C# and .Net Frameworks

Assignment 1

1. Develop the C# program to initialize two dimensional array and print all the elements of the array on the same line separated with space.

AIM:

To develop a C# program that initializes a two-dimensional array and prints all the elements of the array in a single line, separated by spaces.

PROGRAM:

```
using System;
class Program{
   static void Main(){
      Console.Write("Enter the number of rows: ");
      int rows = int.Parse(Console.ReadLine());

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

      int[,] array = new int[rows, columns];
      Console.WriteLine("Enter the elements of the array:");

      for (int i = 0; i < rows; i++){
         for (int j = 0; j < columns; j++){
            array[i, j] = int.Parse(Console.ReadLine());
         }}
```

```
    Console.WriteLine("The elements of the array are:");
    for (int i = 0; i < rows; i++)
    {
       for (int j = 0; j < columns; j++)
       {
          Console.Write(array[i, j] + " ");
       }}
    Console.WriteLine();
}}
```

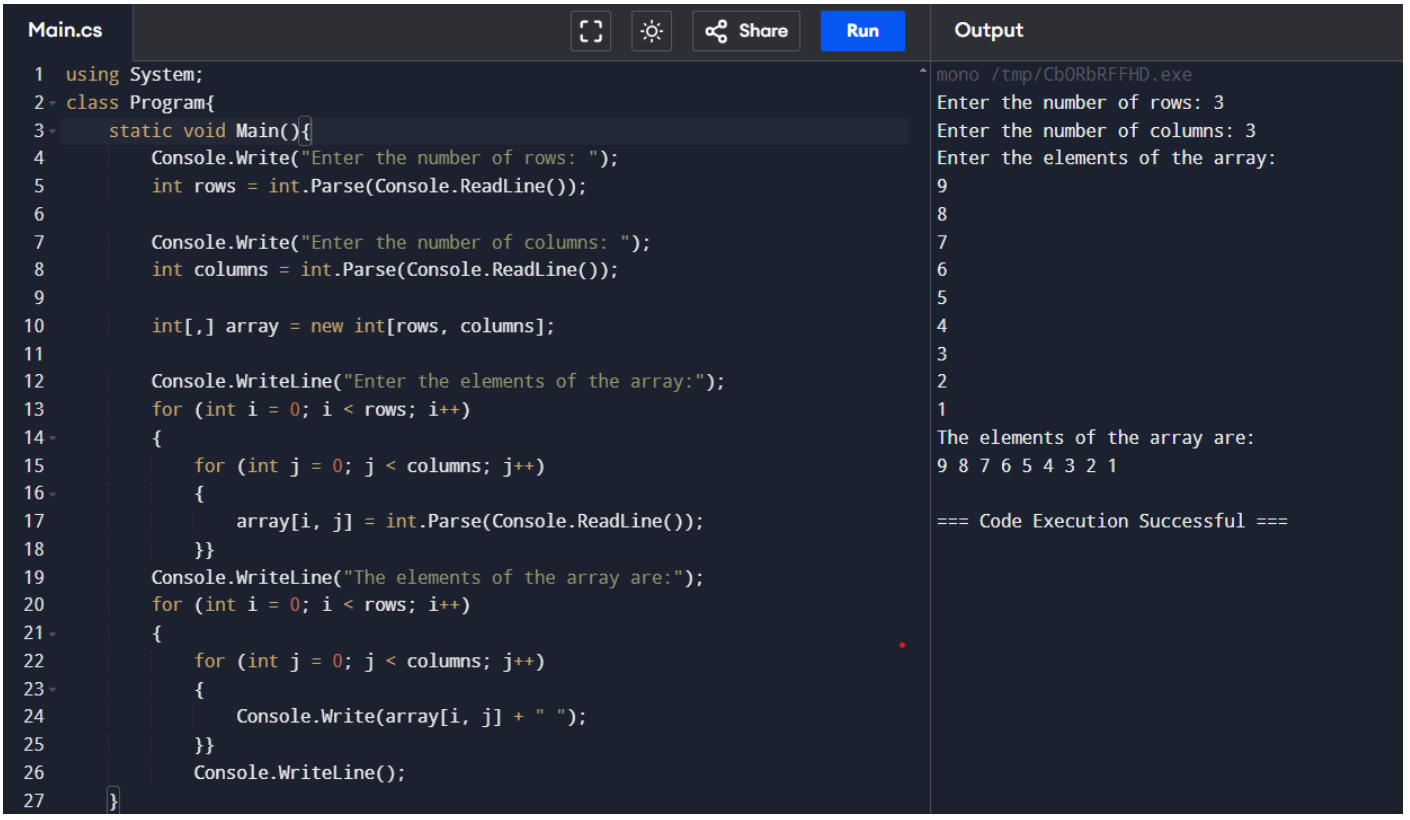
OUTPUT:

