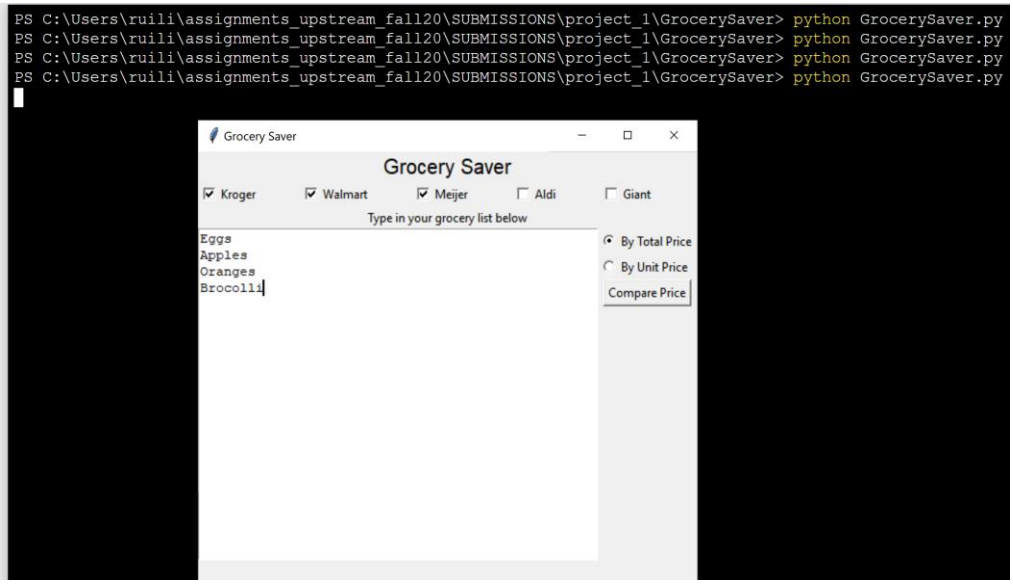
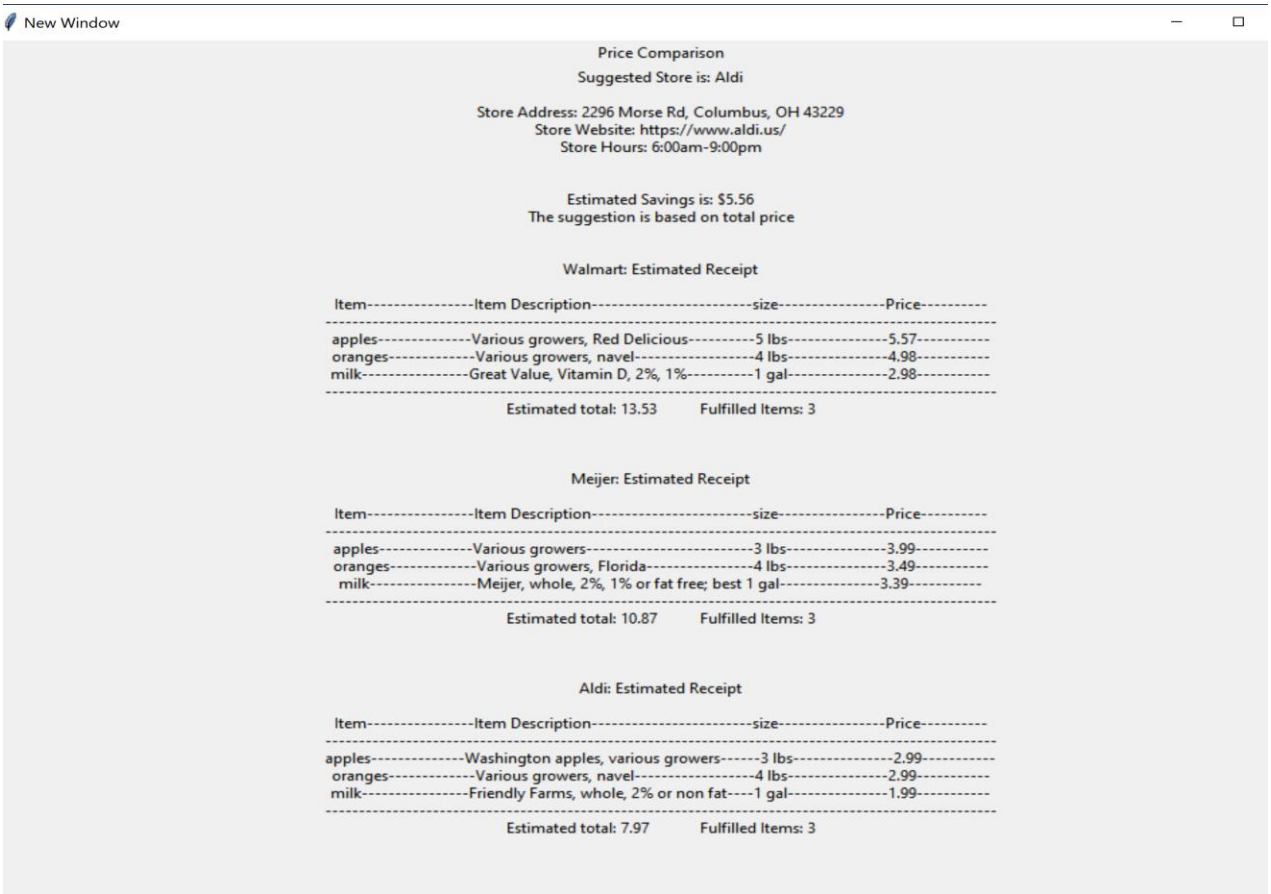
Project 1 Reflection

