C# and .NET

Assignment 1

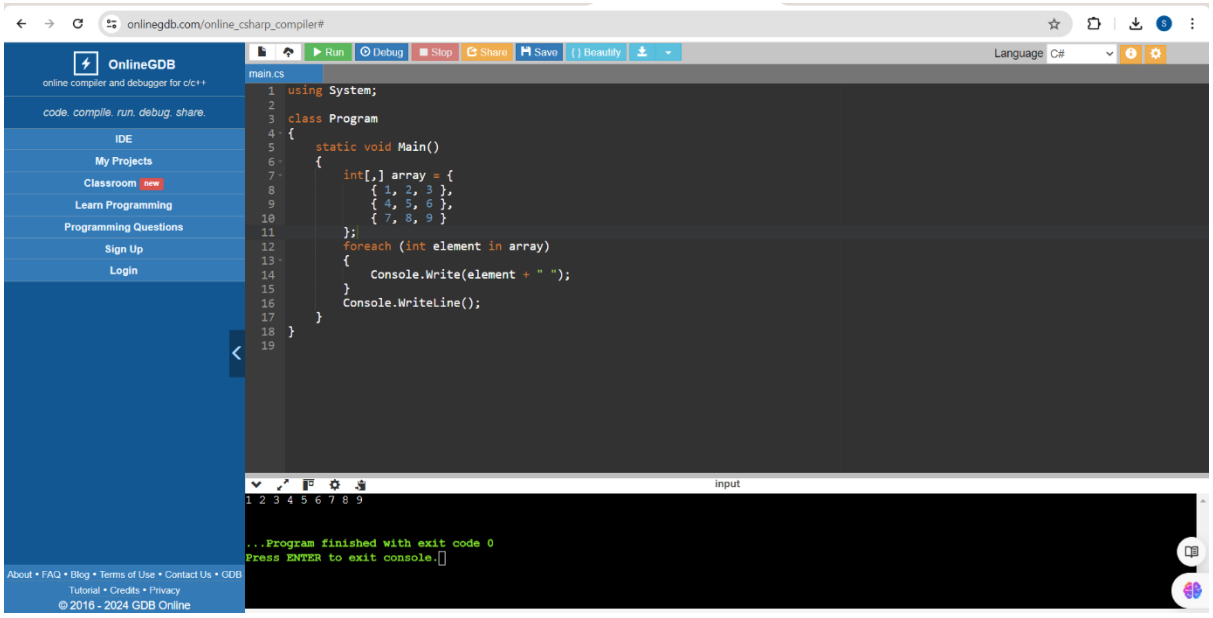
1.Develop the C# program to initialize two dimensional array and print all the elements of the array on the same line separated with space.

Aim:

To initialize and print a 2D array with line separated by space.

Program:

```
using System;
 class Program
{
  static void Main()
  {
     int[,] array = {
        { 1, 2, 3 },
        { 4, 5, 6 },
        { 7, 8, 9 }
     };
     foreach (int element in array)
     {
        Console.Write(element + " ");
     }
     Console.WriteLine();
  }
}
```



Output:

1 2 3 4 5 6 7 8 9

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.