Enter number of rows: 3

Enter the number of columns: 3

Enter the element in the array: 9 8 7 6 5 4 3 2 1

The elements in the array : 9 8 7 6 5 4 3 2 1



2. Aravind wants to apply for competitive exam. He needs to know whether he is eligible to apply. The eligibility criteria is given below:

- **Age should be greater than 18 years, but not more than 30.**
- **The candidate should have passed 10 std with a minimum pass percentage of 65.**

Design the C# program to help him to know his eligibility. If the criteria gets satisfied, print he is eligible else print he is not eligible.

AIM:

To develop a C# program that checks the eligibility of Aravind for a competitive exam based on their age and 10th standard pass percentage.

PROGRAM:

```
using System;


class Program
{
   static void Main()
   {
      Console.Write("Enter Aravind's age: ");
      int age = int.Parse(Console.ReadLine());


      Console.Write("Enter Aravind's 10th standard pass percentage: ");
      double percentage = double.Parse(Console.ReadLine());


      if (age > 18 && age <= 30 && percentage >= 65)
      {
         Console.WriteLine("Aravind is eligible to apply for the exam.");
      }
```

```
        else
        {
            Console.WriteLine("Aravind is not eligible to apply for the exam.");
        }
    }
}
```

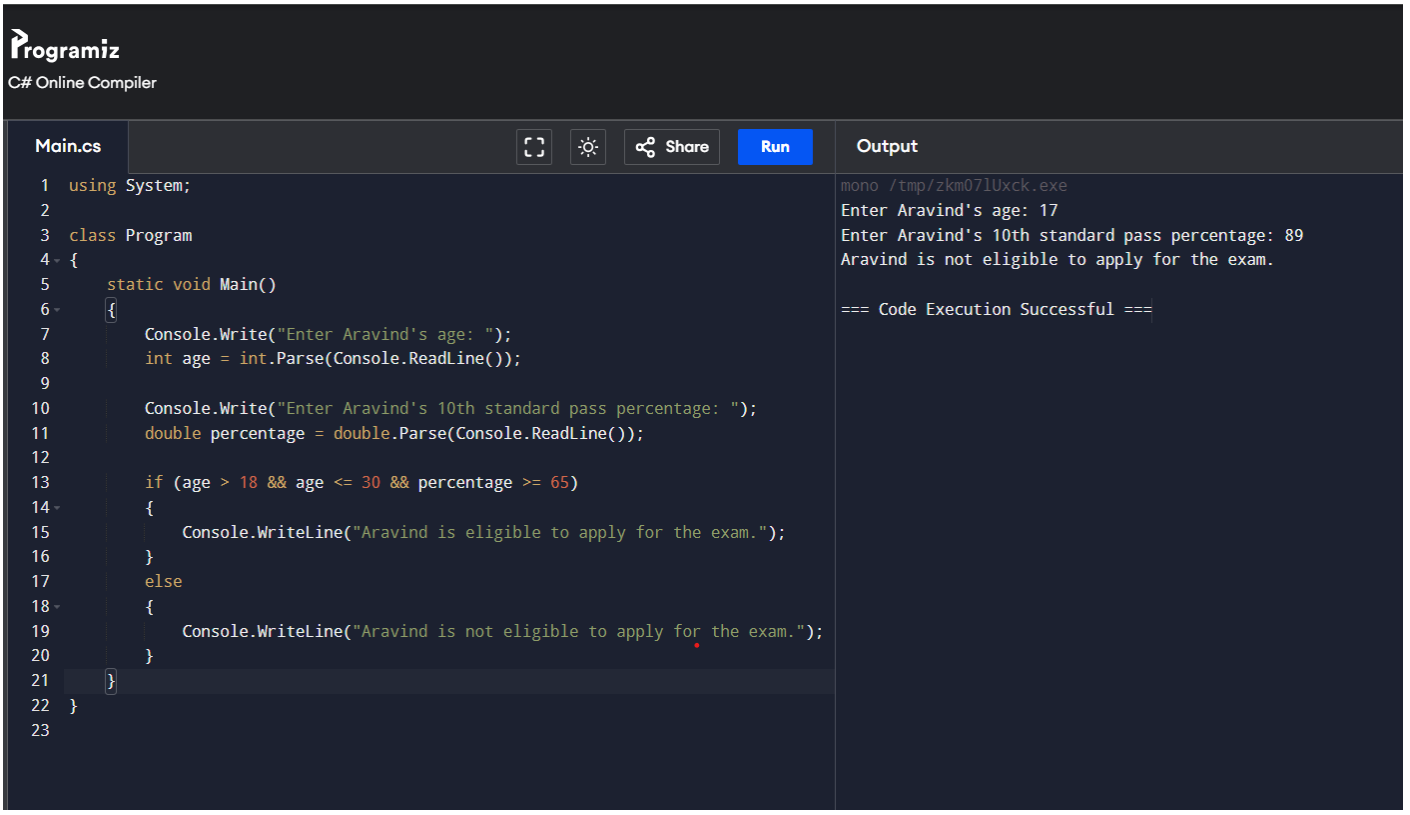
INPUT:

