|  |
| --- |
| **TOLANI COLLEGE OF COMMERCE**  **(AUTONOMOUS)**  150-151, Sher-E-Punjab Society Guru Gobind Singh Road, Andheri East, Mumbai, Maharashtra 400 093    **Department of B.Sc. (Information Technology)**      **CERTIFICATE**  This is to certify that Mr. / Ms. Lavanya patil bearing Roll Nc 80 have completed the practical in the Course of Advance Web Programming in accordance with the syllabus of B.Sc. (Information Technology) Programme of Semester V as prescribed by the Tolani College of Commerce (Autonomous) in the academic year 2024-2025.      **Internal Examiner Programme Coordinator**            **External Examiner**            **Date:**  **College Seal**  Name : Lavanya patil Roll no : 80 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **T.Y.B.Sc.IT Sem 5 ADVANCE WEB PROGRAMMING PRACTICAL**      **INDEX**     |  |  |  |  | | --- | --- | --- | --- | | **Sr No.** | **Practical** | **Date** | **Sign** | | 1. | Working with Basic C# and ASP.NET |  |  | | 2. | Working with Object Oriented C# and ASP.NET |  |  | | 3. | Working with Web Forms and Controls |  |  | | 4. | Working with Form Controls |  |  | | 5. | Working with Navigations, Beautifications and Master  Page |  |  | | 6. | Working with Database |  |  | | 7. | Working with Database |  |  | | 8. | Working with data controls |  |  |                 Name : Lavanya patil Roll no : 80 |

|  |
| --- |
| **T.Y.B.Sc.IT Sem 5 ADVANCE WEB PROGRAMMING PRACTICAL**    **PRACTICAL 1**  **AIM** :- **Working with basic C# and ASP .NET**  **Q.1) A) Create an application that obtains four int values from the user and displays the product.**  **CODE:**  using System;  using System.Collections.Generic; using System.Linq; using System.Text;  namespace ConsoleApplication1  {  class Program  { static void Main(string[] args){ int num1,num2,num3,num4,prod; Console.Write("Enter number1:");  num1 = Int32.Parse(Console.ReadLine()); Console.Write("Enter number 2:");  num2= Convert.ToInt32(Console.ReadLine()); Console.Write("Enter number 3:");  num3= Convert.ToInt32(Console.ReadLine()); Console.Write("Enter number 4:");  num4 = Convert.ToInt32(Console.ReadLine());  prod = num1\*num2\*num3\*num4;  Console.WriteLine(num1+"\*"+num2+"\*" + num3 +"\*"+num4+"=" +prod); Console.ReadKey();  }  }  }    **OUTPUT:**      **B) Create an application to demonstrate string operations.**  **CODE:**  using System;  namespace cmdLineArgs  {  class Program  {  Name : Lavanya patil Roll no : 80 |

|  |  |
| --- | --- |
| **T.Y.B.Sc.IT Sem 5 ADVANCE WEB PROGRAMMING**    static void Main(string[] args)  {  string str = args[0]; int n = Convert.ToInt32(args[1]); Console.WriteLine("String:" + str);  Console.WriteLine("Number:" + n);  }  }  }    **OUTPUT:**                    Name : Lavanya patil Roll no : 80 | **PRACTICAL** |

|  |
| --- |
| **T.Y.B.Sc.IT Sem 5 ADVANCE WEB PROGRAMMING PRACTICAL**    **C) 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**  **Code:**  using System;  namespace PRAC  { class Program { struct Student  { public string studid, name, cname; public int day, month, year;} static void Main(string[] args) { Student[] s = new Student[5]; int i;  for (i = 0; i < 5; i++)  {  Console.Write("Enter Student Id:"); s[i].studid = Console.ReadLine();  Console.Write("Enter Student name : "); s[i].name = Console.ReadLine();  Console.Write("Enter Course name : "); s[i].cname = Console.ReadLine();  Console.Write("Enter date of birth\n Enter day(1-31):"); s[i].day = Convert.ToInt32(Console.ReadLine()); Console.Write("Enter month(1-12):");  s[i].month = Convert.ToInt32(Console.ReadLine()); Console.Write("Enter year:");  s[i].year = Convert.ToInt32(Console.ReadLine()); }  Console.WriteLine("\n\nStudent's List\n"); for (i = 0; i < 5; i++)  {  Console.WriteLine("\nStudent ID : " + s[i].studid);  Console.WriteLine("\nStudent name : " + s[i].name);  Console.WriteLine("\nCourse name : " + s[i].cname);  Console.WriteLine("\nDate of birth(dd-mm-yy) : " + s[i].day + "-" + s[i].month + "-" + s[i].year);  }}}}  **OUTPUT :-**      **D) 1) Create an application to demonstrate following operations**  **Code:** using System;  Name : Lavanya patil Roll no : 80 |

|  |  |
| --- | --- |
| **T.Y.B.Sc.IT Sem 5 ADVANCE WEB PROGRAMMING**    namespace ConsoleApplication3  {  class Program  {  static void Main(string[] args)  {  int num1=0,num2=1,num3,num4,num,counter;  Console.Write ("Upto how many number you want fibonacci series:"); num=int.Parse(Console.ReadLine()); counter=3;  Console.Write(num1+"\t"+num2); while(counter<=num)  {  num3 = num1 + num2; if (counter >= num)  break;  Console.Write("\t" + num3);  num1 = num2; num2 = num3; counter++;  }  }  }  }    **OUTPUT:-**    **2) Test for Prime Numbers**  **Code:**  using System; namespace testprime  {  class Program  {  static void Main(string[] args){ int num, counter;  Console.Write("Enter number:"); num = int.Parse(Console.ReadLine());  for (counter = 2; counter <= num / 2; counter++)  {  if ((num % counter) == 0) break;  Name : Lavanya patil Roll no : 80 | **PRACTICAL** |

