Project Title: Student Grade Management System

Description:

This project aims to develop a **Student Grade Management System (SGMS)** to simplify the management of student records, grades, and course enrollments. The application will use a **Command-Line Interface (CLI)** to provide a straightforward user experience for managing student and course data. The focus will be on **accuracy**, **efficiency**, and **intuitive interactions** while ensuring robust error handling and validation.

Team Members:
Wilson Xu (wx2105@nyu.edu)
Jacqueline Peng (jp5982@nyu.edu)
Claire Kim (sk9571@nyu.edu)

Key Features:

1. **Student Management:**
   - Add new student records with attributes such as name, ID, and enrolled courses.
   - Update existing student information (e.g., name, ID, or courses).
   - Delete student records from the system.
2. **Course Management:**
   - Add new courses to the system with attributes such as course name, course code, and instructor.
   - Update or delete courses from the system.
   - Assign grades to students for specific courses.
3. **Grade Management and Analysis:**
   - Calculate and display the average, minimum, and maximum grades for each student.
   - Categorize students by performance (e.g., "Excellent," "Good," "Needs Improvement") based on grade thresholds.
4. **Search and Reporting:**
   - Search for students by name to view all courses they’ve taken along with their respective grades.
   - Search for courses by name to view all enrolled students and their grades.
   - Generate performance reports for individual students or courses.

Technical Details:

- **Programming Language:**
  - Java (using Object-Oriented Design principles).
- **Core Classes:**
  - `Student`: Represents a student with attributes like name, ID, and a list of enrolled courses.
  - `Course`: Represents a course with attributes like course name, course code, and instructor.
  - `GradeManager`: Central class to manage student records, course enrollments, and grades.
- **CLI Interaction:**
  - Example main menu:

Welcome to the Student Grade Management System!

Please select an option:

1. Manage Students

2. Manage Courses

3. Assign Grades

4. Search Records

5. Generate Reports

6. Exit

- Submenus for specific tasks under each category, such as adding, updating, or searching for records.

UML Diagrams:

1. **Use Case Diagram:**
   - Illustrates interactions between the user and the system for managing students, courses, grades, and searches.
2. **Class Diagram:**
   - Depicts relationships between the `Student`, `Course`, and `GradeManager` classes.
3. **Sequence Diagram:**
   - Details the flow of processes like assigning grades or searching for a student by name.

Data and Functionality:

1. **Student Records:**
    - Maintain a list of students, including their names, IDs, and courses taken.
2. **Course Records:**
    - Maintain a list of courses, including course names, codes, and instructors.
3. **Search and Report Generation:**
    - Search for a student by name to display all enrolled courses and their grades.
    - Search for a course by name to display all enrolled students and their grades.
    - Generate performance reports with overall class statistics or individual student details.
4. **In-Memory Data Storage:**
    - All data (students, courses, grades) will be stored in memory during program execution.
    - Data will reset when the program is closed.
5. **Validation and Error Handling:**
    - Validate unique IDs for students and unique codes for courses.
    - Handle invalid inputs with clear error messages.
    - Prevent assigning grades for courses that a student is not enrolled in.

Example Workflow:

1. **User Starts the Program:**
    - The user is greeted with the main menu and selects an option.
2. **Adding a Student Record:**
    - The user enters the student’s name and ID.
    - The system validates and stores the information.
3. **Adding a Course Record:**
    - The user enters the course name, code, and instructor.
    - The system validates and stores the information.
4. **Assigning Grades:**
    - The user selects a student and assigns grades for specific courses.
    - The system updates the student’s records.
5. **Searching for Records:**
    - The user searches for a student or course to view detailed information.
6. **Generating Reports:**
    - The system displays performance reports for students or courses.
7. **Exiting the Program:**
    - The user exits, and all data is cleared from memory.