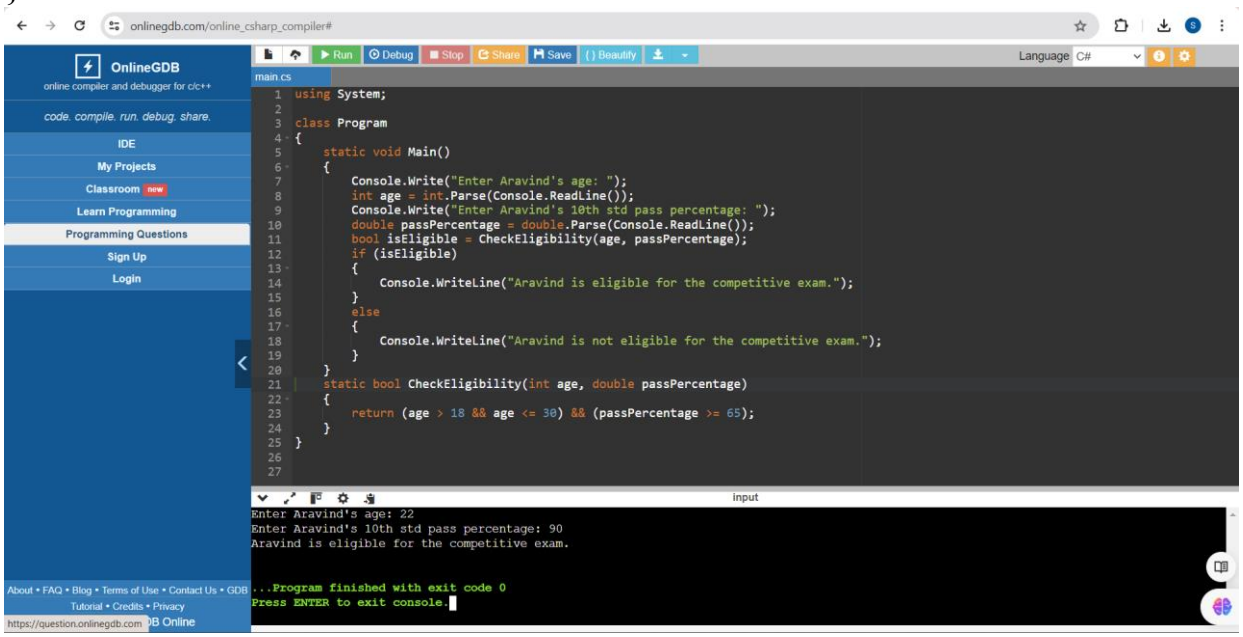
Aim:

To determine and print whether a person named Aravind is eligible to apply 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 std pass percentage: ");
        double passPercentage = double.Parse(Console.ReadLine());
        bool isEligible = CheckEligibility(age, passPercentage);
        if (isEligible)
        {
            Console.WriteLine("Aravind is eligible for the competitive exam.");
        }
        else
        {
            Console.WriteLine("Aravind is not eligible for the competitive exam.");
        }
    }
    static bool CheckEligibility(int age, double passPercentage)
    {
        return (age > 18 && age <= 30) && (passPercentage >= 65);
    }
}
```



Input:

Enter Aravind's age: 22

Enter Aravind's 10th std pass percentage: 90

Output:

Aravind is eligible for competitive 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.**

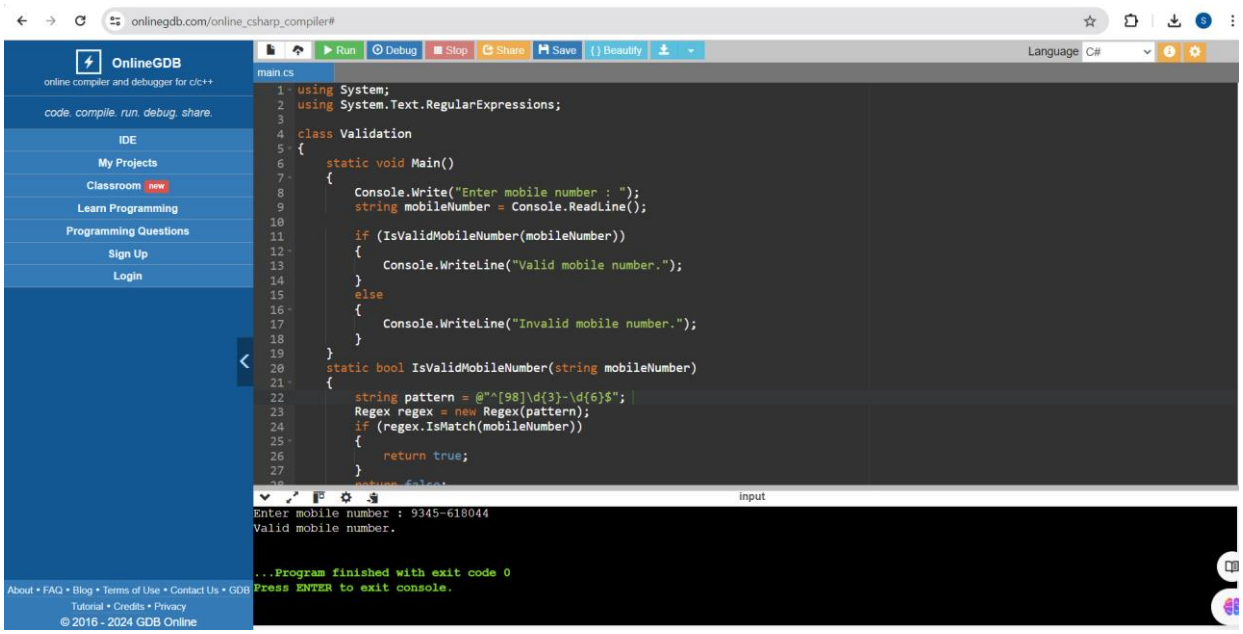
Aim:

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

Program:

```
using System;
using System.Text.RegularExpressions;


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


      if (IsValidMobileNumber(mobileNumber))
      {
         Console.WriteLine("Valid mobile number.");
```

```
        }
        else
        {
            Console.WriteLine("Invalid mobile number.");
        }
    }
    static bool IsValidMobileNumber(string mobileNumber)
    {
        string pattern = @"^[98]\d{3}-\d{6}$";
        Regex regex = new Regex(pattern);
        if (regex.IsMatch(mobileNumber))
        {
            return true;
        }
        return false;
    }
}
```



Input:

Enter mobile number : 9345-618044

Output:

Valid mobile number.

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

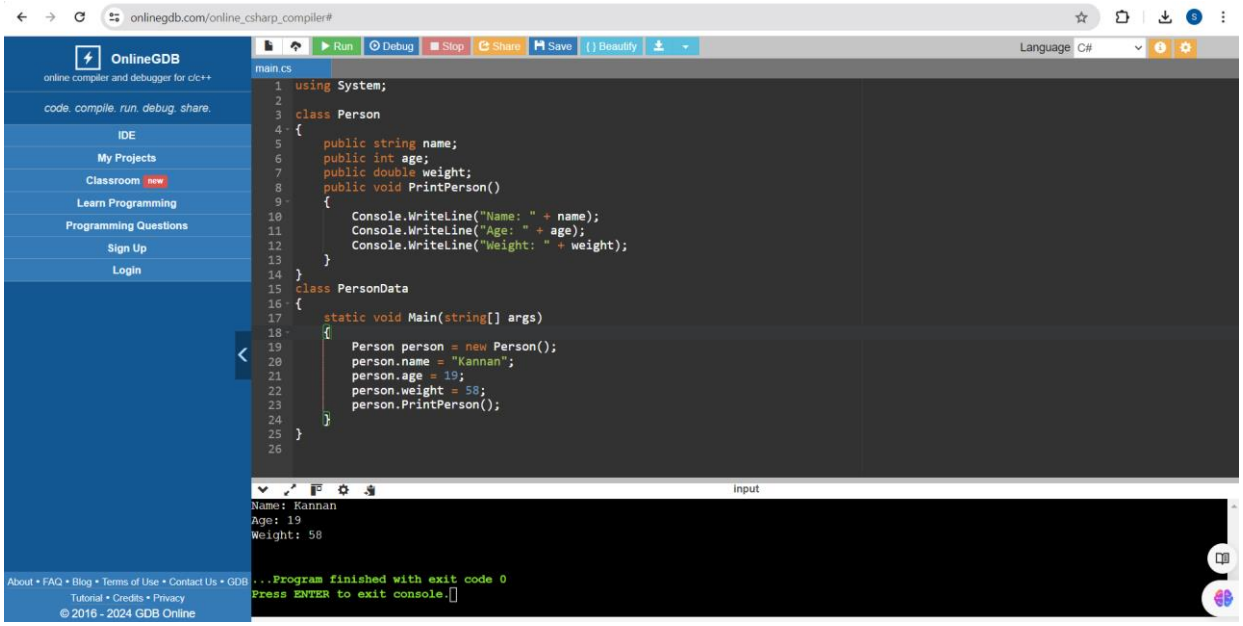
```
Class person {
_____name;
_____age;
_____weight;
Void printperson() {
// write the code to print name, age and weight of a person
}
}
Class persondata {
Static void Main(string[] args) {
person_____= _____;
_____.name = "Kannan";
_____.age = 19;
_____.weight = 58;
// write the statement to access printperson() function
}
}
```

Aim:

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

Program:

```
using System;
class Person
{
   public string name;
   public int age;
   public double weight;
   public void PrintPerson()
   {
      Console.WriteLine("Name: " + name);
      Console.WriteLine("Age: " + age);
      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.