|  |  |
| --- | --- |
| **T.Y.B.Sc.IT Sem 5 ADVANCE WEB PROGRAMMING**    }  if (num == 1)  Console.WriteLine(num + "is neither prime nor composite"); else if(counter<(num/2))  Console.WriteLine(num+"is not prime number"); else  Console.WriteLine(num+"is prime number"); }}}  **Output:**    **3] Test for vowels**  **Code:**  using System;  namespace vowels  {  class Program  {  static void Main(string[] args)  { char ch;  Console.Write("Enter a character : "); ch = (char)Console.Read(); switch (ch)  { case 'a': case 'A': case 'e': case 'E': case 'i': case 'I': case 'o': case 'O': case 'u': case 'U':  Console.WriteLine(ch + "is vowel"); break; default:  Console.Write(ch + "is not a vowel"); break;  }  Console.ReadKey();  }}}    **Output:**  Name : Lavanya patil Roll no : 80 | **PRACTICAL** |

|  |  |  |
| --- | --- | --- |
| **T.Y.B.Sc.IT Sem 5 ADVANCE WEB PROGRAMMING**        **4) Use of foreach loop with arrays.**  **Code:**  using System;  class ExampleForEach  {  public static void Main()  {  string[] str = { "Shield", "Evaluation", "DX" }; foreach (String s in str)  {  Console.WriteLine(s);  }  }    **Output:**    **5) Reverse a number and find sum of digits of a number.**  **Code:**  using System;  namespace reverseNumber  {  class Program  {  static void Main(string[] args)  {  int num,actualnumber,revnum=0,digit,sumDigits=0; Console.Write("Enter number:"); num = int.Parse(Console.ReadLine()); actualnumber = num; while (num > 0)  {  digit = num % 10; revnum = revnum \* 10 + digit; sumDigits=sumDigits+digit; num = num / 10; }  Console.WriteLine("Reverse of " + actualnumber + "=" + revnum); Console.WriteLine("Sum of its digits:" + sumDigits);}}}    Name : Lavanya patil Roll no : 80 |  | **PRACTICAL** |

|  |
| --- |
| **T.Y.B.Sc.IT Sem 5 ADVANCE WEB PROGRAMMING PRACTICAL**    **Output:**                **PRACTICAL 2**  **AIM: Working with Object Oriented C# and ASP .NET**  **A) Create simple application to perform following operations.**    **1) Finding Factorial Value**  **Code:**  using System.Collections.Generic; using System.Linq; using System.Text; namespace factorial  {  class Program  {  static void Main(string[] args)  {  int i, number, fact;  Console.WriteLine("Enter the Number"); number = int.Parse(Console.ReadLine()); fact = number;  for (i = number - 1; i >= 1; i--)  {  fact = fact \* i;  }  Console.WriteLine("\nFactorial of Given Number is: "+fact); Console.ReadLine(); }}}  **Output:**      Name : Lavanya patil Roll no : 80 |

|  |
| --- |
| **T.Y.B.Sc.IT Sem 5 ADVANCE WEB PROGRAMMING PRACTICAL**    **2) Money Conversion**  **Code:**  using System;  using System.Collections.Generic; using System.Linq; using System.Text;  namespace CurrencyConversion  {  class Program  {  static void Main(string[] args)  {  int choice;  Console.WriteLine(&quot;Enter your Choice :\n 1- Dollar to Rupee \n 2-  Euro to Rupee \n 3- Malaysian Ringgit to Rupee &quot;);  choice = int.Parse(Console.ReadLine()); switch (choice)  { case 1:  Double dollar, rupee, val;  Console.WriteLine(&quot;Enter the Dollar Amount :&quot;); dollar = Double.Parse(Console.ReadLine());  Console.WriteLine(&quot;Enter the Dollar Value :&quot;); val = double.Parse(Console.ReadLine());  rupee = dollar \* val;  Console.WriteLine(&quot;{0} Dollar Equals {1} Rupees&quot;, dollar, rupee); break; case 2:  Double Euro, rupe, valu;  Console.WriteLine(&quot;Enter the Euro Amount :&quot;);  Euro = Double.Parse(Console.ReadLine());  Console.WriteLine(&quot;Enter the Euro Value :&quot;); valu = double.Parse(Console.ReadLine());  rupe = Euro \* valu;  Console.WriteLine(&quot;{0} Euro Equals {1} Rupees&quot;, Euro, rupe); break; case 3:  Double ringit, rup, value;  Console.WriteLine(&quot;Enter the Ringgit Amount :&quot;); ringit = Double.Parse(Console.ReadLine());  Console.WriteLine(&quot;Enter the Ringgit Value :&quot;); value = double.Parse(Console.ReadLine()); rup = ringit \* value;  Console.WriteLine(&quot;{0} Malaysian Ringgit Equals {1} Rupees&quot;,  Name : Lavanya patil Roll no : 80 |

|  |  |  |
| --- | --- | --- |
| **T.Y.B.Sc.IT Sem 5** | **ADVANCE WEB PROGRAMMING** | **PRACTICAL** |
| ringit, rup); break;  }  Console.ReadLine();  }}} **Output:**                                        **3) Temperature Converter**    **Code:**  using System;  using System.Collections.Generic; using System.Linq; using System.Text;  namespace temperatureconversion  { class Program  Name : Lavanya patil | Roll no : 80 |  |

|  |
| --- |
| **T.Y.B.Sc.IT Sem 5 ADVANCE WEB PROGRAMMING PRACTICAL**    {  static void Main(string[] args)  {  int celsius, faren;  Console.WriteLine("Enter the Temperature in Celsius(°C) : "); celsius = int.Parse(Console.ReadLine());  faren = (celsius \* 9) / 5 + 32;  Console.WriteLine("0Temperature in Fahrenheit is(°F) : " + faren); Console.ReadLine();  } }  }  **Output:**                          **B)**  **Create simple application to demonstrate use of following concepts.**    **i)**  **Function Overloading**  **Code:**  using System;  namespace swap  {  class Overloading  {  public void swap(ref int n, ref int m)  {  int t; t = n;  Name : Lavanya patil Roll no : 80 |

