Module 1 Capstone - Candy Store Cash Register

You've been asked to develop a cash register application for the Silver Shamrock Candy Company. This application will be used by Silver Shamrock Sales Representatives when making sales to corporate customers. It will keep track of inventory, sales, and money received or returned.

Application Requirements

1. The application should start by showing the Main Menu. The main menu should display the following options:

   ```
   (1) Show Inventory
   (2) Make Sale
   (3) Quit
   ```

2. The application needs to track the following types of inventory items: :
   `Chocolates, Sours, Licorice, Hard Candy`

   - Each Inventory item will have a Name, Price, and indicator whether or not it is individually wrapped.

3. The inventory is stocked via an input file. You will be given a file to test with, but DO NOT hardcode specific values as the inventory file on release will be different.

   - The inventory should automatically restock when the program is run. It should not restock at any other time

4. When the Main Menu is being shown and the user selects `(1) Show Inventory` the Inventory Items should be shown along with the quantity remaining. The Main Menu should then be shown again so the user can make a new choice.

   SAMPLE INVENTORY DISPLPAY

   The items in the file may be different.

   ```
   Id      Name               Wrapper  Qty      Price
   C1      Chocolate Bites    Y        100      $1.10
   C2      Coco Bar           N        42       $2.75
   H1      Pickle Suckers     Y        SOLD OUT $1.45
   H2      DotNet Blocky      Y        100      $1.35
   H3      Java Greener       N        100      $2.56
   H4      Java Bluemataz     N        100      $3.10
   L1      Anise Bite         Y        100      $0.35
   L2      Strawberry Straw   Y        27       $1.25
   S1      Sweet Sours        N        SOLD OUT $0.25
   S2      Bitter Chews       Y        78       $0.90
   S3      Sour Gummy         Y        1        $0.15
   ```

   - Each inventory item should show the inventory id, name, Y or N indicating if it is individually wrapped, quantity in stock, and price.
   - Each item in the inventory system can have a maximum quantity of 100 in stock.
   - Every item is initially stocked to the maximum quantity.
   - If an item is out of stock it should not show 0 as the quantity, instead, it should show the text SOLD OUT.
   - The list of inventory items displayed must be formatted so columns align, and all items are listed in `alphabetical order by inventory id` (eg. A1, A2, A3, A4, B1, B2, etc.). See SAMPLE above (The inventory file will contain different items)

5. When the user selects `(2) Make Sale` on the Main Menu they are guided through the purchasing process via a sub-menu:

   ```
   (1) Take Money
   (2) Select Products
   (3) Complete Sale
   Current Customer Balance: $0.00
   ```

6. The purchase process flow is as follows

- Selecting `(1) Take Money`

  - The user can repeatedly add money to the balance in **whole dollar amounts up to $100**. The user should be

informed if they try to enter an invalid amount.
    - The user should be able to add money multiple times throughout the sale so they can replenish the customer's balance. However, at no time should the current balance be allowed to go above $1000.
    - The Current Account Balance indicates how much money the customer has available in their account to make purchases.
    - Any time money is entered using the add money option, the updated current balance should be displayed showing the new balance.

- Selecting `(2) Select Products` allows the user to select a product and quantity to add to the customer's cart.

    - The list of products should be redisplayed (the same listing as shown when Show Inventory is selected from the main menu) so that the user can see a list of products from which they can select.
    - If the inventory id the user enters does not exist, the user should be informed that the product they selected does not exist and then returned to the Make Sale menu (sub-menu).
    - If a product is sold out, the user should be informed and returned to the Make Sale menu (sub-menu).
    - If not enough of the product is in stock for the quantity the customer requested then the user should be informed there is 'insufficient stock' and returned to the Make Sale menu (sub-menu).
    - If the customer's current balance is not enough to purchase for the quantity the customer requested then the user should be informed there are 'insufficient funds` and returned to the Make Sale menu (sub-menu).
    - If a valid product is selected, and there is enough money available, then the item should be added to the customer's cart and the cost immediately removed from the customer's balance.
    - After the item is added to the cart, the user should return to the Make Sale menu(sub-menu) and the balance on the Make Sale menu (sub-menu)should be updated to show their updated current balance.
    - Removing items from the cart is out of scope for this release, so once a product has been added the product can not be removed from the cart.

- Selecting (3) Complete Sale ends the sale and should show a receipt for the products purchased and the change the sales clerk should return to the customer. This report should be displayed with the following:

    - Change will be reported by showing the denominations the clerk should return to the customer in nickels, dimes, quarters, ones, fives, tens, and twenties. **Important: return change using the smallest amount of bills and coins possible and only in the denominations listed.** The change should be displayed on the screen showing each denomination and the number of each (1 dime, 2 quarters, 1 one, 1 five, 2 twenties). The formatting is up to you as long as it shows this information.
    - The system should be prepared for the next sale by resetting the customer balance to 0 and emptying the shopping cart.
    - The receipt should show the name, product type, and the number of each product purchased, the cost of each item, the total cost for the quantity of that item purchased, the total amount of the sale, and the change to be given including demoninations.