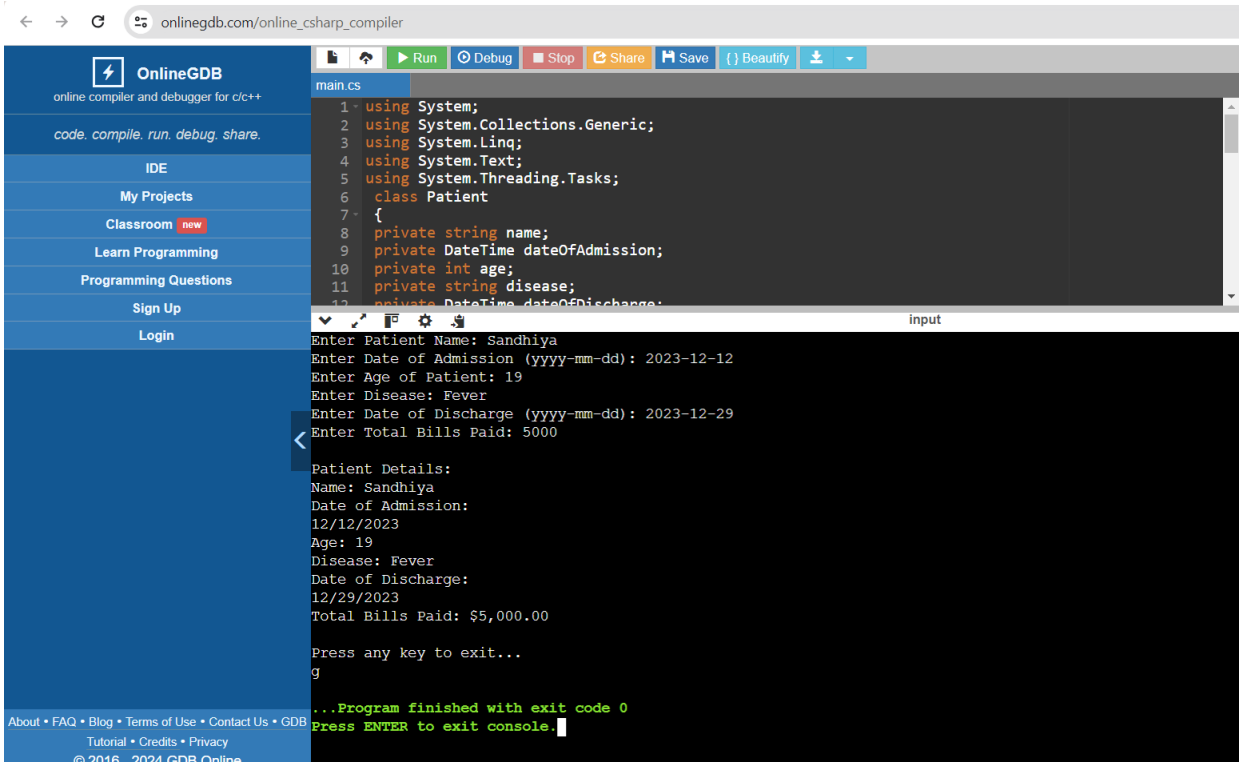
Aim:

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

Program:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
 class Patient
 {
   private string name;
   private DateTime dateOfAdmission;
   private int age;
   private string disease;
   private DateTime dateOfDischarge;
   private decimal totalBillsPaid;
   public void GetPatientInfo()
   {
      Console.Write("Enter Patient Name: ");
      name = Console.ReadLine();
      Console.Write("Enter Date of Admission (yyyy-mm-dd): ");
      dateOfAdmission = DateTime.Parse(Console.ReadLine());
      Console.Write("Enter Age of Patient: ");
      age = int.Parse(Console.ReadLine());
      Console.Write("Enter Disease: ");
      disease = Console.ReadLine();
      Console.Write("Enter Date of Discharge (yyyy-mm-dd): ");
      dateOfDischarge = DateTime.Parse(Console.ReadLine());
      Console.Write("Enter Total Bills Paid: ");
      totalBillsPaid = decimal.Parse(Console.ReadLine());
   }
```

