Agricultural Development Trust’s

Shardabai Pawar Art,Commerce & Science College Shardanagar,

Malegaon (Bk), Tal-Baramati, Dist- Pune 413 115.

Department Of Computer Science

CERTIFICATE

Seat No:-

Date:-

This is to certify that, the work entered in this journal is the work of Mr. / Miss____. Who has worked for the I / II / III / IV semester of the MSc (Comp.Sci) Year 2025- 2026 in the College. He / She has completed work as prescribed by Savitribai Phule Pune University, Pune.

Practical in-Charge **Head of Department**

Internal Examiner **External Examiner**

INDEX

Subject:-CS-514-MJ C#.Net Programming **Seat No:-**

Assignment No1:

C# Introduction

1. Write a C# program to find the factorial of a given number.

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

namespace Ass1_1
{
    internal class Program
    {
        static void Main(string[] args)
        {
            Console.Write("Enter a number: ");
            int number = int.Parse(Console.ReadLine());

            long factorial = 1;

            if (number < 0)
            {
                Console.WriteLine("Factorial is not defined for negative numbers.");
            }
            else
            {
                for (int i = 1; i <= number; i++)
                {
                    factorial *= i;
                }

                Console.WriteLine("Factorial of " + number + " is: " + factorial);
            }
        }
    }
}
```

Output:

```
Enter a number: 4
Factorial of 4 is: 24
```

2. Write a C# program to check whether a given number is prime or not.

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

namespace Ass1_2
{
    internal class Program
    {
        static void Main(string[] args)
        {
            Console.Write("Enter a number: ");
            int number = int.Parse(Console.ReadLine());
            bool isPrime = true;

            if (number <= 1)
            {
                isPrime = false;
            }
            else
            {
                for (int i = 2; i < number; i++)
                {
                    if (number % i == 0)
                    {
                        isPrime = false;
                        break;
                    }
                }
            }

            if (isPrime)
                Console.WriteLine(number + " is a prime number.");
            else
                Console.WriteLine(number + " is not a prime number.");
        }
    }
}
```

Output:

```
Enter a number: 5
5 is a prime number.

Enter a number: 22
22 is not a prime number.
```

3. Write a C# Sharp program to print on screen the output of adding, subtracting, multiplying and dividing of two numbers which will be entered by the user.

```csharp
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Ass1_3
{
    internal class Program
    {
        static void Main(string[] args)
        {
            Console.Write("Enter first number: ");
            double num1 = double.Parse(Console.ReadLine());


            Console.Write("Enter second number: ");
            double num2 = double.Parse(Console.ReadLine());

            double sum = num1 + num2;
            double difference = num1 - num2;
            double product = num1 * num2;
            double quotient = num2 != 0 ? num1 / num2 : double.NaN;

            Console.WriteLine("\nResults:");
            Console.WriteLine("Addition: " + sum);
            Console.WriteLine("Subtraction: " + difference);
            Console.WriteLine("Multiplication: " + product);

            if (num2 != 0)
                Console.WriteLine("Division: " + quotient);
            else
                Console.WriteLine("Division: Cannot divide by zero");
        }
    }
}
```

Output:

```
Enter first number: 50
Enter second number: 60

Results:
Addition: 110
Subtraction: -10
Multiplication: 3000
Division: 0.833333333333333
```

4. Write a C# program to check whether the given string is a palindrome or not.

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

namespace Ass1_4
{
    internal class Program
    {
        static void Main(string[] args)
        {
            Console.Write("Enter a string: ");
            string input = Console.ReadLine();

            string reversed = "";

                        for (int i = input.Length - 1; i >= 0; i--)
            {
                reversed += input[i];
            }

            )
            if (input.Equals(reversed, StringComparison.OrdinalIgnoreCase))
            {
                Console.WriteLine("The string is a palindrome.");
            }
            else
            {
                Console.WriteLine("The string is not a palindrome.");
            }
        }
    }
}
```

Output:

```
Enter a string: 151
The string is a palindrome.

Enter a string: 342
The string is not a palindrome.
```

5. Write a C# program to find the second largest integer in an array using loop?

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

namespace Ass1_5
{
    internal class Program
    {
        static void Main(string[] args)
        {
            int[] arr = { 10, 20, 5, 8, 30 };

            int max = arr[0];
            int secondMax = int.MinValue;

            for (int i = 1; i < arr.Length; i++)
            {
                if (arr[i] > max)
                {
                    secondMax = max;
                    max = arr[i];
                }
                else if (arr[i] > secondMax && arr[i] != max)
                {
                    secondMax = arr[i];
                }
            }

            Console.WriteLine("Second largest number is: " + secondMax);
        }
    }
}
```

Output:

```
Second largest number is: 20
```

6. Write a C# program to sort an array in ascending and descending order.

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

namespace Ass1_6
{
    internal class Program
    {
        static void Main(string[] args)
        {
            int[] arr = { 5, 2, 9, 1, 7 };

            // Sort in ascending order
            Array.Sort(arr);
            Console.WriteLine("Array in Ascending Order:");
            foreach (int num in arr)
            {
                Console.Write(num + " ");
            }

            Console.WriteLine();

            // Sort in descending order
            Array.Reverse(arr);
            Console.WriteLine("Array in Descending Order:");
            foreach (int num in arr)
            {
                Console.Write(num + " ");
            }
        }
    }
}
```

Output:

```
Array in Ascending Order:
1 2 5 7 9
Array in Descending Order:
9 7 5 2 1
```

7. Write a C# program to find minimum & maximum from array?

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

namespace Ass1_7
{
    internal class Program
    {
        static void Main(string[] args)
        {
            int[] arr = { 12, 5, 8, 19, 3, 25 };

            int min = arr[0];
            int max = arr[0];

            for (int i = 1; i < arr.Length; i++)
            {
                if (arr[i] < min)
                    min = arr[i];

                if (arr[i] > max)
                    max = arr[i];
            }

            Console.WriteLine("Minimum value: " + min);
            Console.WriteLine("Maximum value: " + max);
        }
    }

}
```

Output:

```
Minimum value: 3
Maximum value: 25
```

8. Write a C# program to create an MXN matrix and perform the following operation.

a. Addition

b. Multiplication

c. Transpose

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

namespace Ass1_8
{
   internal class Program
   {
      static void Main(string[] args)
      {
         int m = 2;
         int n = 2;

         int[,] A = { { 1, 2 }, { 3, 4 } };
         int[,] B = { { 5, 6 }, { 7, 8 } };

         int[,] sum = new int[m, n];
         int[,] product = new int[m, n];
         int[,] transposeA = new int[n, m];


         for (int i = 0; i < m; i++)
            for (int j = 0; j < n; j++)
               sum[i, j] = A[i, j] + B[i, j];
         for (int i = 0; i < m; i++)
            for (int j = 0; j < n; j++)
            {
               product[i, j] = 0;
               for (int k = 0; k < n; k++)
                  product[i, j] += A[i, k] * B[k, j];
            }

         for (int i = 0; i < m; i++)
            for (int j = 0; j < n; j++)
               transposeA[j, i] = A[i, j];


         Console.WriteLine("Addition:");
         PrintMatrix(sum);

         Console.WriteLine("\nMultiplication:");
         PrintMatrix(product);
```

```
            Console.WriteLine("\nTranspose of A:");
            PrintMatrix(transposeA);
        }

        static void PrintMatrix(int[,] matrix)
        {
            for (int i = 0; i < matrix.GetLength(0); i++)
            {
                for (int j = 0; j < matrix.GetLength(1); j++)
                    Console.Write(matrix[i, j] + " ");
                Console.WriteLine();
            }
        }
    }
}
```

Output:

```
Addition:
6 8
10 12

Multiplication:
19 22
43 50

Transpose of A:
1 3
2 4
```

9. Write a C# program to create an MXN matrix and perform the following operation.

a. Upper Triangular
b. Lower Triangula
c. Addition of row elements
d. Addition of column elements
e. Addition of diagonal elements

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

namespace Asss1_9
{
    using System;

    public class MatrixOperations
    {
        public static void Main(string[] args)
        {
            Console.Write("Enter the number of rows (M): ");
            int M = int.Parse(Console.ReadLine());

            Console.Write("Enter the number of columns (N): ");
            int N = int.Parse(Console.ReadLine());

            int[,] matrix = new int[M, N];

            Console.WriteLine("Enter matrix elements:");
            for (int i = 0; i < M; i++)
            {
                for (int j = 0; j < N; j++)
                {
                    Console.Write($"Enter element at [{i},{j}]: ");
                    matrix[i, j] = int.Parse(Console.ReadLine());
                }
            }

            Console.WriteLine("\nOriginal Matrix:");
            PrintMatrix(matrix);
            if (M == N)
            {
                Console.WriteLine("\nUpper Triangular Matrix:");
                PrintUpperTriangular(matrix);
            }
            else
            {
                Console.WriteLine("\nUpper/Lower Triangular operations are applicable
only for square matrices.");
            }


            if (M == N)
            {
                Console.WriteLine("\nLower Triangular Matrix:");
                PrintLowerTriangular(matrix);
            }
            Console.WriteLine("\nSum of Row Elements:");
            SumRowElements(matrix);
            Console.WriteLine("\nSum of Column Elements:");
            SumColumnElements(matrix);

            if (M == N)
            {
                Console.WriteLine("\nSum of Diagonal Elements:");
                SumDiagonalElements(matrix);
            }
            else
            {
                Console.WriteLine("\nDiagonal sum is applicable only for square
matrices.");
            }
        }

        public static void PrintMatrix(int[,] matrix)
        {
            int M = matrix.GetLength(0);
            int N = matrix.GetLength(1);

            for (int i = 0; i < M; i++)
            {
                for (int j = 0; j < N; j++)
                {
                    Console.Write(matrix[i, j] + "\t");
                }
                Console.WriteLine();
            }
        }

        public static void PrintUpperTriangular(int[,] matrix)
        {
            int M = matrix.GetLength(0);
            int N = matrix.GetLength(1);

            for (int i = 0; i < M; i++)
            {
                for (int j = 0; j < N; j++)
                {
                    if (i > j)                {
                        Console.Write("0\t");
                    }
                    else
                    {
                        Console.Write(matrix[i, j] + "\t");
                    }
                }
                Console.WriteLine();
            }
        }

        public static void PrintLowerTriangular(int[,] matrix)
        {
            int M = matrix.GetLength(0);
            int N = matrix.GetLength(1);

            for (int i = 0; i < M; i++)
            {
                for (int j = 0; j < N; j++)
                {
                    if (i < j)                {
                        Console.Write("0\t");
                    }
                    else
                    {
                        Console.Write(matrix[i, j] + "\t");
                    }
                }
                Console.WriteLine();
            }
        }

        public static void SumRowElements(int[,] matrix)
        {
            int M = matrix.GetLength(0);
            int N = matrix.GetLength(1);

            for (int i = 0; i < M; i++)
            {
                int rowSum = 0;
                for (int j = 0; j < N; j++)
                {
                    rowSum += matrix[i, j];
                }
                Console.WriteLine($"Sum of row {i}: {rowSum}");
            }
        }
```

```
        public static void SumColumnElements(int[,] matrix)
        {
            int M = matrix.GetLength(0);
            int N = matrix.GetLength(1);

            for (int j = 0; j < N; j++)
            {
                int colSum = 0;
                for (int i = 0; i < M; i++)
                {
                    colSum += matrix[i, j];
                }
                Console.WriteLine($"Sum of column {j}: {colSum}");
            }
        }

        public static void SumDiagonalElements(int[,] matrix)
        {
            int M = matrix.GetLength(0);
            int diagonalSum = 0;

            for (int i = 0; i < M; i++)
            {
                diagonalSum += matrix[i, i];
            }
            Console.WriteLine($"Sum of diagonal elements: {diagonalSum}");
        }
    }
}
```

Output:

```
Enter the number of rows (M): 2
Enter the number of columns (N): 2
Enter matrix elements:
Enter element at [0,0]: 3
Enter element at [0,1]: 2
Enter element at [1,0]: 4
Enter element at [1,1]: 2

Original Matrix:
3       2
4       2

Upper Triangular Matrix:
3       2
0       2

Lower Triangular Matrix:
3       0
4       2
```

```
Sum of Row Elements:
Sum of row 0: 5
Sum of row 1: 6

Sum of Column Elements:
Sum of column 0: 7
Sum of column 1: 4

Sum of Diagonal Elements:
Sum of diagonal elements: 5
```

10. Write a C# program to accept one string &character , find the occurrence of char from string using function

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

namespace Ass1_10
{
    class Program
    {
        public static int CountCharacterOccurrences(string text, char characterToFind)
        {
            int count = 0;
            foreach (char c in text)
            {
                if (c == characterToFind)
                {
                    count++;
                }
            }
            return count;
        }

        static void Main(string[] args)
        {
            Console.WriteLine("Enter a string:");
            string inputString = Console.ReadLine();

            Console.WriteLine("Enter the character to find:");
            char characterToSearch = Console.ReadKey().KeyChar;


            int occurrences = CountCharacterOccurrences(inputString,
characterToSearch);

            Console.WriteLine($"The character '{characterToSearch}' appears
{occurrences} time(s) in the string.");
        }
    }
}
```

Output:

```
Enter a string:
Sharvari
Enter the character to find:
qThe character 'q' appears 0 time(s) in the string.

aThe character 'a' appears 2 time(s) in the string.
```

Assignment No 2:

OOPs Concepts:

1. Write a program to define a class Students having data members rollno, name. Accept data for 5 student‟s and display the name of student whose roll no is 3.

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

namespace Ass2_1
{
    internal class Student
    {
        public int RollNo;
        public string Name;

        public Student(int rollNo, string name)
        {
            RollNo = rollNo;
            Name = name;
        }
    }
    class Program
    {

        static void Main(string[] args)
        {
            Student[] students = new Student[5];

            for (int i = 0; i < 5; i++)
            {
                Console.Write($"Enter roll no for student{i + 1}:");
                int rollNo = int.Parse(Console.ReadLine());

                Console.Write($"Enter name for student{i + 1}:");
                String name=Console.ReadLine();

                students[i] = new Student(rollNo, name);

            }
            bool found = false;
            foreach (Student student in students)
            {
                if (student.RollNo == 3)
                {
                    Console.WriteLine("student with roll number 3 found");
                    found = true;
                    break;
                }
            }
            if (!found)
            {
                Console.WriteLine("No student with roll number 3 found");
            }
        }

    }
}
```

Output:

```
Enter roll no for student1:1
Enter name for student1:Sharvari
Enter roll no for student2:2
Enter name for student2:Sayali
Enter roll no for student3:3
Enter name for student3:Dhanashri
Enter roll no for student4:4
Enter name for student4:Swara
Enter roll no for student5:5
Enter name for student5:Ankita
student with roll number 3 found
```

2. Write a program to swap three integer and three float numbers using the concept of Function overloading.

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

namespace Ass2_2
{
    class Swapper
    {
        public void Swap(ref int a, ref int b, ref int c)
        {
            int temp = a;
            a = b;
            b = c;
            c = temp;
        }

        public void Swap(ref float a, ref float b, ref float c)
        {
            float temp = a;
            a = b;
            b = c;
            c = temp;
        }
    }
    class Program
    {
        static void Main()
        {
            Swapper s = new Swapper();

            int a = 1, b = 2, c = 3;
            Console.WriteLine("Before swapping integers:");
            Console.WriteLine("a=" + a + ",b=" + b + ",c=" + c);
            s.Swap(ref a, ref b, ref c);

            Console.WriteLine("After swapping intgers:");
            Console.WriteLine("a=" + a + ",b=" + b + ",c=" + c);

            float x = 1.1f, y = 2.2f, z = 3.3f;
            Console.WriteLine("\nBefore swapping floats:");
            Console.WriteLine("x=" + x + ",y=" + y + ",z=" + z);

            s.Swap(ref x, ref y, ref z);

            Console.WriteLine("After swapping floats:");
            Console.WriteLine("x=" + x + ",y=" + y + ",z=" + z);
```

```
                }

            }
        }
```

Output:

```
Before swapping integers:
a=1,b=2,c=3
After swapping intgers:
a=2,b=3,c=1

Before swapping floats:
x=1.1,y=2.2,z=3.3
After swapping floats:
x=2.2,y=3.3,z=1.1
```

3. Implement a base class Person. Derive classes Student and Instructor from Person. A Person has aname and a birthday. A student has a batch, course and an Instructor has a salary. Write the class definitions, the constructor and the member function print () for all classes.

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

namespace Ass2_3
{
    class Person
    {
        protected string name;
        protected string birthday;

        public Person(string name, string birthday)
        {
            this.name = name;
            this.birthday = birthday;
        }
        public virtual void Print()
        {
            Console.WriteLine("Name:" + name);
            Console.WriteLine("Birthday:" + birthday);

        }

        class Student : Person
        {
            private string batch;
            private string course;
            public Student(string name, string birthday, string batch, string course)
                : base(name, birthday)
            {
                this.batch = batch;
                this.course = course;

            }
            public override void Print()
            {
                base.Print();
                Console.WriteLine("Batch:" + batch);
                Console.WriteLine("Course:" + course);
            }

        }
        class Instructor : Person
        {
            private double salary;

            public Instructor(string name, string birthday, double salary)
                : base(name, birthday)
            {
                this.salary = salary;
            }
            public override void Print()
            {
                base.Print();
                Console.WriteLine("Salary:" + salary);
            }
        }
        class Program
        {

            static void Main(string[] args)
            {
                Student student = new Student("Alice", "2002-04-15", "Batch A", "cs");
                Console.WriteLine("Student Info:");
                student.Print();
                Console.WriteLine("\n ---------\n");

                Instructor instructer = new Instructor("Dr.Smith", "1980-11-10", 75000);
                Console.WriteLine("Instuctor Info:");
                instructer.Print();
            }
        }
    }
}
```

Output

```
Student Info:

Name:Alice

Birthday:2002-04-15

Batch:Batch A

Course:cs

-------

Instuctor Info:

Name:Dr.Smith

Birthday:1980-11-10

Salary:75000
```

4.C# program to demonstrate the example of multilevel inheritance.

