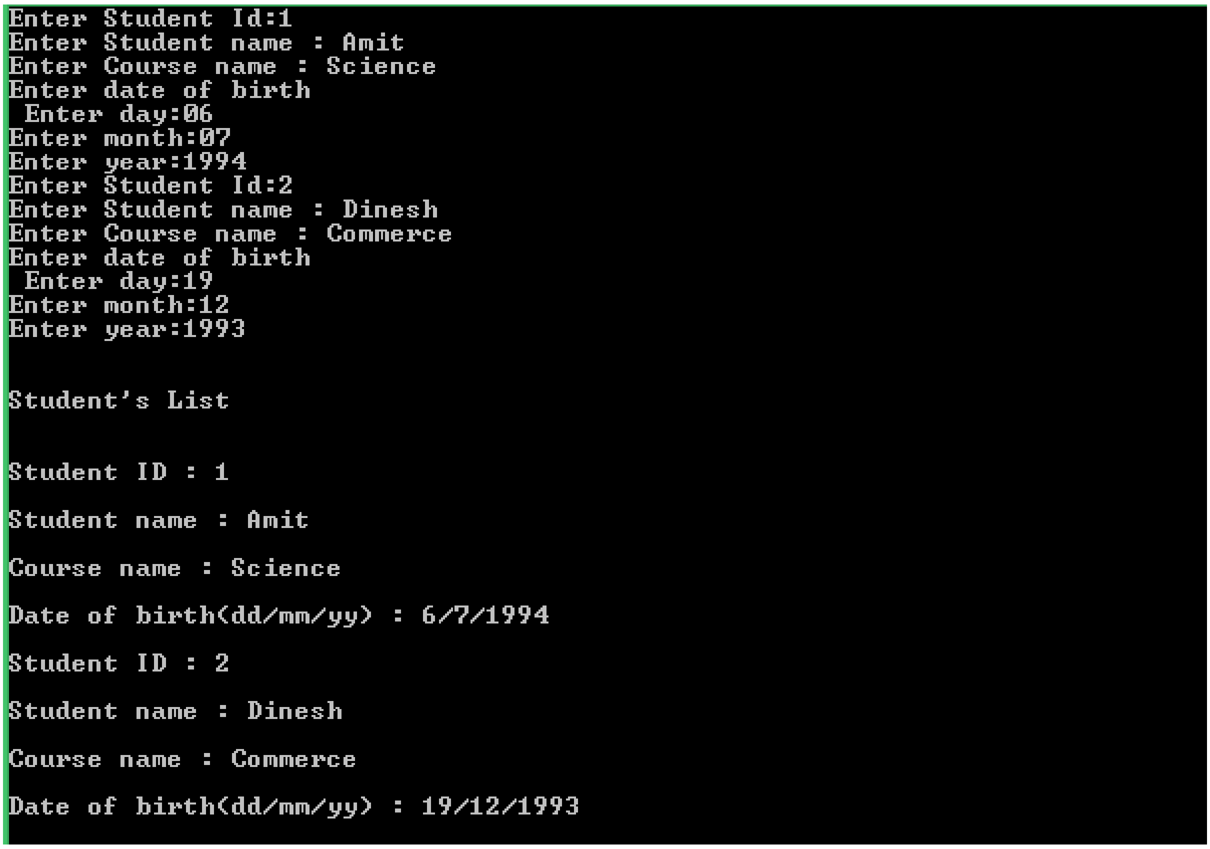
The application should also display the information of all the students once the

data is entered. Implement this using an Array of Structures.

```
using System;

namespace ConsoleApplication4 {

 class Program {

struct Student

{

 public string studentid, name, coursename;

 public int day, month, year;

}

static void Main(string[] args) {

Student[] std = new Student[5];

int i;

for (i = 0; i < 2; i++)

{

Console.Write("Enter Student Id:"); std[i].studentid = Console.ReadLine();
Console.Write("Enter Student name : ");

std[i].name = Console.ReadLine(); Console.Write("Enter Course name : ");

 std[i].coursename = Console.ReadLine(); Console.Write("Enter date of birth\n Enter day:");
std[i].day = Convert.ToInt32(Console.ReadLine());

Console.Write("Enter month:");

std[i].month = Convert.ToInt32(Console.ReadLine()); Console.Write("Enter year:");

std[i].year = Convert.ToInt32(Console.ReadLine());

}

Console.WriteLine("\n\nStudent's List\n");

for (i = 0; i < 2; i++){
```

```
Console.WriteLine("\nStudent ID : " + std[i].studentid);

Console.WriteLine("\nStudent name : " + std[i].name);

Console.WriteLine("\nCourse name : " + std[i].coursename);

Console.WriteLine("\nDate of birth(dd/mm/yy) : " + std[i].day + "/" + std[i].month + "/" +
std[i].year);

}

Console.ReadLine();

}

}
```

Output :



```
Enter Student Id:1
Enter Student name : Amit
Enter Course name : Science
Enter date of birth
 Enter day:06
Enter month:07
Enter year:1994
Enter Student Id:2
Enter Student name : Dinesh
Enter Course name : Commerce
Enter date of birth
 Enter day:19
Enter month:12
Enter year:1993


Student's List


Student ID : 1

Student name : Amit

Course name : Science

Date of birth(dd/mm/yy) : 6/7/1994

Student ID : 2

Student name : Dinesh

Course name : Commerce

Date of birth(dd/mm/yy) : 19/12/1993
```

D)Aim: Create an application to demonstrate following operations.

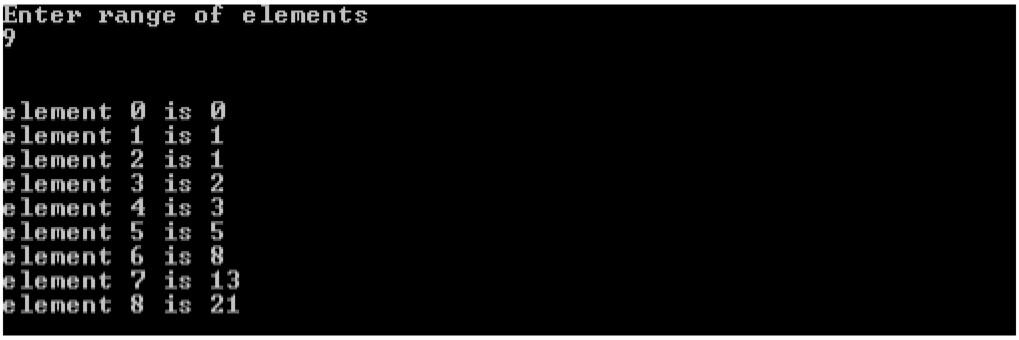
1. Aim: Create an application to generate Fibonacci series.

Program :

```
using System;

namespace ConsoleApplication4
```

```
{
class Program
{
static void Main(string[] args)
{
int m = 0, n = 1, m1, i, num;
Console.WriteLine("Enter range of elements");
num = int.Parse(Console.ReadLine());
Console.WriteLine("\n");
Console.WriteLine("element 0 is "+m);
Console.WriteLine("element 1 is "+n);
for(i = 2; i < num; i++)
{
m1 = m + n;
Console.WriteLine("element "+i+" is "+m1);
m = n;
n = m1;
}
Console.ReadLine();
}
}
}
```

Ouput:



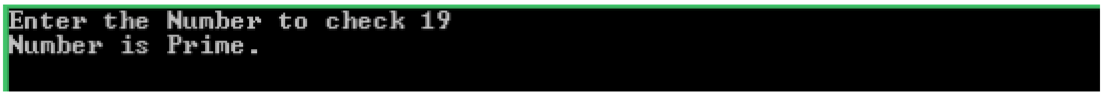
```
Enter range of elements
9


element 0 is 0
element 1 is 1
element 2 is 1
element 3 is 2
element 4 is 3
element 5 is 5
element 6 is 8
element 7 is 13
element 8 is 21
```

2. Aim : Create an application to test for prime numbers.

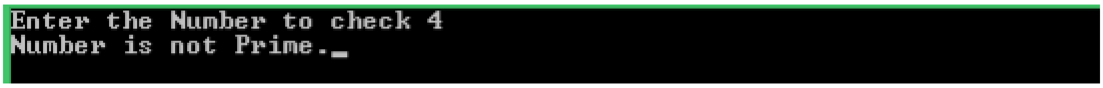
Program :

```
using System;
namespace ConsoleApplication4
{
 class Program
{
 static void Main(string[] args)
{
 int m1, i, m2 = 0, m3 = 0;
 Console.Write("Enter the Number to check ");
 m1 = int.Parse(Console.ReadLine());
 m2 = m1 / 2;
 for (i = 2; i <= m2; i++)
{
 if (m1 % i == 0)
{
 Console.Write("Number is not Prime.");
 m3 = 1;
 break;
}
}
 if (m3 == 0)
Console.Write("Number is Prime.");
Console.ReadLine();
}
}
}
```

Ouput:



```
Enter the Number to check 19
Number is Prime.
```



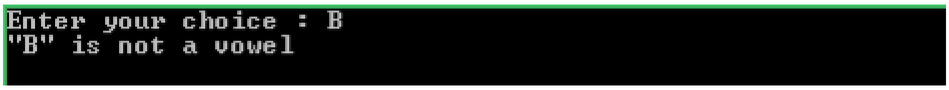
```
Enter the Number to check 4
Number is not Prime._
```

3.Aim: Create an application to test for vowels.

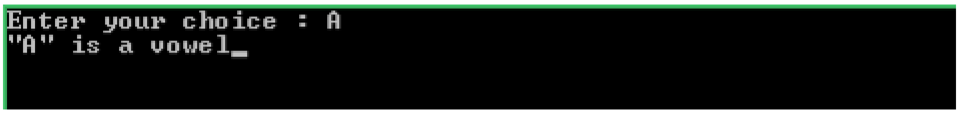
Program :

```
using System;
namespace ConsoleApplication4{
class Program
 {
 static void Main(string[] args)
 {char choice;
 Console.Write("Enter your choice : ");
 choice = (char)Console.Read();
 switch (choice)
 {
 case 'a':
 case 'A':
 case 'e':
 case 'E':
 case 'i':
 case 'I':
 case 'o':
 case 'O':
 case 'u':
 case 'U':
 Console.Write("\"" + choice + "\" " + "is a vowel");
 break;
 default:
 Console.Write("\""+choice+"\" " + "is not a vowel");
 break;
 }
 Console.ReadLine();
 }
 }
 }
```

Ouput:



```
Enter your choice : B
"B" is not a vowel
```



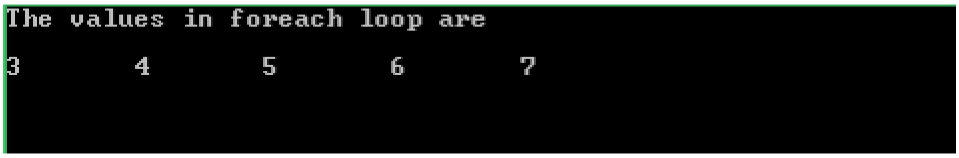
```
Enter your choice : A
"A" is a vowel_
```

4.Aim: Create an application to demonstrate the use of foreach loop with arrays.

