Roll No.**24** Exam Seat No._____

VIVEKANANDEDUCATION SOCIETY'S INSTITUTE OF TECHNOLOGY

Hashu Advani Memorial Complex, Collector's Colony, R. C. Marg,Chembur, Mumbai – 400074. Contact No. 02261532532

Since 1962

CERTIFICATE

Certified that Mr. **NARENDER KESWANI [ROLL NO: 24]** of **FYMCA-1B** has satisfactorily completed a course of the necessary experiments in **MCAL24 – Advanced Web Technologies Lab** under the supervision of **Ms. Ruchi Rautela** in the Institute of Technology in the academic year **2021- 2022.**

Principal Head of Department

Lab In-charge Subject Teacher

V.E.S. Institute of Technology, Collector Colony, Chembur, Mumbai

Department of M.C.A

AWT INDEX

Sr. No	Contents	Date Of Preparation	Date Of Submission	Marks	Sign
1	Design UI based applications using basic Windows forms Controls	09/05/2022	17/05/2022	10	
2	Design Applications using Classes and Objects	10/05/2022	17/05/2022	10	
3	Design Applications using Inheritance and Abstract Classes	17/05/2022	23/05/2022	10	
4	Design a Web Application for an Organization with Registration forms and advanced controls	23/05/2022	30/05/2022	10	
5	Design a webpage to demonstrate a connection oriented architecture	31/05/2022	13/06/2022	10	
6	Design a webpage to demonstrate the use of Databound controls and stored procedure	14/06/2022	20/06/2022	10	
7	Design a webpage to display the use of LINQ and Entity Framework	21/06/2022	27/06/2022	10	
8	Design Web Applications using Client Side Session Management	27/06/2022	04/07/2022	10	
9	Design a web application to Produce and consume a web service	28/06/2022	05/07/2022		
10	Design MVC based Web applications	04/07/2022	05/07/2022		

AIM: Design UI based applications using basic Windows forms Controls

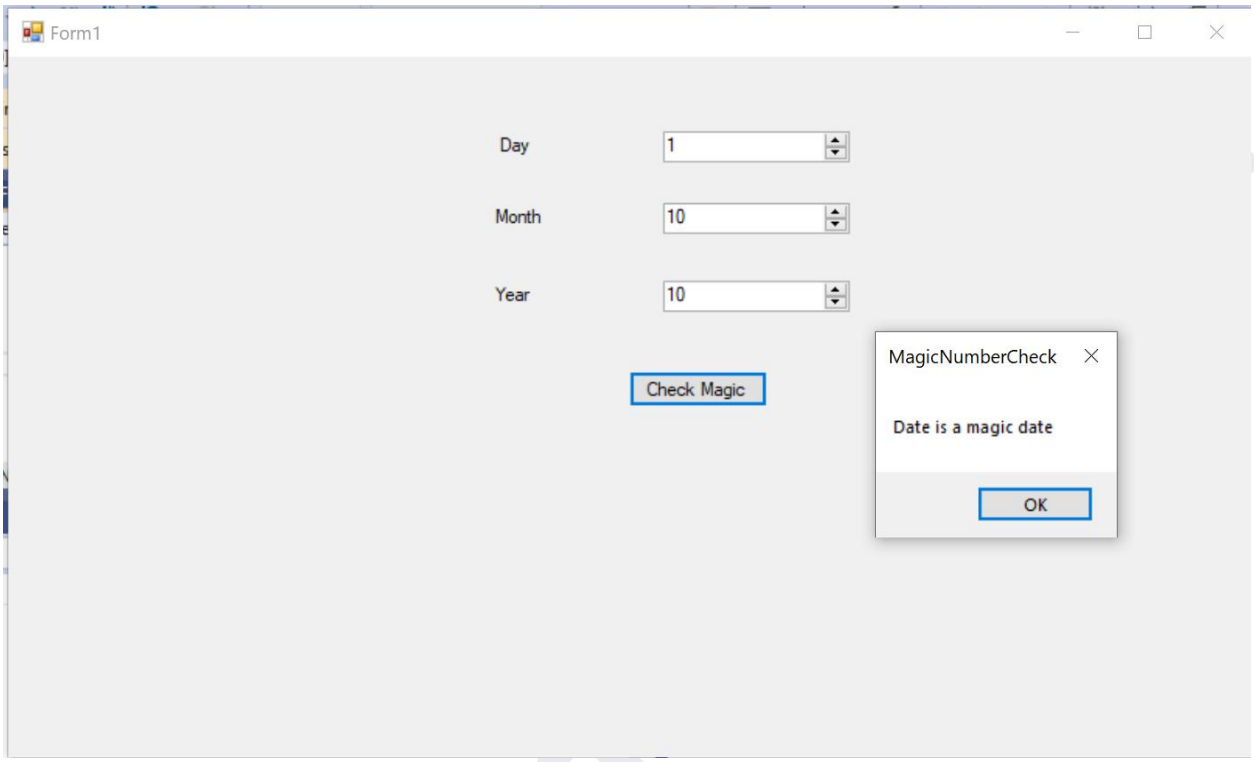
A) Write a Program in C# that ask the user to enter a month, a day and a two digit year. The program should then determine whether the month times a day is equal to the year. If so, it should display the message saying the date is magic. Otherwise not a magic.

SOURCE CODE:

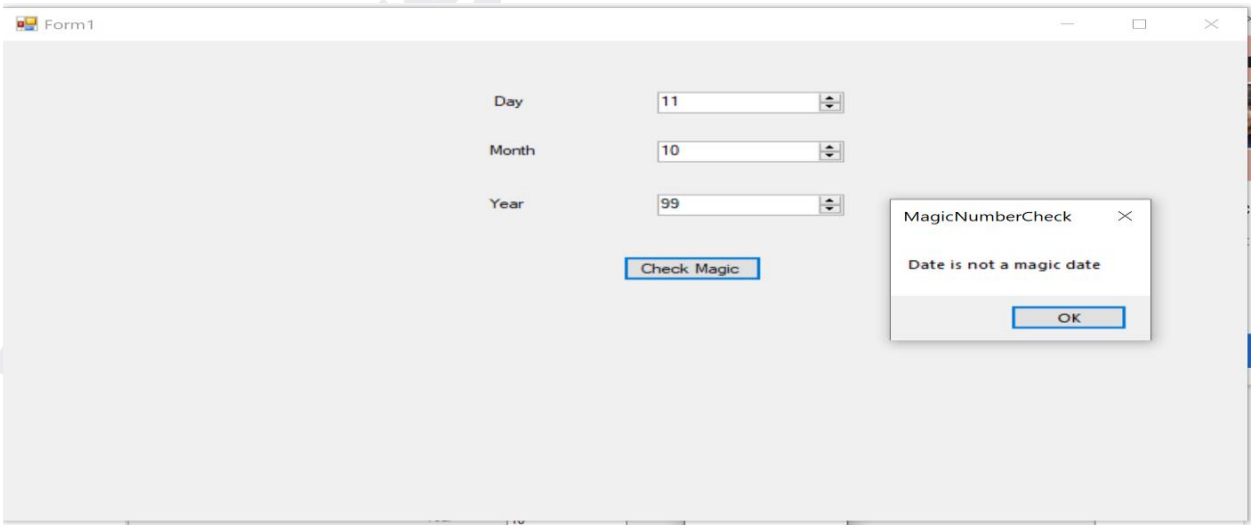
```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace p1MagicNumber
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            int month = Convert.ToInt32(numericUpDown1.Text);
            int date = Convert.ToInt32(numericUpDown2.Text);
            int year = Convert.ToInt32(numericUpDown3.Text);
            if (date * month == year)
            {
                MessageBox.Show("Date is a magic date", "MagicNumberCheck");
            }
            else
            {
                MessageBox.Show("Date is not a magic date", "MagicNumberCheck");
            }
        }
    }
}
```

OUTPUT:

CASE-I:



CASE-II:



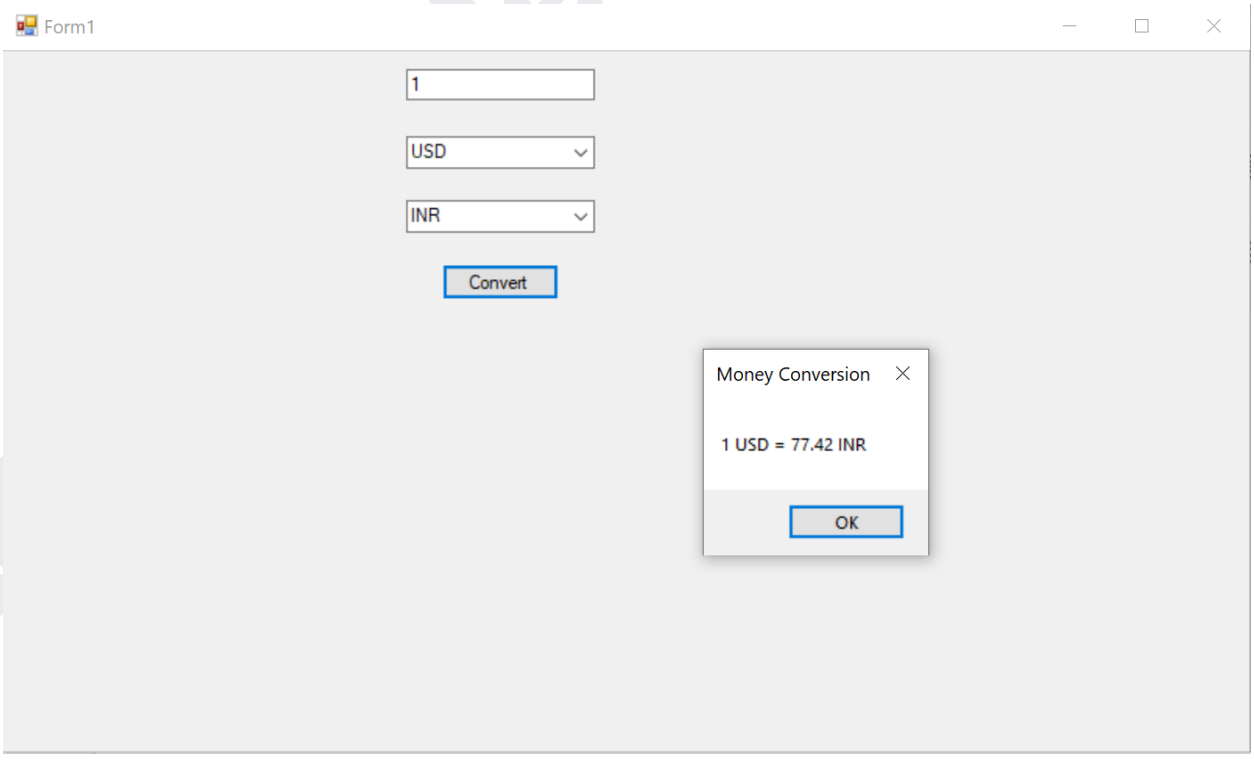
B) Write a Program to perform Money Conversion.

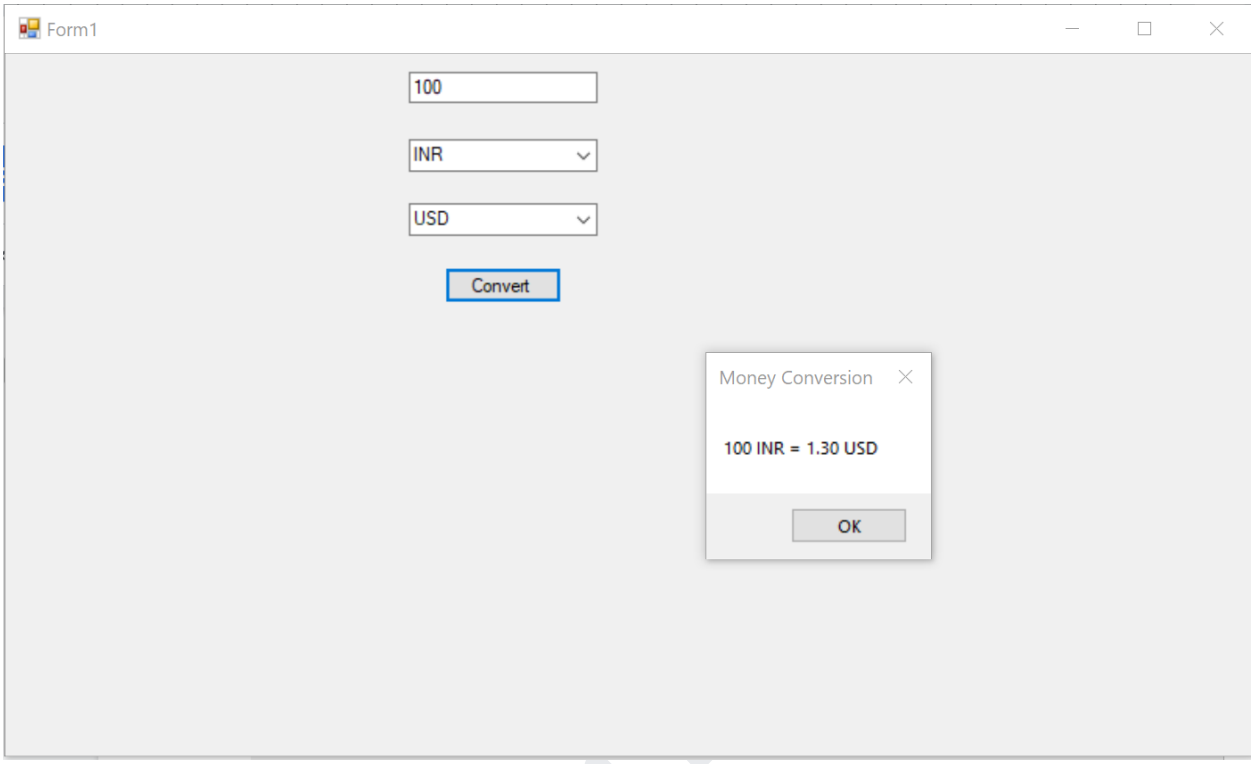
SOURCE CODE:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace p1MagicConversion
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            int amt = Convert.ToInt32(textBox1.Text);
            double value = 0;
            if (comboBox1.Text == comboBox2.Text)
            {
                MessageBox.Show("Conversion Formats can't be same", "Money
Conversion");

            }
            else if (comboBox1.Text == "INR" && comboBox2.Text == "USD")
            {
                value = amt * 0.013;
            }
            else if (comboBox1.Text == "INR" && comboBox2.Text == "EUR")
            {
                value = amt * 0.012;
            }
            else if (comboBox1.Text == "USD" && comboBox2.Text == "INR")
            {
                value = amt * 77.42;
```

```
            }
            else if (comboBox1.Text == "USD" && comboBox2.Text == "EUR")
            {
                value = amt * 0.95;
            }
            else if (comboBox1.Text == "EUR" && comboBox2.Text == "INR")
            {
                value = amt * 81.96;
            }
            else if (comboBox1.Text == "EUR" && comboBox2.Text == "USD")
            {
                value = amt * 1.05;
            }

            MessageBox.Show(amt + " " + comboBox1.Text + " = "+value.ToString("0.00")
+" "+ comboBox2.Text, "Money Conversion");

