## CS 140 – Spring 2014

# Programming Project 3 – Simulation of a standalone convenience Store using Arrays

### Objectives

In this programming project, you will write a program to simulate a standalone convenience store. This will provide you with the opportunity to put together nearly all of the material you have learned so far to create a single program. Most of what you are asked to do is very similar to what you did in the lab assignments, so look back at those labs for ideas. You will be writing code for 2 classes (Invoice and LineItem).

Start by studying the Product.java class (especially how to print values using currency format) and the output file – your program should produce a similar output.

* *Read the Invoice.html file to understand what methods need to be written.*
* *Read the LineItem.html file to understand what methods need to be written.*
* *Cashier.java is the driver module and it has only the main method. You do not have to modify anything in this file.*
* *Product.java is given to you and you do not need to modify anything in this file.*

***The specifications and/or code may be incomplete, or unclear. We expect you to ask questions if you do not understand. This is a common situation – even professional software developers need to ask the customer questions to clarify what the customer wants.***

### Specifications

Your convenience store will have four classes Product, LineItem, Cashier and Invoice. Product and Cashier are given to you – you do *not* need to modify these files. Use them the way they are.

**LineItem Class**

You have to design this class. This class represents one line item in the invoice (receipt). Each LineItem object has two instance variables a ***Product*** and a ***quantity***. This class has 6 methods – study LineItem.html for their functionality.

**Invoice** **Class**

You have to design this class. This class simulates an Invoice (receipt) at a store for one customer. Each invoice has collection of LineItem objects represented as an array of LineItem. This class also keeps track of the number of LineItems (similar to Lab 6- go over the solution to Lab6 before attempting this project).

### This is an Individual Assignment – No Partners

As this is a Project (and not a Lab) you will be working on your own, not with a partner. You should not be sharing your code with anyone else, other than the instructor.

You will need to fork your own private Project3 repository on GitLab for this project. The only person who should have any access to your repository is your instructor.

You can ask questions on Piazza about the requirements and specifications, about setting up your repository on GitLab, about using Git to send you code to the instructor, and general questions about how to write your code. However you should not be posting sections of code and asking others to find your errors.

### Deliverables

Be sure that you have Javadoc-style comments for both of your classes and all of your methods. Write comments for lines of code that you think need explanation. Be sure that you have indented consistently.

The instructor will pull your Project3 from your GitLab repository to grade it.

Make sure:

1. You have pushed all changes to your shared repository. (I can’t access local changes on your computer.)
2. You have added your instructor as *Master* to your shared GitLab repository.

### Extra Credit

You can use ArrayLists to represent LineItem objects. Only the Invoice class will need to be modified to reflect this change. To receive extra credit, your program has to be flawless. You can create a folder ExtraCredit and place your project there. Just to reiterate, only the Invoice.java need to be modified for this. LineItem.java, Cashier.java and Product.java will be the same from Project3

### Due Date/Time

Your project (including the extraCredit) must be pushed to GitLab by Friday, 11th April 2014 at 11:59pm.