|  |
| --- |
| **T.Y.B.Sc.IT Sem 5 ADVANCE WEB PROGRAMMING PRACTICAL**    n = m; m = t; }  public void swap(ref float f1, ref float f2)  { float f; f = f1; f1 = f2; f2 = f;  } }  class program  {  static void Main(string[] args)  {  Overloading objOverloading = new Overloading(); int n = 10, m = 20;  objOverloading.swap(ref n, ref m);  Console.WriteLine("N=" + n + "\tM=" + m);  float f1 = 10.5f, f2 = 20.6f;  objOverloading.swap(ref f1, ref f2);  Console.WriteLine("F1=" + f1 + "\tF2=" + f2);  }}}    **Output:**      **ii) Inheritance**  **a) Single Inheritance**  **Write a program to implement single inheritance from following figure. Accept and display data for one table.**  **Code:**  **Furniture.cs**    using System;  namespace SingleInheritance  {  class Furniture  { string material; float price;  public void getdata()  {  Console.Write("Enter material : ");  Name : Lavanya patil Roll no : 80 |

|  |  |
| --- | --- |
| **T.Y.B.Sc.IT Sem 5 ADVANCE WEB PROGRAMMING**    material = Console.ReadLine(); Console.Write("Enter price : ");  price = float.Parse(Console.ReadLine());  }  public void showdata()  {  Console.WriteLine("Material : " + material);  Console.WriteLine("Price : " + price);  } } }    **Table.cs**    using System;  namespace SingleInheritance  {  class Table:Furniture  { int height, surface\_area; public void getdata()  {  base.getdata();  Console.Write("Enter height: "); height = int.Parse(Console.ReadLine()); Console.Write("Enter surface area: ");  surface\_area = int.Parse(Console.ReadLine());  }  public void showdata()  {  base.showdata();  Console.WriteLine("Height : " + height);  Console.WriteLine("Surface Area : " + surface\_area);  } } }  **Program.cs**    using System;  namespace SingleInheritance  {  class Program  {  static void Main(string[] args)  {  Table t1 = new Table();  t1.getdata(); t1.showdata();  } } }    Name : Lavanya patil Roll no : 80 | **PRACTICAL** |

|  |  |
| --- | --- |
| **T.Y.B.Sc.IT Sem 5 ADVANCE WEB PROGRAMMING**    **Output:**        **b) Multiple Inheritance**  **Code:**  **Gross.cs**    using System;  namespace MultipleInheritance  {  interface Gross  { int ta { get; set;} int da { get; set; }  int GrossSal();  } }    **Employee.cs**    using System;  namespace MultipleInheritance  {  class Employee  {  string name;  public Employee(string name)  { this.name = name; }  public int BasicSal(int basicSal)  { return basicSal; } public void ShowData()  {  Console.WriteLine("Name : " + name);  } } }    **Salary.cs**  Name : Lavanya patil Roll no : 80 | **PRACTICAL** |

|  |  |
| --- | --- |
| **T.Y.B.Sc.IT Sem 5 ADVANCE WEB PROGRAMMING**      using System;  namespace MultipleInheritance  {  class Salary:employee,Gross  { int hra;  public Salary(string name, int hra):base(name)  { this.hra = hra; } public int ta  { get {return S\_ta; } set { S\_ta = value; }  } private int S\_ta; public int da  { get { return S\_da; } set { S\_da = value; }  } private int S\_da; public int GrossSal()  { int gSal;  gSal = hra + ta + da + BasicSal(15000); return gSal;  } public void dispSal()  { base.ShowData();  Console.WriteLine("Gross Sal : " + GrossSal()); } } }    **Program.cs**    using System;  namespace MultipleInheritance  {  class Program  {  static void Main(string[] args)  {  Salary s = new Salary("Prachit", 35000); s.da = 20000;  s.ta = 30000;  s.dispSal();  } } }    Name : Lavanya patil Roll no : 80 | **PRACTICAL** |

|  |  |
| --- | --- |
| **T.Y.B.Sc.IT Sem 5 ADVANCE WEB PROGRAMMING**    **Output:**    **c) Hierarchical Inheritance**  **Code:**  **Employee.cs**    using System;  namespace HeirarchicalInheritance  {  class Employee  {  public virtual void display()  {  Console.WriteLine("Display of Employee class called");  } }}    **Programmer.cs**    using System;  namespace HeirarchicalInheritance  {  class Programmer : Employee  {  public override void display()  {  Console.WriteLine("Display of Programmer class called");  } }}    **Manager.cs**    using System;  namespace HeirarchicalInheritance  {  class Manager : Employee  {  public override void display()  {  Console.WriteLine("Display of Manager class called");  }  }  }    **Program.cs**    using System;  Name : Lavanya patil Roll no : 80 | **PRACTICAL** |

|  |
| --- |
| **T.Y.B.Sc.IT Sem 5 ADVANCE WEB PROGRAMMING PRACTICAL**    namespace HeirarchicalInheritance  {  class Program  {  static void Main(string[] args)  {  Employee objEmployee;  Console.Write("Whose details you want to use to see \n 1.Programmer \n 2.Manager: "); int choice = int.Parse(Console.ReadLine()); if (choice == 1)  {  objEmployee = new Programmer();  objEmployee.display();  }  else if (choice == 2)  {  objEmployee = new Manager();  objEmployee.display();  } else {  Console.WriteLine("Wrong choice entered");  }  }  }  }    **Output:**        **d) Multilevel Inheritance**  **Code: Result.cs** using System;  namespace multilevelinheritance  {  class Result:Test  { int total;  public Result(int roll\_no, string name, int marks1, int marks2) : base(roll\_no, name, marks1, marks2)  { total = getMarks1() + getMarks2();  Name : Lavanya patil Roll no : 80 |

|  |  |
| --- | --- |
| **T.Y.B.Sc.IT Sem 5 ADVANCE WEB PROGRAMMING**    }  public void display()  {  base.display();  Console.WriteLine("Total: " + total);  } }}    Test.cs using System;  namespace multilevelinheritance  {  class Test:student  {  int marks1, marks2;  public Test(int roll\_no, string name, int marks1, int marks2) : base(roll\_no, name)  {  this.marks1 = marks1; this.marks2 = marks2;  }  public int getMarks1()  {  return marks1;  }  public int getMarks2()  {  return marks2;  }  public void dispaly()  {  base.display();  Console.WriteLine("Marks1: " + marks1);  Console.WriteLine("Marks2: " + marks2);  } } } Student.cs using System;  namespace multilevelinheritance  {  class student  {  int roll\_no; string name;  public student(int roll\_no, string name)  { this.roll\_no = roll\_no; this.name = name; }  Name : Lavanya patil Roll no : 80 | **PRACTICAL** |

