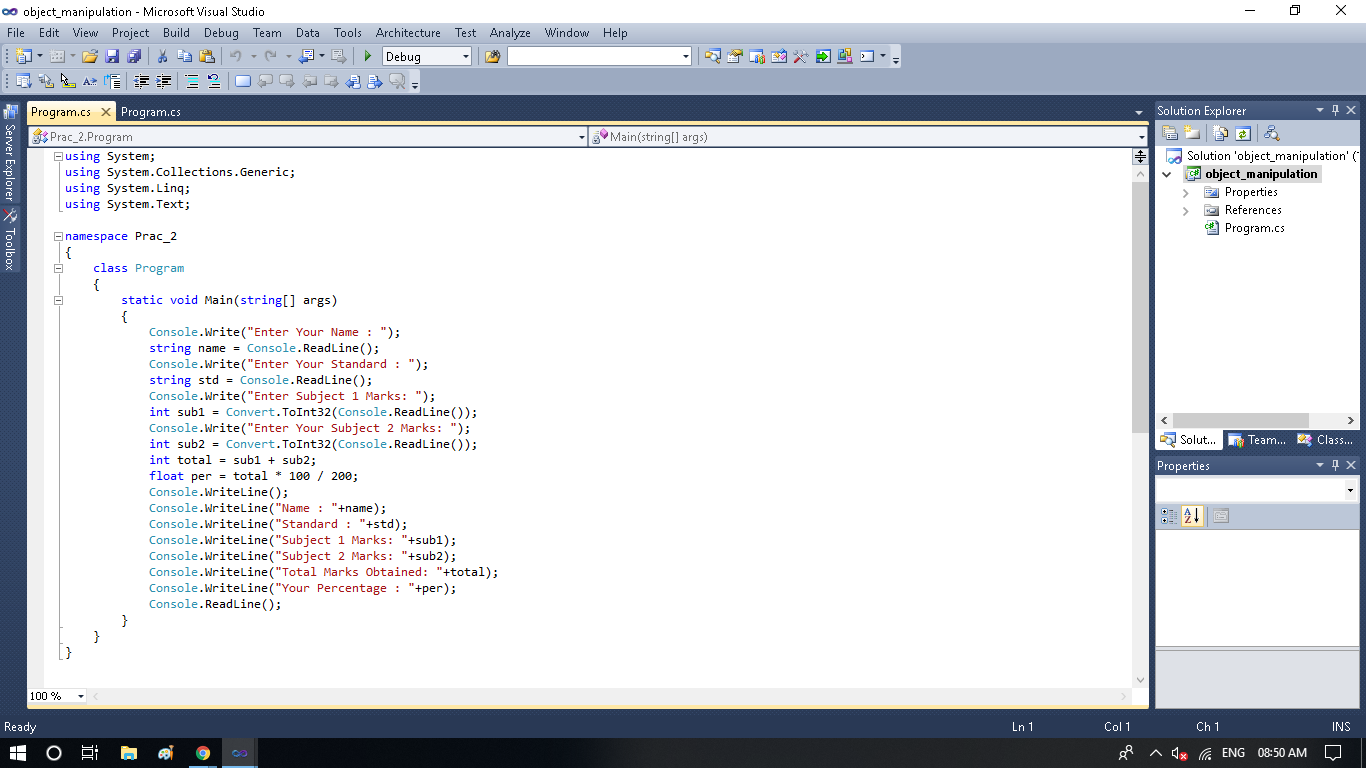
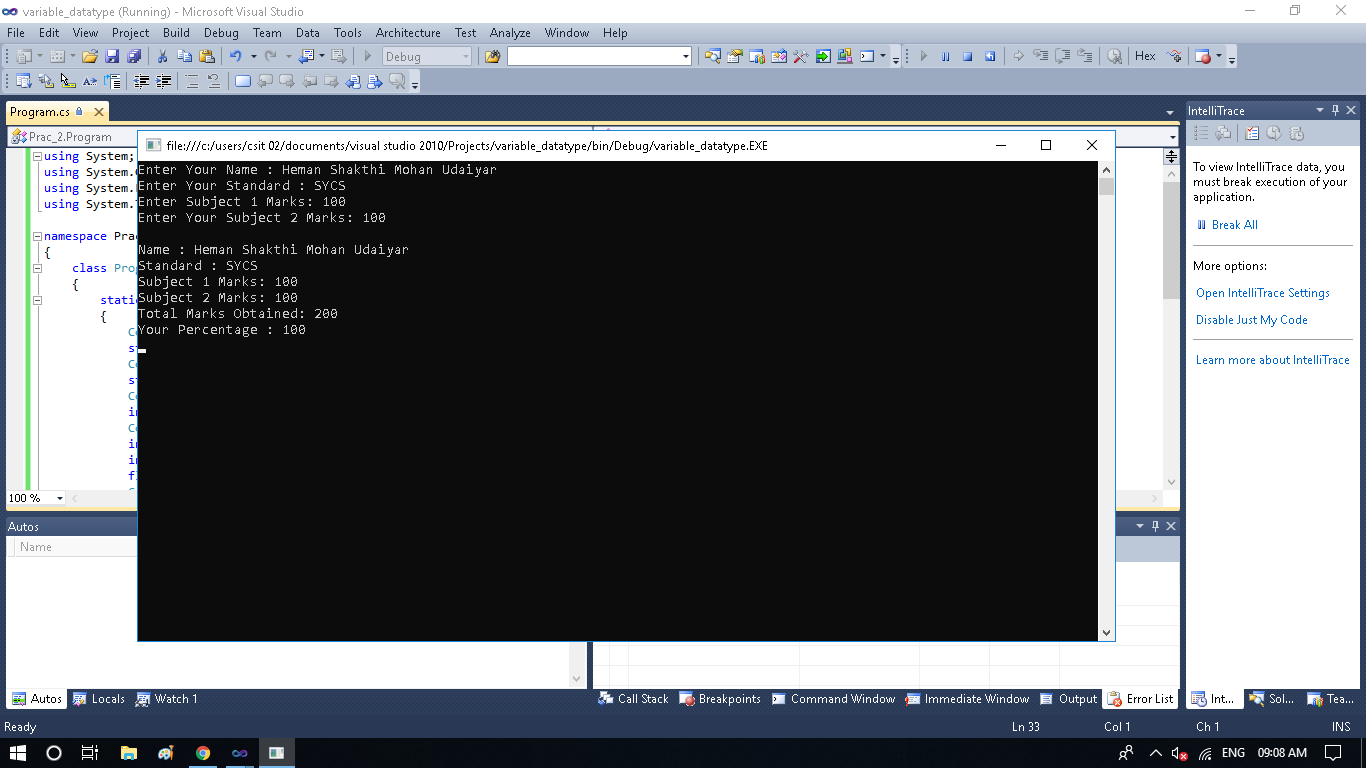
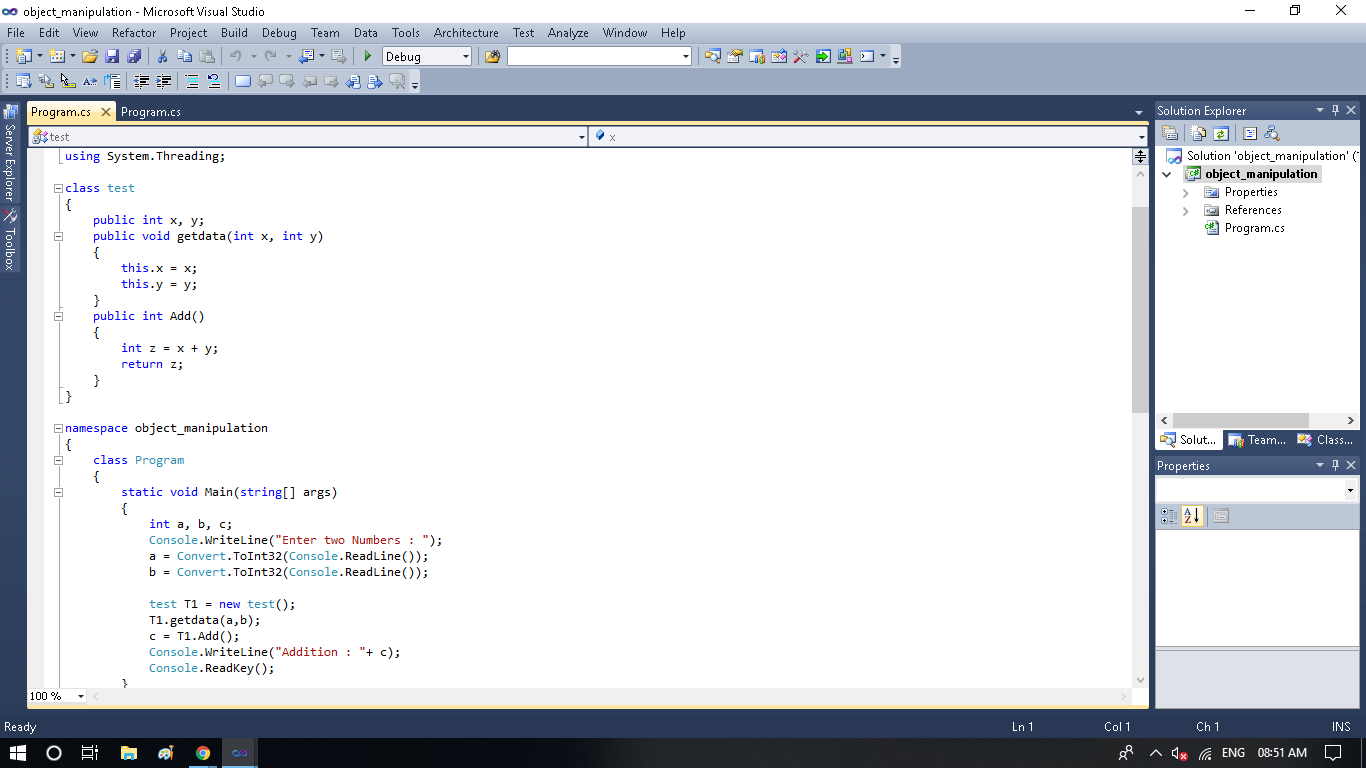
**Practical 1**