Enter Aravind's age: 17

Enter Aravind's 10th Standard pass percentage: 89

OUTPUT:

Aravind is not eligible to apply for the exam.



3. Design the C# console application named validation to get mobile number as input from the user. Validate the mobile number with the following cases:

- **The first four number must be followed by then followed by next six numbers(eg:9894-256874)**
- **Should contains only numbers**
- **Should be of length 10.**
- **The first number should start only with 9 Or 8.**

AIM:

To validate and print whether the given mobile number is valid or not.

PROGRAM:

```
using System;


class Validation
{
   static void Main()
   {
      Console.Write("Enter your mobile number  ");
      string mobileNumber = Console.ReadLine();


      if (IsValidMobileNumber(mobileNumber))
      {
         Console.WriteLine("Mobile number is valid.");
      }
      else
      {
         Console.WriteLine("Invalid mobile number.");
```

```
    }
}
static bool IsValidMobileNumber(string number)
{
    if (number.Length != 11)
    {
        return false;
    }

    string[] parts = number.Split('-');
    if (parts.Length != 2 || parts[0].Length != 4 || parts[1].Length != 6)
    {
        return false;
    }
    if (!IsAllDigits(parts[0]) || !IsAllDigits(parts[1]))
    {
        return false;
    }
    if (parts[0][0] != '9' && parts[0][0] != '8')
    {
        return false;
    }
    return true;
}
static bool IsAllDigits(string str){
    foreach (char c in str)
    {
        if (!char.IsDigit(c)){
```

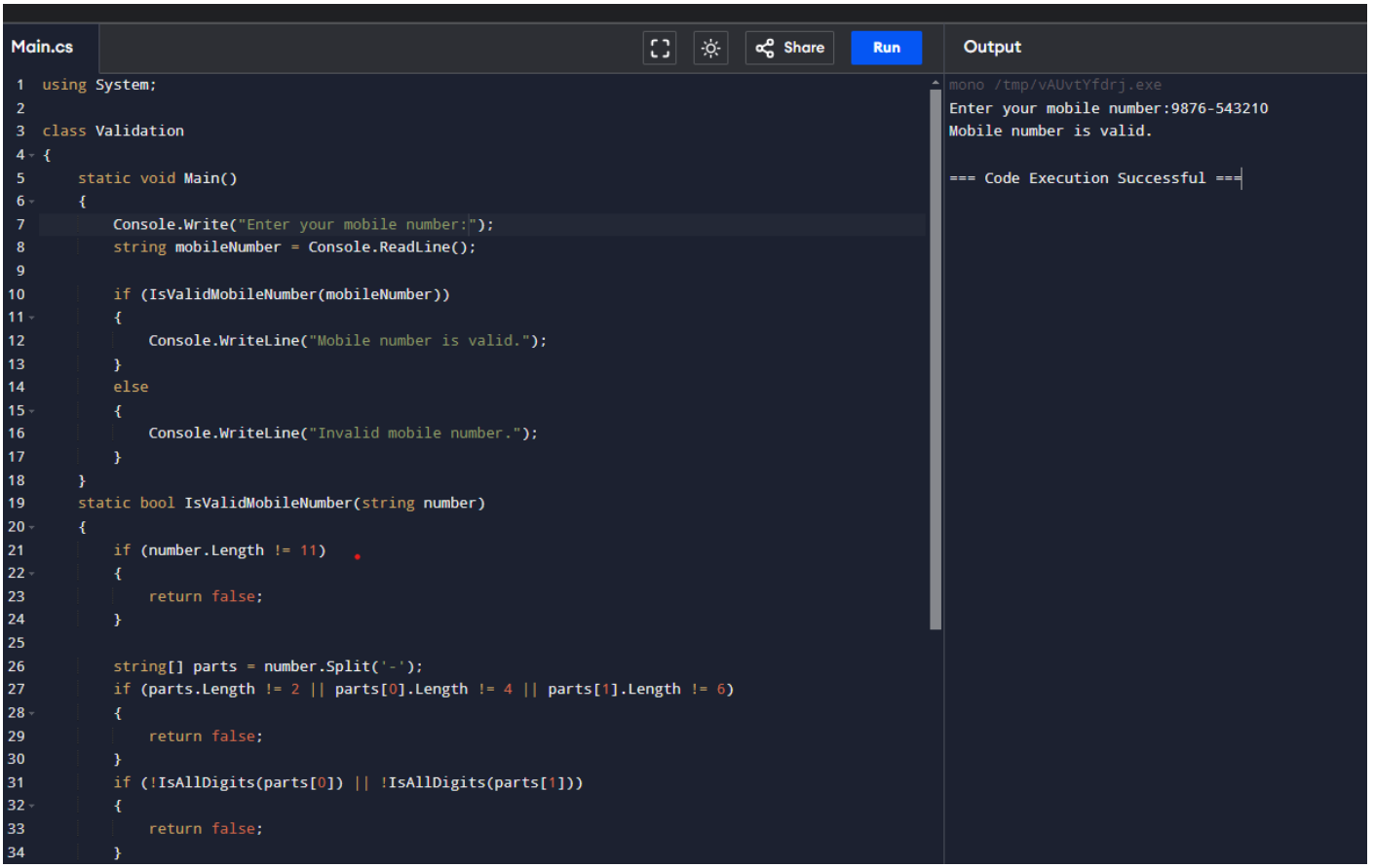
```
            return false;
        }}
        return true;
    }
}
```

