Ryan Choi

November 20, 2024

Foundations of Programming: Python

Assignment 06

https://github.com/megaryanc/IntroToProg-Python-Mod06

Functions

Introduction

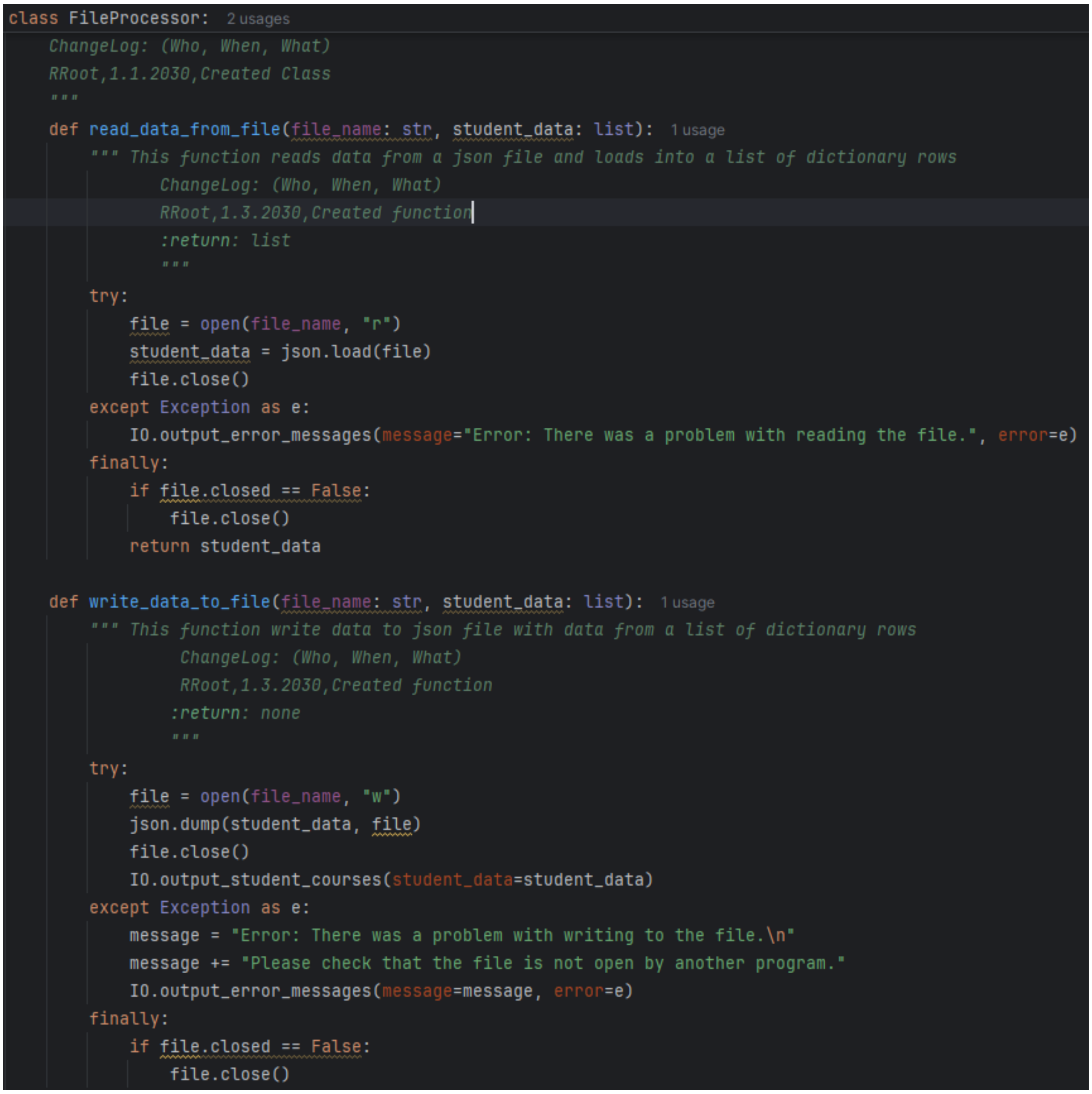
Assignment 06 focused on writing classes to simplify the script. It added a technique of creating classes to simplify the functions to be easily recallable. PyCharm IDE was used for the script with basic outline provided as a starter file.

Define Data Constants and Variables

Data constants and variables were all the same from assignment 05 apart from student_data being defined as a dictionary and using json_data.

