CIS611

Spring 2018

Individual Practice Programming Assignment: PA05

Total Points: 20

Classes and Objects

The purpose of this programming assignment is to:

- Demonstrate defining classes and creating objects
- Access objects via object reference variables
- Distinguish between instance and static variables and methods
- Define private data fields with appropriate get and set methods

Q1 (20 points)**:**
Implement the given Java Application UML diagram and the associated source code files of the UML diagram. You need to complete the implementation of the methods stubs in the provided Java source files. As shown in the UML diagram below, this Java application has 4 classes (`School`, `Course`, `Student`, and `Grade` classes), and you need to take an insightful look at the UML diagram in order to perceive the relationship/association between these classes. The classes are defined and associated together in this project as the following:

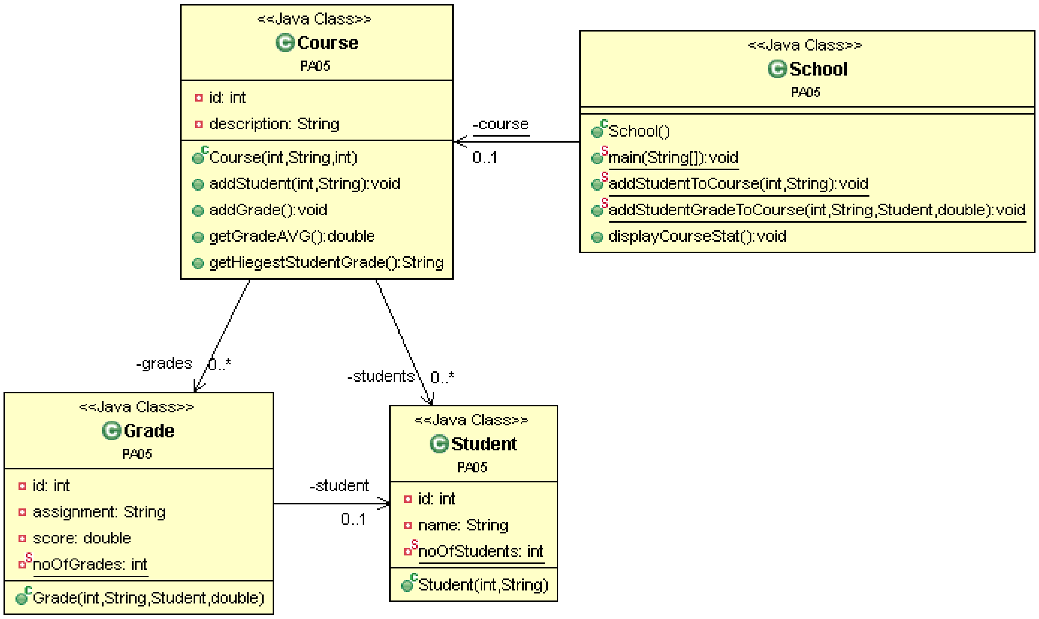
- The `School` class, is the main entry class that has the main method. This is the class that should have all the user GUIs. All the tasks of this assignment should be implemented in this class, and this class invokes methods from the `Course` class in order to perform the project requirements. Thus, it is recommended to complete the unimplemented methods in other classes before implementing the methods in this `School` class. You may either follow the top-down or bottom-up implementation approach.
- The `Course` class is the class that has the course information and it has the list of students and grade object. All the unimplemented methods in this class are invoked by the main `School` class in order to create a course, add student, add a grade, get the grades average, and the name of the student who has the highest grade.
- The `Student` class has the student information and some unimplemented mutator methods used to control changes to variables from outside this class, and these methods are accessed by the `Course` class.
- The `Grade` class has the grade information for a student and some unimplemented mutator methods used to control changes to variables from outside this class, and these methods are accessed by the `Course` class.

Note that, all the class variable are only accessible within the class itself.

This partial Java application is for course registration and grading. A user of this application should be able to do the following tasks in a school:

1. Instantiates or creates a course object in order to add a new course to a school
2. Prompt the user to input the student data in order to add the a student to course, for at most three students
3. Prompt the user to input the grade data for each Student object of the Course object, where there will be at n students, n value is provided by the user input.
4. Displays the course statistics (including the course information and students names), such as the grades average and the name of the student who has the highest grade.

In order for these tasks to be performed, you need implement the methods in the provided Java source files as described in the comments on each method in the Java class's source code. You should use the JOptionPane class to interact with the user, and also input validation is required. You must not change any visibility modifiers in the provided classes. Also, all incomplete methods must be completed in order to get this java project working.



Evaluation Criteria:

- The program must compile cleanly (no compile errors, but compile warnings are sometimes accepted)
- The program should handle invalid data input entries by users and terminate gracefully
- The program should not crash while running and it should terminate
- All tasks (requirements) in this assignment must be completed in order to receive credit
- The correct understanding and implementation (coding) of the requirements (programs should behave as anticipated):
    - The program must terminate with proper/correct outputs
    - All the logical computations should be performed correctly

Submission: (This is an individual Assignment!)

Copy the .java source files from the *src* folder in your *work space* to another folder that should be named following the provided naming format in this course, then zip and upload the file under this assignment answer in Canvas.

> ***File Name:*** *FLLLLPA05.zip (F = first letter in your first name and LLLL = your last name)*

Grading Rubric PA05

Student Name: ____

Question 1

Requirements	Comment	Max Points Allowed	Points Earned
General Code Structure: Proper naming convention used for file (0.25) Comments used in the code to explain the purpose of the code (0.25) Indentation of the code for better readability (0.25) Good choice of the variable names (0.25)		1	
Input, Output, User Interface: Proper coding implementation for the input of the student data (1) Proper coding implementation for the input of the grade data (1) Display the student statistics (2) Exception handling of the invalid input data. For example, if no data is entered, empty space is entered, invalid data is entered, the program should not crash (2)		6	
General Algorithm and Logic: Proper implementation of the classes and its methods, as shown in the UML diagram (11) Logic for at most n students (2)		13	
Total		20	

Total ____/20