        }
    }
}
```

OUTPUT:





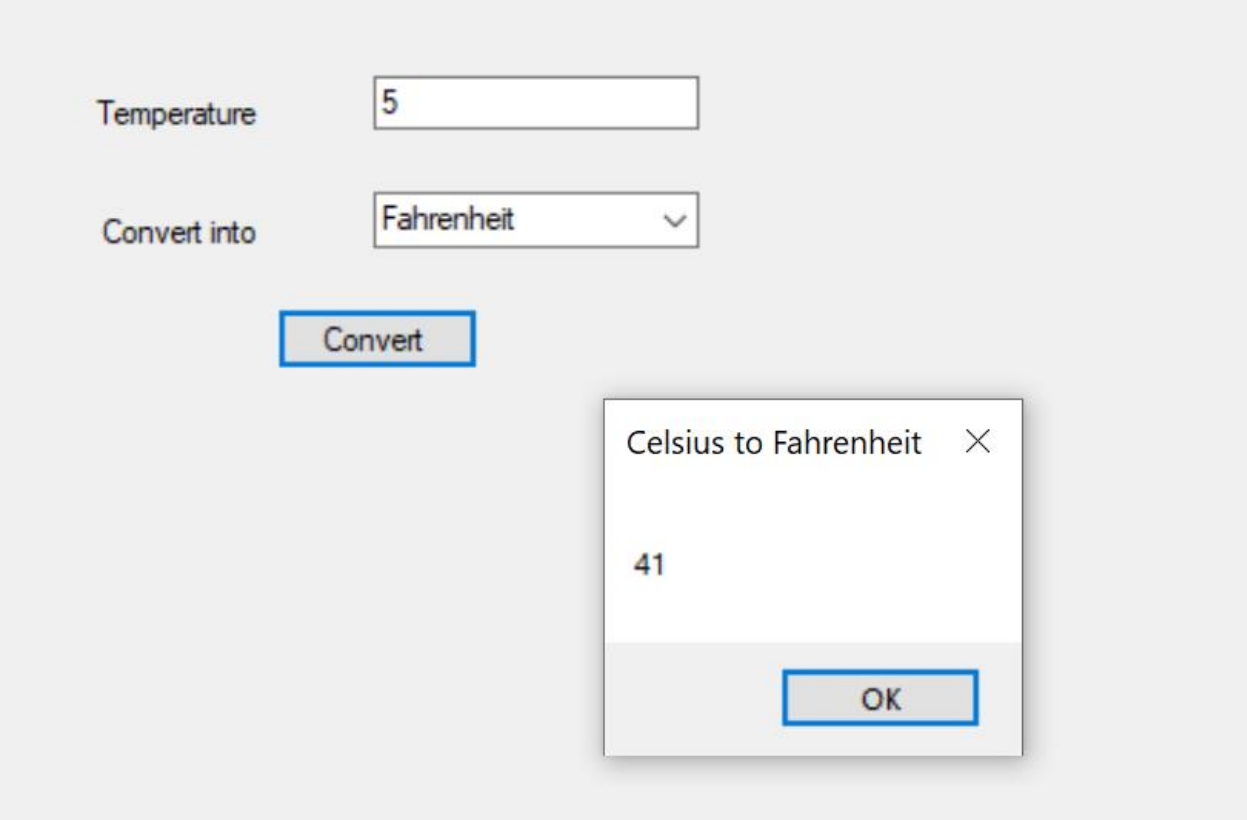
C) **To convert temperature from Fahrenheit to Celsius or vice versa.**

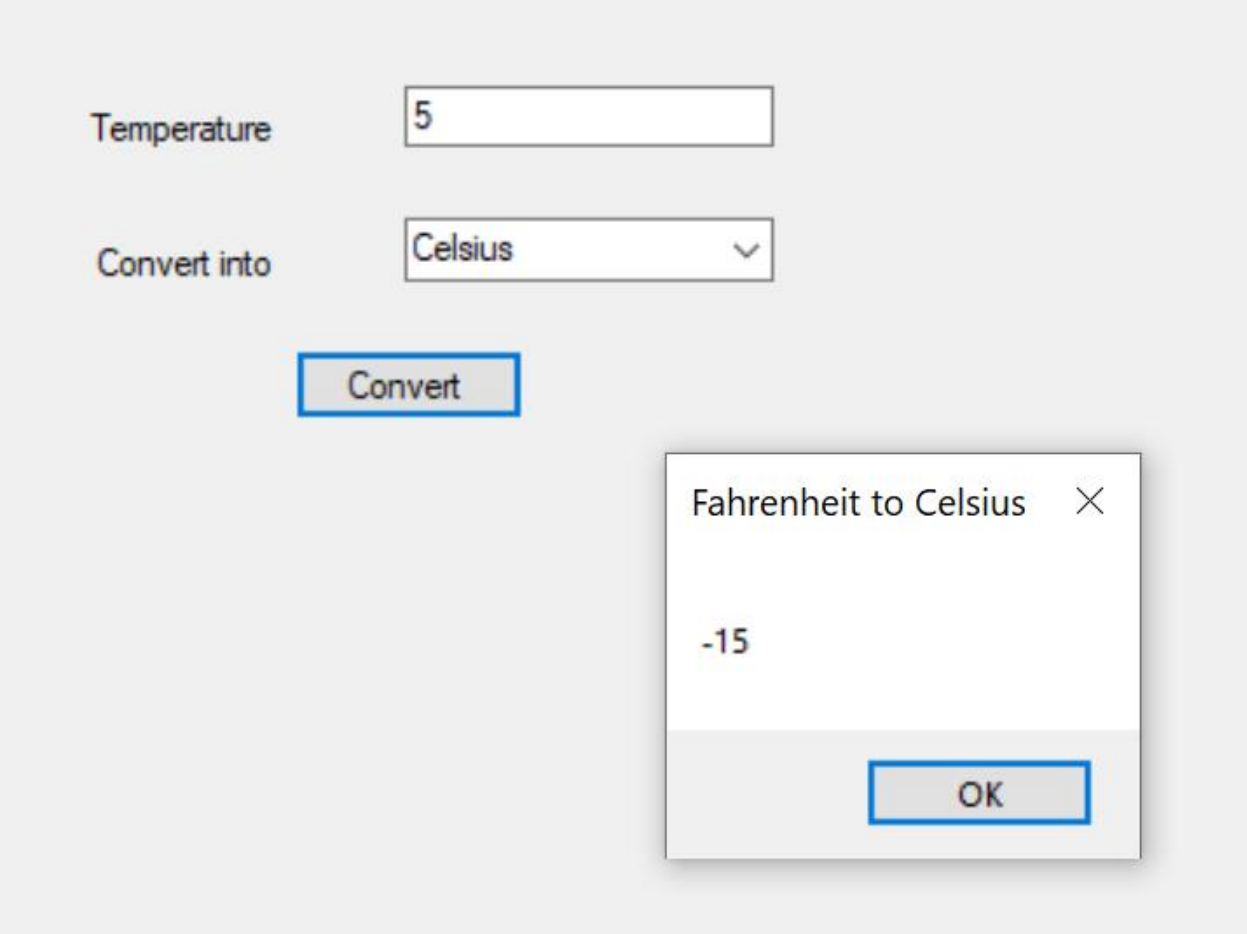
SOURCE CODE:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace p1Temperature
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
```

```
        }

        private void button1_Click(object sender, EventArgs e)
        {

            int temp = Int16.Parse(textBox1.Text);
            double value = 0;

            if (comboBox1.Text == "Fahrenheit")
            {
                value = (temp * 9 / 5) + 32;
                MessageBox.Show(value.ToString(), "Celsius to Fahrenheit");
            }

            if (comboBox1.Text == "Celsius")
            {
                value = (temp - 32) * 5 / 9;
                MessageBox.Show(value.ToString(), "Fahrenheit to Celsius");
            }

        }
    }
}
```

OUTPUT:


Temperature
5
Convert into
Fahrenheit
Convert
Celsius to Fahrenheit
41
OK



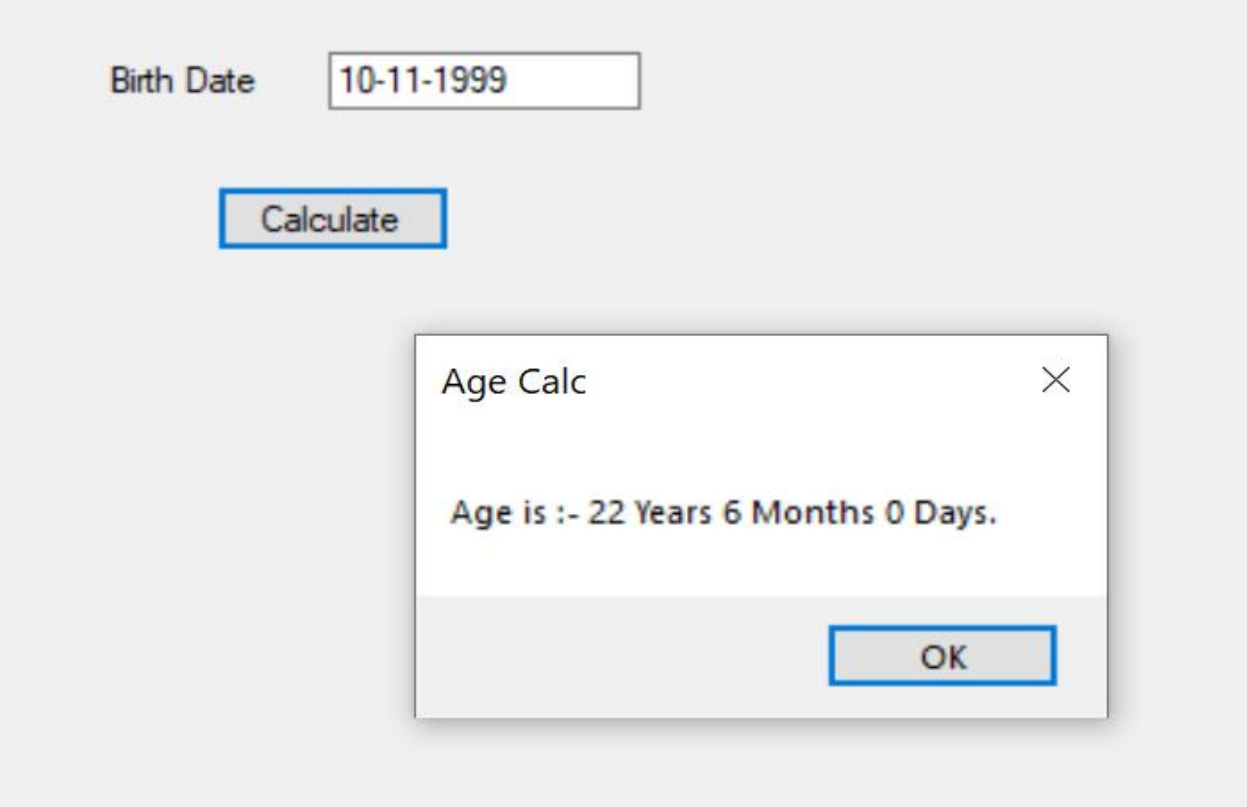
D) Create a Window application to calculate age of a person by providing input as birth date and current date .Current date and Birth date must be in long string format and display the age in terms of years

SOURCE CODE:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace p1Birthday
{
    public partial class Form1 : Form
```

```
    {
        public Form1()
        {
            InitializeComponent();

        }

        private void button1_Click(object sender, EventArgs e)
        {
            string textBox2 = Convert.ToString(DateTime.Now.ToLongDateString());
            DateTime bdate = Convert.ToDateTime(textBox1.Text);
            DateTime cdate = Convert.ToDateTime(textBox2);
            int years = (cdate.Year - bdate.Year) - 1;
            int months = 12 - Math.Abs(cdate.Month - bdate.Month);
            int days = cdate.Day - bdate.Day;

            MessageBox.Show("Age is :- " + years + " Years " + months
            + " Months " + days + " Days. ", "Age Calc");
        }
    }
}
```

OUTPUT:



CONCLUSION:

From this practical, I have learned about the basics of windows forms with c#.

Aim: Design Applications using Classes and Objects

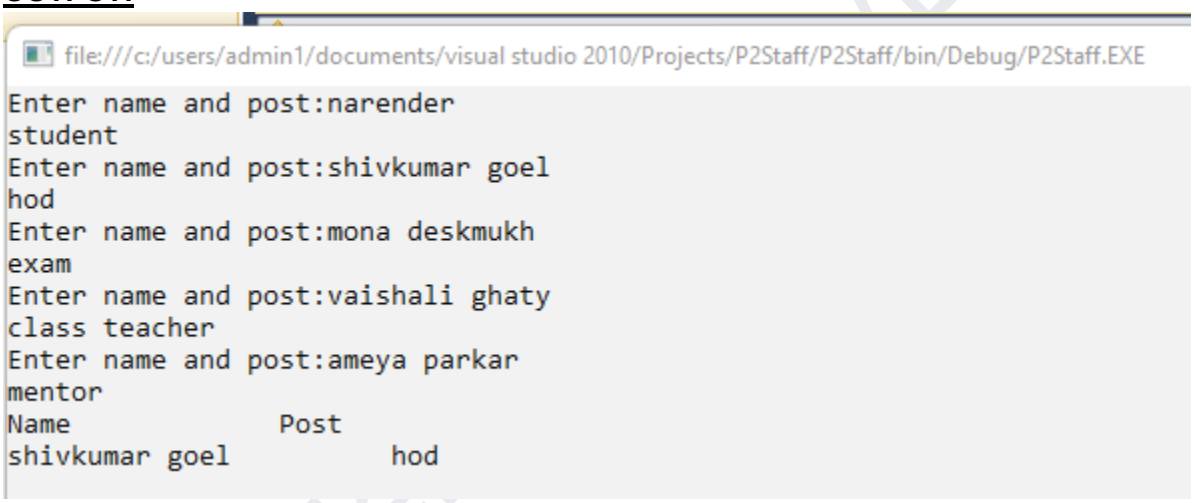
a) Write a program to declare a class "staff" having data members as name and post.accept this data 5 staffs and display names of staff who are HOD.

SOURCE CODE:

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

namespace P2Staff
{
   class staff
   {
      string name, post;
      public void getdata()
      {
         Console.Write("Enter name and post:");
         name = Console.ReadLine();
         post = Console.ReadLine();
      }
      public void display()
      {
         Console.WriteLine(name + "\t\t" + post);
      }
      public string getPost()
      {
         return post;
      }
   }

   class Program
   {
      static void Main(string[] args)
      {
         staff[] objStaff = new staff[5];
         int i;
         for (i = 0; i < 5; i++)
         {
            objStaff[i] = new staff();
            objStaff[i].getdata();
         }
         Console.WriteLine("Name \t\t Post");
```

```
            for (i = 0; i < 5; i++)
            {
                if (objStaff[i].getPost() == "HOD" || objStaff[i].getPost() == "hod")
                    objStaff[i].display();
            }
            Console.ReadLine();

        }
    }
}
```

OUTPUT:



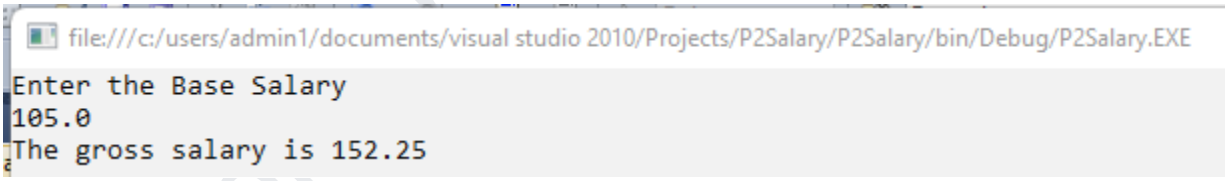
b) Define a class "salary" which will contain member variables Basic, TA, DA, HRA. Write a program using Constructor with default values for DA and HRA and calculate the salary of the employee.

SOURCE CODE:

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

namespace P2Salary
{
   class salary{
   float bsal, hra, da, ta;
   float gsal;
   public void read()
   {
   Console.WriteLine("Enter the Base Salary");
```

```
    bsal = float.Parse(Console.ReadLine());
    }
    public float sal()
    {
    hra=20*bsal/100;
    da=10*bsal/100;
    ta=15*bsal/100;
    gsal=bsal+da+hra+ta;
    return gsal;
    }
    }
    class Program
    {
       static void Main(string[] args)
       {
          salary s = new salary();
          s.read();
          Console.WriteLine("The gross salary is {0}", s.sal());
          Console.ReadLine();
       }
    }
}
```

OUTPUT:



CONCLUSION:

From this practical, I have learned about object-oriented programming concepts in c#.

AIM: Design Applications using Inheritance and Abstract Classes

A) Write a program to implement multilevel inheritance from the following figure. Accept and display data for one student.

SOURCE CODE:

```
using System;

namespace P3MultiLevelInheritance
{
    public class Student
    {
        int rollno;
        string name;
        public void getStudentData()
        {
            Console.WriteLine("Enter Roll No:");
            rollno = Int32.Parse(Console.ReadLine());
            Console.WriteLine("Enter Name:");
            name = Console.ReadLine();
        }
        public void displayStudentData()
        {
            Console.WriteLine("Roll No: "+ rollno);
            Console.WriteLine("Name: "+ name);
        }
    }

    public class Test : Student
    {
        public int marks1, marks2;
        public void getMarks()
        {
            getStudentData();
            Console.WriteLine("Enter Marks 1:");
            marks1 = Int32.Parse(Console.ReadLine());
            Console.WriteLine("Enter Marks 2:");
            marks2 = Int32.Parse(Console.ReadLine());
        }
        public void setMarks()
        {
            displayStudentData();
            Console.WriteLine("Marks1: "+ marks1);
```

```
            Console.WriteLine("Marks2: "+ marks2);
        }
    }

    public class Result : Test
    {
        int calc;
        public void getCalc()
        {
            getMarks();
            calc = (marks1 + marks2)/2;
            setMarks();
            Console.WriteLine("Total: "+ calc);
        }
    }

    class Program
    {
        static void Main(string[] args)
        {
            Result r = new Result();
            r.getCalc();
            Console.ReadLine();
        }
    }
}
```

OUTPUT:

```
C:\Users\NARENDER KESWANI\source\repos\P3MultiLevelInheritance\P3MultiLevelInheritance\bin\...
Enter Roll No:
24
Enter Name:
Narender Keswani
Enter Marks 1:
14
Enter Marks 2:
10
Roll No: 24
Name: Narender Keswani
Marks1: 14
Marks2: 10
Total: 12
```

B) Create a Super Class Student and two subclasses of it, Graduate and UnderGraduate. The members of the Student are name, id, grade, age and address and one method : boolean method IsPassed which takes in the parameter integer grade(0-100) and return true.The two subclasses override the method, for UG its 70% for passing and for G its 80% as passing grade.

SOURCE CODE:

```
using System;

namespace P3Overriding
{
   public class Student
   {
      public int id, grade, age;
      public string name, address;
      public void getData()
      {
         Console.WriteLine("Enter ID:");
         id = Int32.Parse(Console.ReadLine());
         Console.WriteLine("Enter Name:");
         name = Console.ReadLine();
         Console.WriteLine("Enter age:");
         age = Int32.Parse(Console.ReadLine());
         Console.WriteLine("Enter address:");
         address = Console.ReadLine();
         Console.WriteLine("Enter Grade:");
         grade = Int32.Parse(Console.ReadLine());

