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

I am going to create the index landing page

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.*

Not applicable to this version

Task 3: Identify information to be displayed

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

Not applicable to this version

Task 4: Identify user inputs

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

Not applicable to this version

Task 5: Identify any constants or existing data if required

Not applicable to this version

Task 6: Identify indexed data structures

Not applicable to this version

Task 7: Determine what calculations are necessary

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

Not applicable to this version

Task 8: Develop a modular structure for your program

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

INCLUDE Header.html in file

Within python I will use this code from previous project in order to speed up process. This allows me to link a css file

|  |  |
| --- | --- |
|  | @route('/<filename>.css') |
|  | def stylesheets(filename): |
|  | return static\_file('{}.css'.format(filename), root='./assets') |

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.*

Not applicable to this version

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

Usability

Need simple easy to read design, as this program is designed for ages 10+ so there is a wide range of people who should be able to use this.

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.*

Does the website show correctly/how I want it to show?

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.*