INPUT:

Enter the mobile number: 9876-543210

OUTPUT:

The mobile number is valid.



4. Write the missing code snippets and the statements in the C# program given below.

```
Class person {
    ______name;
    ______age;
    ______weight;
Void printperson() {
// write the code to print name, age and weight of a person
Class persondata {
Static void Main(string[] args) {
    Person______ = _____________;
    ______.name = "Kannan";
    ______.age = 19;
    ______.weight = 58;
 // write the statement to access printperson() function}}
```

AIM:

To create a Person class, instantiate it, and print out the person's name, age, and weight using a method.

PROGRAM:

```
class Person
{
   public string name;
   public int age;
   public double weight;

   public void PrintPerson(){
```

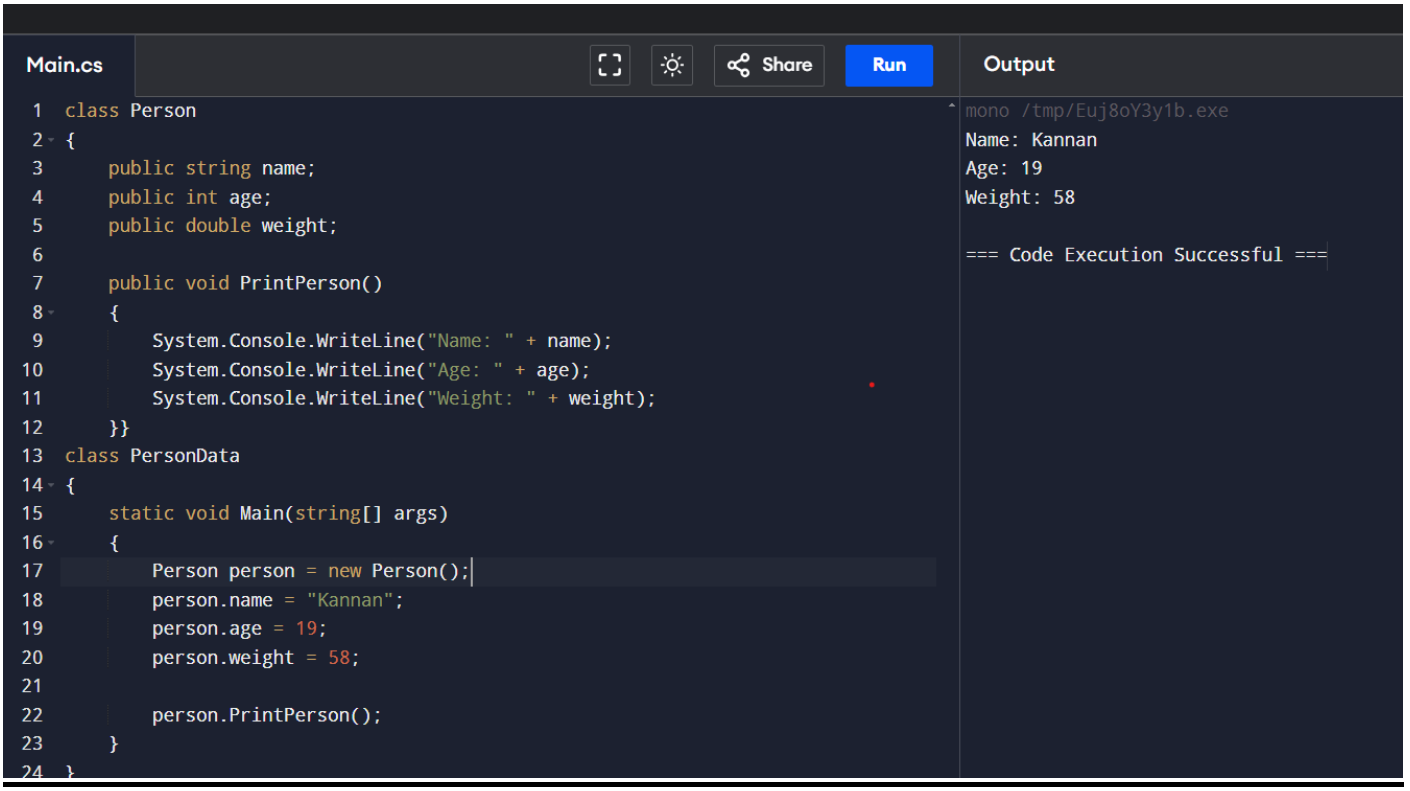
```
        System.Console.WriteLine("Name: " + name);
        System.Console.WriteLine("Age: " + age);
        System.Console.WriteLine("Weight: " + weight);
    }}
class PersonData{
    static void Main(string[] args){
        Person person = new Person();
        person.name = "Kannan";
        person.age = 19;
        person.weight = 58;
        person.PrintPerson();
    }}
```

OUTPUT:

Name: Kannan

Age: 19

Weight: 58



5. A hospital wants to create a console application to maintain its impatient details. The information to store includes:

- **Name of the patient**
- **Date of admission**
- **Age of patient**
- **Disease**
- **Date of discharge**
- **Total bills paid**

Design the C# program with the class name patient with necessary data members to store the above information. The class should have two member functions, one to get the patients information and other to display the information. Create a main

class called hospital to create necessary instances, methods calling statements and display all the details about the patient.

AIM:

To create a Patient class, collect patient information through user input, and display the collected information using methods.

PROGRAM:

```
using System;


class Patient
{
   public string Name { get; set; }
   public string DateOfAdmission { get; set; }
   public int Age { get; set; }
   public string Disease { get; set; }
   public string DateOfDischarge { get; set; }
   public double TotalBills { get; set; }
```

```
public void GetPatientInfo()
{
    Console.Write("Enter the name of the patient: ");
    Name = Console.ReadLine();
    Console.Write("Enter the date of admission (dd/mm/yyyy): ");
    DateOfAdmission = Console.ReadLine();

    Console.Write("Enter the age of the patient: ");
    Age = int.Parse(Console.ReadLine());

    Console.Write("Enter the disease of the patient: ");
    Disease = Console.ReadLine();

    Console.Write("Enter the date of discharge (dd/mm/yyyy): ");
    DateOfDischarge = Console.ReadLine();

    Console.Write("Enter the total bills paid: ");
    TotalBills = double.Parse(Console.ReadLine());
}
public void DisplayPatientInfo()
{
    Console.WriteLine("\n--- Patient Details ---");

    Console.WriteLine("Name: " + Name);
    Console.WriteLine("Date of Admission: " + DateOfAdmission);
    Console.WriteLine("Age: " + Age);
    Console.WriteLine("Disease: " + Disease);
```

```
        Console.WriteLine("Date of Discharge: " + DateOfDischarge);
        Console.WriteLine("Total Bills Paid: " + TotalBills);
    }
}
class Hospital
{
    static void Main(string[] args)
    {
        Patient patient = new Patient();

        patient.GetPatientInfo();

        patient.DisplayPatientInfo();
    }
}
```

INPUT:

Enter the name of the patient: Manish

Enter the date of admission (dd/mm/yyyy): 28/10/2022

Enter the age of the patient: 23

Enter the disease of the patient: Ulcer

Enter the date of discharge (dd/mm/yyyy): 01/12/2022

Enter the total bills paid: 350000

OUTPUT:

--- Patient Details ---

Name: Manish

Date of Admission: 28/10/2022

Age: 23

Disease: Ulcer

Date of Discharge: 01/12/2022

Total Bills Paid: 350000

Main.cs

Share Run

```
using System;

class Patient
{
    public string Name { get; set; }
    public string DateOfAdmission { get; set; }
    public int Age { get; set; }
    public string Disease { get; set; }
    public string DateOfDischarge { get; set; }
    public double TotalBills { get; set; }

    public void GetPatientInfo()
    {
        Console.Write("Enter the name of the patient: ");
        Name = Console.ReadLine();

        Console.Write("Enter the date of admission (dd/mm/yyyy): ");
        DateOfAdmission = Console.ReadLine();

        Console.Write("Enter the age of the patient: ");
        Age = int.Parse(Console.ReadLine());

        Console.Write("Enter the disease of the patient: ");
        Disease = Console.ReadLine();
```

Output

```
mono /tmp/pt6U7S6Rar.exe
Enter the name of the patient: Manish
Enter the date of admission (dd/mm/yyyy): 28/10/2022
Enter the age of the patient: 23
Enter the disease of the patient: Ulcer
Enter the date of discharge (dd/mm/yyyy): 01/12/2022
 Enter the total bills paid: 350000

--- Patient Details ---
Name: Manish
Date of Admission: 28/10/2022
Age: 23
Disease: Ulcer
Date of Discharge: 01/12/2022
Total Bills Paid: 350000

=== Code Execution Successful ===
```

6. Implement the C# code to get two vector number as input, add them and print the sum as another vector. Make use of operator overloading to perform addition of vector numbers.

AIM:

To create a Vector class, overload the '+' operator to add two vectors, and demonstrate vector addition by taking user input for two vectors and displaying their sum.

PROGRAM:

```
using System;

class Vector
{
   public int X { get; set; }
   public int Y { get; set; }

   public Vector(int x, int y)
   {
      X = x;
      Y = y;
   }
   public static Vector operator +(Vector v1, Vector v2)
   {
      return new Vector(v1.X + v2.X, v1.Y + v2.Y);
   }
   public void Display()
   {
      Console.WriteLine($"Vector: ({X}, {Y})");
   }
}
class Program
{
   static void Main(string[] args)
   {
      Console.WriteLine("Enter the first vector:");
      Console.Write("Enter X coordinate: ");
```

```
        int x1 = int.Parse(Console.ReadLine());
        Console.Write("Enter Y coordinate: ");
        int y1 = int.Parse(Console.ReadLine());

        Console.WriteLine("Enter the second vector:");
        Console.Write("Enter X coordinate: ");
        int x2 = int.Parse(Console.ReadLine());
        Console.Write("Enter Y coordinate: ");
        int y2 = int.Parse(Console.ReadLine());

        Vector v1 = new Vector(x1, y1);
        Vector v2 = new Vector(x2, y2);

        Vector sum = v1 + v2;

        Console.WriteLine("The sum of the two vectors is:");
        sum.Display();
    }
}
```

INPUT:

Enter the first vector:

Enter X coordinate: 2

Enter Y coordinate: 8

Enter the second vector:

Enter X coordinate: 1

Enter Y coordinate: 0

OUTPUT:

The sum of the two vectors is:

Vector: (3, 8)

Main.cs

Share Run

```
1  using System;
2
3  class Vector
4  {
5      public int X { get; set; }
6      public int Y { get; set; }
7
8      public Vector(int x, int y)
9      {
10         X = x;
11         Y = y;
12     }
13     public static Vector operator +(Vector v1, Vector v2)
14     {
15         return new Vector(v1.X + v2.X, v1.Y + v2.Y);
16     }
17     public void Display()
18     {
19         Console.WriteLine($"Vector: ({X}, {Y})");
20     }
21 }
22 class Program
23 {
24     static void Main(string[] args)
```

Output

```
mono /tmp/sUarKxQaVK.exe
Enter the first vector:
Enter X coordinate: 2
Enter Y coordinate: 8
Enter the second vector:
Enter X coordinate: 1
Enter Y coordinate: 0
The sum of the two vectors is:
Vector: (3, 8)

=== Code Execution Successful ===
```

7. Create the class student with necessary members to maintain the basic details of a student such as name, age, address and mobile number. Add method getDate() to read the basic details and printData() to print the details of the student. Inherit the student class into the sub class called studentmark with necessary members to maintain student mark details. Override the getData() and printData() in student mark class to read mark details and print the marks, respectively. Also, define a method to find the grade of the student based on his/her marks. Design the student main class to access the member of both the classes in C#.