Program :

```
using System;
namespace ConsoleApplication4
{
 class Program
 {
 static void Main(string[] args)
 {
 int[] a = { 3, 4, 5, 6, 7 };
 Console.WriteLine("The values in foreach loop are \n");
 foreach (int i in a)
 {
 Console.Write(i+"\t");
 }
 Console.ReadLine();
 }
 }
 }
```

Ouput:



```
The values in foreach loop are

3       4       5       6       7
```

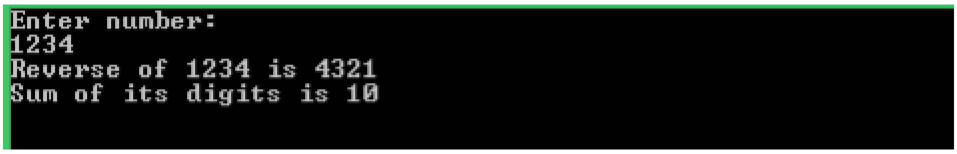
5.Aim: Create an application for reversing a number and find sum of digits of a number.

Program :

```
 using System;
 namespace ConsoleApplication4
 {
```

```
class Program

{

static void Main(string[] args)

{

int num, anum, rev = 0, temp, sum = 0;

Console.Write("Enter number: \n");

num = int.Parse(Console.ReadLine());

anum = num;

while (num > 0)

{

temp = num % 10;

rev = rev * 10 + temp;

sum = sum + temp;

num = num / 10;

}

Console.WriteLine("Reverse of " + anum + " is " + rev);

Console.WriteLine("Sum of its digits is " + sum);

Console.ReadLine();

}

}

}
```

Ouput:



```
Enter number:
1234
Reverse of 1234 is 4321
Sum of its digits is 10
```

Practical No. - 2

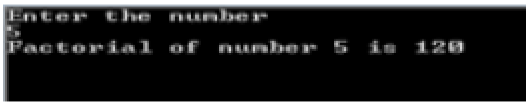
A) Aim : Create simple application to perform following operations

1.Aim : Create simple application for finding factorial value.

Program :

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
namespace ConsoleApplication4
 {
class Program
 {
 public static void Main(string[] args)
 { int i, fact = 1,num1;
 Console.WriteLine("Enter the number");
 num1 = int.Parse(Console.ReadLine());
 for (i = 1; i <= num1; i++)
 { fact = fact * i; }
Console.WriteLine("Factorial of number " + num1 + " is " + fact);
 Console.ReadLine();
 }
 }
 }
```

Output :



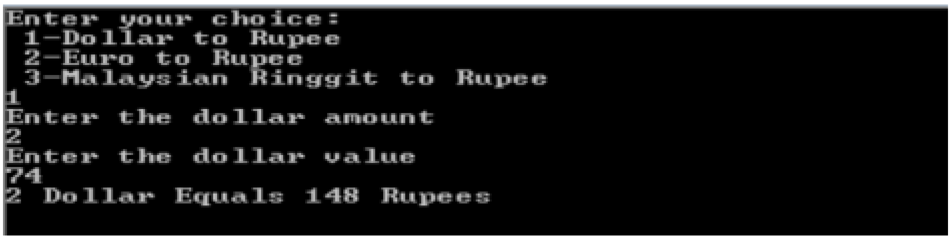
```
Enter the number
5
Factorial of number 5 is 120
```

2.Aim : Create simple application for Money Conversion.

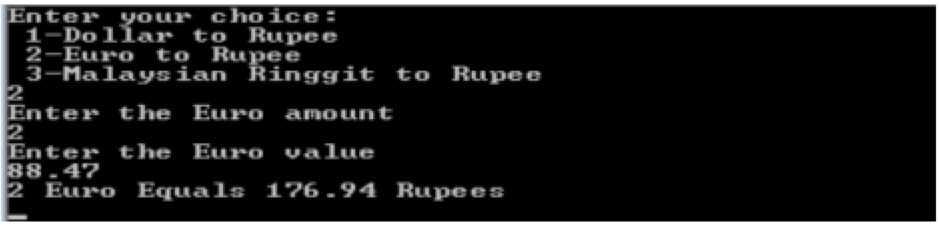
Program :

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
namespace ConsoleApplication4
{
 class Program
 {
 public static void Main(string[] args)
 {
 int choice;
 Console.WriteLine("Enter your choice:\n 1-Dollar to Rupee \n 2-Euro to
Rupee \n 3-Malaysian Ringgit to Rupee ");
 choice = int.Parse(Console.ReadLine());
 switch (choice)
 {
 case 1:
 Double dollar, rupee, val1;
 Console.WriteLine("Enter the dollar amount");
 dollar = double.Parse(Console.ReadLine());
 Console.WriteLine("Enter the dollar value");
 val1 = double.Parse(Console.ReadLine());
 rupee = dollar * val1;
 Console.WriteLine("{0} Dollar Equals {1} Rupees", dollar, rupee);
 break;
 case 2:
 Double euro, rupee2, val2;
 Console.WriteLine("Enter the Euro amount");
 euro = double.Parse(Console.ReadLine());
 Console.WriteLine("Enter the Euro value");
 val2 = double.Parse(Console.ReadLine());
 rupee2 = euro * val2;
 Console.WriteLine("{0} Euro Equals {1} Rupees", euro, rupee2);
 break;
 case 3:
 Double ringit, rupee3, val3;
 Console.WriteLine("Enter the ringgit amount");
 ringit = double.Parse(Console.ReadLine());
 Console.WriteLine("Enter the ringgit value");
 val3 = double.Parse(Console.ReadLine());
 rupee3 = ringit * val3;
 Console.WriteLine("{0} Malaysian Ringgit Equals {1} Rupees", ringit, rupee3);
 break;
 default:
 Console.WriteLine("Invalid Choice");
 break;
 }
 Console.ReadLine();
 }
 }
 }
```

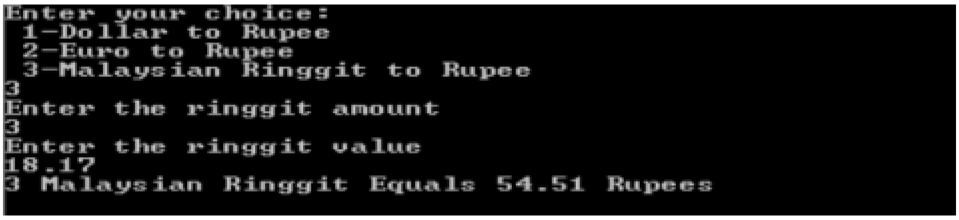
Output :



```
Enter your choice:
 1-Dollar to Rupee
 2-Euro to Rupee
 3-Malaysian Ringgit to Rupee
1
Enter the dollar amount
2
Enter the dollar value
74
2 Dollar Equals 148 Rupees
```



```
Enter your choice:
 1-Dollar to Rupee
 2-Euro to Rupee
 3-Malaysian Ringgit to Rupee
2
Enter the Euro amount
2
Enter the Euro value
88.47
2 Euro Equals 176.94 Rupees
_
```



```
Enter your choice:
 1-Dollar to Rupee
 2-Euro to Rupee
 3-Malaysian Ringgit to Rupee
3
Enter the ringgit amount
3
Enter the ringgit value
18.17
3 Malaysian Ringgit Equals 54.51 Rupees
```

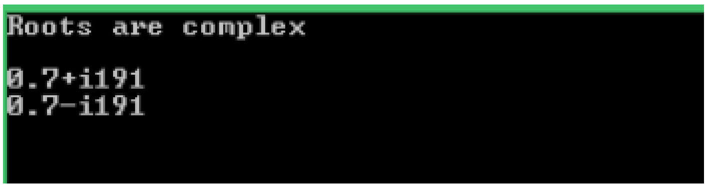
3.Aim: Create simple application for Quadratic Equation.

Program :

```
using System;
namespace ConsoleApplication4
{
 class Program
 {
 void findRoots(int a, int b, int c)
 {
 if (a == 0)
 {
 Console.Write("invalid");
 return;
 }
 int d = b * b - 4 * a * c;
 Double sqrt_val = Math.Abs(d);
 if (d > 0)
 {
 Console.WriteLine(" The roots are real and different\n");
 Console.Write((Double)(-b + sqrt_val) / (2 * a) + "\n" + (Double)(-b - sqrt_val) / (2 * a));
 }
 else
 { Console.WriteLine("Roots are complex \n");
 Console.Write(-(Double)b / (2 * a) + "+i" + sqrt_val + "\n" +
 -(Double)b / (2 * a) + "-i" + sqrt_val);
} }
```

```
 public static void Main(string[] args)
 {
 Program rt = new Program();
 int a=1,b=-7,c=12;
 rt.findRoots(a, b, c);
 Console.ReadLine();
 } } }
```

Output :



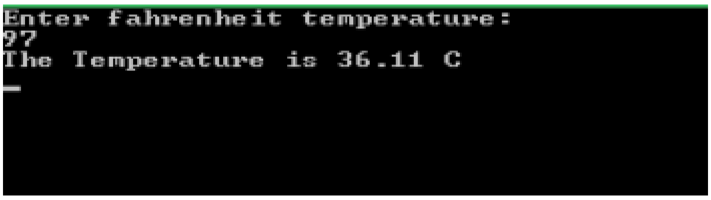
```
Roots are complex

0.7+i191
0.7-i191
```

4.Aim: Create simple application for Temperature Conversion.

a)Program for simple application for Temperature Conversion - Fahrenheit to Celsius :

```
using System;
namespace ConsoleApplication4
{
 class Program
 {
 public static void Main(string[] args)
 {
 double celsius;
 Console.WriteLine("Enter fahrenheit temperature: ");
 double fahreinheit = double.Parse(Console.ReadLine());
 celsius = (fahreinheit - 32) * 5 / 9;
 Console.WriteLine("The Temperature is "+Celsius.ToString(“0.00”)" + "C");
 Console.ReadLine();
 } } }
```

Output :

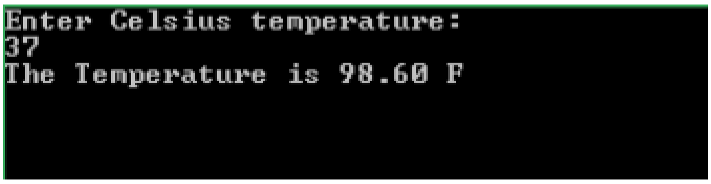


```
Enter fahrenheit temperature:
97
The Temperature is 36.11 C
_
```

b)Program for simple application for Temperature Conversion - Celsius to Fahrenheit :

```
using System;
namespace ConsoleApplication4
{
class Program
 {
 public static void Main(string[] args)
 {
 float fahreinheit;
 Console.WriteLine("Enter Celsius temperature: ");
 float celsius = float.Parse(Console.ReadLine());
 fahreinheit = (celsius*9)/5+32;
 Console.WriteLine("The Temperature is " + fahreinheit.ToString("0.00") + " F");
  Console.ReadLine();
 } } }
```

Output :



```
Enter Celsius temperature:
37
The Temperature is 98.60 F
```

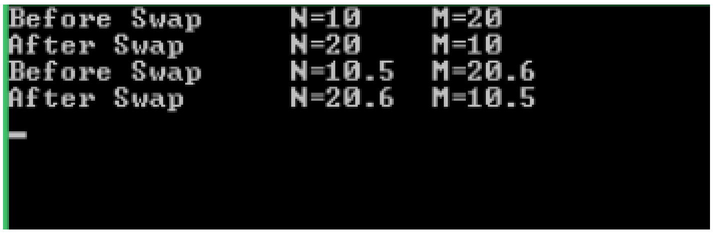
B) Aim : Create simple application to demonstrate use of following concepts :

1.Aim : Create simple application to demonstrate Function Overloading.