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

namespace Ass2_4
{
    class A

    {
        public void DisplayA()
        {
            Console.WriteLine("This is class A");

        }
    }
    class B : A
    {
        public void DisplayB()
        {
            Console.WriteLine("This is class B");

        }
    }
    class C : B {
        public void DisplayC() {
            Console.WriteLine("This is class C");
} static void Main(string[] args)
        {
            C obj=new C();
            obj.DisplayA();
            obj.DisplayB();
            obj.DisplayC();
        }
    }
}
```

Output:

```
This is class A

This is class B

This is class C
```

5. Write an application that receives the following information from a set of students:

Student Id:

Student Name:

Course Name:

Date of Birth:

The application should also display the information of all the students once the data is Entered.

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

namespace Asss2_5
{
    class Student
    {
        public int StudentId;
        public string StudentName;
        public string CourseName;
        public string DateOfBirth;

        public void Input()
        {
            Console.WriteLine("Enter Student ID:");
            StudentId = int.Parse(Console.ReadLine());

            Console.WriteLine("Enter Student Name:");
            StudentName = Console.ReadLine();

            Console.WriteLine("Enter Student Course Name:");
            CourseName = Console.ReadLine();

            Console.WriteLine("Enter Date Of Birth (dd-mm-yyyy):");
            DateOfBirth = Console.ReadLine();

        }
        public void Display()
            {
            Console.WriteLine("\n---- Student Details ----");
            Console.WriteLine("Student ID :" + StudentId);
            Console.WriteLine("Student Name :" + StudentName);
            Console.WriteLine("Course Name :" + CourseName);
            Console.WriteLine("Date of Birth :" + DateOfBirth);
        }
        static void Main(string[] args)
        {

            Student student = new Student();
            student.Input();
            student.Display();
        }
    }
}
```

Output:

```
Enter Student ID:

1

Enter Student Name:

Sharvari

Enter Student Course Name:

CS

Enter Date Of Birth (dd-mm-yyyy):

23-08-2004

---- Student Details ----

Student ID :1

Student Name :Sharvari

Course Name :CS

Date of Birth :23-08-2004
```

6. Write a program to declare class Distance having data members dist1, dist2, dist3. Initialize the two data members using constructor and store their addition in third data member using function and display addition.

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

namespace Ass2_6
{
    class Distance
    {

        private float dist1;
        private float dist2;
        private float dist3;
        private float dist4;

        public Distance(float d1, float d2, float d3)
        {
            dist1 = d1;
            dist2 = d2;
            dist3 = d3;
            dist4 = 0;
        }
        public void AddDistance()
        {
        dist4 = dist1 + dist2 + dist3;
        }
        public void Display()
        {
            Console.WriteLine("Distance 1:" + dist1);
            Console.WriteLine("Distance 2:" + dist2);
            Console.WriteLine("Distance 3:" + dist3);
            Console.WriteLine("Total Distance (dist1+ dist2+ dist3):" + dist4);
        }
        class Program
        {
            static void Main()
            {
                Distance d = new Distance(12.5f, 8.3f, 4.7f);
                d.AddDistance();
                d.Display();
            }
        }
    }
}
```

Output:

```
Distance 1:12.5

Distance 2:8.3

Distance 3:4.7

Total Distance (dist1+ dist2+ dist3):25.5
```

7.Program to implement the following multiple inheritance using interface.

```
using Ass2_7;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Ass2_7
{
    interface Gross
    {
        float GrossSal();
    }
    class Employee
    {
        protected string name;
        protected float basicSalary;

        public void GetData()
        {
            Console.WriteLine("Enter Employee Name:");
            name = Console.ReadLine();

            Console.WriteLine("Enter Basic Salary:");
            basicSalary = float.Parse(Console.ReadLine());
        }
    }
    class Salary : Employee, Gross
    {
        private float tada;
        private float gross;
        public float GrossSal()
        {
            tada = 0.3f * basicSalary;
            gross = basicSalary + tada;
            return gross;
        }
        public void DispSal()
        {
            Console.WriteLine("\n--- Salary Details ---");
            Console.WriteLine("Employee Name:" + name);
            Console.WriteLine("Basic Salary:" + basicSalary);
            Console.WriteLine("TADA (30%):" + tada);
            Console.WriteLine("Gross Salary:" + gross);

        }
        static void Main()
        {
            Salary emp = new Salary();
            emp.GetData();
            emp.GrossSal();
            emp.DispSal();
        }
    }
}
```

Output:

```
Enter Employee Name:
Vaishu
Enter Basic Salary:
80000

--- Salary Details ---
Employee Name:Vaishu
Basic Salary:80000
TADA (30%):24000
Gross Salary:104000
```

8. Write a program for above class hierarchy for the Employee where the base class is Employee and derived class and Programmer and Manager. Here make display function virtual which is common for all and which will display information of Programmer and Manager interactively.

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

namespace Ass2_8
{
    class Employee
    {
        public string Name { get; set; }
        public int Age { get; set; }


        public virtual void Display()
        {
            Console.WriteLine("Employee Information:");
            Console.WriteLine("Name: " + Name);
            Console.WriteLine("Age: " + Age);
        }
    }

    class Programmer : Employee
    {
        public string ProgrammingLanguage { get; set; }


        public override void Display()
        {
            Console.WriteLine("\n--- Programmer Information ---");
            Console.WriteLine("Name: " + Name);
            Console.WriteLine("Age: " + Age);
            Console.WriteLine("Programming Language: " + ProgrammingLanguage);
        }
    }

    class Manager : Employee
    {
        public int TeamSize { get; set; }

        public override void Display()
        {
            Console.WriteLine("\n--- Manager Information ---");
            Console.WriteLine("Name: " + Name);
            Console.WriteLine("Age: " + Age);
            Console.WriteLine("Team Size: " + TeamSize);
        }
    }

    class Program
    {
        static void Main()
        {
            Console.WriteLine("Enter Programmer details:");
            Programmer p = new Programmer();
            Console.Write("Name: ");
            p.Name = Console.ReadLine();
            Console.Write("Age: ");
            p.Age = Convert.ToInt32(Console.ReadLine());
            Console.Write("Programming Language: ");
            p.ProgrammingLanguage = Console.ReadLine();

            Console.WriteLine("\nEnter Manager details:");
            Manager m = new Manager();
            Console.Write("Name: ");
            m.Name = Console.ReadLine();
            Console.Write("Age: ");
            m.Age = Convert.ToInt32(Console.ReadLine());
            Console.Write("Team Size: ");
            m.TeamSize = Convert.ToInt32(Console.ReadLine());

            Employee emp1 = p;
            Employee emp2 = m;

            emp1.Display();
            emp2.Display();

            Console.ReadLine();
        }
    }

}
```

Output:

```
Enter Programmer details:
Name: Dhanashri
Age: 21
Programming Language: .NET

Enter Manager details:
Name: Sharvari
Age: 21
Team Size: 10

--- Programmer Information ---
```

```
Name: Dhanashri
Age: 21
Programming Language: .NET

--- Manager Information ---
Name: Sharvari
Age: 21
Team Size: 10
```

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

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



class Student
{
    protected int Roll_no;
    protected string Name;

    public void GetStudentData()
    {
        Console.Write("Enter Roll Number: ");
        Roll_no = Convert.ToInt32(Console.ReadLine());

        Console.Write("Enter Name: ");
        Name = Console.ReadLine();
    }

    public void DisplayStudentData()
    {
        Console.WriteLine("\n--- Student Details ---");
        Console.WriteLine("Roll Number: " + Roll_no);
        Console.WriteLine("Name: " + Name);
    }
}

class Test : Student
{
    protected int marks1, marks2;

    public void GetMarks()
    {
        Console.Write("Enter Marks 1: ");
        marks1 = Convert.ToInt32(Console.ReadLine());

        Console.Write("Enter Marks 2: ");
        marks2 = Convert.ToInt32(Console.ReadLine());
    }

    public void DisplayMarks()
    {
        Console.WriteLine("Marks 1: " + marks1);
        Console.WriteLine("Marks 2: " + marks2);
    }
}

class Result : Test
{
    private int total;

    public void CalculateTotal()
    {
        total = marks1 + marks2;
    }

    public void DisplayResult()
    {
        DisplayStudentData();
        DisplayMarks();
        Console.WriteLine("Total Marks: " + total);
    }
}

class Program
{
    static void Main()
    {
        Result r = new Result();


        r.GetStudentData();
        r.GetMarks();


        r.CalculateTotal();
        r.DisplayResult();