|  |  |
| --- | --- |
| **T.Y.B.Sc.IT Sem 5 ADVANCE WEB PROGRAMMING**    public student() { } public void display()  {  Console.WriteLine("Roll no: " + roll\_no);  Console.WriteLine("Name: " + name);  } } }    Program.cs  using System; namespace multilevelinheritance  {  class Program  {  static void Main(string[] args)  {  Result r1 = new Result(65, "Raina", 90, 90); r1.display(using System;  }  } }  **Output:**                            **iii** ) **Constructor Overloading**  **Code:**  Salary.cs using System; namespace SalaryConstructure  Name : Lavanya patil Roll no : 80 | **PRACTICAL** |

|  |  |
| --- | --- |
| **T.Y.B.Sc.IT Sem 5 ADVANCE WEB PROGRAMMING**    {  class Salary  { int basic, ta, da, hra; public Salary() {da = 9000; hra = 6000;  }  public void getdata()  {Console.Write("Enter basic salary : "); basic = int.Parse(Console.ReadLine());  Console.Write("Enter travelling allowance : "); ta = int.Parse(Console.ReadLine());  }public void showdata()  {Console.WriteLine("Basic salary : " + basic);  Console.WriteLine("Dearness allowence : " + da);  Console.WriteLine("Housing rent allowence : " + hra);  Console.WriteLine("Travelling allowence : " + ta);  Console.WriteLine("Gross Salary : " + (basic + da + hra + ta)); } } }  Program.cs using System;  namespace SalaryConstructure  {  class Program  {  static void Main(string[] args){ Salary s = new Salary(); s.getdata();  s.showdata();  } } }    **Output:**    **iv) Interfaces Code:**  ODDEVEN.cs    using System;  using System.Collections.Generic; using System.Linq; using System.Text; namespace InterFaceDemo {  interface IOne { void ONE();  Name : Lavanya patil Roll no : 80 | **PRACTICAL** |

|  |  |
| --- | --- |
| **T.Y.B.Sc.IT Sem 5 ADVANCE WEB PROGRAMMING**    }  interface ITwo { void TWO();  }  interface IThree: IOne {  void THREE();  }  interface IFour { void FOUR();  }  interface IFive: IThree { void FIVE();  }  interface IEVEN: ITwo, IFour {} class ODDEVEN: IEVEN, IFive  {  public void ONE()  {  Console.WriteLine("This is ONE");  }  public void TWO() {  Console.WriteLine("This is TWO");  }  public void THREE() {  Console.WriteLine("This is THERE");  }  public void FOUR() {  Console.WriteLine("This is FOUR");  }  public void FIVE() {  Console.WriteLine("This is FIVE");  }  }  }  Program.cs using System;  using System.Collections.Generic; using System.Linq; using System.Text; namespace InterFaceDemo { class Program { static void Main(string[] args) { Console.WriteLine("This is ODD"); IFive obj1 = new ODDEVEN();  obj1.ONE(); obj1.THREE(); obj1.FIVE();  Name : Lavanya patil Roll no : 80 | **PRACTICAL** |

Console.WriteLine("\n\nThis is EVEN"); IEVEN obj2 = new ODDEVEN();

obj2.TWO(); obj2.FOUR(); Console.ReadLine();

}

}

}

**Output:**

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

**i) Using Delegates and events**

**Code:**

TrafficSignal.cs using System;

namespace TrafficDelegateExample

{

public delegate void TrafficDel(); class TrafficSignal

{

public static void Yellow()

{

Console.WriteLine("Yellow light signals to get ready");

}

public static void Green()

{

Console.WriteLine("Green light signals to go"); }

|  |  |
| --- | --- |
| **T.Y.B.Sc.IT Sem 5 ADVANCE WEB PROGRAMMING**    public static void Red()  {  Console.WriteLine("Red light signals to stop"); }  TrafficDel[] td = new TrafficDel[3]; public void IdentifySignal()  {  td[0] = new TrafficDel(Yellow); td[1] = new TrafficDel(Green); td[2] = new TrafficDel(Red);  }  public void display()  { td[0](); td[1](); td[2]();  }  } }    Program.cs using System;  namespace TrafficDelegateExample  {  class Program  {  static void Main(string[] args)  {  TrafficSignal ts = new TrafficSignal(); ts.IdentifySignal(); ts.display();  } } }    **Output:**                  Name : Lavanya patil Roll no : 80 | **PRACTICAL** |

# PRACTICAL 3

**AIM: - Working with Web Forms and Controls.**

**A) Demonstrate the use of Calendar control to perform following operations.**

1. **Display messages in a calendar control**
2. **Display vacation in a calendar control.**

## c) Selected day in a calendar control using style

**d) Difference between two calendar dates.**

**Code:**

## Webform1.aspx

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

Inherits="Practical\_3a.WebForm1" %>

<!DOCTYPE html>

<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" BackColor="#FFFFCC"

BorderColor="#FFCC66" BorderWidth="1px" DayNameFormat="Shortest"

Font-Names="Verdana" Font-Size="8pt" ForeColor="#663399"

Height="200px"

NextPrevFormat="ShortMonth" OnDayRender="Calendar1\_DayRender"

ShowGridLines="True" Width="300px"

OnSelectionChanged="Calendar1\_SelectionChanged" >

<DayHeaderStyle BackColor="#FFCC66" Font-Bold="True" Height="1px" />

<NextPrevStyle BorderStyle="Solid" BorderWidth="2px" Font-Size="9pt"

ForeColor="#FFFFCC" />

<OtherMonthDayStyle BackColor="#FFCC99" BorderStyle="Solid"

ForeColor="#CC9966" />

<SelectedDayStyle BackColor="Red" Font-Bold="True" />

<SelectorStyle BackColor="#FFCC66" />

<TitleStyle BackColor="#990000" Font-Bold="True" Font-Size="9pt"

ForeColor="#FFFFCC" />

<TodayDayStyle BackColor="#FFCC66" ForeColor="White" />

<WeekendDayStyle Height="50px" />

</asp:Calendar>

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

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

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

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

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

<asp:Button ID="btnResult" runat="server" Text="Show Result"