```
    public void DisplayPatientInfo()
    {
    Console.WriteLine("\nPatient Details:");
    Console.WriteLine($"Name: {name}");
    Console.WriteLine($"Date of Admission:
    {dateOfAdmission.ToShortDateString()}");
    Console.WriteLine($"Age: {age}");
    Console.WriteLine($"Disease: {disease}");
    Console.WriteLine($"Date of Discharge:
    {dateOfDischarge.ToShortDateString()}");
    Console.WriteLine($"Total Bills Paid: {totalBillsPaid:C}");
    }
}
class Program
{
    static void Main(string[] args)
    {
    Patient patient = new Patient();
    patient.GetPatientInfo();
    patient.DisplayPatientInfo();
    Console.WriteLine("\nPress any key to exit...");
    Console.ReadKey();
    }
}
```



Input:

Enter Patient Name:Sandhiya

Enter Date of Admission (yyyy-mm-dd): 2023-12-12

Enter Age of Patient: 19

Enter Disease: Fever

Enter Date of Discharge (yyyy-mm-dd): 2023-12-29

Enter Total Bills Paid: 5000

Output:

Patient Details:

Name: Sandhiya

Date of Admission: 12/12/2023

Age: 19

Disease: Fever

Date of Discharge: 29/12/2023

Total Bills Paid: $5000

Press any key to exit…

g

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.

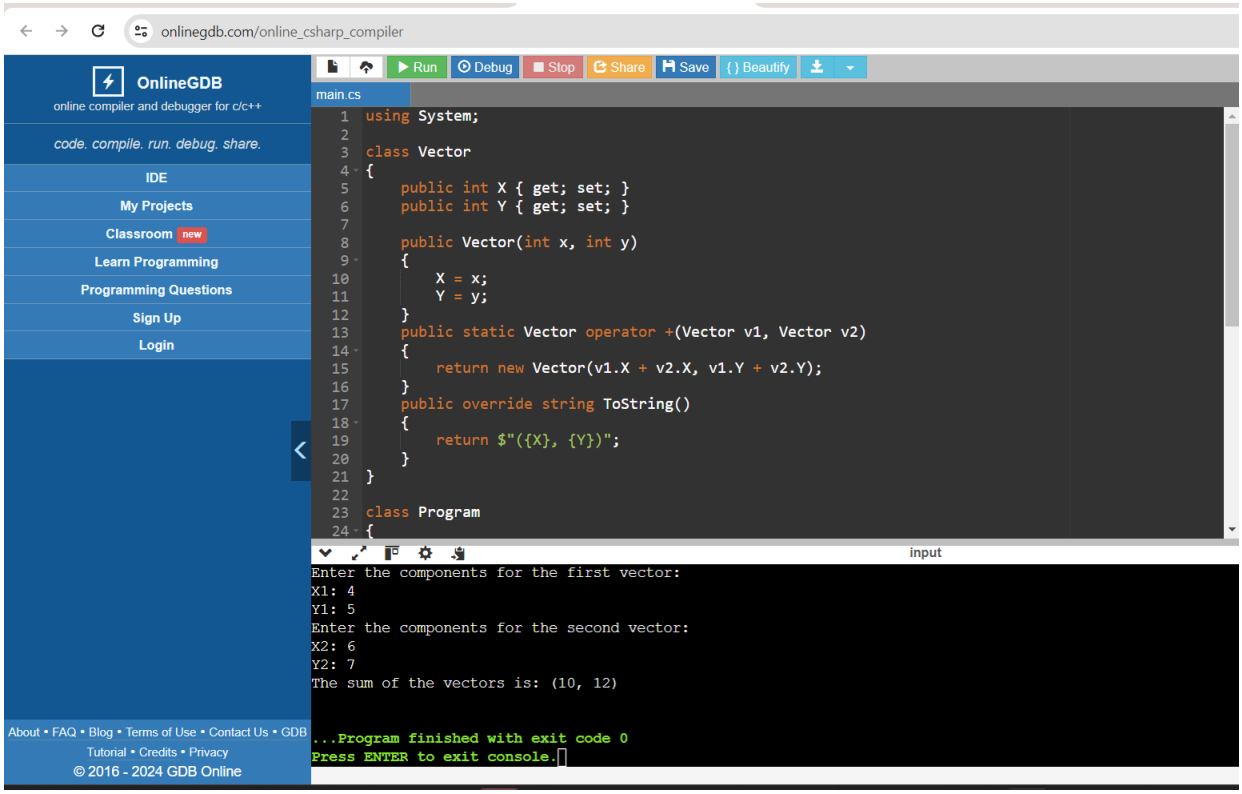
Aim:

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 override string ToString()
   {
      return $"({X}, {Y})";
```

```
}
}
class Program
{
    static void Main()
    {
        Console.WriteLine("Enter the components for the first vector:");
        Console.Write("X1: ");
        int x1 = int.Parse(Console.ReadLine());
        Console.Write("Y1: ");
        int y1 = int.Parse(Console.ReadLine());
        Console.WriteLine("Enter the components for the second vector:");
        Console.Write("X2: ");
        int x2 = int.Parse(Console.ReadLine());
        Console.Write("Y2: ");
        int y2 = int.Parse(Console.ReadLine());

        Vector v1 = new Vector(x1, y1);
        Vector v2 = new Vector(x2, y2);
        Vector result = v1 + v2;
        Console.WriteLine("The sum of the vectors is: " + result);
    }
}
```