        Console.ReadLine();
    }
}
```

Output:

```
Enter Roll Number: 3
Enter Name: Sayali
Enter Marks 1: 59
Enter Marks 2: 89

--- Student Details ---
Roll Number: 3
Name: Sayali
Marks 1: 59
Marks 2: 89
Total Marks: 148
```

Assignment No 3:

Data Structure

1. Write a C# program to implement a stack with push and pop operations. Find the top element of the stack and check if the stack is empty or not.

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

namespace Ass3_1
{
    class Program
    {
        static void Main(string[] args)
        {
            Stack<int>stack = new Stack<int>();
            stack.Push(1);
            stack.Push(2);
            stack.Push(3);
            Console.WriteLine("Top Element:"+stack.Peek());

            Console.WriteLine("Popped:" + stack.Pop());
            Console.WriteLine("Popped:" + stack.Pop());

            Console.WriteLine("Is stack empty?" + (stack.Count == 0 ? "yes" : "No"));

            if (stack.Count > 0)
                Console.WriteLine("Top Element is:" + stack.Peek());
            else
                Console.WriteLine("stack is empty");

        }
    }
}
```

Output:

```
Top Element:3
Popped:3
Popped:2
Is stack empty?No
Top Element is:1
```

2.Write a C# program to find the top and bottom elements of a given stack.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace Ass3_2
{
   class Program
   {
      static void Main(string[] args)
      {
         Stack<int> stack = new Stack<int>();
         stack.Push(1);
         stack.Push(2);
         stack.Push(3);
         stack.Push(4);
         stack.Push(5);
         stack.Push(6);
         stack.Push(7);
         Console.WriteLine("Stack contents (top to bottom):");
         foreach (int item in stack)
         {
            Console.WriteLine(item);
         }
         if (stack.Count > 0)
         {
            int topElement = stack.Peek();
            Console.WriteLine("\nTop Element:" + topElement);
         }
         if (stack.Count > 0)
         {
            int[] stackArray = stack.ToArray();
            int bottomElement = stackArray[stackArray.Length - 1];
            Console.WriteLine("Bottom Element:" + bottomElement);
         }
      }
   }
}
```

Output:

```
Stack contents (top to bottom):
7
6
5
4
3
2
1
Top Element:7
Bottom Element:
```

Assignment No 4
Multithreading and I/O Stream

1.C# program to assign the name to the thread.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Threading;
namespace Ass3_3
{
   class Program
   {
      static void MyMethod()
      {
         Console.WriteLine("Thread Name:" + Thread.CurrentThread.Name);
      }
      static void Main(string[] args)
      {
         Thread t=new Thread(MyMethod);
         t.Name = "MySimpleThread";
         t.Start();
      }
   }
}
```

Output:

```
Thread Name:MySimpleThread
```

2.C# program to demonstrate the concept of parameter passing for thread

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

namespace Ass3_4
{
    class Program
    {

            static void ShowNumber(object number)
            {
            Console.WriteLine("Number of Threads:" + number);
            }

            static void Main(string[] args)
            {
            Thread thread = new Thread(ShowNumber);
            thread.Start(100);

            }
        }
    }
```

Output:

```
Number of Threads:100
```

3.C# program to read data from file character by character till the end of the file.

```
using System;
using System.Collections.Generic;
using System.IO;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Ass3_5
{
    class Program
    {
        static void Main()
        {
            string filePath = "\"C:\\Users\\Admin\\OneDrive\\Documents\\ic.jpeg\"";

            try
            {
                using (StreamReader reader = new StreamReader(filePath))
                {
                    int ch;

                    while ((ch = reader.Read()) != -1)
                    {
                        Console.Write((char)ch);
                    }
                }
            }
            catch (FileNotFoundException)
            {
                Console.WriteLine("File not Found" + filePath);
            }
            catch (Exception ex)
            {
                Console.WriteLine("Error Reading File:" + ex.Message);
            }

        }
    }
}
```

Output:

```
Error Reading File:Illegal characters in path.
```

4.C# program to compare the content of two files using StreamReader class

```
using System;
using System.Collections.Generic;
using System.IO;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Ass3_6
{
    class Program
    {
        public static void Main(string[] args)
        {
            string file1 = "file1.txt";
            string file2 = "file2.txt";

            try
            {
                using (StreamReader reader1 = new StreamReader(file1))
                using (StreamReader reader2 = new StreamReader(file2))
                {
                    int ch1,ch2;

                    while ((ch1 = reader1.Read()) != -1 && (ch2 = reader2.Read()) != -1)
                    {
                        if (ch1 != ch2)
                        {
                            Console.WriteLine("File are different");
                            return;
                        }
                    }
                    if (reader1.EndOfStream && reader2.EndOfStream)
                        Console.WriteLine("Files are Identical");
                    else
                        Console.WriteLine("Files are different in length");
                }

            }
            catch (Exception ex)
            {
                Console.WriteLine("Error:" + ex.Message);

            }
        }
    }
}
```

Output:

```
Error:Could not find file 'C:\Users\Admin\source\repos\Ass3_6\Ass3_6\bin\Debug\file1.txt'.
```

5.C# program to get the size of a specified folder including sub-folder

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.IO;

namespace Ass3_7
{
    class Program
    {
        static void Main(string[] args)
        {
            string folderPath = @"C:\YourFolderPath";

            try
            {
                long folderSize = GetFolderSize(folderPath);
                Console.WriteLine($"Total size of folder'{folderPath}'is{folderSize}bytes");
            }
            catch (Exception ex)
            {

                    Console.WriteLine("Error" + ex.Message);
                }
            }
            static long GetFolderSize(string folderPath)
            {
                long size = 0;

                string[] file = Directory.GetFiles(folderPath, "*", SearchOption.AllDirectories);

                foreach (string file2 in file)
                {

                        FileInfo fileInfo = new FileInfo(file2);
                        size += fileInfo.Length;
                    }
                    return size;
                }
            }
        }
```

Output:

6.C# program to demonstrate the BinaryReader and BinaryWriterclasses

```
using System;
using System.Collections.Generic;
using System.IO;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Ass3_8
{
    class Program
    {
        static void Main(string[] args)
        {
            string filePath = "data.bin";

            using (BinaryWriter writer = new BinaryWriter(File.Open(filePath,
FileMode.Create)))
            {
                writer.Write(123);
                writer.Write(12.3);
                writer.Write("Hello world");

            }
            Console.WriteLine("Data Written to file");

            using (BinaryReader reader = new BinaryReader(File.Open(filePath,
FileMode.Open)))
            {
                int intValue = reader.ReadInt32();
                double doubleValue = reader.ReadDouble();
            string stringValue = reader.ReadString();
            Console.WriteLine("Data read from file:");
            Console.WriteLine("Integer:" + intValue);
                Console.WriteLine("Double:" + doubleValue);
                Console.WriteLine("string:" + stringValue);
            }
        }
    }
}
```

Output:

```
Data Written to file

Data read from file:

Integer:123

Double:12.3

string:Hello
```

Assignment No.5

Assembly

1. Write a C# program which will demonstrate use of private assembly.

```
using System;
namespace MathLibrary
{
    public class MathOperations
    {
        public int Add(int a, int b)
        {
            return a + b;
        }
        public int Multiply(int a, int b)
        {
            return a * b;
        }
    }
}

using System;
using MathLibrary;
namespace ConsoleApp
{
    class Program
    {
        static void Main(string[] args)
        {
            MathOperations math = new MathOperations();
            int sum = math.Add(5, 7);
            int product = math.Multiply(3, 4);
            Console.WriteLine($"Sum: {sum}");
            Console.WriteLine($"Product: {product}");
        }
    }
}
```

Output:

```
Sum: 12
Product: 12
```

2. Write a C# program which will demonstrate use of public assembly.

```
namespace MyLibrary
{
    public class MyClass
    {
        public string SayHello(string name)
        {
            return $"Hello, {name}!";
        }
    }
}

using System;
using MyLibrary;
class Program
{
    static void Main()
    {
        var obj = new MyClass();
        string message = obj.SayHello("World");
        Console.WriteLine(message);
    }
}
```

Output: Hello, World!

