Harpreet Bassi

August 4, 2024

Foundations of Programming, Python

Assignment 06

GitHub Link- https://github.com/harpreetkbassi/IntroToProg-Python-Mod06

Functions

Intro

In assignment 06, I learned about the additional programming tools and techniques using PyCharm. In this document, I will explain the steps I took to create an interactive program similar to assignment 05, where I used my knowledge of functions, classes, and the separation of concerns programming pattern. And finally learned how to implement this data in a JavaScript Object Notation (JSON) file and GitHub.

Creating the Code

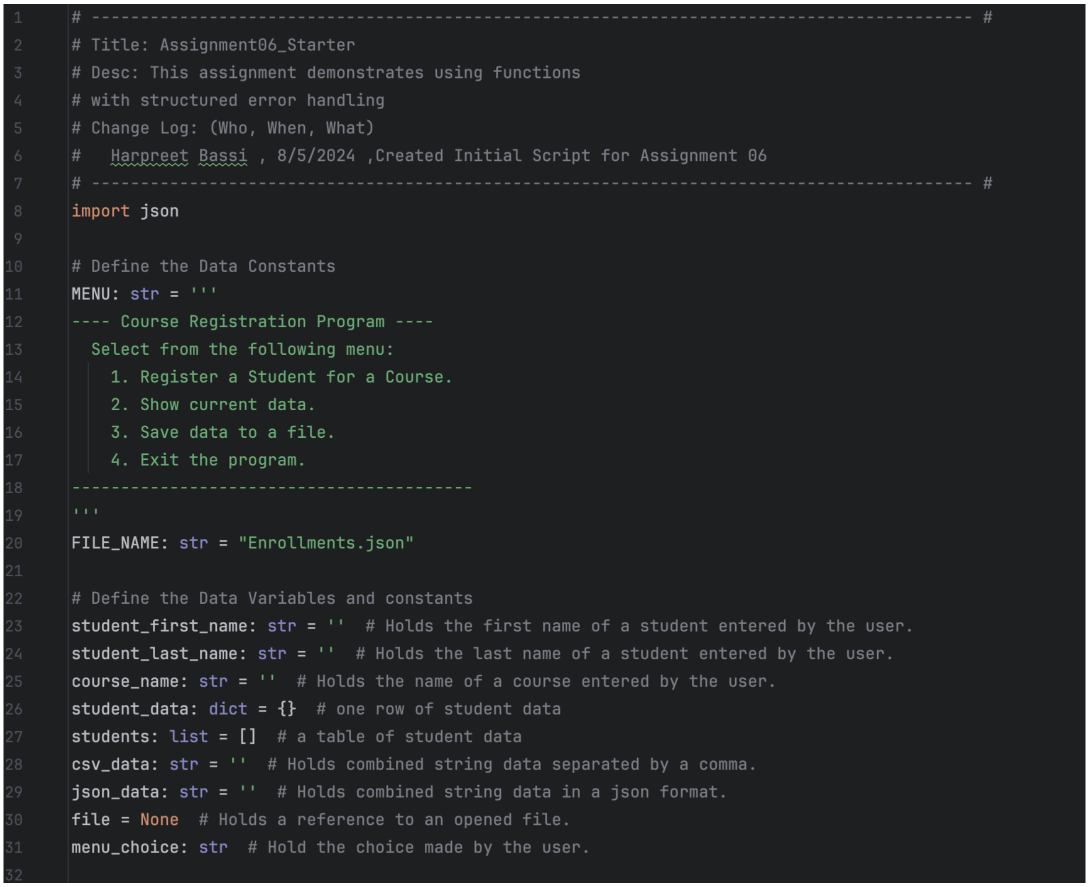
I started off by opening PyCharm IDE and Assignment06.py, edited the script header, and then began writing my code. As seen in Figure 1, first thing I started with was include the line that imports the "JSON" module, which is used for working with JSON data in Python. I the "MENU" line which is a multi-string that contains the menu options displayed to the user. I also included the "FILE_NAME" (Enrollments.json) is the name of the file where the data will be stored. I then included a set of lines that define the variables and constants used in the script. All of them include the placeholders regarding the student information, a list to keep all the student data, and variables for handling file operations and menu choices.

As seen in Figure 1, I then added a block of where I told the code to open the "Enrollments.json" file and read its contents into the "students" list using the "json.load". The purpose of the json.load() method is to take a valid JSON string and convert it into a Python Dictionary. I then wrote the code to execute that if the file does not exist, to print an error message of "Error: {FILE_NAME} not found. A new file will be created when you save data". I then added the code that if the file is not in JSON format, to print another error message of "Error: The file format is incorrect. Please confirm it is in JSON format". I also then added the code where it catches all the exception handlers within the try-except block. The purpose of it was to catch any exceptions that were not explicitly handled by the previous except clauses. And then the last thing in the block I did was include the "finally" block, and wrote to the code to make sure the file is closed if it was not closed and still open.

As shown in Figure's 2-5, I then went onto the part of the code where I would define the Functions. The purpose of using the Functions was because it is "a convenient way to divide your code into useful blocks, allowing us to order our code, make it more readable, reuse it and save some time. Also, functions are a key way to define interfaces so programmers can share their code" (Functions, learnpython.org). The first part of the code I included the "def" keyword which is defining the functions, and then included the functions name and the blocks name. In this case, I started out with the function that would print out the error messages. In the code, I told it to print out a general error message with its information if there is an error. As shown in Figure 3, I then included the code where I told it to print the menu to the user, included another block where it requests the user to enter their menu choice and display it to them, included the function which prints the current list of students in the course registration in a formatted sequence, included the function that requests the user to enter the student information and then appends it to the "student_data" list and brings a "ValueError" if any of the input fields are empty and then prints an error message if something is empty. I then added the function that reads the data from the JSON file into the "student_data" list, also added the file not found error, JSON decode error, and other exceptions to print an error message that the file is not found. As shown in Figure 4, I finally then included the function that writes the "student_data" list to the JSON file and handles any of the exceptions by printing the error messages, if there is any error that occurs.

