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 and print a 2D array containing numbers 1 to 9.

# PROGRAM:

using System;

class Program

{

static void Main()

{

int[,] array = {

{ 1, 2, 3 },

{ 4, 5, 6 },

{ 7, 8, 9 }

};

int rows = array.GetLength(0);

int cols = array.GetLength(1);

for (int i = 0; i < rows; i++)

{

for (int j = 0; j < cols; j++)

{

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

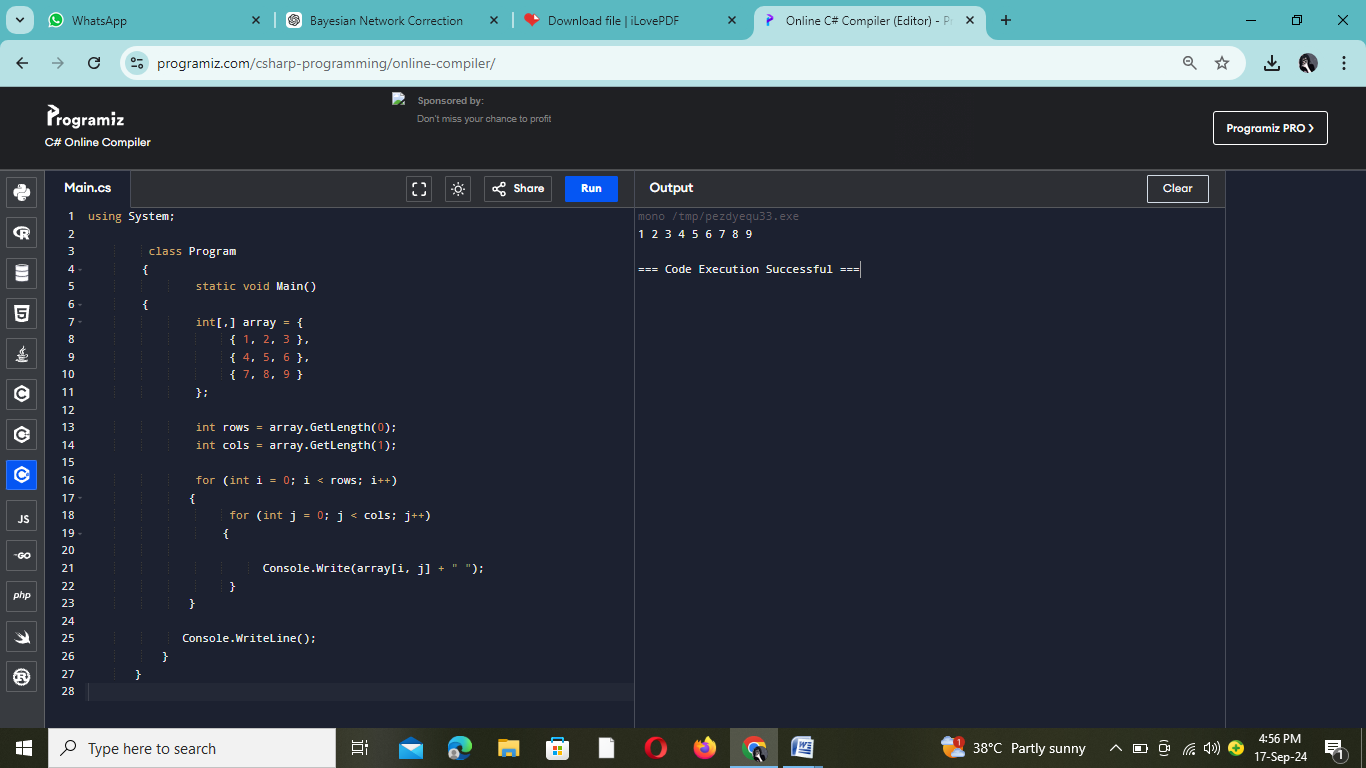
}

}

Console.WriteLine();

}

}



# OUTPUT:

## 1 2 3 4 5 6 7 8 9

1. **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.**
   * **Thecandidateshouldhavepassed10stdwithaminimumpass 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.");

# }

# }

# }

# C:\Users\FCS\Pictures\Screenshots\Screenshot (12).png

# INPUT:

Enter your age : 19

Enter your 10th standard pass percentage:58.5

# OUTPUT:

You are not eligible to apply for the competitive exam.