Instructions about how we should go about testing and using your project

- Go to terminal and navigate to the GrocerySaver folder
- Type in python GrocerySaver.py to launch the program



- Type in Grocery items
- Select at least a store (or multiple stores)
- Select the saving option: by total price or by unit price
- Click on the “Compare Price” button
- The Price Comparison window will pop up and display suggestion



What you completed and anything that you didn't complete in the project that you would finish later

This project is working as designed as the proposal. Using **tkinter** for UI, **sqlite** for database and **requests** for API calls.

Main functions:

- The Grocery Saver UI takes the grocery lists entered by the user and save the grocery list to a csv file for later use
- The Grocery Saver UI takes the store selections made by the user and save the store list to csv files for later use
- User input validation – if the grocery list is empty or store is not selected, notification window will pop up
- By Total Price and By Unit Price option on UI is for user to pick the price saving model
- Once clicking "Compare Price" button, the program will loop through each store and check price of each item on the shopping list.
- Check price function supports both checking price from the SQLite database or checking from the store API

Highlight of the classes

All functions are wrapped in classes, here are some inheritances and polymorphisms

Inheritance and Polymorphism

- StoreShoppingList is an inheritance to the shoppinglist class – each store inherits the parent shopping list but has its own fulfillment.
- SuggestionbyTotal is an inheritance to Suggestion class.
    - Inherit parent attributes, extend parent methods
    - get_estimated_saving method is basing on the total price.
    - get_suggested_store method is based on number of fulfilments, among the stores that fulfills the most items, comparing the total price and suggesting the store with the cheapest price
    - repr method formats the estimated receipts including item, description, size, price
- SuggestionbyUnitTotal is an inheritance to Suggestion class
    - Inherit parent attributes, extend parent methods
    - get_estimated_saving method is basing on the total unit price.
    - get_suggested_store method is based on number of fulfilments, among the stores fulfilling the most items, comparing the total unit price and suggesting the store with the cheapest price
    - repr method formats the estimated receipts including item, description, size, price, unit size, unit type

Discuss challenges you faced and how you overcame them

1. It was not easy to get the API working. I've got Kroger API working and successfully call Kroger API to retrieve grocery item information. Not all grocery stores have API available. Some stores (like Walmart) have API, however, it doesn't allow new application to register for API due to holiday season.

   I overcame this problem by switching gear to load grocery price by store to SQLite database. I developed the databases and the SQLite connections to retrieve grocery item information.

   Also, I stored the indicator whether a store has API, and check DB price or API price basing on if store API is available. This is for ease and flexibility of API extension in the future . If the store API is available in the future, I can easily extend and switch the API on.

2. Get the repr to display in a nice format was also challenging.
   It took me a while to display the estimated receipt in aligned columns. I overcame the issue by using .ljust(length, "-"), so the columns are all in the same length and left aligned.

Things I want to finish later:

Because I only got Kroger API working, so right now the API call codes are in get_price_from_API method in the Item class. To make the code more elegant, I need to wrap API related information in separate API class. To improve the code, I can create a new class – API. API url, json response, access token can be attributes of the API class, and get API response can be a class method.