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 initialize two dimensional array containing numbers 1 to 9.

PROGRAM :

```
using System;

class Program

{

   static void Main()

   {

      int[,] array = new int[,] { { 1, 2, 3 }, { 4, 5, 6 }, { 7, 8, 9 } };

      for (int i = 0; i < array.GetLength(0); i++)

      {

         for (int j = 0; j < array.GetLength(1); j++)

         {

            Console.Write(array[i, j] + " ");

         }

      }

   }

}
```

Main.cs

Run

Output

Clear

```
using System;
class Program
{
    static void Main()
    {
        int[,] array = new int[,] { { 1, 2, 3 }
            , { 4, 5, 6 }, { 7, 8, 9 } };
        for (int i = 0; i < array.GetLength(0);
            i++)
        {
            for (int j = 0; j < array.GetLength
                (1); j++)
            {
                Console.Write(array[i, j] + " "
                    );
            }
        }
    }
}
```

```
mono /tmp/fWa30j1c7u.exe
1 2 3 4 5 6 7 8 9
=== Code Execution Successful ===
```

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 your age: ");

      int age = Convert.ToInt32(Console.ReadLine());
```

```
        Console.Write("Enter your 10th standard percentage: ");
        double percentage = Convert.ToDouble(Console.ReadLine());
        if (age > 18 && age <= 30 && percentage >= 65)
        {
            Console.WriteLine("You are eligible to apply.");
        }
        else
        {
            Console.WriteLine("You are not eligible to apply.");
        }
    }
}
```

Main.cs | Run | Output | Clear

```
using System;
class Program
{
    static void Main()
    {
        Console.Write("Enter your age: ");
        int age = Convert.ToInt32(Console
            .ReadLine());
        Console.Write("Enter your 10th standard
            percentage: ");
        double percentage = Convert.ToDouble
            (Console.ReadLine());
        if (age > 18 && age <= 30 && percentage
            >= 65)
        {
            Console.WriteLine("You are eligible
                to apply.");
        }
        else
        {
            Console.WriteLine("You are not
                eligible to apply.");
        }
    }
}
```

```
mono /tmp/TK7UXAnR3l.exe
Enter your age: 20
Enter your 10th standard percentage: 54.3
You are not eligible to apply.

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

INPUT :

Enter your age : 20

Enter your 10th standard percentage : 54.3

OUTPUT :

Your are not eligible to apply.

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;
namespace Validation
{
  class Program
  {
    static void Main(string[] args)
    {
      Console.Write("Enter your mobile number (xxxxx-xxxxxx): ");
      string mobileNumber = Console.ReadLine();
      mobileNumber = mobileNumber.Replace("-", "");
      if (ValidateMobileNumber(mobileNumber))
      {
        Console.WriteLine("Valid mobile number.");
      }
      else
      {
        Console.WriteLine("Invalid mobile number.");
      }
}
static bool ValidateMobileNumber(string mobileNumber)
{
  if (mobileNumber.Length != 10)
    return false;
  if (mobileNumber[0] != '8' && mobileNumber[0] != '9')
    return false;
  foreach (char c in mobileNumber)
```

```
        {
            if (!Char.IsDigit(c))
                return false;
        }

        return true;
    }
  }
}
```

Main.cs

Run

Output

Clear

```
using System;
using System.Text.RegularExpressions;
namespace Validation
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.Write("Enter your mobile
                number (xxxxx-xxxxxx): ");
            string mobileNumber = Console
                .ReadLine();
            mobileNumber = mobileNumber.Replace
                ("-", "");
            if (ValidateMobileNumber
                (mobileNumber))
            {
                Console.WriteLine("Valid mobile
                    number.");
            }
            else
            {
                Console.WriteLine("Invalid
                    mobile number.");
            }
        }
        static bool ValidateMobileNumber(string
            mobileNumber)
        {
            if (mobileNumber.Length != 10)
                return false;
            if (mobileNumber[0] != '8' &&
                mobileNumber[0] != '9')
                return false;
            foreach (char c in mobileNumber)
            {
                if (!Char.IsDigit(c))
                    return false;
            }

            return true;
        }
    }
}
```

```
mono /tmp/1n1QjT7Wry.exe
Enter your mobile number (xxxxx-xxxxxx): 9626012863
Valid mobile number.