1. **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.**
   * **Shouldbeoflength10.**
   * **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 your mobile number (format XXXX-XXXXXX): ");

string mobileNumber = Console.ReadLine();

if (ValidateMobileNumber(mobileNumber))

{

Console.WriteLine("The mobile number is valid.");

}

else

{

Console.WriteLine("Invalid mobile number.");

}

}

static bool ValidateMobileNumber(string number)

{

string pattern = @"^[89]\d{3}-\d{6}$";

if (Regex.IsMatch(number, pattern))

{

return true;

}

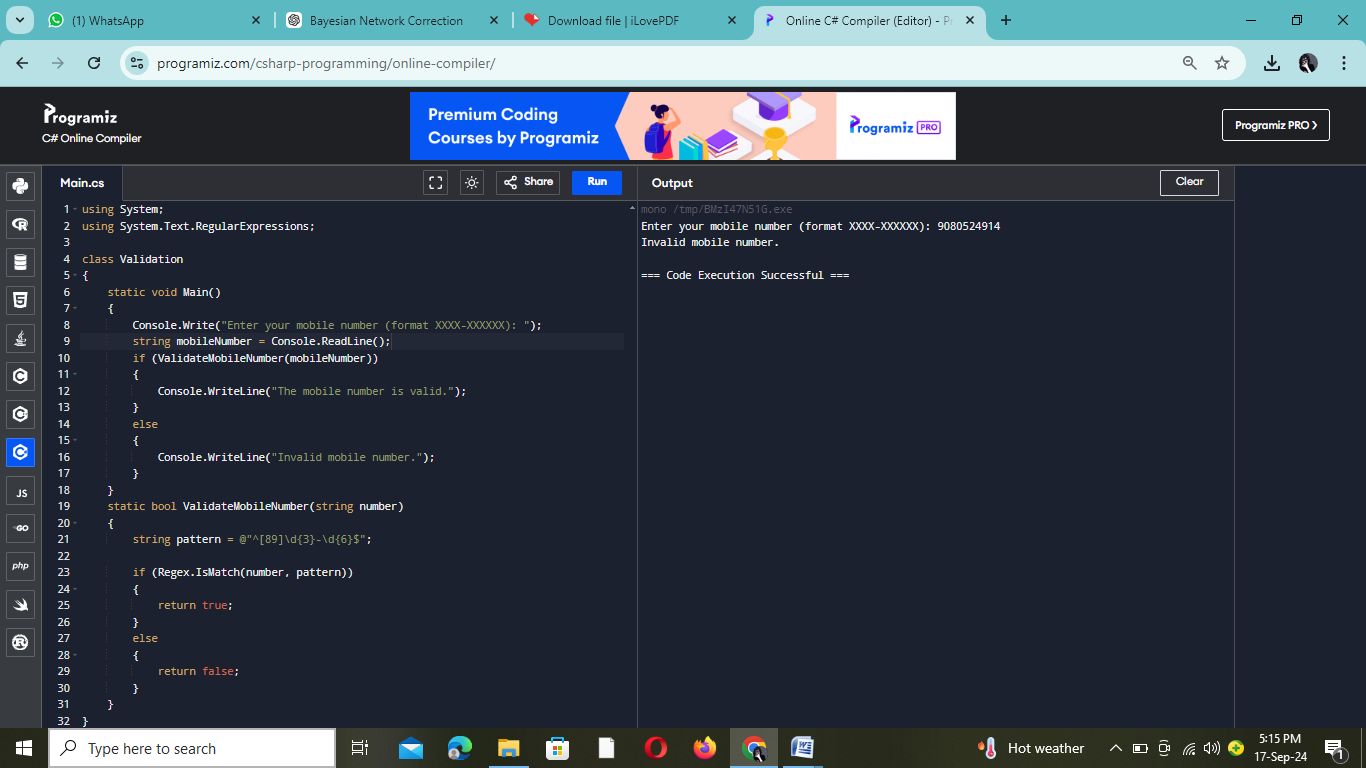
else

{

return false;

}

}

}

# INPUT:

# 9080524914

# OUTPUT:

Invalid mobile number.