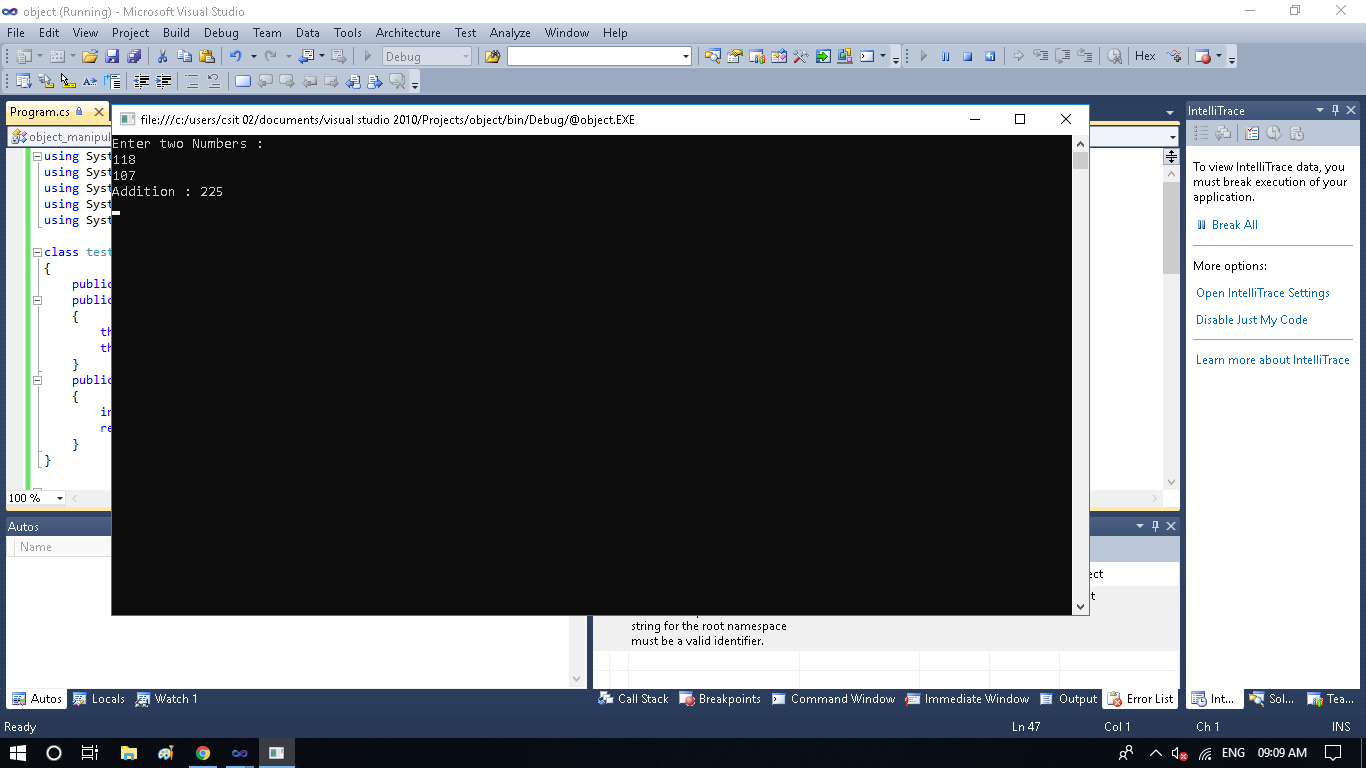
1. **Variable and Datatype**

****

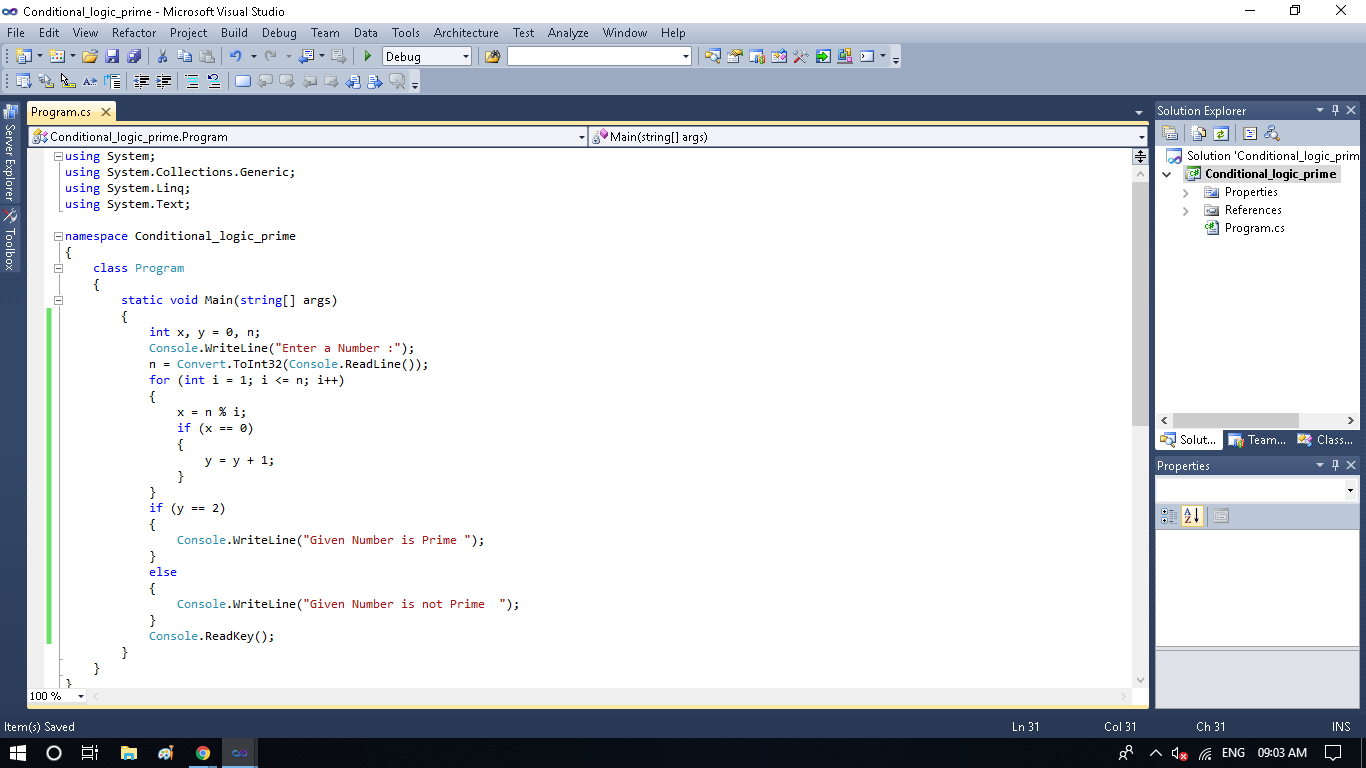
****

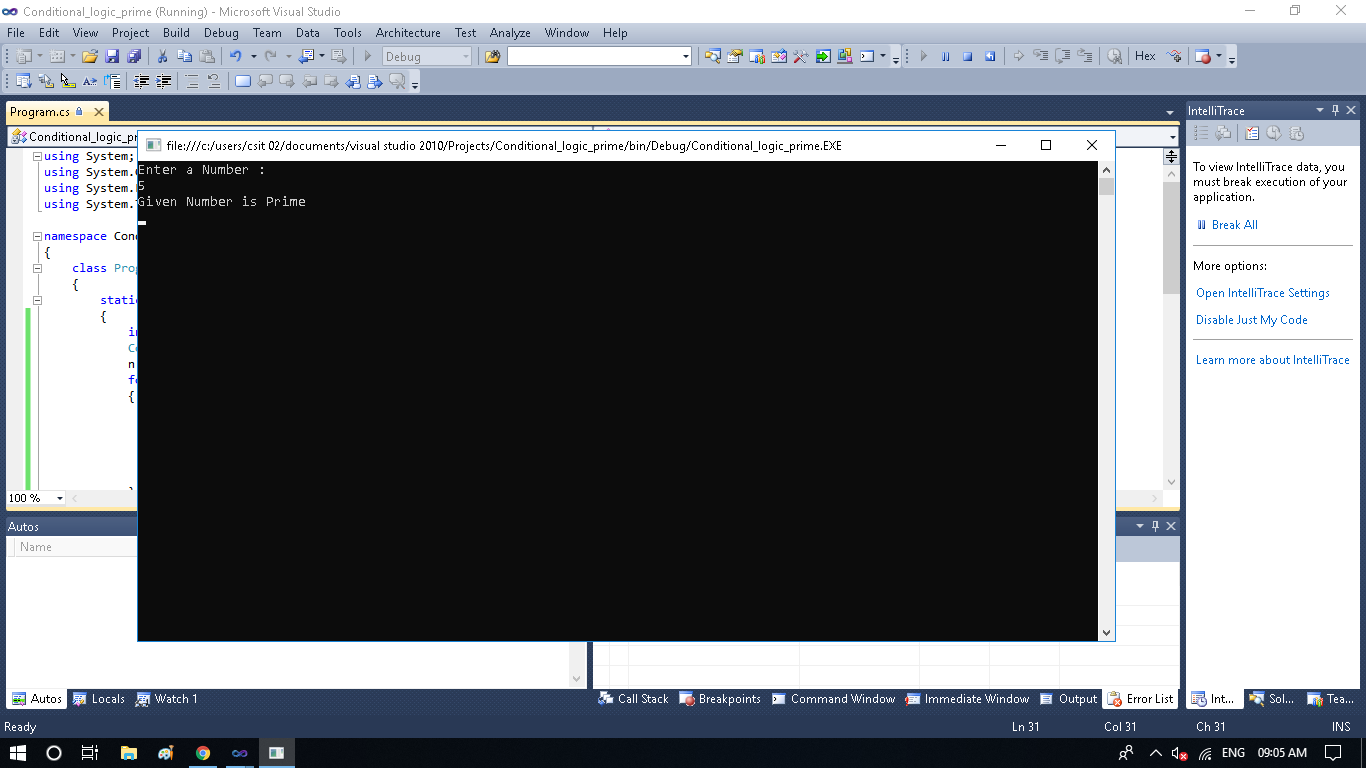
1. **Object-based Manipulation**

****

****

1. **Conditional -Logic : Prime**

****

****

**Practical No 1.4**

using System;

using System.Linq;

using System.Text;

namespace Prac\_2\_4

{

class Program

{

static void Main(string[] args)

{

Console.Write(" Enter the action to be performed: ");

Console.Write(" Press 1 for Addition ");

Console.Write(" Press 2 for Subtraction ");

Console.Write(" Press 3 for Multiplication ");

Console.Write(" Press 4 for Division ");

int action = Convert.ToInt32

(Console.ReadLine());

Console.WriteLine("Enter 1st number: ");

int input\_1 = Convert.ToInt32

(Console.ReadLine());

Console.WriteLine("Enter 2st number: ");

int input\_2 = Convert.ToInt32

(Console.ReadLine());

int result = 0;

switch (action)

{

case 1:

{

Console.Write("Addition of " + input\_1 + "and " + input\_2 + "is" + Addition(input\_1, input\_2));

break;

}

case 2:

{

Console.Write("Subtraction of " + input\_1 + "and " + input\_2 + "is" + Subtraction(input\_1, input\_2));

break;

}

case 3:

{

Console.Write("Multiplication of " + input\_1 + "and " + input\_2 + "is" + Multiplication(input\_1, input\_2));

break;

}

case 4:

{

Console.Write("Division of " + input\_1 + "and " + input\_2 + "is" + Division(input\_1, input\_2));

break;

}

default:

{

Console.WriteLine("Wrong Input ");

break;

}

Console.WriteLine("The Result is { 0 } ", result);

}

Console.ReadKey();

}

public static int Addition(int input\_1, int input\_2)

{

int result = input\_1 + input\_2;

return result;

}

public static int Subtraction(int input\_1, int input\_2)

{

int result = input\_1 + input\_2;

return result;

}

public static int Multiplication(int input\_1, int input\_2)

{

int result = input\_1 + input\_2;

return result;

}

public static int Division(int input\_1, int input\_2)

{

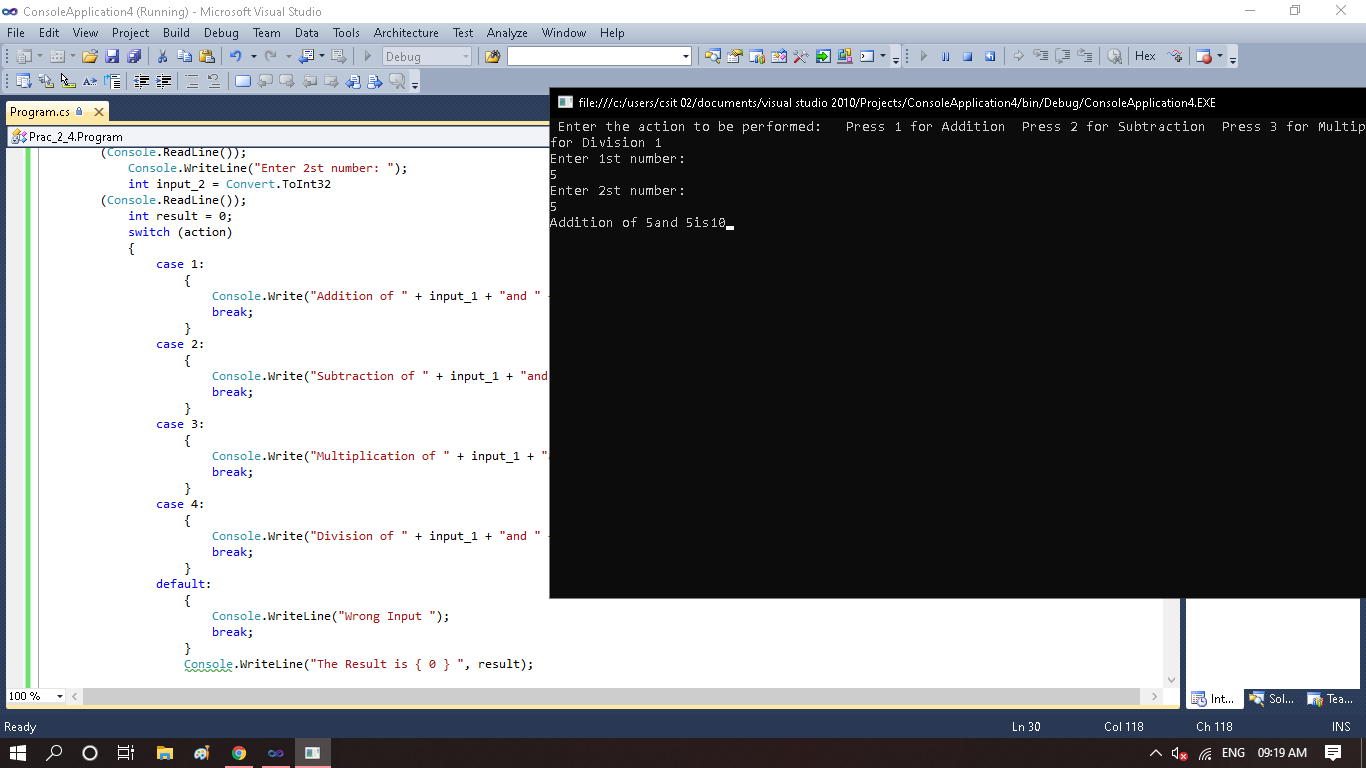
int result = input\_1 + input\_2;

return result;

}

}

}

****

**Practical No 1.5**

using System;

namespace Prac\_2\_5

{

class Program

{

static void Main(string[] args)

{

double radius, area;

Console.WriteLine("Enter the Radius: ");

radius = Convert.ToDouble

(Console.ReadLine());

area = Math.PI \* radius \* radius;

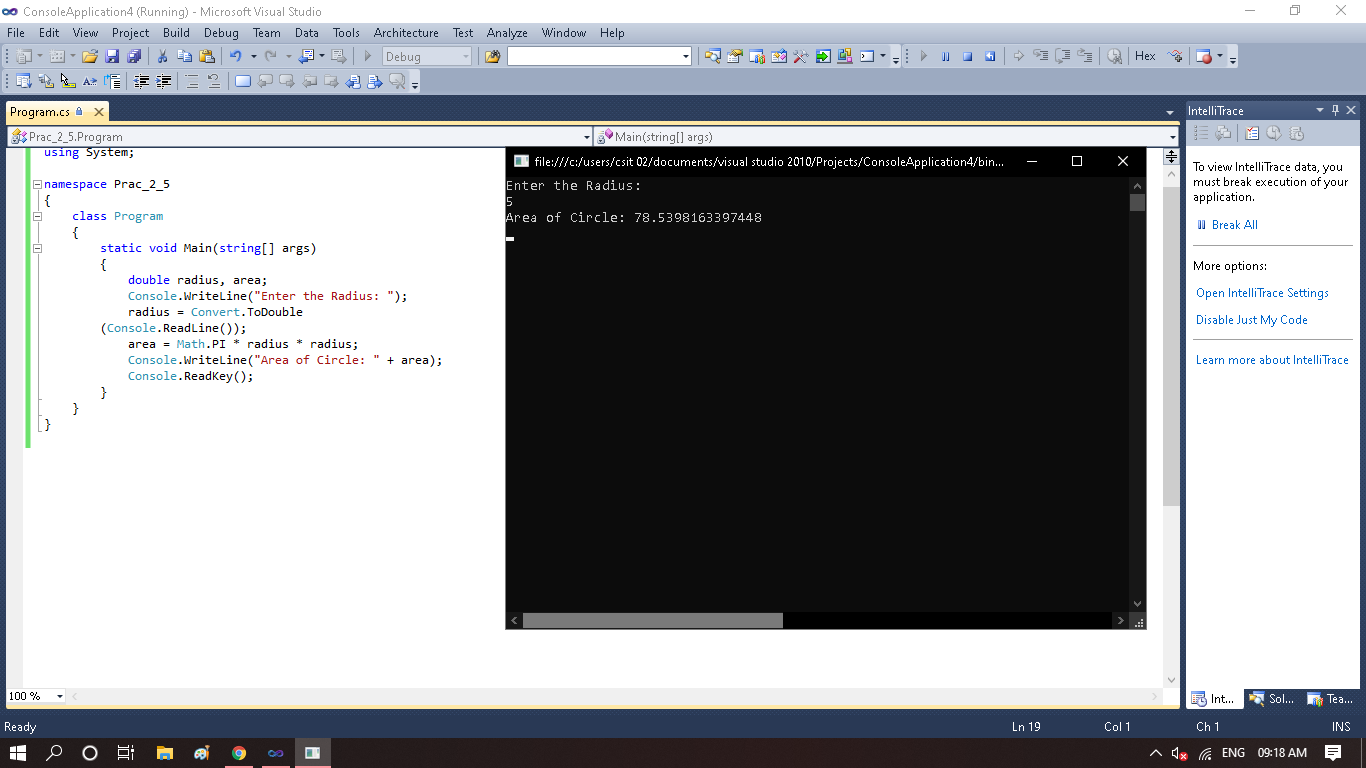
Console.WriteLine("Area of Circle: "+area);

Console.ReadKey();

}

}

}

****

**Practical 2.1**

*Constructor Overloading*

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace ConsoleApplication3

{

class Addition

{

int a , b ,sum;

public Addition()

{

Console.WriteLine("Addition of "+5+ " and "+6+" = " +(5+6));

}

public Addition (int a , int b)

{

this.a=a;

this.b=b;

sum=a+b;

Console.WriteLine("Addition of " +a+ " and " +b+ " = "+ sum);

}

}

class Program

{

static void Main(string[] args)

{

Addition Ad = new Addition();

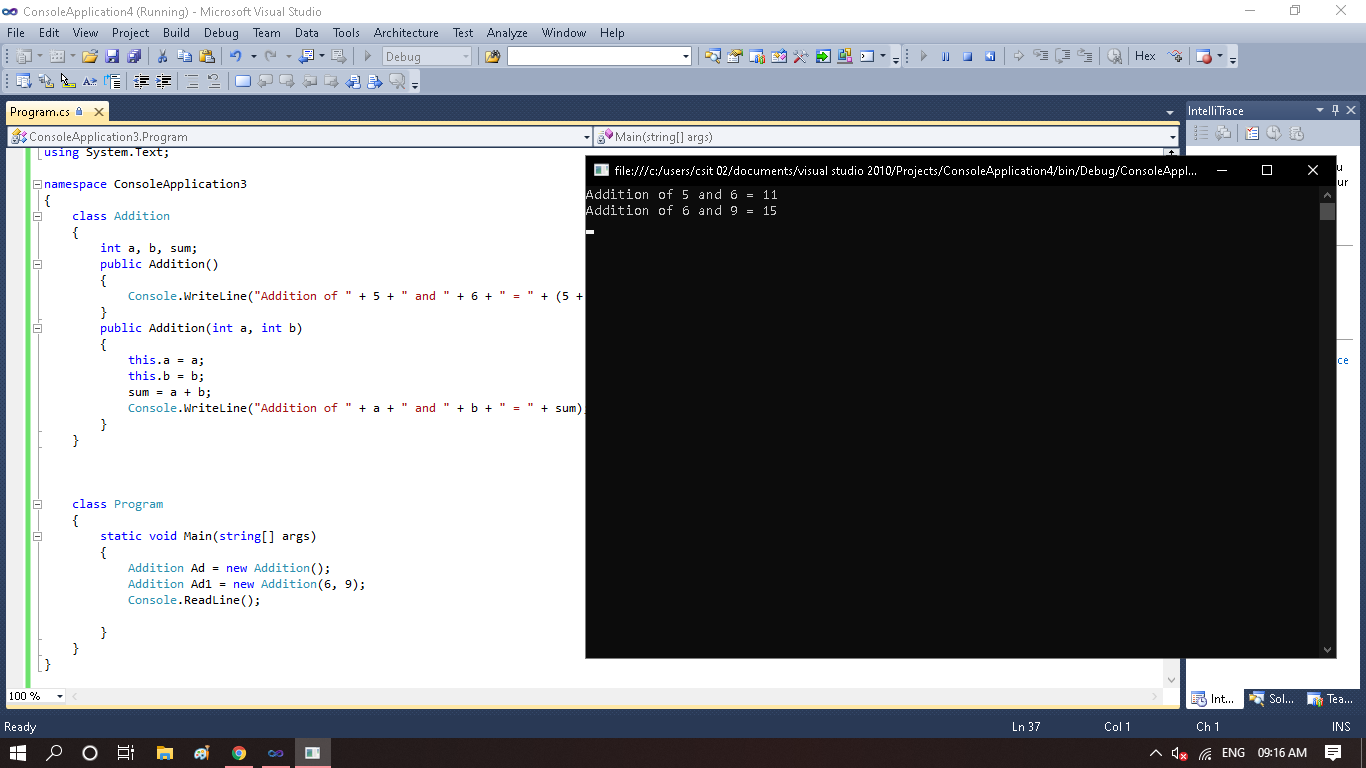
Addition Ad1 = new Addition(6,9);

Console.ReadLine();

}

}

}



**Practical 2.2**

*Inheritance*

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace ConsoleApplication7

{

public class room

{

public int length;

public int breadth;

public room()

{

Console.WriteLine("Enter length");

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

Console.WriteLine("Enter breadth");

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

}

public int area()

{

return (length \* breadth);

}

}

public class bedroom : room

{

int height;

public bedroom()

: base()

{

Console.WriteLine("Enter height");

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

}

public int volume()

{

return (length \* breadth \* height);

}

}

class inhertancetest

{

public static void Main(string[] args)

{

bedroom room1 = new bedroom();

int x = room1.area();