OnClick="btnResult\_Click" />

<asp:Button ID="btnReset" runat="server" Text="Reset" OnClick="btnReset\_Click" />

</div>

</form>

</body>

</html>

## Webform1.apsx.cs

using System;

using System.Web.UI.WebControls; namespace Practical\_3a

{

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

{

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

{

Calendar1.Caption = "SAMBARE";

Calendar1.FirstDayOfWeek = FirstDayOfWeek.Sunday;

Calendar1.NextPrevFormat = NextPrevFormat.ShortMonth;

Calendar1.TitleFormat = TitleFormat.Month;

Label2.Text = "Today's Date: " + Calendar1.TodaysDate.ToShortDateString();

Label3.Text = "Ganpati Vacation Start: 9-13-2018";

TimeSpan d = new DateTime(2018, 9, 13) - DateTime.Now;

Label4.Text = "Days Remaining For Ganpati Vacation: " + d.Days.ToString();

TimeSpan d1 = new DateTime(2018, 12, 31) - DateTime.Now;

Label5.Text = "Days Remaining for New Year: " + d1.Days.ToString();

if (Calendar1.SelectedDate.ToShortDateString() == "9-13-2018") Label3.Text = "<b>Ganpati Festival Start</b>";

if (Calendar1.SelectedDate.ToShortDateString() == "9-23-2018")

Label3.Text = "<b>Ganpati Festival End</b>";

}

protected void Calendar1\_DayRender(object sender, DayRenderEventArgs e)

{

if (e.Day.Date.Day == 5 && e.Day.Date.Month == 9)

{

e.Cell.BackColor = System.Drawing.Color.Yellow; Label lbl = new Label(); lbl.Text = "<br>Teachers Day!"; e.Cell.Controls.Add(lbl); Image g1 = new Image(); g1.ImageUrl = "td.jpg"; g1.Height = 20; g1.Width = 20;

e.Cell.Controls.Add(g1);

}

if (e.Day.Date.Day == 13 && e.Day.Date.Month == 9)

{

Calendar1.SelectedDate = new DateTime(2018, 9, 12);

Calendar1.SelectedDates.SelectRange(Calendar1.SelectedDate,

Calendar1.SelectedDate.AddDays(10));

Label lbl1 = new Label(); lbl1.Text = "<br>Ganpati!"; e.Cell.Controls.Add(lbl1);

}

}

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

{

Label1.Text = "";

Label2.Text = "";

Label3.Text = "";

Label4.Text = "";

Label5.Text = "";

Calendar1.SelectedDates.Clear();

} protected void Calendar1\_SelectionChanged(object sender, EventArgs e)

{

Label1.Text = "Your Selected Date: " + Calendar1.SelectedDate.Date.ToString();

}

}

}

**Output:**

**B) Demonstrate the use of Treeview control perform following operatio**

|  |  |
| --- | --- |
| **T.Y.B.Sc.IT Sem 5 ADVANCE WEB PROGRAMMING**     1. **Treeview control and data list.** 2. **Treeview operations.**   **Add XML File Website -> Add -> XML File and Name it ‘stdetail’.**    **Code:**  stdetail.xml  <?xml version="1.0" encoding="utf-8" ?>  <studentdetail>  <student>  <sid>1</sid>  <sname>Tushar</sname>  <sclass>TYIT</sclass>  </student>  <student>  <sid>2</sid>  <sname>Sonali</sname>  <sclass>TYCS</sclass>  </student>  <student>  <sid>3</sid>  <sname>Yashashree</sname>  <sclass>TYIT</sclass>  </student>  <student>  <sid>4</sid>  <sname>Vedshree</sname>  <sclass>TYCS</sclass>  </student>  </studentdetail>  Default2.aspx  <form id="form1" runat="server">  <div>  Treeview control navigation:<asp:TreeView ID = "TreeView1" runat = "server" Width =  "150px" ImageSet="Arrows">  <HoverNodeStyle Font-Underline="True" ForeColor="#5555DD" /> <Nodes>  <asp:TreeNode Text = "ASP.NET Practs" Value = "New Node">  <asp:TreeNode Text = "Calendar Control" Value = "RED"  NavigateUrl="~/calndrCtrl.aspx">  </asp:TreeNode>  <asp:TreeNode Text = "Constructor Overloading" Value = "GREEN"  NavigateUrl="~/clsconstrc.aspx"></asp:TreeNode>  <asp:TreeNode NavigateUrl="~/singleInh.aspx" Text="Inheritance"  Name : Lavanya patil Roll no : 80 | **PRACTICAL** |

|  |  |
| --- | --- |
| **T.Y.B.Sc.IT Sem 5 ADVANCE WEB PROGRAMMING**    Value="BLUE"></asp:TreeNode>  <asp:TreeNode NavigateUrl="~/clsProp.aspx" Text="Class Properties" Value="Class  Properties"></asp:TreeNode>  </asp:TreeNode>  </Nodes>  <NodeStyle Font-Names="Tahoma" Font-Size="10pt" ForeColor="Black"  HorizontalPadding="5px" NodeSpacing="0px" VerticalPadding="0px" />  <ParentNodeStyle Font-Bold="False" />  <SelectedNodeStyle Font-Underline="True" ForeColor="#5555DD"  HorizontalPadding="0px" VerticalPadding="0px" />  </asp:TreeView>  <br />  Fetch Datalist Using XML data : </div>  <asp:DataList ID="DataList1" runat="server">  <ItemTemplate>  <table class = "table" border="1">  <tr>  <td>Roll Num : <%# Eval("sid") %><br />  Name : <%# Eval("sname") %><br />  Class : <%# Eval("sclass")%>  </td>  </tr>  </table>  </ItemTemplate>  </asp:DataList>    **Default1.aspx.cs** using System.Data;  public partial class \_Default : System.Web.UI.Page  {  protected void Page\_Load(object sender, EventArgs e)  {  if (!IsPostBack)  {  BindData();  } }  protected void BindData()  {  DataSet ds = new DataSet();  ds.ReadXml(Server.MapPath("stdetail.xml")); if (ds != null && ds.HasChanges())  {  DataList1.DataSource = ds;  DataList1.DataBind(); }  Name : Lavanya patil Roll no : 80 | **PRACTICAL** |

else

{

DataList1.DataBind();