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

INPUT :

Enter your mobile number (xxxxx-xxxxx) : 9626012863

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

}

}

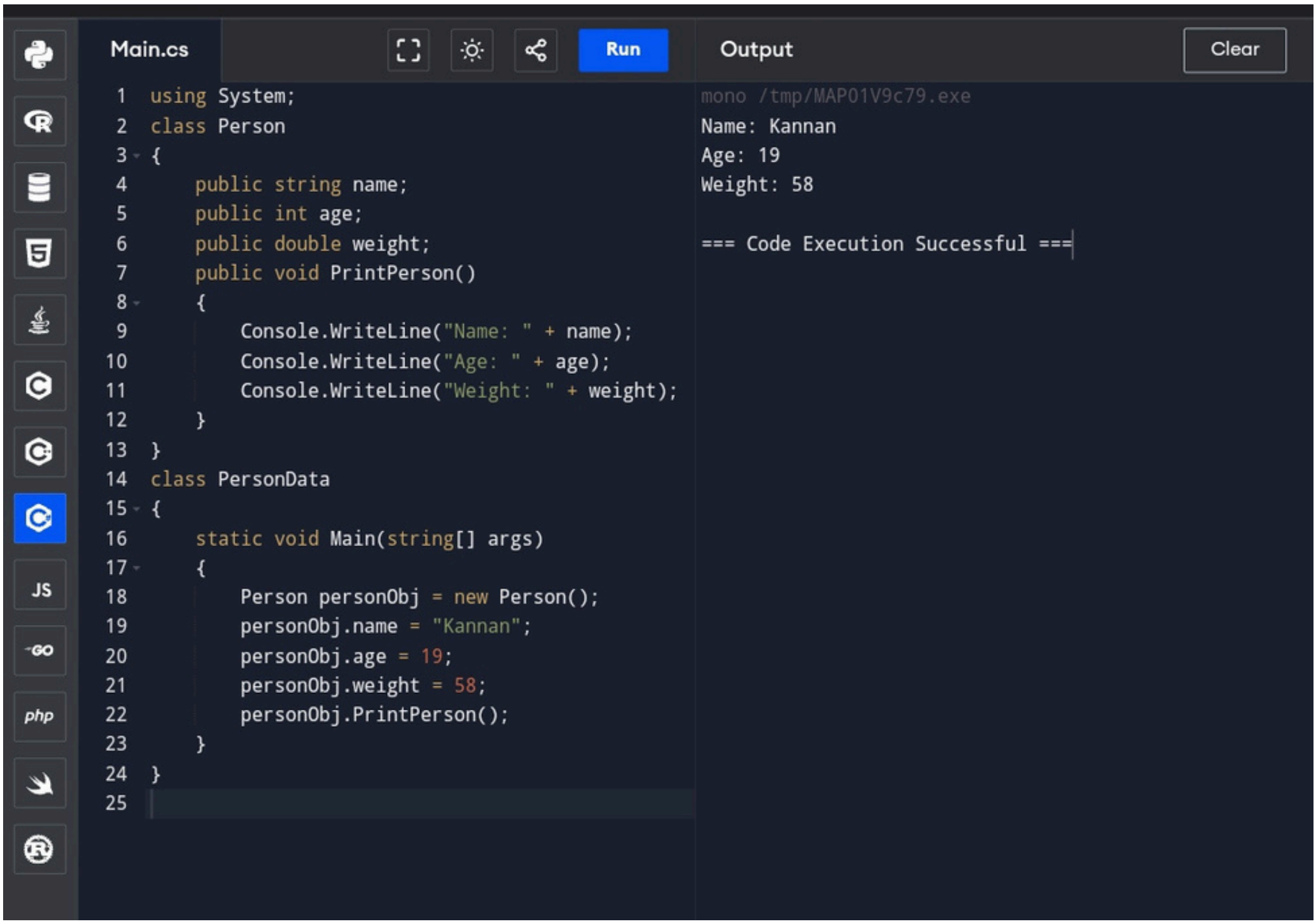
AIM :

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 personObj = new Person();
    personObj.name = "Kannan";
    personObj.age = 19;
    personObj.weight = 58;
    personObj.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;

    public string dateOfAdmission;

    public int age;

    public string disease;

    public string dateOfDischarge;

    public double totalBillsPaid;

    public void GetPatientInfo()

    {

        Console.Write("Enter patient's name: ");

        name = Console.ReadLine();

        Console.Write("Enter date of admission (dd/MM/yyyy): ");

        dateOfAdmission = Console.ReadLine();

        Console.Write("Enter patient's age: ");

        age = Convert.ToInt32(Console.ReadLine());