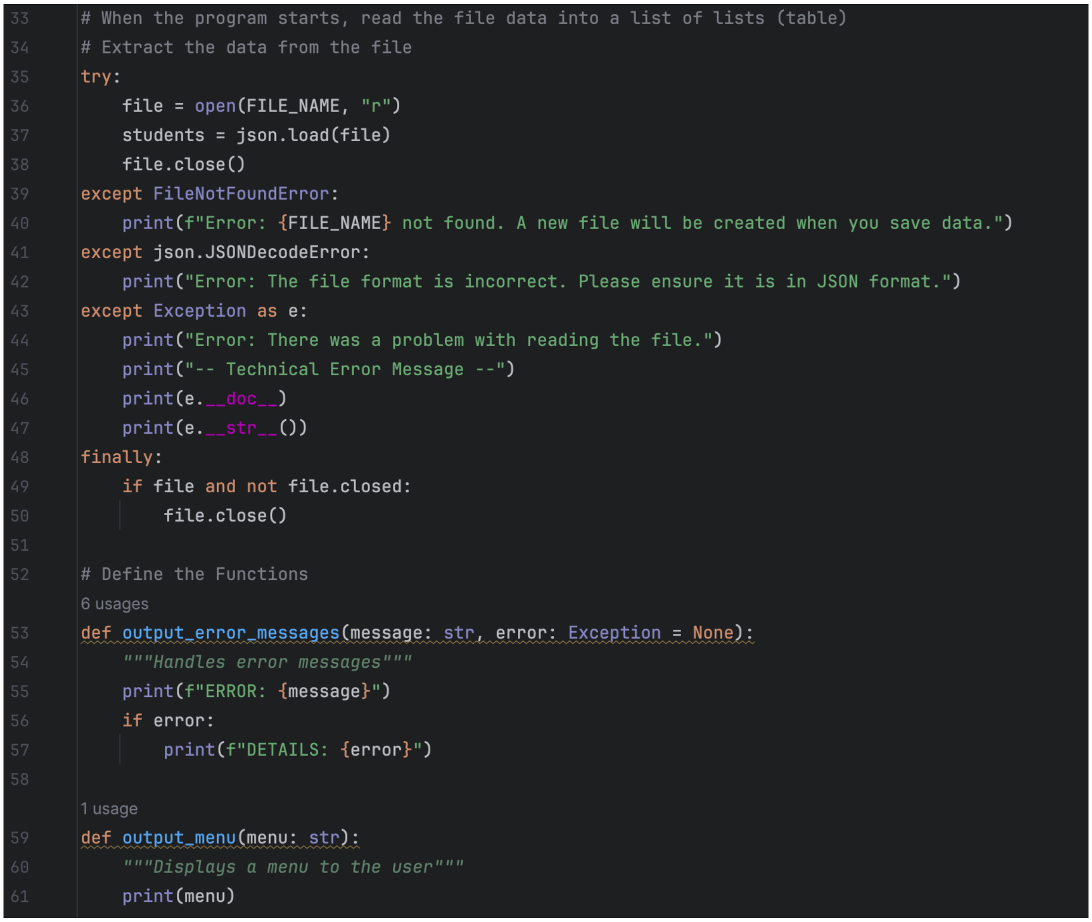
The last part I added was to present and process the data. As shown in Figure 5, I started the main loop of the program, and wrote to print the starting message using the print() function, display the menu, receive the users menu choice, and then printed the choice for the debugging. I then added the block that handles the user's menu choice. If the user chose "1", then it calls the "input_student_data" to get the new student data. If the user chose "2", then it calls the "output_student_courses" to show the current data. If the user chose "3", then it calls the "write_data_to_file" to save the data and then display the current data. If the user chose "4", then it prints the exit message ("Exiting the program. Goodbye."), breaks the loop by using "break" to end the program. Lastly, I added the code that if what the user chose is invalid, then it will print the error message of "Invalid menu choice, please try again".

Lastly, the final line of code I added was the message that prints "Program Ended" using the print() function.



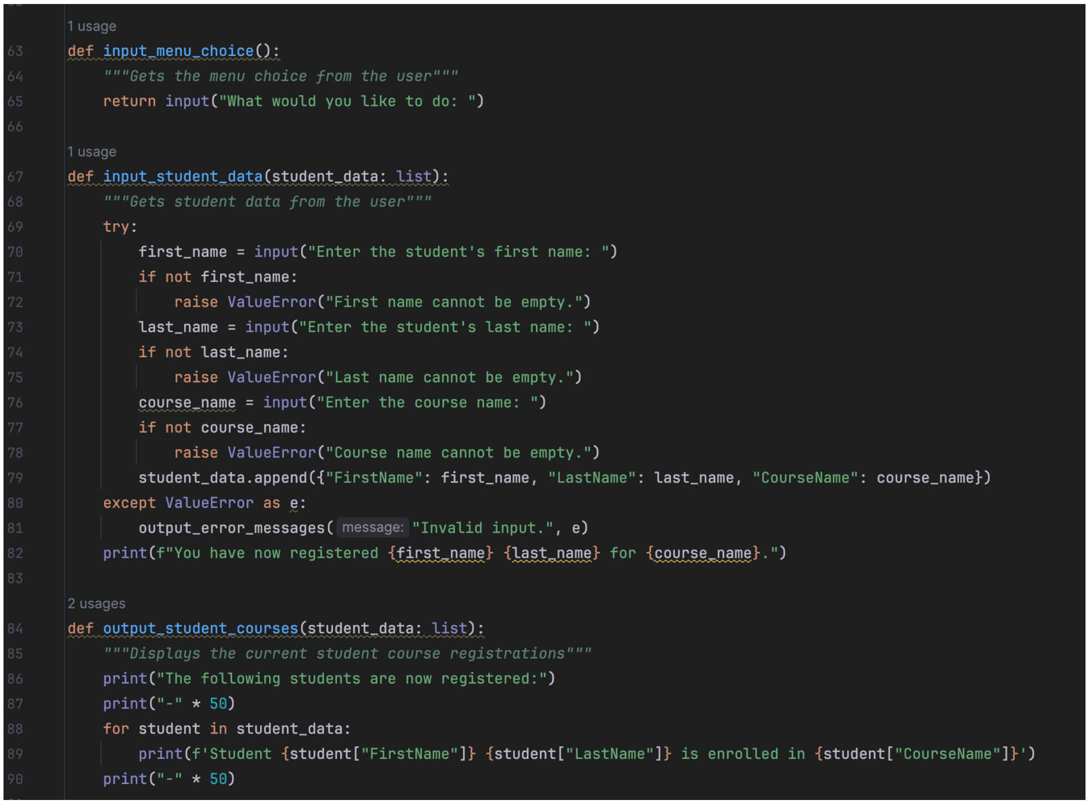
```python
# ------------------------------------------------------------------------------------------ #
# Title: Assignment06_Starter
# Desc: This assignment demonstrates using functions
# with structured error handling
# Change Log: (Who, When, What)
#   Harpreet Bassi , 8/5/2024 ,Created Initial Script for Assignment 06
# ------------------------------------------------------------------------------------------ #
import json

# Define the Data Constants
MENU: str = '''
---- 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.
-----------------------------------------
'''
FILE_NAME: str = "Enrollments.json"

# Define the Data Variables and constants
student_first_name: str = ''  # Holds the first name of a student entered by the user.
student_last_name: str = ''  # Holds the last name of a student entered by the user.
course_name: str = ''  # Holds the name of a course entered by the user.
student_data: dict = {}  # one row of student data
students: list = []  # a table of student data
csv_data: str = ''  # Holds combined string data separated by a comma.
json_data: str = ''  # Holds combined string data in a json format.
file = None  # Holds a reference to an opened file.
menu_choice: str  # Hold the choice made by the user.

