FOOD PROJECT V3
NEW IMPLEMENTATIONS
- RESTAURANT IMPLEMENTATIONS
  - NEED TO ADD A WAY TO FEATURE MEAL OPTIONS
  - ADD A MAP OF MEAL OBJECTS THAT CONTAINS PRICES AND PRICES AVAILABLE
    - I might not need to do this as the implementation for obtaining meal informations for orders can just be done using a csv document
- ADMIN IMPLEMENTATIONS
  - ABLE TO UPDATE RESTAURANTS MEALS AND PRICES
    - NEED A WAY TO SAVE, LOAD, AND ALTER A FILE THAT WOULD STORE THE MEAL OBJECTS OF A DEDICATED RESTAURANT
  - ABLE TO EXPORT ALL INFORMATION IN A FILE
- CUSTOMER
  - ABLE TO SELECT SPECIFIC MEALS AND EVEN DUPLICATES FROM ONE A SINGLE RESTAURANT IN AN ORDER
  - ABLE TO SELECT MEALS OR SEE A MEAL MENU

GAMEPLAN

IN ORDER TO IMPLEMENT MENUS, I WILL END UP CREATING CSV DOCUMENTS FOR EACH INSTANTIATION OF A RESTAURANT.

I WILL ALSO END UP HAVING THE ADMIN PUT A MINIMUM OF 3 MENU ITEMS FOR EACH RESTAURANT WITH PRICES. IN MY OPINION, A RESTAURANT THAT PROVIDES LESS THAN THREE MENU OPTIONS IS NOT A RESTAURANT, BUT A FOOD STAND. JUST ME THO. THIS HAS SINCE CHANGED AND IS NOW UP TO THE ADMIN TO PUT WHATEVER AMOUNT OF MENU ITEMS HAVING AT LEAST AN ITEM. DON'T WANT TO BE EXCLUSIVE TO FOOD STANDS EITHER.

I WAS THINKING THAT I WOULD MAKE A GLOBAL MEAL VECTOR THAT WOULD BE ABLE TO TEMPORARILY STORE THE INFORMATION (MEAL OBJECT)THAT IS PARSED FROM THE PROPER DEDICATED CSV FILE. THE VECTOR WOULD PARSE IN THE DATA WHEN IT NEEDS TO, HAVE IT'S IMPLEMENTATIONS IN THE PROGRAM DONE, THEN WHEN IT'S DONE, THE VECTOR WOULD PARSE THE DATA INTO THE CSV FILE TO SAVE ANY CHANGES DONE TO IT.

A MEAL OBJECT WOULD BE ABLE TO STORE THE MEAL NAME, AND THE MEAL PRICE IN ORDER TO STORE THE MEALS IN AN OBJECT TO RETRIEVE THE INFORMATION STORED THERE.

THE UPDATES WOULD BE DONE WITHIN THE TEMPORARY VECTOR WHENEVER THE RESTAURANT WAS LOADED IN ALONGSIDE IT'S MENU. ADMIN WOULD HAVE IMPLEMENTATION TO ALTER THE DATA STORED IN VECTOR TO BE SAVED FOR LATER.

EXPORTATION OF ALL INFORMATION IN A FILE WOULD BE DONE IN A CSV FILE OF EVERYTHING.

I think that this version of the app was slightly difficult. I didn't realize that altering previous implementations and updating old functions would be such a task. Also, having to rewrite large portions of different interacting parts to fit within the new functions was difficult in terms of tasks, forcing me to rely on my old notes. Opened up my eyes to the structure of my code and how messy it is. I should have been more conservative with how much I packed into a single file and spread things out. I wish I was more liberal with my green text. It does help to keep a sense of documentation handy though.