        Console.Write("Enter disease: ");

        disease = Console.ReadLine();

        Console.Write("Enter date of discharge (dd/MM/yyyy): ");

        dateOfDischarge = Console.ReadLine();

        Console.Write("Enter total bills paid: ");

        totalBillsPaid = Convert.ToDouble(Console.ReadLine());

    }

    public void DisplayPatientInfo()

    {

        Console.WriteLine("\nPatient Information:");

        Console.WriteLine("----------------------");

        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: " + totalBillsPaid);
    }
}
class Hospital
{
    static void Main(string[] args)
    {
        Patient patient1 = new Patient();
        patient1.GetPatientInfo();
        patient1.DisplayPatientInfo();
        Console.ReadKey();
    }
}
```

Main.cs

Run

Output

Clear

```
using System;
class Patient
{
    public string name;
    public string dateOfAdmission;
    public int age;
    public string disease;
    public string dateOfDischarge;
    public double totalBillsPaid;
    public void GetPatientInfo()
    {
        Console.Write("Enter patient's name: "
            );
        name = Console.ReadLine();
        Console.Write("Enter date of admission
            (dd/MM/yyyy): ");
        dateOfAdmission = Console.ReadLine();
        Console.Write("Enter patient's age: ");
        age = Convert.ToInt32(Console.ReadLine
            ());
        Console.Write("Enter disease: ");
        disease = Console.ReadLine();
        Console.Write("Enter date of discharge
            (dd/MM/yyyy): ");
        dateOfDischarge = Console.ReadLine();
        Console.Write("Enter total bills paid:
            ");
        totalBillsPaid = Convert.ToDouble
            (Console.ReadLine());
    }
    public void DisplayPatientInfo()
    {
        Console.WriteLine("\nPatient
            Information:");
        Console.WriteLine("
            ----------------------");
        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: "
            + totalBillsPaid);
    }
}
class Hospital
{
    static void Main(string[] args)
    {
        Patient patient1 = new Patient();
        patient1.GetPatientInfo();
        patient1.DisplayPatientInfo();
        Console.ReadKey();
    }
```

```
mono /tmp/JnnWFwsCv6.exe
Enter patient's name: Sri Vishruthi
Enter date of admission (dd/MM/yyyy): 14/09/2024
Enter patient's age: 19
Enter disease: Fever
Enter date of discharge (dd/MM/yyyy): 16/09/2024
Enter total bills paid: 2000

Patient Information:
----------------------
Name: Sri Vishruthi
Date of Admission: 14/09/2024
Age: 19
Disease: Fever
Date of Discharge: 16/09/2024
Total Bills Paid: 2000
```

```
46  }
47
```

INPUT :

Enter patient's name : Sri Vishruthi

Enter date of admission (dd/mm/yyyy) : 14/09/2024

Enter patient's age : 20

Enter disease : Fever

Enter date of discharge (dd/mm/yyyy) : 16/09/2024

Enter total bills paid : 2000

OUTPUT :

Patient Information :

Name : Sri Vishruthi

Date of admission : 14/09/2024

Age : 19

Disease : Fever

Date of discharge : 16/09/2024

Total bills paid : 2000

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 double x;

   public double y;

   public Vector(double x, double y)

   {

      this.x = x;

      this.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.Write("Enter first vector (x y): ");
        string[] input1 = Console.ReadLine().Split(' ');
        Vector v1 = new Vector(double.Parse(input1[0]), double.Parse(input1[1]));
        Console.Write("Enter second vector (x y): ");
        string[] input2 = Console.ReadLine().Split(' ');
        Vector v2 = new Vector(double.Parse(input2[0]), double.Parse(input2[1]));
        Vector sum = v1 + v2;
        Console.WriteLine("Sum: " + sum);
    }
}
```