Program :

```
using System;
namespace ConsoleApplication4
{
 class Program
 {
 public void swap(ref int n,ref int m)
 {
 int t;
 t = n;
 n=m;
 m = t;
 }
 public void swap(ref float f1, ref float f2)
 {
 float f;
 f = f1;
 f1 = f2;
 f2 = f;
 }
 public static void Main(string[] args)
 {
 Program p1 = new Program();
 int n = 10, m = 20;
 Console.WriteLine("Before Swap "+"\tN=" + n + "\tM=" + m);
 p1.swap(ref n,ref m);
 Console.WriteLine("After Swap " + "\tN=" + n + "\tM=" + m);
 float f1 = 10.5f, f2 = 20.6f;
 Console.WriteLine("Before Swap " + "\tN=" + f1 + "\tM=" + f2);
 p1.swap(ref f1, ref f2);
 Console.WriteLine("After Swap " + "\tN=" + f1 + "\tM=" + f2);
 Console.ReadLine();
 }
 }
 }
```

Output :



```
Before Swap      N=10     M=20
After Swap       N=20     M=10
Before Swap      N=10.5   M=20.6
After Swap       N=20.6   M=10.5
_
```

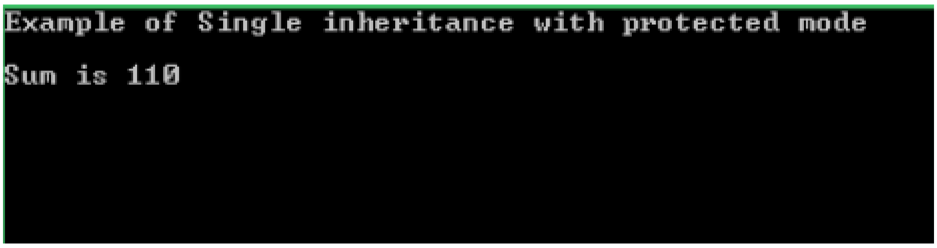
2.Aim : Create simple application to demonstrate Inheritance (all types).

a. Program for Single Inheritance:

```
using System;
 namespace ConsoleApplication4
 {
 public class base1
 {
 protected int a = 50;
 protected int b = 60;
 }
 public class base2 : base1
 {
 public void show()
 {
 int c;
 c = a + b;
 Console.WriteLine("Example of Single inheritance with protected mode " + "\n\nSum is " +
c);
 }}
 class Program
 {
```

```
public static void Main(string[] args)
{
base2 c2 = new base2();
c2.show();
Console.ReadLine();
}
}
}
```

Output :

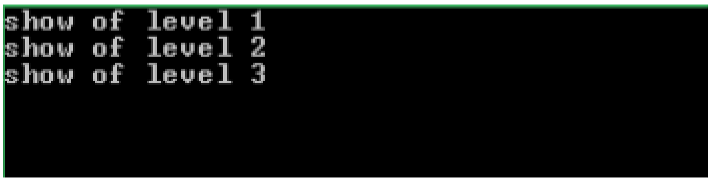


```
Example of Single inheritance with protected mode

Sum is 110
```

b. Program for Multilevel Inheritance :

```
using System;
namespace ConsoleApplication4
{
class test
{
public void show()
{
Console.WriteLine("show of level 1");
}
}
class testme : test
{
public void showme()
{
base.show();
Console.WriteLine("show of level 2");
}
}
class Testus : testme
{
public void showus()
{
base.showme();
Console.WriteLine("show of level 3");
} }
public class program
{
public static void Main(string[] args)
{
Testus t1 = new Testus();
t1.showus();
Console.ReadLine();
}
}
}
```

Output :



```
show of level 1
show of level 2
show of level 3
```

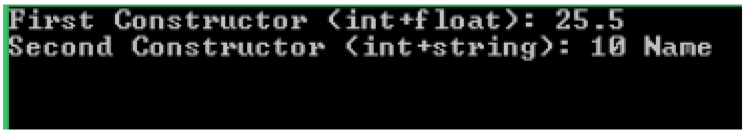
3.Aim: Create simple application to demonstrate Constructor overloading.

Program :

```
using System;
namespace ConsoleApplication4
{
class Add {
int x, y;
double f;
string s;
public Add(int a, double b)
{
x = a;
f = b;
}
public Add(int a, string b)
{
y = a;
s = b;
}
public void show()
{
Console.WriteLine("First Constructor (int+float): {0} ",(x+f));
}
public void show1()
{
Console.WriteLine("Second Constructor (int+string): {0} ", (y + s));
} }
class Program
{ public static void Main(string[] args)
{
Add g = new Add(10, 15.5);
g.show();
```

```
 Add m = new Add(10," Name");
 m.show1();
 Console.ReadLine();
 }
 }
 }
```

Output :



```
First Constructor (int+float): 25.5
Second Constructor (int+string): 10 Name
```

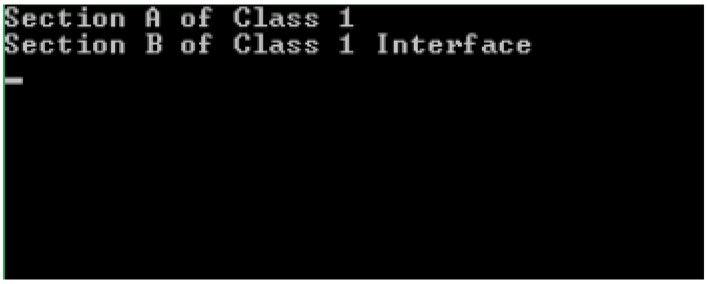
4.Aim: Create simple application to demonstrate Interfaces.

Program :

```
using System;
 namespace ConsoleApplication4{
 public interface Class1
 {
 void draw();
 }
 public class Class1A : Class1
 {
 public void draw()
 {
 Console.WriteLine("Section A of Class 1");
 } }
 public class Class1B : Class1
 { public void draw()
 {
 Console.WriteLine("Section B of Class 1 Interface");
 } }
 public class TestInterface{
```

```
public static void Main() {
Class1 d;
d = new Class1A();
d.draw();
d = new Class1B();
d.draw();
Console.ReadLine();
}
}
}
```

Output :



```
Section A of Class 1
Section B of Class 1 Interface
_
```

C) Aim: Create simple application to demonstrate use of following concepts **:**

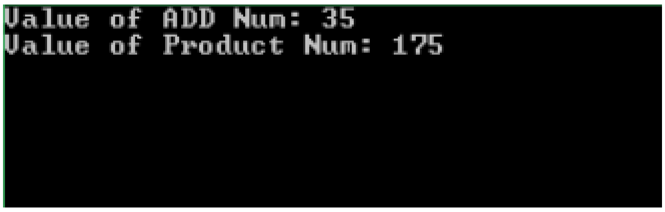
1.Aim : Create simple application for using Delegates and events.

Program :

```
using System;
namespace ConsoleApplication4
{
delegate int NumberChanger(int n);
class Program
{ static int num = 10;
public static int AddNum(int p)
{
num += p;
return num;
```

```
 }
 public static int MultNum(int q)
 {
 num * = q;
 return num;
 }
 public static int getNum()
 {
 return num;
 }
 public static void Main(string[] args)
 {
 NumberChanger nc1 = new NumberChanger(AddNum);
 NumberChanger nc2 = new NumberChanger(MultNum);
//calling the methods using the delegate objects
 nc1(25);
 Console.WriteLine("Value of ADD Num: {0}", getNum());
 nc2(5);
 Console.WriteLine("Value of Product Num: {0}", getNum());
 Console.ReadKey(); //Console.ReadLine();
 }
 }
 }
```

Output :



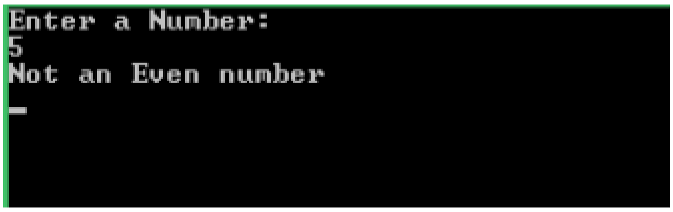
```
Value of ADD Num: 35
Value of Product Num: 175
```

2.Aim : Create simple application to demonstrate Exception handling.

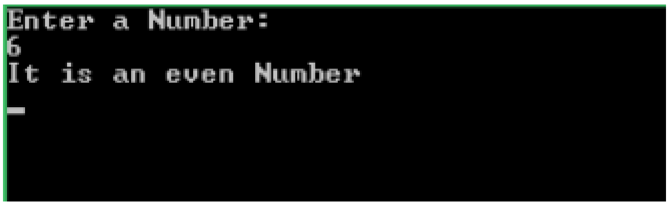
Program :

```
using System;
 namespace ExceptionHandlingExample
 {
 class NotEvenException : Exception
 {
 public NotEvenException(String msg)
 : base(msg) { }
 }
 class Program
 {
 public static void Main(string[] args)
 {
 int num;
 try
 {
 Console.WriteLine("Enter a Number: ");
 num = int.Parse(Console.ReadLine());
 if ((num % 2) != 0)
 {
 throw new NotEvenException("Not an Even number");
 }
 else
 {
 Console.WriteLine("It is an even Number");
 Console.ReadLine();
 }
 }
 catch (NotEvenException e) { Console.WriteLine(e.Message); }
 Console.ReadLine();
 } } }
```

Output :



```
Enter a Number:
5
Not an Even number
_
```



```
Enter a Number:
6
It is an even Number
_
```

Practical No. - 3

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

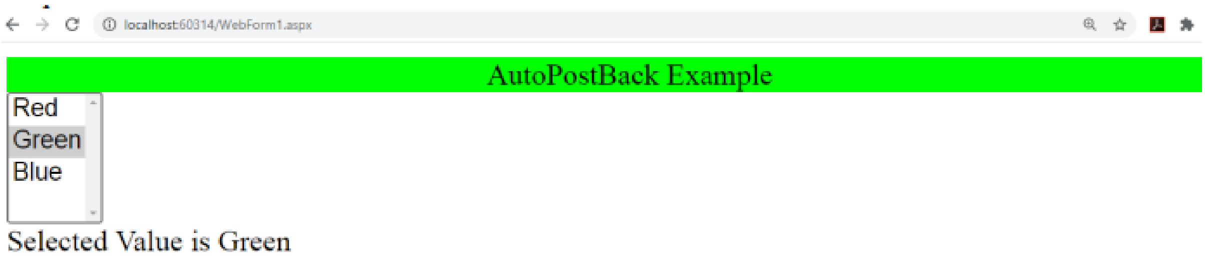
Program for Webform1.aspx page :