} }

}

**Output:**

# PRACTICAL 4

## AIM: Working with form controls

**A) Create an example of a registration form that demonstrates the use of various validation controls in ASP.NET Code:**

### Registration.aspx

<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Registration.aspx.cs"

Inherits="ValidationDemo.Registration" %>

<!DOCTYPE html>

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

<head runat="server">

<title>Registration Form</title>

<style type="text/css">

.form-group { margin-bottom: 15px; }

.error { color: red; }

</style>

</head>

<body>

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

<h2>Registration Form</h2>

<div class="form-group">

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

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

<asp:RequiredFieldValidator ID="rfvUsername" runat="server"

ControlToValidate="txtUsername" ErrorMessage="Username is required" CssClass="error" Display="Dynamic"></asp:RequiredFieldValidator> </div>

<div class="form-group">

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

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

<asp:RequiredFieldValidator ID="rfvEmail" runat="server"

ControlToValidate="txtEmail" ErrorMessage="Email is required"

CssClass="error" Display="Dynamic"></asp:RequiredFieldValidator>

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

ControlToValidate="txtEmail" ErrorMessage="Invalid email format"

ValidationExpression="\w+([-+.']\w+)\*@\w+([-.]\w+)\*\.\w+([-.]\w+)\*"

CssClass="error" Display="Dynamic"></asp:RegularExpressionValidator> </div>

<div class="form-group">

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

<asp:TextBox ID="txtPassword" runat="server"

TextMode="Password"></asp:TextBox>

<asp:RequiredFieldValidator ID="rfvPassword" runat="server"

ControlToValidate="txtPassword" ErrorMessage="Password is required" CssClass="error" Display="Dynamic"></asp:RequiredFieldValidator> </div>

<div class="form-group">

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

<asp:TextBox ID="txtConfirmPassword" runat="server"

TextMode="Password"></asp:TextBox>

<asp:CompareValidator ID="cvPassword" runat="server"

ControlToCompare="txtPassword" ControlToValidate="txtConfirmPassword"

ErrorMessage="Passwords do not match" CssClass="error"

Display="Dynamic"></asp:CompareValidator>

</div>

<div class="form-group">

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

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

<asp:RangeValidator ID="rvAge" runat="server"

ControlToValidate="txtAge" ErrorMessage="Age must be between 18 and 100"

MinimumValue="18" MaximumValue="100" Type="Integer"

CssClass="error" Display="Dynamic"></asp:RangeValidator> </div>

<div class="form-group">

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

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

<asp:CustomValidator ID="cvWebsite" runat="server"

ControlToValidate="txtWebsite" ErrorMessage="Invalid website URL"

OnServerValidate="cvWebsite\_ServerValidate"

CssClass="error" Display="Dynamic"></asp:CustomValidator> </div>

<asp:Button ID="btnSubmit" runat="server" Text="Submit"

OnClick="btnSubmit\_Click" />

<asp:ValidationSummary ID="ValidationSummary1" runat="server"

HeaderText="Please correct the following errors:"

ShowMessageBox="true" ShowSummary="false" />

</form>

</body>

</html>

### Registration.aspx.cs

using System;

using System.Web.UI.WebControls;

namespace ValidationDemo

{

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

{

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

{

}

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

{ if (Page.IsValid)

{

Response.Write("Registration successful!");

}

}

protected void cvWebsite\_ServerValidate(object source, ServerValidateEventArgs args)

{ try

{

Uri uri = new Uri(args.Value);

args.IsValid = (uri.Scheme == Uri.UriSchemeHttp || uri.Scheme ==

Uri.UriSchemeHttps);

}

catch {

args.IsValid = false;

}

}}}

**Output:**

**B) Create a web form to demonstrate the AdRotator Control.**

**Code:** Misba

Misba@gmail.com

### XML File

<Advertisements>

<Ad>

<ImageUrl>rose1.jpg</ImageUrl>

<NavigateUrl>http://www.1800flowers.com</NavigateUrl>

<AlternateText>

Order flowers, roses, gifts and more

</AlternateText>

<Impressions>20</Impressions>

<Keyword>flowers</Keyword>

</Ad>

<Ad>

<ImageUrl>rose2.jpg</ImageUrl>

<NavigateUrl>http://www.babybouquets.com.au</NavigateUrl>

<AlternateText>Order roses and flowers</AlternateText>

<Impressions>20</Impressions>

<Keyword>gifts</Keyword>

</Ad>

<Ad>

<ImageUrl>rose3.jpeg</ImageUrl>

<NavigateUrl>http://www.flowers2moscow.com</NavigateUrl>

<AlternateText>Send flowers to Russia</AlternateText>

<Impressions>20</Impressions>

<Keyword>russia</Keyword>

</Ad>

</Advertisements>

### Default.aspx

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

Inherits="WebApplication1.WebForm1" %>

<!DOCTYPE html>

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

<head runat="server">

<title>Advertisements</title>

</head>

<body>

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

<div>

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

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

</div>

</form>

</body>

</html> **Output:**

C) **Create web form to demonstrate use User Controls** **Code:**

**LoginControl.ascx:**

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

CodeBehind="LoginControl.ascx.cs" Inherits="YourNamespace.LoginControl" %>

<div>

<h2>Login</h2>

<div>

<label for="txtUsername">Username:</label>

<asp:TextBox ID="txtUsername" runat="server"></asp:TextBox> </div>

<div>

<label for="txtPassword">Password:</label>

<asp:TextBox ID="txtPassword" runat="server"

TextMode="Password"></asp:TextBox>

</div> <div>

<asp:Button ID="btnLogin" runat="server" Text="Login" OnClick="btnLogin\_Click"/> </div>

<asp:Label ID="lblMessage" runat="server" ForeColor="Red"></asp:Label>

</div>

Add code-behind for LoginControl.ascx.cs: using System;

namespace YourNamespace

{

public partial class LoginControl : System.Web.UI.UserControl

{

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

{

}

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

{

if (txtUsername.Text == "admin" && txtPassword.Text == "password")

{

lblMessage.Text = "Login successful!";

} else

{

lblMessage.Text = "Invalid username or password.”