Console.WriteLine("Area = " + x);

int y = room1.volume();

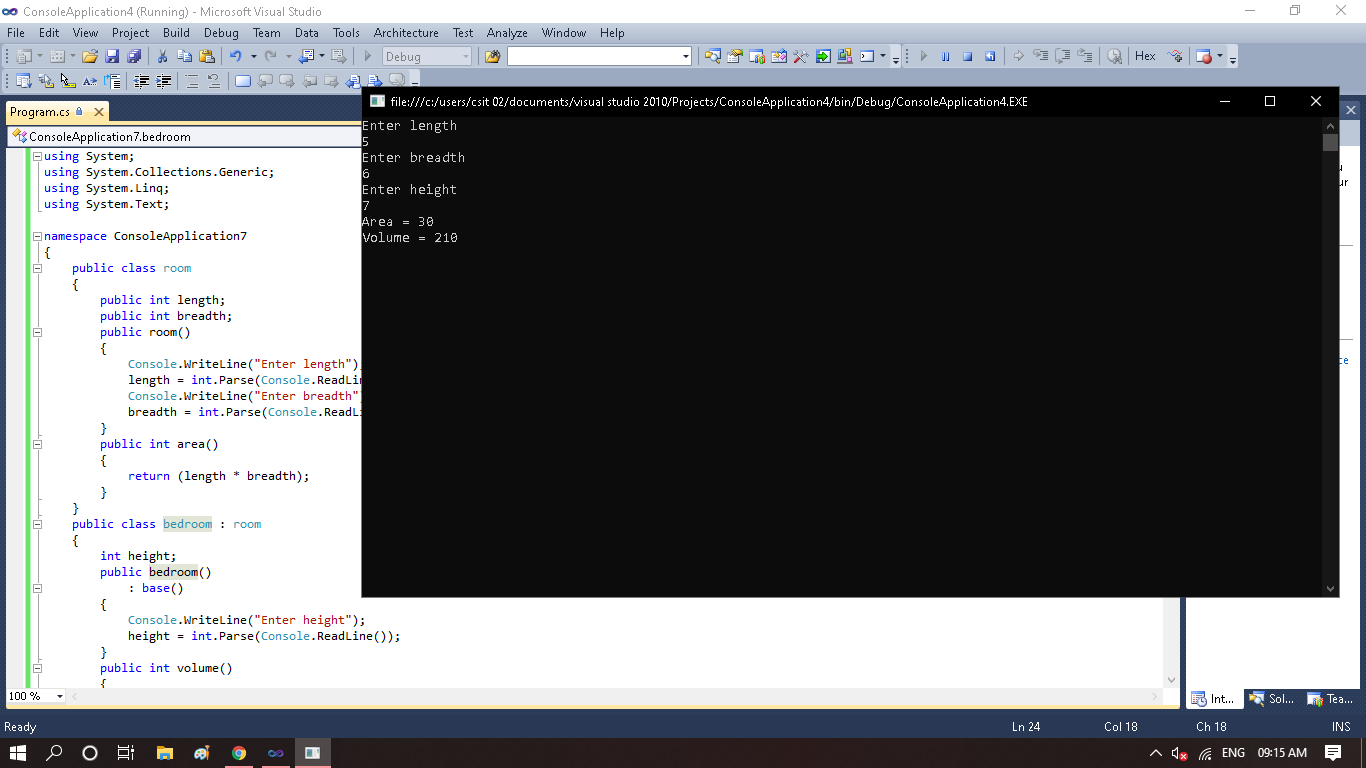
Console.WriteLine("Volume = " + y);

Console.ReadKey();

}

}

}



**Practical 2.3**

*Namespace*

using System;

using GroupA = Group.SubGroupA;

using GroupB = Group.SubGroupB;

namespace Practical\_2

{

class Program

{

static void Main(string[] args)

{

GroupA.classGroup.print();

GroupB.classGroup.print();

Console.ReadKey();

}

}

}

namespace Group

{

namespace SubGroupA

{

class classGroup

{

public static void print()

{

Console.WriteLine("Hello Namespace 1");

}

}

}

}

namespace Group

{

namespace SubGroupB

{

class classGroup

{

public static void print()

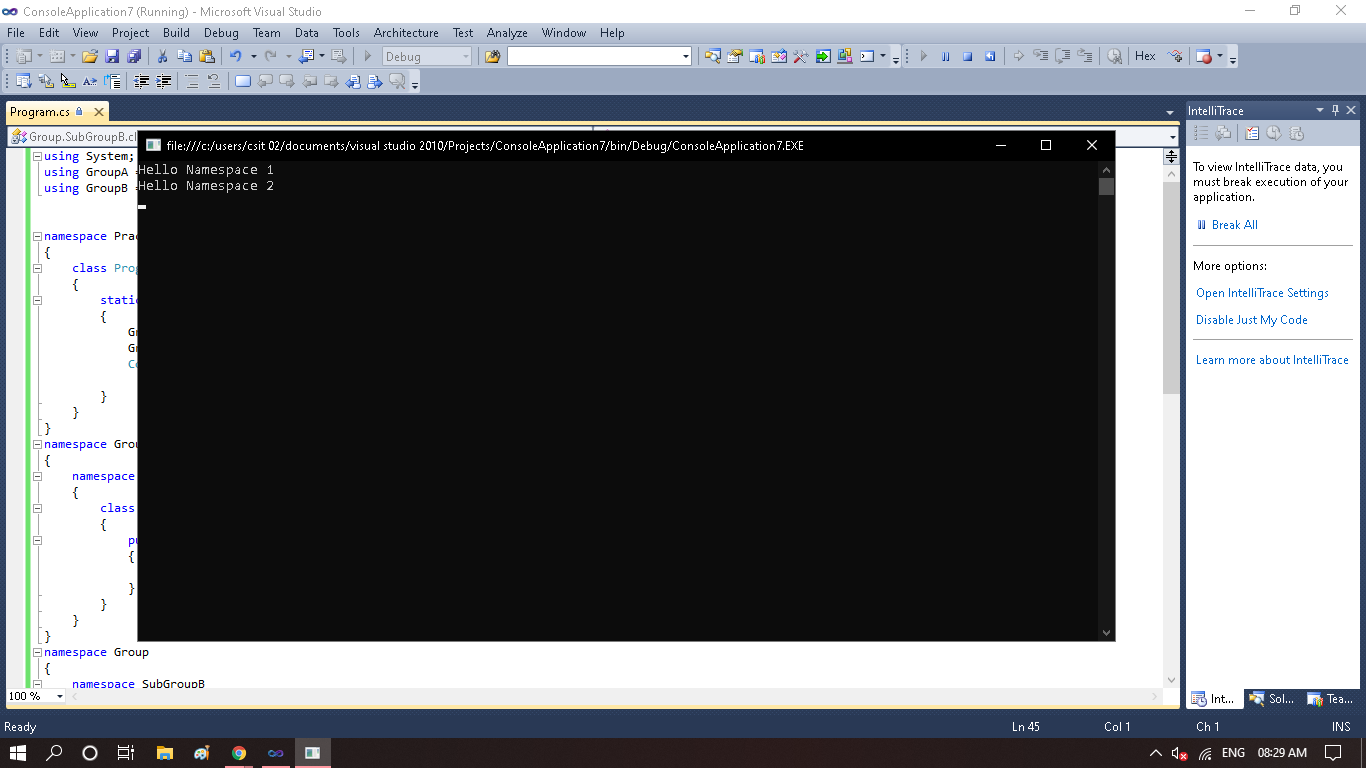
{

Console.WriteLine("Hello Namespace 2");

}

}

} }



**Practical 2.4**

*Factorial*

using System;

namespace Practical\_2\_4

{

class Program

{

static void Main(string[] args)

{

int i, fact = 1, number;

Console.Write("Enter any Number : ");

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

for (i=1;i<=number;i++)

{

fact = fact \* i;

}

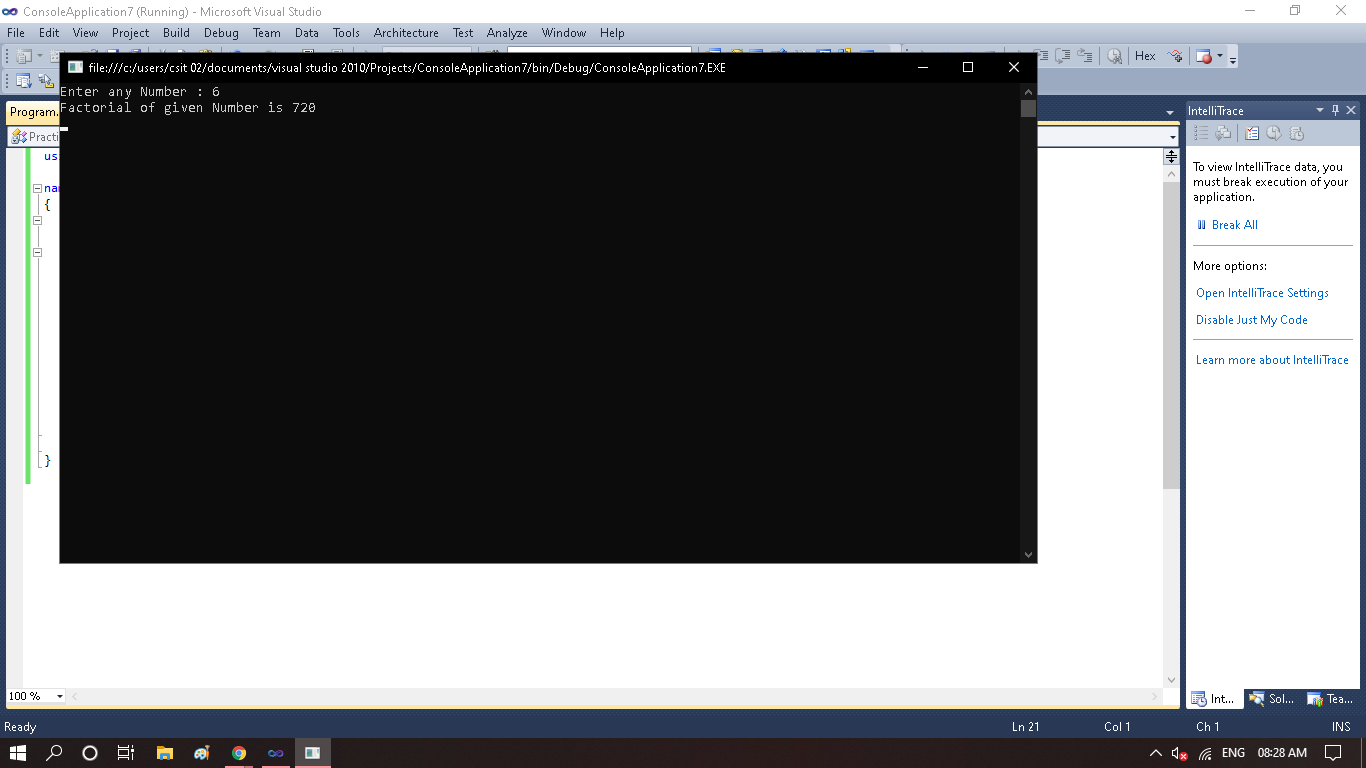
Console.WriteLine("Factorial of given Number is " + fact);

Console.ReadKey();

}

}

}



**Practical 2.5**

*CallbyValue*

using System;

namespace CallbyValue

{

class Program

{

public void Show(int val)

{

val \*= val;

Console.WriteLine("Value inside the show function " + val);

}

static void Main(string[] args)

{

int val = 50;

Program program = new Program();

Console.WriteLine("Value before Calling the function " + val);

program.Show(val);

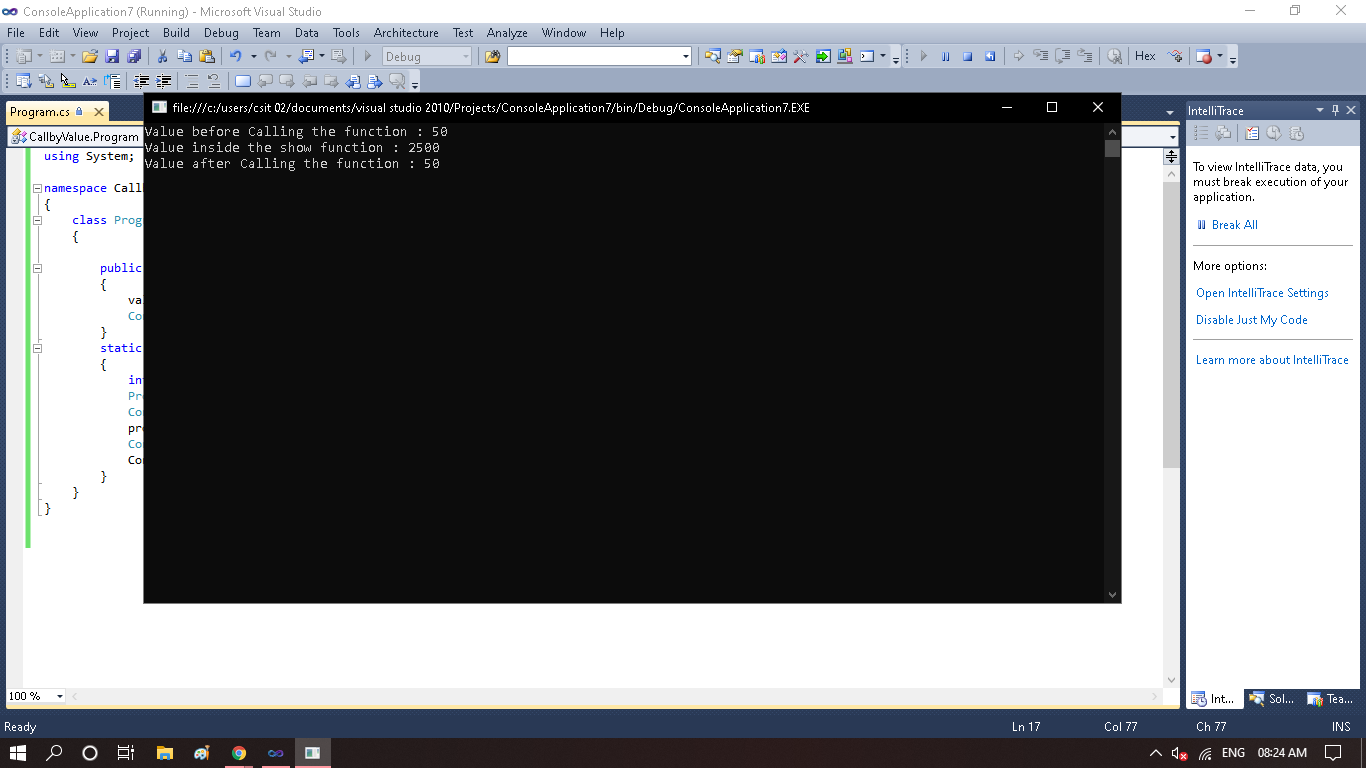
Console.WriteLine("Value after Calling the function " + val);

Console.ReadKey();

}

}

}



**Practical 2.6**

*CallbyReference*

using System;

namespace CallbyReference

{

class Program

{

public void Show(ref int val)

{

val \*= val;

Console.WriteLine("Value inside the show function " + val);

}

static void Main(string[] args)

{

int val = 50;

Program program = new Program();

Console.WriteLine("Value before Calling the function " + val);

program.Show(ref val);