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="WebForm1.aspx.cs"
Inherits="Ptractical3.WebForm1" %>

<!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="background-color: #00FF00">

 AutoPostBack Example

 </div>

 <div>

 <asp:ListBox ID="ListBox1" runat="server" AutoPostBack="True"

 onselectedindexchanged="ListBox1_SelectedIndexChanged">

 <asp:ListItem>Red</asp:ListItem>

 <asp:ListItem>Green</asp:ListItem>

 <asp:ListItem>Blue</asp:ListItem>

 </asp:ListBox>

 <br />

 <asp:Label ID="Label1" runat="server" Text="Label"></asp:Label>

 </div>

 </form>

</body>

</html>
```

Program for Webform1.aspx.CS page :

```
using System;

using System.Collections.Generic;
using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace Ptractical3

 {

public partial class WebForm1 : System.Web.UI.Page

 {

 protected void Page_Load(object sender, EventArgs e)

 {

 }

 protected void ListBox1_SelectedIndexChanged(object sender, EventArgs e)

 {

 Label1.Text = "Selected Value is " + ListBox1.Text;

 }

 }

 }
```

Output :



AutoPostBack Example

Selected Value is Green

B) Aim : Demonstrate the use of Calendar control to perform following operations :

1.Aim : Create program for displaying selected day in a calendar control using style

Program for Webform1.aspx page :

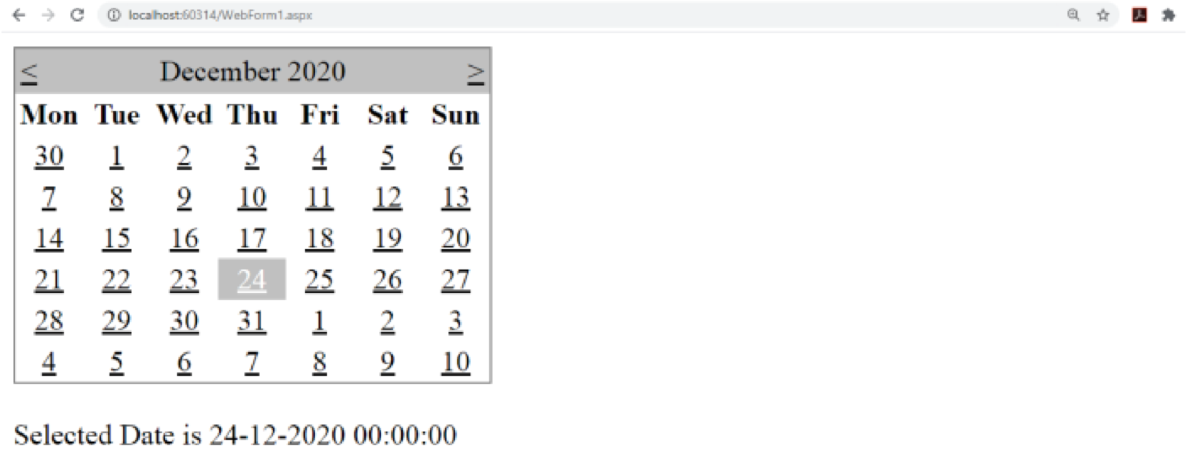
```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="WebForm1.aspx.cs"
Inherits="Ptractical3.WebForm1" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
 <title></title>
</head>
<body>
 <form id="form1" runat="server">
 <div>
 <asp:Calendar ID="Calendar1" runat="server"
 onselectionchanged="Calendar1_SelectionChanged"></asp:Calendar>
 <br />
 <asp:Label ID="ShowDate" runat="server" Text="Label"></asp:Label>
 </div>
 </form>
</body>
</html>
```

Program for Webform1.aspx.cs page :

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
namespace Ptractical3
{
```

```
public partial class WebForm1 : System.Web.UI.Page
{
protected void Page_Load(object sender, EventArgs e){
}
 protected void Calendar1_SelectionChanged(object sender, EventArgs e)
 {
 ShowDate.Text = " Selected Date is " + Calendar1.SelectedDate;
 } } }
```

Output :



localhost:60314/WebForm1.aspx

< December 2020 >

Mon	Tue	Wed	Thu	Fri	Sat	Sun
30	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	5	6	7	8	9	10

Selected Date is 24-12-2020 00:00:00

2.Aim : Create program to display difference between two calendar dates

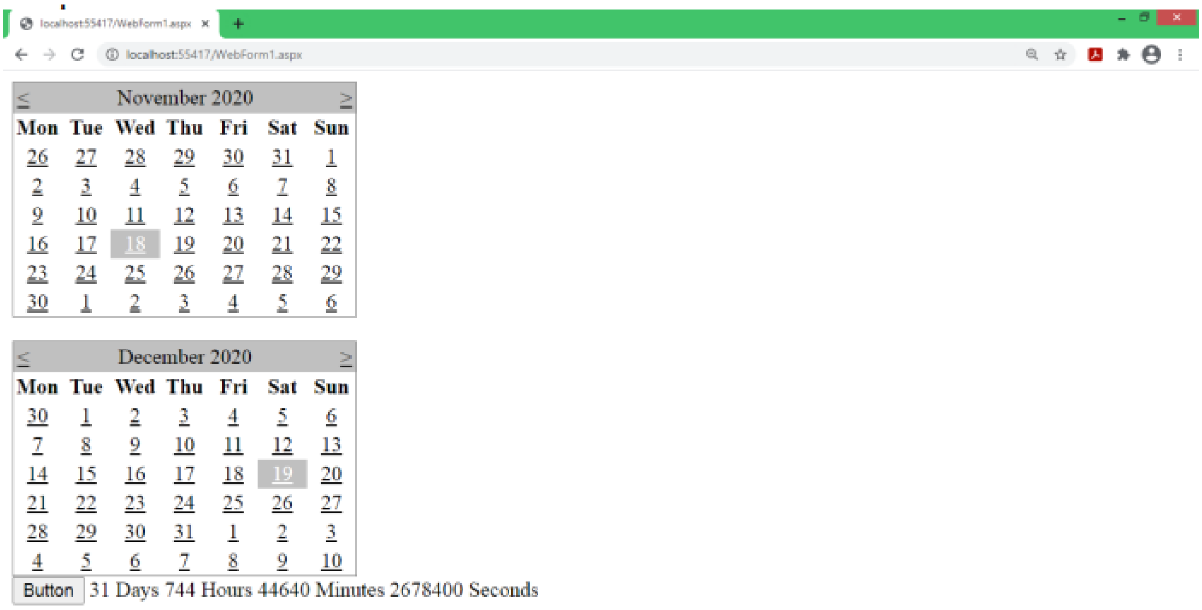
Program for Webform1.aspx page :

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="WebForm1.aspx.cs"
Inherits="WebApplication9.WebForm1" %>
<!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:Calendar ID="Calendar2" runat="server"></asp:Calendar> &nbsp;<asp:Calendar
ID="Calendar3" runat="server"></asp:Calendar>
 <asp:Button ID="Button1" runat="server" onclick="Button1_Click" Text="Button" />
<asp:Label ID="Label1" runat="server" Text="Label"></asp:Label>
 <br />
 <br />
 </div>
 </form>
</body>
</html>
```

Program for Webform1.aspx.cs page :

```
using System;
using System.Collections.Generic;
using System.Linq; using
System.Web; using
System.Web.UI;
using System.Web.UI.WebControls;
using System.Xml; using
System.IO; using System.Xml.Linq;
namespace WebApplication9
{
public partial class WebForm1 : System.Web.UI.Page
 {
 protected void Page_Load(object sender, EventArgs e)
 {
 }
 protected void Button1_Click(object sender, EventArgs e)
 {
 DateTime dt1 = Calendar2.SelectedDate;
 DateTime dt2 = Calendar3.SelectedDate;
 TimeSpan ts = dt2 - dt1;
 Label1.Text = ts.TotalDays.ToString() + " Days "+ "" + ts.TotalHours.ToString() + " Hours "+
 "" + ts.TotalMinutes.ToString() + " Minutes "+ "" + ts.TotalSeconds.ToString() + " Seconds ";

 }

 }

 }
```

Output :



< November 2020 >

Mon	Tue	Wed	Thu	Fri	Sat	Sun
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	1	2	3	4	5	6

< December 2020 >

Mon	Tue	Wed	Thu	Fri	Sat	Sun
30	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	5	6	7	8	9	10

```
31 Days 744 Hours 44640 Minutes 2678400 Seconds
```

C) Aim: Demonstrate the use of Tree view control perform operations.

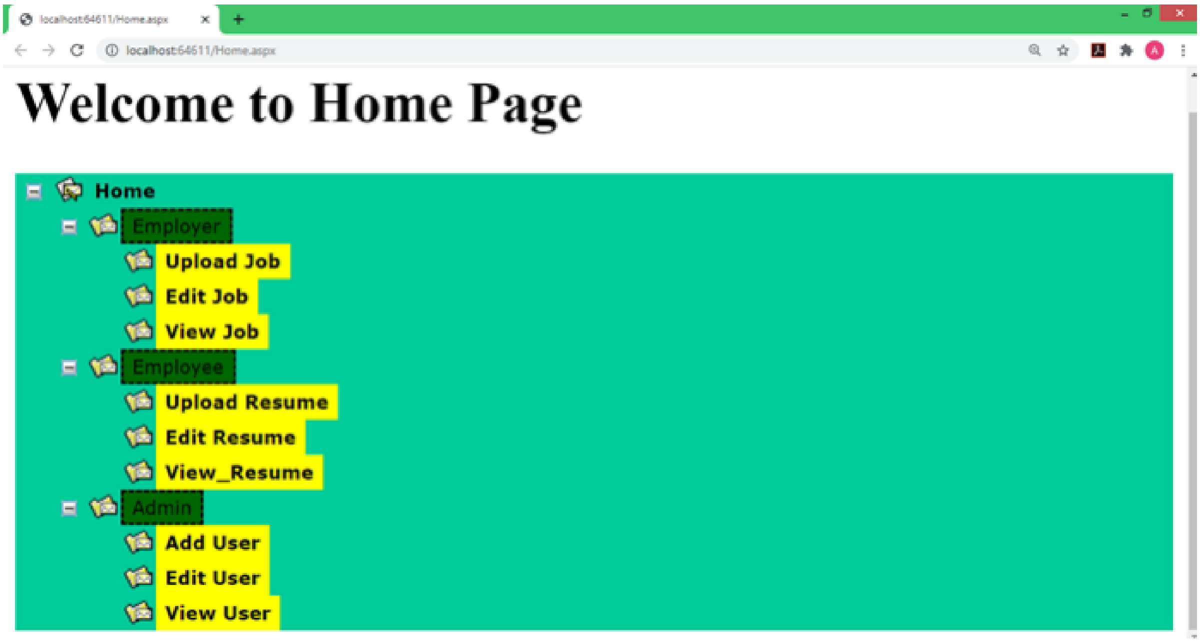
Program :

