Name: Nelida Romo Tijerina
Date: November 20, 2023
Course: IT FDN 110 B Au 23: Foundations of Programming: Python

Assignment 06 – Functions, Classes, and using the Separation of Concerns Pattern

Introduction

The objective of this assignment is to create a Python program that demonstrates using constants, variables, and print statements to display a message about a student's registration for a Python course.

This program is very similar to Assignment05, but **It adds the use of functions, classes, and using the separation of concerns pattern**

This document includes a description of the challenges that I faced while coding this assignment.

Topics included in this document are about challenges that I had while learning about how to separate my code into Data, Processing and Presentation (Input/Output), also to use functions and how to call them from the main body of the program also to make the groups of functions associated with each of the classes: the fileprocessor class or with the IO class.

1. Traceback Error: NoneType object is not iterable.

I was having the error showed in **Figure 1**, because the variable that I was using in the for loop had not a proper iterable value.

```
Debug - Contents of students: None
Here are all the rows of the data from the file:
Traceback (most recent call last):
  File "/Users/nely/Desktop/PythonAssignments/Assignment06c2.py", line 177, in
<module>
    for student_data_2 in students:
TypeError: 'NoneType' object is not iterable
```

Figure 1. Traceback error: "NoneType" object is not iterable

2. Duplicating students list.

Another error that I was having when reading or writing to the JSON file is that I was duplicating the list students and the reason was because I was appending the new data student_data to students but in addition I was writing the command: return students. Then I was having the double in JSON file of what was in the students list.

3. Previous content was being deleted in the JSON file.

I was deleting previous data in the JSON file, because I was only printing the last student registered. In **Figure 2a,** included below, it is showed the code that I wrote to do that.

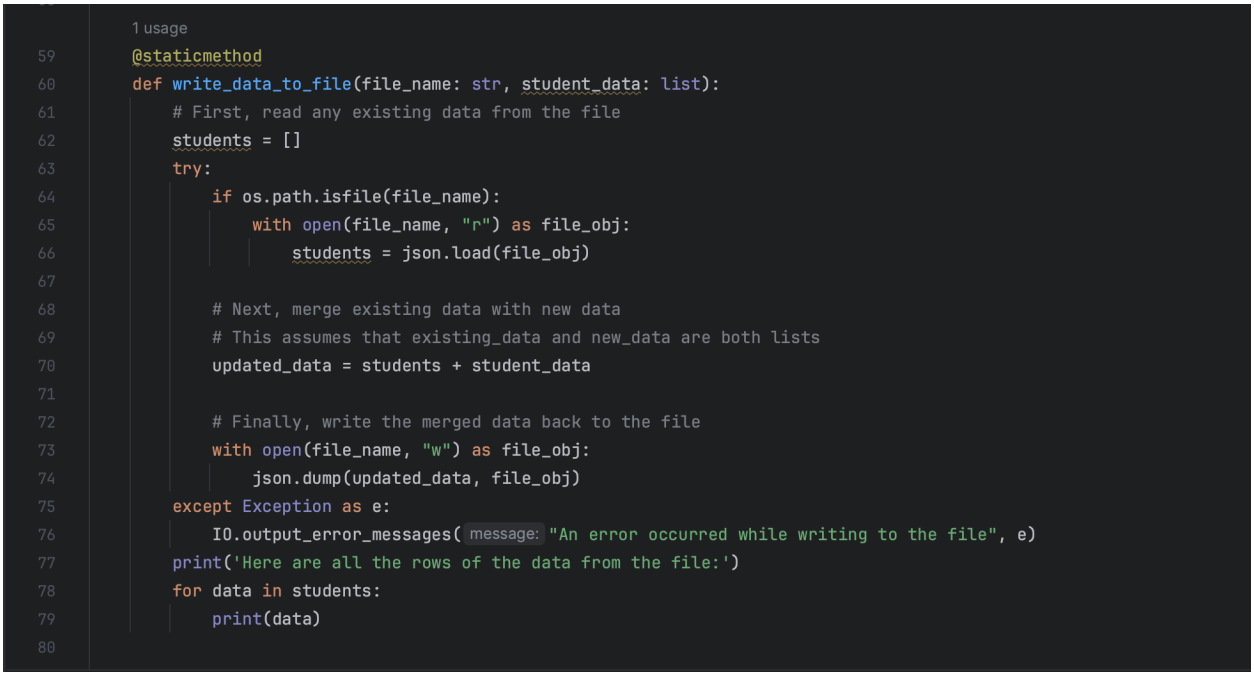
```
@staticmethod
def write_data_to_file(file_name: str, student_data: list):
  try:
    with open(FILE_NAME_JSON, "w") as file_obj:
      json.dump(students, file_obj)
  except TypeError as e:
    IO.output_error_messages("Please check that the data is a valid JSON format",
e)
  except Exception as e:
    IO.output_error_messages("There was a non-specific error!", e)
  print('Here are all the rows of the data from the file:')
  for student_data in students:
      print(student_data)
```