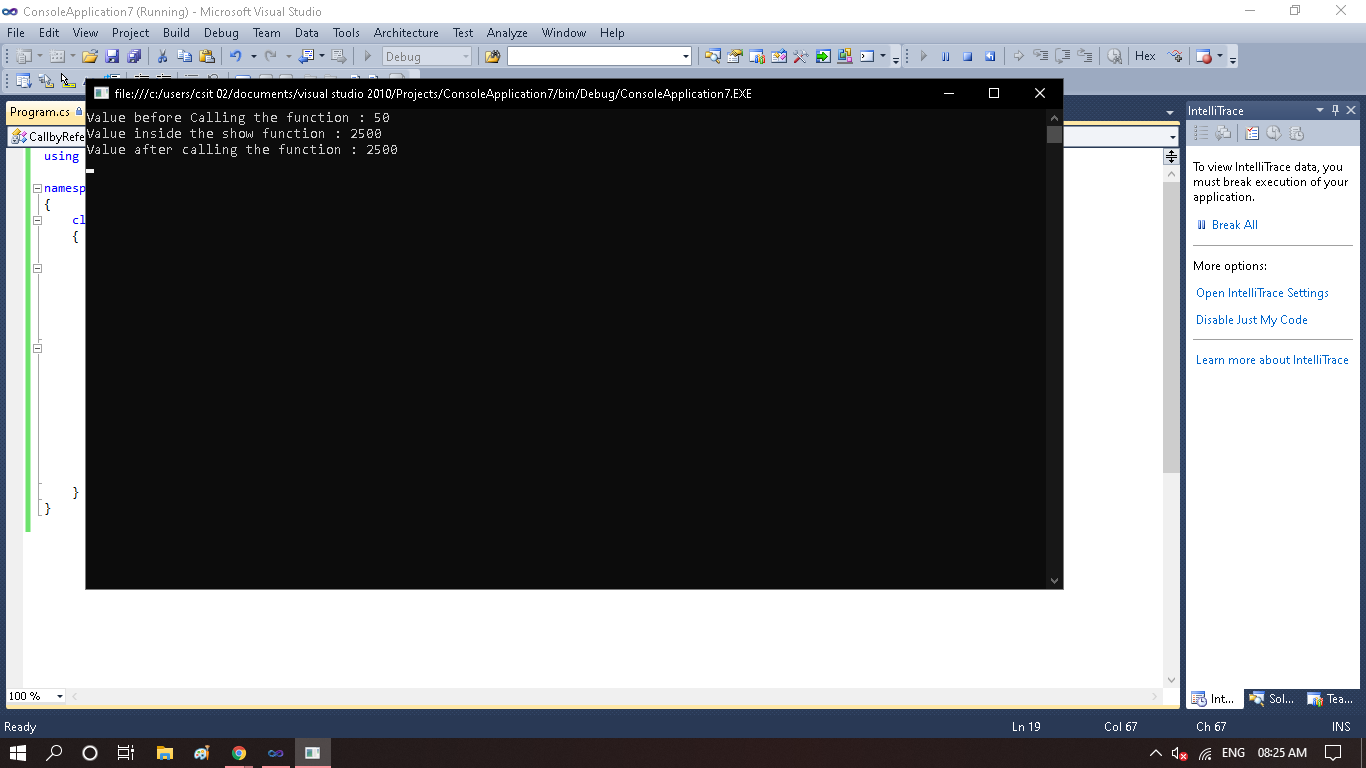
Console.WriteLine("Value after calling the function "+val);

Console.ReadKey();

}

}

}



**Practical 2.7**

*Even Or Odd*

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace EvenorOdd

{

class Program

{

static void Main(string[] args)

{

int i;

Console.Write("Enter a Number : ");

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

if (i % 2 == 0)

{

Console.Write("Entered Number is an Even Number");

Console.Read();

}

else

{

Console.Write("Entered Number is an Odd Number");

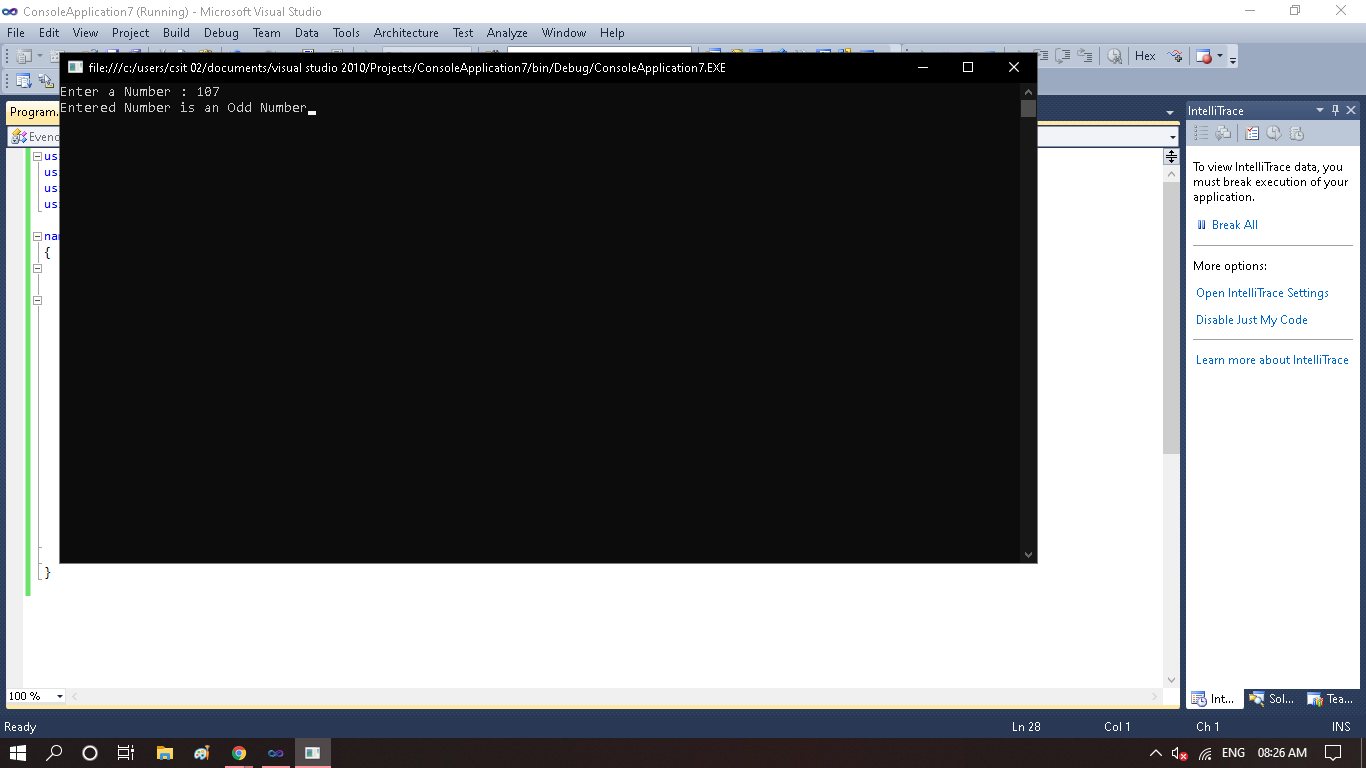
Console.Read();

}

}

}

}



**Practical 2.8**

*Fibonacci Series*

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace fibonaciseries

{

class Program

{

static void Main(string[] args)

{

int i, count, n1 = 0, n2 = 1, n3 = 0;

Console.Write("Enter the Limit : ");

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

Console.WriteLine(n1);

Console.WriteLine(n2);

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

{

n3 = n1 + n2;

Console.WriteLine(n3);

n1 = n2;

n2 = n3;

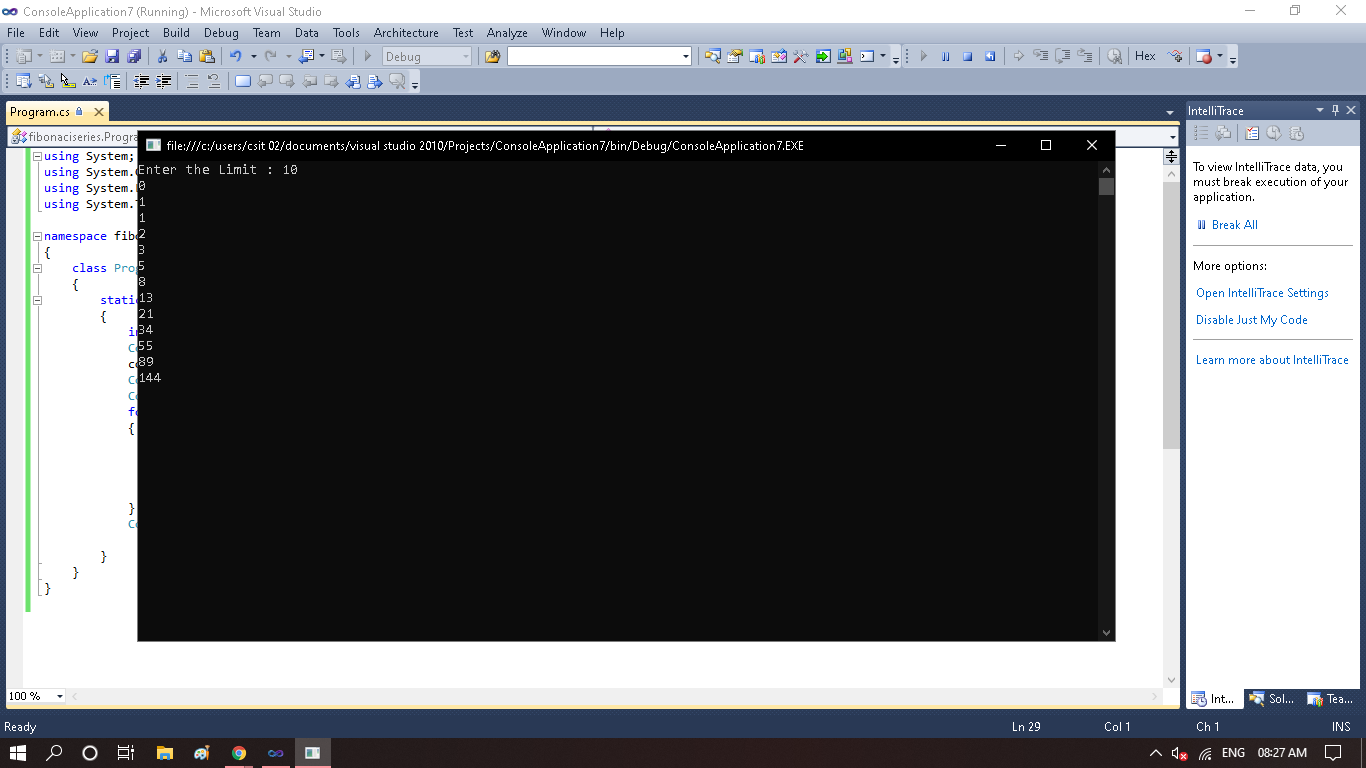
}

Console.ReadLine();

}

}

}



**Practical 2.9**

*Multiplication -CallbyValue*

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace ConsoleApplication8

{

class Program

{

public void Multiply(int num1,int num2)

{

int num3 = num1\*num2;

Console.Write("The Multiplication of numbers is "+num3);

}

static void Main(string[] args)

{

int num1 ,num2;

Console.Write("Enter the First number :");

num1 = int.Parse(Console.ReadLine());

Console.Write("Enter the First number :");

num2 = int.Parse(Console.ReadLine());

Program program = new Program();

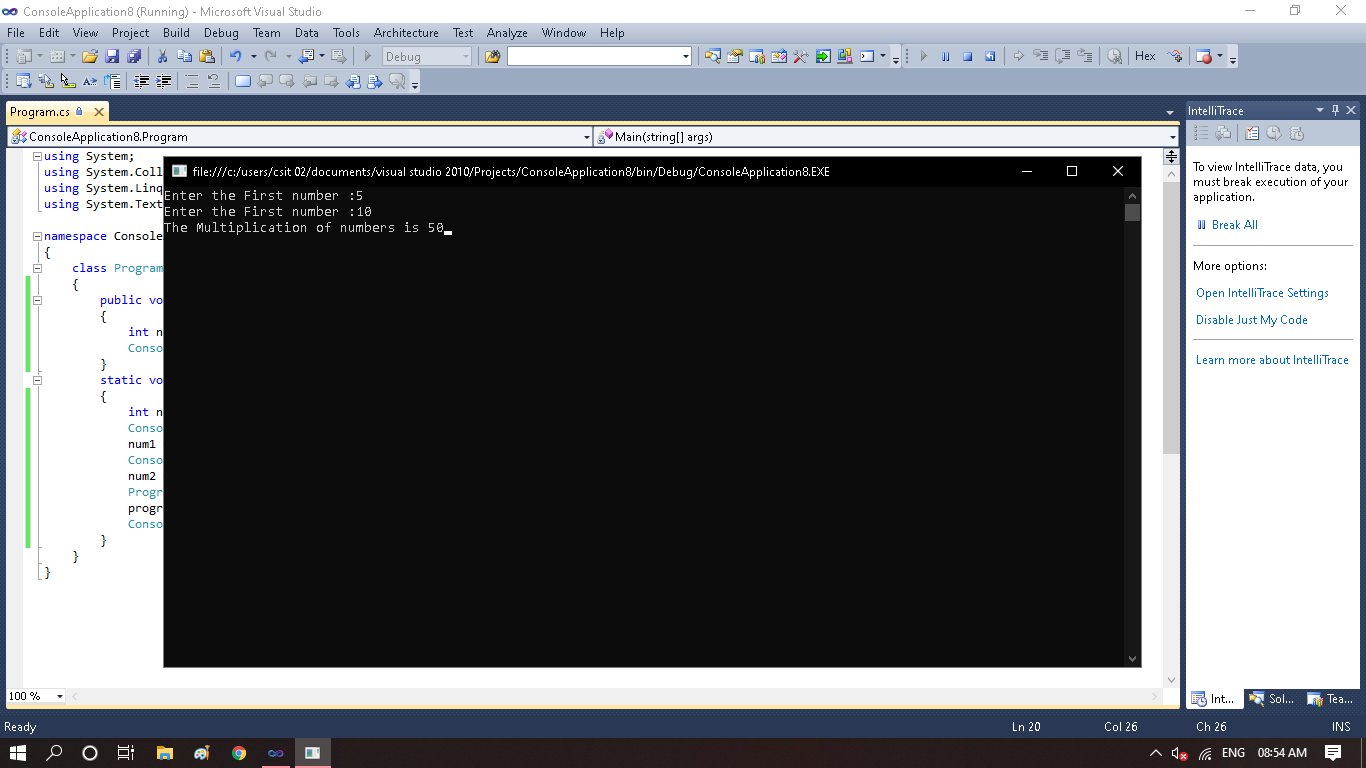
program.Multiply(num1,num2);

Console.ReadKey();

}

}

}



**Practical 2.10**

*Write a Code for swapping of 2 numbers using temporary variable by using call by Reference*

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace ConsoleApplication8

{

class Program

{

static void SwapNum(ref int x,ref int y)

{

int tempswap = x;

x = y;

y = tempswap;

}

static void Main(string[] args)

{

int a,b;

Console.Write("Enter the value of A :");

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

Console.Write("Enter the value of B :");

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

Console.WriteLine("Value of A and B before Swapping : ");

Console.WriteLine("A = " + a);

Console.WriteLine("B = " + b);

SwapNum(ref a, ref b);

Console.WriteLine("Value of A and B after Swapping : ");

Console.WriteLine("A = " + a);

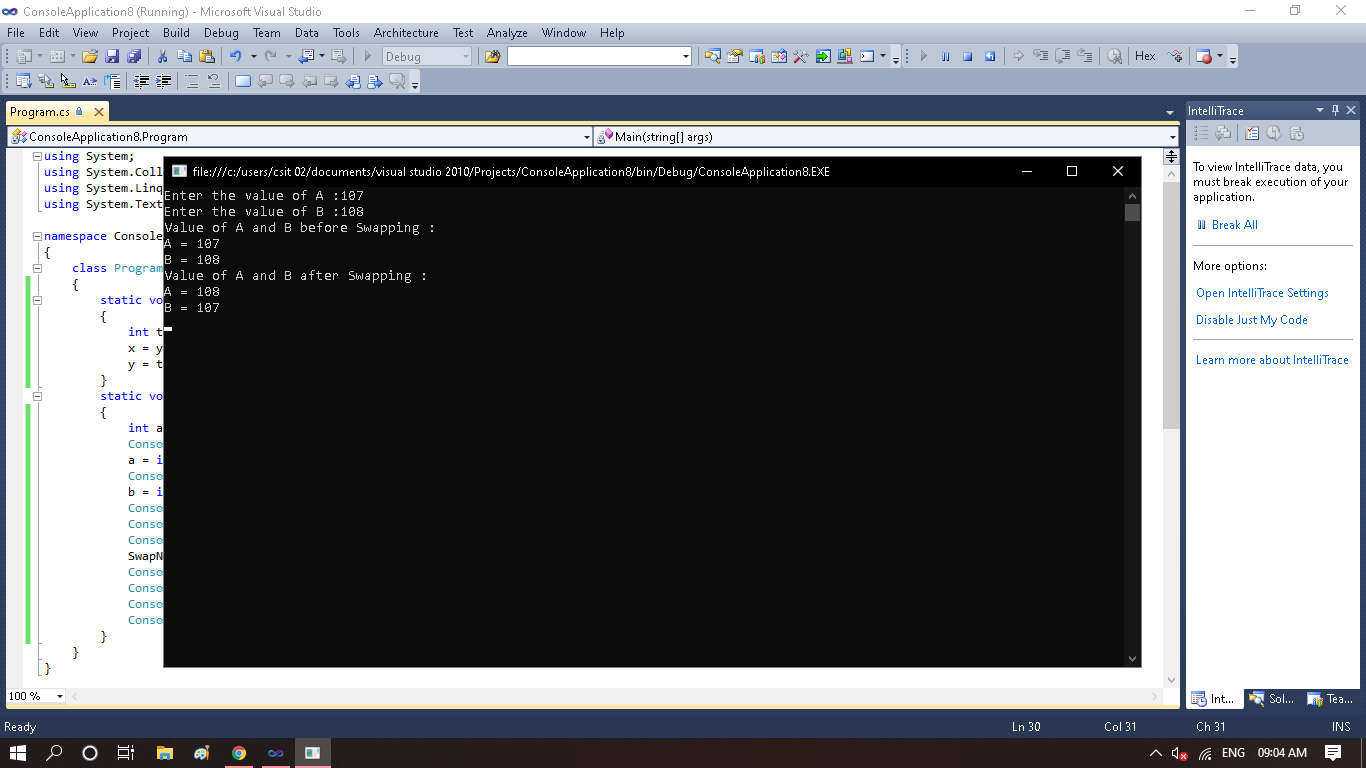
Console.WriteLine("B = " + b);