```
<%@ Page Language="C#" AutoEventWireup="true"
CodeBehind="Home.aspx.cs" Inherits="WebApplication10.WebForm1" %>
<!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>
 <h1>Welcome to Home Page</h1>
 <asp:TreeView ID="TreeView1" runat="server" ImageSet="Inbox"
 BackColor="#00CC99" BorderColor="#000099" Font-Bold="True">
 <HoverNodeStyle Font-Underline="True" BackColor="Red" />
 <LeafNodeStyle BackColor="Yellow" />
 <Nodes>
 <asp:TreeNode NavigateUrl="~/Home.aspx" Target="_blank"
Text="Home"
 Value="Home">
 <asp:TreeNode NavigateUrl="~/Employer.aspx" Target="_blank"
Text="Employer"
 Value="Employer">
 <asp:TreeNode NavigateUrl="~/Upload_Job.aspx"
Target="_blank" Text="Upload Job"
 Value="Upload Job"></asp:TreeNode>
 <asp:TreeNode NavigateUrl="~/Edit_Job.aspx" Target="_blank"
Text="Edit Job"
 Value="Edit Job"></asp:TreeNode>
 <asp:TreeNode NavigateUrl="~/View_Job.aspx" Target="_blank"
```

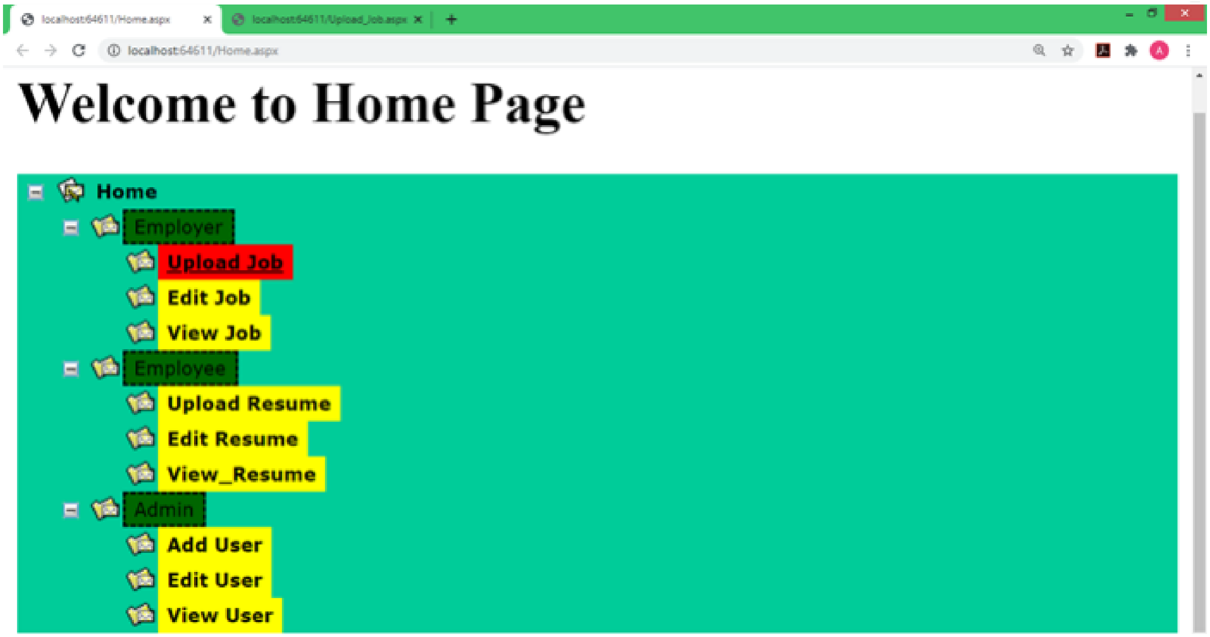
```
Text="View Job"
 Value="View Job"></asp:TreeNode>
 </asp:TreeNode>
 <asp:TreeNode NavigateUrl="~/Employee.aspx" Target="_blank"
Text="Employee"
 Value="Employee">
<asp:TreeNode NavigateUrl="~/Upload_Resume.aspx"
Target="_blank"
 Text="Upload Resume" Value="Upload
Resume"></asp:TreeNode>
 <asp:TreeNode NavigateUrl="~/Edit_Resume.aspx"
Target="_blank"
 Text="Edit Resume" Value="Edit Resume"></asp:TreeNode>
<asp:TreeNode NavigateUrl="~/View_Resume.aspx" Target="_blank"
Text="View_Resume" Value="View_Resume"></asp:TreeNode>
</asp:TreeNode>
<asp:TreeNode NavigateUrl="~/Admin.aspx" Target="_blank"
Text="Admin" Value="Admin">
<asp:TreeNode NavigateUrl="~/Add_User.aspx"
Target="_blank" Text="Add User" Value="Add User"></asp:TreeNode>
<asp:TreeNode NavigateUrl="~/Edit_User.aspx" Target="_blank" Text="Edit User"
Value="Edit User"></asp:TreeNode>
<asp:TreeNode NavigateUrl="~/View_User.aspx" Target="_blank" Text="View User"
Value="View User"></asp:TreeNode>
</asp:TreeNode>
</asp:TreeNode>
</Nodes>
<NodeStyle Font-Names="Verdana" Font-Size="8pt" ForeColor="Black"
HorizontalPadding="5px" NodeSpacing="0px" VerticalPadding="0px" />
<ParentNodeStyle Font-Bold="False" BackColor="#006600" BorderStyle="Dashed" />
<SelectedNodeStyle Font-Underline="True" HorizontalPadding="0px" VerticalPadding="0px"
BackColor="Fuchsia" Font-Bold="True" />
</asp:TreeView>
```

```
</div>

</form>

</body>

</html>
```

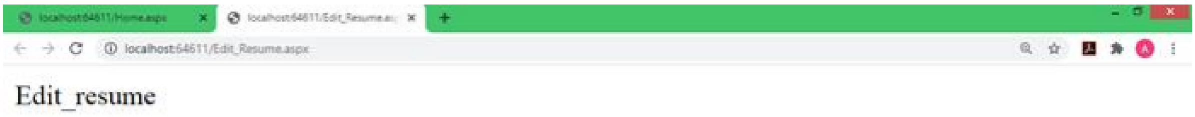
Output :



Welcome to Home Page



Welcome to Home Page


localhost:64611/Home.aspx
localhost:64611/Edit_Resume.as
localhost:64611/Edit_Resume.aspx

Edit_resume

Practical No. - 4

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

Program :

Home.aspx

```
<%@ Page Language="C#" AutoEventWireup="true"
CodeBehind="Home.aspx.cs"
Inherits="WebApplication10.WebForm1" %>
<!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-style: dashed; position:inherit">
<h1>Validation Example</h1>
<br />
<asp:Label ID="Label1" runat="server" Text="Candidate Name">
</asp:Label>
 <asp:TextBox ID="cname" runat="server"></asp:TextBox>
<asp:RequiredFieldValidator ID="RequiredFieldValidator1" runat="server"
ControlToValidate="cname" ErrorMessage="*">
</asp:RequiredFieldValidator>
<br />
<asp:Label ID="Label2" runat="server" Text="Address">
</asp:Label>
&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;
&nbsp;&nbsp;&nbsp;
<asp:TextBox ID="CAddress" runat="server"></asp:TextBox>
```

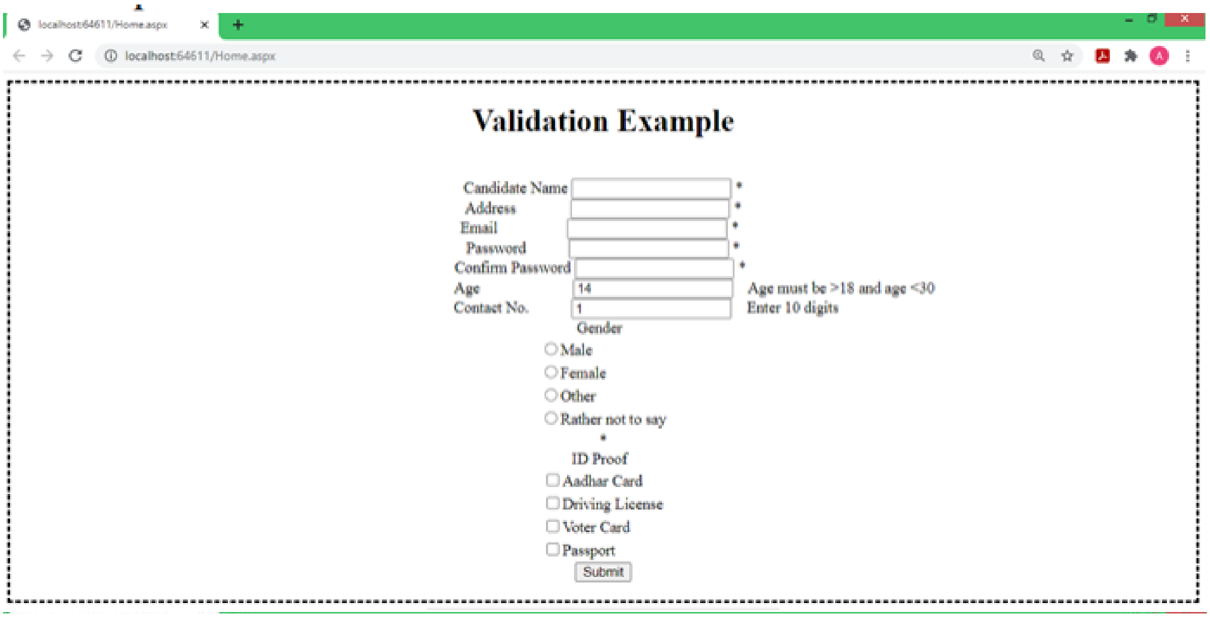
```
<asp:RequiredFieldValidator ID="RequiredFieldValidator2" runat="server"
ControlToValidate="CAddress" ErrorMessage="*">
</asp:RequiredFieldValidator>
<br />

&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;
&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;
&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;
&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;
&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;
<asp:Label ID="Label3" runat="server" Text="Email"></asp:Label>
&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;
&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;
<asp:TextBox ID="Email" runat="server">
</asp:TextBox>
<asp:RequiredFieldValidator ID="RequiredFieldValidator3"
runat="server" ControlToValidate="Email" ErrorMessage="*">
</asp:RequiredFieldValidator>
<asp:RegularExpressionValidatorID="RegularExpressionValidator1" runat="server"
ControlToValidate="Email"
ErrorMessage="Format(abc@example.com)"
ValidationExpression="\w+([-+.']\w+)*@\w+([-.]\w+)*\.\w+([-.]\w+)*">
</asp:RegularExpressionValidator>
<br />
<asp:Label ID="Label8" runat="server" Text="Password">
</asp:Label>
&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;
<asp:TextBox ID="password" runat="server" TextMode="Password">
</asp:TextBox>
<asp:RequiredFieldValidator ID="RequiredFieldValidator4"
runat="server"
 ControlToValidate="password"
```

