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

In version 1.3 I am making my purchase page and the ability for users to purchase comics

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.*

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.*

Classes:

Comic, stores information on my comics.

Task 3: Identify information to be displayed

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

How many comics in stock, purchase button, which comic you want purchased, how many comics user wants to purchase.

Task 4: Identify user inputs

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

How many comics they want to purchase, and what comic they want to purchase

Task 5: Identify any constants or existing data if required

Comics test data

Task 6: Identify indexed data structures

Task 7: Determine what calculations are necessary

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

Stock minus the amount of stock user has purchased.

Task 8: Develop a modular structure for your program

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

Functions:

Index

Product\_page

New Functions:

Purchase\_page

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.*

I had to make it so that they could only purchase one comic at a time because it was simpler.

I linked my product page to my navbar so you can easily get back to it.

Task 13: Document testing

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

Even though stock got beneath 0 it still let you purchase it. To solve this problem I disabled my button when stock is 0 or lower. For some reason it still did not work so I changed it to link to # instead of my purchase page when stock is 0 or lower.

The background picture would duplicate itself until it filled the screen. I fixed this by getting a 1920 x 1080 image

My purchasing page was not working. This was because it could not identify the comic id. I fixed this by adding comic id in the route of the