ITSE 2321 – Object-Oriented Programming (Java)
Bonus Program

Write a class named *Retail_Item* that holds data about an item in a retail store. The class should store the following data in attributes:

- Item Number
- Item Description
- Units in Inventory
- Price

Create another class named *Cash_Register* that can be used with the *Retail_Item* class. The *Cash_Register* class should be able to internally keep a list of *Retail_Item* objects. The class should have the following methods:

- A method named *display_menu* that displays the snapshot of data in the store.
- A method named *purchase_item* that accepts a *Retail_Item* object as an argument. Each time the *purchase_item* method is called, the *Retail_Item* that is passed as an argument should be added to the list.
- A method named *get_total* that returns the total price of all the *Retail_Items* objects stored in the *Cash_Register's* internal list.
- A method named *show_items* that displays data about the *Retail_Item* objects stored in the *Cash_Register* object's internal list.
- A method named clear that should clear the *Cash_Register* object's internal list.

Write a TestBonus class that uses ArrayList to store *Retail_Item* objects. Each object in the ArrayList should hold data about an item in the retail store (Item Number, Item Description, Units in Inventory, and Price). The test class should also use the *Cash_Register* objects to allow the user to select several items for purchase, using a menu. See the program menu on the second page. These objects must be stored in an ArrayList. When the user is ready to check out, the program should display a list of all the items he or she has selected for purchase, as well as the total price, taxes (8.25%), and the final price. The data for the retail store is available in the file **Bonus.txt**. **(Note: This program might be tested with a different file so allow the user to enter the name of the file.)**

Allow the user to run the program as many times as possible until a sentinel value, **less than zero (0)**, has been entered for the selected item. No input, processing, or output should happen in the main method. All work should be delegated to other non-static methods.

You will not get credit if the program is not written or does not work as expected.

Test the program with your own data. Create a folder named, **fullname_bonus**. Copy your source codes and the output file to the folder. Zip the folder and upload it to Blackboard.

When you upload your source code and output to Blackboard, schedule a time with me to answer a few questions about your code. You must answer all questions correctly to get credit for the program.

Program Menu

1. Pants
2. Jeans
3. Shirt
4. Dress
5. Socks
6. Sweater
7. Jacket
8. Suit
9. Swimsuit
10. Skirt
11. Clear Cash Register
12. Show Inventory
13. Check Out