

Final Project Report
Joseph Nelson Farrell
CS 5001
04/24/2023

Project Description:

In my project I implemented a program reads in csv files, executes a simulation, tracks the sales and inventory levels of a restaurant, and produces a pdf export. I designed 6 classes to accomplish this: *Inventory_Item*, *Inventory_List*, *Recipe*, *Menu*, *Order_List* and *Restaurant*. I also designed 2 functions that read in csv files and populate the classes. One of the functions creates a list of *Inventory_Item* objects and then populates an *Inventory_List* with those objects. The other function creates a list of *Recipe* objects and then populates a *Menu* object with the *Recipe* objects.

Once the *Menu* object and *Inventory_List* objects are created, which are both essentially lists, they are used to create an instance of *Restaurant*. The final class, *Order_List* is generated as an attribute of *Inventory_List*. Within the *Restaurant* class there is an attribute *day()* that has a positive integer parameter, *patrons*. This method executes a simulation of patrons coming to the restaurant and ordering random items. The majority of the simulation occurs within the *Inventory_List* class in the method *ordered*. The attributes attached to the various classes track the sales in dollars, the number of items sold, the amount of the ingredients used, the amount of the ingredients held in inventory, etc. After the simulation executes, the program produces a summary report pdf displaying this information in various lists and graphs. The report is generated by a helper function designed outside of the classes.

In this project I was able to refine many of the skills we learned throughout this semester. Including object-oriented programming, writing functions, using various data structures and data types, importing and exporting files, creating graphics, etc. As I designed the program, I was also able to practice and utilize the three forms of defensive programming covered in the course, asking permission with repetition, asking permission by raising errors, and asking forgiveness. I also implemented extensive tests which helped inform my defensive programming.

Changes From Planning Document 2:

Fundamentally, there were not a lot of changes between planning document 2 and implementation. The changes that I did make were related to how the classes were be set up and how they function. I decided on separate classes for the items and lists (*Inventory_Item/Inventory_List* and *Recipe/Menu*) because they seemed structurally different, one being an item and the other being a group of those items. I also added a class for *Order_List* when in reality maybe it could have been attribute. I made it class because I wanted to give it methods that only pertained to that object. I structured the program in this way so that classes would be smaller and have more specific attributes and methods.

Another difference is that I intended to execute the bulk of the simulation in the *Restaurant* class, but I ended up performing most of the simulation in the *Inventory_List* class. This seemed to come about naturally during implementation, but upon reflection, or if I this were a real-world program, I would probably move more of the simulation to the *Restaurant* class. In the end, the program is able to accomplish everything that was intended.

Reflection:

During this project I learned a lot about end-to-end implementation of a program. Including planning what classes and functions I would need and how the data will be stored. I learned how to create classes with attributes and methods that work cohesively with each other within a larger program. I learned about the nuances between different data structures and how and when to use the various options. I learned how to write functions that import files and then create instances of classes with the data contained in the file. I practiced adding methods to an existing class using inheritance. I also learned how to create graphics and add the graphics, with text, to output pdf files. If I could go back and start over, I think the only major change I would make is moving where a lot of the simulation occurs. Potentially, I would remove the *Order_List* class.

Acknowledgements:

1. W3schools.com
 - a. Matplotlib bar plot
 - b. Pandas
 - c. Unittest with isinstance
2. Stackoverflow.com
 - a. Reading in csv files
 - b. Populating objects from csv
 - c. Adjusting axis labels in matplotlib
 - d. Creating pdf outputs
3. Realpython.com
 - a. String detecting
4. Docs.python.org
 - a. assertRaises
 - b. Dictionaries
 - c. Unittest
5. Geeksforgeeks.org
 - a. Raising FileNotFoundError
 - b. Dictionary syntax
 - c. Class inheritance
6. Youtube
 - a. Pdf output, <https://www.youtube.com/watch?v=v-ZP8BLmJlk>
 - b. Pdf output, <https://www.youtube.com/watch?v=q70xzDG6nls>
7. Class Lecture
8. Class modules
9. Greg Valcourt
 - a. Reading in csv
 - b. Populating objects from csv