  The Product Types should be displayed using the following descriptions

  Chocolates: `Chocolate Confectionery`

  Sours: `Sour Flavored Candies`

  Licorce: `Licorce and Jellies`

  Hard Candy: `Hard Tack Confectionery`

  The receipt formatting is up to you, but it must appear in columns with the data lined up and currency should be displayed with a dollar sign and 2 decimal places.

  EXAMPLE ON SCREEN REPORT

```
10  Chocolate Bites   Chocolate Confectionery  $1.10  $11.00
22  Strawberry Straw  Licorce and Jellies      $1.25  $27.50
2   Pickle Suckers    Hard Tack Confectionery  $1.45  $2.90
50  Sweet Sours       Sour Flavored Candies    $0.25  $12.50
12  Anise Bite        Licorce and Jellies      $0.35  $4.20

Total: $58.10

Change: $91.90
```

```
(4) Twenties, (1) Tens, (1) Ones, (3) Quarters, (1) Dimes, (1) nickels
```

- After the sale is completed, the system should be **returned to the MAIN MENU and THE APPLICATION SHOULD CONTINUE TO RUN UNTIL THE USER SELECTS TO QUIT THE PROGRAM.** Once returned to the main menu if show inventory is selected it should show the updated quantities as reduced by the previous sale.

7. All purchases must be audited to track items and amounts for each sale.
    - Each purchase should generate a line in a file called Log.txt
    - The Log.txt file should persist (appended to, not overwritten) when the program starts.
    - The audit entry should include the date and time, action taken, the customer's balance before that action and and new customer balance after the action was completed
    - Actions Taken may be:
        - MONEY RECIEVED
        - CHANGE GIVEN
        - NUMBER_ORDERED PRODUCT_NAME PRODUCT_CODE
    - The audit entries should be in the format:

```
09/28/2021 12:00:00 PM MONEY RECIEVED: $50.00 $50.00
09/28/2021 12:00:15 PM MONEY RECIEVED: $100.00 $150.00
09/28/2021 12:00:20 PM 10 Chocolate Bites C1 $11.00 $139.00
09/28/2021 12:01:25 PM 22 Strawberry Straw L2 $27.50 $111.50
09/28/2021 12:03:12 PM 2 Pickle Suckers H1 $2.90 $108.60
09/28/2021 12:03:57 PM 50 Sweet Sours S1 $12.50 $96.10
09/28/2021 12:05:02 PM 12 Anise Bite L1 $4.20 $91.90
09/28/2021 12:05:58 PM CHANGE GIVEN: $91.90 $0.00
```

Unit Tests

Unit tests are required to demonstrate that the application works as expected. **The Menu does not need to be unit tested.** Hint: You should, however, test inventory levels, user balances, and any public method that takes a clear input and can return an expected result. etc. DO NOT save unit testing until the end!

Inventory Data File

The input file that stocks the products is a pipe (|) delimited file. Each line is a separate product in the file and follows the below format.

Columns:

```
Type|Id|Name|Price|Wrapper
```

- **Product Type:** The type of the item: (CH) Chocolates, (SR) Sours, (LI) Licorice, (HC) Hard Candy.
- **Inventory Id:** The inventory id used to identify the product.
- **Product Name:** The display name of the product.
- **Price:** The purchase price for the product.
- **Wrapper:** Indicates if the item is indiviually wrapped. `T` if wrapped and `F` if is not

An example input file has been provided in your repository. This input file IS AN EXAMPLE of what an inventory file will look like and while it should be used for development, it may not be the final file used with the system when released. You can assume that any file you receive will always have the same name, be in the same file location, and in the same format, but the actual inventory ids, product names, prices, and whether it is wrapped may change.

Total System Sales Report (Bonus Requirement)

The total system sales report file is pipe-delimited for consistency and should be written to a file named TotalSales.rpt

Each line is a separate product with the total number and amount sold for the applicable product.

At the end of the report is a blank line followed by **TOTAL SALES** $dollar amount indicating the gross sales of the system.

The Total System Sales Report should be persisted (saved) and not reset when the application is restarted.

Example Total System Sales Report

```
C1|Chocolate Bites|150|$165.00
S2|Bitter Chews|2034|$2034.90
L2|Strawberry Straw|211|$263.75
H1|Pickle Suckers|18|$26.10
C2|Coco Bar|876|$2409.00
S1|Sweet Sours|73|$72.75
L1|Anise Bite|12|$4.20

**TOTAL SALES** $4735.70
```