**Name:** Laila Shafei

**Date:** 22nd of November 2024

**Course:** IT FDN 110 A

**Assignment 06 – Creating a program that** **uses functions, classes, and using the separation of concerns pattern.**

**Introduction**

The journey of developing the Course Registration Program has been an insightful and enriching experience; however, I did have some challenges understanding the topic and digesting the new functions. Hence, it took me longer than expected to complete the assignment and I was not able to submit the assignment on time. This project involved designing and implementing a Python-based application to manage student course registrations effectively, emphasizing structured error handling and user interaction. It provided a hands-on opportunity to apply theoretical knowledge to practical challenges, enabling a deeper understanding of file operations, error management, and modular programming.

**Project Overview**

The program was designed to allow users to register students for courses by entering their first and last names, along with the course name. In addition to, displaying the current list of enrolled students and their respective courses. Subsequently, saving the data persistently in a JSON file for future use. The application was developed with modularity in mind, incorporating a clear separation of concerns between data processing (handled by the FileProcessor class) and user interaction (managed by the IO class).

**Challenges Encountered and Solutions**

Throughout the development process, several challenges arose, each providing a valuable learning experience:

1. **Error in Control Flow**: At one point, the program encountered a SyntaxError due to improper indentation in the control flow. This error stemmed from a commented-out if statement, which disrupted the structure of subsequent elif blocks. Therefore, I carefully reviewed the indentation and logic, ensuring all control flow statements were properly aligned. This reinforced the importance of clean and readable code structure.
2. **Handling File Operations**: File handling introduced complexities, such as ensuring files were closed correctly even in the event of errors. So, I implemented try...except...finally blocks and using the with statement for file operations, I ensured robust error handling and proper resource management.
3. **Validating User Input**: Users might input invalid data, such as numbers in names or incorrect menu choices. Hence, I incorporated custom exceptions and validations to catch and respond to such cases.
4. **Debugging JSON Operations**: Reading and writing JSON data required careful attention to ensure compatibility with the data structure used in the program. Errors occurred when the JSON file was empty or corrupted. Therefore, I implemented error handling for JSON operations and tested the program with various scenarios to ensure robust handling of edge cases.

**Skills Gained from the assignment**

Although, this assignment seemed a bit challenging and require so much attention to details, I developed several skills in return:

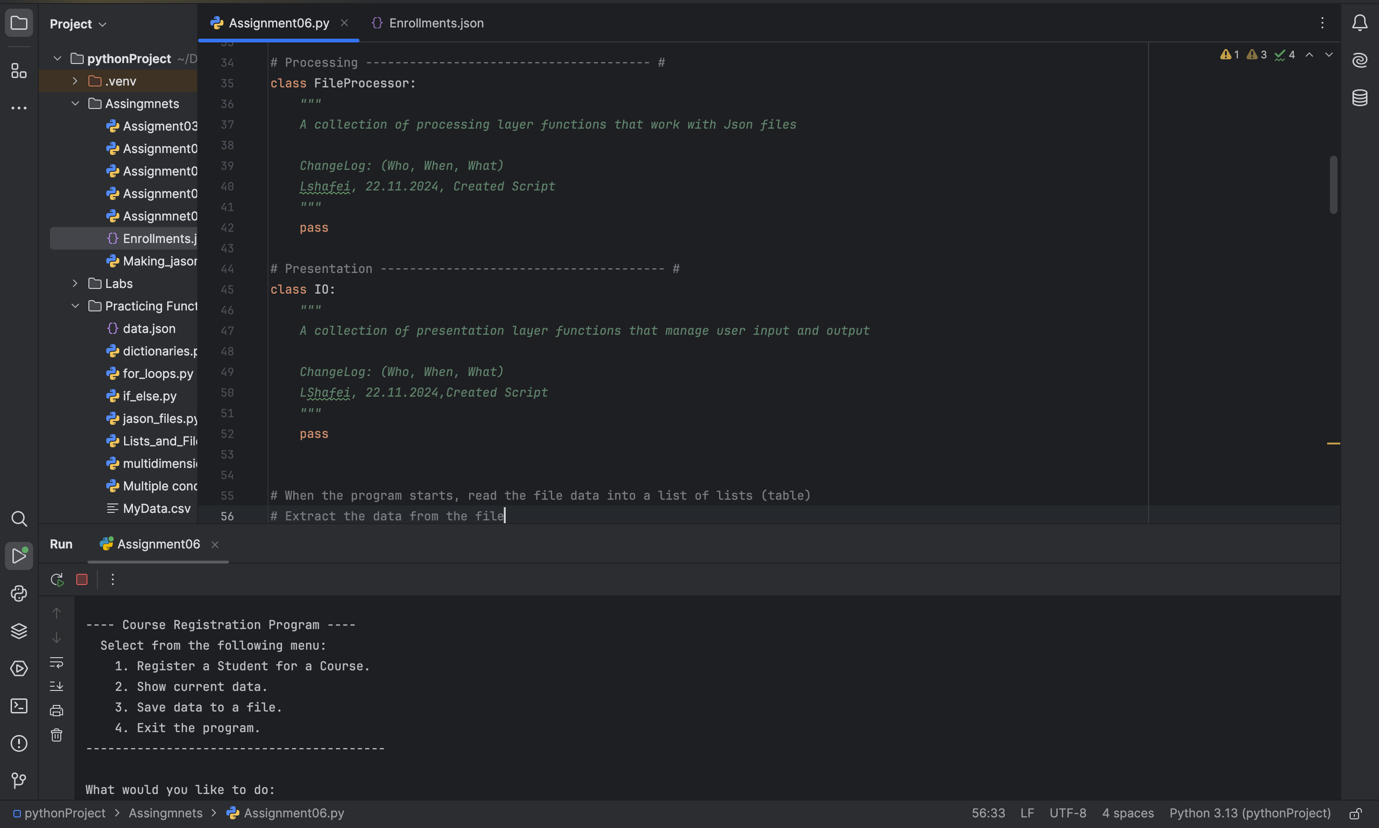
1. **Modular Programming**: I learned how to structure a program into reusable components, improving maintainability and readability.
2. **Error Handling**: Implementing structured error handling reinforced the importance of anticipating and gracefully managing potential failures.
3. **File Management**: The project provided a deeper understanding of reading and writing data to persistent storage using JSON.
4. **User Interaction Design**: Designing a user-friendly interface emphasized the significance of clear instructions and error feedback.