}

}

}

}

### Default.aspx

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

<%@ Register Src="~/LoginControl.ascx" TagPrefix="uc" TagName="LoginControl" %>

<!DOCTYPE html>

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

<head runat="server">

<title>User Control Demo</title>

</head>

<body>

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

<div>

<h1>Welcome to User Control Demo</h1>

<uc:LoginControl runat="server" ID="LoginControl" /> </div>

</form>

</body>

</html>

**Output:**

**PRACTICAL 5**

**Create Web Form to demonstrate use of Website Navigation controls and Site Map.**

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

**Code:**

Web.sitemap file:

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

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

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

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

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

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

<siteMapNode url="~/Product1.aspx" title="Product 1" description="Product 1 page" /> <siteMapNode url="~/Product2.aspx" title="Product 2" description="Product 2 page" /> </siteMapNode>

</siteMapNode>

</siteMap>

**Site.Master file:**

<%@ Master Language="C#" AutoEventWireup="true" CodeBehind="Site1.master.cs" Inherits="Pract5A.Site1" %>

<!DOCTYPE html>

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

<head runat="server">

<title></title>

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

</asp:ContentPlaceHolder>

</head>

<body>

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

<div>

<asp:Menu ID="NavigationMenu" runat="server"

DataSourceID="SiteMapDataSource1" Orientation="Horizontal">

</asp:Menu>

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

<asp:SiteMapPath ID="SiteMapPath1" runat="server">

</asp:SiteMapPath>

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

</asp:ContentPlaceHolder>

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

DataSourceID="SiteMapDataSource1">

</asp:TreeView>

</div>

</form>

</body>

</html>

### WebForm1.aspx

<%@ Page Language="C#" MasterPageFile="~/Site1.Master" AutoEventWireup="true"

CodeBehind="WebForm1.aspx.cs" Inherits="Pract5A.WebForm1" %>

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

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="MainContent" runat="server">

<h1>Welcome to the Home Page</h1>

<p>This is the main content of the home page.</p>

</asp:Content>

### Product1.aspx

<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Product1.aspx.cs" Inherits="Pract5A.Product1" %>

<!DOCTYPE html>

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

<head runat="server">

<title></title>

</head>

<body>

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

<div>

<h1>This is Product Page</h1>

</div>

</form>

</body>

</html>

### Product2.aspx

<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Product2.aspx.cs" Inherits="Pract5A.Product2" %>

<!DOCTYPE html>

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

<head runat="server">

<title></title>

</head>

<body>

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

<div>

<h2>Product 2 Page</h2>

</div>

</form>

</body>

</html>

**Output:**

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

**Code:**

### Master1.master

<%@ Master Language="C#" AutoEventWireup="true" CodeBehind="Site1.master.cs"

Inherits="prac5b.Site1" %>

<!DOCTYPE html>

<html>

<head runat="server">

<title></title>

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

</asp:ContentPlaceHolder>

</head>

<body>

<link href="StyleSheet1.css" rel="stylesheet" type="text/css" />

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

<div>

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

</div> </form> </body>

</html>

### WebForm1.aspx

<%@ Page Title="" Language="C#" MasterPageFile="~/Site1.Master"

AutoEventWireup="true" CodeBehind="WebForm1.aspx.cs" Inherits="prac5b.WebForm1"

Theme ="Skin1"%>

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

<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">

<asp:Label ID="Label1" runat="server" SkinId="lbl" Text="Select The date"></asp:Label>

<asp:Calendar ID="Calendar1" runat="server"></asp:Calendar>

<br />

<asp:HyperLink ID="HyperLink1" runat="server"

NavigateUrl="~/WebForm2.aspx">Next</asp:HyperLink>

</asp:Content>

WebForm2.aspx

<%@ Page Title="" Language="C#" MasterPageFile="~/Site1.Master"

AutoEventWireup="true" CodeBehind="WebForm2.aspx.cs" Inherits="prac5b.WebForm2"

Theme="Skin1" %>

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

<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">

<p>

<br />

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

<p>

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

</p>

</asp:Content>

Skin1.skin

<asp:Label runat="server" SkinId="lbl" backcolor="blue"/>

StyleSheet1.css body {

background-color: gray; font:italic;

}

**Output:**

### B) Create a web application to demonstrate various states of ASP.NET Pages

**i) View State Code:**

### WebForm1.aspx

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

Inherits="Practical\_5c.WebForm1" %>

<!DOCTYPE html>

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

<head runat="server">

<title>ViewState Demo</title>

</head>

<body>

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

<div>

<h1> ViewState Demo</h1>

<h2>1. Basic ViewState</h2>

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

<asp:Button ID="btnBasic" runat="server" Text="Update"

OnClick="btnBasic\_Click"/>

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

<h2>2.ViewState Disabled</h2>

<asp:TextBox ID="txtDisabled" runat="server"

EnableViewState="false"></asp:TextBox>

<asp:Button ID="btnDisabled" runat="server" Text="Update"

OnClick="btnDisabled\_Click" />

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

<h3>3. Custom ViewState</h3>

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

<asp:Button ID="btnCustom" runat="server" Text="Increment"

OnClick="btnCustom\_Click" />

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

</div>

</form>

</body>

</html>

### WebForm1.aspx.cs

using System;

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

using System.Web.UI.WebControls;

namespace Practical\_5c

{

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

{

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

{

if(!IsPostBack)

{

ViewState["Counter"] = 0;

}

}

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

{

lblBasic.Text = $"You entered: {txtBasic.Text}";

}

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

{

lblDisabled.Text = $"You entered: {txtDisabled.Text}";

}

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

{

int counter = (int)ViewState["Counter"]; counter++;

ViewState["Counter"] = counter; lblCustom.Text = $"Counter: {counter}";

}

} }

**Output:**

### ii) Session State

**Code:**

WebForm1.aspx

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

Inherits="Practical\_5c.\_2.WebForm1" %>

<!DOCTYPE html>

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

<head runat="server">

<title>Session State Demo</title>

</head>

<body>

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

<div>

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

<asp:Button ID="btnSaveSession" runat="server" Text="Save to Session"

OnClick="btnSaveSession\_Click"/>

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

<asp:Button ID="btnRetrieveSession" runat="server" Text="Retrieve from Session" OnClick="btnRetrieveSession\_Click" />

