Self Assessment - Sofia Wheeler

- ***Critically analyze/evaluate how much time was spent learning syntax & structure, programming concepts vs. actually programming, and how does this reflect on the final quality of your end result.***

About half of the semester was spent learning syntax and structure, specifically dot syntax for OOP. This large amount of time learning syntax lessened the final quality of my end result because though I understood how the concepts functioned, I was not comfortable enough actually using these concepts in my code. When I finally tried programming I would be derailed by debugging errors in syntax. This prevented me from exploring libraries and APIs more.

- ***Comment on your successes and frustrations with Processing and P5.js.***

My biggest frustration with both Processing and P5.js was getting the index to work correctly. I didn't understand how file paths worked until I went to a session with TA Lilly and found out that the exact location and file needs to be correct for every new file that is called to and that the main sketch is the last file that should be run. Another success was using both the SceneManager Library and P5 Play library, specifically being able to turn images into sprites.

- ***Compare and contrast OOP versus Procedural Programming. How are they similar? How are they different?*** I am looking for you to explain this in your own words, what these concepts are. I am NOT looking for you to tell me how you used these in your project.

Procedural programming uses functions while OOP uses objects, which are easier to add new data and functions to using attributes and methods. Procedural is a top down approach meaning that it provides step-by-step instructions. Rather than use multiple functions and sets of data, the use of procedural programming is more centered in passing data. OOP allows for more flexibility and customization as methods of classes allow for multiple functions belonging to a specific object rather than just existing on their own. OOP also makes it easy to

modify and maintain code as new objects can be made with the characteristics of existing objects.

- ***Specifically considering your final project: What programming concepts solidified in your final project? What did you learn with reference to programming? Did you have a break through?***

Using OOP, brushes, images, preload, event listeners, and libraries (like SceneManager, p5 Play, p5 Sound Library) solidified in my final project. I learned how to turn images into sprites and manipulate those sprites, how to reformat inspirational code into classes that could make my main sketch more cohesive, how to manipulate the new parameters in the P5 Play library, and how to import sound. My breakthrough was teaching myself how to use sprites because they have many more functionalities programmed into them from the p5 play library that are harder to achieve with regular OOP objects.

- ***Specifically considering your final project: Were you able to resolve your own bugs? What tricks did you learn in the process to help? Did you do any debugging?***

The first debugging I had to do was with my SceneManager library. I was able to resolve my own bug by deleting an unused scene, which was confusing the computer. I mostly used the Console to find errors and debug as well as consulted friends who could perhaps see errors that my familiar eyes couldn't.

- **What was your intended milestone? Did you make it? Did another one pop up? Tell me about this. Tell me how you resolved it.**

My intended milestone was to use a library and sound, both of which I was able to accomplish. I used both the SceneManager Library and P5 Play Library. Another milestone popped up in my Final Critique, when peers suggested that I add music since my theme is so music-based. I learned how to import and call to sound using the sound library. At first the audio was distorted, but I realized that I had put it into the seas loop which made it play every time the code ran, causing the distortion. I debugged by switching its location to the setup of the scene.

- **What are you *most* proud of, with reference to your final project?**

I am most proud of using the libraries because they come with a new set of syntax that I had to learn by studying examples. For example, I learned that

sprites in the p5 play libraries have different properties already coded in the library that I can manipulate with dot syntax. I am also proud of how well organized my files are because it makes it easier to find bugs.

- ***How do you think you'll move forward with programming? will you keep doing it? How does this relate to other classes you are either taking or wish to take?***

I want to use this kind of creative coding in the future to make more interactive websites and graphics for websites. Before school started, my mom asked me to make her a digital graphic for her personal business's website, but I had no clue what to use. I now can try making a graphic for her site using P5.js or Processing. I don't think I will actively pursue coding, though I'm glad I am familiar enough with it to make designs if I need to because it is a useful skill to have in my back pocket. I hope to level up the designs of my websites in my spring course, Web Dev, using code. I definitely wanted to code for my Visual Foundation Studio class when we were given the project of designing meaningful interactions in 12 screens. I used Figma for that project and found their few interactions were (tap, drag, etc) limiting compared the mouse event options in P5. Though I haven't found code the easiest, I appreciate how it allows me to express my digital ideas and others to experience those ideas in ways that sketching or speaking can't capture.