```
ErrorMessage="*"></asp:RequiredFieldValidator>
 <br />
&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;
&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;
&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;
&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;
&nbsp;&nbsp;&nbsp;
 <asp:Label ID="Label9" runat="server" Text="Confirm
Password"></asp:Label>
 <asp:TextBox ID="confirm_password" runat="server"
TextMode="Password"></asp:TextBox>
 <asp:RequiredFieldValidator ID="RequiredFieldValidator5"
runat="server"
 ControlToValidate="confirm_password"
ErrorMessage="*"></asp:RequiredFieldValidator>
 <asp:CompareValidator ID="CompareValidator1"
runat="server"
 ControlToCompare="password"
ControlToValidate="confirm_password"
 ErrorMessage="Password does not
match"></asp:CompareValidator>
 <br />
 <asp:Label ID="Label4" runat="server"
Text="Age"></asp:Label>
&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;
&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;
&nbsp;&nbsp;&nbsp;&nbsp;
 <asp:TextBox ID="Age" runat="server"></asp:TextBox>
 <asp:RequiredFieldValidator ID="RequiredFieldValidator6" runat="server"
ControlToValidate="Age"
ErrorMessage="*"></asp:RequiredFieldValidator>
```

```
 <asp:RangeValidator ID="RangeValidator1" runat="server"
 ControlToValidate="Age" ErrorMessage="Age must be
>18 and age <30" MaximumValue="30"
 MinimumValue="18"
Type="Integer"></asp:RangeValidator>

 <br />
 &nbsp; <asp:Label ID="Label5" runat="server"
Text="Contact No."></asp:Label>
&nbsp;&nbsp;&nbsp;&nbsp; &nbsp;&nbsp;&nbsp;&nbsp;
 <asp:TextBox ID="contact" runat="server"></asp:TextBox>
 <asp:RequiredFieldValidator ID="RequiredFieldValidator7"
runat="server"
 ControlToValidate="contact"
ErrorMessage="*"></asp:RequiredFieldValidator>
 <asp:RegularExpressionValidator
ID="RegularExpressionValidator2" runat="server"
ControlToValidate="contact" ErrorMessage="Enter 10 digits"

ValidationExpression="\d{10}"></asp:RegularExpressionValidator>

 <br />
 <asp:Label ID="Label6" runat="server"
Text="Gender"></asp:Label>
&nbsp;<asp:RadioButtonList ID="Gender" runat="server"
AutoPostBack="True">
 <asp:ListItem>Male</asp:ListItem>
 <asp:ListItem>Female</asp:ListItem>
 <asp:ListItem>Other</asp:ListItem>
 <asp:ListItem>Rather not to say</asp:ListItem>
 </asp:RadioButtonList>
<asp:RequiredFieldValidator ID="RequiredFieldValidator8"
```

```
runat="server"
 ControlToValidate="Gender"
ErrorMessage="*"></asp:RequiredFieldValidator>
 <br />
 <asp:Label ID="Label7" runat="server" Text="ID
Proof"></asp:Label>
&nbsp;<asp:CheckBoxList ID="ID_Proof" runat="server">
 <asp:ListItem>Aadhar Card</asp:ListItem>
 <asp:ListItem>Driving License</asp:ListItem>
 <asp:ListItem>Voter Card</asp:ListItem>
 <asp:ListItem>Passport</asp:ListItem>
 </asp:CheckBoxList>
 <asp:Button ID="Button1" runat="server" Text="Submit"
onclick="Button1_Click" />
 <br />
 <br />
</div>
</form>
</body></html>
```

Output :

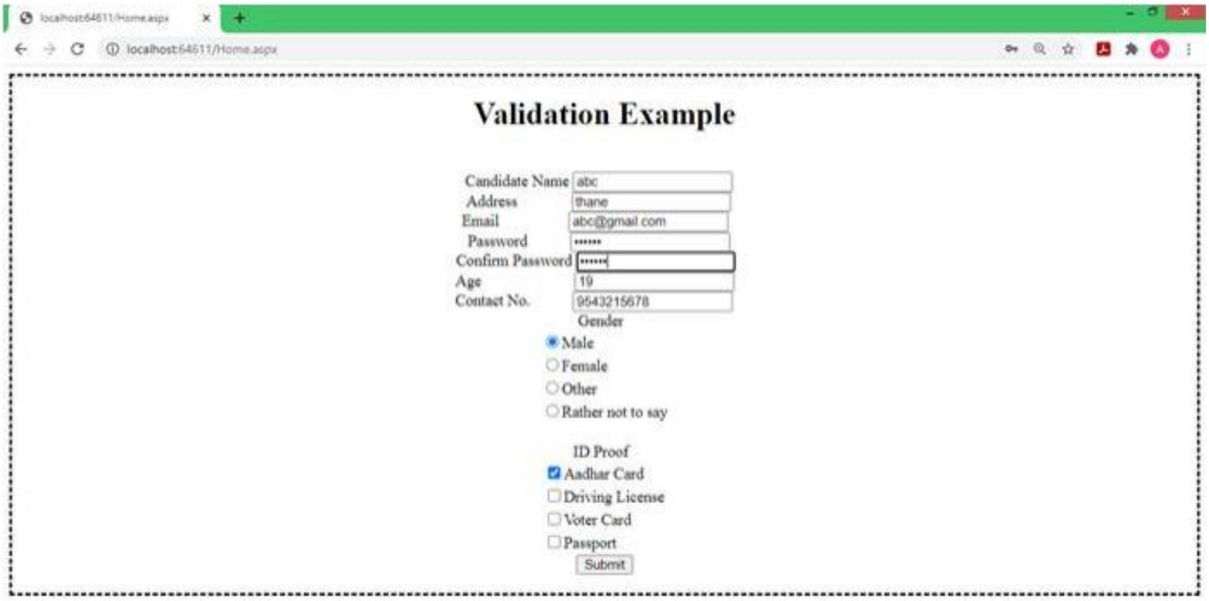


Validation Example

Candidate Name *
Address *
Email *
Password *
Confirm Password *
Age 14 Age must be >18 and age <30
Contact No. 1 Enter 10 digits

Gender
○ Male
○ Female
○ Other
○ Rather not to say
*

ID Proof
□ Aadhar Card
□ Driving License
□ Voter Card
□ Passport



Validation Example

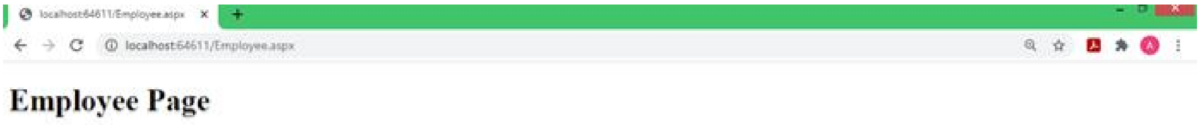
Candidate Name	abc
Address	thane
Email	abc@gmail.com
Password	••••••
Confirm Password	••••••
Age	19
Contact No.	9543215678

Gender

● Male
○ Female
○ Other
○ Rather not to say

ID Proof

☑ Aadhar Card
□ Driving License
□ Voter Card
□ Passport



Employee Page

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

Add Home.aspx

- Create a new folder named as images
- Add images to the folder
- Add the .xml file to project.
- Insert advertisement info in the xml file named AdRotator.xml
- Select AdRotator Control on the Design Page
- Select xml datasource control on the design page
- Select the data source file named AdRotator.xml

Program for Home.aspx :

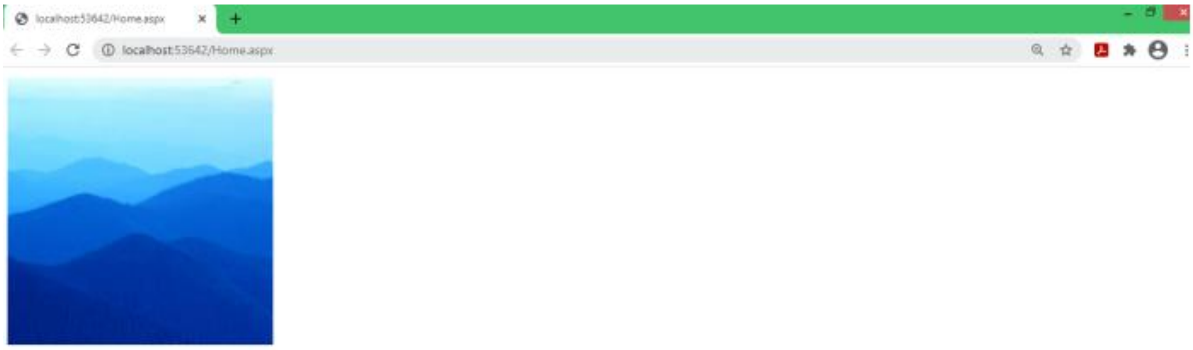
```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Home.aspx.cs"
Inherits="AdRotatorExample.Home" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
 <title></title>
</head>
<body>
```

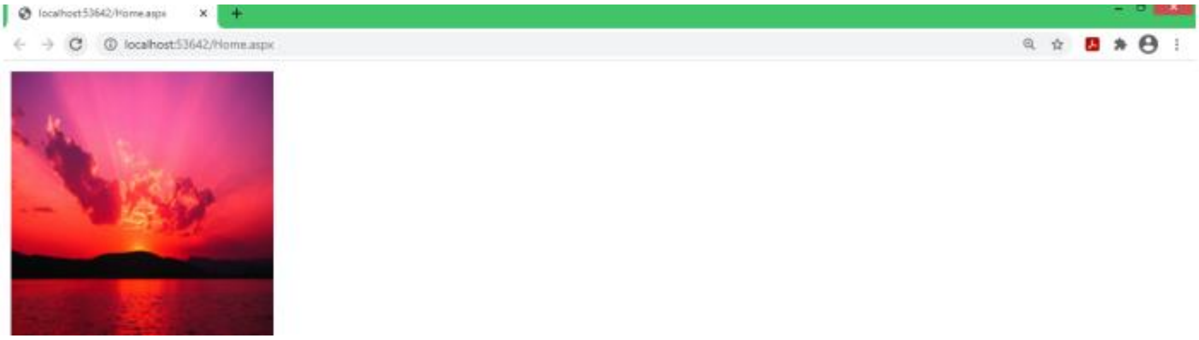
```
 <form id="form1" runat="server">
 <div>
 <asp:AdRotator ID="AdRotator1" runat="server" DataSourceID="XmlDataSource1"
 Target="_self" Width="200px" Height="200px" />
 <br />
 <asp:XmlDataSource ID="XmlDataSource1" runat="server"
 DataFile="~/Adrotator.xml"></asp:XmlDataSource>
 </div>
 </form>
</body>
</html>
```

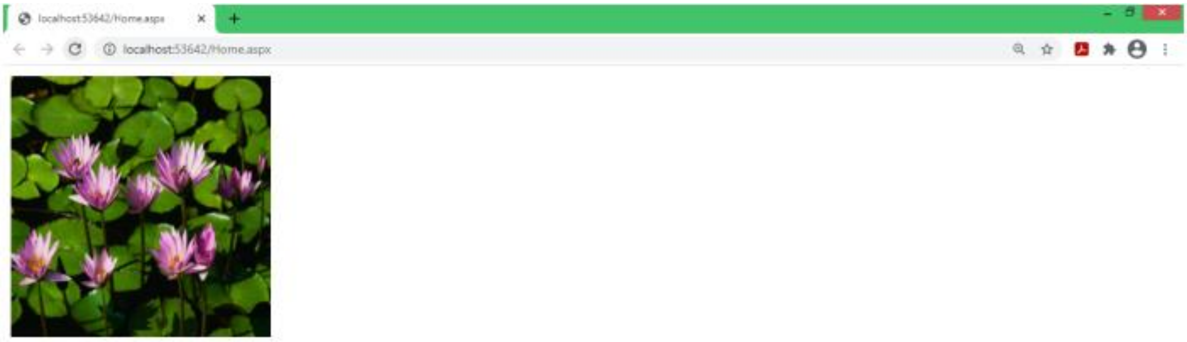
Program for AdRotator.xml page :