Processing

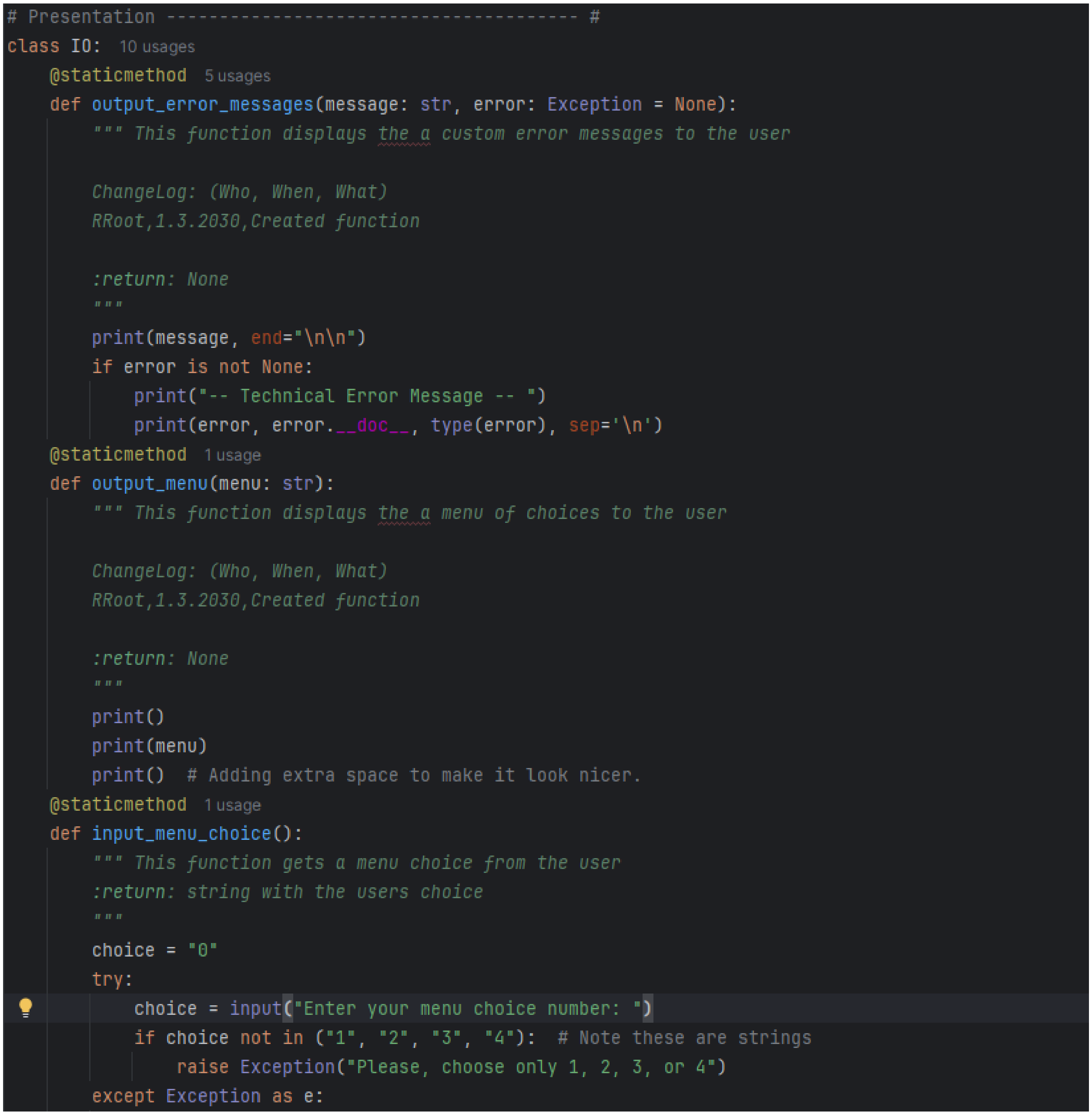
The first class created in the script was for file processing. The definitions were from lab3 and it had functions to read/write file with exceptions. The following class with functions are shown in figure 1.



```
class FileProcessor:
    ChangeLog: (Who, When, What)
    RRoot,1.1.2030,Created Class
    """
    def read_data_from_file(file_name: str, student_data: list):
        """ This function reads data from a json file and loads into a list of dictionary rows
               ChangeLog: (Who, When, What)
               RRoot,1.3.2030,Created 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="Error: There was a problem with reading the file.", error=e)
        finally:
            if file.closed == False:
                file.close()
            return student_data

    def write_data_to_file(file_name: str, student_data: list):
        """ This function write data to json file with data from a list of dictionary rows
                 ChangeLog: (Who, When, What)
                 RRoot,1.3.2030,Created function
                :return: none
                """
        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 file.closed == False:
                file.close()
```

Figure 1.class for FileProcessor.

