Function Fanatics Store Inventory Tracking System

Introduction: Our Program's purpose is to track inventory sold and profits for a store. We created the program to allow a user to choose between an employee or customer menu for a store, facilitating the tracking of prices and items in stock and sold in the store for the customer and employees.

Motivation: We decided to create this program because we thought it would be a practical application that could be potentially used in the real world.

Organization: The project is divided into a few main parts, The first being the main function which comprises the code to open the .txt file needed to start the program and embedded for loops to create arrays for our prices, items, and product string. Then it goes through a while loop and switch statement for our main menu. From there the program is comprised of several functions starting with the employee and public menus. These 2 menus are similar to the main menu where we used while loops and switch statements to call on various functions for each respective menu. From there we created the item menu functions which contain more while loops and switch statements for the public menu so the user can choose between a subset menu of items for the final purchase and as well track the stock and items sold for each item. We then created functions for the employee menu which consist of embedded for loops to track total sales and remaining stock. We then passed all the necessary arrays and int values to each function to make them all work according to their role in the program. The formatting of the program was mainly done with setprecision, setw, and /t so most of it was done automatically and not hard coded. We then created a pointer that pointed to the total in the public menu that was incremented by price. Sales were incremented each time an option was selected then outputted those totals after the customer is done with purchasing items to show them a receipt. To print out an employee receipt to an external file we created an output file named Receipt and then had 2 embedded for loops to pass the itemized list and remaining stock to the file. We then copied those 2 embedded for loops to then print the list and stock to the screen as well.

Testing: While building the program we tested each menu we implemented after creating them to make sure that the program would run properly. The same was done when creating the functions for the items. We also tested for error cases for input validation. Once we found that a piece of code worked we would copy that section and reuse it for other parts of the program. Doing so allowed us to build each part of the program fairly quickly and mostly bug-free. We also figured out how to use the push and pull features from GitHub which made passing code and files from each person much easier and faster.

Conclusion: In conclusion, Our program allows a customer to purchase items how much of each item remains, and its price. After they are done purchasing it will show how many items they have purchased and how much is owed. It also allows an employee or store owner to see how much they have made from sales, itemized remaining stock, and print out a receipt to an external file so they can see the total sales and itemized sales and stock. We believe the project to be a success as we were able to accomplish the function we desired the program to have. As well we learned how to push and pull from GitHub which was very useful in passing code to each other. The Project took around 16 hours to complete with us meeting multiple times to collaborate. Overall we believe that this project was challenging but fun at the same time. Learning how to use GitHub was complicated but once we figured it out, it was very cool to see how it works. If given additional time we could iterate and improve our program further.