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

**Class person{**

**name;**

**age;**

**weight;Voidprint person(){**

**//writethecodetoprintname,ageandweightofaperson**

**}**

**}**

**Classpersondata{**

**StaticvoidMain(string[]args){ person** **=**  **;**

**.name="Kannan";**

**.age=19;**

**.weight=58;**

**//writethestatementtoaccessprintperson()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 float 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();

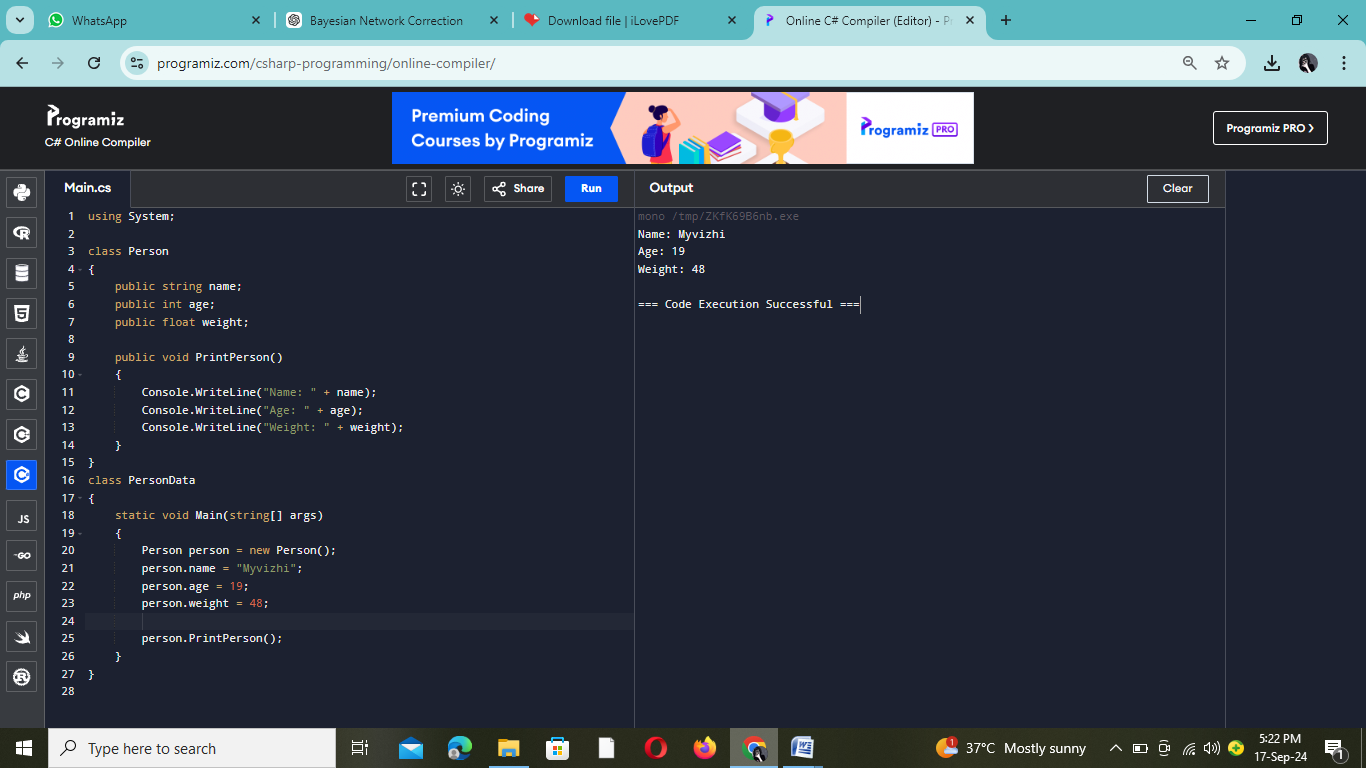
person.name = "Myvizhi";

person.age = 19;

person.weight = 48;

person.PrintPerson();