The second class was created for the menu choices that previously had a function to each one. Figure 2 shows the various definitions created for each menu choice. The following definitions were recalled using their functions. The script was referenced from lab 03.

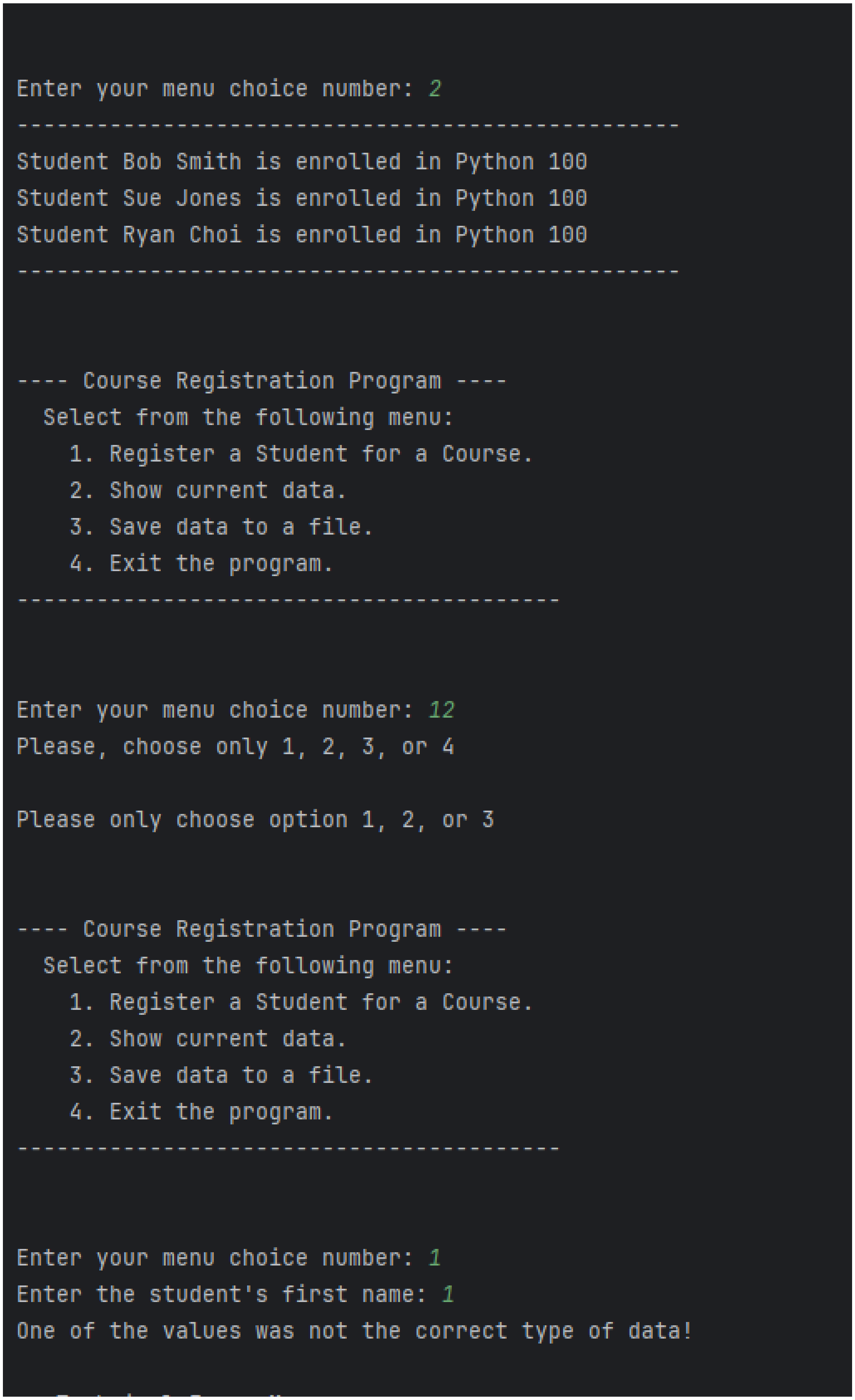


```
# Presentation --------------------------------------- #
class IO:  10 usages
    @staticmethod  5 usages
    def output_error_messages(message: str, error: Exception = None):
        """ This function displays the a custom error messages to the user

        ChangeLog: (Who, When, What)
        RRoot,1.3.2030,Created function

        :return: None
        """
        print(message, end="\n\n")
        if error is not None:
            print("-- Technical Error Message -- ")
            print(error, error.__doc__, type(error), sep='\n')
    @staticmethod  1 usage
    def output_menu(menu: str):
        """ This function displays the a menu of choices to the user

        ChangeLog: (Who, When, What)
        RRoot,1.3.2030,Created function

        :return: None
        """
        print()
        print(menu)
        print()  # Adding extra space to make it look nicer.
    @staticmethod  1 usage
    def input_menu_choice():
        """ This function gets a menu choice from the user
        :return: string with the users choice
        """
        choice = "0"
        try:
            choice = input("Enter your menu choice number: ")
            if choice not in ("1", "2", "3", "4"):  # Note these are strings
                raise Exception("Please, choose only 1, 2, 3, or 4")
        except Exception as e:
```