```

Figure 1: Screenshot of my Final Script in the PyCharm development program



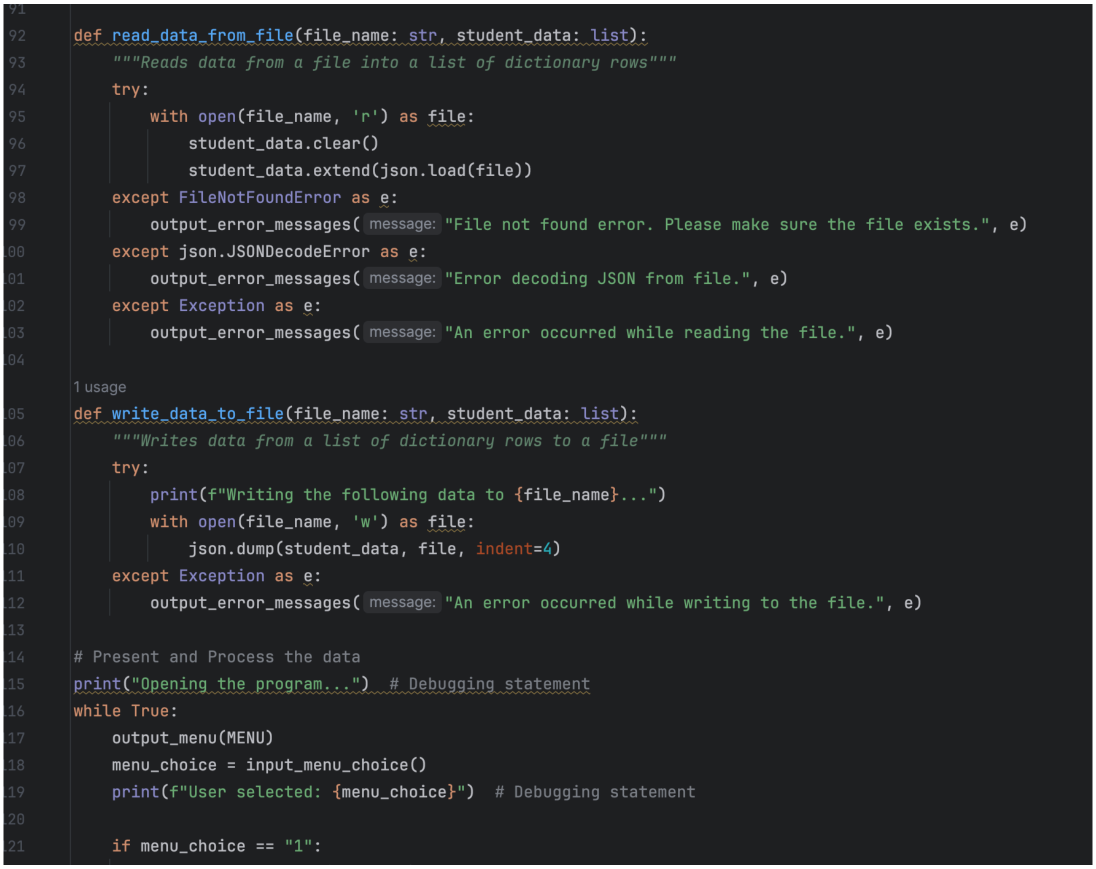
```python
# When the program starts, read the file data into a list of lists (table)
# Extract the data from the file
try:
    file = open(FILE_NAME, "r")
    students = json.load(file)
    file.close()
except FileNotFoundError:
    print(f"Error: {FILE_NAME} not found. A new file will be created when you save data.")
except json.JSONDecodeError:
    print("Error: The file format is incorrect. Please ensure it is in JSON format.")
except Exception as e:
    print("Error: There was a problem with reading the file.")
    print("-- Technical Error Message --")
    print(e.__doc__)
    print(e.__str__())
finally:
    if file and not file.closed:
        file.close()

# Define the Functions
6 usages
def output_error_messages(message: str, error: Exception = None):
    """Handles error messages"""
    print(f"ERROR: {message}")
    if error:
        print(f"DETAILS: {error}")

1 usage
def output_menu(menu: str):
    """Displays a menu to the user"""
    print(menu)
```