Main.cs

Run

Output

Clear

```
using System;
class Vector
{
    public double x;
    public double y;
    public Vector(double x, double y)
    {
        this.x = x;
        this.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.Write("Enter first vector (x y
            ): ");
        string[] input1 = Console.ReadLine
            ().Split(' ');
        Vector v1 = new Vector(double.Parse
            (input1[0]), double.Parse(input1[1]
            ));
        Console.Write("Enter second vector (x y
            ): ");
        string[] input2 = Console.ReadLine
            ().Split(' ');
        Vector v2 = new Vector(double.Parse
            (input2[0]), double.Parse(input2[1]
            ));
        Vector sum = v1 + v2;
        Console.WriteLine("Sum: " + sum);
    }
}
```

```
mono /tmp/NBckLV3PZs.exe
Enter first vector (x y): 2 3
Enter second vector (x y): 4 5
Sum: (6, 8)

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

INPUT :

Enter first vector (x y): 2 3

Enter second vector (x y): 4 5

OUTPUT :

Sum: (6, 8)

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

{

   public string name;

   public int age;

   public string address;

   public string mobileNumber;

   public virtual void GetData()

   {

      Console.Write("Enter name: ");

      name = Console.ReadLine();

      Console.Write("Enter age: ");

      age = Convert.ToInt32(Console.ReadLine());

      Console.Write("Enter address: ");

      address = Console.ReadLine();

      Console.Write("Enter mobile number: ");

      mobileNumber = Console.ReadLine();

   }

   public virtual void PrintData()

   {

      Console.WriteLine("Name: " + name);

      Console.WriteLine("Age: " + age);

      Console.WriteLine("Address: " + address);

      Console.WriteLine("Mobile Number: " + mobileNumber);

   }

}

class StudentMark : Student

{

   public int marks;

   public override void GetData()

   {
```

```
    base.GetData();
    Console.Write("Enter marks: ");
    marks = Convert.ToInt32(Console.ReadLine());
}
public override void PrintData()
{
    base.PrintData();
    Console.WriteLine("Marks: " + marks);
    Console.WriteLine("Grade: " + GetGrade());
}
public string GetGrade()
{
      if (marks >= 90)
         return "A";
      else if (marks >= 80)
         return "B";
      else if (marks >= 70)
         return "C";
      else if (marks >= 60)
         return "D";
      else
         return "F";
   }
}
class StudentMain
{
   static void Main()
   {
      StudentMark student = new StudentMark();
      student.GetData();
      student.PrintData();
   }
}
```

Main.cs

Run

Output

Clear

```
using System;
class Student
{
    public string name;
    public int age;
    public string address;
    public string mobileNumber;
    public virtual void GetData()
    {
        Console.Write("Enter name: ");
        name = Console.ReadLine();
        Console.Write("Enter age: ");
        age = Convert.ToInt32(Console.ReadLine
            ());
        Console.Write("Enter address: ");
        address = Console.ReadLine();
        Console.Write("Enter mobile number: ");
        mobileNumber = Console.ReadLine();
    }
    public virtual void PrintData()
    {
        Console.WriteLine("Name: " + name);
        Console.WriteLine("Age: " + age);
        Console.WriteLine("Address: " + address
            );
        Console.WriteLine("Mobile Number: " +
            mobileNumber);
    }
}
class StudentMark : Student
{
    public int marks;
    public override void GetData()
    {
        base.GetData();
        Console.Write("Enter marks: ");
        marks = Convert.ToInt32(Console
            .ReadLine());
    }
    public override void PrintData()
    {
        base.PrintData();
        Console.WriteLine("Marks: " + marks);
        Console.WriteLine("Grade: " + GetGrade
            ());
    }
    public string GetGrade()
    {
        if (marks >= 90)
            return "A";
        else if (marks >= 80)
            return "B";
        else if (marks >= 70)
            return "C";
        else if (marks >= 60)
            return "D";
        else
```

```
mono /tmp/jCm3eE1ptB.exe
Enter name: Sri Vishruthi
Enter age: 19
Enter address: Chithode
Enter mobile number: 9626012863
Enter marks: 80
Name: Sri Vishruthi
Age: 19
Address: Chithode
Mobile Number: 9626012863
Marks: 80
Grade: B

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