Console.ReadKey();

}

}

}



**Practical 3**

*Web Application To Create Dropdownlist with different color and change label according to selected item of dropdownList.(autoPastBack)*

**Default.aspx**

<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Default.aspx.cs" Inherits="WebApplication1.\_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:DropDownList ID="DropDownList1" runat="server" AutoPostBack="True" Height="20px" Width="150px" OnSelectedIndexChanged= "DropDownList1\_SelectedIndexChanged">

<asp:ListItem>RED</asp:ListItem>

<asp:ListItem>GREEN</asp:ListItem>

<asp:ListItem>BLUE</asp:ListItem>

<asp:ListItem>BLACK</asp:ListItem>

<asp:ListItem>YELLOW</asp:ListItem>

<asp:ListItem>BROWN</asp:ListItem>

</asp:DropDownList>

<div>

</div>

<p>

&nbsp;</p>

<p>

&nbsp;</p>

<p>

&nbsp;</p>

<p>

<asp:Label ID="Label1" runat="server" Text="HEYATHI" Font-Size="XX-Large"></asp:Label>

</p>

</form>

</body>

</html>

**Default.aspx.cs**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace WebApplication1

{

public partial class \_Default : System.Web.UI.Page

{

protected void Page\_Load(object sender, EventArgs e)

{

}

protected void DropDownList1\_SelectedIndexChanged(object sender, EventArgs e)

{

if (DropDownList1.SelectedIndex == 0)

{

Label1.ForeColor = System.Drawing.Color.Red;

}

if (DropDownList1.SelectedIndex == 1)

{

Label1.ForeColor = System.Drawing.Color.Green;

}

if (DropDownList1.SelectedIndex == 2)

{

Label1.ForeColor = System.Drawing.Color.Blue;

}

if (DropDownList1.SelectedIndex == 3)

{

Label1.ForeColor = System.Drawing.Color.Black;

}

if (DropDownList1.SelectedIndex == 4)

{

Label1.ForeColor = System.Drawing.Color.Yellow;

}

if (DropDownList1.SelectedIndex == 5)

{

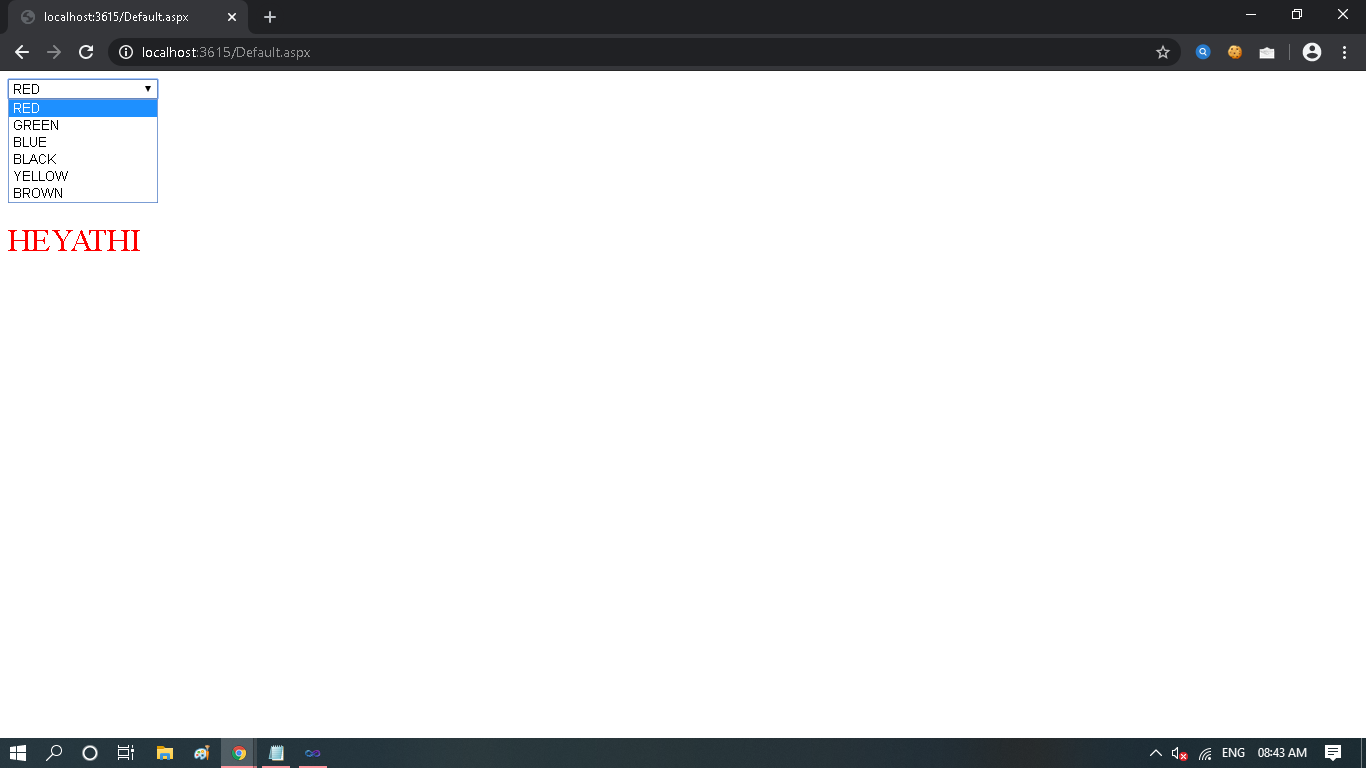
Label1.ForeColor = System.Drawing.Color.Brown;

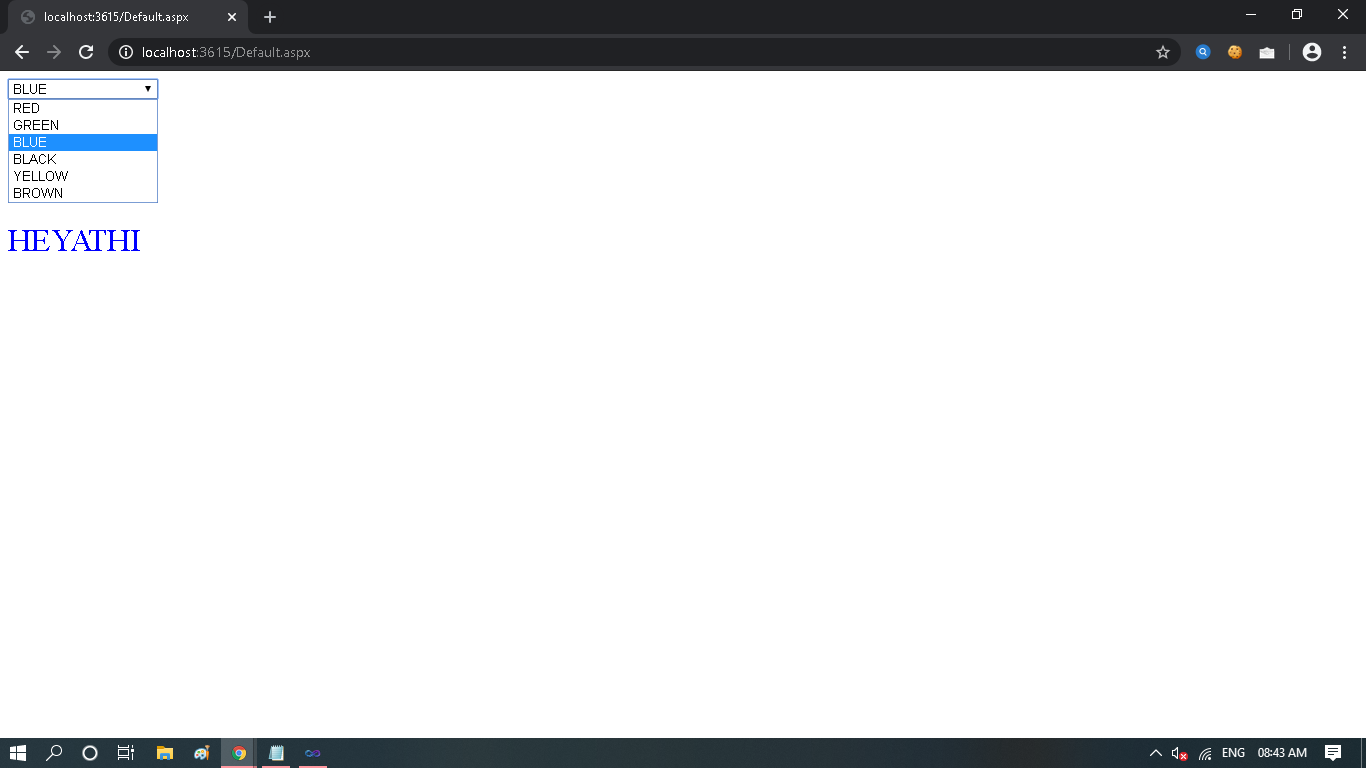
}

}

}

}





**Practical 4**

*Create Web Application to display different Advertisement .*

**XMLFile1.xml**

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

<Advertisement>

<Ad>

<ImageUrl>pic1.jpeg</ImageUrl>

<NavigateUrl>https://www.instagram.com/\_heyathi/</NavigateUrl>

<Impression>19</Impression>

<Keyword>Flower</Keyword>

</Ad>

<Ad>

<ImageUrl>pic2.jpg</ImageUrl>

<NavigateUrl>https://www.instagram.com/\_heyathi/</NavigateUrl>

<Impression>19</Impression>

<Keyword>Flower</Keyword>

</Ad>

<Ad>

<ImageUrl>pic3.jpg</ImageUrl>

<NavigateUrl>https://www.instagram.com/\_heyathi/</NavigateUrl>

<Impression>19</Impression>

<Keyword>Flower</Keyword>

</Ad>

</Advertisement>

**Default.aspx**

<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Default.aspx.cs" Inherits="MvcApplication1.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>

</div>

<asp:AdRotator ID="AdRotator1" runat="server" DataSourceID="XmlDataSource1" />

<asp:XmlDataSource ID="XmlDataSource1" runat="server" DataFile="~/XMLFile1.xml">

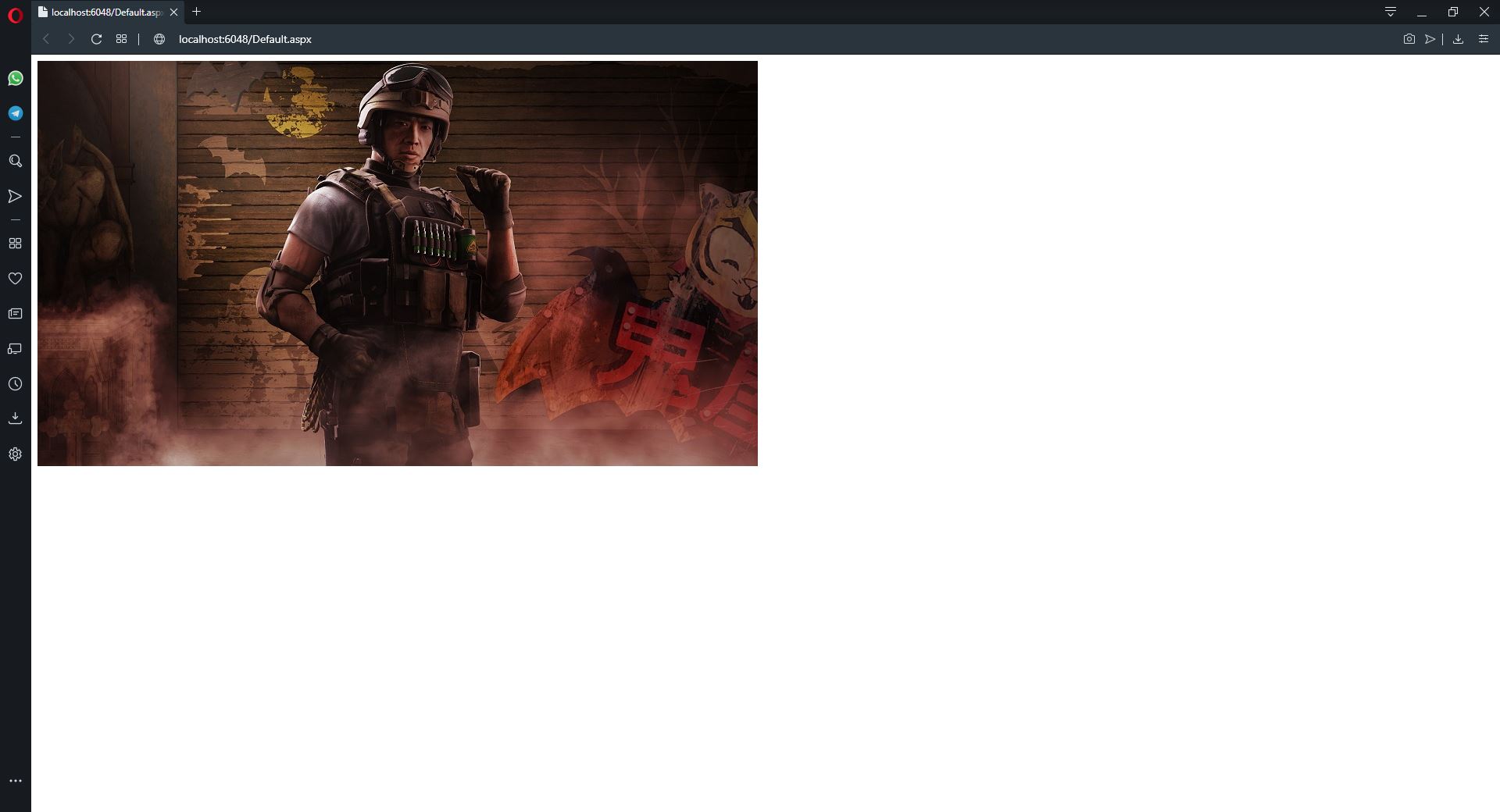
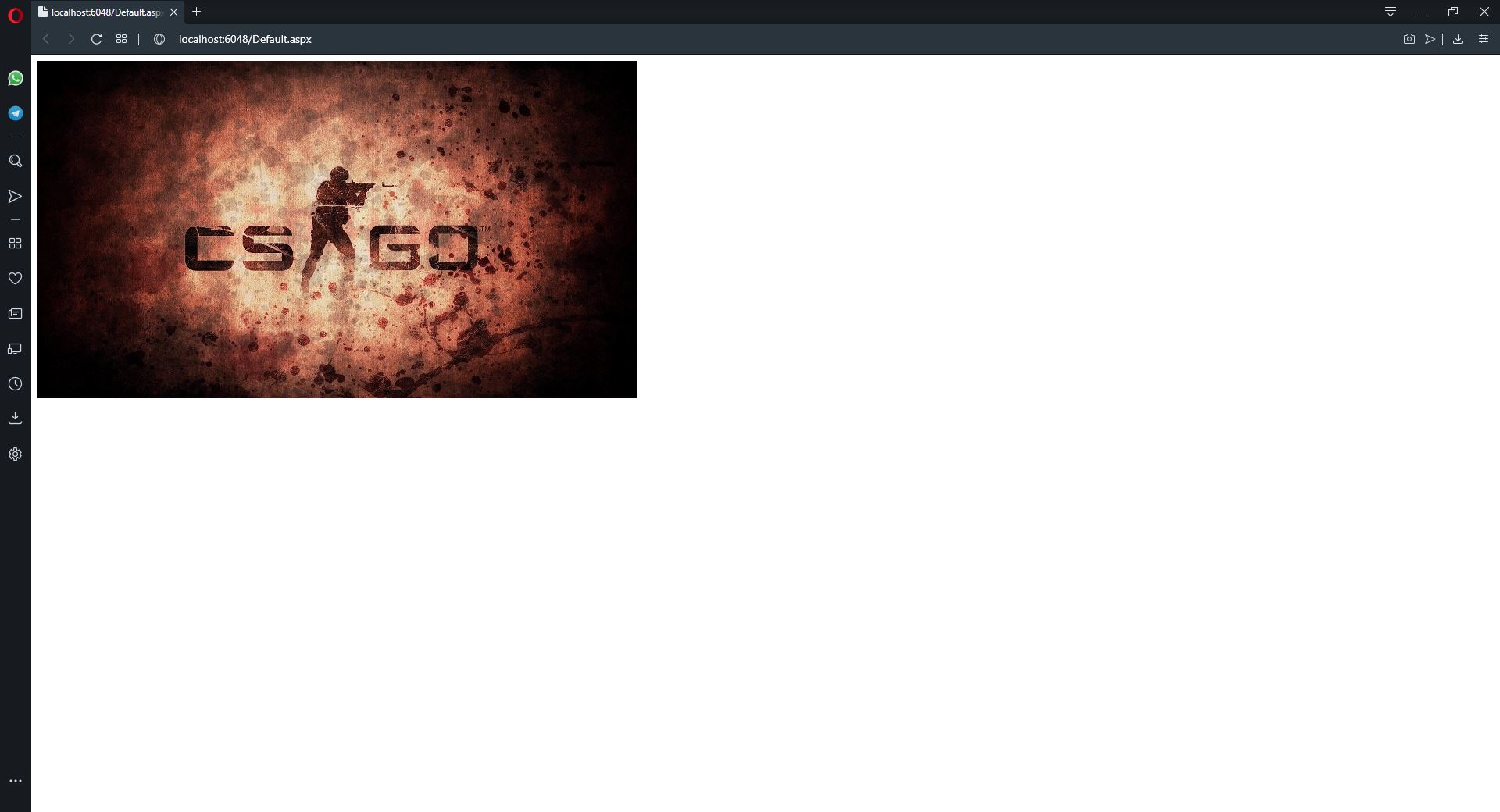
</asp:XmlDataSource>

