Task 1: Sketch interface design

See images in planning folder, basic structure has been planned out, colour scheme and look will change as I test different colours.

Task 2: Identify any classes required

There will be a class named Comic, this class will hold all the information about the comics

And will look something like this, each attribute holding the comic name, the current stock level of that comic and the amount of books sold

Comic = (“water woman”,”45”(stock level),”0(books sold)”)

Task 3: Identify information to be displayed

Buttons to link to index page, stock levels info page, a sell book success page, a input page (asks the user how many books they want to restock) and a re-stock book success page.

The info that will be showed is the:

Amount of books that have been sold (Books Sold: x)

Amount of books in stock (stock level: x)

The comic name

The comic description

Task 4: Identify user inputs

The Input page will take an int, this will be the number that the user wants to restock. Called something like localhost:8080/restock/x

Task 5: Identify any constants or existing data if required

The comic book name and picture.

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.*