Report on CA 2

Simon Kennedy

C19496436 DT211c / TU857

CLASSES: This assignment contains 4 classes. ShoppingCart, Customer, LoyalCustomer, and Items. The ShoppingCart class contains information such as the contents of the cart, the total cost and the shipping cost. It contains a function to print the contents of the cart. If the value associated with the product key is above 0, this means the user has added this item to the cart and adds it to a list. It then prints the list with the leftover punctuation removed. The \_\_str\_\_ method is used to print the shipping cost equal to 2 decimal places.

The Customer class inherits from the ShoppingCart class. It contains a customer type to identify what items they can access. This information is communicated through the \_\_str\_\_ method. The two other functions add\_item and remove\_item are used to add or remove items from the cart. The add\_item function takes the relevant object, the quantity to be changed and a variable to confirm what item is being changed. It then increases the quantity of the item, and stores it in the shopping cart dictionary. It calculates the total cost from the current\_item.price attribute from the class, and calculates the shipping as 10% of the total cost. Remove\_item works in reverse, taking the quantity times the item price from the current total.

LoyalCustomer is a class that inherits from Customer. The only differences are that LoyalCustomer has an extra attribute to identify them as a loyal customer, and a \_\_str\_\_ method that communicates this.

The items class is used for items to sell. It contains the attributes name, stock and price. The add\_stock and remove\_stock functions are as they are named, They increase and decrease the stock based on the quantity argument. The remove\_stock function has an extra check to change the stock value to 0 if it is less than 0. The \_\_str\_\_ method returns the attributes. The \_\_lt\_\_ and \_\_gt\_\_ are used so that stock can be compared with integer values, specifically “quantity”.

USER MANUAL:

Before you can use the other options, you must first create a customer. Loyal customers have access to the exclusive “Case” item, whereas the other customers can only access the other items. Once a customer is created, the user can view the available items by pressing 2. Users can add or remove items with 3. All available items are listed, but exclusive items cant be added, and will produce an error. The current shopping cart can be viewed by pressing 4. The user can checkout by pressing 5. This will show the current cart, subtotal, shipping cost, and total. The use can then confirm if they wish to checkout. If they checkout, they are thanked for their purchased and the program ends.

PROBLEMS AND CHALLENGES IN DEVELOPMENT:

When using the \_\_str\_\_ method to print the attributes, it would add “None” to the end of the string. To fix this, the “print\_items” function was created in the ShoppingCart class.

Keeping the main function readable was a big challenge. Python doesn’t have a switch statement, so it can be a bit dense with the amount of if, elif and else statements. Keeping a large amount of whitespace between options helps in readability, but pycharm gives warnings If there is too much whitespace between lines of code.