3. Write a C# program which will demonstrate use of shared assembly.

```
namespace SharedLibrary
{
    public class MessageProvider
    {
        public string GetMessage(string name)
        {
            return $"Hello, {name} from Shared Assembly!";
        }
    }
}
using System;
using SharedLibrary;

class Program
{
    static void Main()
    {
        var msg = new MessageProvider().GetMessage("App1");
        Console.WriteLine(msg);
    }
}
```

Output:

```
Hello, Alice from Shared Assembly!
Press any key to exit...
```

Assignment No.6

Exception Handling:

1. Write a C# program that reads a list of integers from the user. Handle the exception that occurs if the user enters a value outside the range of Int32

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace ExceptionH
{
    internal class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Enter integers one per line. Type 'done' to finish:\n");

            while (true)
            {
                Console.Write("Enter an integer: ");
                string input = Console.ReadLine();

                if (input.Trim().ToLower() == "done")
                {
                    break;
                }

                try
                {
                    int number = Convert.ToInt32(input);
                    Console.WriteLine($"You entered: {number}");
                }
                catch (OverflowException)
                {
                          Console.WriteLine("Error: Number is too large or too small for Int32.");
                }
                catch (FormatException)
                {
                        Console.WriteLine("Error: Invalid input. Please enter a valid integer.");
                }
            }

            Console.WriteLine("\nFinished reading input. Press any key to exit.");
            Console.ReadKey();
        }
    }
}
```

Output:

```
Enter an integer: 34

You entered: 34

Enter an integer: t

Error: Invalid input. Please enter a valid integer.
```

2.Write a C# program that prompts the user to input a numeric integer and throws an exception if the number is less than 0 or greater than 1000.

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

namespace ExceptionH1
{
    internal class Program
    {
        static void Main(string[] args)
        {
            try
            {
                Console.Write("Enter an integer between 0 and 1000: ");
                string input = Console.ReadLine();
                int number = int.Parse(input);

                if (number < 0 || number > 1000)
                {
                    throw new ArgumentOutOfRangeException("number", number, "Number must be between 0 and 1000.");
                }
                Console.WriteLine($"You entered a valid number: {number}");
            }
            catch (FormatException)
            {
                Console.WriteLine("Error: Invalid input. Please enter a numeric integer.");
            }
            catch (ArgumentOutOfRangeException ex)
            {
                Console.WriteLine($"Error: {ex.Message}");
            }
            Console.WriteLine("\nPress any key to exit...");
            Console.ReadKey();
        }
    }
}
```

Output:

```
Enter an integer between 0 and 1000: 540

You entered a valid number: 540

Enter an integer between 0 and 1000: 1001

Error: Number must be between 0 and 1000.

Parameter name: number

Actual value was 1001
```

Assignment No.7

Windows Programming

1.Write a C# program to create a window application to perform following basic arithmetic operations.

```
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 form1
{
   public partial class Form1 : Form
   {
      double FirstNumber;
      double SecondNumber;
      string Operation;

      public Form1()
      {
         InitializeComponent();
      }

      private void button1_Click(object sender, EventArgs e)
      {
         if (textBox1.Text == "0" && textBox1.Text != null)
         {
            textBox1.Text = "1";
         }
         else
         {
            textBox1.Text = textBox1.Text + "1";
         }
      }

      private void button14_Click(object sender, EventArgs e)
      {
         if (textBox1.Text == "0" && textBox1.Text != null)
         {
            textBox1.Text = "2";
         }
         else
         {
            textBox1.Text = textBox1.Text + "2";
```

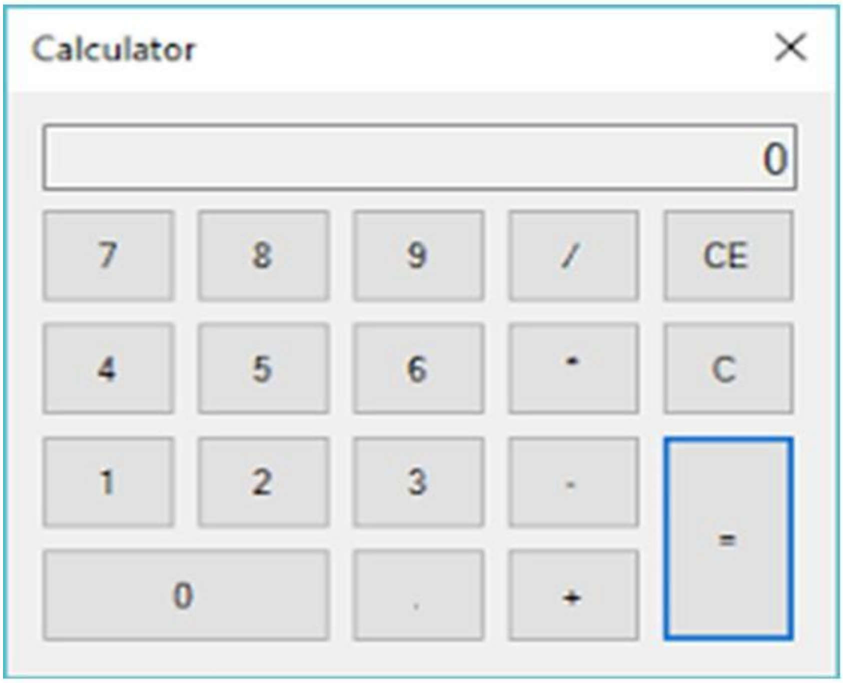
```
    }
}

private void button13_Click(object sender, EventArgs e)
{
    if (textBox1.Text == "0" && textBox1.Text != null)
    {
        textBox1.Text = "3";
    }
    else
    {
        textBox1.Text = textBox1.Text + "3";
    }
}

private void button12_Click(object sender, EventArgs e)
{
    if (textBox1.Text == "0" && textBox1.Text != null)
    {
        textBox1.Text = "4";
    }
    else
    {
        textBox1.Text = textBox1.Text + "4";
    }
}

private void button11_Click(object sender, EventArgs e)
{
    if (textBox1.Text == "0" && textBox1.Text != null)
    {
        textBox1.Text = "5";
    }
    else
    {
        textBox1.Text = textBox1.Text + "5";
    }
}

private void button10_Click(object sender, EventArgs e)
{
    if (textBox1.Text == "0" && textBox1.Text != null)
    {
        textBox1.Text = "6";
    }
    else
    {
        textBox1.Text = textBox1.Text + "6";
    }
}
```

```
private void button9_Click(object sender, EventArgs e)
{
    if (textBox1.Text == "0" && textBox1.Text != null)
    {
        textBox1.Text = "7";
    }
    else
    {
        textBox1.Text = textBox1.Text + "7";
    }
}

private void button15_Click(object sender, EventArgs e)
{
    if (textBox1.Text == "0" && textBox1.Text != null)
    {
        textBox1.Text = "8";
    }
    else
    {
        textBox1.Text = textBox1.Text + "8";
    }
}

private void button2_Click(object sender, EventArgs e)
{
    if (textBox1.Text == "0" && textBox1.Text != null)
    {
        textBox1.Text = "9";
    }
    else
    {
        textBox1.Text = textBox1.Text + "9";
    }

}

private void button3_Click(object sender, EventArgs e)
{
    if (textBox1.Text == "0" && textBox1.Text != null)
    {
        textBox1.Text = "0";
    }
    else
    {
        textBox1.Text = textBox1.Text + "0";
    }
}
```

```
private void button8_Click(object sender, EventArgs e)
{
    FirstNumber = Convert.ToDouble(textBox1.Text);
    textBox1.Text = "0";
    Operation = "+";
}

private void button7_Click(object sender, EventArgs e)
{
    FirstNumber = Convert.ToDouble(textBox1.Text);
    textBox1.Text = "0";
    Operation = "-";
}

private void button6_Click(object sender, EventArgs e)
{
    FirstNumber = Convert.ToDouble(textBox1.Text);
    textBox1.Text = "0";
    Operation = "*";

}

private void button5_Click(object sender, EventArgs e)
{
    FirstNumber = Convert.ToDouble(textBox1.Text);
    textBox1.Text = "0";
    Operation = "/";
}

private void button16_Click(object sender, EventArgs e)
{
    textBox1.Text="0";
}

private void button4_Click(object sender, EventArgs e)
{
    double Result;

    SecondNumber = Convert.ToDouble(textBox1.Text);
    if (Operation == "+")
    {
        Result = (FirstNumber + SecondNumber);
        textBox1.Text=Convert.ToString(Result);
        FirstNumber = Result;
    }
    if (Operation == "-")
    {
        Result = (FirstNumber - SecondNumber);
        textBox1.Text = Convert.ToString(Result);
        FirstNumber = Result;
```

```
            }
            if (Operation == "*")
            {
                Result = (FirstNumber * SecondNumber);
                textBox1.Text = Convert.ToString(Result);
                FirstNumber = Result;
            }
            if (Operation == "/")
            {

                if (SecondNumber == 0)
                {
                    textBox1.Text = "Cannot Divide by zero";
                }
                else
                {
                    Result = (FirstNumber /SecondNumber);
                    textBox1.Text = Convert.ToString(Result);
                    FirstNumber = Result;
                }
            }
        }
    }
}
```

Output:

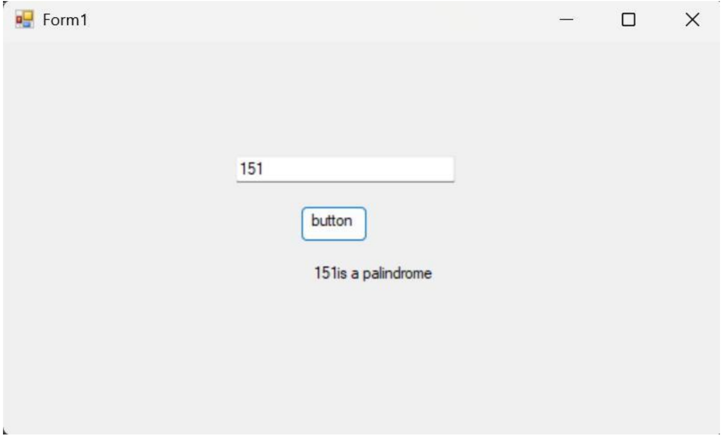


2. Create an application that accepts a number from a user in the textbox named num“. Check whether the number in the textbox num“ is palindrome or not. Print the message accordingly in the label control named lbldisplay when the user clicks on the button check.

```
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 slip10_2
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            string input = textBox1.Text.Trim();
            if (!int.TryParse(input, out int number))
            {
                label1.Text = "Please enter a valid integer";
                return;
            }
            if (IsPalindrome(number))

            {
                label1.Text = $"{number}is a palindrome";
            }
            else
            {
                label1.Text = $"{number}is a not palindrome";

            }
        }

        private bool IsPalindrome(int number)
        {
            string str = number.ToString();
            char[] arr = str.ToCharArray();
            Array.Reverse(arr);
```

```
                string reversed = new string(arr);
                return str == reversed;

            }
        }
    }
```

Output

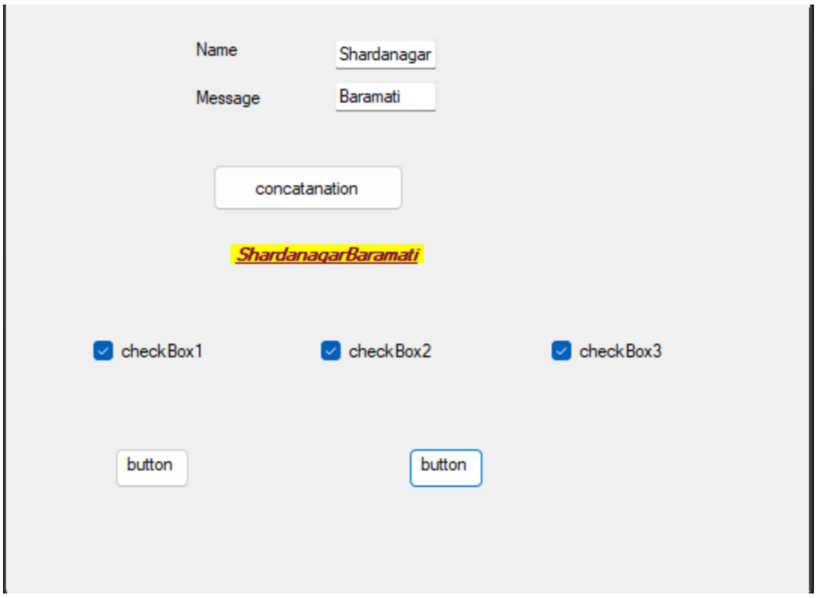


151is a palindrome

3.Write a C# program which will ask the user to input his name and a message and perform the following operations: 1. Display their concatenation on the label. 2. To change the format of the label using checkboxes bold, underlined and italic. 3. To change the colour of label (background and foreground) using colour dialog control.

```
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 slip4_2
 {
 public partial class Form1 : Form
{
private void UpdateLabelFont()
 {
 FontStyle style = FontStyle.Regular;
if(checkBox1.Checked )
style |= FontStyle.Bold;
 if (checkBox2.Checked)
style |= FontStyle.Italic;
if (checkBox3.Checked)
 style |= FontStyle.Underline;
 label3.Font=new Font(label3.Font.FontFamily,label3.Font.Size,style);
}
 public Form1()
{
InitializeComponent();
}
 private void button1_Click(object sender, EventArgs e)
{
 string input1=textBox1.Text;
 string input2=textBox2.Text;
string concatenated=input1 + input2; label3.Text = concatenated;
}
 private void checkBox1_CheckedChanged(object sender, EventArgs e)
 {
UpdateLabelFont();
 }
private void checkBox2_CheckedChanged(object sender, EventArgs e)
 {
UpdateLabelFont();
}
 private void checkBox3_CheckedChanged(object sender, EventArgs e)
{
 UpdateLabelFont();
}
 private void button2_Click(object sender, EventArgs e)
{
 if (colorDialog1.ShowDialog() == DialogResult.OK)
 {
label3.ForeColor = colorDialog1.Color;
}
 }
private void button3_Click(object sender, EventArgs e)
{
if (colorDialog1.ShowDialog() == DialogResult.OK)
{
 label3.BackColor = colorDialog1.Color;
    }
   }
 }
}
```

Output:



Name: Shardanagar

Message: Baramati

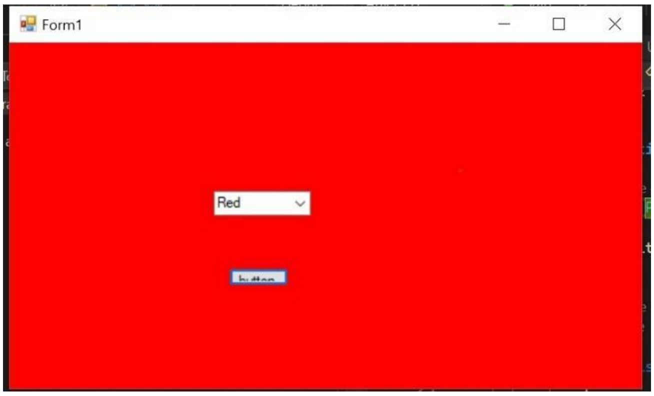
ShardanagarBaramati

☑ checkBox1 ☑ checkBox2 ☑ checkBox3

4.Create a user control that contains a list of colors. Add a button to the Form or testbox which when clicked changes the color of the Form or textbox to the color selected from the list.

```
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 form2
{
    public partial class Form1 : Form

        public Form1()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            this.BackColor = userControl11.SelectedColor;
        }
    }
}
```

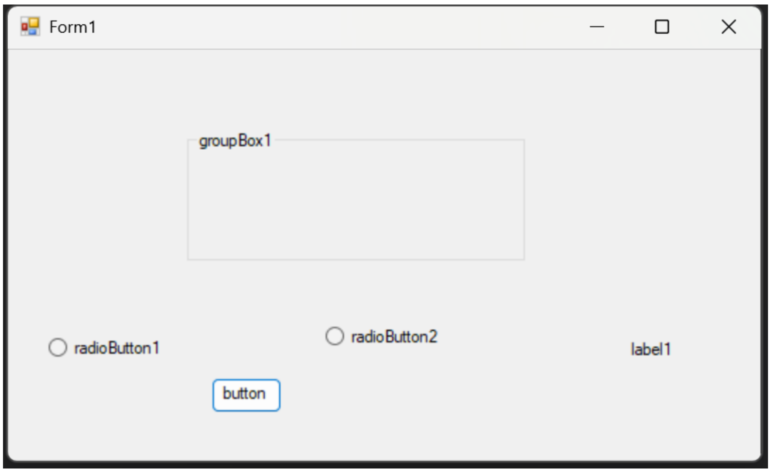
Output:



5.Create a RadioButtonList that displays the names of some flowers in two columns. Bind a label to the RadioButtonList so that when the user selects an option from the list and clicks on a button, the label displays the flower selected by the user.

```
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 slip6_2
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            string[] flowers = { "Rose", "Lotus", "jaee", "jue", "lily" };
            int columnCount = 2;
            int row = 0, column = 0;

            int spacingX = 120;
            int spacingY = 30;
            for (int i = 0; i < flowers.Length; i++)
            {
                RadioButton rb = new RadioButton();
                rb.Text = flowers[i];
                rb.AutoSize = true;
                rb.Location = new Point(10 + column * spacingX, 10 + row * spacingY);
                rb.Tag = flowers[i];
                groupBox1.Controls.Add(rb);

                column++;
                if (column >= columnCount)
                {
                    column = 0;
                    row++;
                }
            }
        }

        private void groupBox1_Enter(object sender, EventArgs e)
        {

        }
        private void radioButton1_CheckedChanged(object sender, EventArgs e)
        {
            foreach (Control ctrl in groupBox1.Controls)
            {
                if (ctrl is RadioButton rb && rb.Checked)
                {
                    label1.Text = "Selected Flower: " + rb.Tag.ToString();
                    return;
                }
            }
            label1.Text = "Please Select a flower.";
        }
    }
}
```

Output:



Assignment No.8

Database Connectivity using ADO.Net:

1. Write a C# program to create a table “student” with attributes rollno, name and per. Insert at least five records in a table and perform the following queries:
a. Delete the details of the students who got less than 40% marks.

b. Display the student’s name who got more than 90% marks.

c. Display the name of the students who are starting with K character.