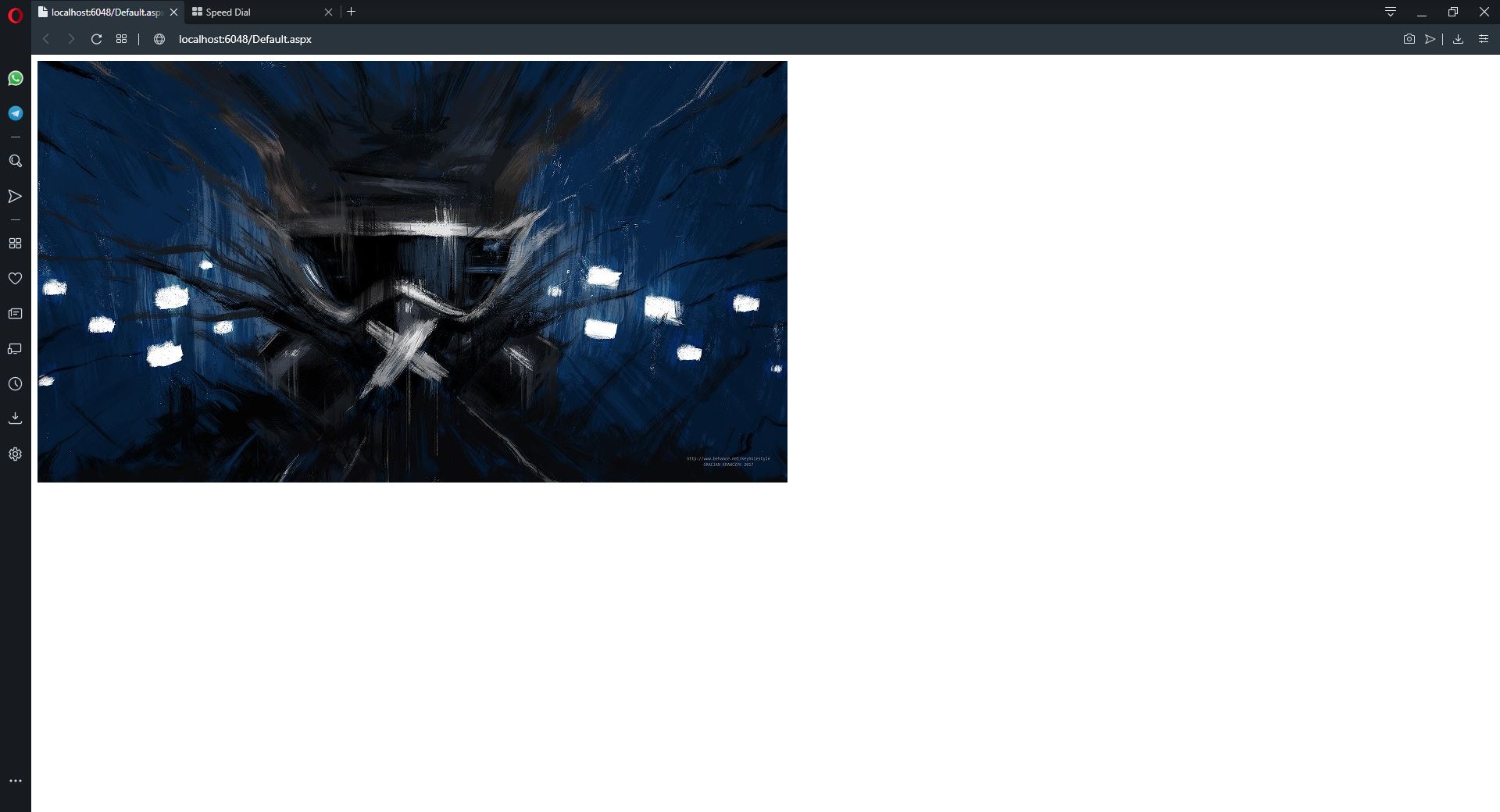
</form>

</body>

</html>

**Output :**





**Practical No.5**

***Aim :*** *Create Web Application to Demonstrate MultiView*

**Default.aspx**

<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Default.aspx.cs" Inherits="WebApplication7.\_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:MultiView ID="MultiView1" runat="server">

<asp:View ID="View1" runat="server">

<h4>Student Details</h4>

<p>Student FirstName</p>

<asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>

<p>Student LastName</p>

<asp:TextBox ID="TextBox2"

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

<asp:Button ID="Button2" runat="server" Text="Next" OnClick="Button2\_Click" />

</asp:View>

<asp:View ID="View2" runat="server">

<h4>Student Course Details</h4>

<p>Student Course</p> <asp:TextBox ID="TextBox3" runat="server"></asp:TextBox>

<p>Student Branch</p> <asp:TextBox ID="TextBox4" runat="server"></asp:TextBox>

<asp:Button ID="Button3" runat="server" Text="Previous" OnClick="Button3\_Click" />

<asp:Button ID="Button4" runat="server" Text="Next" OnClick="Button4\_Click" />

</asp:View>

<asp:View ID="View3" runat="server">

<h4>Student Personal Details</h4>

<p>Student Email</p>

<asp:TextBox ID="TextBox5" runat="server"></asp:TextBox>

<p>Student City</p>

<asp:TextBox ID="TextBox6" runat="server"></asp:TextBox>

<p>Student State</p>

<asp:TextBox ID="TextBox7" runat="server"></asp:TextBox>

<asp:Button ID="Button5" runat="server" Text="Previous" OnClick="Button5\_Click" />

<asp:Button ID="Button6" runat="server" Text="Next" OnClick="Button6\_Click" />

</asp:View>

<asp:View ID="View4" runat="server">

<h4>Student Details</h4>

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

<p>Student LastName : <asp:Label ID="Label2" runat="server" Text=""></asp:Label> </p>

<h4>Student Course Details</h4>

<p>Student Course : <asp:Label ID="Label3" runat="server" Text=""></asp:Label> </p>

<p>Student Branch : <asp:Label ID="Label4" runat="server" Text=""></asp:Label> </p>

<h4>Student Details</h4>

<p>Student EmailID :<asp:Label ID="Label5" runat="server" Text=""></asp:Label> </p>

<p>Student City : <asp:Label ID="Label6" runat="server" Text=""></asp:Label> </p>

<p>Student State : <asp:Label ID="Label7" runat="server" Text=""></asp:Label> </p>

</asp:View>

</asp:MultiView>

</div>

</form>

</body>

</html>

**Default.aspx.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 \_Default : System.Web.UI.Page

{

protected void Page\_Load(object sender, EventArgs e)

{

if (!IsPostBack)

{

MultiView1.ActiveViewIndex = 0;

}

}

protected void Button2\_Click(object sender, EventArgs e)

{

MultiView1.ActiveViewIndex = 1;

}

protected void Button4\_Click(object sender, EventArgs e)

{

MultiView1.ActiveViewIndex = 2;

}

protected void Button3\_Click(object sender, EventArgs e)

{

MultiView1.ActiveViewIndex = 1;

}

protected void Button5\_Click(object sender, EventArgs e)

{

MultiView1.ActiveViewIndex = 2;

}

protected void Button6\_Click(object sender, EventArgs e)

{

MultiView1.ActiveViewIndex = 3;

Label1.Text = TextBox1.Text;

Label2.Text = TextBox2.Text;

Label3.Text = TextBox3.Text;

Label4.Text = TextBox4.Text;

Label5.Text = TextBox5.Text;

Label6.Text = TextBox6.Text;

Label7.Text = TextBox7.Text;

}

}

}

**Practical No. 6**

*Overloading*

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace Method\_overloading

{

class Meh\_Overloading

{

public int volume(int x)

{

return (x \* x \* x);

}

public double volume(double r, int h)

{

return (3.14 \* r \* r \* h);

}

public int volume(int l, int b, int h)

{

return (l \* b \* h);

}

}

class Program

{

static void Main(string[] args)

{

int l, b, h, VolumeofCube, VolumeofBox;

double r, VolumeofCylinder;

Console.WriteLine("Enter length ,breadth,height and radius : ");

l = Convert.ToInt32(Console.ReadLine());

b = Convert.ToInt16(Console.ReadLine());

r = Convert.ToInt32(Console.ReadLine());

h = Convert.ToInt16(Console.ReadLine());

Meh\_Overloading Object1 = new Meh\_Overloading();

VolumeofCube = Object1.volume(1);

VolumeofCylinder = Object1.volume(r, h);

VolumeofBox = Object1.volume(l, b, h);

Console.WriteLine("Volume of Cube : " + VolumeofCube);

Console.WriteLine("Volume of Cylinder : " + VolumeofCylinder);

Console.WriteLine("Volume of Box : " + VolumeofBox);

Console.ReadKey();

}

}

}



**Practical no.7**

*Design a ASP.Net Pages for State management using Cookies*

**Default.aspx**

<%@ Page Title="Home Page" Language="C#" MasterPageFile="~/Site.master" AutoEventWireup="true"

CodeBehind="Default.aspx.cs" Inherits="WebApplication13.\_Default" %>

<asp:Content ID="HeaderContent" runat="server" ContentPlaceHolderID="HeadContent">

</asp:Content>

<asp:Content ID="BodyContent" runat="server" ContentPlaceHolderID="MainContent">

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

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

<asp:TextBox ID="TextBox1" runat="server"></asp:TextBox><br /><br />

<asp:Label ID="Label2" runat="server" Text="City "></asp:Label>

<asp:TextBox ID="TextBox2" runat="server"></asp:TextBox><br /><br />

<asp:Button ID="Button1" runat="server" Text="Next Page" onclick="Button1\_Click"/>

</form>

</asp:Content>

**Default.aspx.cs**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace WebApplication13

{

public partial class \_Default : System.Web.UI.Page

{

protected void Page\_Load(object sender, EventArgs e)

{

}

protected void Button1\_Click(object sender, EventArgs e)

{

HttpCookie cookie = new HttpCookie("UserInfo");

cookie["Name"] = TextBox1.Text;

cookie["City"] = TextBox2.Text;

cookie.Expires = DateTime.Now.AddDays(30);

Response.Cookies.Add(cookie);

Response.Redirect("About.aspx");

}

}

}

**About.aspx**

<%@ Page Title="About Us" Language="C#" MasterPageFile="~/Site.master" AutoEventWireup="true"

CodeBehind="About.aspx.cs" Inherits="WebApplication13.About" %>

<asp:Content ID="HeaderContent" runat="server" ContentPlaceHolderID="HeadContent">

</asp:Content>

<asp:Content ID="BodyContent" runat="server" ContentPlaceHolderID="MainContent">

<h2>

About

</h2>

<form>

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

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

</form>

</asp:Content>

**About.aspx.cs**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace WebApplication13

{

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

{

protected void Page\_Load(object sender, EventArgs e)

{

HttpCookie cookie = Request.Cookies["UserInfo"];

if (cookie != null)

{

lblName.Text = cookie["Name"];

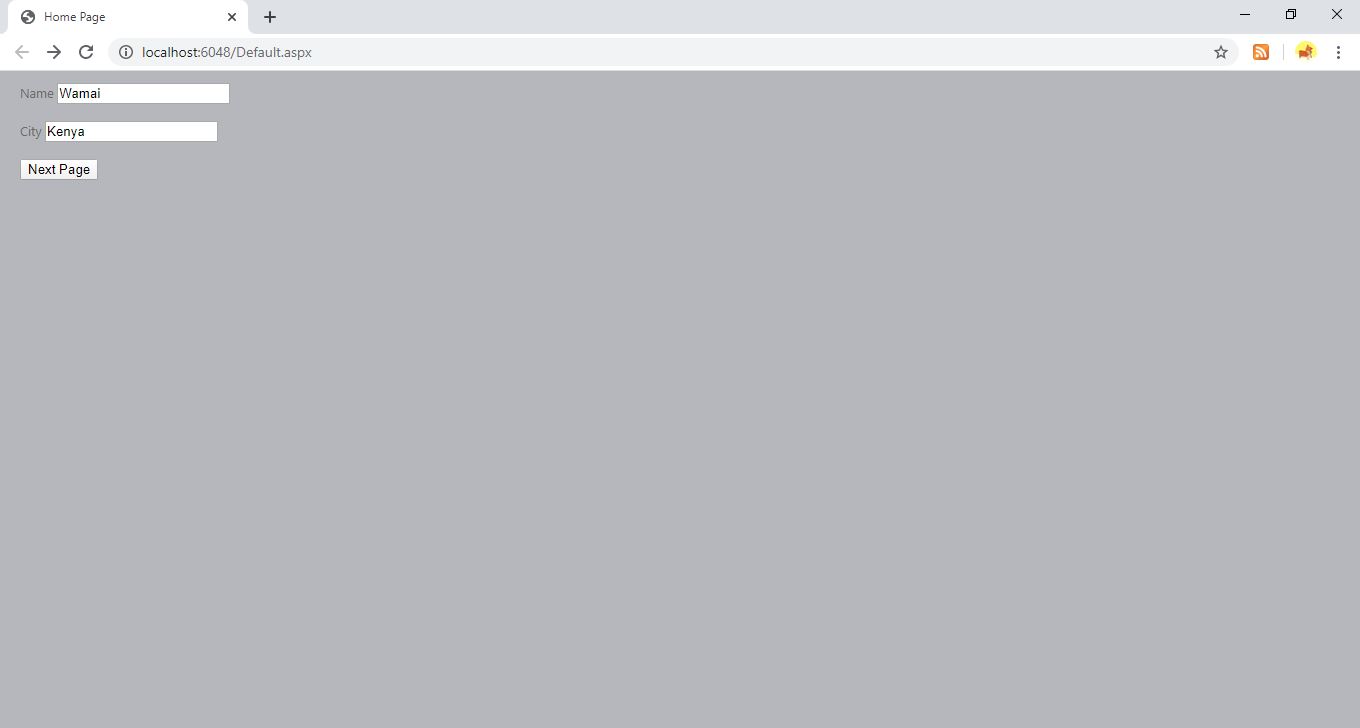
lblCity.Text = cookie["City"];