Figure 2: Screenshot of my Final Script in the PyCharm development program



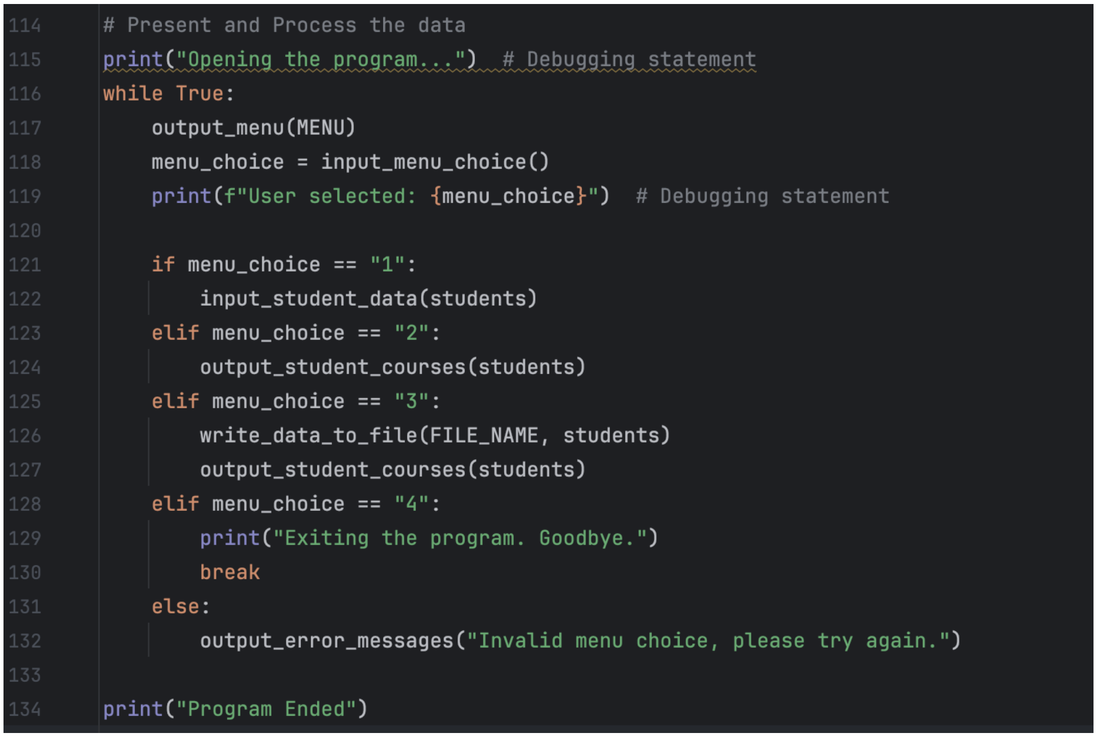
```python
def input_menu_choice():
    """Gets the menu choice from the user"""
    return input("What would you like to do: ")

def input_student_data(student_data: list):
    """Gets student data from the user"""
    try:
        first_name = input("Enter the student's first name: ")
        if not first_name:
            raise ValueError("First name cannot be empty.")
        last_name = input("Enter the student's last name: ")
        if not last_name:
            raise ValueError("Last name cannot be empty.")
        course_name = input("Enter the course name: ")
        if not course_name:
            raise ValueError("Course name cannot be empty.")
        student_data.append({"FirstName": first_name, "LastName": last_name, "CourseName": course_name})
    except ValueError as e:
        output_error_messages("Invalid input.", e)
    print(f"You have now registered {first_name} {last_name} for {course_name}.")

def output_student_courses(student_data: list):
    """Displays the current student course registrations"""
    print("The following students are now registered:")
    print("-" * 50)
    for student in student_data:
        print(f'Student {student["FirstName"]} {student["LastName"]} is enrolled in {student["CourseName"]}')
    print("-" * 50)
```

Figure 3: Screenshot of my Final Script in the PyCharm development program



```python
def read_data_from_file(file_name: str, student_data: list):
    """Reads data from a file into a list of dictionary rows"""
    try:
        with open(file_name, 'r') as file:
            student_data.clear()
            student_data.extend(json.load(file))
    except FileNotFoundError as e:
        output_error_messages("File not found error. Please make sure the file exists.", e)
    except json.JSONDecodeError as e:
        output_error_messages("Error decoding JSON from file.", e)
    except Exception as e:
        output_error_messages("An error occurred while reading the file.", e)

def write_data_to_file(file_name: str, student_data: list):
    """Writes data from a list of dictionary rows to a file"""
    try:
        print(f"Writing the following data to {file_name}...")
        with open(file_name, 'w') as file:
            json.dump(student_data, file, indent=4)
    except Exception as e:
        output_error_messages("An error occurred while writing to the file.", e)

# Present and Process the data
print("Opening the program...")  # Debugging statement
while True:
    output_menu(MENU)
    menu_choice = input_menu_choice()
    print(f"User selected: {menu_choice}")  # Debugging statement

    if menu_choice == "1":
```

Figure 4: Screenshot of my Final Script in the PyCharm development program



```python
# Present and Process the data
print("Opening the program...")  # Debugging statement
while True:
    output_menu(MENU)
    menu_choice = input_menu_choice()
    print(f"User selected: {menu_choice}")  # Debugging statement

    if menu_choice == "1":
        input_student_data(students)
    elif menu_choice == "2":
        output_student_courses(students)
    elif menu_choice == "3":
        write_data_to_file(FILE_NAME, students)
        output_student_courses(students)
    elif menu_choice == "4":
        print("Exiting the program. Goodbye.")
        break
    else:
        output_error_messages("Invalid menu choice, please try again.")

print("Program Ended")
```

