Reflection Questions

1. Why is file storage important when you're using Python? What would happen if you didn't store local files?

File storage is essential in Python for persisting data beyond program execution. It enables testing, debugging, and integration with external systems. Without file storage, you'd lose data once the program terminates, limiting testing and debugging capabilities. Interacting with other Python scripts and external services would also become challenging. Overall, file storage enhances the functionality, reliability, and scalability of Python applications.

2. In this Exercise you learned about the pickling process with the **pickle.dump()** method. What are pickles? In which situations would you choose to use pickles and why?

Pickles in Python are a way to save complex data like lists or dictionaries to a file. You'd use pickles when you need to save Python objects for later use, like caching data or storing program states. They're handy for keeping the structure of your data intact when saving and loading. Pickles offer a straightforward method to store and retrieve Python objects efficiently.

3. In Python, what function do you use to find out which directory you're currently in? What if you wanted to change your current working directory?

To find out the current directory in Python, you would use the os.getcwd() function from the os module. If you wanted to change your current working directory, you would use the os.chdir() function, also from the os module.

4. Imagine you're working on a Python script and are worried there may be an error in a block of code. How would you approach the situation to prevent the entire script from terminating due to an error?

I would use a try-except-else-finally block to handle errors. By wrapping the suspected problematic code within a try block, you can catch any raised exceptions in the except block, preventing the script from terminating prematurely. The else block can be used to execute code only if no exceptions occurred, while the finally block ensures that certain cleanup actions are performed, regardless of whether an exception occurred. This way, your script can continue running even if errors occur in specific sections of the code.

5. You're now more than halfway through Achievement 1! Take a moment to reflect on your learning in the course so far. How is it going? What's something you're proud of so far? Is there something you're struggling with? What do you need more practice with? Feel free to use these notes to guide your next mentor call.

So far it’s going great. I’m still really enjoying the simple syntax of python and am coming to appreciate its different approach to data manipulation in contrast to JavaScript’s process. It has definitely opened my eyes a bit to the innovation that came with writing programming languages.