

Name Hamad khan

Reg No SP23 BSE 006

Section A

Student.java

```
public class Student {  
    private int id;  
    private String name;  
    private String email;  
  
    public Student(int id, String name, String email) {  
        this.id = id;  
        this.name = name;  
        this.email = email;  
    }  
  
    // Getters and setters  
    public int getId() { return id; }  
    public void setId(int id) { this.id = id; }  
  
    public String getName() { return name; }  
    public void setName(String name) { this.name = name; }
```

```
public String getEmail() { return email; }

public void setEmail(String email) { this.email = email; }

@Override

public String toString() {

    return "Student ID: " + id + ", Name: " + name + ", Email: " + email;

}

}
```

StudentValidator.java

```
public class StudentValidator {

    public boolean validate(Student student) {

        return student.getId() > 0 &&

            student.getName() != null && !student.getName().isEmpty() &&

            student.getEmail() != null && student.getEmail().contains("@");

    }

}
```

StudentRepository.java

```
import java.util.HashMap;

public class StudentRepository {

    private HashMap<Integer, Student> studentDB = new HashMap<>();

}
```

```
public void save(Student student) {  
    studentDB.put(student.getId(), student);  
}  
  
public Student findById(int id) {  
    return studentDB.get(id);  
}  
  
public void update(Student student) {  
    studentDB.put(student.getId(), student);  
}  
}
```

StudentService.java

```
public class StudentService {  
    private StudentRepository repository = new StudentRepository();  
    private StudentValidator validator = new StudentValidator();  
  
    public String addStudent(Student student) {  
        if (validator.validate(student)) {  
            repository.save(student);  
            return "Student added successfully.";  
        } else {  
            return "Invalid student data.";  
        }  
    }  
}
```

```

    }

    public String editStudent(Student student) {
        if (repository.findById(student.getId()) == null) {
            return "Student not found.";
        }

        if (validator.validate(student)) {
            repository.update(student);
            return "Student updated successfully.";
        } else {
            return "Invalid data for update.";
        }
    }

    public Student getStudent(int id) {
        return repository.findById(id);
    }
}

```

StudentController.java

```

public class StudentController {

    private StudentService service = new StudentService();

    public void addStudent(int id, String name, String email) {

```

```

        Student student = new Student(id, name, email);
        String result = service.addStudent(student);
        System.out.println(result);
    }

    public void editStudent(int id, String name, String email) {
        Student student = new Student(id, name, email);
        String result = service.editStudent(student);
        System.out.println(result);
    }

    public void showStudent(int id) {
        Student student = service.getStudent(id);
        if (student != null)
            System.out.println(student);
        else
            System.out.println("Student not found.");
    }
}

```

Main.java

```

import java.util.Scanner;

public class Main {
    public static void main(String[] args) {

```

```
Scanner scanner = new Scanner(System.in);

StudentController controller = new StudentController();

while (true) {

    System.out.println("\n1. Add Student\n2. Edit Student\n3. View Student\n4. Exit");

    System.out.print("Select an option: ");

    int option = scanner.nextInt();

    scanner.nextLine(); // consume newline

    if (option == 4) break;

    System.out.print("Enter ID: ");

    int id = scanner.nextInt();

    scanner.nextLine();

    System.out.print("Enter Name: ");

    String name = scanner.nextLine();

    System.out.print("Enter Email: ");

    String email = scanner.nextLine();

    switch (option) {

        case 1 -> controller.addStudent(id, name, email);

        case 2 -> controller.editStudent(id, name, email);

        case 3 -> controller.showStudent(id);

        default -> System.out.println("Invalid choice.");

    }

}
```

}

}