Figure 5: Screenshot of my Final Script in the PyCharm development program

Troubleshooting

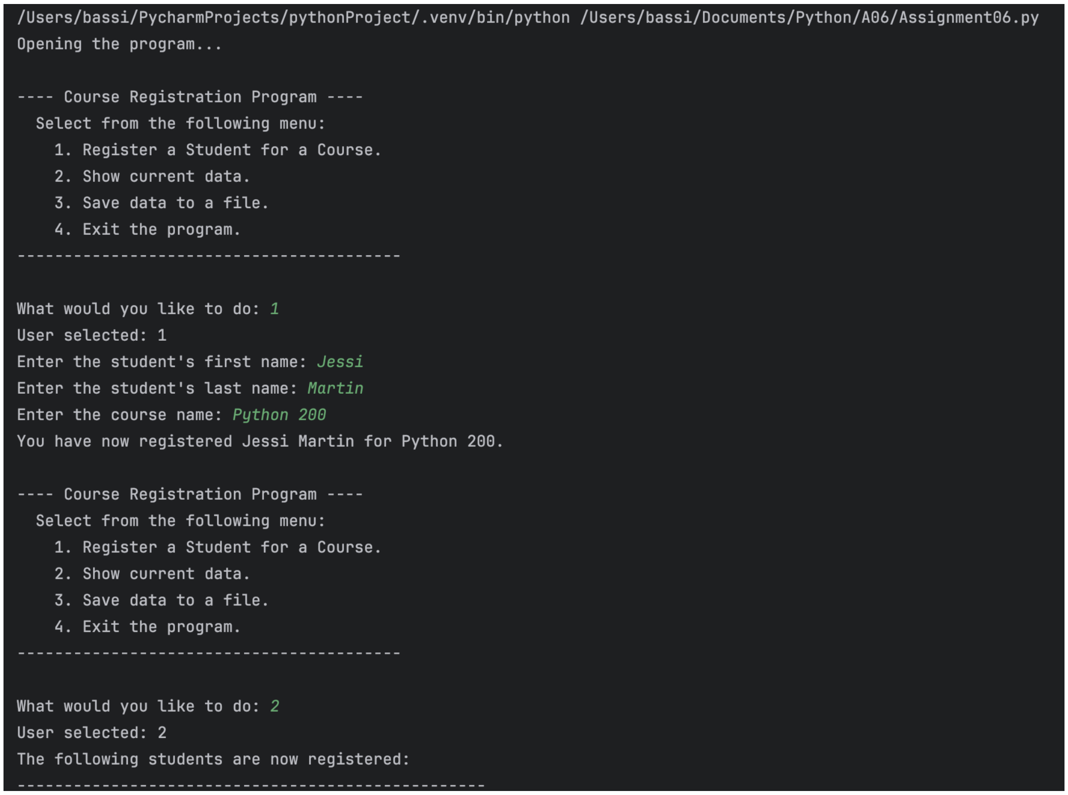
I did not run into many errors in this script; therefore, I did not have much troubleshooting to do for Assignment 06.

Save the Script

I created a folder in Documents called “Python” and saved my python script as “Assignment06.py”.

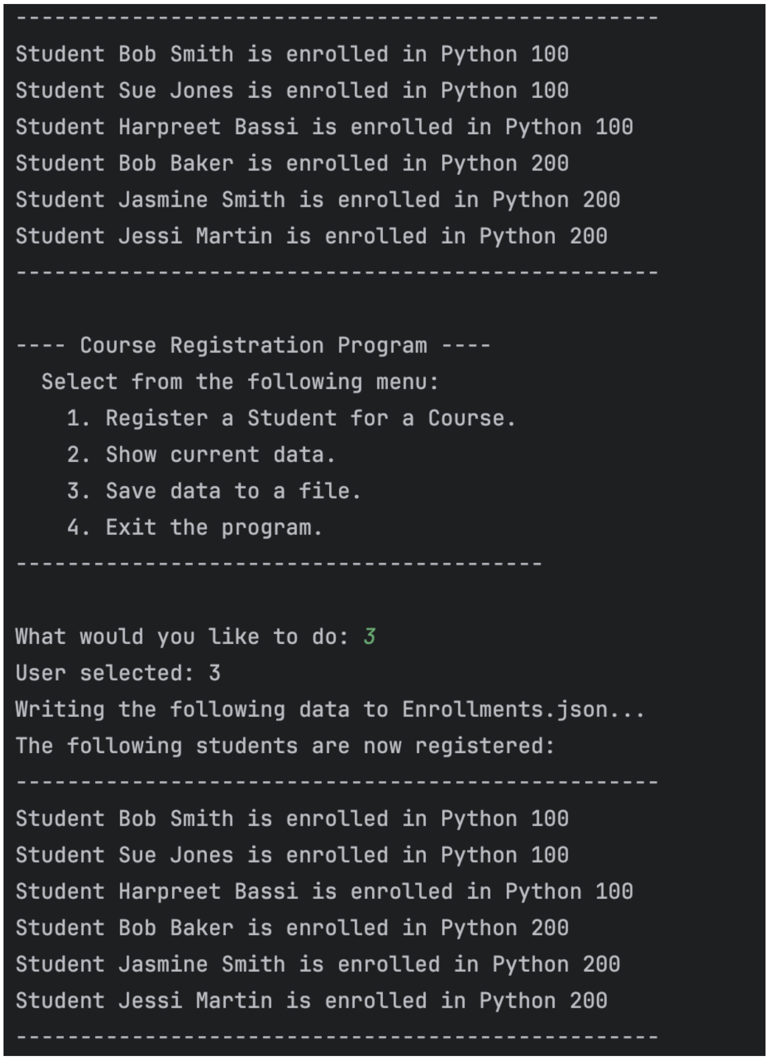
Running the Script in Terminal and PyCharm

I opened the Terminal console on my MacBook and went to the folder where the python script is located using the cd (change directory) command. I started off with using the cd command for Documents. I mainly used this for the Documents for tracking purposes for myself. I then went onto using the cd command for the Python folder in Documents, used the cd command again for the “A06” folder where the script is located, and then went onto using “python3 Assignment06.py” which then directed me to the code. Once I inputted the information the code prompted me to, I was able to successfully have the program run on both PyCharm (Figure 6-8) and Terminal (Figure 9-9.5).