```
<?xml version="1.0" encoding="utf-8" ?>
<Advertisements>
 <Ad>
 <ImageUrl>~\images\Bluehills.jpg</ImageUrl>
 <NavigateUrl>Ad1.aspx</NavigateUrl>
 <AlternateText>page1</AlternateText>
 <Keyword>B</Keyword>
 <Impressions>2</Impressions>
 <Caption>This is the caption for Ad#1</Caption>
 </Ad>
 <Ad>
 <ImageUrl>~\images\Sunset.jpg</ImageUrl>
 <NavigateUrl>Ad2.aspx</NavigateUrl>
 <AlternateText>page2</AlternateText>
 <Keyword>S</Keyword>
 <Impressions>3</Impressions>
 <Caption>This is the caption for Ad#2</Caption>
 </Ad>
 <Ad>
 <ImageUrl>~\images\Waterlilies.jpg</ImageUrl>
 <NavigateUrl>Ad3.aspx</NavigateUrl>

 <AlternateText>page3</AlternateText>

 <Keyword>W</Keyword>

 <Impressions>4</Impressions>

 <Caption>This is the caption for Ad#2</Caption>

 </Ad>

</Advertisements>
```

Output :







C) Aim: Create Web Form to demonstrate use User Controls.

- Add .ascx file
- Insert some controls to the file
- Add .aspx file
- Insert Register Directive to Implementation.aspx which includes tag name and tag prefix.

Program for CustomUserControl.ascx page :

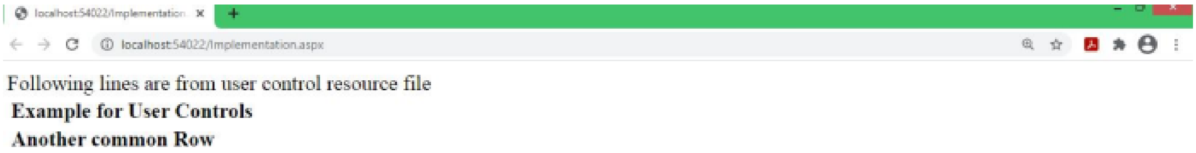
```
<%@ Control Language="C#" AutoEventWireup="true"

CodeBehind="CustomUserControl.ascx.cs"

Inherits="UserControls1.CustomUserControl" %>

<table>

<tr>

<td><b>Example for User Controls</b></td>

</tr>

<tr>

<td><b>Another common Row</b></td>

</tr>

</table>
```

Program for Implementation.aspx page :

```
<%@ Page Language="C#" AutoEventWireup="true"

CodeBehind="Implementation.aspx.cs" Inherits="UserControls1.Implementation" %>

<%@ Register Src="~/CustomUserControl.ascx" TagName="WebControl"

TagPrefix="TWeb" %>

<!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="Following lines are from user control
```

```
resource file"></asp:Label>
 <TWeb:WebControl ID="Header" runat="server"/>
 </div>
 </form>
</body>
</html>
```

Output :


localhost:54022/Implementation
localhost:54022/Implementation.aspx

Following lines are from user control resource file

Example for User Controls

Another common Row

Practical No. - 5

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

Master Page

- Add a new project as web application.
- Master page by right clicking on solution and select master page with name as site.master
- Add Styles to master page.
- Add content pages in project and link them to site.master
- Add custom content in other content pages.

Program for Site.Master page :

```
<%@ Master Language="C#" AutoEventWireup="true"
CodeBehind="Site.master.cs" Inherits="WebApplication8.Site" %>
<!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>
 <asp:ContentPlaceHolder ID="head" runat="server">
 </asp:ContentPlaceHolder>
 <style type="text/css">
 .style1
 { width: 485px; }
 .style2
 {
 width: 395px;
 }
 </style>
</head>
<body>
 <form id="form1" runat="server">
 <div style="background-color: #00FF00; height: 73px;">
 <asp:Menu ID="Menu1" runat="server"
 DataSourceID="SiteMapDataSource1"
 StaticSubMenuIndent="16px">
 <DynamicHoverStyle BackColor="Aqua" BorderColor="Fuchsia" />
 <DynamicMenuItemStyle BackColor="#0000CC" />
 <DynamicMenuStyle BackColor="Red" />
 <DynamicItemTemplate>
 <%# Eval("Text") %>
 </DynamicItemTemplate>
 </asp:Menu>
 <asp:SiteMapPath ID="SiteMapPath1" runat="server"
 ToolTip="navgation breadcrumbs">
 <NodeStyle BorderColor="#660066" />
 </asp:SiteMapPath>
 </div>
<div>
 <table style="width: 100%; height: 139px;">
 <tr> <td class="style2">
 <asp:TreeView ID="TreeView1" runat="server"
DataSourceID="SiteMapDataSource1"
 ImageSet="WindowsHelp" MaxDataBindDepth="2">
 <HoverNodeStyle Font-Underline="True" ForeColor="#6666AA" />
 <NodeStyle Font-Names="Tahoma" Font-Size="8pt"
ForeColor="Black"
 HorizontalPadding="5px" NodeSpacing="0px"
VerticalPadding="1px" />
 <ParentNodeStyle Font-Bold="False" />
 <SelectedNodeStyle BackColor="#B5B5B5" FontUnderline="False"
 HorizontalPadding="0px" VerticalPadding="0px" />
 </asp:TreeView></td>
 <td class="style1">
 <asp:ContentPlaceHolder ID="ContentPlaceHolder1"
runat="server">
 </asp:ContentPlaceHolder></td>
 <td>&nbsp;</td> </tr>
 </table></div>
 <div style="background-color: #FFFF00; height: 123px;">
 <asp:SiteMapDataSource ID="SiteMapDataSource1" runat="server"/>
</div>
</form>
</body>
</html>
```

Program for Home.aspx page :

```
<%@ Page Title="" Language="C#" MasterPageFile="~/Site.Master"
AutoEventWireup="true" CodeBehind="Home.aspx.cs"
Inherits="WebApplication8.Home" %>
<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">
</asp:Content>
<asp:Content ID="Content2" runat="server"
 contentplaceholderid="ContentPlaceHolder1">
&nbsp;&nbsp; This is Home page
</asp:Content>
```

Program for About.aspx page :

```
<%@ Page Title="" Language="C#" MasterPageFile="~/Site.Master"
AutoEventWireup="true" CodeBehind="About.aspx.cs"
Inherits="WebApplication8.About" %>
<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">
</asp:Content>
<asp:Content ID="Content2" runat="server"
 contentplaceholderid="ContentPlaceHolder1">
 This is about us page</asp:Content>
```

Program for Contact.aspx page :

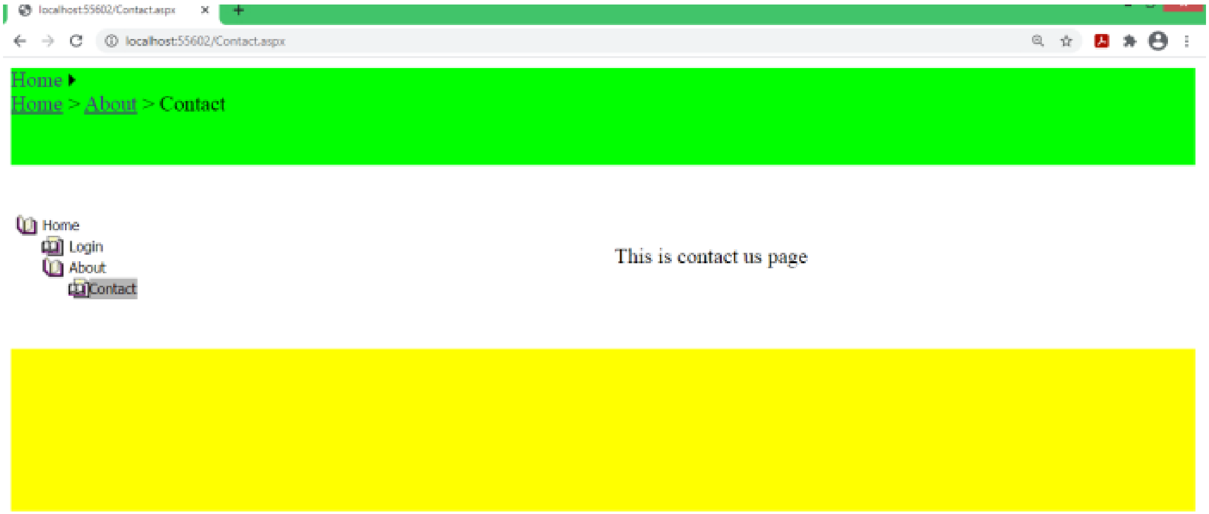
```
<%@ Page Title="" Language="C#" MasterPageFile="~/Site.Master"
```

```
AutoEventWireup="true" CodeBehind="Contact.aspx.cs"

Inherits="WebApplication8.Contact" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head"

runat="server"></asp:Content>

<asp:Content ID="Content2" runat="server"

 contentplaceholderid="ContentPlaceHolder1">

This is contact us page

</asp:Content>
```

Program for Web.Sitemap page (For Adding Navigation) :

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

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

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

 <siteMapNode title="Login" url=""/>

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

 <siteMapNode url="~/Contact.aspx" title="Contact" target="_blank"/>

 </siteMapNode>

 </siteMapNode>

</siteMap>
```

Output :



This is contact us page

B) Aim : Create a web application to demonstrate use State management 1) Cookie State

Program for Default.aspx page :

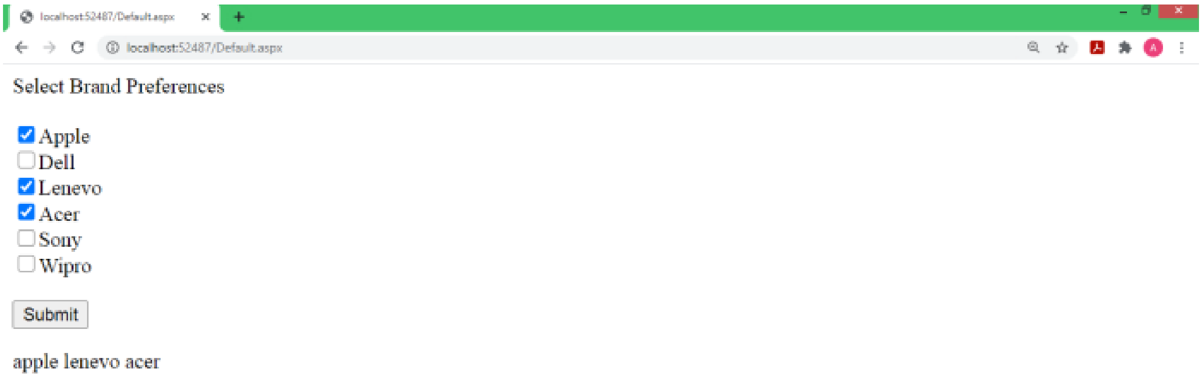
```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Default.aspx.cs"
Inherits="CookieExample.Default" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
 <title></title>
</head>
<body>
 <form id="form1" runat="server">
 <asp:Label ID="Label1" runat="server" Text="Select Brand Preferences"></asp:Label>
 <br />
 <br />
 <asp:CheckBox ID="apple" runat="server" Text="Apple" />
 <br />
 <asp:CheckBox ID="dell" runat="server" Text="Dell" />
 <br />
 <asp:CheckBox ID="lenevo" runat="server" Text="Lenevo" />
 <br />
 <asp:CheckBox ID="acer" runat="server" Text="Acer" />
 <br />
 <asp:CheckBox ID="sony" runat="server" Text="Sony" />
 <br />
 <asp:CheckBox ID="wipro" runat="server" Text="Wipro" />
 <br />
 <br />
 <asp:Button ID="Button1" runat="server" OnClick="Button1_Click" Text="Submit" />
 <p>
 <asp:Label ID="Label2" runat="server"></asp:Label>
 </p>
```

