Name: Diellza Mehmeti
Date: 11/20/2024
Course: Foundations of Programming: Python
Assignment: 06
https://github.com/diellzak1/IntroToProg-Python-Mod06

Classes and Functions

Introduction

In this week, I was introduced to classes and function in Python. This was a challenging module since it was the first time that I learned about classes and functions. At the same time, it was interesting to go through all the script and rearrange by creating classes and functions. I continued working on my code from assignment 05 where I organized the code by classes and functions. I ran a few testing in Python and CMD to see if everything works as intended. Finally, I created a repository in GitHub where I saved the script and the documentation.

Script Header, Comments, Constants, and Variables

I started the assignment by utilizing the setup that Professor Randal had saved. I updated the name, title, and logs in the script header. Comments were set and I added a few to document my code. I then declared the two constants given in the assignment and two variables. The rest of the variables that were used in the previous assignment were deleted since now we are using functions.

Classes and Functions

I created two classes, FileProcessor and IO. Each of these classes will group different functions. The names of the classes that are given indicate what type of functions each code will have. The FileProcessor class holds the two functions that read the data from json file, and the other functions will write to the json file. The IO class has five functions that relate to the data input and output. started the assignment by using syntax to read the data from the json file. I also included a few errors exception handlings if the file does not exist when the program starts to read the file. I tested the errors in this part of the code, and they worked as intended.

Here is the example of the FileProcessor class in PyCharm:

```
class FileProcessor:
    """A collection of processing layer functions that work with Json files

        ChangeLog: (Who, When, What)
        DMehmeti, 11.20.2024,Created Class
        """
    @staticmethod
    def read_data_from_file(file_name: str, student_data: list):
        """This function reads the file data into a list of dictionaries

           ChangeLog: (Who, When, What)
           DMehmeti, 11.20.2024,created a function
           :return: list
```

```
        """
        try:
            file = open(file_name, "r")
            student_data = json.load(file)
            file.close()
        except Exception as e:
            IO.output_error_messages(message="There was a problem reading the
file", error=e)
        finally:
            if file.closed == False:
                file.close()
        return student_data

    @staticmethod
    def write_data_to_file(file_name: str, student_data: list):
        """This function writes the file data into a json file

             ChangeLog: (Who, When, What)
             DMehmeti, 11.20.2024,created a function
             :return: list
              """
        try:
            file = open(file_name, "w")
            json.dump(student_data, file)
            file.close()
            IO.output_student_courses(student_data=student_data)
        except Exception as e:
            message="Error: There was a problem with writing to the file.\n"
            message+= "Please check that the file is not open by another
program."
            IO.output_error_messages(message=message, error=e)
        finally:
            if not file.closed:
                file.close()
```

Testing and running the script

I tested the code after each change that I made by using the breakpoints. As the code is getting more complex, I found it helpful to do continues testing. I ran in some errors when trying to read the data from the file. I was able to resolve this by reading from the file when the main body of the script starts. I also tested scenarios in the code if I chose a menu choice outside 1-4 and ensured that it works properly. Once I finalized the testing portion, I ran the code in PyCharm, and then in CMD. I saved the file with the “py” extension, uploaded it in the GitHub and shared it in the discussion board for my peers to review.

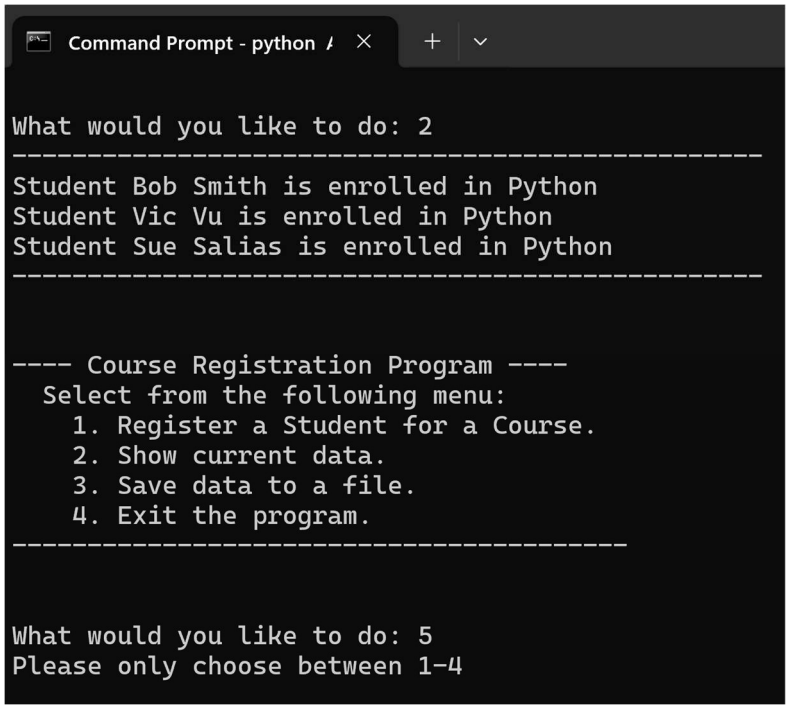

Figure 1: Running Python code in CMD and displaying menu choice 2 and 5

Summary

In this week's assignment, I continued to work from assignment five. This week I learned about how to better organize my code through the use of classes and functions. This was a challenging module but also very interesting to edit the code from assignment five and handle various errors through testing. I used two classes to group a few functions. In the first class, FileProcessor, I used two functions to read and write json files. In the second class, IO, I used five functions to handle user input and output. I continued to use error handling codes (try-except) when the file is read, when the user enters their first name and last name, and when the file is written. Once the code was tested, I ran it in PyCharm and CMD. Finally, I saved the code in GitHub and shared it with my peers for review.