Figure 2a. Previous content being deleted

To solve that first I tried with using “a” instead of “w”, however that did not work, because it was braking my JSON file so what I did instead was the following steps in this order: read, update and write to update what I had in my JSON data with new data.

Then, I had to read the existing data, update the existing data with the new data and write the updated data back to the file.

You can find my final code for this part in **Figure 2b.**



```
1 usage
@staticmethod
def write_data_to_file(file_name: str, student_data: list):
    # First, read any existing data from the file
    students = []
    try:
        if os.path.isfile(file_name):
            with open(file_name, "r") as file_obj:
                students = json.load(file_obj)

        # Next, merge existing data with new data
        # This assumes that existing_data and new_data are both lists
        updated_data = students + student_data

        # Finally, write the merged data back to the file
        with open(file_name, "w") as file_obj:
            json.dump(updated_data, file_obj)
    except Exception as e:
        IO.output_error_messages( message: "An error occurred while writing to the file", e)
    print('Here are all the rows of the data from the file:')
    for data in students:
        print(data)
```

Figure 2b. Final Code: Read, update and write to my JSON file

4. Traceback error for menu choice 3

In this part the challenge that I had is that some times when I entered data for a student here, the student appeared duplicated in the list students, after saving and then selecting Menu Option 2.

What I did in this section that solved the problem is that I created and empty dictionary for each student registered, including this line inside the loop for each iteration. Please see Line 52 in **Figure 3**.

```
Enter your menu choice number: 3
Traceback (most recent call last):
  File "/Users/nely/Desktop/PythonAssignments/Assignment06.py", line 206, in
<module>
    FileProcessor.write_data_to_file(file_name=FILE_NAME_JSON,
student_data=students)
TypeError: write_data_to_file() got an unexpected keyword argument 'file_name'
```

Figure 3. Traceback Error Menu Choice 3: Argument file_name

What I found, is that this discrepancy indicated that I was not having this error with the function itself but rather with how the function was being called.

What I was doing is that I was assigning the function to the variable students.

students = FileProcessor.write_data_to_file.

To solve this, I only deleted the list students and I only wrote the name of the function.

5. Printing nothing – Menu Option 2

I was printing only empty list and dictionaries from the JSON file. What I did was a debug printing what I had in students using the code showed in **Figure 4** below.

```
FileProcessor.read_data_from_file(file_name=FILE_NAME_JSON, student_data=s

# Debugging print statement
print('Debug - Contents of students:', students)

print('Here are all the rows of the data from the file:')
for student_data in students:
    print(student_data)
```

Figure 4. Debugging the contents of students

That was helpful because I was able to see if I there was something of if was empty. I confirmed that students list was empty.
What I was doing is that (I do not know why) I only had:

```
json.load(file_obj)
```

Without assigning this to any variable! I think this part was deleted by mistake.