}

}

# OUTPUT:

Name : Myvizhi Age:19

Weight:48

1. **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 DateTime DateOfAdmission { get; set; }

public int Age { get; set; }

public string Disease { get; set; }

public DateTime DateOfDischarge { get; set; }

public double TotalBillsPaid { get; set; }

public void GetPatientInfo()

{

Console.Write("Enter patient's 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 = Convert.ToInt32(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 = Convert.ToDouble(Console.ReadLine());

}

public void DisplayPatientInfo()

{

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

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

Console.WriteLine("Date of Admission: " +DateOfAdmission.ToString("yyyy-MM-dd"));

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

Console.WriteLine("Disease: " + Disease);

Console.WriteLine("Date of Discharge: " + DateOfDischarge.ToString("yyyy-MM-dd"));

Console.WriteLine("Total Bills Paid: $" + TotalBillsPaid);

}

}

class Hospital

{

static void Main(string[] args)

{

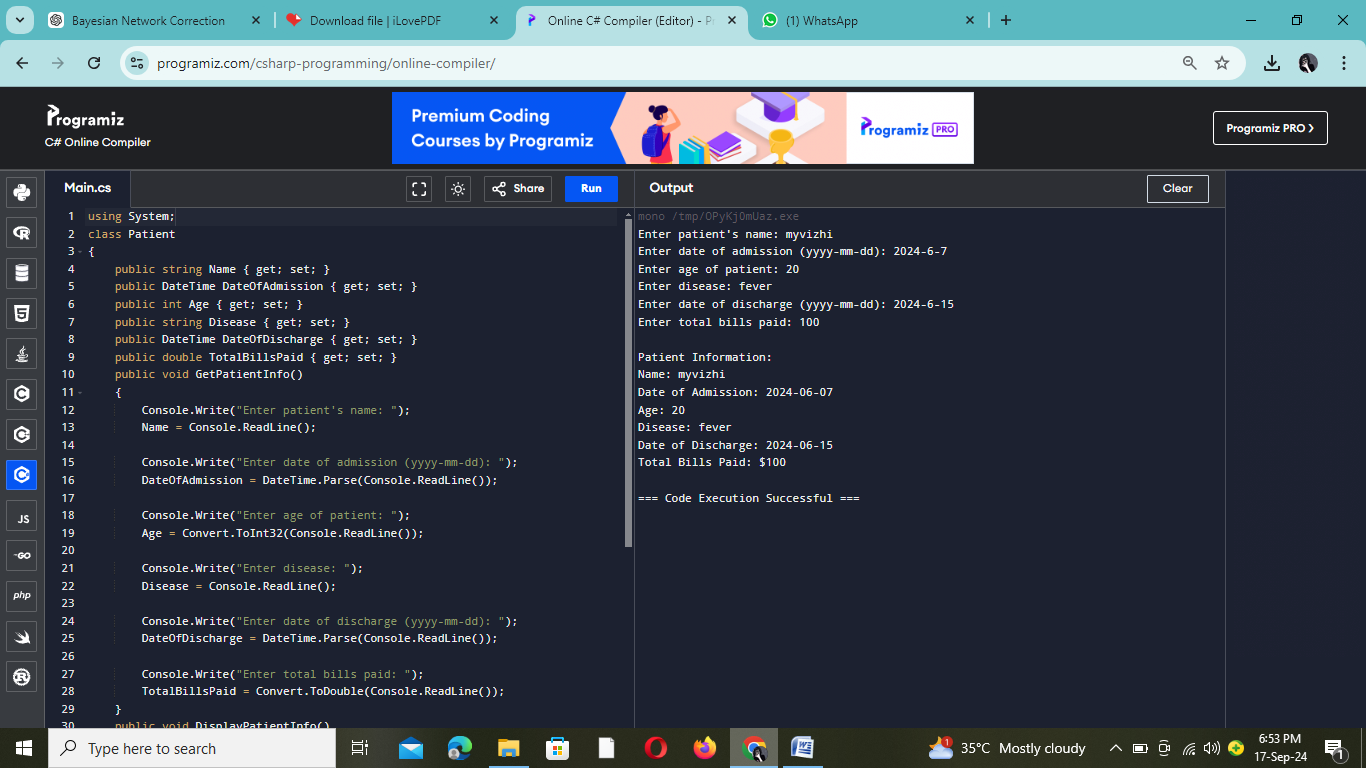
Patient patient = new Patient();