AIM:

To create a Student class and a derived StudentMark class, which inherits and extends the base class to include mark details, calculates grades based on marks, and demonstrates polymorphism through overridden methods.

PROGRAM:

```
using System;

class Student
{
   protected string name;
   protected int age;
   protected string address;
   protected string mobileNumber;

   public virtual void GetData()
   {
      Console.Write("Enter student's name: ");
      name = Console.ReadLine();

      Console.Write("Enter student's age: ");
      age = Convert.ToInt32(Console.ReadLine());

      Console.Write("Enter student's address: ");
      address = Console.ReadLine();

      Console.Write("Enter student's mobile number: ");
      mobileNumber = Console.ReadLine();
   }

   public virtual void PrintData()
   {
      Console.WriteLine("\nStudent Details:");
```

```
        Console.WriteLine("Name: " + name);
        Console.WriteLine("Age: " + age);
        Console.WriteLine("Address: " + address);
        Console.WriteLine("Mobile Number: " + mobileNumber);
    }
}

class StudentMark : Student
{
    private int mark1, mark2, mark3;

    public override void GetData()
    {
        base.GetData();

        Console.Write("Enter marks for subject 1: ");
        mark1 = Convert.ToInt32(Console.ReadLine());

        Console.Write("Enter marks for subject 2: ");
        mark2 = Convert.ToInt32(Console.ReadLine());

        Console.Write("Enter marks for subject 3: ");
        mark3 = Convert.ToInt32(Console.ReadLine());
    }

    public override void PrintData()
    {
        base.PrintData();
```

```
        Console.WriteLine("\nMarks:");
        Console.WriteLine("Subject 1: " + mark1);
        Console.WriteLine("Subject 2: " + mark2);
        Console.WriteLine("Subject 3: " + mark3);
        Console.WriteLine("Grade: " + GetGrade());
    }

    public string GetGrade()
    {
        double average = (mark1 + mark2 + mark3) / 3.0;

        if (average >= 90)
            return "A";
        else if (average >= 75)
            return "B";
        else if (average >= 50)
            return "C";
        else
            return "Fail";
    }
}

class StudentMain
{
    static void Main(string[] args)
    {
        StudentMark student = new StudentMark();
```

```
        student.GetData();
        student.PrintData();
    }
}
```

INPUT:

Enter student's name: Mano

Enter student's age: 18

Enter student's address: Gandhi Street colony

Enter student's mobile number: 8976453212

Enter marks for subject 1: 98

Enter marks for subject 2: 78

Enter marks for subject 3: 95

OUTPUT:

Student Details:

Name: Mano

Age: 18

Address: Gandhi Street colony

Mobile Number: 8976453212

Marks:

Subject 1: 98

Subject 2: 78

Subject 3: 95

Grade: A

Main.cs | Share | Run | Output

```
using System;

class Student
{
    protected string name;
    protected int age;
    protected string address;
    protected string mobileNumber;

    public virtual void GetData()
    {
        Console.Write("Enter student's name: ");
        name = Console.ReadLine();

        Console.Write("Enter student's age: ");
        age = Convert.ToInt32(Console.ReadLine());

        Console.Write("Enter student's address: ");
        address = Console.ReadLine();

        Console.Write("Enter student's mobile number: ");
        mobileNumber = Console.ReadLine();
    }

    public virtual void PrintData()
    {
```

```
mono /tmp/LWOSpH7IX4.exe
Enter student's name: Mano
Enter student's age: 18
Enter student's address: Gandhi Street colony
Enter student's mobile number: 8976453212
Enter marks for subject 1: 98
Enter marks for subject 2: 78
Enter marks for subject 3: 95

Student Details:
Name: Mano
Age: 18
Address: Gandhi Street colony
Mobile Number: 8976453212

Marks:
Subject 1: 98
Subject 2: 78
Subject 3: 95
Grade: A

=== Code Execution Successful ===
```

8. Design sample C# program with class name employee to compute net salary of the employee using the basic salary, if for the job_catg is 1 use table-l else use table-II. Use constructor to initialize basic salary,hra,da,pf and loan. The employee class should contain input() method to get input for job_catg, empno, empname, calculateSalary() method to compute salary and display() method to print the details.