Input:

Enter the components for the first vector:

X1:4

Y2:5

Enter the components for the second vector:

X2:6

Y2:7

Output:

The sum of the vectors is: (10, 12)

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:

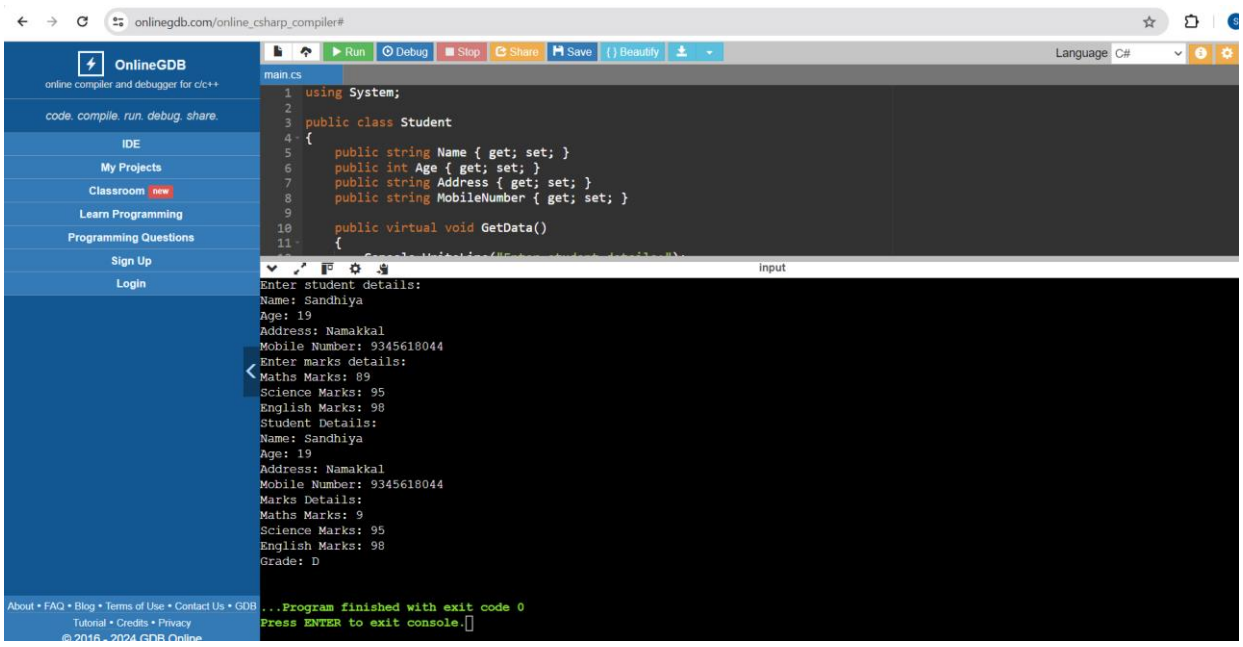
To create a Student class and a derived StudentMark class, which inherits and extends the base class to include mark details, calculates grades based on marks, and demonstrates polymorphism through overridden methods.