      }

      public void displayData()
      {
         Console.WriteLine("ID: " + id);
         Console.WriteLine("Name: "+ name);
         Console.WriteLine("Age: "+ age);
         Console.WriteLine("Address: "+ address);
         Console.WriteLine("Grade: " + grade);
      }
      public virtual Boolean IsPassed()
      {
         return true;
      }
   }

   public class Graduate : Student
   {
```

```
        public override bool IsPassed()
        {
            if (grade >= 80 && grade <= 100)
            {
                return true;
            }
            else
            {
                return false;
            }
        }

        }

        public class Undergraduate : Student
        {

        public override bool IsPassed()
        {
            if (grade >= 70 && grade <= 100)
            {
                return true;
            }
            else
            {
                return false;
            }
        }

    }

    class Program
        {
            static void Main(string[] args)
            {
                Graduate g = new Graduate();
                g.getData();
                g.displayData();
                g.IsPassed();

                if (g.IsPassed() == true)
                {
                    Console.WriteLine("Passed in Graduation.");
                }
                else
                {
```

```
            Console.WriteLine("Failed in Graduation.");
        }

        Undergraduate ug = new Undergraduate();
        ug.getData();
        ug.displayData();
        ug.IsPassed();

        if (ug.IsPassed() == true)
        {
            Console.WriteLine("Passed in UnderGraduation.");
        }
        else
        {
            Console.WriteLine("Failed in UnderGraduation.");
        }

        Console.ReadLine();
    }
  }
}
```

OUTPUT:

```
C:\Users\NARENDER KESWANI\source\repos\P3Overriding\P3C
Narender Keswani
Enter age:
21
Enter address:
Ulhasnagar
Enter Grade:
82
ID: 1
Name: Narender Keswani
Age: 21
Address: Ulhasnagar
Grade: 82
Passed in Graduation.
Enter ID:
02
Enter Name:
Neel Deshmukh
Enter age:
66
Enter address:
nallasoapara
Enter Grade:
79
ID: 2
Name: Neel Deshmukh
Age: 66
Address: nallasoapara
Grade: 79
Passed in UnderGraduation.
```

C) Program to calculate To find the area of various shapes: Rectangle, Circle, Ellipse, Square and Triangle using abstract class and abstract method.

SOURCE CODE:

```
using System;

namespace P3Abstract
{
```

```
public abstract class Shape
{
   public double area;
   public abstract void Area();
   public void displayData()
   {
      Console.WriteLine("Area is: " + area);
   }
}

public class Rectangle : Shape
{
   public double len, breadth;

   public void getData()
   {
      Console.WriteLine("Enter length of Rectangle");
      len = Double.Parse(Console.ReadLine());
      Console.WriteLine("Enter breadth of Rectangle");
      breadth = Double.Parse(Console.ReadLine());
   }
   public override void Area()
   {
      getData();
      area = len * breadth;
      displayData();
   }
}

public class Circle : Shape
{
   public double r;
   public void getData()
   {
      Console.WriteLine("Enter radius of Circle");
      r = Double.Parse(Console.ReadLine());
   }
   public override void Area()
   {
      getData();
      area = 3.14 * r * r;
      displayData();
   }
}

public class Ellipse : Shape
{
```

```
    public double a,b;
    public void getData()
    {
        Console.WriteLine("Enter a axis of Ellipse");
        a = Double.Parse(Console.ReadLine());
        Console.WriteLine("Enter b axis of Ellipse");
        b = Double.Parse(Console.ReadLine());
    }
    public override void Area()
    {
        getData();
        area = 3.14 * a * b;
        displayData();
    }
}

public class Square : Shape
{
    public double s;
    public void getData()
    {
        Console.WriteLine("Enter side of Square");
        s = Double.Parse(Console.ReadLine());
    }
    public override void Area()
    {
        getData();
        area = s * s;
        displayData();
    }
}

public class Triangle : Shape
{
    public double h, b;
    public void getData()
    {
        Console.WriteLine("Enter height of Triangle");
        h = Double.Parse(Console.ReadLine());
        Console.WriteLine("Enter breadth of Triangle");
        b = Double.Parse(Console.ReadLine());
    }
    public override void Area()
    {
        getData();
        area = 0.5 * h * b;
        displayData();
```

```
        }
    }

    class Program
    {

        static void Main(string[] args)
        {
            Rectangle r = new Rectangle();
            r.Area();
            Circle c = new Circle();
            c.Area();
            Ellipse e = new Ellipse();
            e.Area();
            Square s = new Square();
            s.Area();
            Triangle t = new Triangle();
            t.Area();
            Console.ReadLine();
        }
    }
}
```

OUTPUT:

```
C:\Users\NARENDER KESWANI\source\repos\P3Abstract\I
Enter length of Rectangle
10
Enter breadth of Rectangle
20
Area is: 200
Enter radius of Circle
5
Area is: 78.5
Enter a axis of Ellipse
5
Enter b axis of Ellipse
6
Area is: 94.2
Enter side of Square
9
Area is: 81
Enter height of Triangle
14
Enter breadth of Triangle
6
Area is: 42
```

CONCLUSION:

From this practical, I have learned about types of inheritance and overriding in C#.

AIM: DESIGN A WEB APPLICATION FOR AN ORGANIZATION WITH REGISTRATION FORMS AND ADVANCED CONTROLS

A) **Design online registration form for the participation of technical events (use HTML Controls, validation controls) and display all the data on the other page. Also design a layout using Master Page.**

SOURCE CODE:

Site.master:

```
<%@ Master Language="C#" AutoEventWireup="true" CodeFile="Site.master.cs"
Inherits="SiteMaster" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
<head runat="server">
  <title></title>
  <link href="~/Styles/Site.css" rel="stylesheet" type="text/css" />

  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"
    rel="stylesheet" integrity="sha384-
EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOmLASjC"
    crossorigin="anonymous" />
  <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js"
    integrity="sha384-
MrcW6ZMFYlzcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/tWtIaxVXM"
    crossorigin="anonymous"></script>
  <script
src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.9.2/dist/umd/popper.min.js"
    integrity="sha384-
IQsoLXl5PILFhosVNubq5LC7Qb9DXgDA9i+tQ8Zj3iwWAwPtgFTxbJ8NT4GN1R8p"
    crossorigin="anonymous"></script>
  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.min.js"
    integrity="sha384-
cVKIPhGWiC2Al4u+LWgxfKTRIcfu0JTxR+EQDz/bgldoEyl4H0zUF0QKbrJ0EcQF"
    crossorigin="anonymous"></script>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/angular.js/1.8.2/angular.min.js"
integrity="sha512-
7oYXeK0OxTFxndh0erL8FsjGvrl2VMDor6fVqzlLGfwOQQqTbYsGPv4ZZ15QHfSk80doyaM0ZJdv
kyDcVO7KFA==" crossorigin="anonymous" referrerpolicy="no-referrer"></script>
  <asp:ContentPlaceHolder ID="HeadContent" runat="server">
  </asp:ContentPlaceHolder>
</head>
<body>
  <form id="Form1" runat="server" method="post">
    <div class="container" style="padding: 25px">
      <div style="font-weight: bold; margin-block-end: 10px; text-align: center; border-
style: solid">
        <h1>SMART INDIA HACKATHON 2022
        </h1>
```

```
        </div>
        <div style="float: left; width: 22%; height: auto; border-style: solid; margin-block-
start: 15px; padding: 10px; margin-left: 10px; margin-right: 5px">
          <asp:Menu ID="NavigationMenu" runat="server" CssClass="menu"
EnableViewState="false" IncludeStyleBlock="false" Orientation="Vertical">
            <Items>
              <asp:MenuItem NavigateUrl="~/Default.aspx" Text="Home" />
              <asp:MenuItem NavigateUrl="~/About.aspx" Text="About" />
            </Items>
          </asp:Menu>
        </div>
        <div style="float: left; width: 73%; border-style: solid; margin-block-start: 15px;
margin-left: 5px; padding: 10px; margin-block-end: 10px">
          <asp:ContentPlaceHolder ID="MainContent" runat="server" />
        </div>

        <p style="border-style: solid; text-align: center; font-weight: bold">
          Copyright 2022 , Developed by Narender Keswani
        </p>
      </div>
    </form>
</body>
</html>
```

Default.aspx:

```
<%@ Page Title="Home Page" Language="C#" MasterPageFile="~/Site.master"
AutoEventWireup="true"
    CodeFile="Default.aspx.cs" Inherits="_Default" %>

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

    <div class="mb-3">
      <label for="exampleFormControlInput1" class="form-label">Email address</label>
      <asp:TextBox ID="email" runat="server" CssClass="form-control" placeholder="Enter
your email"></asp:TextBox>
      <asp:RequiredFieldValidator ID="RequiredFieldValidator1"
        runat="server" ErrorMessage="Please enter your email"
        ControlToValidate="email"></asp:RequiredFieldValidator>
      <asp:RegularExpressionValidator
        ID="regEmail"
        ControlToValidate="email"
        Text="Enter valid email id"
        ValidationExpression="\w+([-+.']\w+)*@\w+([-.]\w+)*\.\w+([-.]\w+)*"
        runat="server" />
    </div>

    <div class="mb-3">
      <label for="exampleFormControlInput1" class="form-label">Select College:</label>
      <asp:DropDownList ID="college" runat="server" CssClass="form-select">
```

```
            <asp:ListItem Value="">Please Select</asp:ListItem>
            <asp:ListItem>JHC </asp:ListItem>
            <asp:ListItem>SPIT</asp:ListItem>
            <asp:ListItem>VESIT</asp:ListItem>
          </asp:DropDownList>
          <asp:RequiredFieldValidator ID="RequiredFieldValidator9"
            runat="server" ErrorMessage="Please select your college"
            ControlToValidate="college"></asp:RequiredFieldValidator>
        </div>

        <div class="mb-3">
          <label>Team Mentor Name:</label>
          <asp:TextBox ID="tmn" runat="server" CssClass="form-control" placeholder="Enter
your Team Mentor"></asp:TextBox>
          <asp:RequiredFieldValidator ID="RequiredFieldValidator2"
            runat="server" ErrorMessage="Please enter team mentor name"
            ControlToValidate="tmn"></asp:RequiredFieldValidator>
        </div>

        <div class="mb-3">
          <label>Team Member 1 Name:</label>
          <asp:TextBox ID="tm1" runat="server" CssClass="form-control" placeholder="Enter
your Team Member Name 1"></asp:TextBox>
          <asp:RequiredFieldValidator ID="RequiredFieldValidator3"
            runat="server" ErrorMessage="Please enter team member name 1"
            ControlToValidate="tm1"></asp:RequiredFieldValidator>
        </div>

        <div class="mb-3">
          <label>Team Member 2 Name:</label>
          <asp:TextBox ID="tm2" runat="server" CssClass="form-control" placeholder="Enter
your Team Member Name 2"></asp:TextBox>
          <asp:RequiredFieldValidator ID="RequiredFieldValidator4"
            runat="server" ErrorMessage="Please enter team member name 2"
            ControlToValidate="tm2"></asp:RequiredFieldValidator>
        </div>

        <div class="mb-3">
          <label>Team Member 3 Name:</label>
          <asp:TextBox ID="tm3" runat="server" CssClass="form-control" placeholder="Enter
your Team Member Name 3"></asp:TextBox>
          <asp:RequiredFieldValidator ID="RequiredFieldValidator5"
            runat="server" ErrorMessage="Please enter team member name 3"
            ControlToValidate="tm3"></asp:RequiredFieldValidator>
        </div>

        <div class="mb-3">
          <label>Team Member 4 Name:</label>
          <asp:TextBox ID="tm4" runat="server" CssClass="form-control" placeholder="Enter
your Team Member Name 4"></asp:TextBox>
          <asp:RequiredFieldValidator ID="RequiredFieldValidator6"
```

```
            runat="server" ErrorMessage="Please enter team member name 4"
            ControlToValidate="tm4"></asp:RequiredFieldValidator>
        </div>

        <div class="mb-3">
            <label>Team Member 5 Name:</label>
            <asp:TextBox ID="tm5" runat="server" CssClass="form-control" placeholder="Enter
your Team Member Name 5"></asp:TextBox>
            <asp:RequiredFieldValidator ID="RequiredFieldValidator7"
                runat="server" ErrorMessage="Please enter team member name 5"
                ControlToValidate="tm5"></asp:RequiredFieldValidator>
        </div>

        <div class="mb-3">
            <label>Team Member 6 Name:</label>
            <asp:TextBox ID="tm6" runat="server" CssClass="form-control" placeholder="Enter
your Team Member Name 6"></asp:TextBox>
            <asp:RequiredFieldValidator ID="RequiredFieldValidator8"
                runat="server" ErrorMessage="Please enter team member name 6"
                ControlToValidate="tm6"></asp:RequiredFieldValidator>
        </div>
        <div class="mb-3">
            <asp:Button ID="Button1" runat="server" Text="Register" CssClass="btn btn-primary"
PostBackUrl="~/Default2.aspx" />
        </div>
</asp:Content>
```

Default2.aspx:

```
<%@ Page Language="C#" AutoEventWireup="true"  MasterPageFile="~/Site.master"
CodeFile="Default2.aspx.cs" Inherits="Default2" %>

<asp:Content ID="HeaderContent" runat="server" ContentPlaceHolderID="HeadContent">
</asp:Content>
<asp:Content ID="BodyContent" runat="server" ContentPlaceHolderID="MainContent">
    <div>
 <asp:Label ID="Smsg" runat="server" Text="" CssClass="form-control"></asp:Label><br />
 <asp:Label ID="Semail" runat="server" Text="" CssClass="form-control"></asp:Label><br />
 <asp:Label ID="Stm" runat="server" Text="" CssClass="form-control"></asp:Label><br />
 <asp:Label ID="Scollege" runat="server" Text="" CssClass="form-control"></asp:Label><br
 />
 <asp:Label ID="Stm1" runat="server" Text="" CssClass="form-control"></asp:Label><br />
 <asp:Label ID="Stm2" runat="server" Text="" CssClass="form-control"></asp:Label><br />
 <asp:Label ID="Stm3" runat="server" Text="" CssClass="form-control"></asp:Label><br />
 <asp:Label ID="Stm4" runat="server" Text="" CssClass="form-control"></asp:Label><br />
 <asp:Label ID="Stm5" runat="server" Text="" CssClass="form-control"></asp:Label><br />
 <asp:Label ID="Stm6" runat="server" Text="" CssClass="form-control"></asp:Label><br />
 </div>
</asp:Content>
```

Default2.aspx.cs:

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

public partial class Default2 : System.Web.UI.Page
{
   protected void Page_Load(object sender, EventArgs e)
   {
      if (PreviousPage != null)
      {
         ContentPlaceHolder placeHolder =
(ContentPlaceHolder)PreviousPage.Master.FindControl("MainContent");
         TextBox email = (TextBox)placeHolder.FindControl("email");
         Smsg.Text = "<h3 style='color:green;'> You are successfully registered for SIH 2022
with following details: </h3>";
         Semail.Text = "<b>Email :- </b>" + email.Text + "</b>";
         DropDownList college = (DropDownList)placeHolder.FindControl("college");
         Scollege.Text = "<b>College :- </b>" + college.Text + "</b>";
         TextBox tm = (TextBox)placeHolder.FindControl("tmn");
         Stm.Text = "<b>Team Mentor: </b>" + tm.Text + "</b>";
         TextBox tm1 = (TextBox)placeHolder.FindControl("tm1");
         Stm1.Text = "<b>Team Member 1: </b>" + tm1.Text + "</b>";
         TextBox tm2 = (TextBox)placeHolder.FindControl("tm2");
         Stm2.Text = "<b>Team Member 2: </b>" + tm2.Text + "</b>";
         TextBox tm3 = (TextBox)placeHolder.FindControl("tm3");
         Stm3.Text = "<b>Team Member 3: </b>" + tm3.Text + "</b>";
         TextBox tm4 = (TextBox)placeHolder.FindControl("tm4");
         Stm4.Text = "<b>Team Member 4: </b>" + tm4.Text + "</b>";
         TextBox tm5 = (TextBox)placeHolder.FindControl("tm5");
         Stm5.Text = "<b>Team Member 5: </b>" + tm5.Text + "</b>";
         TextBox tm6 = (TextBox)placeHolder.FindControl("tm6");
         Stm6.Text = "<b>Team Member 6: </b>" + tm6.Text + "</b>";
      }
      }
   }
```