</div>

</form>

</body>

</html>

WebForm1.aspx.cs

using System;

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

using System.Web.UI.WebControls;

namespace Practical\_5c.\_2

{

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

{

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

{

Session["UserName"] = txtName.Text;

lblSessionResult.Text = "Name saved to session!";

}

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

{

if(Session["UserName"] != null)

{

lblSessionResult.Text = "Stored Name: " + Session["UserName"].ToString();

} else

{

lblSessionResult.Text = "No name found in session";

}

}}}

**Output:**

# PRACTICAL 6

## AIM: Demonstrate the use of DataList link Control

**Code:**

### Default.aspx

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

Inherits="prac6\_c.WebForm1" %>

<!DOCTYPE html>

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

<head runat="server">

<title></title>

</head>

<body>

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

<div>

<h2>Book List</h2>

<asp:DataList ID="dlBooks" runat="server" RepeatColumns="2">

<ItemTemplate>

<div style="margin-bottom: 10px; padding: 10px; border:1px solid #ccc;">

<h3><%#Eval("Title") %></h3>

<p>Author: <%#Eval("Author") %></p>

|  |
| --- |
| %>  %> |

<p>Price: $<%# Eval("Price", "{0:F2}") </p>

<p>Price: $<%# Eval("Price", "{0:F2}") </p>

</div>

</ItemTemplate>

</asp:DataList>

</div>

### 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 prac6\_c

{

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

{

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

{

if (!IsPostBack)

{

BindDataList();

}

}

private void BindDataList()

{

List<Book> books = new List<Book>

{

new Book { Title = "The Great Gatsby", Author = "F. Scott Fitzgerald", Price =

12.99m}, new Book { Title = "To Kill a MockingBird", Author = "George Orwell", Price =

11.99m}, new Book { Title = "Pride and Prejudice", Author = "Jane Austen", Price = 9.99m}

};

dlBooks.DataSource = books; dlBooks.DataBind();

}

}

public class Book

{

public string Title { get; set; } public string Author { get; set; }

public decimal Price { get; set; }

}

}

</form>

</body>

</html>

**Output:**

# PRACTICAL 7

**AIM: Working with Database**

**A) Create a web application for inserting and deleting record from a database (Using Execute Non-Query)**

**Code:**

## Default.aspx

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

Inherits="DatabaseWebApp.Default" %>

<!DOCTYPE html>

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

<head runat="server">

<title>Database Operations</title>

</head>

<body>

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

<div>

<h2>Insert Record</h2>

<asp:TextBox ID="txtName" runat="server" placeholder="Name"></asp:TextBox>

<asp:TextBox ID="txtEmail" runat="server" placeholder="Email"></asp:TextBox>

<asp:Button ID="btnInsert" runat="server" Text="Insert"

OnClick="btnInsert\_Click"/>

<h2>Delete Record</h2>

<asp:TextBox ID="txtId" runat="server" placeholder="ID"></asp:TextBox>

<asp:Button ID="btnDelete" runat="server" Text="Delete"

OnClick="btnDelete\_Click" />

<h2>Records</h2>

<asp:GridView ID="gvRecords" runat="server"

AutoGenerateColumns="true"></asp:GridView>

</div>

</form>

</body>

</html>

## Default.aspx.cs

using System; using System.Configuration; using System.Data; using System.Data.SqlClient; using System.Web.UI; namespace DatabaseWebApp

{

public partial class Default : Page

{

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

{

if (!IsPostBack)

{

BindGridView();

}

}

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

{

string name = txtName.Text;

string email = txtEmail.Text;

string query = "INSERT INTO Users (Name, Email) VALUES (@Name, @Email)";

ExecuteNonQuery(query, new SqlParameter("@Name", name), new

SqlParameter("@Email", email));

BindGridView();

ClearInputs();

}

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

{ int id;

if (int.TryParse(txtId.Text, out id))

{

string query = "DELETE FROM Users WHERE Id = @Id"; ExecuteNonQuery(query, new SqlParameter("@Id", id));

BindGridView();

ClearInputs();

}

}

private void ExecuteNonQuery(string query, params SqlParameter[] parameters)

{

string connectionString =

ConfigurationManager.ConnectionStrings["DefaultConnection"].ConnectionString;

using (SqlConnection connection = new SqlConnection(connectionString))

{

using (SqlCommand command = new SqlCommand(query, connection))

{

command.Parameters.AddRange(parameters); connection.Open(); command.ExecuteNonQuery();

}

}

}

private void BindGridView()

{

string connectionString =

ConfigurationManager.ConnectionStrings["DefaultConnection"].ConnectionString; string query = "SELECT \* FROM Users";

using (SqlConnection connection = new SqlConnection(connectionString))

{

using (SqlCommand command = new SqlCommand(query, connection))

{

connection.Open();

SqlDataAdapter adapter = new SqlDataAdapter(command);

DataTable dt = new DataTable();

adapter.Fill(dt); gvRecords.DataSource = dt;

gvRecords.DataBind();

}

}

}

private void ClearInputs()

{

txtName.Text = string.Empty; txtEmail.Text = string.Empty;

txtId.Text = string.Empty;

}

}

}

**Output:**

# PRACTICAL 8

**Aim: Create a web application to demonstrate the use of Ajax Controls**

**Code:**

## Default.aspx

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

Inherits="WebApplication7.WebForm1" %>

<!DOCTYPE html>

<!DOCTYPE html>

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

<head runat="server">

<title>Simple AJAX Demo</title>

</head>

<body>

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

<asp:ScriptManager ID="ScriptManager1" runat="server"></asp:ScriptManager>

<div>

<h1>Simple AJAX Demo</h1>

<h2>1. UpdatePanel Example</h2>

<asp:UpdatePanel ID="UpdatePanel1" runat="server">

<ContentTemplate>

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

<asp:Button ID="btnUpdateTime" runat="server" Text="Update Time"

OnClick="btnUpdateTime\_Click" />

</ContentTemplate>

</asp:UpdatePanel>

</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 WebForm1 : System.Web.UI.Page

{

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

{

}

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

{

lblTime.Text = "Current time:" + DateTime.Now.ToString("HH:mm:ss");

}

}

}

**Output:**