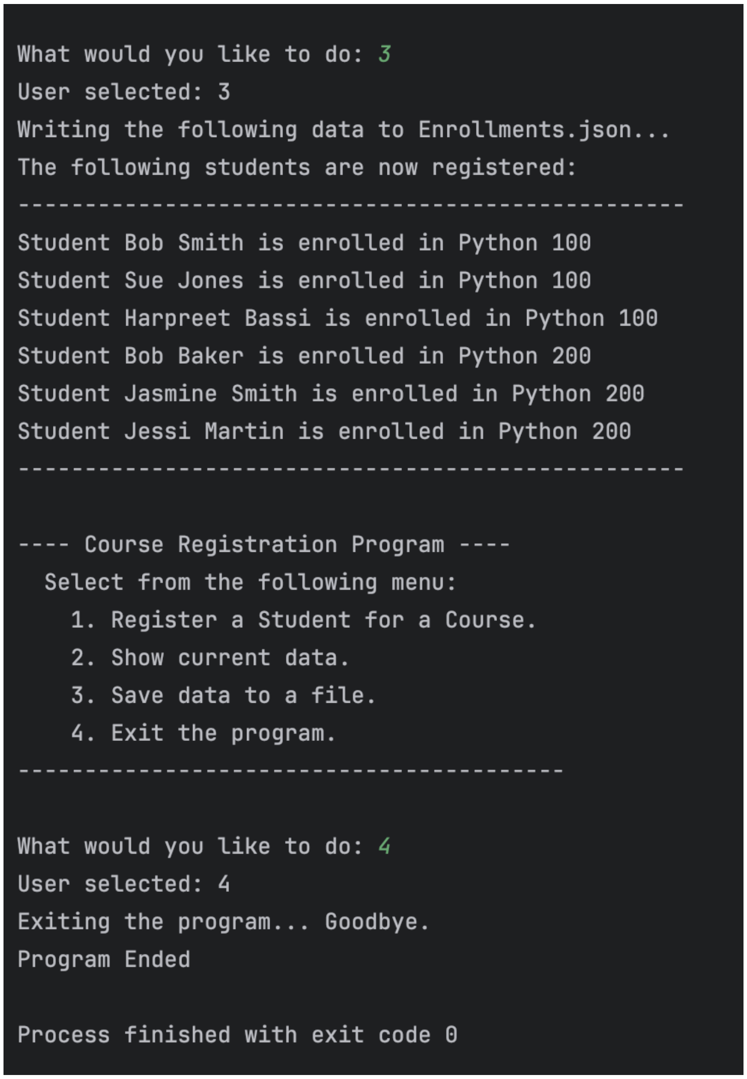
```
/Users/bassi/PycharmProjects/pythonProject/.venv/bin/python /Users/bassi/Documents/Python/A06/Assignment06.py
Opening the program...

---- 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.
-----------------------------------------

What would you like to do: 1
User selected: 1
Enter the student's first name: Jessi
Enter the student's last name: Martin
Enter the course name: Python 200
You have now registered Jessi Martin for Python 200.

---- 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.
-----------------------------------------

What would you like to do: 2
User selected: 2
The following students are now registered:
--------------------------------------------------
```

Figure 6: Screenshot of the final run for Assignment05.py in PyCharm.



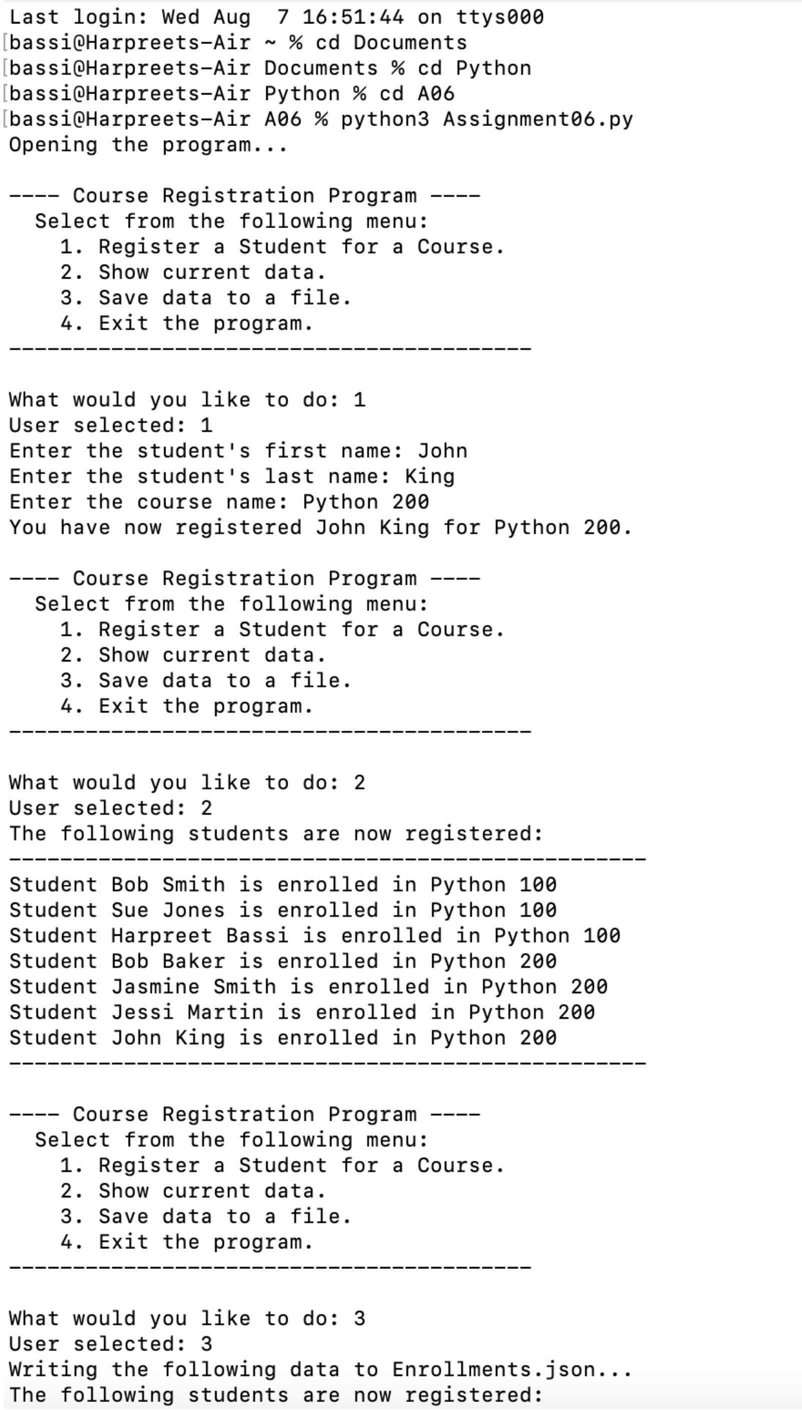
```
--------------------------------------------------
Student Bob Smith is enrolled in Python 100
Student Sue Jones is enrolled in Python 100
Student Harpreet Bassi is enrolled in Python 100
Student Bob Baker is enrolled in Python 200
Student Jasmine Smith is enrolled in Python 200
Student Jessi Martin is enrolled in Python 200
--------------------------------------------------

---- 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.
-----------------------------------------

What would you like to do: 3
User selected: 3
Writing the following data to Enrollments.json...
The following students are now registered:
--------------------------------------------------
Student Bob Smith is enrolled in Python 100
Student Sue Jones is enrolled in Python 100
Student Harpreet Bassi is enrolled in Python 100
Student Bob Baker is enrolled in Python 200
Student Jasmine Smith is enrolled in Python 200
Student Jessi Martin is enrolled in Python 200
--------------------------------------------------
```

