### CS 305: HW 0 Spring 2016

**During the semester, there is free tutoring available to you for CS 305. Sundays – Thursdays, 7 – 9 pm in Shiley 208. Look for Caleb or Nerissa.**

**Due Date:** Friday, Jan 15 at 11:25 AM. Your code files (.java files) and short answer questions should be submitted as a single zip file to Moodle.

This assignment is meant to be *review* of what you learned in CS 203. It reviews classes, instance variables, constructors, methods, arrays, loops, and selection. If this assignment is difficult for you, please take the time during the first week of class to practice your programming skills ☺. You may want to see Tammy to get more practice and to discuss strategies in order to be successful in CS 305.

This assignment should be completed **individually**.

**Specification**

The implementation should include 3 Java classes:

* + Main (provided to you)
  + Receipt (you should implement this class)
  + RetailItem (you should implement this class)

Here are the class diagrams for the two classes you should implement:

RetailItem

------------------------------------------------------------

-number: int

-name : String

-price : int

-------------------------------------------------------------

+RetailItem(number : int, name : String, price : int)

+print() : void

+getPrice() : int

Receipt

-------------------------------------

-items : RetailItem[]

-numItems : int

-maxItems : int

-------------------------------------

+Receipt(maxItems : int)

+addItem(item : RetailItem) : int

+printReceipt() : void

Recall that – means private and + means public. Thus, the instance variables in the classes should be private and the constructors and methods should be public.

*Receipt Class*

This project models a Receipt as a collection of RetailItems. A Receipt object also has maxItems (the maximum number of RetailItems a Receipt object can contain) and numItems (the current number of RetailItems stored in the Receipt object).

The constructor for the Receipt class should check the parameter maxItems. If MaxItems <= 0, then set the instance variable maxItems to 10. If the parameter is > 0, set the instance variable maxItems to the value of the parameter maxItems. Then, create a new array of RetailItem objects of size maxItems, and set numItems to 0.

The addItem method should check to see if numItems >= maxItems. If so, this item cannot be added to the Receipt, so the method should print “Cannot add another item to receipt. Maximum size has been reached.” If this is the case, the method should return the value -1. Otherwise, if there is a place to store the item, then put item in the array at position numItems, increment numItems by 1, and return 0.

The printReceipt method should print “Receipt:” followed by a newline. Then, it should print every item in items, using the print() method in the RetailItem class. It should calculate the total of the items in the Receipt, using the getPrice() method in the RetailItem class. It should then print the total in the format “Total: $dollars.cents”. Note that the prices in RetailItem are stored as ints, so you will need to convert cents to dollars and cents.

*RetailItem*

The RetailItem class models a specific item that could be scanned at a store. For this project, we will keep it simple. A RetailItem has a number (barcode or ID number), a name, and a price (stored as cents).

The constructor should take a number, name, and price and assign the instance variables accordingly.

The print method should print the item in the format:

Item <number> <tab> <name> <tab> $dollars.cents

(see example output below for the format)

The getPrice method should return the value of the instance variable price.

*Error-checking*

All methods should do proper error-checking of the parameters. If the parameter value does not make sense (for example, a negative price), then the code should set an appropriate value such as 0.

*Documentation*

The code should be well-commented, well indented, and include enough whitespace to be readable.

**Example Output**  
The following should be printed to the console/screen when the main method is executed.

Receipt:

Item 1001 Snickers $0.75

Item 3033 Starbucks Blend $6.99

Item 2222 Chips Ahoy $3.29

Item 9212 Large T Shirt $15.99

Total: 27.02

Receipt:

Item 1001 Snickers $0.75

Item 3033 Starbucks Blend $6.99

Item 2222 Chips Ahoy $3.29

Item 9212 Large T Shirt $15.99

Item 1234 Dozen Eggs $3.69

Item 1234 Dozen Eggs $3.69

Item 5545 Clementines $7.99

Item 3222 Orowheat bread $4.99

Item 5122 Red grapes $5.49

Total: 52.87

Cannot add another item to receipt. Maximum size has been reached.

Receipt:

Item 1001 Snickers $0.75

Item 3033 Starbucks Blend $6.99

Item 2222 Chips Ahoy $3.29

Item 9212 Large T Shirt $15.99

Item 1234 Dozen Eggs $3.69

Item 1234 Dozen Eggs $3.69

Item 5545 Clementines $7.99

Item 3222 Orowheat bread $4.99

Item 5122 Red grapes $5.49

Item 1121 Vanilla yogurt $0.75

Total: 53.62

Receipt:

Item 3033 Starbucks Blend $6.99

Item 5545 Clementines $7.99

Total: 14.98

Cannot add another item to receipt. Maximum size has been reached.  
**Additional Enrichment**  
If you have time, feel free to add the components to your program. Please document the additional features in your code and your written summary. This is not extra credit, but a chance for you to do some exploration.

* Add a deleteItem method to the Receipt class. This should delete the item based on the item’s number and shift the contents so the array stays “packed”.
* Add more attributes to the RetailItem class, such as “category” (food, apparel, home, etc.). You may need to modify the Main class to account for these new attributes.
* If you have RetailItems with attributes, print the receipt items according to categories instead of the order that the items are stored in the Receipt.
* Print the receipt items in sorted order from cheapest to most expensive product.

**Logistics**

1. Download the starter code Main.java located on Moodle.  You may use BlueJ or another IDE to develop your Java classes. You will need to add two more classes. You MUST name them Receipt.java and RetailItem.java.
2. Remember that there is a difference between integer division and division using doubles in Java.
3. When you are finished and ready to submit:
   1. Create a new folder called username\_HW0.
      1. (For example, Tammy’s would be vandegri\_HW0.)
   2. Copy your Receipt.java file into this folder.
   3. Copy your RetailItem.java file into this folder.
   4. Copy your Main.java file into this folder.
   5. Copy your HW0Summary.docx file into this folder.
   6. Zip the folder by right-clicking and Send To->Compressed Folder (on Windows). There should also be a zip utility on Mac.
4. **What to turn in:** Submit your **username\_HW0.zip** file to Moodle before the due date and time. After logging into learning.up.edu, navigate to CS 305. You will find a link to submit this file. You do not need to print anything.

**Grading Guidelines (total of 20 points)**

Your code files will be graded on a scale from 0 to 7 in two separate categories:

* Code Quality: Design, commenting, whitespace, readability, using private and public correctly
* Code Operation: Does code do what is listed in the specification?

Your summary report will be graded on a scale from 0 to 6 based on completeness and clarity.

#### HW 0 Report Guidelines and Format – use the template provided below

#### -------------

#### Name:

#### CS 305 HW 0

**1. (2 pts) Testing:** Describe how you tested and evaluated your program. If your program does not meet the specifications, please note these differences. Include a representative printout from executing your program with the original main method.

**2. (1 pt) Evaluation**: Evaluate the quality of your code: Is it readable and easy to follow? Do the parts make sense? Are your comments helpful?

**3. (3 pts) Questions**:

1a. At the bottom of the main method, uncomment the printing of the objects fred and shop. Run the main method. What is printed for these objects?

1b. Re-run the code again. What is printed for these objects?

1c. What do these values represent?

2. In the main method, you will see a comment after the shirt item is added to shop. Draw the shop’s items array and its contents at this point of the code execution. If an array cell is empty, label it with the word “null”.

3a. Describe the most challenging aspect(s) for you when writing this program. (Hopefully, this assignment is review for you from CS 203.)

3b. How much time did you spend in total on this homework assignment (including the report)?

**Appendix A**: Copy and paste your java code here (use Courier New 9pt font so the characters line up correctly)