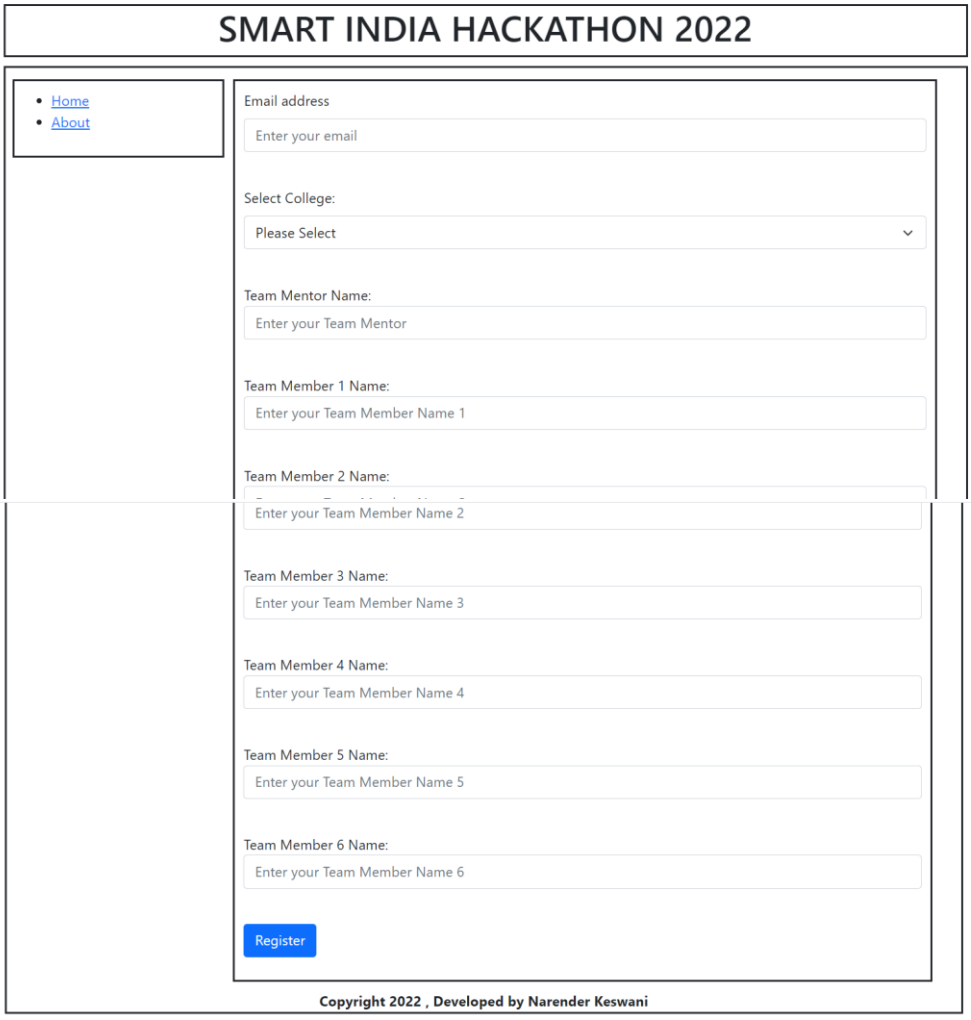
About.aspx:

```
<%@ Page Title="About Us" Language="C#" MasterPageFile="~/Site.master"
AutoEventWireup="true"
   CodeFile="About.aspx.cs" Inherits="About" %>

<asp:Content ID="HeaderContent" runat="server" ContentPlaceHolderID="HeadContent">
</asp:Content>
<asp:Content ID="BodyContent" runat="server" ContentPlaceHolderID="MainContent">
   <h2>
      About Us
   </h2>
```

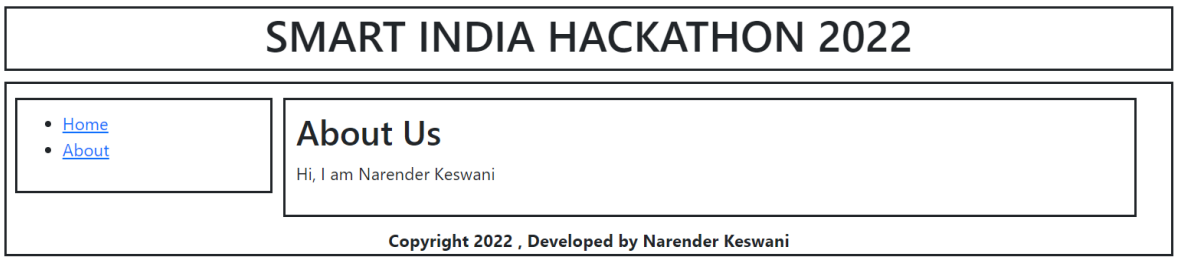
```
    <p>
     Hi,
       I am Narender Keswani
    </p>
</asp:Content>
```

OUTPUT:

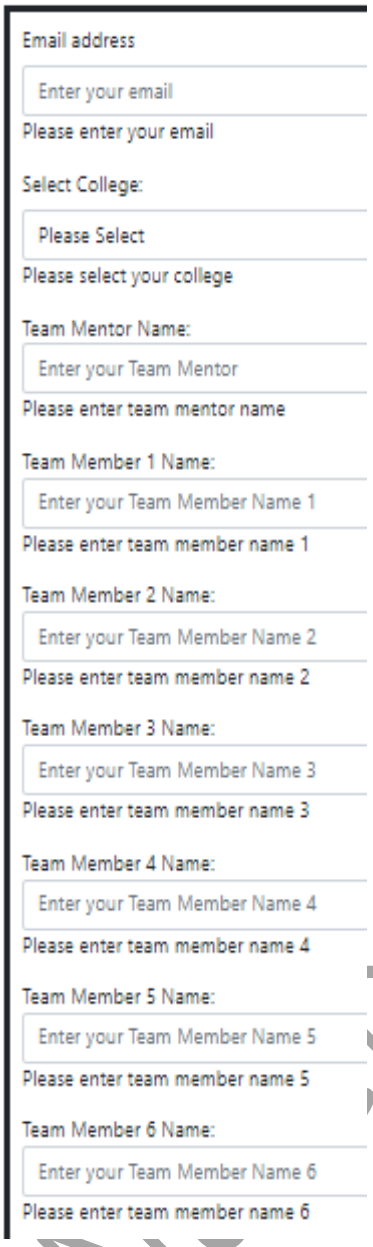
HOMEPAGE:



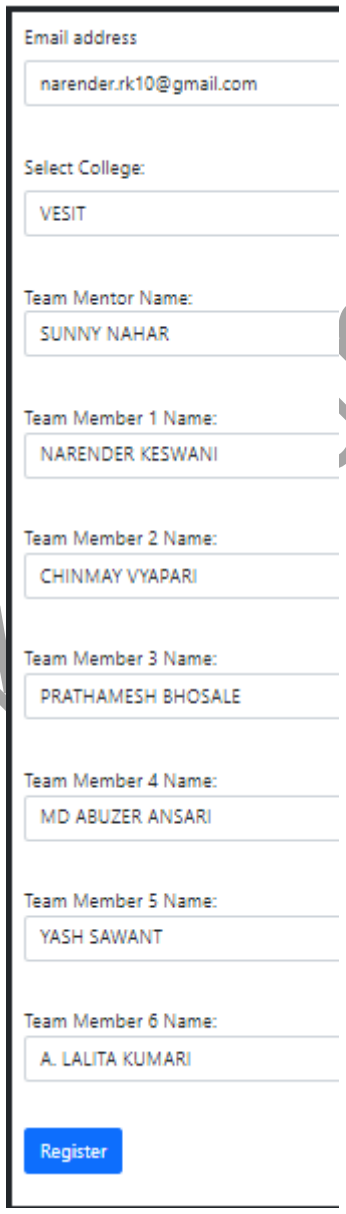
ABOUT US PAGE:



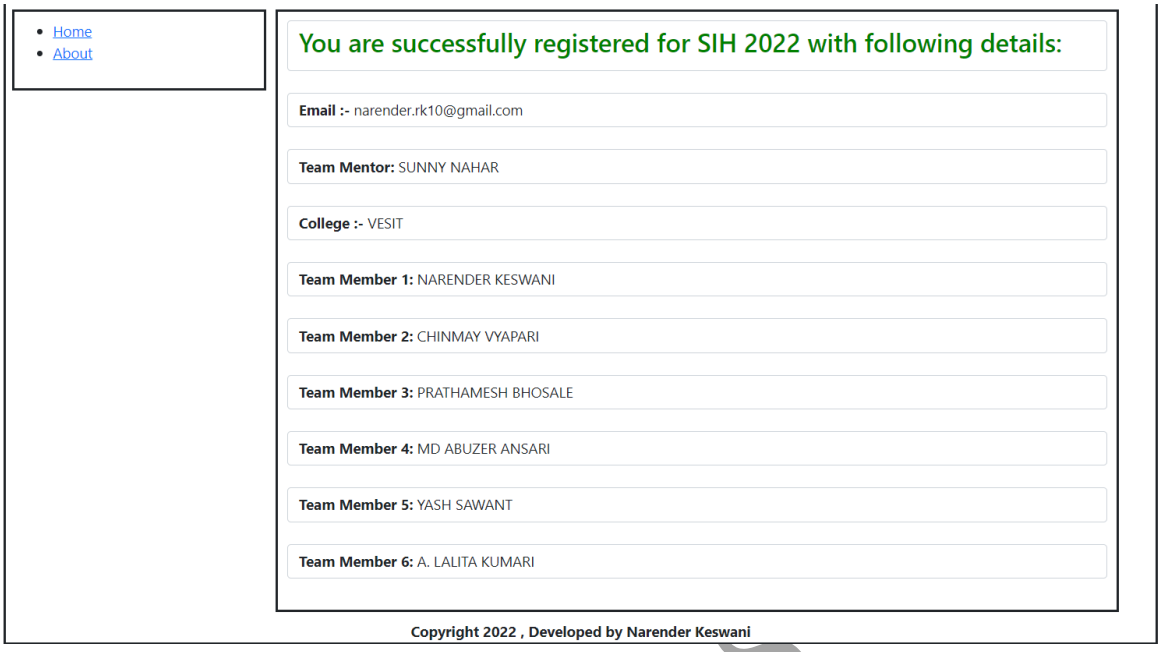
VALIDATION:



FORM FILLING:



Showing data on next page:



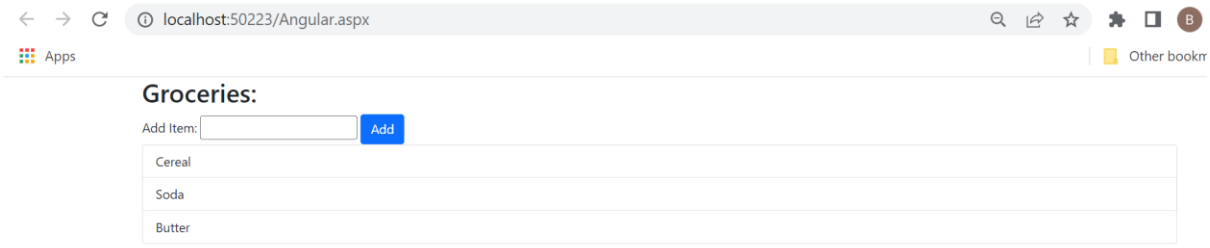
B) Build a simple angular web application.

SOURCE CODE:

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

<!DOCTYPE html>
<html>
<head>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/angular.js/1.8.2/angular.min.js"
    integrity="sha512-
7oYXeK0OxTFxndh0erL8FsjGvrl2VMDor6fVqzlLGfwOQQqTbYsGPv4ZZ15QHfSk80doyaM0ZJdv
kyDcVO7KFA=="
    crossorigin="anonymous" referrerpolicy="no-referrer"></script>
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"
    rel="stylesheet"
    integrity="sha384-
EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOmLASjC"
    crossorigin="anonymous">
  <script
    src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js"
    integrity="sha384-
MrcW6ZMFYlzcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/tWtIaxVXM"
    crossorigin="anonymous"></script>
</head>
<body ng-app="myApp">
  <div>
    <div class="container" ng-controller="groceryCtrl">
      <h2>Groceries:</h2>
      Add Item:
      <input type="text" ng-model="addGrocery" />
      <button class="btn btn-primary" ng-click="addItem()">Add</button>
```

```
        <ul class="list-group">
          <li class="list-group-item" ng-repeat="g in Groceries">{{ g }} </li>
        </ul>
      </div>
    </div>
    <script type="text/javascript">
      angular.module('myApp', [])
        .controller('groceryCtrl', function GroceryController($scope) {
          $scope.Groceries = [];
          $scope.addItem = function () {
            $scope.Groceries.push($scope.addGrocery);
            $scope.addGrocery = "";
          }
        });
    </script>
</body>
</html>
```

OUTPUT:

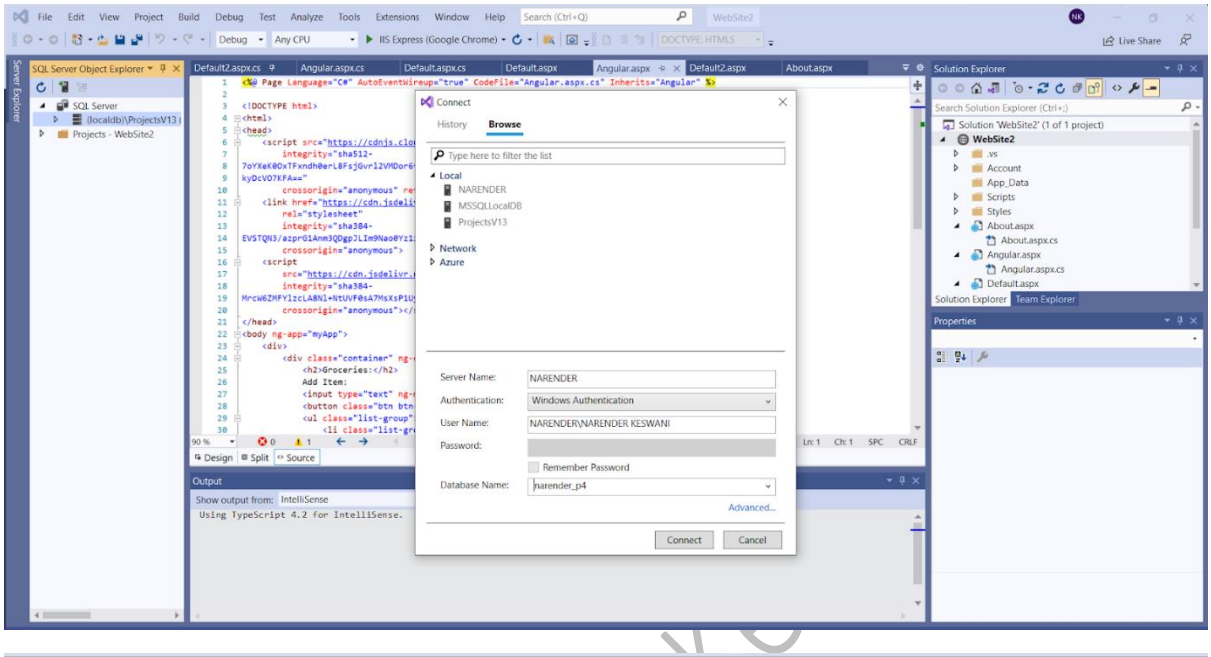


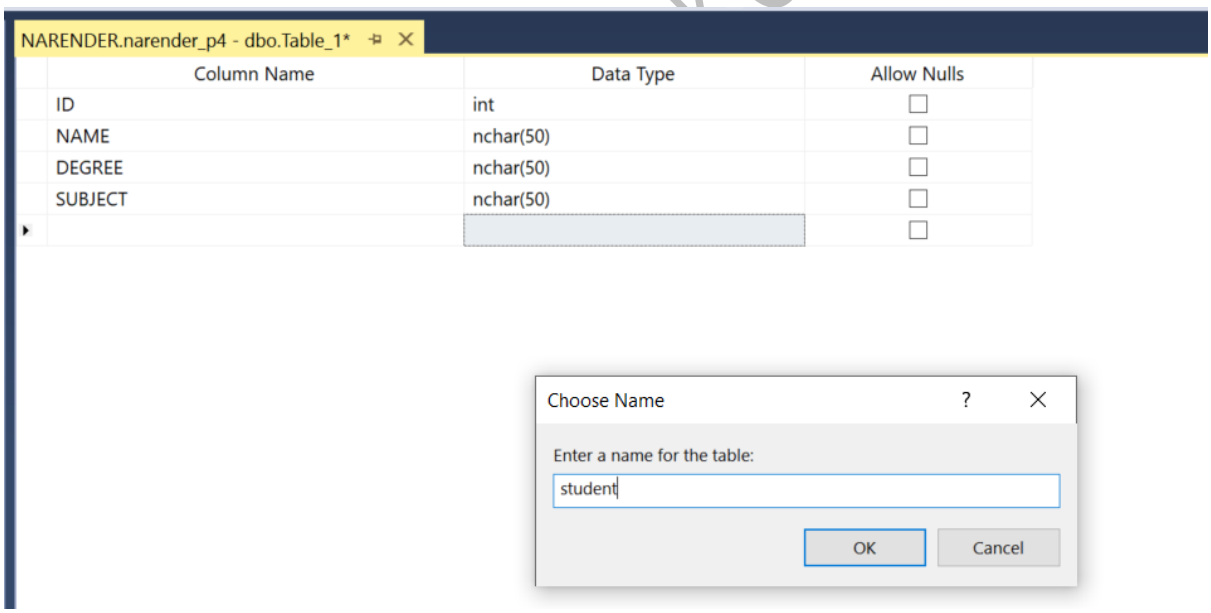
CONCLUSION:

From this practical, I have learned how to create web application in asp.net.

AIM: Design a web page to demonstrate a connection oriented architecture & disconnected architecture.

DB Config:





WebConfig:

```
<connectionStrings>
        <add name="ApplicationServices" connectionString="server=. ;
database=narender_p4; Trusted_Connection=Yes;" providerName="System.Data.SqlClient"/>
</connectionStrings>
```

A) Design a web page to demonstrate a connection oriented architecture. Fetch Student details from database such as Roll_no, Name, Program(eg. MCA), Course(eg. AWT), etc.

SOURCE CODE:

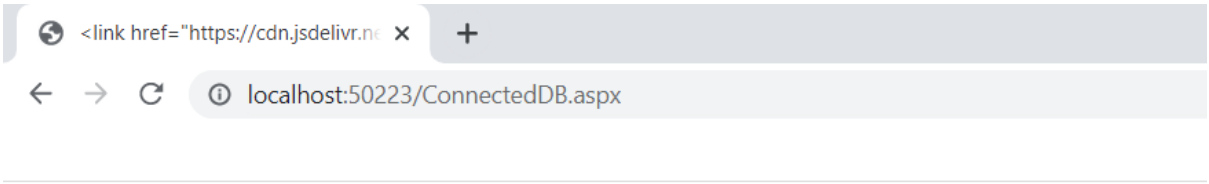
ConnectedDb.aspx:

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

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server"></head>
<body>
  <form id="form1" runat="server">
    <div class="container">
      <h2>Display Data using Connected Architecture</h2>
      <asp:GridView ID="GridView1" runat="server"></asp:GridView>
    </div>
  </form>
</body>
</html>
```

ConnectedDb.aspx.cs:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data;
using System.Data.SqlClient;
using System.Configuration;

public partial class ConnectedDB : System.Web.UI.Page
{
  protected void Page_Load(object sender, EventArgs e)
  {
    string connStr=
ConfigurationManager.ConnectionStrings["ApplicationServices"].ConnectionString;
    SqlConnection con = new SqlConnection(connStr);
    {
      SqlCommand query = new SqlCommand(" select * from student ", con);
      con.Open();
      SqlDataReader rdr = query.ExecuteReader();
      GridView1.DataSource = rdr;
      GridView1.DataBind();
    }
  }
}
```

OUTPUT:



Display Data using Connected Architecture

ID	NAME	DEGREE	SUBJECT
1	Narender	MCA	AWT
2	Neel	Bvoc	JS
3	Parnot	Btech	JAVA

B) Design a web page to demonstrate a disconnected architecture. Fetch Student details from database such as Roll_no, Name, Program(eg. MCA), Course(eg. AWT), etc.

SOURCE CODE:

UnConnectedDb.aspx:

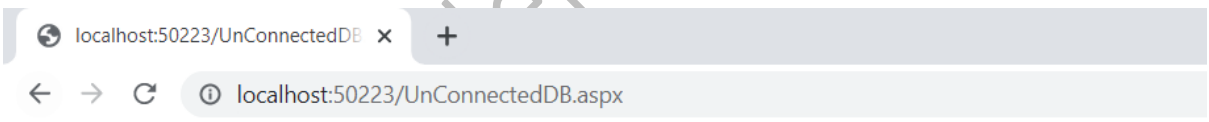
```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="UnConnectedDB.aspx.cs"
Inherits="UnConnectedDB" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
</head>
<body>
  <form id="form1" runat="server">
    <div class="container">
      <h2>Display Data using Disconnected Architecture</h2>
      <asp:GridView ID="GridView1" runat="server"></asp:GridView>
    </div>
  </form>
</body>
</html>
```

UnConnectedDb.aspx.cs:

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

```
using System.Web.UI.WebControls;
using System.Data;
using System.Data.SqlClient;
using System.Configuration;

public partial class UnConnectedDB : System.Web.UI.Page
{
   protected void Page_Load(object sender, EventArgs e)
   {
      string connStr =
ConfigurationManager.ConnectionStrings["ApplicationServices"].ConnectionString;
      SqlConnection conn = new SqlConnection(connStr);
      {
         string query = "Select * from Student";
         SqlDataAdapter da = new SqlDataAdapter(query, conn);
         DataSet ds = new DataSet();
         da.Fill(ds);
         GridView1.DataSource = ds;
         GridView1.DataBind();
      }
   }
}
```

OUTPUT:



Display Data using Disconnected Architecture

ID	NAME	DEGREE	SUBJECT
1	Narender	MCA	AWT
2	Neel	Bvoc	JS
3	Parnot	Btech	JAVA

CONCLUSION:

From this practical, I have learned how to connect asp.net with database, also learned about connected and disconnected architecture.

AIM: DEMONSTRATE USAGE OF DATA BOUND CONTROLS, SIMPLE STORED PROCEDURE AND PARAMETERIZED STORED PROCEDURES IN ASP.NET

WebConfig:

<add name="ApplicationServices" connectionString="server=. ; database=narender_p4; Trusted_Connection=Yes;" providerName="System.Data.SqlClient"/>

DATABASE:

ID	NAME	DEGREE	SUBJECT
1	Narender ...	MCA ...	AWT ...
2	Neel ...	Bvoc ...	JS ...
3	Parnot ...	Btech ...	JAVA ...

A) Create a web page that demonstrates the use of data bound controls of ASP.NET.

SOURCE CODE:

DataBound.aspx:

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

<!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>
  <h2>Display Data using Connected Architecture</h2>
  <asp:GridView ID="GridView1" runat="server" AutoGenerateColumns="False"
CellPadding="6" OnRowCancelingEdit="GridView1_RowCancelingEdit"
OnRowEditing="GridView1_RowEditing" OnRowUpdating="GridView1_RowUpdating">
      <Columns>
        <asp:TemplateField>
          <ItemTemplate>
            <asp:Button ID="btn_Edit" runat="server" Text="Edit" CommandName="Edit"
/>
          </ItemTemplate>
          <EditItemTemplate>
            <asp:Button ID="btn_Update" runat="server" Text="Update"
CommandName="Update"/>
            <asp:Button ID="btn_Cancel" runat="server" Text="Cancel"
CommandName="Cancel"/>
          </EditItemTemplate>
```

```
            </asp:TemplateField>
            <asp:TemplateField HeaderText="ID">
              <ItemTemplate>
                <asp:Label ID="lbl_ID" runat="server" Text='<%#Eval("ID") %>'></asp:Label>
              </ItemTemplate>
            </asp:TemplateField>
            <asp:TemplateField HeaderText="NAME">
              <ItemTemplate>
                <asp:Label ID="lbl_Name" runat="server" Text='<%#Eval("NAME")
%>'></asp:Label>
              </ItemTemplate>
              <EditItemTemplate>
                <asp:TextBox ID="txt_Name" runat="server" Text='<%#Eval("NAME")
%>'></asp:TextBox>
              </EditItemTemplate>
            </asp:TemplateField>
             <asp:TemplateField HeaderText="COURSE">
              <ItemTemplate>
                <asp:Label ID="lbl_Course" runat="server" Text='<%#Eval("DEGREE")
%>'></asp:Label>
              </ItemTemplate>
              <EditItemTemplate>
                <asp:TextBox ID="txt_Course" runat="server" Text='<%#Eval("DEGREE")
%>'></asp:TextBox>
              </EditItemTemplate>
            </asp:TemplateField>
            <asp:TemplateField HeaderText="SUBJECT">
              <ItemTemplate>
                <asp:Label ID="lbl_Subject" runat="server" Text='<%#Eval("SUBJECT")
%>'></asp:Label>
              </ItemTemplate>
              <EditItemTemplate>
                <asp:TextBox ID="txt_Subject" runat="server" Text='<%#Eval("SUBJECT")
%>'></asp:TextBox>
              </EditItemTemplate>
            </asp:TemplateField>
          </Columns>
          <HeaderStyle BackColor="#663300" ForeColor="#ffffff"/>
          <RowStyle BackColor="#e7ceb6"/>
        </asp:GridView>
    </div>
    </form>
</body>
</html>
```

DataBound.aspx.cs:

```
using System;
using System.Web.UI.WebControls;
using System.Data;
using System.Data.SqlClient;
using System.Configuration;
```

```
public partial class DataBound : System.Web.UI.Page
{
   //Connection String from web.config File
   string cs =
ConfigurationManager.ConnectionStrings["ApplicationServices"].ConnectionString;
   SqlConnection con;
   SqlDataAdapter adapt;
   DataTable dt;
   protected void Page_Load(object sender, EventArgs e)
   {
      if(!IsPostBack)
      {
         ShowData();
      }
   }
   //ShowData method for Displaying Data in Gridview
   protected void ShowData()
   {
      dt = new DataTable();
      con = new SqlConnection(cs);
      con.Open();
      adapt = new SqlDataAdapter("Select ID,NAME,DEGREE,SUBJECT from student",con);
      adapt.Fill(dt);
      if(dt.Rows.Count>0)
      {
         GridView1.DataSource = dt;
         GridView1.DataBind();
      }
      con.Close();
   }

   protected void GridView1_RowEditing(object sender,
System.Web.UI.WebControls.GridViewEditEventArgs e)
   {
      //NewEditIndex property used to determine the index of the row being edited.
      GridView1.EditIndex = e.NewEditIndex;
      ShowData();
   }
   protected void GridView1_RowUpdating(object sender,
System.Web.UI.WebControls.GridViewUpdateEventArgs e)
   {
      //Finding the controls from Gridview for the row which is going to update
      Label id=GridView1.Rows[e.RowIndex].FindControl("lbl_ID") as Label;
      TextBox name = GridView1.Rows[e.RowIndex].FindControl("txt_NAME") as TextBox;
      TextBox course = GridView1.Rows[e.RowIndex].FindControl("txt_COURSE") as TextBox;
      TextBox subject = GridView1.Rows[e.RowIndex].FindControl("txt_SUBJECT") as TextBox;
      con = new SqlConnection(cs);
      con.Open();
      //updating the record
```

```
        SqlCommand cmd = new SqlCommand("Update student set
NAME='"+name.Text+"',DEGREE='"+course.Text+"',SUBJECT='"+subject.Text+"' where
ID="+Convert.ToInt32(id.Text),con);
        cmd.ExecuteNonQuery();
        con.Close();
        //Setting the EditIndex property to -1 to cancel the Edit mode in Gridview
        GridView1.EditIndex = -1;
        //Call ShowData method for displaying updated data
        ShowData();
    }
    protected void GridView1_RowCancelingEdit(object sender,
System.Web.UI.WebControls.GridViewCancelEditEventArgs e)
    {
        //Setting the EditIndex property to -1 to cancel the Edit mode in Gridview
        GridView1.EditIndex = -1;
        ShowData();
    }
}
```

OUTPUT:

BEFORE EDITING:

Display Data using Connected Architecture

	ID	NAME	COURSE	SUBJECT
Edit	1	Narender	MCA	AWT
Edit	2	Neel	Bvoc	JS
Edit	3	Parnot	Btech	JAVA

EDITING:

Display Data using Connected Architecture

	ID	NAME	COURSE	SUBJECT
Edit	1	Narender	MCA	AWT
Update Cancel	2	Neel Deshmukh	Bvoc SD	JS
Edit	3	Parnot	Btech	JAVA

AFTER EDITING:

Display Data using Connected Architecture

	ID	NAME	COURSE	SUBJECT
Edit	1	Narender	MCA	AWT
Edit	2	Neel Deshmukh	Bvoc SD	JS
Edit	3	Parnot	Btech	JAVA

B) Design a web page to demonstrate the working of a simple stored procedure.

DATABASE:

```
CREATE OR ALTER PROCEDURE simpleStudentProcedureByNarender
AS
select * from narender_p4.dbo.student
GO;
```

SOURCE CODE:

StoredProcedure.aspx:

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

<!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:GridView ID="GridView1" runat="server">
      </asp:GridView>
   </div>
   </form>
</body>
</html>
```

StoredProcedure.aspx.cs:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data;
using System.Data.SqlClient;
using System.Configuration;

public partial class StoredProcedure : System.Web.UI.Page
{
   string cs =
ConfigurationManager.ConnectionStrings["ApplicationServices"].ConnectionString;
   SqlConnection con;
   protected void Page_Load(object sender, EventArgs e)
   {
      con = new SqlConnection(cs);
      con.Open();

      System.Data.SqlClient.SqlCommand objCmd = new
System.Data.SqlClient.SqlCommand("simpleStudentProcedureByNarender", con);

   objCmd.CommandType = System.Data.CommandType.StoredProcedure;

   GridView1.DataSource = objCmd.ExecuteReader();

   GridView1.DataBind();

   con.Close();
   }
}
```

OUTPUT:

ID	NAME	DEGREE	SUBJECT
1	Narender	MCA	AWT
2	Neel Deshmukh	Bvoc SD	JS
3	Parnot	Btech	JAVA

C) Design a web page to demonstrate the working of parameterized stored procedure.

DATABASE:

```
CREATE OR ALTER PROCEDURE paramStudentProcedureByNarender(
@ID INT, @NAME NVARCHAR(50), @DEGREE NVARCHAR(50), @SUBJECT NVARCHAR(50) )
AS
BEGIN
INSERT INTO student( [ID], [NAME], [DEGREE], [SUBJECT] )
VALUES ( @ID, @NAME, @DEGREE, @SUBJECT )
END
```

SOURCE CODE:

ParameterSP.aspx:

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


<!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="ID"></asp:Label>

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

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

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

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

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

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

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

        <asp:Button ID="Button1" runat="server" Text="Submit Record"
OnClick="Button1_Click" />
```

```
            <asp:Label ID="lblResult" runat="server" Text=""></asp:Label>
        </div>
        </form>
    </body>
    </html>
```

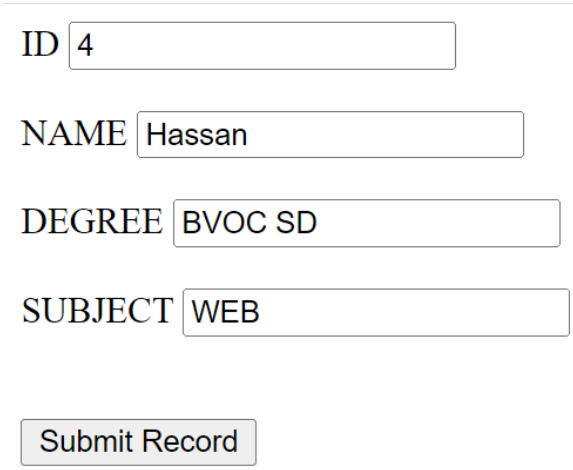
ParameterSP.aspx.cs:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data;
using System.Data.SqlClient;
using System.Configuration;

public partial class ParameterSP : System.Web.UI.Page
{
    string cs = ConfigurationManager.ConnectionStrings["ApplicationServices"].ConnectionString;
    SqlConnection con;
    SqlCommand cmd = new SqlCommand();
    SqlParameter sp1, sp2, sp3, sp4;
    protected void Page_Load(object sender, EventArgs e)
    {

    }
    protected void Button1_Click(object sender, EventArgs e)
    {
        con = new SqlConnection(cs);
        cmd = new SqlCommand("paramStudentProcedureByNarender", con);
        cmd.CommandType = CommandType.StoredProcedure;
        sp1 = new SqlParameter("@ID", SqlDbType.Int);
        sp2 = new SqlParameter("@NAME", SqlDbType.NVarChar);
        sp3 = new SqlParameter("@DEGREE", SqlDbType.NVarChar);
        sp4 = new SqlParameter("@SUBJECT", SqlDbType.NVarChar);
        sp1.Value = Convert.ToInt32(TextBox1.Text);
        sp2.Value = TextBox2.Text;
        sp3.Value = TextBox3.Text;
        sp4.Value = TextBox4.Text;
        cmd.Parameters.Add(sp1);
        cmd.Parameters.Add(sp2);
        cmd.Parameters.Add(sp3);
        cmd.Parameters.Add(sp4);
        con.Open();
        try
```

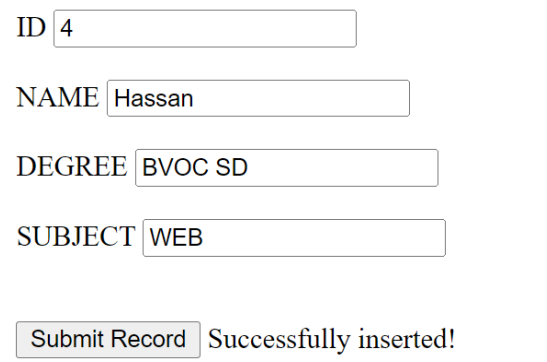
```
        {
          cmd.ExecuteNonQuery();
          lblResult.Text = "Successfully inserted!";
        }
        catch (Exception)
        {
          lblResult.Text = "Error was occured!";
        }
        con.Close();
    }
}
```

OUTPUT:

Before Inserting:



After Inserting:



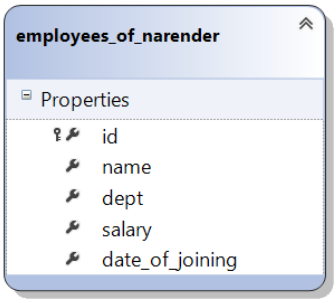
CONCLUSION:

From this practical, I have learned about data bound controls, simple stored procedure and parameterized stored procedures in ASP.NET.