Figure 7: Screenshot of the final run for Assignment05.py in PyCharm.



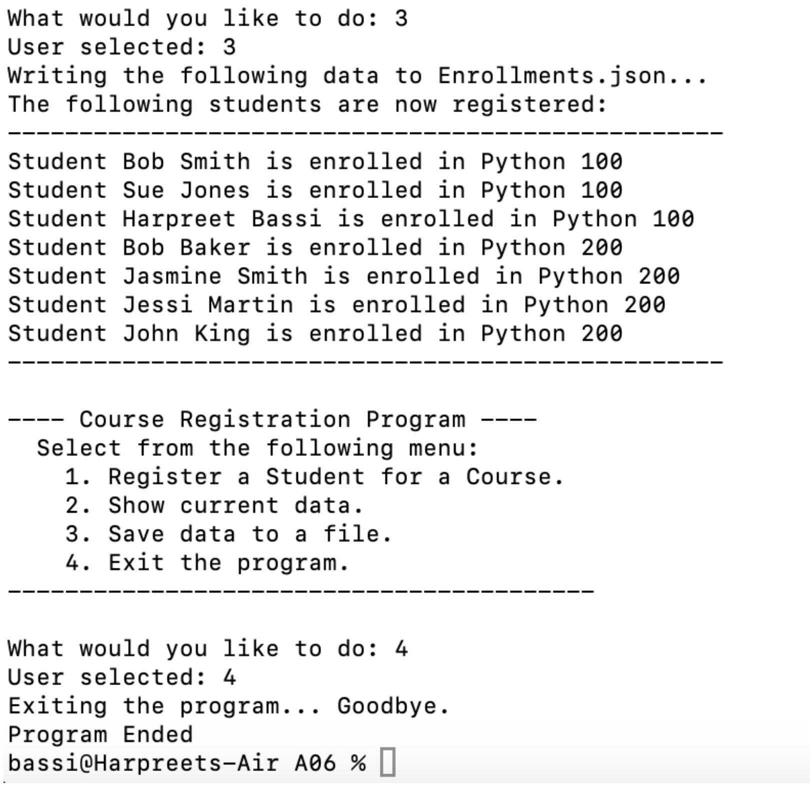
```
What would you like to do: 3
User selected: 3
Writing the following data to Enrollments.json...
The following students are now registered:
--------------------------------------------------
Student Bob Smith is enrolled in Python 100
Student Sue Jones is enrolled in Python 100
Student Harpreet Bassi is enrolled in Python 100
Student Bob Baker is enrolled in Python 200
Student Jasmine Smith is enrolled in Python 200
Student Jessi Martin is enrolled in Python 200
--------------------------------------------------

---- 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.
-----------------------------------------

What would you like to do: 4
User selected: 4
Exiting the program... Goodbye.
Program Ended

Process finished with exit code 0
```

Figure 8: Screenshot of the final run for Assignment05.py in PyCharm.



```
Last login: Wed Aug  7 16:51:44 on ttys000
[bassi@Harpreets-Air ~ % cd Documents
[bassi@Harpreets-Air Documents % cd Python
[bassi@Harpreets-Air Python % cd A06
[bassi@Harpreets-Air A06 % python3 Assignment06.py
Opening the program...

---- 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.
-----------------------------------------

What would you like to do: 1
User selected: 1
Enter the student's first name: John
Enter the student's last name: King
Enter the course name: Python 200
You have now registered John King for Python 200.

---- 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.
-----------------------------------------

What would you like to do: 2
User selected: 2
The following students are now registered:
--------------------------------------------------
Student Bob Smith is enrolled in Python 100
Student Sue Jones is enrolled in Python 100
Student Harpreet Bassi is enrolled in Python 100
Student Bob Baker is enrolled in Python 200
Student Jasmine Smith is enrolled in Python 200
Student Jessi Martin is enrolled in Python 200
Student John King is enrolled in Python 200
--------------------------------------------------

---- 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.
-----------------------------------------

What would you like to do: 3
User selected: 3
Writing the following data to Enrollments.json...
The following students are now registered:
```

Figure 9: Screenshot of the commands to locate the folder, script, and run Assignment05.py