patient.GetPatientInfo();

patient.DisplayPatientInfo();

}

}



# INPUT:

Enter patient name : Myvizhi

Enter date of admission (dd/mm/yyyy):7/6/2024

Enter patient age: 20

Enter the disease: Fever

Enter date of discharge (dd/mm/yyyy):15/6/2024 Enter total bills paid: 6000

# OUTPUT:

Name : Myvizhi

DateofAdmission:7/6/2024 Age: 20

Disease : Fever

Date of Discharge : 15/6/2024 Total Bills Paid: 6000

1. **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 aVector 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.Write("Enter X component of the first vector: ");

# int x1 = Convert.ToInt32(Console.ReadLine());

# Console.Write("Enter Y component of the first vector: ");

# int y1 = Convert.ToInt32(Console.ReadLine());

# Vector vector1 = new Vector(x1, y1);

# Console.Write("Enter X component of the second vector: ");

# int x2 = Convert.ToInt32(Console.ReadLine());

# Console.Write("Enter Y component of the second vector: ");

# int y2 = Convert.ToInt32(Console.ReadLine());

# Vector vector2 = new Vector(x2, y2);

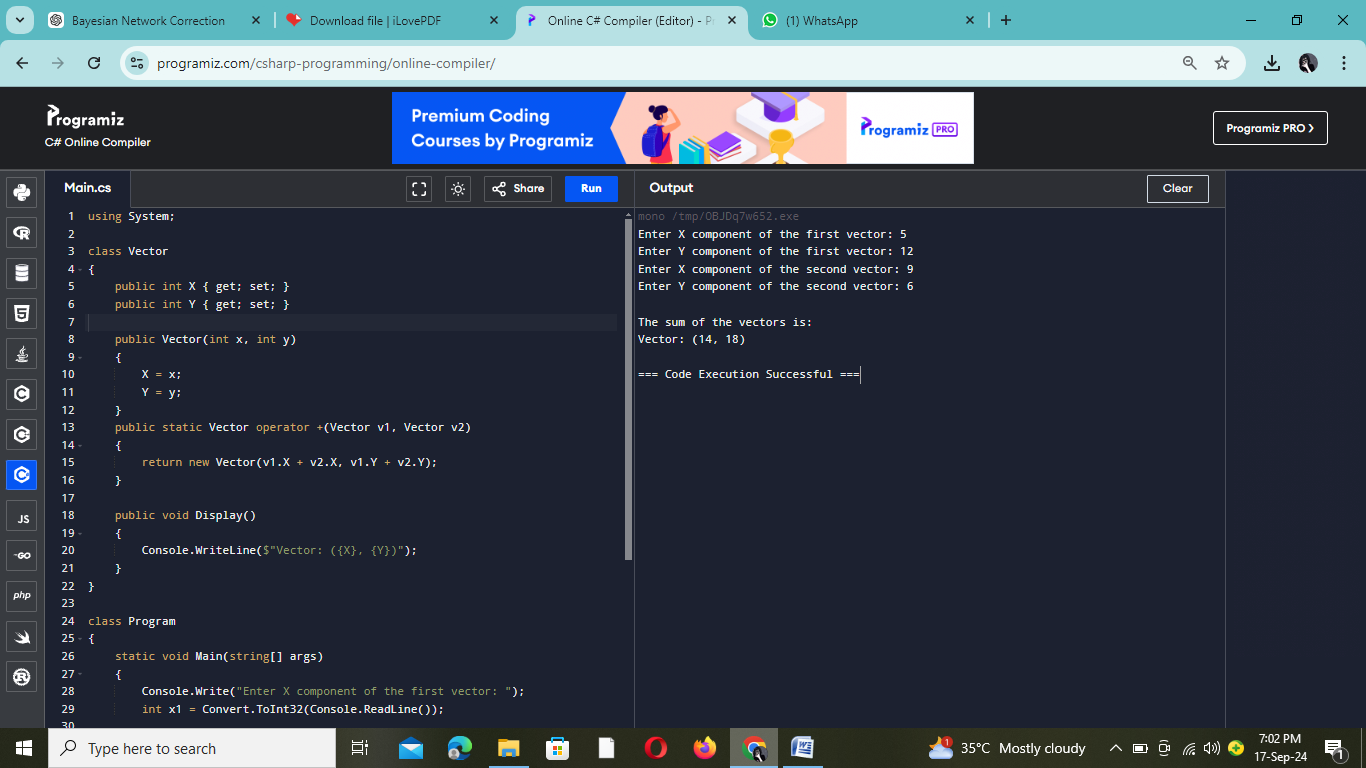
# Vector resultVector = vector1 + vector2;

