Java Project On Online Course Management System Using Java As Front End And My SQL As Backend

Name:P.Manohar
Reg No:192210723

JAVA:

```
import java.util.*;

// Class to represent a course
class Course {
    private String id;
    private String name;
    private String instructor;
    private List<String> students;

    public Course(String id, String name, String instructor) {
        this.id = id;
        this.name = name;
        this.instructor = instructor;
        this.students = new ArrayList<>();
    }

    public String getId() {
        return id;
    }

    public String getName() {
        return name;
    }

    public String getInstructor() {
        return instructor;
    }

    public List<String> getStudents() {
        return students;
    }

    public void enrollStudent(String studentId) {
```

```
        students.add(studentId);
    }

    public void printCourseInfo() {
        System.out.println("Course ID: " + id);
        System.out.println("Course Name: " + name);
        System.out.println("Instructor: " + instructor);
        System.out.println("Students Enrolled: " + students.size());
    }
}

// Class to manage courses
class CourseManager {
    private List<Course> courses;

    public CourseManager() {
        this.courses = new ArrayList<>();
    }

    public void addCourse(Course course) {
        courses.add(course);
    }

    public Course getCourseById(String courseId) {
        for (Course course : courses) {
            if (course.getId().equals(courseId)) {
                return course;
            }
        }
        return null;
    }

    public void printAllCourses() {
        for (Course course : courses) {
            course.printCourseInfo();
            System.out.println("------------------------");
        }
    }
}

public class OnlineCourseManagementSystem {
    public static void main(String[] args) {
        CourseManager courseManager = new CourseManager();
```

```
        Course javaCourse = new Course("1", "Java Programming", "John Doe");
        javaCourse.enrollStudent("101");
        javaCourse.enrollStudent("102");
        courseManager.addCourse(javaCourse);

        Course sqlCourse = new Course("2", "SQL Fundamentals", "Jane Smith");
        sqlCourse.enrollStudent("103");
        courseManager.addCourse(sqlCourse);

        courseManager.printAllCourses();
    }
}
```

Course Java :

```
public class Course {
    private int id;
    private String name;
    private String instructor;

    public Course(int id, String name, String instructor) {
        this.id = id;
        this.name = name;
        this.instructor = instructor;
    }

    // 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 getInstructor() {
        return instructor;
    }

    public void setInstructor(String instructor) {
        this.instructor = instructor;
    }

    @Override
    public String toString() {
        return "Course{" +
                "id=" + id +
                ", name='" + name + '\'' +
                ", instructor='" + instructor + '\'' +
                '}';
    }
}
```

Student.Java:

```
public class Student {
    private int id;
    private String name;
    private String email;
    private int courseId;

    public Student(int id, String name, String email, int courseId) {
        this.id = id;
        this.name = name;
        this.email = email;
        this.courseId = courseId;
    }

    // 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;
    }

    public int getCourseId() {
        return courseId;
    }

    public void setCourseId(int courseId) {
        this.courseId = courseId;
    }

    @Override
    public String toString() {
        return "Student{" +
                "id=" + id +
                ", name='" + name + '\'' +
                ", email='" + email + '\'' +
                ", courseId=" + courseId +
                '}';
    }
}
```

Course Manager.Java:

```
import java.sql.*;
import java.util.ArrayList;
import java.util.List;

public class CourseManager {
```

```
    private static final String JDBC_URL =
"jdbc:mysql://localhost:3306/course_management";
    private static final String USERNAME = "your_username";
    private static final String PASSWORD = "your_password";

    public List<Course> getAllCourses() {
        List<Course> courses = new ArrayList<>();
        try (Connection conn = DriverManager.getConnection(JDBC_URL, USERNAME,
PASSWORD);
             Statement stmt = conn.createStatement()) {
            String query = "SELECT * FROM courses";
            ResultSet rs = stmt.executeQuery(query);
            while (rs.next()) {
                int id = rs.getInt("id");
                String name = rs.getString("name");
                String instructor = rs.getString("instructor");
                Course course = new Course(id, name, instructor);
                courses.add(course);
            }
        } catch (SQLException e) {
            e.printStackTrace();
        }
        return courses;
    }

    public void enrollStudent(int studentId, int courseId) {
        try (Connection conn = DriverManager.getConnection(JDBC_URL, USERNAME,
PASSWORD);
             PreparedStatement stmt = conn.prepareStatement("INSERT INTO students (id,
name, email, course_id) VALUES (?, ?, ?, ?)")) {
            // For simplicity, assume that student name and email are provided
            stmt.setInt(1, studentId);
            stmt.setString(2, "Student " + studentId);
            stmt.setString(3, "student" + studentId + "@example.com");
            stmt.setInt(4, courseId);
            stmt.executeUpdate();
        } catch (SQLException e) {
            e.printStackTrace();
        }
    }

    public static void main(String[] args) {
        CourseManager courseManager = new CourseManager();
```

```
        // Get all courses
        List<Course> courses = courseManager.getAllCourses();
        System.out.println("Courses:");
        for (Course course : courses) {
            System.out.println(course);
        }

        // Enroll a student in a course
        int studentId = 1;
        int courseId = 1;
        courseManager.enrollStudent(studentId, courseId);
        System.out.println("\nStudent enrolled successfully.");
    }
}
```

MY SQL:

```
-- Create database
CREATE DATABASE IF NOT EXISTS course_management;

-- Use the database
USE course_management;

-- Create courses table
CREATE TABLE IF NOT EXISTS courses (
    id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(255) NOT NULL,
    instructor VARCHAR(255) NOT NULL
);

-- Create students table
CREATE TABLE IF NOT EXISTS students (
    id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(255) NOT NULL,
    email VARCHAR(255) NOT NULL,
    course_id INT,
    FOREIGN KEY (course_id) REFERENCES courses(id)
);
```