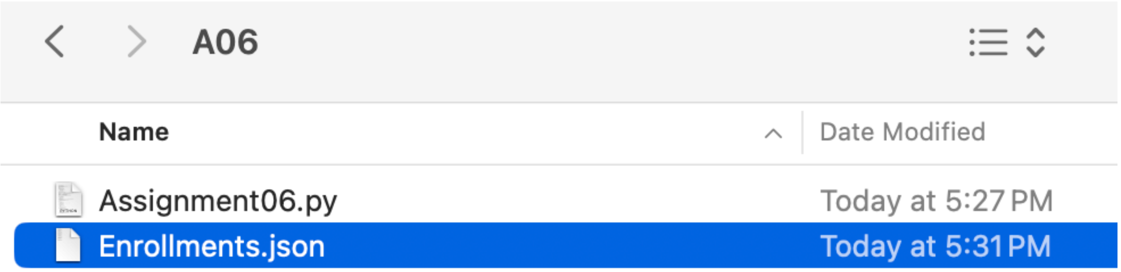


```
What would you like to do: 3
User selected: 3
Writing the following data to Enrollments.json...
The following students are now registered:
--------------------------------------------------
Student Bob Smith is enrolled in Python 100
Student Sue Jones is enrolled in Python 100
Student Harpreet Bassi is enrolled in Python 100
Student Bob Baker is enrolled in Python 200
Student Jasmine Smith is enrolled in Python 200
Student Jessi Martin is enrolled in Python 200
Student John King is enrolled in Python 200
--------------------------------------------------

---- 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.
-----------------------------------------

What would you like to do: 4
User selected: 4
Exiting the program... Goodbye.
Program Ended
bassi@Harpreets-Air A06 % 
```

Figure 9.5: Screenshot of the commands to locate the folder, script, and run Assignment05.py

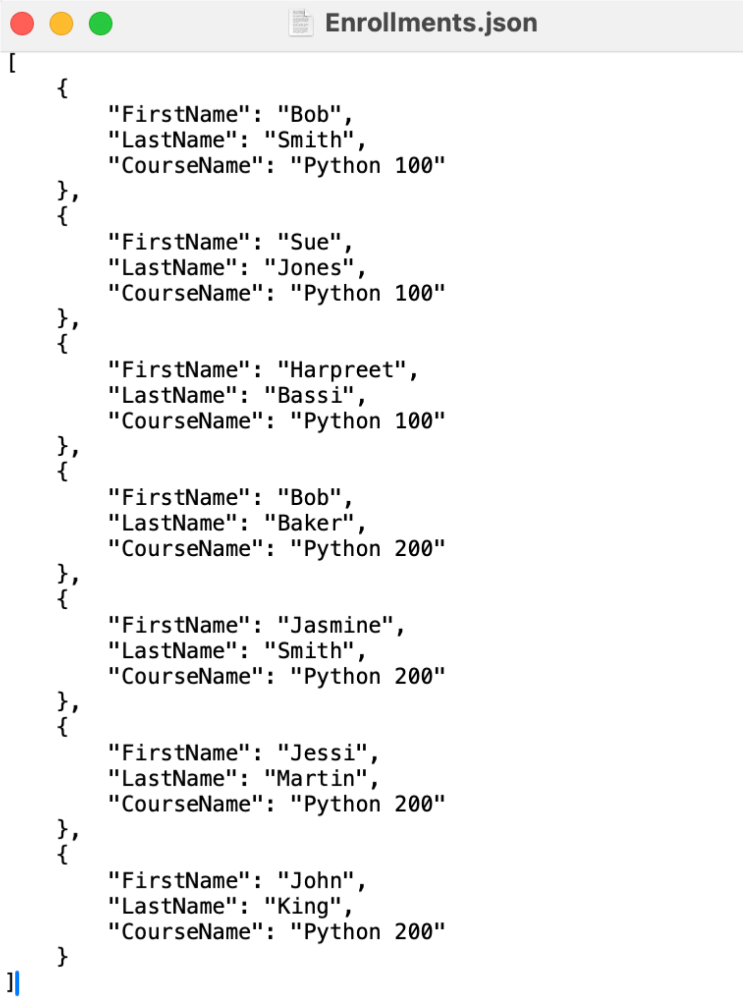
File Processing

Once I ran the code on PyCharm and Terminal, I then went to my Python folder which would have the JSON file saved. As seen in Figure 10, I could see the JSON file that was labeled as "Enrollments.json" just like how I had indicated in my code. I opened the file, which then opened to the TextEdit in Mac, as shown in Figure 11. As you can see in Figure 10 and 11, the code successfully implemented the information in a JSON file.



Name	Date Modified
Assignment06.py	Today at 5:27 PM
Enrollments.json	Today at 5:31 PM

Figure 10: Screenshot of Python folder which includes the JSON file.



```
[
    {
        "FirstName": "Bob",
        "LastName": "Smith",
        "CourseName": "Python 100"
    },
    {
        "FirstName": "Sue",
        "LastName": "Jones",
        "CourseName": "Python 100"
    },
    {
        "FirstName": "Harpreet",
        "LastName": "Bassi",
        "CourseName": "Python 100"
    },
    {
        "FirstName": "Bob",
        "LastName": "Baker",
        "CourseName": "Python 200"
    },
    {
        "FirstName": "Jasmine",
        "LastName": "Smith",
        "CourseName": "Python 200"
    },
    {
        "FirstName": "Jessi",
        "LastName": "Martin",
        "CourseName": "Python 200"
    },
    {
        "FirstName": "John",
        "LastName": "King",
        "CourseName": "Python 200"
    }
]
```

Figure 11: Screenshot of JSON file opened in TextEdit on Mac.

Summary

Applying the Module 06 lecture notes and video, I was able to implement and execute an effective Python program that demonstrated my understanding of functions, classes, and the separation of concerns programming pattern. Also, learned how to implement this data in a JavaScript Object Notation (JSON) file and GitHub. This led me to effectively create a JSON data file of all the information needed regarding a student’s registration for a Python course.

References:

- https://www.learnpython.org/en/Functions