# Console.WriteLine("\nThe sum of the vectors is:");

# resultVector.Display();

# }

}



# INPUT:

Enter the components of the first vector: X: 5

Y: 12

Enter the components of the second vector: X: 9

Y: 6

**OUTPUT:**

The sum of the two vectors is: Vector: (14, 18)

1. **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 student mark with necessary members to maintain student mark details. Over ride 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 aderivedStudentMark class, whichinherits andextendsthebaseclasstoincludemarkdetails,calculatesgradesbasedon marks, and demonstrates polymorphism through overridden methods.

# PROGRAM:

using System;

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.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

{

public int Marks { get; set; }

public override void GetData()

{

base.GetData();

Console.Write("Enter student's 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 >= 75)

return "B";

else if (Marks >= 50)

return "C";

else

return "Fail";

}

}

class StudentMain

{

static void Main(string[] args)

{

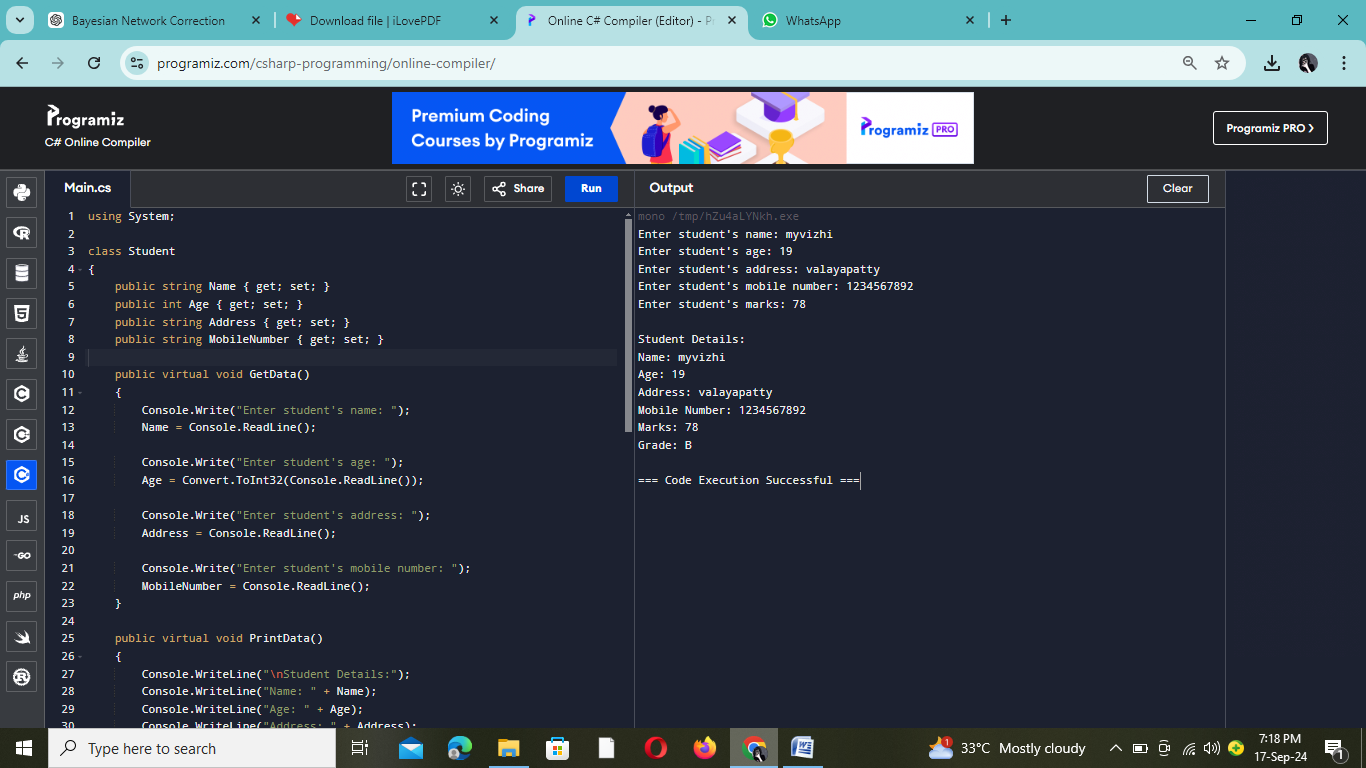
StudentMark student = new StudentMark();

student.GetData();

student.PrintData();

}

}



**INPUT:**

Enter student’s name : Myvizhi

Enter student's age: 19

Enter student's address : Valayapatty

Enter student's mobile number : 1234567892 Enter student's marks: 78

**OUTPUT:**

Name: Myvizhi

Age : 19

Address : Valayapatty

MobileNumber:1234567892

Marks:78 Grade: B

1. **Design sample C# program with class name employee to compute net salary of the employee using the basic salary, if for the job\_category 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\_category, empno, empname, calculateSalary() method to compute salary and display() method to print the details.**

|  |  |
| --- | --- |
| **Table-I** | **Table-II** |
| **BASIC=Rs.8,000**  **HRA=10%ofbasicDA=20%of basic LOAN=Rs. 300**  **PF=Rs.500** | **BASIC=Rs.15,000**  **HRA=20%ofbasicDA=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

