Task 0: Explain what you are doing/ going to accomplish:

I am going to create a website that manages a school canteen, it will manage the stock of items on the menu and students will be able to order food. I will create a program that stores the students name and what they ordered in an array. The inputs will be processed with a python bottle program and displayed on an html website.

Task 1: Sketch interface design

*Draft a rough design for the interface that allows the user to trigger functionality in task 1, while also annotating where the information in task 2 will be displayed. Create another sketch listing the interface widgets used to create the interface.*

View these png images in the planning folder:

* Homepage.png

Task 2: Identify any classes required

*Explain what the class will represent, plus listing what information will be stored in the class and any functions the class will have.*

There will be a class that holds the Items on the menu and the initial stock levels / price of that item

Task 3: Identify information to be displayed

*What information will the interface need to display to the user?*

There will be:

* A homepage with info about what the school canteen is, general info like open times and location.
* A page that displays food item, stock level, price, food items sold.
* A page that takes input from the operator about the students order, input boxes.

These are the first pages I have planned I will add additional pages below if they are needed

Task 4: Identify user inputs

*What program functions can the user trigger through the interface?*

The inputs will be takin in on the page that asks for the students order

Task 5: Identify any constants or existing data if required

The price of items will be constant

Task 6: Identify indexed data structures

Task 7: Determine what calculations are necessary

*Write out the calculations the program will have to compute.*

Task 8: Develop a modular structure for your program

*Describe any functions that the computer program will have, identifying any sub-functions where required.*

Task 9: Define the functions identified

*Describe the functions for both the main program and any classes in terms of input and/or output where required. You may choose to do this with flow charts or pseudo-code (not Python code!). Add in additional steps or explanations using sequential, conditional, iterative statements where required. Identify global and/or local variables.*

Task 10: Address any relevant implications such as usability, functionality, legal/ethical requirements.

Task 11: Document test cases for testing the program

*Document any testing that can be used to test your program. If any input is inputted using the keyboard, describe the expected input, plus any exceptional, boundary or invalid cases.*

Task 12: Refine the plan

*Note any modifications here when iterating through the development cycles.*

Task 13: Document testing

*Show screenshots of your program working with descriptions of each image. These images should test the tests cases listed above.*

Task 14: Evaluation

*How did your version turn out?*