Design patterns and principles

SuperSet ID:6412063

Exercise 10: Implementing the MVC Pattern

Code:

public class Student {

private String id;

private String name;

private String grade;

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

this.id = id;

this.name = name;

this.grade = grade;

}

public String getId() {

return id;

}

public void setId(String id) {

this.id = id;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public String getGrade() {

return grade;

}

public void setGrade(String grade) {

this.grade = grade;

}

}

public class StudentView {

public void displayStudentDetails(String id, String name, String grade) {

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

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

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

}

}

public class StudentController {

private Student student;

private StudentView view;

public StudentController(Student student, StudentView view) {

this.student = student;

this.view = view;

}

public void setStudentName(String name) {

student.setName(name);

}

public void setStudentId(String id) {

student.setId(id);

}

public void setStudentGrade(String grade) {

student.setGrade(grade);

}

public String getStudentName() {

return student.getName();

}

public String getStudentId() {

return student.getId();

}

public String getStudentGrade() {

return student.getGrade();

}

public void updateView() {

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

}

}

public class MVCTest {

public static void main(String[] args) {

Student student = new Student("101", "Alice", "A");

StudentView view = new StudentView();

StudentController controller = new StudentController(student, view);

controller.updateView();

controller.setStudentName("Bob");

controller.setStudentGrade("B+");

controller.updateView();

}

}

Output:

A screen shot of a computer

Description automatically generated