```
            return "F";
        }
    }
    class StudentMain
    {
        static void Main()
        {
            StudentMark student = new StudentMark
                ();
            student.GetData();
            student.PrintData();
        }
    }
```

INPUT :

Enter name: Sri Vishruthi

Enter age: 19

Enter address: Chithode

Enter mobile number: 9626012863

Enter marks: 80

OUTPUT :

Name: Sri Vishruthi

Age: 19

Address: Chithode

Mobile Number: 9626012863

Marks: 80

Grade: B

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.

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;
private double hra;
private double da;
private double pf;
private double loan;
private double netSalary;
public Employee(int jobCatg)
{
   if (jobCatg == 1)
   {
      basicSalary = 8000;
      hra = basicSalary * 0.10;
      da = basicSalary * 0.20;
      pf = 500;
      loan = 300;
   }
   else if (jobCatg == 2)
   {
      basicSalary = 15000;
      hra = basicSalary * 0.20;
      da = basicSalary * 0.30;
      pf = 1000;
      loan = 600;
   }
}
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());
}
public void CalculateSalary()
{
   netSalary = basicSalary + hra + da - pf - loan;
```

```
}
public void Display()
{
    Console.WriteLine("Employee Number: " + empNo);
    Console.WriteLine("Employee Name: " + empName);
    Console.WriteLine("Job Category: " + jobCatg);
    Console.WriteLine("Basic Salary: " + basicSalary);
    Console.WriteLine("HRA: " + hra);
    Console.WriteLine("DA: " + da);
    Console.WriteLine("PF: " + pf);
    Console.WriteLine("Loan: " + loan);
    Console.WriteLine("Net Salary: " + netSalary);
    }
}
class Program
{
    static void Main()
    {
        Employee emp = new Employee(0);
        emp.Input();
        emp.CalculateSalary();
        emp.Display();
    }
}
```

Main.cs

Run

Output

Clear

```
using System;
class Employee
{
    private int empNo;
    private string empName;
    private int jobCatg;
    private double basicSalary;
    private double hra;
    private double da;
    private double pf;
    private double loan;
    private double netSalary;
    public Employee(int jobCatg)
    {
        if (jobCatg == 1)
        {
            basicSalary = 8000;
            hra = basicSalary * 0.10;
            da = basicSalary * 0.20;
            pf = 500;
            loan = 300;
        }
        else if (jobCatg == 2)
        {
            basicSalary = 15000;
            hra = basicSalary * 0.20;
            da = basicSalary * 0.30;
            pf = 1000;
            loan = 600;
        }
    }
    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());
    }
    public void CalculateSalary()
    {
        netSalary = basicSalary + hra + da - pf
            - loan;
    }
    public void Display()
    {
        Console.WriteLine("Employee Number: " +
            empNo);
        Console.WriteLine("Employee Name: " +
            empName);
        Console.WriteLine("Job Category: " +
            jobCatg);
```

```
mono /tmp/LBY5OWuh4A.exe
Enter employee number: 102
Enter employee name: John Doe
Enter job category (1 or 2): 2
Employee Number: 102
Employee Name: John Doe
Job Category: 2
Basic Salary: 0
HRA: 0
DA: 0
PF: 0
Loan: 0
Net Salary: 0

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

```
50          Console.WriteLine("Basic Salary: " +
                basicSalary);
51          Console.WriteLine("HRA: " + hra);
52          Console.WriteLine("DA: " + da);
53          Console.WriteLine("PF: " + pf);
54          Console.WriteLine("Loan: " + loan);
55          Console.WriteLine("Net Salary: " +
                netSalary);
56      }
57  }
58  class Program
59  {
60      static void Main()
61      {
62          Employee emp = new Employee(0);
63          emp.Input();
64          emp.CalculateSalary();
65          emp.Display();
```

```
            basicSalary = 8000;
            hra = basicSalary * 0.10;
            da = basicSalary * 0.20;
            pf = 500;
            loan = 300;
        }
        else if (jobCatg == 2)
        {
            basicSalary = 15000;
            hra = basicSalary * 0.20;
            da = basicSalary * 0.30;
            pf = 1000;
            loan = 600;
        }
    }
    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());
    }
    public void CalculateSalary()
    {
        netSalary = basicSalary + hra + da - pf
            - loan;
    }
    public void Display()
    {
        Console.WriteLine("Employee Number: " +
            empNo);
        Console.WriteLine("Employee Name: " +
            empName);
        Console.WriteLine("Job Category: " +
            jobCatg);
        Console.WriteLine("Basic Salary: " +
            basicSalary);
        Console.WriteLine("HRA: " + hra);
        Console.WriteLine("DA: " + da);
        Console.WriteLine("PF: " + pf);
        Console.WriteLine("Loan: " + loan);
        Console.WriteLine("Net Salary: " +
```

```
                netSalary);
56      }
57  }
58  class Program
59  {
60      static void Main()
61      {
62          Employee emp = new Employee(0);
63          emp.Input();
64          emp.CalculateSalary();
65          emp.Display();
66      }
67  }
```

INPUT :

Enter employee number: 102

Enter employee name: John Doe

Enter job category (1 or 2): 2

OUTPUT :

Employee Number: 102

Employee Name: John Doe

Job Category: 2

Basic Salary: 0

HRA: 0

DA: 0

PF: 0

Loan: 0

Net Salary: 0