AIM: DESIGN A WEBPAGE TO DISPLAY THE USE OF LINQ.

A) Design a web page to display the employee information from table to grid control. Use LINQ to SQL Build websites to demonstrate the working of entity frameworks in dot net.

SOURCE CODE:



ReadLinqToSql.aspx:

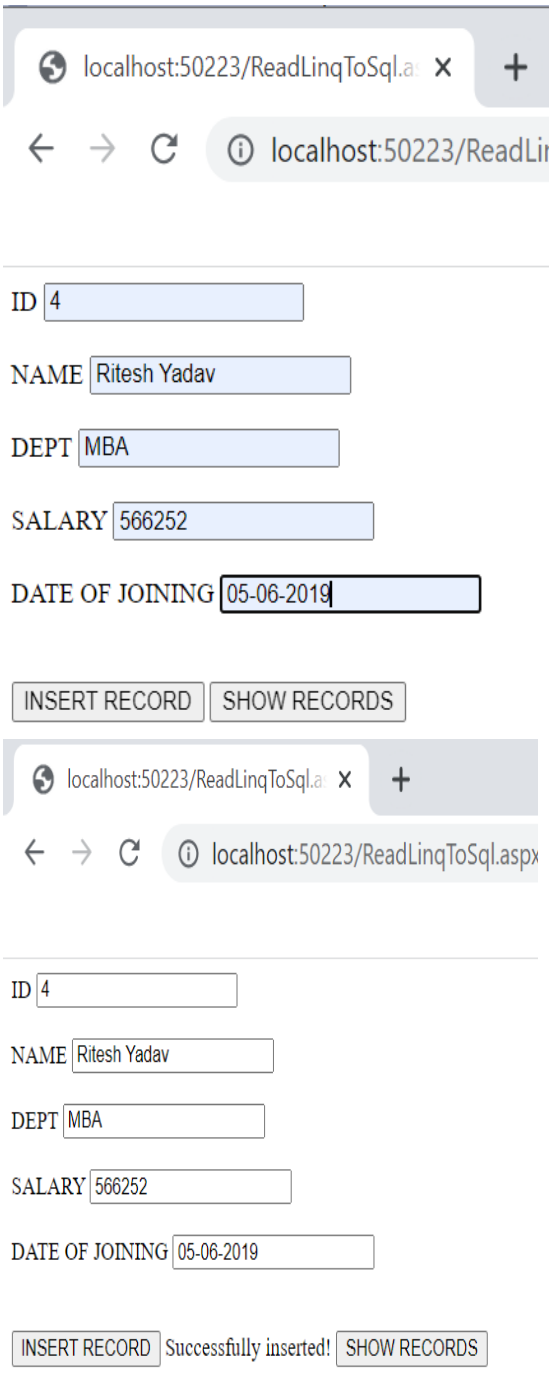
```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="ReadLinqToSql.aspx.cs"
Inherits="ReadLinqToSql" %>

<!DOCTYPE html>

<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="ID"></asp:Label>
 <asp:TextBox ID="TextBox1" runat="server"></asp:TextBox><br /><br />
 <asp:Label ID="Label2" runat="server" Text="NAME"></asp:Label>
 <asp:TextBox ID="TextBox2" runat="server"></asp:TextBox><br /><br />
 <asp:Label ID="Label3" runat="server" Text="DEPT"></asp:Label>
 <asp:TextBox ID="TextBox3" runat="server"></asp:TextBox><br /><br />
 <asp:Label ID="Label4" runat="server" Text="SALARY"></asp:Label>
 <asp:TextBox ID="TextBox4" runat="server"></asp:TextBox><br /><br />
 <asp:Label ID="Label5" runat="server" Text="DATE OF JOINING"></asp:Label>
 <asp:TextBox ID="TextBox5" runat="server"></asp:TextBox><br /><br /><br />
 <asp:Button ID="Button1" runat="server" Text="INSERT RECORD"
OnClick="Button1_Click" />
      <asp:Label ID="lblResult" runat="server" Text=""></asp:Label>
      <asp:Button ID="Button2" runat="server" Text="SHOW RECORDS"
OnClick="Button2_Click" />
```

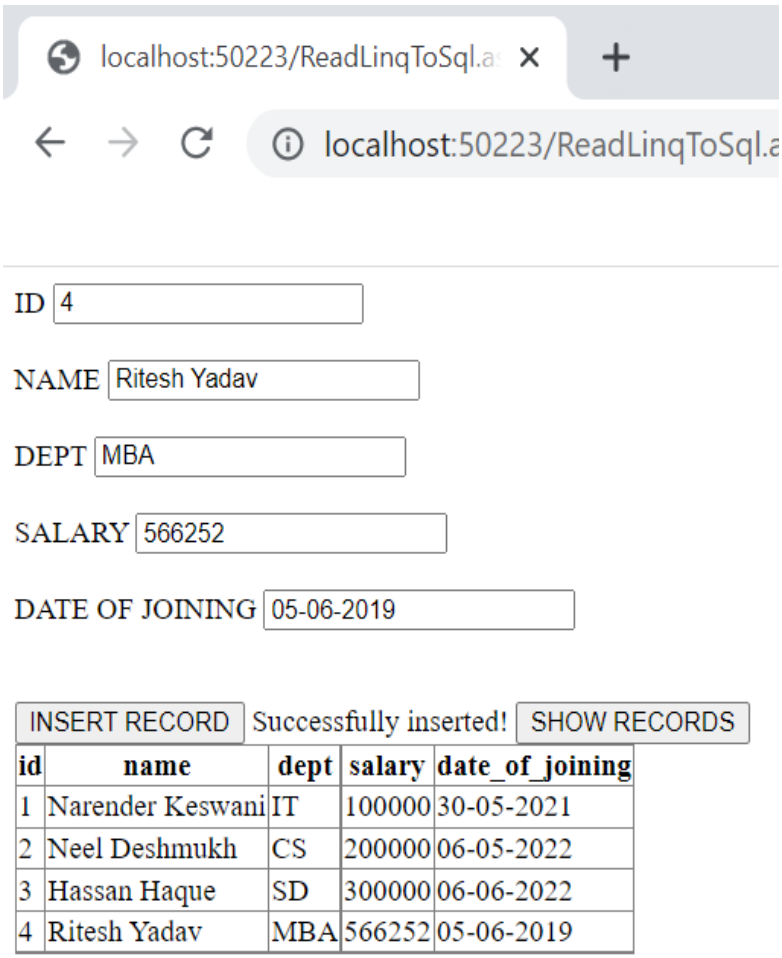
```
            <asp:GridView ID="GridView1" runat="server">
            </asp:GridView>
        </div>
    </form>
</body>
</html>
```

ReadLinqToSql.aspx.cs:

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

public partial class ReadLinqToSql : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {

    }

    protected void Button1_Click(object sender, EventArgs e)
    {
        using (DataClassesDataContext dc = new DataClassesDataContext())
        {
            employees_of_narender en = new employees_of_narender
            {
                id = Convert.ToInt32(TextBox1.Text),
                name = TextBox2.Text,
                dept = TextBox3.Text,
                salary = Convert.ToInt32(TextBox4.Text),
                date_of_joining = TextBox5.Text,
            };
            dc.employees_of_narenders.InsertOnSubmit(en);
            dc.SubmitChanges();
            lblResult.Text = "Successfully inserted!";
        }
    }

    protected void Button2_Click(object sender, EventArgs e)
    {
        DataClassesDataContext dc = new DataClassesDataContext();
        GridView1.DataSource = (from a in dc.employees_of_narenders select a);
        GridView1.DataBind();
    }
}
```

OUTPUT:

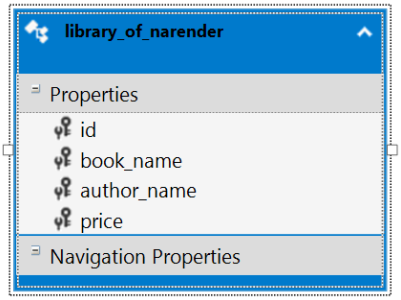

localhost:50223/ReadLinqToSql.a
localhost:50223/ReadLi
ID 4
NAME Ritesh Yadav
DEPT MBA
SALARY 566252
DATE OF JOINING 05-06-2019
INSERT RECORD
SHOW RECORDS
localhost:50223/ReadLinqToSql.a
localhost:50223/ReadLinqToSql.aspx
ID 4
NAME Ritesh Yadav
DEPT MBA
SALARY 566252
DATE OF JOINING 05-06-2019
INSERT RECORD
Successfully inserted!
SHOW RECORDS



id	name	dept	salary	date_of_joining
1	Narender Keswani	IT	100000	30-05-2021
2	Neel Deshmukh	CS	200000	06-05-2022
3	Hassan Haque	SD	300000	06-06-2022
4	Ritesh Yadav	MBA	566252	05-06-2019

B) Design a library system in ASP.NET and show all the book details in a Gridview dynamically using ADO.NET Entity Framework.

SOURCE CODE:



Default3.aspx:

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Default3.aspx.cs"
Inherits="Default3" %>

<!DOCTYPE html>
```

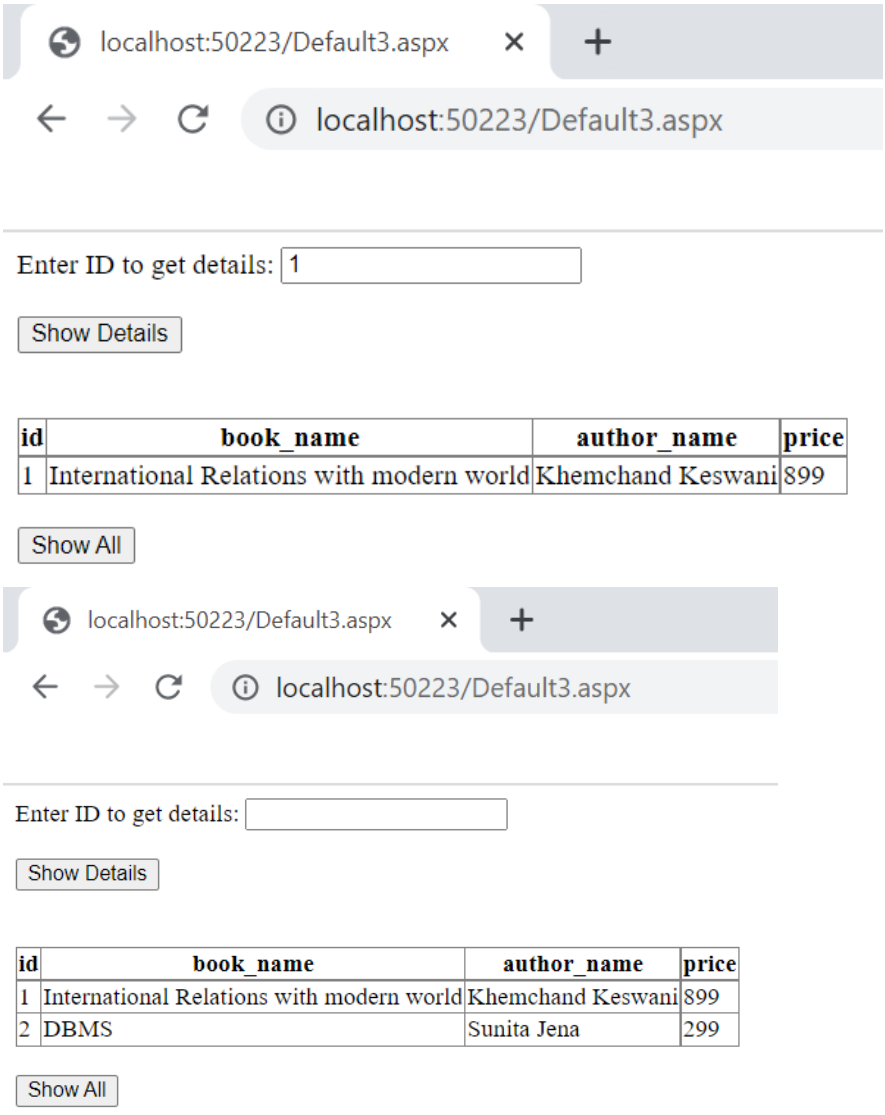
```
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
   <title></title>
</head>
<body>
   <form id="form1" runat="server">
<div>
 Enter ID to get details:
   <asp:TextBox ID="txtSrNo"
runat="server"></asp:TextBox>
 <br /><br />
 <asp:Button ID="cmdShow" runat="server" Text="Show Details"
OnClick="cmdShow_Click" />
 </div><br /><br />
 <div>
 <asp:GridView ID="GridView1" runat="server"></asp:GridView>
 <br />
 <asp:Button ID="cmdShowAll" runat="server" Text="Show All"
OnClick="cmdShowAll_Click" />
 </div>
   </form>
</body>
</html>
```

Default3.aspx.cs:

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

public partial class Default3 : System.Web.UI.Page
{
   protected void Page_Load(object sender, EventArgs e)
   {

   }

   protected void cmdShow_Click(object sender, EventArgs e)
   {
      int srno = Convert.ToInt32(txtSrNo.Text);
      narender_p7Entities ctx = new narender_p7Entities();
      GridView1.DataSource = (from b in ctx.library_of_narender
                  where b.id == srno
                  select b).ToList();
      GridView1.DataBind();
   }
```

```
    protected void cmdShowAll_Click(object sender, EventArgs e)
    {
      narender_p7Entities ctx = new narender_p7Entities();
      GridView1.DataSource = (from a in ctx.library_of_narender select a).ToList();
      GridView1.DataBind();
    }

}
```

OUTPUT:



CONCLUSION:

From this practical, I have learned & implemented the linq to sql, ADO.NET in ASP.NET.

AIM:Design Web Applications using Client Side Session Management.

WAP to implement Client side state management techniques and Server side state management techniques on the form design attached with the assignment.

A) SESSION MANAGEMENT:

SOURCE CODE:

SM.aspx:

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="SM.aspx.cs" Inherits="SM" %>
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
   <title></title>
</head>
<body>
   <form id="form1" runat="server">
      <div>
      Name: <asp:TextBox ID="name" runat="server"></asp:TextBox>
      <br />
      Age: <asp:TextBox ID="age" runat="server"></asp:TextBox>
      <br />
      Branch: <asp:TextBox ID="branch" runat="server"></asp:TextBox>
      <br />
      <asp:Button ID="Button1" runat="server" OnClick="Button1_Click" Text="Submit" />
   </div>
   </form>
</body>
</html>
```

SM.aspx.cs:

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

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

```
    }

    protected void Button1_Click(object sender, EventArgs e)
    {
        Session["name"] = name.Text;
        Session["age"] = age.Text;
        Session["branch"] = branch.Text;
        Response.Redirect("SM2.aspx");

    }

}
```

SM2.aspx:

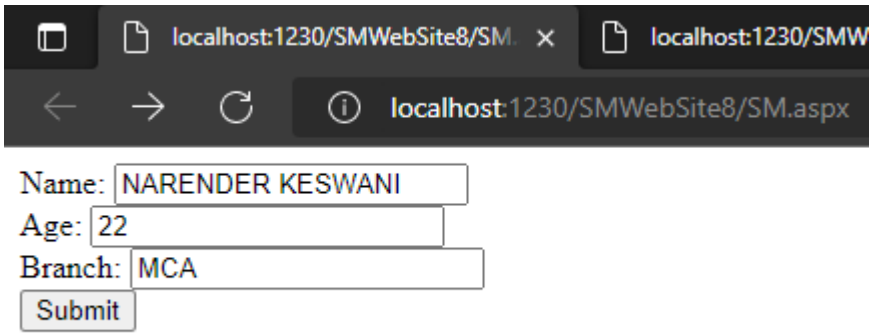
```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="SM2.aspx.cs" Inherits="SM2" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
</head>
<body>
    <form id="form1" runat="server">
   <div>
        Name: <asp:Label ID="name" runat="server"></asp:Label>
        <br />
        Age: <asp:Label ID="age" runat="server"></asp:Label>
        <br />
        Branch: <asp:Label ID="branch" runat="server"></asp:Label>
        <br />
    </div>
    </form>
</body>
</html>
```

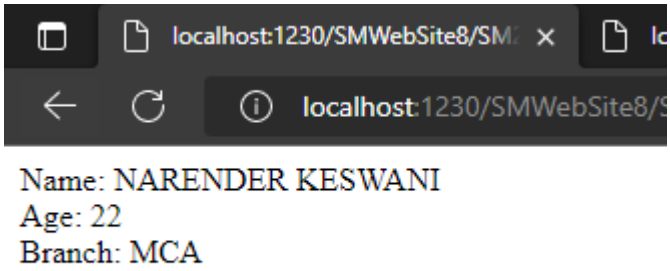
SM2.aspx.cs:

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

public partial class SM2 : System.Web.UI.Page
{
```

```
    protected void Page_Load(object sender, EventArgs e)
    {
        if (Session["name"] != null)
        {
            name.Text = Session["name"].ToString();
        }
        if (Session["age"] != null)
        {
            age.Text = Session["age"].ToString();
        }
        if (Session["branch"] != null)
        {
            branch.Text = Session["branch"].ToString();
        }