Table-I	Table-II
BASIC=Rs. 8,000 HRA=10% of basic DA=20% of basic LOAN=Rs. 300 PF=Rs. 500	BASIC=Rs. 15,000 HRA=20% of basic DA=30% of basic LOAN=Rs. 600 PF=1000

AIM:

To create an Employee class that calculates and displays an employee's net salary based on their job category, with salary components and deductions, and demonstrates encapsulation and methods.

PROGRAM:

```
using System;


class Employee
{
   private int empNo;
   private string empName;
   private int jobCatg;
   private double basicSalary, hra, da, pf, loan, netSalary;


   public Employee(double basic, double hraPercent, double daPercent, double
pfAmount, double loanAmount)
   {
      basicSalary = basic;
      hra = (hraPercent / 100) * basicSalary;
      da = (daPercent / 100) * basicSalary;
      pf = pfAmount;
      loan = loanAmount;
   }
   public void Input(){
      Console.Write("Enter employee number: ");
      empNo = Convert.ToInt32(Console.ReadLine());


      Console.Write("Enter employee name: ");
      empName = Console.ReadLine();


      Console.Write("Enter job category (1 or 2): ");
```

```
        jobCatg = Convert.ToInt32(Console.ReadLine());


        if (jobCatg == 1)
        {
            SetSalaryDetails(8000, 10, 20, 500, 300);
        }
        else if (jobCatg == 2){
            SetSalaryDetails(15000, 20, 30, 1000, 600);
        }else{
            Console.WriteLine("Invalid job category! Defaulting to category 1.");
            SetSalaryDetails(8000, 10, 20, 500, 300);
        }}
    private void SetSalaryDetails(double basic, double hraPercent, double
daPercent, double pfAmount, double loanAmount)
    {
        basicSalary = basic;
        hra = (hraPercent / 100) * basicSalary;
        da = (daPercent / 100) * basicSalary;
        pf = pfAmount;
        loan = loanAmount;
    }
    public void CalculateSalary(){
        netSalary = basicSalary + hra + da - (pf + loan);}
    public void Display()
    {
        Console.WriteLine("\nEmployee Details:");
        Console.WriteLine("Employee Number: " + empNo);
        Console.WriteLine("Employee Name: " + empName);
```

```
        Console.WriteLine("Job Category: " + jobCatg);
        Console.WriteLine("Basic Salary: Rs. " + basicSalary);
        Console.WriteLine("HRA: Rs. " + hra);
        Console.WriteLine("DA: Rs. " + da);
        Console.WriteLine("PF: Rs. " + pf);
        Console.WriteLine("Loan: Rs. " + loan);
        Console.WriteLine("Net Salary: Rs. " + netSalary);
    }}
class EmployeeMain{
    static void Main(string[] args){
        Employee employee = new Employee(0, 0, 0, 0, 0);

        employee.Input();
        employee.CalculateSalary();
        employee.Display();
    }}
```

INPUT:

```
Enter employee number: 123
Enter employee name: Nelson
Enter job category (1 or 2): 1
```

OUTPUT:

```
Employee Details:
Employee Number: 123
Employee Name: Nelson
Job Category: 1
Basic Salary: Rs. 8000
HRA: Rs. 800
```

DA: Rs. 1600

PF: Rs. 500

Loan: Rs. 300

Net Salary: Rs. 9600

Main.cs

Share Run

```
using System;
class Employee
{
    private int empNo;
    private string empName;
    private int jobCatg;
    private double basicSalary, hra, da, pf, loan, netSalary;

    public Employee(double basic, double hraPercent, double
        daPercent, double pfAmount, double loanAmount)
    {
        basicSalary = basic;
        hra = (hraPercent / 100) * basicSalary;
        da = (daPercent / 100) * basicSalary;
        pf = pfAmount;
        loan = loanAmount;
    }
    public void Input()
    {
        Console.Write("Enter employee number: ");
        empNo = Convert.ToInt32(Console.ReadLine());

        Console.Write("Enter employee name: ");
        empName = Console.ReadLine();

        Console.Write("Enter job category (1 or 2): ");
```

Output

```
mono /tmp/H2awXttlxA.exe
Enter employee number: 123
Enter employee name: Nelson
Enter job category (1 or 2): 1

Employee Details:
Employee Number: 123
Employee Name: Nelson
Job Category: 1
Basic Salary: Rs. 8000
HRA: Rs. 800
DA: Rs. 1600
PF: Rs. 500
Loan: Rs. 300
Net Salary: Rs. 9600

=== Code Execution Successful ===
```

Submitted By

NIVETHA BP

73772214173

III- CSE-B