# {

# public int EmpNo { get; set; }

# public string EmpName { get; set; }

# public int JobCategory { get; set; }

# public double BasicSalary { get; set; }

# public double HRA { get; set; }

# public double DA { get; set; }

# public double PF { get; set; }

# public double Loan { get; set; }

# public double NetSalary { get; set; }

# public Employee(double basicSalary, double hra, double da, double pf, double loan)

# {

# BasicSalary = basicSalary;

# HRA = hra;

# DA = da;

# PF = pf;

# Loan = loan;

# }

# 

# 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): ");

# JobCategory = Convert.ToInt32(Console.ReadLine());

# }

# public void CalculateSalary()

# {

# if (JobCategory == 1)

# {

# BasicSalary = 8000;

# HRA = BasicSalary \* 0.10;

# DA = BasicSalary \* 0.20;

# PF = 500;

# Loan = 300;

# }

# else if (JobCategory == 2)

# {

# BasicSalary = 15000;

# HRA = BasicSalary \* 0.20;

# DA = BasicSalary \* 0.30;

# PF = 1000;

# Loan = 600;

# }

# else

# {

# Console.WriteLine("Invalid job category.");

# return;

# }

# 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: " + JobCategory);

# 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(string[] args)

# {

# Employee employee = new Employee(0, 0, 0, 0, 0);

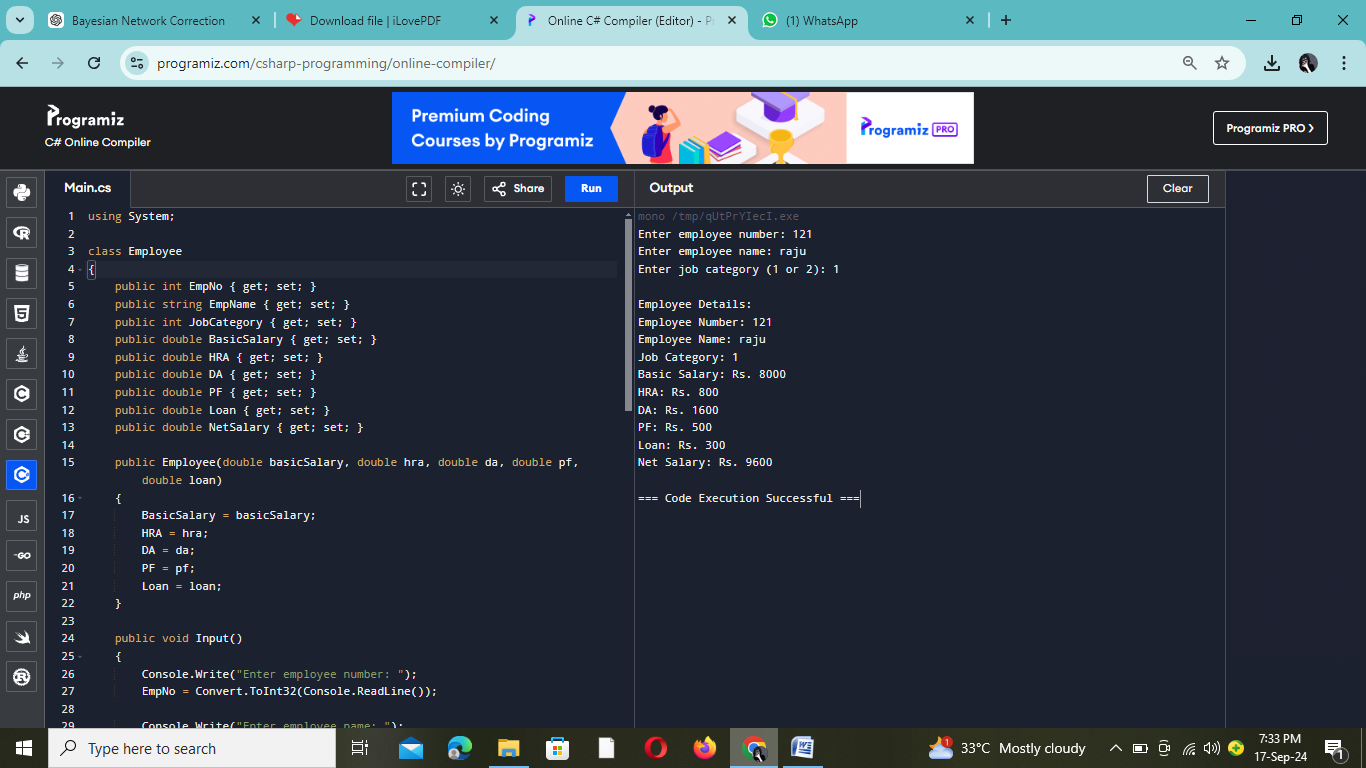
# employee.Input();

# employee.CalculateSalary();

# employee.Display();

# }

}



# INPUT:

Enter Employee Number : 121

Enter Employee Name : Raju Enter Job Category (1 or 2): 1 **OUTPUT:**

Employee Number : 121

Employee Name : Raju Job Category : 1

Basic Salary : 8000

HRA : 800

DA : 1600

PF: 500

Loan: 600

Net Salary : 9600

BY: MYVIZHI R 73772214165 III-B.E CSE