
SCIT

School of Computing and Information Technology
Faculty of Engineering & Information Sciences

CSIT121

Object Oriented Design and Programming Assignment 1

Due date: 24 April 2022

Objectives

- Apply OO design technique: allocation of classes and responsibilities.
- Use OO features: class, instance variable, constructor and method to implement the solution.
- Adopt good programming practices: modular code, appropriate comments, etc.

Online Quiz (3 marks)

To be announced.

Programming Task (4 marks)

Write a Java program to allow user to input, save and manipulate a list of test results. A test result includes the applicant Id, name and the two test scores of an applicant taking a test. The program will allow the user to input test result repeatedly until the user indicates to stop. The following screenshot of the program execution illustrates the program flow. You may use an instance of **ArrayList** to save the results. <https://docs.oracle.com/javase/8/docs/api/java/util/ArrayList.html>

```
Candidate Id: C001
Name       : Alice
Test 1 score: 60
Test 2 score: 85
Enter another record? (y/n) y

Candidate Id: C004
Name       : Eric Chan
Test 1 score: 58
Test 2 score: 74
Enter another record? (y/n) n
```

When the data entry is done, the program will display a list of options as follow.

```
1 Search test result by applicant Id
2 Search test result(s) by applicant name
3 Search test result(s) by overall score
4 Show test result(s) with highest overall score
5 Show failure list
6 Quit
Please choose an option:
```

Option 1

The program will prompt the user to enter an applicant Id, search and display the corresponding test result. The program will display an appropriate message if the there is no matching test result.

Option 2

The program will prompt the user to enter an applicant name, search and display the corresponding test result(s). The program will display an appropriate message if the there is no matching test result.

Option 3

The program will prompt the user to enter two scores: lower bound and upper bound. The program will display test results whose overall score is between the range (inclusive of both ends). The program will display an appropriate message if the there is no matching test result.

Option 4

The program will display test results whose overall score is the same as the highest overall score of the results in the ArrayList. The program will display an appropriate message if the there is no matching test result.

Option 5

The program will display test results whose overall score is below the passing marking. The program will display an appropriate message if the there is no matching test result.

Option 6

The program will exit the loop and terminate.

Class Design

The program will consist of **two or more** classes depending on your design. One class will contain the main method, and is responsible for interacting with the user (input/output) among other things.

The **TestResult** class will include instance variables to store the applicant Id, name and the two test scores. The TestResult class must include appropriate constructor(s), getters, setters and the following methods:

- A method to compute the **overall score** using the formula:
 - $\text{overall score} = \text{test score 1} \times 40\% + \text{test score 2} \times 60\%$
- A method to compute and **return the grade** based on the overall score:
 - Overall score $\geq 75 \rightarrow$ "Good"
 - Overall score is between 50 and 74 (inclusive at both ends) \rightarrow "Pass"
 - Overall score $< 50 \rightarrow$ "Fail"
- A method to compare **two instances of TestResult**:
 - The method will include a parameter of TestResult.
 - The method will compare the applicant Ids and return true if are the same Id.
 - <https://www.softwaretestinghelp.com/equals-method-in-java/>
 - <https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html>

You may include additional instance and/or static methods as you deemed fit.

Source code comments

Please include appropriate header and block comments in the program.

Submission

Please submit a single Java file (containing the above classes) to Moodle.