**Exercise 10: Implementing the MVC Pattern**

**Scenario:**

You are developing a simple web application for managing student records using the MVC pattern.

**Steps:**

1. **Create a New Java Project:**
   * Create a new Java project named **MVCPatternExample**.
2. **Define Model Class:**
   * Create a class **Student** with attributes like **name, id, and grade**.
3. **Define View Class:**
   * Create a class **StudentView** with a method **displayStudentDetails()**.
4. **Define Controller Class:**
   * Create a class **StudentController** that handles the communication between the model and the view.
5. **Test the MVC Implementation:**
   * Create a main class to demonstrate creating a **Student**, updating its details using **StudentController**, and displaying them using **StudentView**.

Code –

package Design\_Patterns\_And\_Principles.MVCPatternExample;

 class Student {

    private String name;

    private int id;

    private String grade;

    // Constructor

    public Student(String name, int id, String grade) {

        this.name = name;

        this.id = id;

        this.grade = grade;

    }

    // Getters and Setters

    public String getName() {

        return name;

    }

    public void setName(String name) {

        this.name = name;

    }

    public int getId() {

        return id;

    }

    public void setId(int id) {

        this.id = id;

    }

    public String getGrade() {

        return grade;

    }

    public void setGrade(String grade) {

        this.grade = grade;

    }

}

 class StudentView {

    public void displayStudentDetails(String studentName, int studentId, String studentGrade) {

        System.out.println("Student Details:");

        System.out.println("Name: " + studentName);

        System.out.println("ID: " + studentId);

        System.out.println("Grade: " + studentGrade);

    }

}

 class StudentController {

    private Student model;

    private StudentView view;

    // Constructor

    public StudentController(Student model, StudentView view) {

        this.model = model;

        this.view = view;

    }

    // Getters for model details

    public String getStudentName() {

        return model.getName();

    }

    public void setStudentName(String name) {

        model.setName(name);

    }

    public int getStudentId() {

        return model.getId();

    }

    public void setStudentId(int id) {

        model.setId(id);

    }

    public String getStudentGrade() {

        return model.getGrade();

    }

    public void setStudentGrade(String grade) {

        model.setGrade(grade);

    }

    // Update view with student details

    public void updateView() {

        view.displayStudentDetails(model.getName(), model.getId(), model.getGrade());

    }

}

public class MVCDemo {

    public static void main(String[] args) {

        // Create a Student object (Model)

        Student student = new Student("John Doe", 123, "A");

        // Create a StudentView object (View)

        StudentView studentView = new StudentView();

        // Create a StudentController object (Controller)

        StudentController studentController = new StudentController(student, studentView);

        // Display initial student details

        studentController.updateView();

        // Update student details

        studentController.setStudentName("Jane Doe");

        studentController.setStudentId(456);

        studentController.setStudentGrade("B");

        // Display updated student details

        studentController.updateView();

    }

}