8. Implement a java application based on the MVC design pattern.
Input student Rolnlo, name ,marks in three subject calculate result and grade and display the result in neat format.

Percentage of Marks	Grade
Above 90%	A
80% to 90%	B
70% to 80%	C
60% to 70%	D
Below 60%	E

```
// Model class
class Student {
    private String rollNo;
    private String name;
    private int[] marks;

    public Student(String rollNo, String name, int[] marks) {
        this.rollNo = rollNo;
        this.name = name;
        this.marks = marks;
    }

    public double calculatePercentage() {
        int totalMarks = 0;
        for (int mark : marks) {
            totalMarks += mark;
        }
        return (double) totalMarks / marks.length;
    }
    public char calculateGrade() {
        double percentage = calculatePercentage();
        if (percentage > 90) {
            return 'A';
        } else if (percentage >= 80) {
            return 'B';
        } else if (percentage >= 70) {
            return 'C';
        } else if (percentage >= 60) {
            return 'D';
        } else {
            return 'E';
        }
    }

    public String getRollNo() {
        return rollNo;
    }

    public String getName() {
        return name;
    }
```

```
    public int[] getMarks() {
        return marks;
    }
}

// View class
class StudentView {
    public void displayStudentDetails(Student student) {
        System.out.println("Student Details:");
        System.out.println("Roll No: " + student.getRollNo());
        System.out.println("Name: " + student.getName());
        System.out.println("Marks:");
        int[] marks = student.getMarks();
        System.out.println("Subject 1: " + marks[0]);
        System.out.println("Subject 2: " + marks[1]);
        System.out.println("Subject 3: " + marks[2]);
        System.out.println("Percentage: " + student.calculatePercentage());
        System.out.println("Grade: " + student.calculateGrade());
    }
}

// Controller class
class StudentController {
    private Student model;
    private StudentView view;

    public StudentController(Student model, StudentView view) {
        this.model = model;
        this.view = view;
    }

    public void updateView() {
        view.displayStudentDetails(model);
    }
}
// Main class
public class MVCDemo {
    public static void main(String[] args) {
        // Create a student object
        int[] marks = {85, 75, 90};
        Student student = new Student("123456", "Vivek", marks);
        // Create view to display student details
        StudentView view = new StudentView();
        // Create controller
        StudentController controller = new StudentController(student, view);

        // Display student details
        controller.updateView();
    }
}
```

Output

Student Details:
Roll No: 123456
Name: Vivek
Marks:
Subject 1: 85
Subject 2: 75
Subject 3: 90
Percentage: 83.33333333333333
Grade: B

Steps for Execution

Create a New Project:

Choose Java with Maven -> Java Application
Give Project name MVCStudentApp

Create Packages

Right-click on the "Source Packages" folder in the project explorer.
Select "New" -> "Java Package".
Enter a package name (e.g., "MVC").
Click "Finish".

Create Classes (4 Classes - Student, StudentView, StudentController, and MVCDemo)

Right-click on the package you created.
Select "New" -> "Java Class".
Enter the class name

Copy the code for the Student, StudentView, StudentController, and MVCDemo classes into their respective files.
Run the Application:
Right-click on the MVCDemo.java file.
Select "Run File".