Reflection Diary

Note: This reflection has been completed after completion of the project and is a summary the main issues I encountered and how they were resolved rather than a detailed diary

Design

I initially looked into creating a multiple choice quiz with questions, Answers and answer key stored in an external csv file to be loaded but after research and trying to get the csv loaded successfully I abandoned this idea as requiring too much research and work in the time I had available. As a result of this I went to one of my earlier app ideas which was an electronics calculator function.

Implementation

Reading from an External File
I wanted to retain the importing of an external file so carried out research on the os library and started with just calling the insructions.txt file, However this kept on producing an error. I went searching the internet and at the following address (https://www.reddit.com/r/learnpython/comments/12cs01e/filenotfounderror_errno_2_no_such_file_or/?rdt=38315) which suggested a method of capturing the current working directory and how to append that to the file name so that it will always call the absolute reference to the working folder. This resolved that issue. I then wanted to read the text from the file and located the solution at (https://www.freecodecamp.org/news/how-to-read-files-in-python/) which utilised an iterative loom to go through the text file line by line and print it out.

Implementing a Menu
I wanted to implement a menu choice for both calling the functions and choosing which of ohms laws that could be selected and was struggling to find a way to do this. I initially used "while choice != (1,2,3)" which appeared to work but upon further testing I realised was working by accident and did not have true error correction so back to the internet and this link(https://stackoverflow.com/questions/46620572/user-input-choice-using-while-loops-and-if-loops-in-python) provided the solution that i needed and has worked. In Essence the while loop locks up the program unless one of the correct choices is selected. I realise this is not the most elegant of solutions but "If it's stupid and it works, It's not stupid" seemed to apply here.

Functions
Most of the coding in the 3 main functions was relatively straight forward as the structure of the equations being used are relatively easy to implement with the exception of the Resistors in parallel. The basic equation is as follows
$\frac{1}{R_{total}} = \frac{1}{R_1} + \frac{1}{R_2} + \frac{1}{R_3} \dots$
I initially tried to do this all in one step but could not figure out how to implement this and eventually realised that It can be split into 2 steps.

1. Each time I get a resistor value divide 1 by this value then add to R total.
2. Once I have collected all the values in Rtotal then divide 1 by this value.
   To ensure this worked I went to an online Parallel resistor calculator (https://www.allaboutcircuits.com/tools/parallel-resistance-calculator/) and ran a series of tests to confirm that the output I was getting matched the output of this which it did.

Testing

To ensure that the program was working I carried out test as I was working on the blocks taking known values that I had worked out before hand and running the program inputting these values to ensure that the answer I was getting matched the proposed values which they did so I was pleased about this.

Final thoughts

This has been a very interesting exercise and going through the process has been extremely beneficial as it has allowed me to build on what I had done previously and shown me some options to improve my teaching of python in class. Now that I have this start made I want to continue with this project in my own time to firstly expand it to include more calculators as well as looking to implement a GUI and to develop this further which seems less daunting after this module.