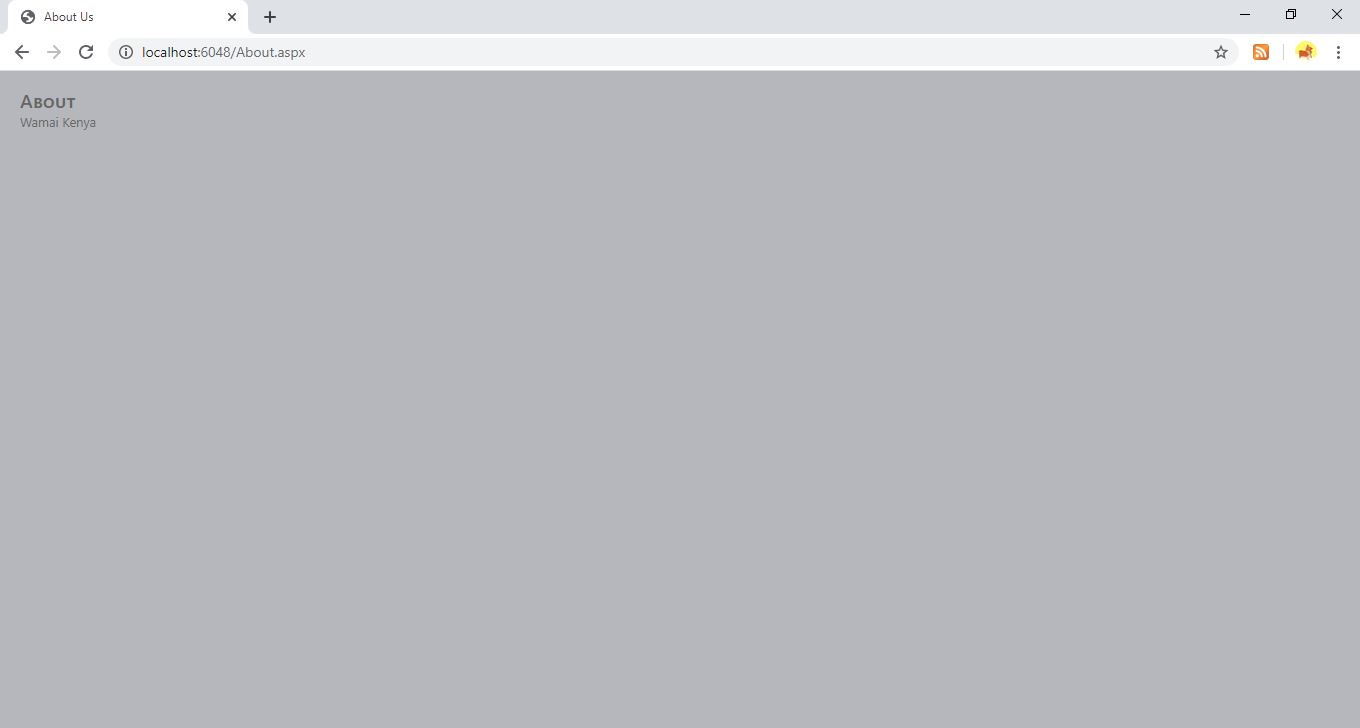
}

}

}

}





**Practical no.8**

*Design a ASP.Net page for performance improvement using cache*

**Default.aspx**

<%@ Page Title="Home Page" Language="C#" MasterPageFile="~/Site.master" AutoEventWireup="true"

CodeBehind="Default.aspx.cs" Inherits="WebApplication10.\_Default" %>

<%@ OutputCache Duration="20" Location="Server" VaryByParam = "None" %>

<asp:Content ID="HeaderContent" runat="server" ContentPlaceHolderID="HeadContent">

</asp:Content>

<asp:Content ID="BodyContent" runat="server" ContentPlaceHolderID="MainContent">

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

</asp:Content>

**Default.cs**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace WebApplication10

{

public partial class \_Default : System.Web.UI.Page

{

protected void Page\_Load(object sender, EventArgs e)

{

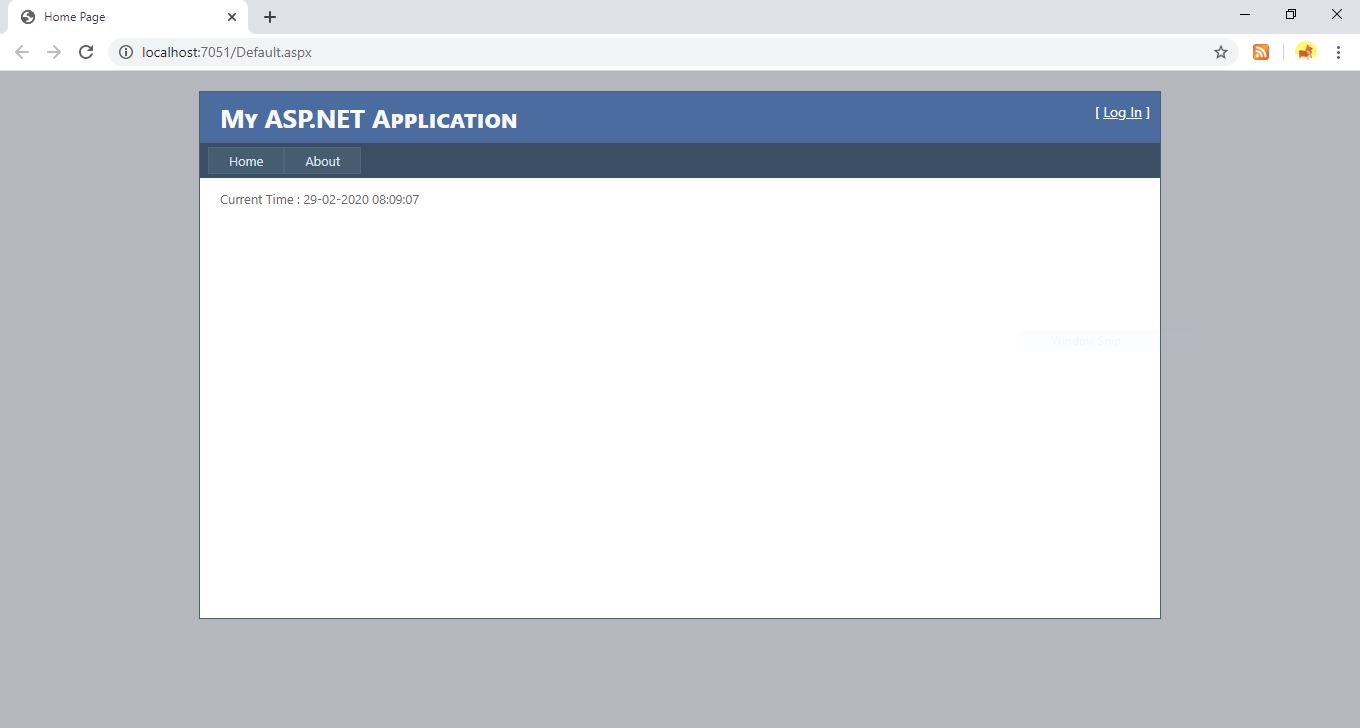
Label1.Text = "Current Time : ";

Label1.Text = Label1.Text + DateTime.Now.ToString();

}

}

}



**Practical no.9**

*Design ASP.NET Pages with Navigation control to demonstrate TreeView Control*

*Design ASP.NET Pages and use it other 4-5 content pages using TreeView Control*

**Site.Master**

<%@ Master Language="C#" AutoEventWireup="true" CodeBehind="Site.master.cs" Inherits="WebApplication16.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>

<link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css" />

<script type="text/javascript" src="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/js/bootstrap.min.js"></script>

</head>

<body>

<h1 align="center" style="background-color: #999999; font-family: 'Microsoft JhengHei UI'; font-size: x-large;">HEYATHI INDUSTRIES</h1>

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

<div class="container">

<div class="row">

<div class="col-3">

<asp:TreeView ID="TreeView1" runat="server">

<Nodes>

<asp:TreeNode Text="Home" NavigateUrl="Home.aspx"></asp:TreeNode>

<asp:TreeNode NavigateUrl="About.aspx" Text="About">

<asp:TreeNode NavigateUrl="History.aspx" Text="History"></asp:TreeNode></asp:TreeNode>

</Nodes>

</asp:TreeView>

</div>

<div class="col-9">

<asp:ContentPlaceHolder ID="ContentPlaceHolder1" runat="server">

<h2>TEXT TEXT TEXT</h2>

</asp:ContentPlaceHolder></div>

</div>

</div>

</form>

</body>

</html>

**Home.aspx**

<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Home.aspx.cs" Inherits="WebApplication16.WebForm1" MasterPageFile="Site.Master" %>

<asp:Content

ID="content1" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">

<h1 align="center" style="font-family: 'Bahnschrift SemiBold'; font-size: x-large; color: #FF0000;">The Mission of HEYATHI Industries is to Provide 24X7 Gaming Services</h1>

</asp:Content>

**About.aspx**

<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="About.aspx.cs" Inherits="WebApplication16.About" MasterPageFile="Site.Master" %>

<asp:Content

ID="content1" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">

<h1 align="center" style="font-family: 'Bahnschrift SemiBold'; font-size: x-large; color: #FF0000;">HEYATHI Industries annual turn over is approx US$1300 billion</h1>

</asp:Content>

**History.aspx**

<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="History.aspx.cs" Inherits="WebApplication16.WebForm2" MasterPageFile="Site.Master"%>

<asp:Content

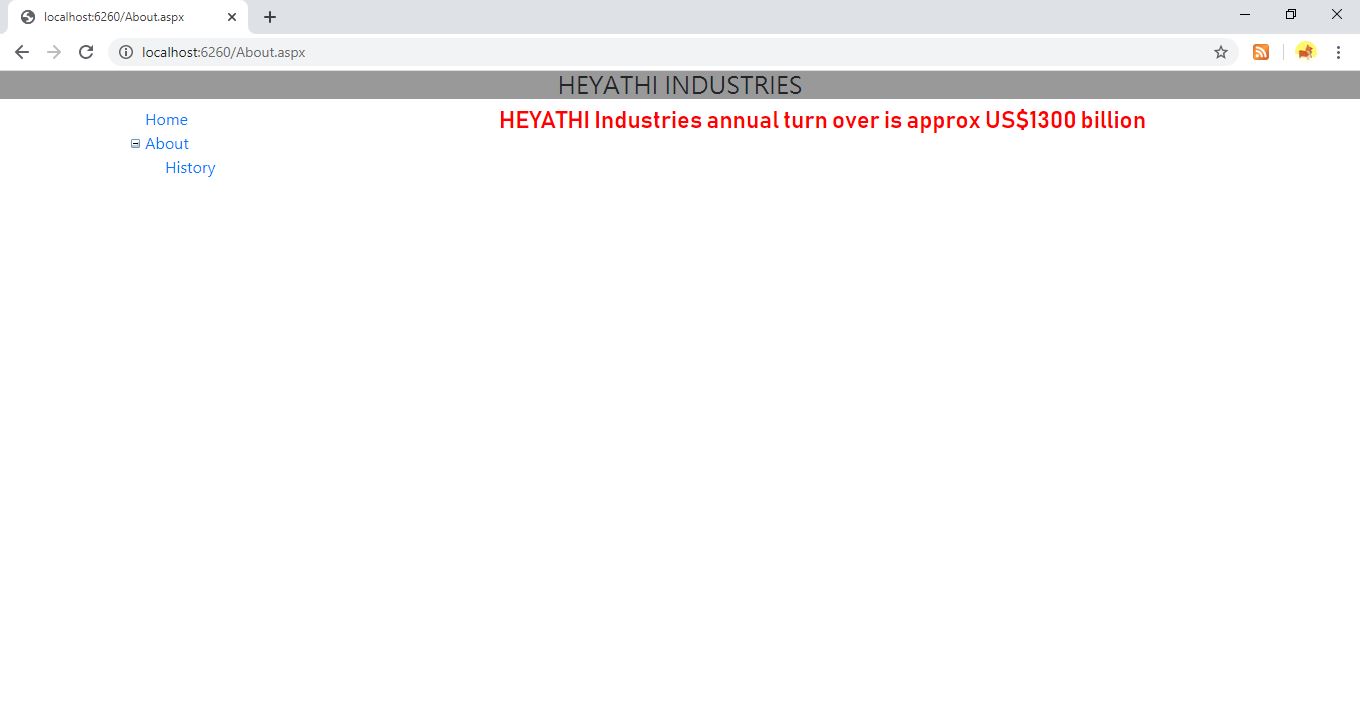
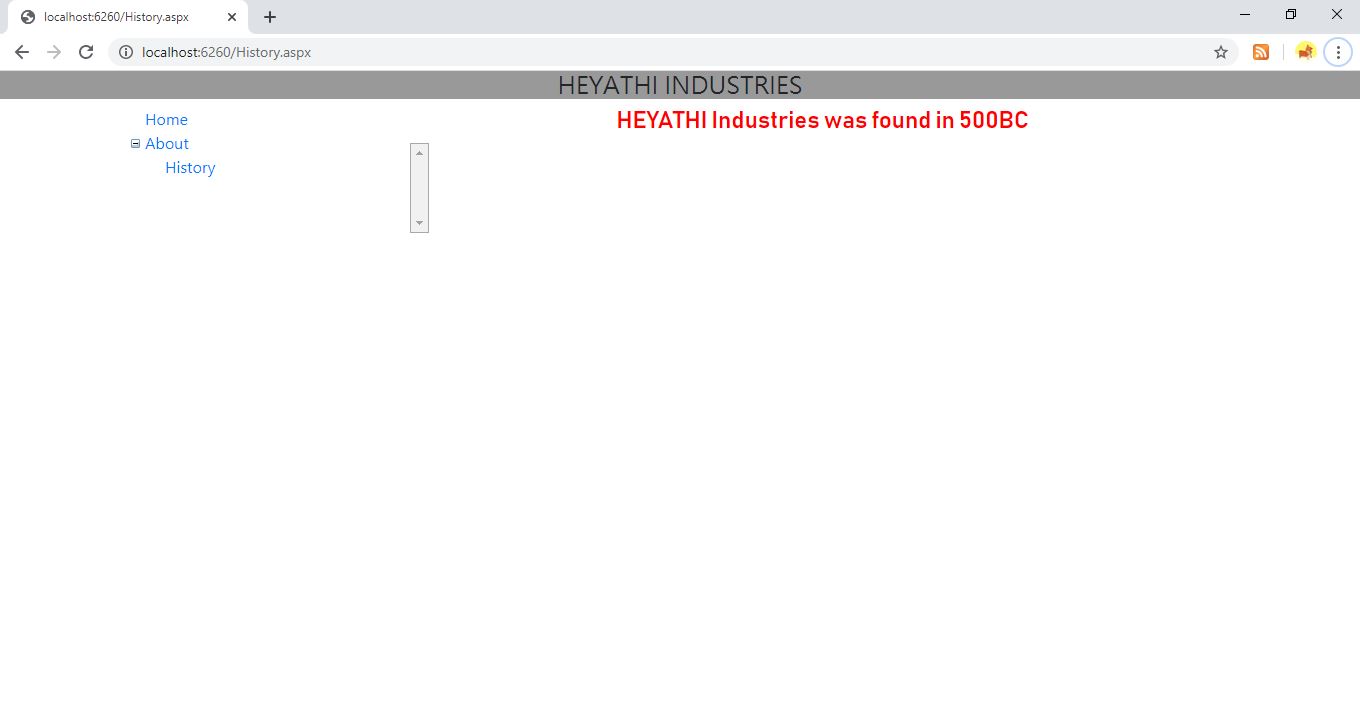
ID="content1" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">

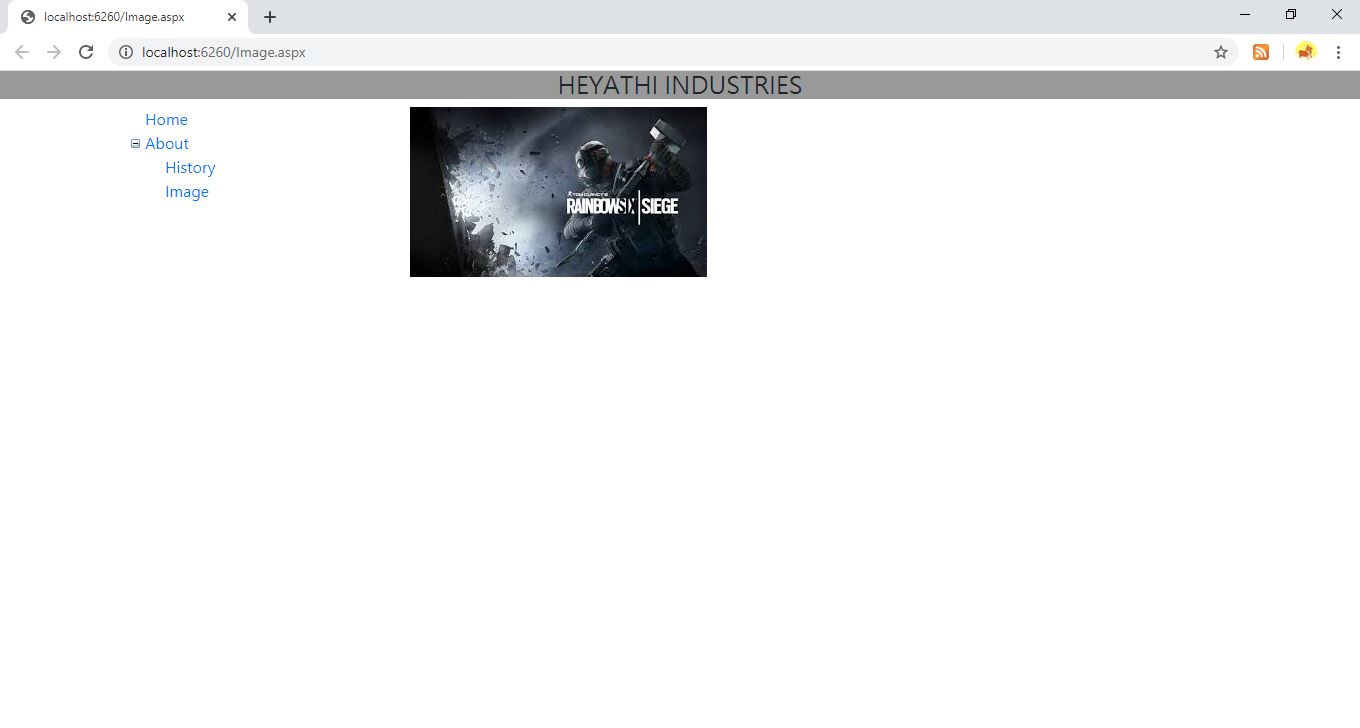
<h1 align="center" style="font-family: 'Bahnschrift SemiBold'; font-size: x-large; color: #FF0000;">HEYATHI Industries was found in 500BC</h1>