Figure 2. examples of definitions created.

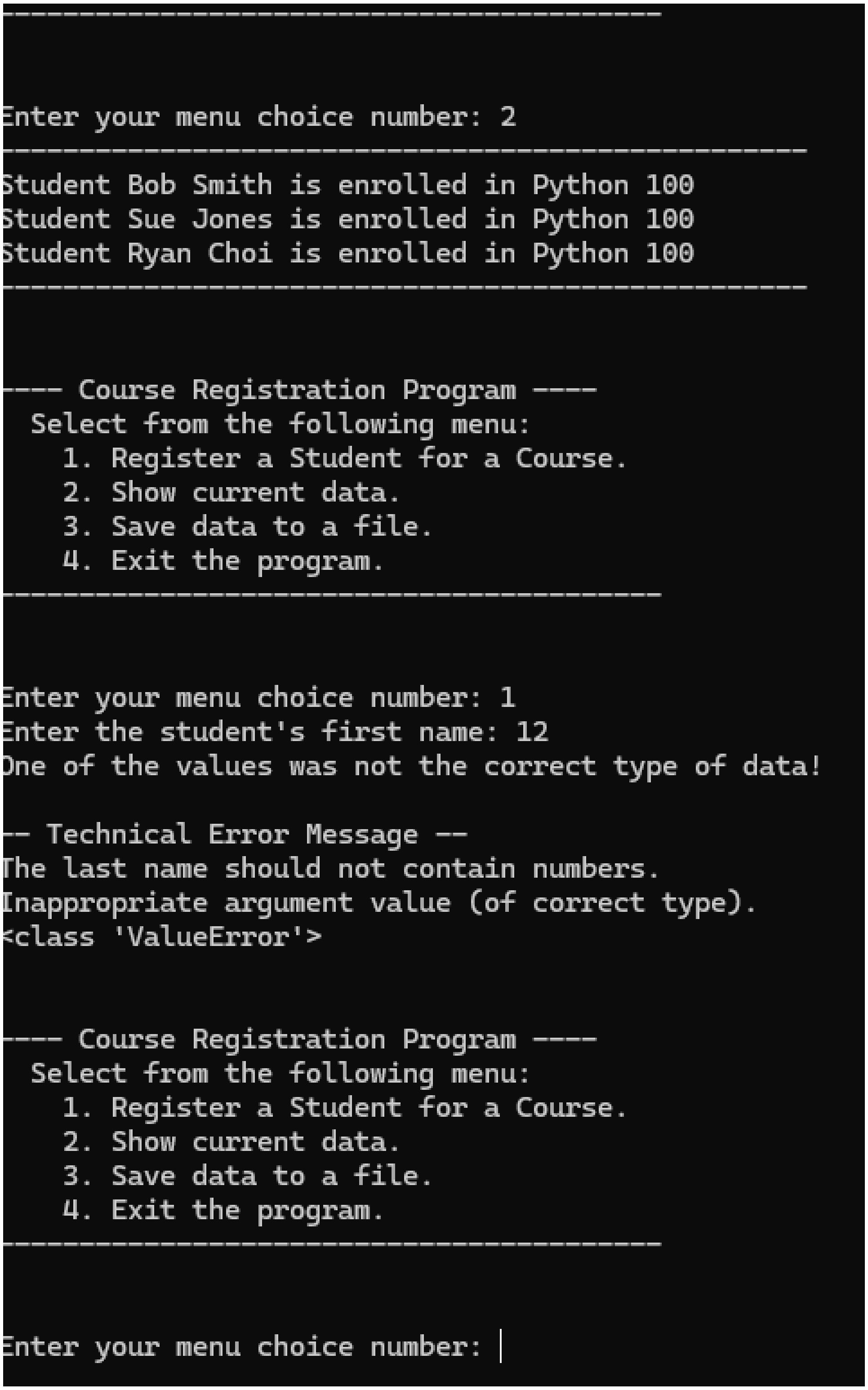
Testing

The program was tested in PyCharm and CMD with the results shown in figure 3 and 4. The proper error handling was examined with if multiple registrations were taking place. The file output was to json with the result show in in figure 5.



