ENSF 409 – Principles of Software Development
Summer 2021

Lab Assignment #3:

Due Dates
Submit electronically on D2L **before <u>2:00 PM on Friday July 16th 2021</u>**

The objectives of this lab are to:

1) write and test a Java application with multiple classes.
2) understand the concept of class relationships such as association, and aggregation, and composition.
3) implement classes in Java with the association, aggregation, and composition relationships among them.
4) Become familiar with modeling concepts and modeling tools

<u>The following rules apply to this lab and all other lab assignments in future:</u>

1. Before submitting your lab reports, take a moment to make sure that you are handing in all the material that is required. If you forget to hand something in, that is your fault; you can't use `I forgot' as an excuse to hand in parts of the assignmentlate.
2. **<u>20% marks</u>** will be deducted from the assignments handed in up to **<u>24 hours</u>** after each due date. It means if your mark is X out of Y, you will only gain 0.8 times X. There will be no credit for assignments turned in later than 24 hours after the due dates; they will be returned unmarked. Exceptions can be made but the students must inform that Instructor beforehand to accommodate if acceptable

Exercise- 1: Implementing the retail store program (30 Marks) You are asked to implement your design of the retail store for Exercise 3 of lab 2.**Task 1:**

Task 1: Implementing the Backend

Implement your class diagram. Note: your implementation must follow your design exactly. Your classes and methods must be highly cohesive and must adhere to the *single responsibility principle.*

The **single-responsibility principle** (**SRP**) is a computer-programming principle that states that every module, class or function in a computer program should have responsibility over a single part of that program's functionality, and it should encapsulate that part. All of that module, class or function's services should be narrowly aligned with that responsibility.

Backend refers to the inner workings of a program that the user will not directly interact with. This includes classes and their implementations that the user does not need to know about their existence and implementation.

Task 2: Implement the Frontend:

Implement a main function to test your program using the two textfiles (i.e. `items.txt`, and `suppliers.txt`) from D2L. These files are posted under the lab 2 folder on D2L. You can download the input files for the items to test your application.

To properly test your application, you are to create an interactive console-based menu. The program should keep on running (i.e. should keep presenting the user with the menu) until the user quits. You can design your menu and make changes, but the following items should be represented in your menu:

1. List all tools (this must be handled by the proper `toString` methods in the backend).
2. Search for tool by toolName
3. Search for tool by toolID
4. Check item quantity
5. Decrease Item quantity (This is to simulate a sale of the item. Once the item count goes under 40, this function should trigger the creation of an order as shown in lab 2) .
6. Quit

Hint: Use a while loop and switch statement or if-else structure to design your menu

Frontend is the part of a program that the user will interact with such a program UI. In this case that will be the main function that presents the menu to the user and provide file input output for reading and storing data

Task 3: Add Java documentation to your source code and generate Java doc files.

What to hand in: Submit your source code and all required files for this exercise. Include your Javadoc files in a sub-folder in the exercise folder with exercise number as name, this folder will be part of the final compressed file submission in your project file.

Exercise- 2: Using the Backend for the Course Registration System (15 Marks)

In this exercise you are asked to develop the front end for the Course Registration System that we have been working on during class (the finalized version of this code will be on D2L **after our Lecture on Monday 12th of July**). Your tasks are as follows:

Task 1 – Complete the back end such that it complies with the requirements/restrictions mentioned in the requirements from the slides:

- *Student are supposed to be able to select their courses from course-catalogue and register for maximum 6 courses.*
- *Each course, maybe offered in 1 or more sections (offerings), must have a minimum of 8 students to run, and may have one or more prerequisite.*

Task 2 - Implement a console-based menu to properly call the methods form the backend. The program should keep on running (i.e. should keep presenting the user with the menu) until the user quits.

To keep things simple, you can create few objects of types Course, Student, and Course Offering in main. (i.e. simulate having a file or database) to test the program functionality. Your menu should contain the following items:

1. Search catalogue courses
2. Add course to student courses
3. Remove course from student courses
4. View All courses in catalogue
5. View all courses taken by student
6. Quit

Important note 1: Your fronted implementation must only take place in the main method of class FrontEnd. You must NOT implement any of the functionality of the above menu in your frontend. i.e. Your frontend must only be used to call methods from the backend.

Important note 2: You must implement any missing functionality in your backend. For example, you must add code to implement remove course.

Javadoc is not required for this exercise.

What to hand in: Submit your source code and all required files for this exercise in a folder with exercise number as name, this folder will be part of the final compressed file submission.