**Repository from GitHub:**

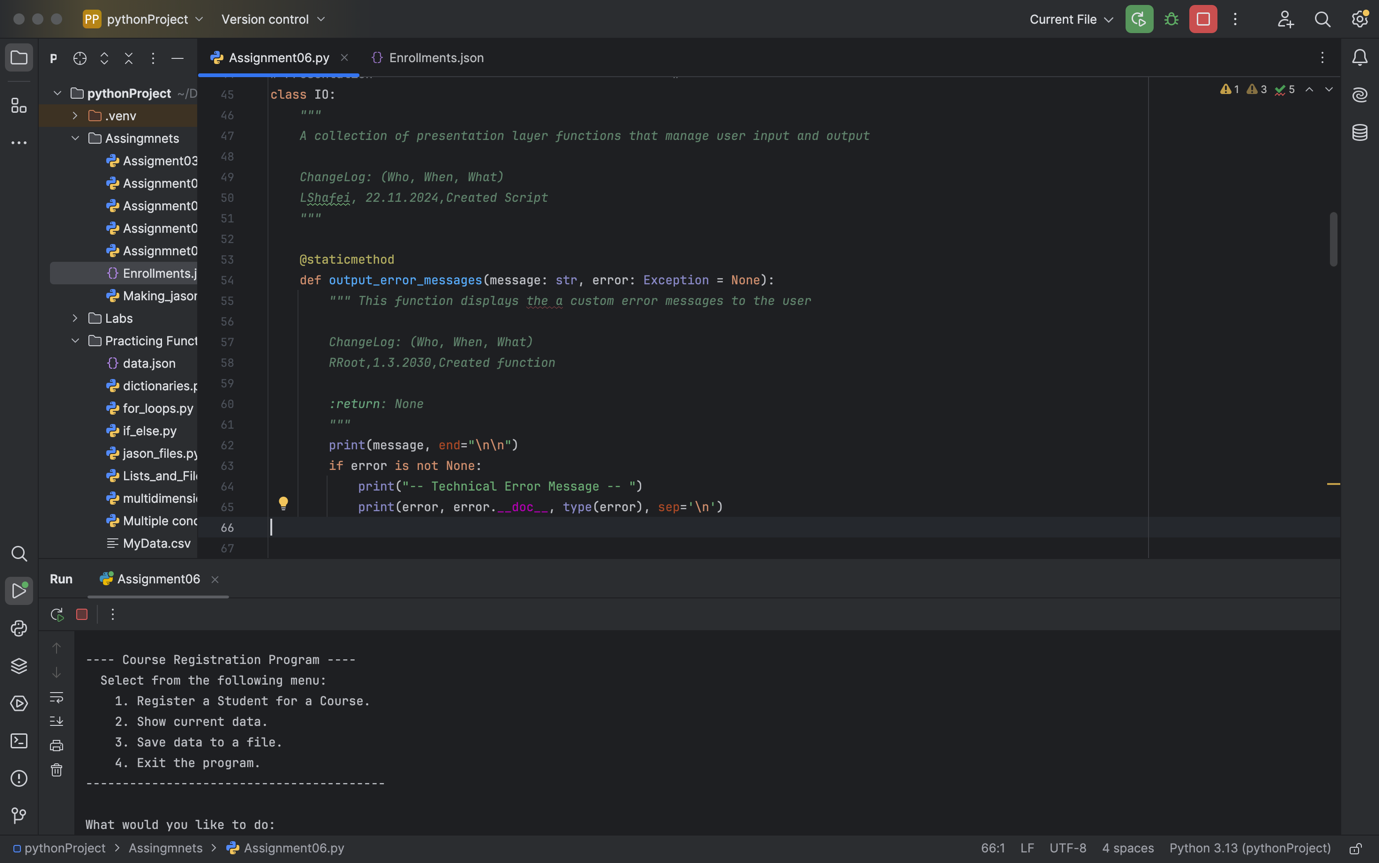
**Summary**

Finally, the development of the Course Registration Program was both challenging and rewarding. From conceptualization to debugging, the project provided a comprehensive learning experience that deepened my understanding of Python programming and practical software development. The challenges I faced, particularly in error handling, input validation, and file