You can see my code for this section in **Figure 5.**

Also, the way worked for me for calling the function is showed in the same **Figure 5**. In **Figure 6** you can see my function that reads from JSON file.

```python
elif menu_choice == "2":
    if 'FirstName' in student_data and student_data['FirstName'] == "":
        print("There is no student data, select option 1 to provide data")
        continue
    else:

        students = FileProcessor.read_data_from_file(file_name=FILE_NAME_JSON, student_data=students)

        # Debugging print statement
        print('Debug - Contents of students:', students)

        print('Here are all the rows of the data from the file:')
        for student_data_2 in students:
            print(student_data_2)
```

Figure 5. Menu Option 2: Presenting Data.

```python
    @staticmethod
    def read_data_from_file(file_name: str, student_data: list):
        import json
        try:
            with open(file_name, "r") as file_obj:
                students = json.load(file_obj)
            # new_student_data = student_data
            # students.append(new_student_data)
        except FileNotFoundError as e:
            IO.output_error_messages( message: "Text file must exist before running this script!", e)
        except Exception as e:
            IO.output_error_messages( message: "There was a non-specific error!", e)
        return students
```

Figure 6. Read data from JSON file

6. There was extra data being saved into JSON file

When I was doing the Menu choice 2, I was having the error showed in **Figure 7**.

```
Enter your menu choice number: 2
There was a non-specific error!: Extra data: line 1 column 476 (char 475)
Debug - Contents of students: []
Here are all the rows of the data from the file:
```

Figure 7. Extra data being saved in JSON file

What was happening was that there were empty curly brackets after each dictionary in the JSON file. Please see **Figure 8**.

{"FirstName": "Nelly", "LastName": "Romo", "CourseName": "Chem"}, {}, {"FirstName": "A", "LastName": "B", "CourseName": "C"}, {}

Figure 8. Curly brackets empty saved into the JSON file

Additionally, I was still having duplicates of all the content in students so what I found is that I was appending twice the data! I was using the IO.input as showed in **Figure 9** and after that I was doing the students.append as well after that last line showed in **Figure 9**, I deleted the append and only left the IO.input.

I did not know that IO.input already appends the student_data to students.

All my previous problems disappeared.

```
if menu_choice == "1":
    IO.input_student_data(students)
```

Figure 9. IO.input_student_data

7. Memory cache differs from content in JSON file.

I was having differences between the file JSON and memory cache which caused errors when executing my program.

To solve this I imported OS and used os.path.isfile as seen in **Figure 10** below. I submitted a query to chatGPT to solve this part. That solved my problem of these differences.

```python
@staticmethod
def write_data_to_file(file_name: str, student_data: list):
    # First, read any existing data from the file
    students = []
    try:
        if os.path.isfile(file_name):
            with open(file_name, "r") as file_obj:
                students = json.load(file_obj)

        # Next, merge existing data with new data
        # This assumes that existing_data and new_data are both lists
        updated_data = students + student_data

        # Finally, write the merged data back to the file
        with open(file_name, "w") as file_obj:
            json.dump(updated_data, file_obj)
    except Exception as e:
        IO.output_error_messages( message: "An error occurred while writing to the file", e)
    print('Here are all the rows of the data from the file:')
    for data in students:
        print(data)
```

Figure 10. os.path.isfile

Summary

The objective of this assignment was as in the previous assignments, to create a Python program that demonstrates using constants, variables, and print statements to display a message about a student's registration for a Python course.

This program is very similar to Assignment05, but **It adds the use of functions, classes, and using the separation of concerns pattern**

This document included a description of the challenges that I faced while performing this assignment.

It is worth to mention that this assignment was very relevant in my learning about classes and also, about learning the art of separating into very structured parts in the program such as Data, Processing and Presentation (Input/Output).

This assignment was also especially challenging for me. I required triple the time that I needed for the previous assignments. However, I enjoyed greatly while doing this Assignment.

Citations

1. Writing professional papers:
https://www.youtube.com/watch?v=9ojhSW9ljjo&feature=youtu.be

2. Open AI ChatGPT, Oct. 2023, chat.openai.com/chat: A few aspects of this assignment were informed by queries submitted to the ChatGPT.

3. Slides and videos from class, laboratories and Demo/Videos of the course in this Module 6.