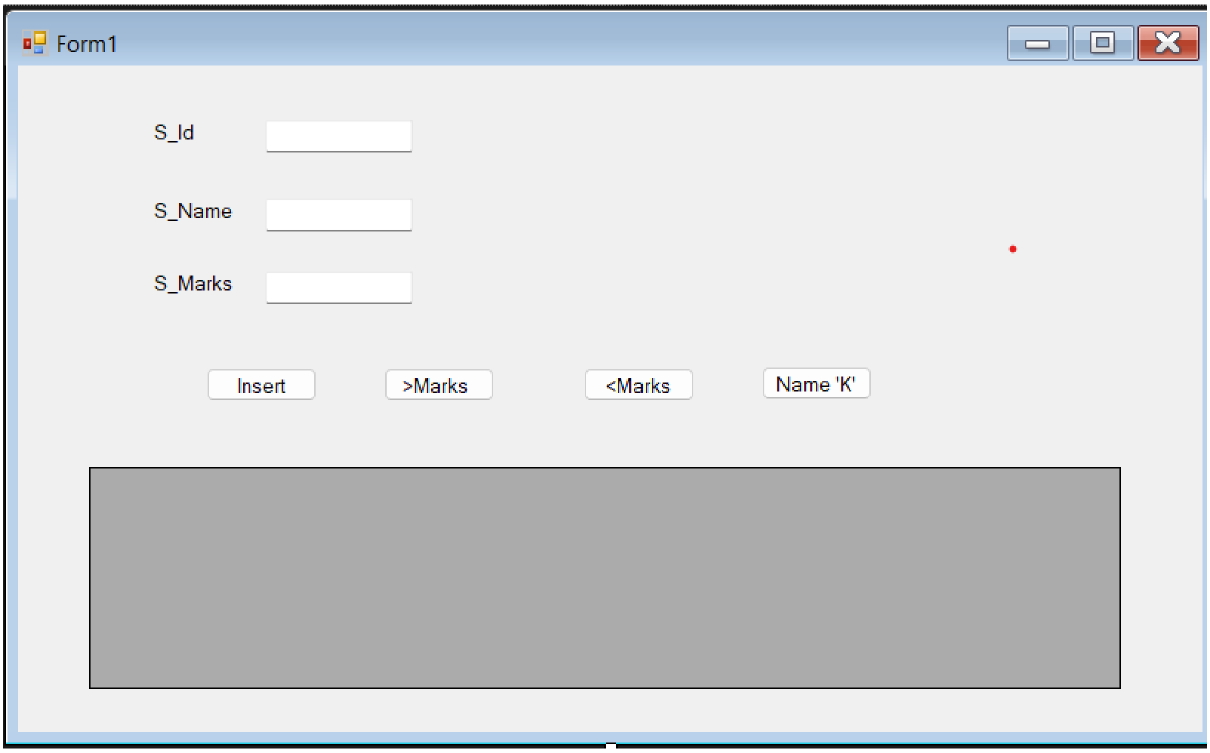
#Create database

Create a table student with:

- rollno (int)
- name (string)
- per (float/decimal)

```
using System;

using System.Data;

using System.Data.SqlClient;

using System.Windows.Forms;

namespace Stud_Table

{

   public partial class Form1 : Form

   {

      public Form1()

      {

         InitializeComponent();

      }


      private void button1_Click(object sender, EventArgs e)

      {

         using (SqlConnection con = new SqlConnection(@"Data
Source=AKSHAY\SQLEXPRESS;Initial Catalog=CRUD;Integrated
Security=True;Encrypt=False"))
```

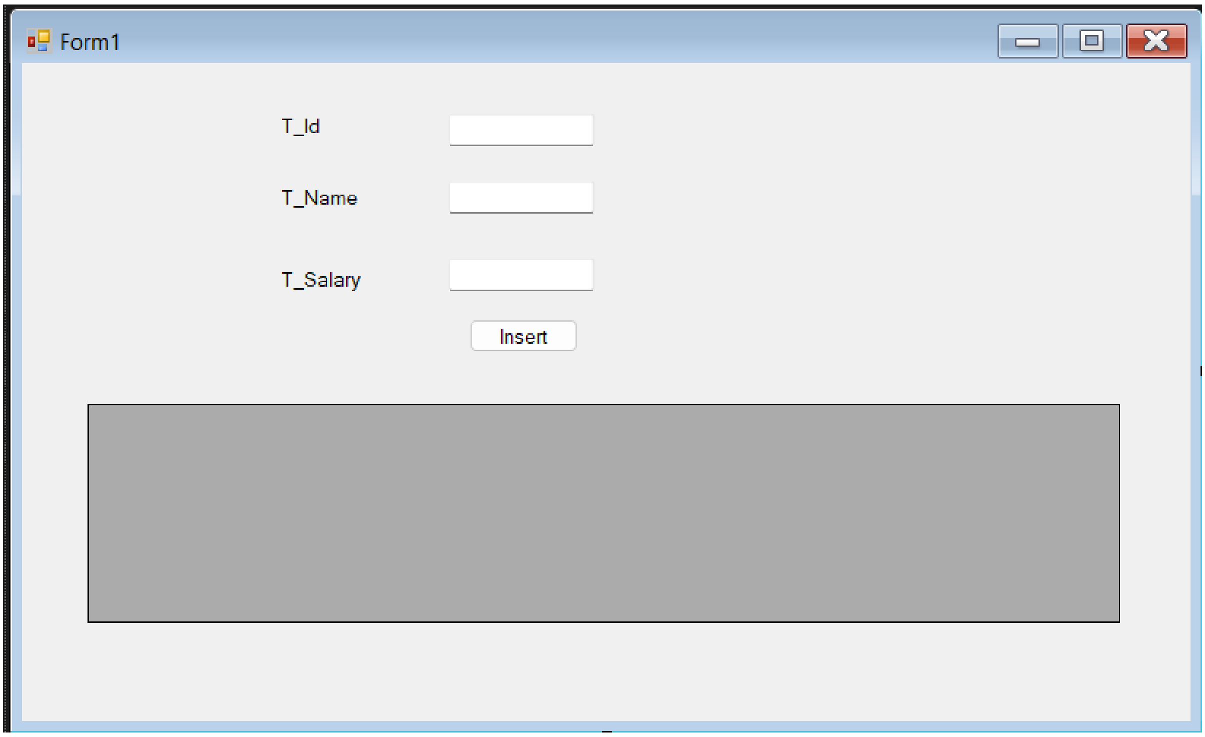
```
            {
                string query = "INSERT INTO Stud_T (S_Id, S_Name, S_Marks) VALUES
(@S_Id, @S_Name, @S_Marks)";
                SqlCommand cmd = new SqlCommand(query, con);
                cmd.Parameters.AddWithValue("@S_Id", int.Parse(textBox1.Text));
                cmd.Parameters.AddWithValue("@S_Name", textBox2.Text);
                cmd.Parameters.AddWithValue("@S_Marks", float.Parse(textBox3.Text));

                con.Open();
                cmd.ExecuteNonQuery();
                con.Close();

                MessageBox.Show("Stud Added");
                // LoadData();
            }
        }

        private void button2_Click(object sender, EventArgs e)
        {
            using (SqlConnection con = new SqlConnection(@"Data
Source=AKSHAY\SQLEXPRESS;Initial Catalog=CRUD;Integrated
Security=True;Encrypt=False"))
            {

                {
                    string query = "SELECT * FROM Stud_T WHERE S_Marks > @S_Marks";

                    SqlCommand cmd = new SqlCommand(query, con);
                    cmd.Parameters.AddWithValue("@S_Marks", float.Parse(textBox3.Text));

                    SqlDataAdapter da = new SqlDataAdapter(cmd);
                    DataTable dt = new DataTable();

                    da.Fill(dt);

                    dataGridView1.DataSource = dt;

                }

            }

        }


        private void button4_Click(object sender, EventArgs e)

        {

            using (SqlConnection con = new SqlConnection(@"Data
Source=AKSHAY\SQLEXPRESS;Initial Catalog=CRUD;Integrated
Security=True;Encrypt=False"))

            {


                {

                    string query = "SELECT * FROM Stud_T WHERE S_Marks < @S_Marks";


                    SqlCommand cmd = new SqlCommand(query, con);

                    cmd.Parameters.AddWithValue("@S_Marks", float.Parse(textBox3.Text));


                    SqlDataAdapter da = new SqlDataAdapter(cmd);

                    DataTable dt = new DataTable();

                    da.Fill(dt);

                    dataGridView1.DataSource = dt;

                }

            }

        }


        private void button3_Click(object sender, EventArgs e)

        {
            using (SqlConnection con = new SqlConnection(@"Data
Source=AKSHAY\SQLEXPRESS;Initial Catalog=CRUD;Integrated
Security=True;Encrypt=False"))
            {
                string query = "SELECT * FROM Stud_T WHERE S_Name LIKE 'k%'";
                SqlCommand cmd = new SqlCommand(query, con);
                SqlDataAdapter da = new SqlDataAdapter(cmd);
                DataTable dt = new DataTable();
                da.Fill(dt);
                dataGridView1.DataSource = dt;
            }
        }
    }
}
```