operations, taught me the value of careful planning, iterative testing, and robust coding practices. This project highlighted the importance of breaking problems into manageable components and addressing issues systematically. It also underscored the significance of clear and maintainable code, not only for current functionality but for future scalability.

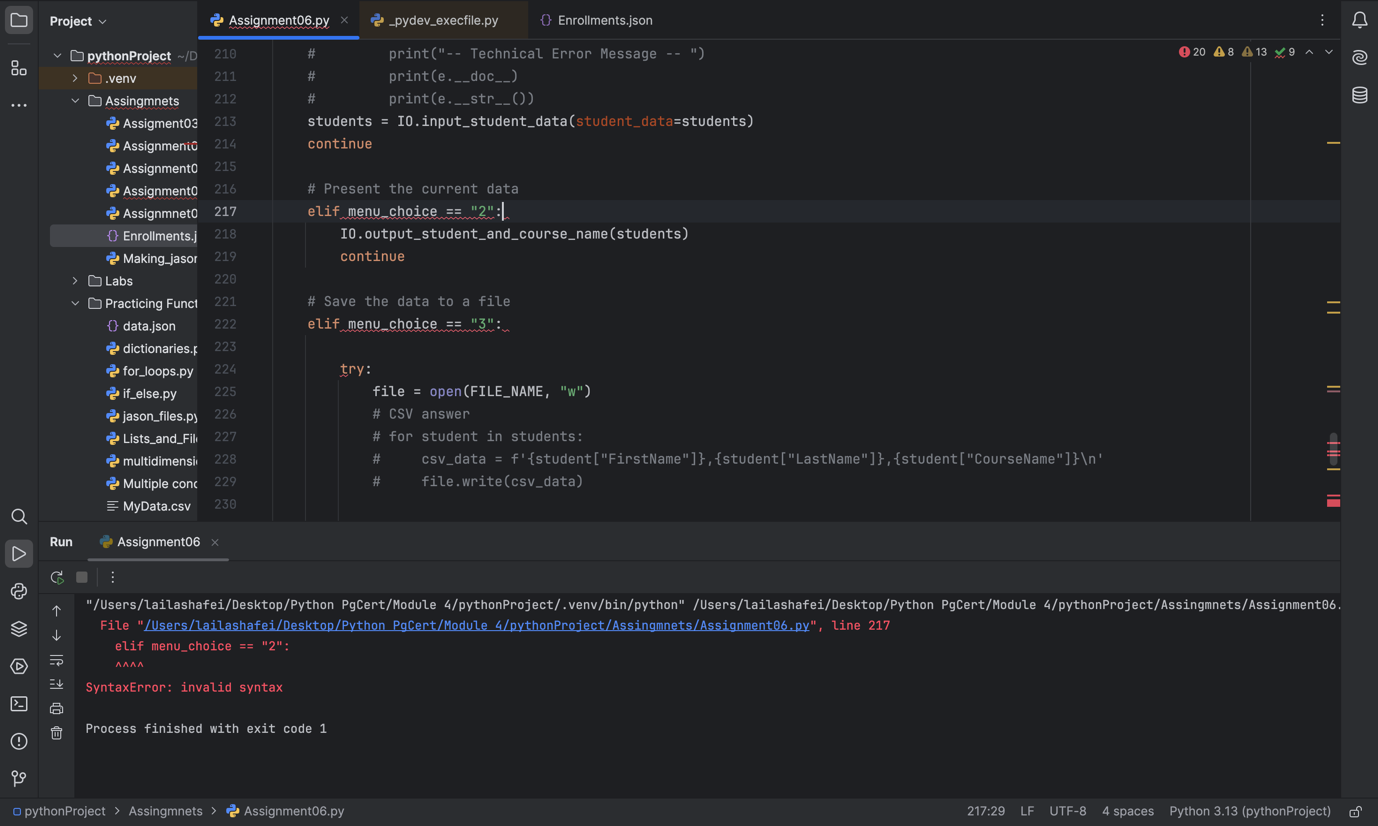
**Appendix for snips from the process of coding**