    }
}
```

OUTPUT:





B) COOKIE:

SOURCE CODE:

Cookie.aspx:

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

<!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>
      Name: <asp:TextBox ID="name" runat="server"></asp:TextBox>
      <br />
      Age: <asp:TextBox ID="age" runat="server"></asp:TextBox>
      <br />
      Branch: <asp:TextBox ID="branch" runat="server"></asp:TextBox>
      <br />
      <asp:Button ID="Button1" runat="server" OnClick="Button1_Click" Text="Submit" />
   </div>
   </form>
</body>
</html>
```

Cookie.aspx.cs:

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

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

```
    }

     protected void Button1_Click(object sender, EventArgs e)
    {
        HttpCookie userInfo = new HttpCookie("userInfo");
        userInfo["name"] = name.Text;
        userInfo["age"] = age.Text;
        userInfo["branch"] = branch.Text;
        Response.Cookies.Add(userInfo);
        Response.Redirect("C2.aspx");
    }


}
```

C2.aspx:

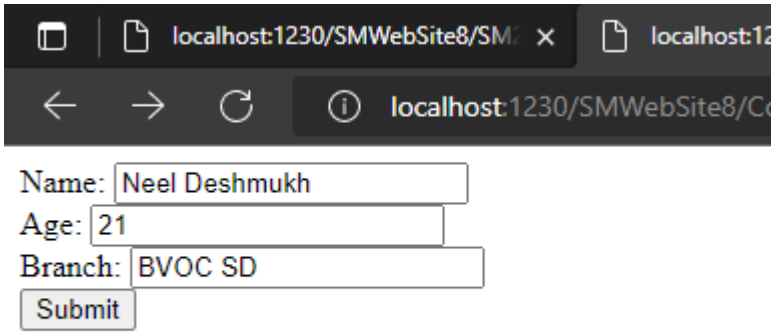
```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="C2.aspx.cs" Inherits="C2" %>

<!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>
        Name: <asp:Label ID="name" runat="server"></asp:Label>
        <br />
        Age: <asp:Label ID="age" runat="server"></asp:Label>
        <br />
        Branch: <asp:Label ID="branch" runat="server"></asp:Label>
        <br />
    </div>
    </form>
</body>
</html>
```

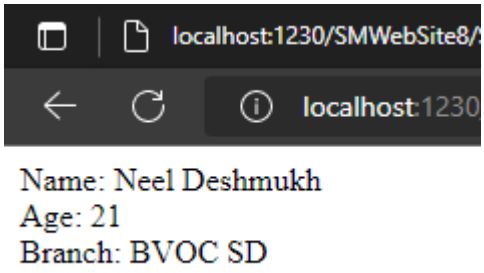
C2.aspx.cs:

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

```
using System.Web.UI;
using System.Web.UI.WebControls;

public partial class C2 : System.Web.UI.Page
{
   protected void Page_Load(object sender, EventArgs e)
   {
      HttpCookie reqCookies = Request.Cookies["userInfo"];
      if (reqCookies != null)
      {
         name.Text = reqCookies["name"].ToString();
         age.Text = reqCookies["age"].ToString();
         branch.Text = reqCookies["branch"].ToString();
      }
   }
}
```

OUTPUT:





C) QUERY STRINGS:

SOURCE CODE:

QueryString.aspx:

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="QueryString.aspx.cs"
Inherits="QueryString" %>
<!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>
      Name: <asp:TextBox ID="name" runat="server"></asp:TextBox>
      <br />
      Age: <asp:TextBox ID="age" runat="server"></asp:TextBox>
      <br />
      Branch: <asp:TextBox ID="branch" runat="server"></asp:TextBox>
      <br />
      <asp:Button ID="Button1" runat="server" OnClick="Button1_Click" Text="Submit" />
   </div>
   </form>
</body>
</html>
```

QueryString.aspx.cs:

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

public partial class QueryString : System.Web.UI.Page
{
   protected void Page_Load(object sender, EventArgs e)
   {

   }
```

```
    protected void Button1_Click(object sender, EventArgs e)
    {
        Response.Redirect("QS2.aspx?name=" + name.Text + "&age=" + age.Text + "&branch=" +
branch.Text);
    }
}
```

QS2.aspx:

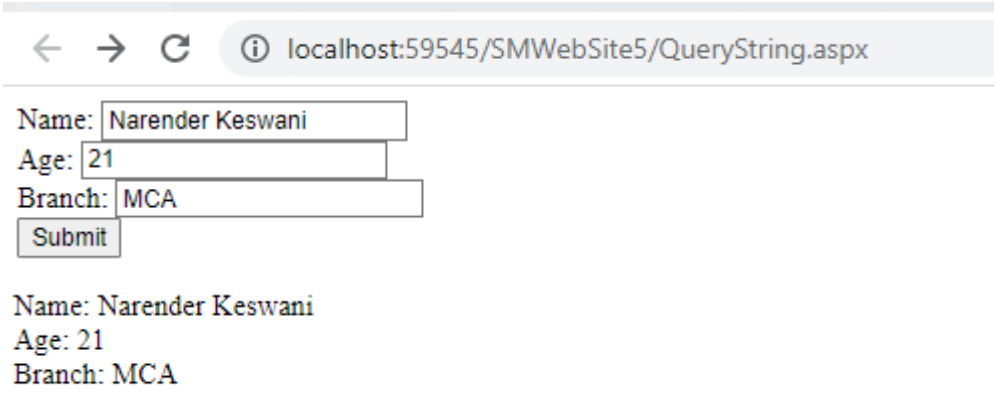
```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="QS2.aspx.cs" Inherits="QS2" %>

<!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>
        Name: <asp:Label ID="name" runat="server"></asp:Label>
        <br />
        Age: <asp:Label ID="age" runat="server"></asp:Label>
        <br />
        Branch: <asp:Label ID="branch" runat="server"></asp:Label>
        <br />
    </div>
    </form>
</body>
</html>
```

QS2.aspx.cs:

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

public partial class QS2 : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        string names = Request.QueryString["name"];
        string ages = Request.QueryString["age"];
        string branchs = Request.QueryString["branch"];
```

```
        name.Text = names;
        age.Text = ages;
        branch.Text = branchs;
    }
}
```

OUTPUT:



D) ViewState:

SOURCE CODE:

ViewState.aspx:

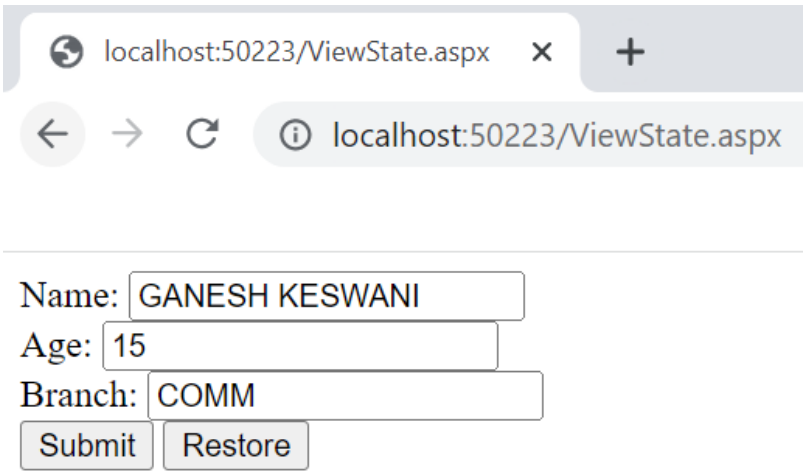
```
<%@ Page Language="C#" AutoEventWireup="true" EnableViewState="true"
EnableViewStateMac="true" CodeFile="ViewState.aspx.cs" Inherits="ViewState" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
</head>
<body>
    <form id="form1" runat="server">
      <div>
        Name: <asp:TextBox ID="name" runat="server"></asp:TextBox>
        <br />
        Age: <asp:TextBox ID="age" runat="server"></asp:TextBox>
        <br />
        Branch: <asp:TextBox ID="branch" runat="server"></asp:TextBox>
        <br />
```

```
        <asp:Button ID="Button1" runat="server" OnClick="Button1_Click" Text="Submit" />
        <asp:Button ID="Button2" runat="server" OnClick="Button2_Click" Text="Restore"/>
    </div>
    </form>
</body>
</html>
```

ViewState.aspx.cs:

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

public partial class ViewState : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {

    }
    protected void Button1_Click(object sender, EventArgs e)
    {
        ViewState["name"] = name.Text;
        name.Text = "";
        ViewState["age"] = age.Text;
        age.Text = "";
        ViewState["branch"] = branch.Text;
        branch.Text = "";
    }

    protected void Button2_Click(object sender, EventArgs e)
    {
        name.Text = ViewState["name"].ToString();
        age.Text = ViewState["age"].ToString();
        branch.Text = ViewState["branch"].ToString();
    }
}
```

OUTPUT:



E) HiddenFields:

SOURCE CODE:

HF.aspx:

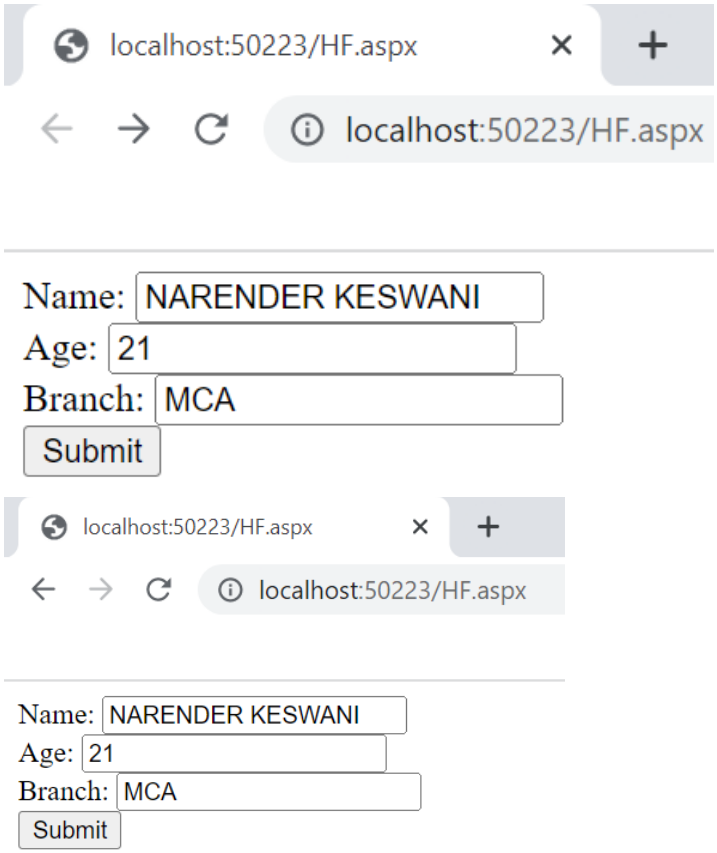
```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="HF.aspx.cs" Inherits="HF" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
   <title></title>
</head>
<body>
   <form id="form1" runat="server">
    <div>
      Name: <asp:TextBox ID="name" runat="server"></asp:TextBox>
      <br />
      Age: <asp:TextBox ID="age" runat="server"></asp:TextBox>
      <br />
      Branch: <asp:TextBox ID="branch" runat="server"></asp:TextBox>
      <br />
      <asp:Button ID="Button1" runat="server" OnClick="Button1_Click" Text="Submit" /> <br />
      <asp:HiddenField ID="HiddenField1" runat="server"></asp:HiddenField>
      <br />
      <asp:HiddenField ID="HiddenField2" runat="server"></asp:HiddenField>
      <br />
      <asp:HiddenField ID="HiddenField3" runat="server"></asp:HiddenField>
```

```
        <br />
         <asp:Label ID="Label1" runat="server"></asp:Label>
        <br />
        <asp:Label ID="Label2" runat="server"></asp:Label>
        <br />
        <asp:Label ID="Label3" runat="server"></asp:Label>
        <br />
    </div>
    </form>
</body>
</html>
```

HF.aspx.cs:

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

public partial class HF : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {

    }

    protected void Button1_Click(object sender, EventArgs e)
    {

        HiddenField1.Value = name.Text;
        HiddenField2.Value = age.Text;
        HiddenField3.Value = branch.Text;

        Label1.Text = HiddenField1.Value.ToString();
        Label2.Text = HiddenField2.Value.ToString();
        Label3.Text = HiddenField3.Value.ToString();

    }

}
```

OUTPUT:



F) ApplicationState:

SOURCE CODE:

ApplicationState.aspx:

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

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
   <title></title>
</head>
```

```
<body>
   <form id="form1" runat="server">
    <div>
      Name: <asp:TextBox ID="name" runat="server"></asp:TextBox>
      <br />
      Age: <asp:TextBox ID="age" runat="server"></asp:TextBox>
      <br />
      Branch: <asp:TextBox ID="branch" runat="server"></asp:TextBox>
      <br />
      <asp:Button ID="Button1" runat="server" OnClick="Button1_Click" Text="Submit" />
   </div>
   </form>
</body>
</html>
```

ApplicationState.aspx.cs:

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

public partial class ApplicationState : System.Web.UI.Page
{
   protected void Page_Load(object sender, EventArgs e)
   {

   }
   protected void Button1_Click(object sender, EventArgs e)
   {

      Application["name"] = name.Text;
      Application["age"] = age.Text;
      Application["branch"] = branch.Text;
      Response.Redirect("AS2.aspx");
   }
}
```

AS2.aspx:

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="AS2.aspx.cs" Inherits="AS2" %>

<!DOCTYPE html>
```

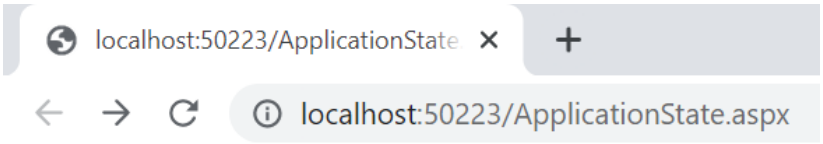
```
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
   <title></title>
</head>
<body>
   <form id="form1" runat="server">
        <div>
      Name: <asp:Label ID="name" runat="server"></asp:Label>
      <br />
      Age: <asp:Label ID="age" runat="server"></asp:Label>
      <br />
      Branch: <asp:Label ID="branch" runat="server"></asp:Label>
      <br />
   </div>

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

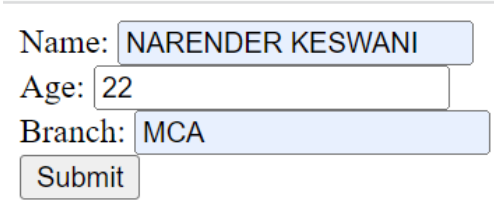
AS2.aspx.cs:

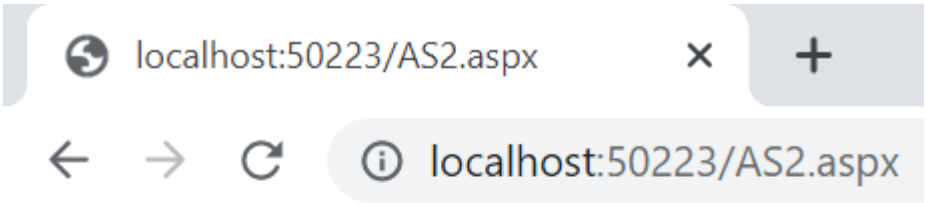
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

public partial class AS2 : System.Web.UI.Page
{
   protected void Page_Load(object sender, EventArgs e)
   {
      if (Application["name"] != null)
      {
         name.Text = Application["name"].ToString();
      }
      if (Application["age"] != null)
      {
         age.Text = Application["age"].ToString();
      }
      if (Application["branch"] != null)
```

```
        {
            branch.Text = Application["branch"].ToString();
        }
    }
}
```

OUTPUT:







CONCLUSION:

From this practical, I have learned and implemented client & server-side session management in asp.net.

AIM: DESIGN WEB APPLICATION TO PRODUCE AND CONSUME A WEB SERVICE & WCF SERVICE

Design Web Application to produce and Consume a web Service

A) Create an XML web service that returns all the student details from the student table. Write an Application that uses this service to display student details in datagrid view control.