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

Program for 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;
namespace CookieExample
{
 public partial class Default : System.Web.UI.Page
 {
 protected void Page_Load(object sender, EventArgs e)
 {
 Response.Cookies["computer"].Expires = DateTime.Now.AddDays(-1);
 }
 protected void Button1_Click(object sender, EventArgs e)
 {
 Label2.Text = "";
 // --------------- Adding Coockies ---------------------//
 if (apple.Checked)
 Response.Cookies["computer"]["apple"] = "apple";
 if (dell.Checked)
 Response.Cookies["computer"]["dell"] = "dell";
 if (lenevo.Checked)
 Response.Cookies["computer"]["lenevo"] = "lenevo";
 if (acer.Checked)
 Response.Cookies["computer"]["acer"] = "acer";
 if (sony.Checked)
 Response.Cookies["computer"]["sony"] = "sony";
 if (wipro.Checked)

 Response.Cookies["computer"]["wipro"] = "wipro";

 // --------------- Fetching Cookies ----------------------//

 if (Request.Cookies["computer"].Values.ToString() != null)

 {

 if (Request.Cookies["computer"]["apple"] != null)

 Label2.Text += Request.Cookies["computer"]["apple"] + " ";

 if (Request.Cookies["computer"]["dell"] != null)

 Label2.Text += Request.Cookies["computer"]["dell"] + " ";

 if (Request.Cookies["computer"]["lenevo"] != null)

 Label2.Text += Request.Cookies["computer"]["lenevo"] + " ";

 if (Request.Cookies["computer"]["acer"] != null)

 Label2.Text += Request.Cookies["computer"]["acer"] + " ";

 if (Request.Cookies["computer"]["sony"] != null)

 Label2.Text += Request.Cookies["computer"]["sony"] + " ";

 if (Request.Cookies["computer"]["wipro"] != null)

 Label2.Text += Request.Cookies["computer"]["wipro"] + " ";

 }else Label2.Text = "Please select your choice";

 Response.Cookies["computer"].Expires = DateTime.Now.AddDays(-1);

 } } }
```

Output :



Select Brand Preferences

☑ Apple
□ Dell
☑ Lenevo
☑ Acer
□ Sony
□ Wipro

apple lenevo acer

Aim : Create a web application to demonstrate Session State.

Program for Default.aspx page :

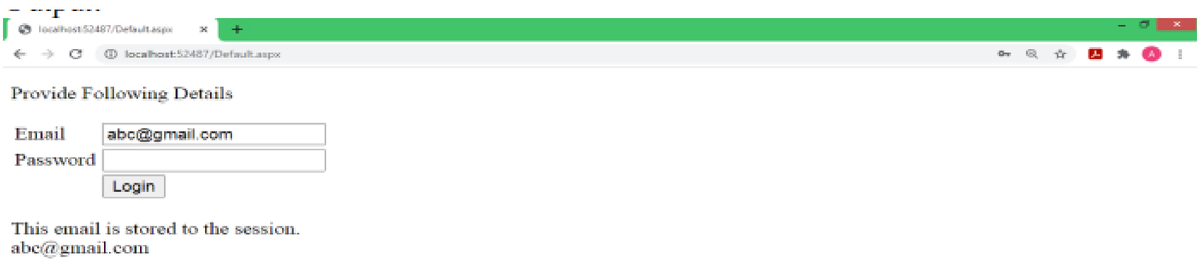
```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Default.aspx.cs"
Inherits="CookieExample.Default" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
 <title></title>
</head>
<body>
<form id="form1" runat="server">
<div>
<p>Provide Following Details</p>
<table class="auto-style1">
<tr>
<td class="auto-style2">Email</td>
<td>
<asp:TextBox ID="email" runat="server" TextMode="Email"></asp:TextBox>
</td>
</tr>
<tr>
<td class="auto-style2">Password</td>
<td>
<asp:TextBox ID="password" runat="server" TextMode="Password"></asp:TextBox>
</td>
</tr>
<tr>
<td class="auto-style2"> </td>
<td>
<asp:Button ID="login" runat="server" Text="Login" OnClick="login_Click" />
</td>
</tr>
```

```
</table>
<br />
<asp:Label ID="Label3" runat="server"></asp:Label>
<br />
<asp:Label ID="Label4" runat="server"></asp:Label>
</div>
</form>
</body>
</html>
```

Program for 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;
namespace CookieExample
{
 public partial class Default : System.Web.UI.Page
 {
 protected void Page_Load(object sender, EventArgs e)
 {
 }
 protected void login_Click(object sender, EventArgs e)
 {
 if (password.Text == "qwe123")
 {
 // Storing email to Session variable
 Session["email"] = email.Text;
 }
 // Checking Session variable is not empty
 if (Session["email"] != null)
 {
 // Displaying stored email
 Label3.Text = "This email is stored to the session.";
 Label4.Text = Session["email"].ToString();
 } } } }
```

Output:



Provide Following Details

Email abc@gmail.com

Password

Login

This email is stored to the session.
abc@gmail.com

Aim : Create a web application to demonstrate Application State.

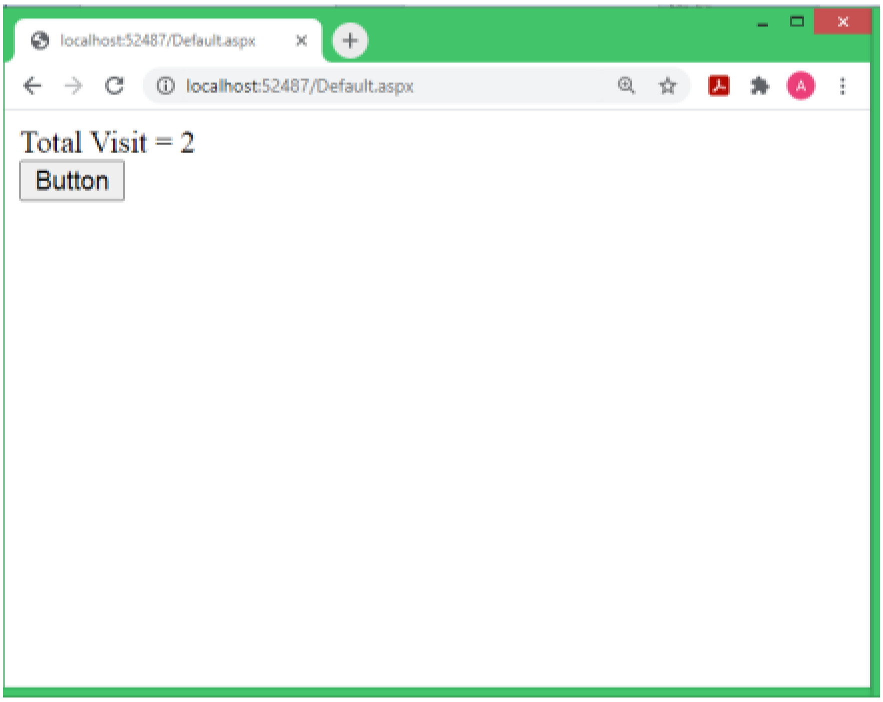
Program for Default.aspx page :

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Default.aspx.cs"
Inherits="CookieExample.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>
 <asp:Label ID="Label1" runat="server" Text="Label"></asp:Label>
 <br />
 <asp:Button ID="Button1" runat="server" onclick="Button1_Click" Text="Button" />
 </div>
 </form>
</body>
</html>
```

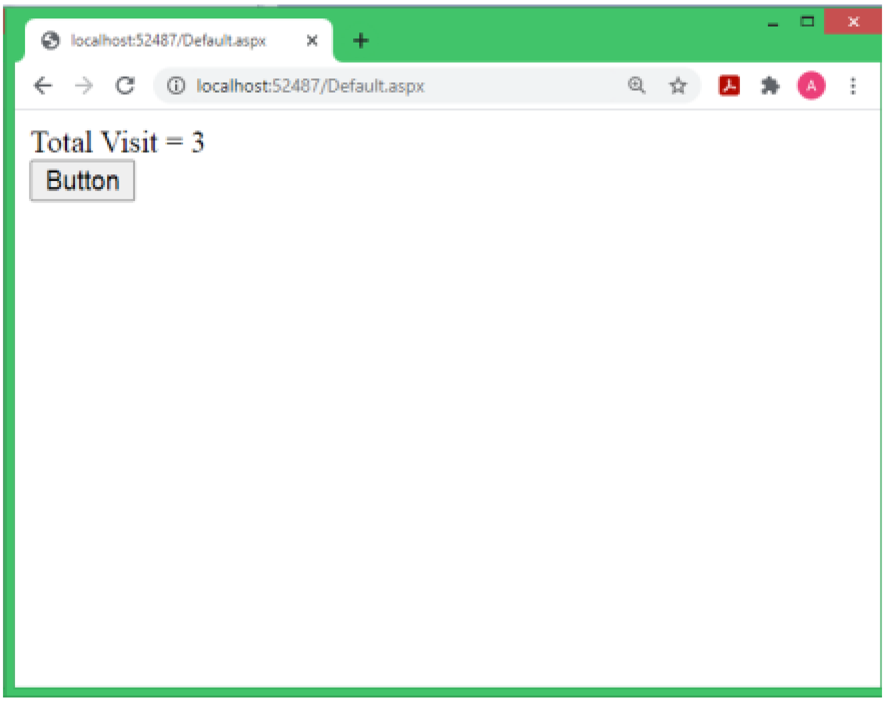
Program for Default.aspx.cs page :

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
namespace CookieExample
{
 public partial class Default : System.Web.UI.Page
 {
 protected void Page_Load(object sender, EventArgs e){
 }
protected void Button1_Click(object sender, EventArgs e)
 {
 int cnt = 0;
 if (Session["Visit"] != null)
 {
 cnt = Convert.ToInt32(Session["Visit"].ToString());
 }
 cnt = cnt + 1;
 Session["Visit"] = cnt;
 Label1.Text = "Total Visit = " + cnt.ToString();
 } } }
```

Output:



Total Visit = 2



Total Visit = 3