Program:

```
using System;
public class Student
{
    public string Name { get; set; }
    public int Age { get; set; }
    public string Address { get; set; }
    public string MobileNumber { get; set; }
    public virtual void GetData()
    {
        Console.WriteLine("Enter student details:");
        Console.Write("Name: ");
        Name = Console.ReadLine();
        Console.Write("Age: ");
        Age = int.Parse(Console.ReadLine());
        Console.Write("Address: ");
        Address = Console.ReadLine();
        Console.Write("Mobile Number: ");
        MobileNumber = Console.ReadLine();
    }
    public virtual void PrintData()
    {
        Console.WriteLine("Student Details:");
        Console.WriteLine($"Name: {Name}");
```

```
        Console.WriteLine($"Age: {Age}");
        Console.WriteLine($"Address: {Address}");
        Console.WriteLine($"Mobile Number: {MobileNumber}");
    }
}
public class StudentMark : Student
{
    public int MathsMarks { get; set; }
    public int ScienceMarks { get; set; }
    public int EnglishMarks { get; set; }
    public override void GetData()
    {
        base.GetData();
        Console.WriteLine("Enter marks details:");
        Console.Write("Maths Marks: ");
        MathsMarks = int.Parse(Console.ReadLine());
        Console.Write("Science Marks: ");
        ScienceMarks = int.Parse(Console.ReadLine());
        Console.Write("English Marks: ");
        EnglishMarks = int.Parse(Console.ReadLine());
    }
    public override void PrintData()
    {
        base.PrintData();
        Console.WriteLine("Marks Details:");
        Console.WriteLine($"Maths Marks: {MathsMarks}");
        Console.WriteLine($"Science Marks: {ScienceMarks}");
        Console.WriteLine($"English Marks: {EnglishMarks}");
        Console.WriteLine($"Grade: {FindGrade()}");
```

```
}
    public string FindGrade()
    {
        double average = (MathsMarks + ScienceMarks + EnglishMarks) / 3.0;
        if (average >= 90) return "A";
        else if (average >= 80) return "B";
        else if (average >= 70) return "C";
        else if (average >= 60) return "D";
        else return "F";
    }
}
class Program
{
    static void Main()
    {
        StudentMark student = new StudentMark();
        student.GetData();
        student.PrintData();
    }
}
```



Input:

Enter Student details:
Name: Sandhiya
Age: 19
Address: Namakkal
Number: 9345618044
Maths Marks: 89
Science Marks:95
English Marks:98

Output:

Enter Student details:
Name: Sandhiya
Age: 19
Address: Namakkal
Number: 9345618044
Maths Marks: 89
Science Marks:95
English Marks:98
Grade: D

8. Design sample C# program with class name employee to compute netsalary 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.