SOURCE CODE:

StudentService.aspx:

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="StudentService.aspx.cs"
Inherits="StudentService" %>
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
  <title></title>
</head>
<body>
  <form id="form1" runat="server">
    <div>
      <asp:GridView ID="GridView1" runat="server"></asp:GridView>
    </div>
  </form>
</body>
</html>
```

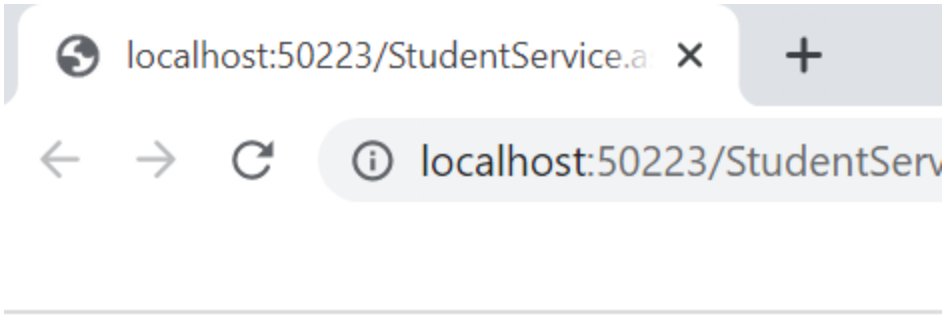
StudentService.aspx.cs:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data;
using System.Data.SqlClient;
using System.Configuration;
public partial class StudentService : System.Web.UI.Page
{
  protected void Page_Load(object sender, EventArgs e)
  {
    WebService1 ws1 = new WebService1();
    DataSet ds = ws1.GetData();
    GridView1.DataSource = ds;
    GridView1.DataBind();
  }
}
```

WebService1.asmx.cs:

```
using System;
using System.Collections.Generic;
using System.Data;
using System.Data.SqlClient;
using System.Linq;
using System.Web;
using System.Web.Services;
using System.Configuration;


/// <summary>
/// Summary description for WebService1
/// </summary>
[WebService(Namespace = "http://tempuri.org/")]
[WebServiceBinding(ConformsTo = WsiProfiles.BasicProfile1_1)]
// To allow this Web Service to be called from script, using ASP.NET AJAX, uncomment the following line.
// [System.Web.Script.Services.ScriptService]
public class WebService1 : System.Web.Services.WebService
{

    public WebService1()
    {

        //Uncomment the following line if using designed components
        //InitializeComponent();
    }

    [WebMethod]
    public DataSet GetData()
    {
        string connStr = ConfigurationManager.ConnectionStrings["narender_p9ConnectionString"].ConnectionString;
        SqlConnection conn = new SqlConnection(connStr);
        SqlDataAdapter da = new SqlDataAdapter("Select * from Student", conn);
        DataSet ds = new DataSet();
        da.Fill(ds);
        return ds;
    }
}
```

OUTPUT:



ID	NAME	DEGREE	SUBJECT
1	Narender	MCA	AWT
2	Neel Deshmukh	Bvoc SD	JS
3	Parnot	Btech	JAVA
4	Hassan	BVOC SD	WEB

Design Web Application to produce and Consume a WCF Service

B) Design an Application to fetch data from the EMP table. Test the service and design a Web client to consume this service.

SQL SERVER:

Results Messages

	id	name	dept	salary	date_of_joining
1	1	Narender Keswani	IT	100000	30-05-2021
2	2	Neel Deshmukh	CS	200000	06-05-2022
3	3	Hassan Haque	SD	300000	06-06-2022
4	4	Ritesh Yadav	MBA	566252	05-06-2019

WCF Web Service:

WebConfig:

```
  <add name="narender_p9ConnectionString" connectionString="server=. ; database=narender_p7;
Trusted_Connection=Yes;" providerName="System.Data.SqlClient" />
```

IService1.cs:

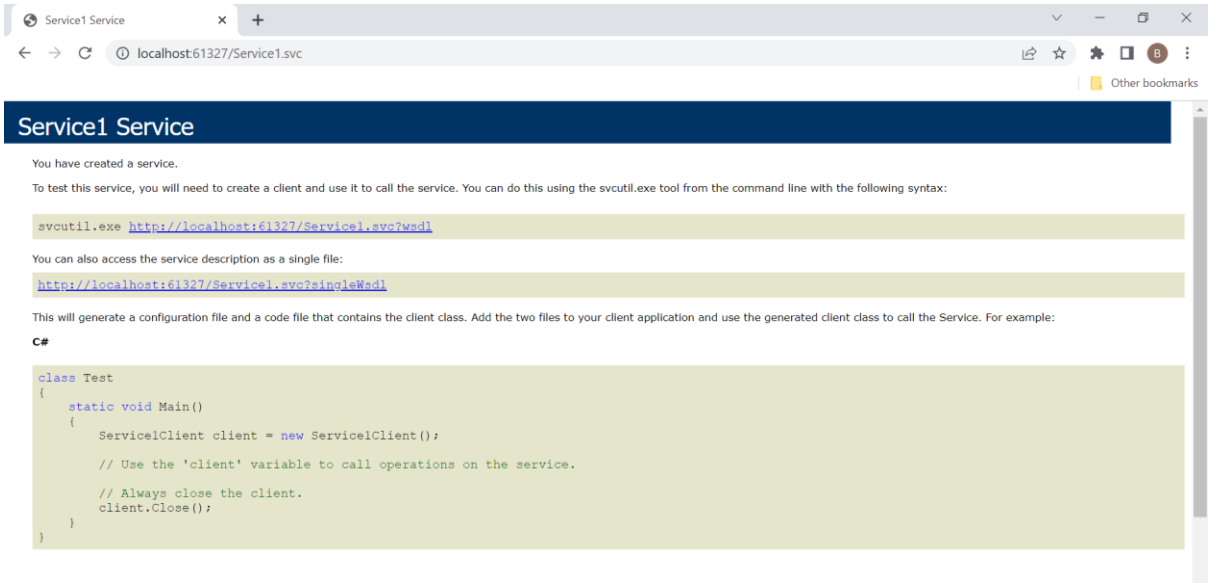
```
using System;
using System.Collections.Generic;
```

```
using System.Linq;
using System.Runtime.Serialization;
using System.ServiceModel;
using System.ServiceModel.Web;
using System.Text;
using System.Data.SqlClient;
using System.Data;

namespace WcfService1
{
   // NOTE: You can use the "Rename" command on the "Refactor" menu to change the interface name "IService1" in both code and config file together.

   [ServiceContract]
   public interface IService1
   {
      [OperationContract]
      DataSet getNarendersEmployees();
   }

   [DataContract]
   public class NarendersEmployees
   {

      int id;
      string name;
      string dept;
      int salary;
      string date_of_joining;

      [DataMember]
      public int Id
      {
         get { return id; }
         set { id = value; }
      }
      [DataMember]
      public string Name
      {
         get { return name; }
         set { name = value; }
      }
      [DataMember]
      public string Dept
      {
         get { return dept; }
         set { dept = value; }
      }
      [DataMember]
      public int Salary
      {
```

```
        get { return salary; }
        set { salary = value; }
      }
      [DataMember]
      public string Date_of_joining
      {
        get { return date_of_joining; }
        set { date_of_joining = value; }

      }

    }
}
```

Service1.svc:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Runtime.Serialization;
using System.ServiceModel;
using System.ServiceModel.Web;
using System.Text;
using System.Data;
using System.Data.SqlClient;
using System.Configuration;

namespace WcfService1
{
  // NOTE: You can use the "Rename" command on the "Refactor" menu to change the class name "Service1" in code, svc and config file together.
  // NOTE: In order to launch WCF Test Client for testing this service, please select Service1.svc or Service1.svc.cs at the Solution Explorer and start debugging.
  public class Service1 : IService1
  {
    public DataSet getNarendersEmployees()
    {
      string connStr = ConfigurationManager.ConnectionStrings["narender_p9ConnectionString"].ConnectionString;
      SqlConnection conn = new SqlConnection(connStr);
      conn.Open();
      SqlCommand cmd = new SqlCommand("Select * from employees_of_narender", conn);
      SqlDataAdapter sda = new SqlDataAdapter(cmd);
      DataSet ds = new DataSet();
      sda.Fill(ds);
      cmd.ExecuteNonQuery();
      conn.Close();
      return ds;
    }
  }
}
```

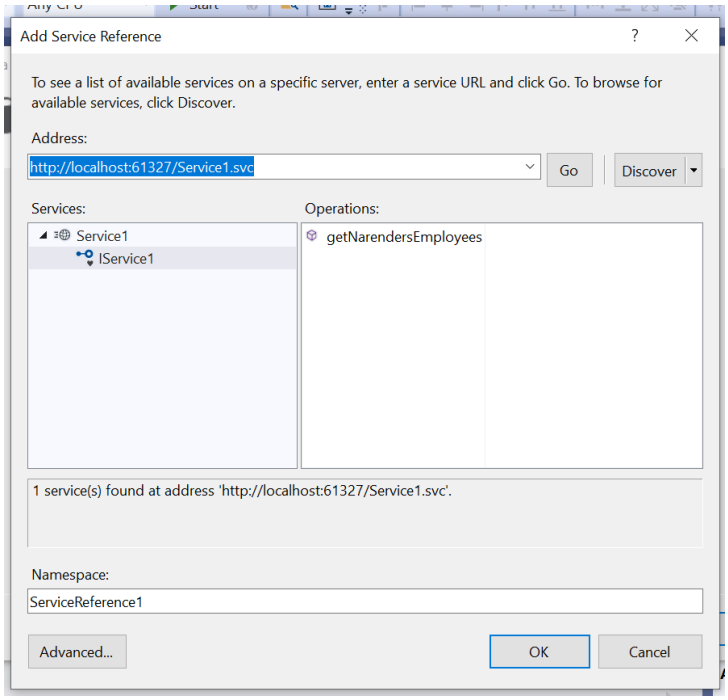


Windows Forms App:

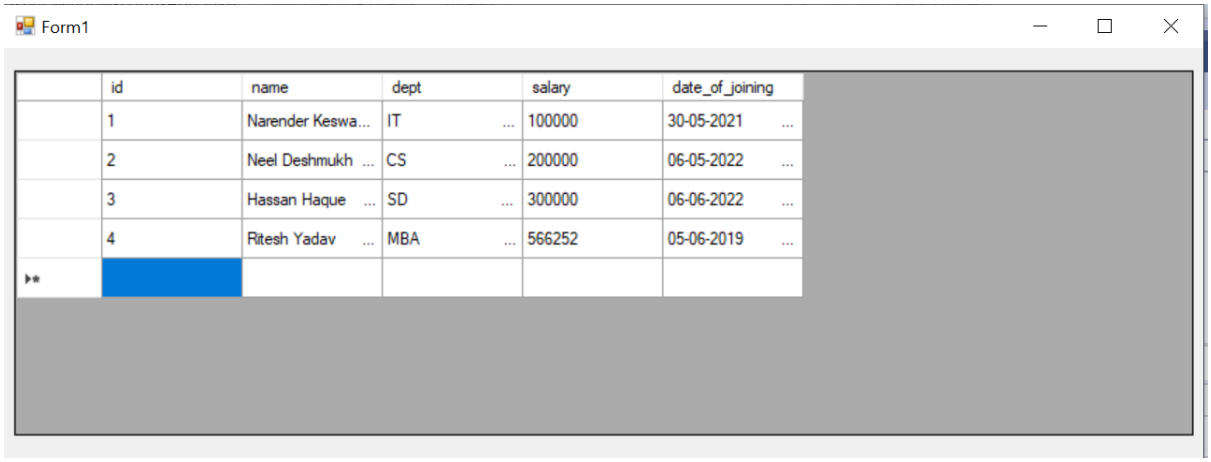
Form1.cs:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace WindowsFormsApp9BWCF
{
   public partial class Form1 : Form
   {
      ServiceReference1.Service1Client obj = new ServiceReference1.Service1Client(); // Add service
reference
      public Form1()
      {
         InitializeComponent();
         DataSet ds = new DataSet();
         ds = obj.getNarendersEmployees();
         dataGridView1.DataSource = ds.Tables[0];
      }
   }
}
```

ADD SERVICE REFERENCE:



OUTPUT:



CONCLUSION:

From this practical, I have learned & implemented the creation & consumption of webservice in asp.net.

AIM: Design MVC based Web applications

SOUCE CODE:

MODEL:
Employee.cs:

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

namespace WebApplication3.Models
{
   public class Employee
   {
      public string Name { get; set; }
      public string Address { get; set; }
      public int Age { get; set; }
   }
}
```

CONTROLLER:
HomeController.cs:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;
using WebApplication3.Models;

namespace WebApplication3.Controllers
{
   public class HomeController : Controller
   {
      public ActionResult Index()
      {
         return View();
      }

      public ActionResult About()
      {
         ViewBag.Message = "Your application description page.";

         return View();
```

```
    }

    public ActionResult Contact()
    {
        ViewBag.Message = "Your contact page.";

        return View();
    }

    [HttpGet]
    public ActionResult Emp()
    {
        Employee emp = new Employee()
        {
            Address = "Ulhasnagar",
            Name = "Nender Keswani",
            Age = 22
        };
        return View(emp);
    }
    [HttpPost]
    public ActionResult Emp(Employee emp)
    {
        return View("DisplayData", emp);
    }
  }
}
```

VIEWS:

Emp.cshtml:

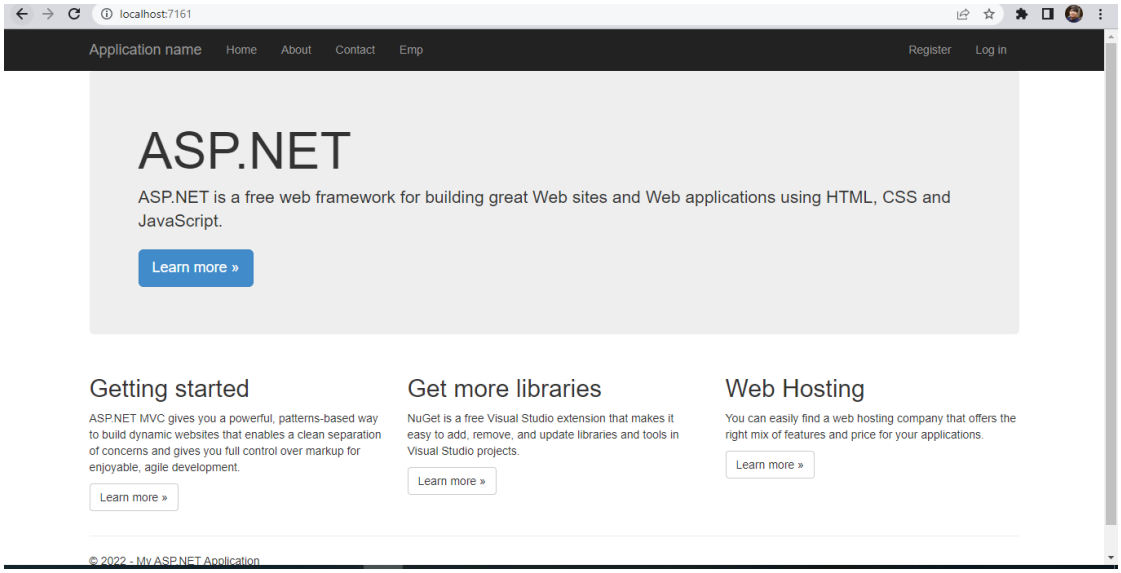
```
@{
    Layout = "~/Views/Shared/_Layout.cshtml";
}
@model WebApplication3.Models.Employee
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width" />
    <title>Index</title>
</head>
<body>
    <div>
        @using (Html.BeginForm("Myform"))
        {
```

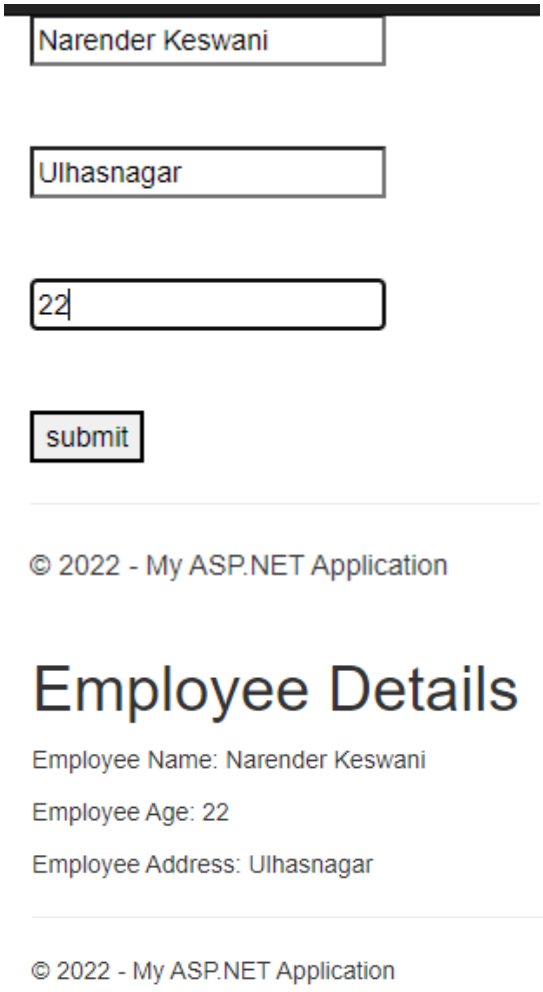
```
        @Html.TextBoxFor(x => x.Name) <br /> <br /> <br />
        @Html.TextBoxFor(x => x.Address) <br /> <br /> <br />
        @Html.TextBoxFor(x => x.Age) <br /> <br /> <br />
        <input type="submit" value="submit" />
      }
   </div>
</body>
</html>
```

DisplayData.cshml:

```
@{
   Layout = "~/Views/Shared/_Layout.cshtml";
}


<!DOCTYPE html>
<html>
<head>
   <meta name="viewport" content="width=device-width" />
   <title>DisplayData</title>
</head>
<body>
   <div>
      <h1>Employee Details</h1>
      <p>Employee Name: @Model.Name</p>
      <p>Employee Age: @Model.Age</p>
      <p>Employee Address: @Model.Address</p>
   </div>
</body>
</html>
```

OUTPUT:





CONCLUSION:

From this practical, I have learned & implemented MVC in asp.net