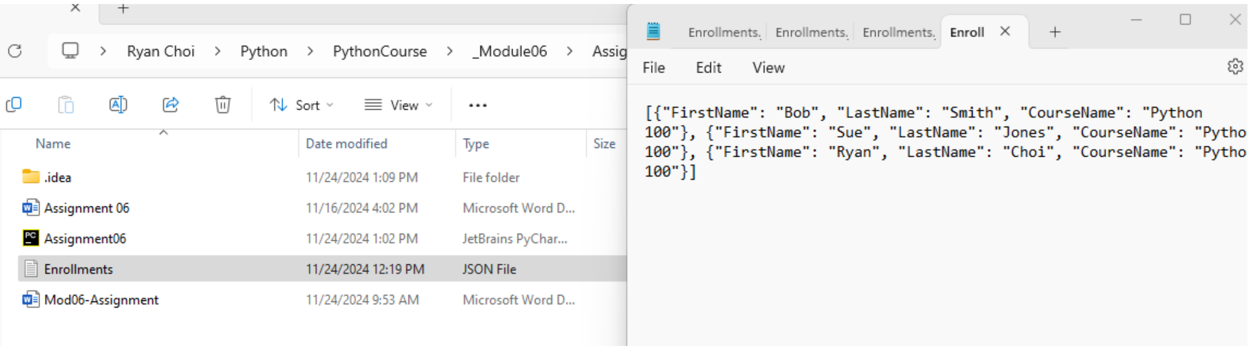
```
Enter your menu choice number: 2
--------------------------------------------------
Student Bob Smith is enrolled in Python 100
Student Sue Jones is enrolled in Python 100
Student Ryan Choi is enrolled in Python 100
--------------------------------------------------



---- Course Registration Program ----
  Select from the following menu:
    1. Register a Student for a Course.
    2. Show current data.
    3. Save data to a file.
    4. Exit the program.
-----------------------------------------



Enter your menu choice number: 12
Please, choose only 1, 2, 3, or 4

Please only choose option 1, 2, or 3



---- Course Registration Program ----
  Select from the following menu:
    1. Register a Student for a Course.
    2. Show current data.
    3. Save data to a file.
    4. Exit the program.
-----------------------------------------



Enter your menu choice number: 1
Enter the student's first name: 1
One of the values was not the correct type of data!
```

Figure 3. Demonstrates the requirements being met on PyCharm with error handling and multiple registrations.



```
-----------------------------------------



Enter your menu choice number: 2
--------------------------------------------------
Student Bob Smith is enrolled in Python 100
Student Sue Jones is enrolled in Python 100
Student Ryan Choi is enrolled in Python 100
--------------------------------------------------



---- Course Registration Program ----
  Select from the following menu:
    1. Register a Student for a Course.
    2. Show current data.
    3. Save data to a file.
    4. Exit the program.
-----------------------------------------



Enter your menu choice number: 1
Enter the student's first name: 12
One of the values was not the correct type of data!

-- Technical Error Message --
The last name should not contain numbers.
Inappropriate argument value (of correct type).
<class 'ValueError'>


---- Course Registration Program ----
  Select from the following menu:
    1. Register a Student for a Course.
    2. Show current data.
    3. Save data to a file.
    4. Exit the program.
-----------------------------------------



Enter your menu choice number: 
```

Figure 4. Demonstrates the requirements being met on CMD.



Name	Date modified	Type	Size
.idea	11/24/2024 1:09 PM	File folder	
Assignment 06	11/16/2024 4:02 PM	Microsoft Word D...	
Assignment06	11/24/2024 1:02 PM	JetBrains PyChar...	
Enrollments	11/24/2024 12:19 PM	JSON File	
Mod06-Assignment	11/24/2024 9:53 AM	Microsoft Word D...	

```
[{"FirstName": "Bob", "LastName": "Smith", "CourseName": "Python
100"}, {"FirstName": "Sue", "LastName": "Jones", "CourseName": "Pytho
100"}, {"FirstName": "Ryan", "LastName": "Choi", "CourseName": "Pytho
100"}]
```

Figure 5. Data recorded in the csv.

Summary

The assignment was useful in learning how to write classes to organize the functions. It was simpler to understand the script and I will use this technique when writing longer scripts.