2. Write a C# program to create Teacher table with attributes TID, TName, and salary. Insert at least five records in a table and display them on the Gridview control.

```
using System;

using System.Data;

using System.Data.SqlClient;

using System.Windows.Forms;

namespace S_Data

{

   public partial class Form1 : Form

   {

      public Form1()

      {

         InitializeComponent();

      }

      private void button1_Click(object sender, EventArgs e)

      {

         using (SqlConnection con = new SqlConnection(@"Data
Source=AKSHAY\SQLEXPRESS;Initial Catalog=CRUD;Integrated
Security=True;Encrypt=False"))

         {

            string query = "INSERT INTO T_Table (T_Id, T_Name, T_Salary) VALUES
(@S_Id, @S_Name, @S_Marks)";

            SqlCommand cmd = new SqlCommand(query, con);

            cmd.Parameters.AddWithValue("@S_Id", int.Parse(textBox1.Text));

            cmd.Parameters.AddWithValue("@S_Name", textBox2.Text);

            cmd.Parameters.AddWithValue("@S_Marks", int.Parse(textBox3.Text));

            con.Open();

            cmd.ExecuteNonQuery();

            con.Close();

            MessageBox.Show("Stud Added");

            // LoadData();

         }
```

```
        }
    }
}
```

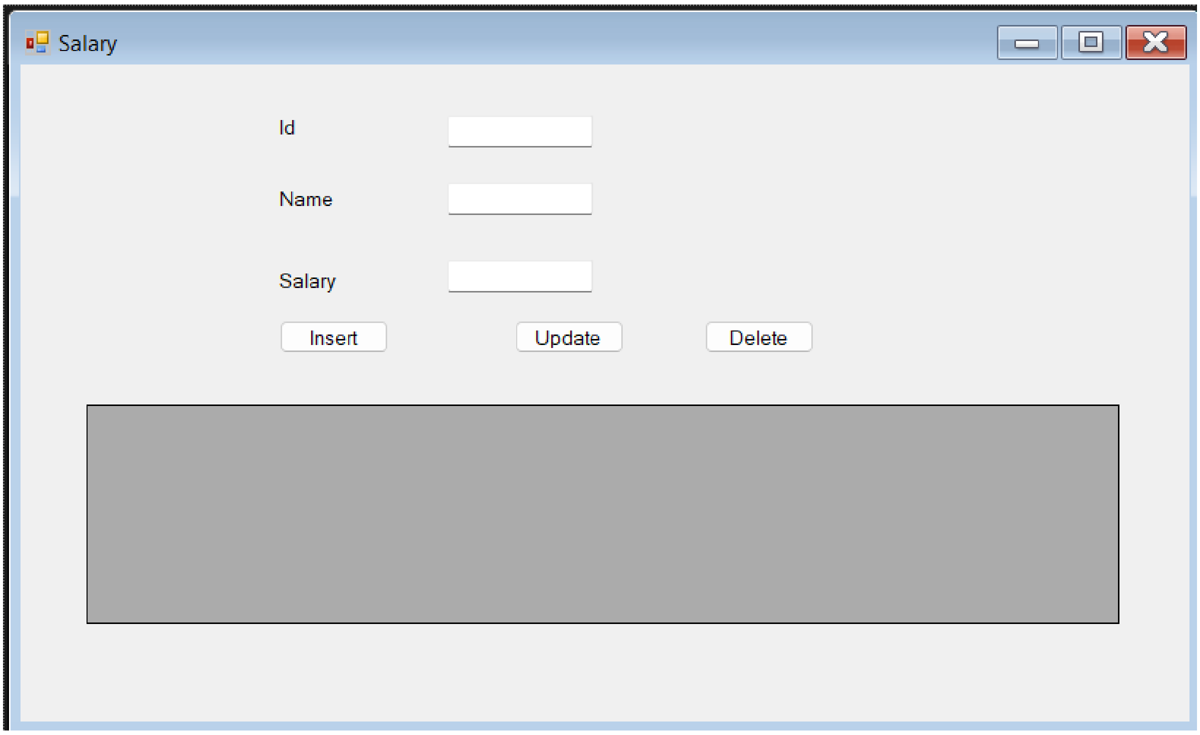

Form1
T_Id
T_Name
T_Salary
Insert

3. Write a C# program using ADO.NET to perform insert, delete, update operations on Emp (ENo, EName and Salary) table.

```
using System;
using System.Data;
using System.Data.SqlClient;
using System.Windows.Forms;
public partial class Form1 : Form
{
    string connString = @"Data Source=(localdb)\MSSQLLocalDB;Initial Catalog=EmpDB;Integrated Security=True";

    public Form1()
    {
        InitializeComponent();
        InitializeDatabase();
    }

    private void InitializeDatabase()
    {
        using (SqlConnection conn = new SqlConnection(connString))
        {
            conn.Open();

            // Create database if not exists
            using (SqlCommand cmd = new SqlCommand("IF DB_ID('EmpDB') IS NULL CREATE DATABASE EmpDB;", conn))
            {
                cmd.ExecuteNonQuery();
            }

            conn.ChangeDatabase("EmpDB");
            // Create Emp table if not exists
            string createTable = @"
                IF OBJECT_ID('Emp', 'U') IS NULL
                CREATE TABLE Emp (
                    ENo INT PRIMARY KEY,
                    EName NVARCHAR(50),
                    Salary FLOAT
                );";
            SqlCommand createCmd = new SqlCommand(createTable, conn);
            createCmd.ExecuteNonQuery();
        }
    }

    private void btnInsert_Click(object sender, EventArgs e)
    {
        using (SqlConnection conn = new SqlConnection(connString))
        {
            conn.Open();
            conn.ChangeDatabase("EmpDB");

            string insert = "INSERT INTO Emp (ENo, EName, Salary) VALUES (@ENo,
@EName, @Salary)";
            SqlCommand cmd = new SqlCommand(insert, conn);
            cmd.Parameters.AddWithValue("@ENo", int.Parse(txtENo.Text));
            cmd.Parameters.AddWithValue("@EName", txtEName.Text);
            cmd.Parameters.AddWithValue("@Salary", float.Parse(txtSalary.Text));

            try
            {
                cmd.ExecuteNonQuery();
```

```
                    MessageBox.Show("Record inserted.");
                    LoadData();
                }
                catch (SqlException ex)
                {
                    MessageBox.Show("Error: " + ex.Message);
                }
            }
        }

        private void btnUpdate_Click(object sender, EventArgs e)
        {
            using (SqlConnection conn = new SqlConnection(connString))
            {
                conn.Open();
                conn.ChangeDatabase("EmpDB");

                string update = "UPDATE Emp SET EName = @EName, Salary = @Salary WHERE ENo = @ENo";
                SqlCommand cmd = new SqlCommand(update, conn);
                cmd.Parameters.AddWithValue("@ENo", int.Parse(txtENo.Text));
                cmd.Parameters.AddWithValue("@EName", txtEName.Text);
                cmd.Parameters.AddWithValue("@Salary", float.Parse(txtSalary.Text));

                int rows = cmd.ExecuteNonQuery();
                if (rows > 0)
                {
                    MessageBox.Show("Record updated.");
                    LoadData();
                }
                else
                {
                    MessageBox.Show("Record not found.");
                }
            }
        }

        private void btnDelete_Click(object sender, EventArgs e)
        {
            using (SqlConnection conn = new SqlConnection(connString))
            {
                conn.Open();
                conn.ChangeDatabase("EmpDB");

                string delete = "DELETE FROM Emp WHERE ENo = @ENo";
                SqlCommand cmd = new SqlCommand(delete, conn);
                cmd.Parameters.AddWithValue("@ENo", int.Parse(txtENo.Text));

                int rows = cmd.ExecuteNonQuery();
                if (rows > 0)
                {
                    MessageBox.Show("Record deleted.");
                    LoadData();
                }
                else
                {
                    MessageBox.Show("Record not found.");
                }
            }
        }
```

```
private void btnLoad_Click(object sender, EventArgs e)
{
    LoadData();
}
private void LoadData()
{
    using (SqlConnection conn = new SqlConnection(connString))
    {
        conn.Open();
        conn.ChangeDatabase("EmpDB");
        string select = "SELECT * FROM Emp";
        SqlDataAdapter adapter = new SqlDataAdapter(select, conn);
        DataTable dt = new DataTable();
        adapter.Fill(dt);
        dataGridView1.DataSource = dt;
    }
}
```



```
}
```