<asp:ListBox ID="ListBox1" runat="server">

</asp:ListBox>

</asp:Content>





**Practical no.10**

*Design ASP.NET Pages and perform Validation using Various validation control*

**Default.aspx**

<%@ Page Title="Home Page" Language="C#" MasterPageFile="~/Site.master" AutoEventWireup="true"

CodeBehind="Default.aspx.cs" Inherits="WebApplication12.\_Default" %>

<asp:Content ID="HeaderContent" runat="server" ContentPlaceHolderID="HeadContent">

</asp:Content>

<asp:Content ID="BodyContent" runat="server" ContentPlaceHolderID="MainContent">

<asp:Label ID="Label1" runat="server" Text="Roll No.: "></asp:Label>

<asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>

<asp:RequiredFieldValidator

ID="RequiredFieldValidator1" runat="server" ErrorMessage="Cannot leave Empty" ControlToValidate="TextBox1">

</asp:RequiredFieldValidator>

<asp:RegularExpressionValidator ID="RegularExpressionValidator1" runat="server" ErrorMessage="Wrong Format (Eg.S123)" ControlToValidate="TextBox1" ValidationExpression="^S[0-9]{3}">

</asp:RegularExpressionValidator>

<br/><br/>

<asp:Label ID="Label2" runat="server" Text="Student Name :"></asp:Label>

<asp:TextBox ID="TextBox2" runat="server"></asp:TextBox>

<asp:RegularExpressionValidator ID="RegularExpressionValidator2" runat="server" ErrorMessage="Name cannot Contain Digits,Special Characters !" ValidationExpression="^[A-Z]{3}" ControlToValidate="TextBox2">

</asp:RegularExpressionValidator>

<br/><br/>

<asp:Label ID="Label3" runat="server" Text="Age : "></asp:Label>

<asp:TextBox ID="TextBox3" runat="server"></asp:TextBox>

<asp:RangeValidator ID="RangeValidator1" runat="server" ErrorMessage="Age Group between 15-30 Only" ControlToValidate="TextBox3" MaximumValue="30" MinimumValue="15"></asp:RangeValidator>

<br/><br/>

<asp:Label ID="Label4" runat="server" Text="Password : "></asp:Label>

<asp:TextBox ID="TextBox4" runat="server"></asp:TextBox>

<br/><br/>

<asp:Label ID="Label5" runat="server" Text="Confirm Password : "></asp:Label>

<asp:TextBox ID="TextBox5" runat="server"></asp:TextBox>

<asp:CompareValidator ID="CompareValidator1" runat="server" ErrorMessage="Password did not Match" ControlToValidate="TextBox5" ControlToCompare="TextBox4"></asp:CompareValidator>

<br/><br/>

<asp:Label ID="Label6" runat="server" Text="Mobile Number : "></asp:Label>

<asp:TextBox ID="TextBox6" runat="server"></asp:TextBox>

<asp:RegularExpressionValidator ID="RegularExpressionValidator3" runat="server"

ErrorMessage="Number Number should be 10 digits" ControlToValidate="TextBox6"

ValidationExpression="^[0-9]{10}"></asp:RegularExpressionValidator>

<br/><br/>

<asp:Label ID="Label7" runat="server" Text="Email-ID : "></asp:Label>

<asp:TextBox ID="TextBox7" runat="server"></asp:TextBox>

<asp:RegularExpressionValidator ID="RegularExpressionValidator4" runat="server"

ErrorMessage="Invalid Email-ID" ControlToValidate="TextBox7"

ValidationExpression="\w+([-+.']\w+)\*@\w+([-.]\w+)\*\.\w+([-.]\w+)\*"></asp:RegularExpressionValidator>

<asp:Button ID="submitform" runat="server" Text="Submit" Height="37px" OnClick="Button1\_Click" />

</asp:Content>

**Default.cs**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace WebApplication12

{

public partial class \_Default : System.Web.UI.Page

{

protected void Page\_Load(object sender, EventArgs e)

{

}

protected void Button1\_Click(object sender, EventArgs e)

{

if (Page.IsValid)

{

submitform.Text = "VALID FORM !";

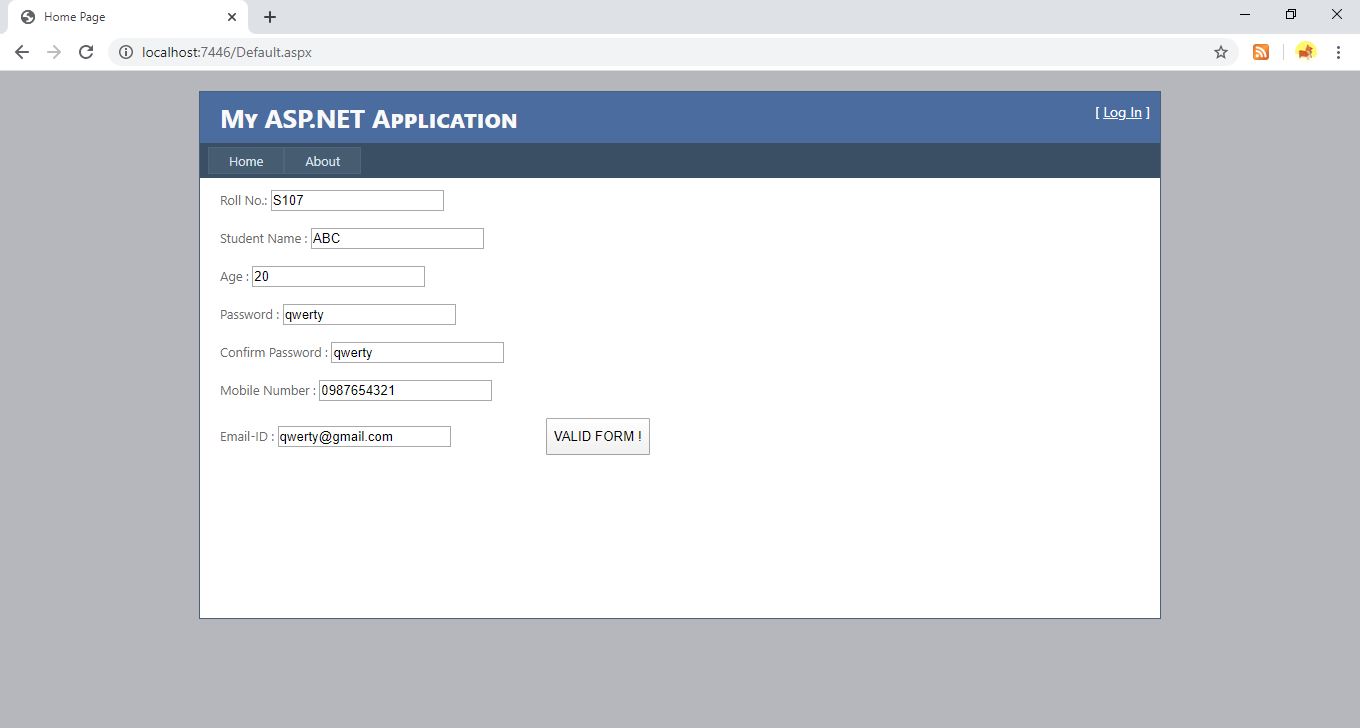
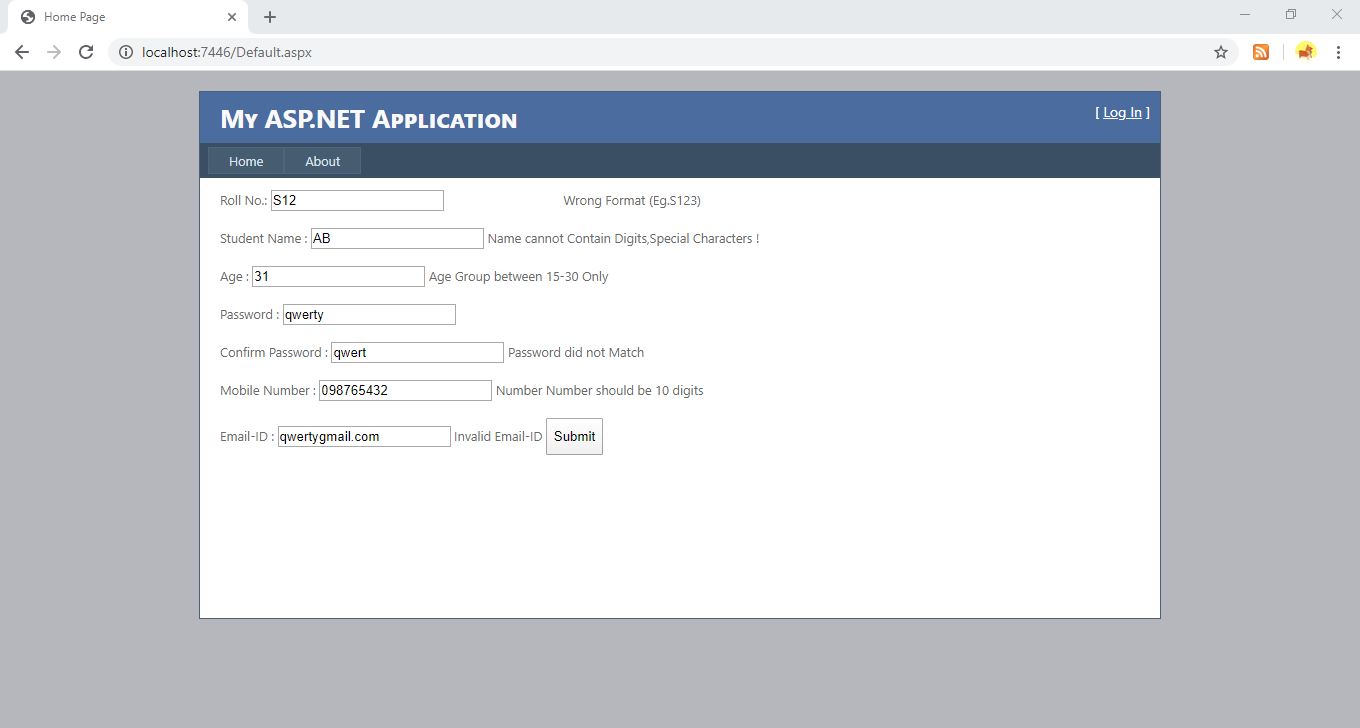
}

}

}

}

**Output:**



**Practical no.11**

*Create a ASP.Net Page to display Table using Grid View and perform CRUD Operations on Table*

**Webform.aspx**

<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="WebForm1.aspx.cs" Inherits="WebApplication20.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>

<h1>WEB FORM</h1>

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

<div>

<asp:GridView ID="GridView1" runat="server" BackColor="#DEBA84"

BorderColor="#DEBA84" BorderStyle="None" BorderWidth="1px" CellPadding="3"

CellSpacing="2" AutoGenerateColumns="False" DataKeyNames="id"

DataSourceID="SqlDataSource1" ShowFooter="True"

onselectedindexchanged="GridView1\_SelectedIndexChanged">

<FooterStyle BackColor="#F7DFB5" ForeColor="#8C4510" />

<HeaderStyle BackColor="#A55129" Font-Bold="True" ForeColor="White" />

<PagerStyle ForeColor="#8C4510" HorizontalAlign="Center" />

<RowStyle BackColor="#FFF7E7" ForeColor="#8C4510" />

<SelectedRowStyle BackColor="#738A9C" Font-Bold="True" ForeColor="White" />

<SortedAscendingCellStyle BackColor="#FFF1D4" />

<SortedAscendingHeaderStyle BackColor="#B95C30" />

<SortedDescendingCellStyle BackColor="#F1E5CE" />

<SortedDescendingHeaderStyle BackColor="#93451F" />

<Columns>

<asp:CommandField ShowDeleteButton="True" ShowEditButton="True" />

<asp:TemplateField HeaderText="id" InsertVisible="False" SortExpression="id">

<EditItemTemplate>

<asp:Label ID="Label1" runat="server" Text='<%# Eval("id") %>'></asp:Label>

</EditItemTemplate>

<ItemTemplate>

<asp:Label ID="Label1" runat="server" Text='<%# Bind("id") %>'></asp:Label>

</ItemTemplate>

<FooterTemplate>

<asp:LinkButton ID="lbInsert" Onclick="lbInsert\_Click" runat="server">Insert</asp:LinkButton>

</FooterTemplate>

</asp:TemplateField>

<asp:TemplateField HeaderText="name" SortExpression="name">

<EditItemTemplate>

<asp:TextBox ID="TextBox1" runat="server" Text='<%# Bind("name") %>'></asp:TextBox>

</EditItemTemplate>

<ItemTemplate>

<asp:Label ID="Label2" runat="server" Text='<%# Bind("name") %>'></asp:Label>

</ItemTemplate>

<FooterTemplate>

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

<asp:RequiredFieldValidator ID="RequiredFieldValidator1" runat="server" ErrorMessage="Cannot leave Empty" ControlToValidate="txtname"></asp:RequiredFieldValidator>

</FooterTemplate>

</asp:TemplateField>

<asp:TemplateField HeaderText="gender" SortExpression="gender">

<EditItemTemplate>

<asp:DropDownList ID="DropDownList1" runat="server" SelectedValue='<%# Bind("gender") %>'>

<asp:ListItem>Select Gender</asp:ListItem>

<asp:ListItem>Male</asp:ListItem>

<asp:ListItem>Female</asp:ListItem>

</asp:DropDownList>

</EditItemTemplate>

<ItemTemplate>

<asp:Label ID="Label3" runat="server" Text='<%# Bind("gender") %>'></asp:Label>

</ItemTemplate>

<FooterTemplate>

<asp:DropDownList ID="DropDownList2" runat="server">

<asp:ListItem Text="Select Gender"></asp:ListItem>

<asp:ListItem Text="Male"></asp:ListItem>

<asp:ListItem Text="Female"></asp:ListItem>

</asp:DropDownList>

</FooterTemplate>

</asp:TemplateField>

<asp:TemplateField HeaderText="city" SortExpression="city">

<EditItemTemplate>

<asp:TextBox ID="TextBox3" runat="server" Text='<%# Bind("city") %>'></asp:TextBox>

</EditItemTemplate>

<ItemTemplate>

<asp:Label ID="Label4" runat="server" Text='<%# Bind("city") %>'></asp:Label>

</ItemTemplate>

<FooterTemplate>

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

<asp:RequiredFieldValidator ID="RequiredFieldValidator3" runat="server" ErrorMessage="Cannot leave Empty" ControlToValidate="txtCity"></asp:RequiredFieldValidator>

</FooterTemplate>

</asp:TemplateField>

</Columns>

<EmptyDataTemplate>The grid is ready</EmptyDataTemplate>

</asp:GridView>

</div>

<asp:SqlDataSource ID="SqlDataSource1" runat="server"

ConnectionString="<%$ ConnectionStrings:ConnectionString %>"

DeleteCommand="DELETE FROM [student] WHERE [id] = @id"

InsertCommand="INSERT INTO student(name, gender, city) VALUES (@name, @gender, @city)"

SelectCommand="SELECT \* FROM [student]"

UpdateCommand="UPDATE [student] SET [name] = @name, [gender] = @gender, [city] = @city WHERE [id] = @id">

<DeleteParameters>

<asp:Parameter Name="id" Type="Int32" />

</DeleteParameters>

<InsertParameters>

<asp:Parameter Name="name" Type="String" />

<asp:Parameter Name="gender" Type="String" />

<asp:Parameter Name="city" Type="String" />

</InsertParameters>

<UpdateParameters>

<asp:Parameter Name="name" Type="String" />

<asp:Parameter Name="gender" Type="String" />

<asp:Parameter Name="city" Type="String" />

<asp:Parameter Name="id" Type="Int32" />

</UpdateParameters>

</asp:SqlDataSource>

</form>

</body>

</html>

**Webform.aspx.cs**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace WebApplication20

{

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

{

protected void Page\_Load(object sender, EventArgs e)

{

}

protected void lbInsert\_Click(object sender, EventArgs e)

{

SqlDataSource1.InsertParameters["name"].DefaultValue =

((TextBox)GridView1.FooterRow.FindControl("txtName")).Text;

SqlDataSource1.InsertParameters["gender"].DefaultValue =

((DropDownList)GridView1.FooterRow.FindControl("DropDownList2")).Text;

SqlDataSource1.InsertParameters["city"].DefaultValue =

((TextBox)GridView1.FooterRow.FindControl("txtCity")).Text;

SqlDataSource1.Insert();

}

protected void GridView1\_SelectedIndexChanged(object sender, EventArgs e)

{

}

}

}

**Output :**