****

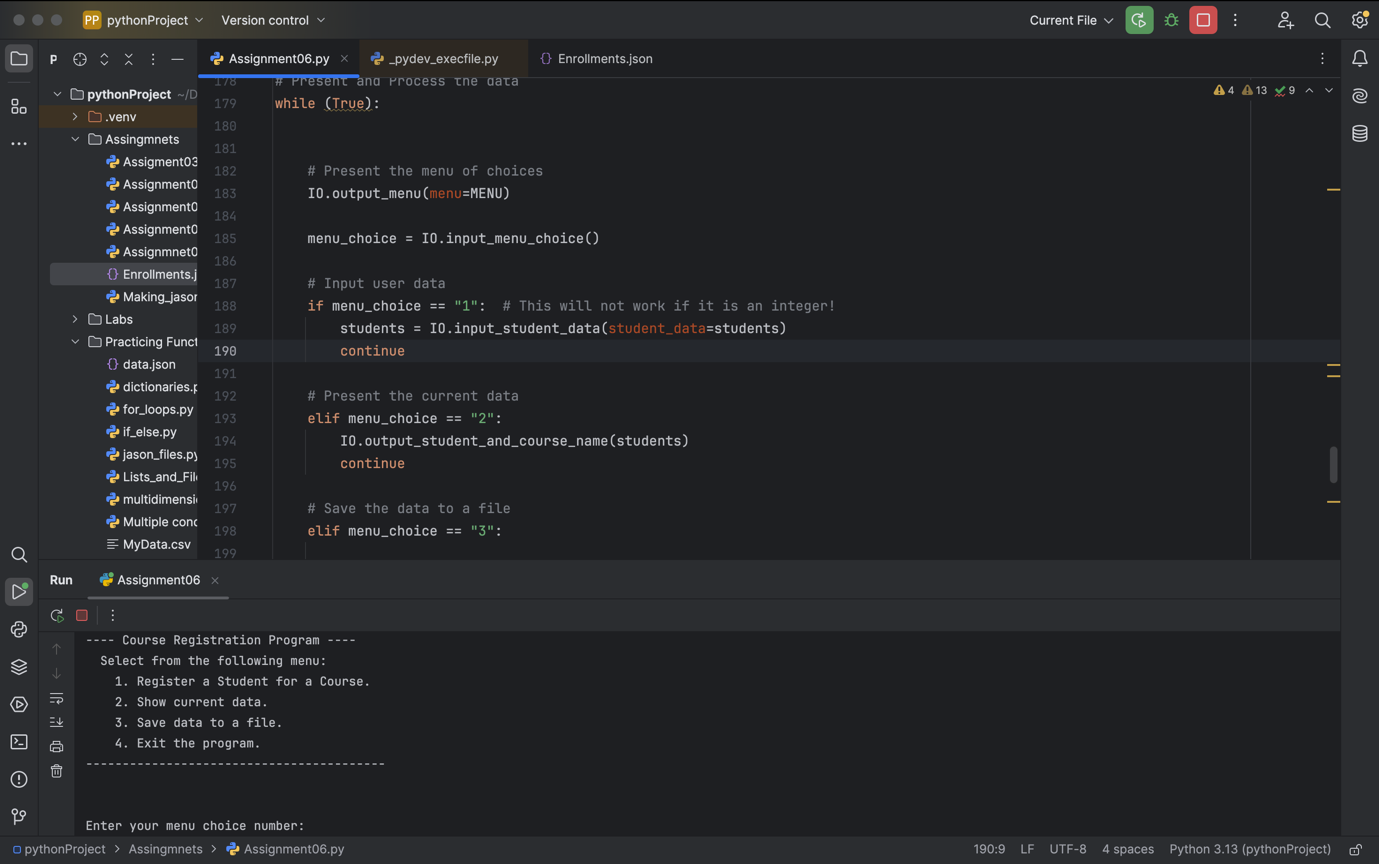
**Figure 1**

****

**Figure 2**

****

**Figure 3**

****

**Figure 4**