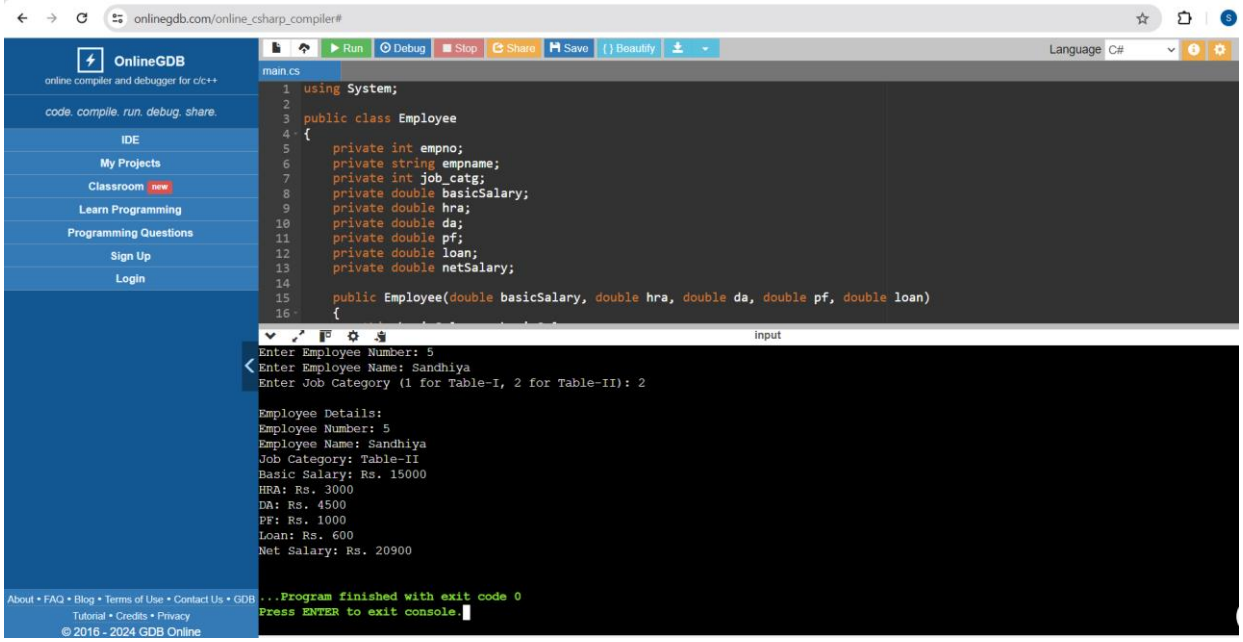
Program:

```
using System;
public class Employee
{
   private int empno;
   private string empname;
   private int job_catg;
   private double basicSalary;
   private double hra;
   private double da;
   private double pf;
   private double loan;
   private double netSalary;

   public Employee(double basicSalary, double hra, double da, double pf, double loan)
   {
      this.basicSalary = basicSalary;
      this.hra = hra;
      this.da = da;
      this.pf = pf;
      this.loan = loan;
   }
   public void Input()
   {
      Console.Write("Enter Employee Number: ");
      empno = int.Parse(Console.ReadLine());
```

```
Console.Write("Enter Employee Name: ");
empname = Console.ReadLine();
Console.Write("Enter Job Category (1 for Table-I, 2 for Table-II): ");
job_catg = int.Parse(Console.ReadLine());
if (job_catg == 1)
{
   // Table-I
   basicSalary = 8000;
   hra = 0.10 * basicSalary;
   da = 0.20 * basicSalary;
   loan = 300;
   pf = 500;
}
else if (job_catg == 2)
{
   basicSalary = 15000;
   hra = 0.20 * basicSalary;
   da = 0.30 * basicSalary;
   loan = 600;
   pf = 1000;
}
else
{
   Console.WriteLine("Invalid Job Category. Setting default values.");
   basicSalary = 0;
   hra = 0;
   da = 0;
   loan = 0;
   pf = 0;
```

```
}
        CalculateSalary();
    }
    private 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($"Job Category: {(job_catg == 1 ? "Table-I" : "Table-
II")}");
        Console.WriteLine($"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 Program
{
    static void Main()
    {
        Employee emp = new Employee(0, 0, 0, 0, 0);


        emp.Input();
```

```
        emp.Display();
    }
}
```



Input:

Enter Employee Number: 5
Enter Employee Name: Sandhiya
Enter Job Category (1 for Table-I, 2 for Table-II): 2

Output:

Employee Details:
Employee Number: 5
Enter Employee Name: Sandhiya
Job Category: 2
Basic Salary: Rs. 15000
HRA: Rs. 3000
DA: Rs. 4500
PF Deduction: Rs. 1000
Loan: Rs. 600
Net Salary: Rs. 20900

BY:

SANDHIYA R

73772214191

III – B.E CSE