Name: Phuong Nguyen
Date: 5/21/2025
Course: Foundations Of Programming: Python

Assignment 05 – Dictionaries, files and exception handling

Introduction

This week, I learned how to work with json files and how to use dictionaries and write script to handle error exceptions. I learned how to import json files and how to make changes to it. I also learned about the GitHub website that developers used to store and share their code.

Writing the Script

This week's assignment is similar to Assignment 04 except we have to add dictionaries and error exceptions. The basic information was already given with the starter file, so I copied over the header, the constants and variables over and updated it with what is required for this assignment. In addition I have to add a script to import the json file over and used the enrollments.json that was already given (Figure 1).

Assignment05.py | Enrollments.json

```
# ------------------------------------------------------------------------------------------ #
# Title: Assignment05
# Desc: This assignment demonstrates using dictionaries, files, and exception handling
# Change Log: (Who, When, What)
#   PNguyen,5/21/2025,Created Script
#   <Your Name Here>,<Date>, <Activity>
# ------------------------------------------------------------------------------------------ #

import json

# Define the Data Constants
MENU: str = ("---- Course Registration Program ----.\nSelect from the following menu:.\n1. Register a Student for a Course..\n"
             "2. Show current data.\n3. Save data to a file.\n4. Exit the program.\n---------------------------------------- ")

# Define the Data Constants
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
json_data: str = ''  # Holds combined string data in json
file = None  # Holds a reference to an opened file.
menu_choice: str  # Hold the choice made by the user.

# Extract the data from the file
try:
    file = open(FILE_NAME, "r")
    print(type(file))
    students = json.load(file)
    file.close()
except Exception as e:
    print("Error: There was a problem with reading the file.")
    print("Please check that the file exists and that it is in a json format.")
    print("--Technical Error Message--")
    print(e.__doc__)
    print(e.__str__())
    file.close()
```

Figure 1: Basic Information

My next task is similar to last week which is to write a code that displays the menu for the user to choose from and enter their inputs. Depending on what they choose and their inputs, different questions/options are displayed. In addition, this week I had to add an error handling to it, that will display an error if the value entered is incorrect. To accomplish this, I used the try function and if value is not alpha character, there will be an error raised. I also asked it to print the error message to the user. (Figure 2).

```
# Present and Process the data
while (True):

    # Present the menu of choices
    print(MENU)
    menu_choice = input("What would you like to do: ")

    # Input user data
    if menu_choice == "1":

        try:
            student_first_name = input("Enter the student's first name: ")
            if not student_first_name.isalpha():
                raise ValueError("The last name should not contain numbers.")
            student_last_name = input("Enter the student's last name: ")
            if not student_last_name.isalpha():
                raise ValueError("The last name should not contain numbers.")
            course_name = input("Please enter the name of the course: ")
            student_data = {"FirstName": student_first_name,
                            "LastName": student_last_name,
                            "CourseName": course_name}
            students.append(student_data)
            print(f"You have registered {student_first_name} {student_last_name} for {course_name}.")
        except ValueError as e:
            print(e)
            print("-- Technical Error Message --")
            print(e.__doc__)
            print(e.__str__())
        except Exception as e:
            print("Error: There was a problem with your entered data.")
            print("-- Technical Error Message --")
            print(e.__doc__)
            print(e.__str__())
        continue
```

Figure 2: While loop / error exception

Menu options 2 is the same as last week with no changes. Menu 3 now involves saving the file to the json file instead of the csv. I also added similar error exception if there is a problem with saving the data to the json file. (Figure 3).

```
    # Present the current data
    elif menu_choice == "2":

        # Process the data to create and display a custom message
        print("-"*50)
        for student in students:
            print(f'Student {student["FirstName"]} '
                  f'{student["LastName"]} is enrolled in {student["CourseName"]}')
        print("-"*50)
        continue

    # Save the data to a file
    elif menu_choice == "3":

        try:
            file = open(FILE_NAME, "w")

            json.dump(students, file)
            file.close()
            print("The following data was saved to file!")
            for student in students:
                (f'Student {student["FirstName"]} '
                    f'{student["LastName"]} is enrolled in {student["CourseName"]}')
        except Exception as e:
            if file.closed == False:
                file.close()
            print("Error: There was a problem with writing to the file.")
            print("Please check that the file is not open by another program.")
            print(e.__doc__)
            print(e.__str__())
        continue
```

Figure 3: Options 2 and 3

Lastly menu options 4 is to exit the program. This didn't change from last week. (Figure 4).

```
109     # Stop the loop
110     elif menu_choice == "4":
111         break  # out of the loop
112     else:
113         print("Please only choose option 1, 2, 3, or 4")
114
115 print("Program Ended")
116
```

Figure 4 Option 4 Exit the program

Testing the Script

With my script in place, it is time for me to test and make sure it works in both PyCharm and the Command Prompt. After some trial and error, I was able to successfully get my script to work in PyCharm, it also opened and saved the json file as intended (Figure 5). It also worked in the command prompt, as you can see the json file was updated and saved to a later time when I ran it (Figure 6). The error message also works as intended (Figure 7).

```
----------------------------------------
What would you like to do: 1
Enter the student's first name: Bobby
Enter the student's last name: Tran
Please enter the name of the course: Python 100
You have registered Bobby Tran for 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.
----------------------------------------
What would you like to do: 2
--------------------------------------------------
Student Bob Smith is enrolled in Python 100
Student Sue Jones is enrolled in Python 100
Student Phuong Nguyen is enrolled in Python 100
Student Bobby Tran 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.
----------------------------------------
What would you like to do: 3
The following data was saved to file!
---- 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
Program Ended
```

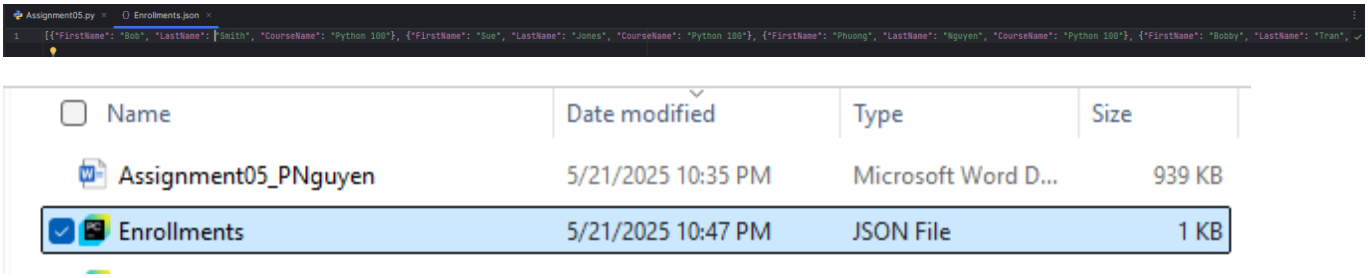

Figure 5: Test in PyCharm

```
---- 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
Enter the student's first name: Test
Enter the student's last name: One
Please enter the name of the course: Python 100
You have registered Test One for 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.
-----------------------------------------
What would you like to do: 2
-------------------------------------------------
Student Bob Smith is enrolled in Python 100
Student Sue Jones is enrolled in Python 100
Student Phuong Nguyen is enrolled in Python 100
Student Bobby Tran is enrolled in Python 100
Student Test One 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.
-----------------------------------------
What would you like to do: 3
The following data was saved to file!
---- 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
Program Ended
```

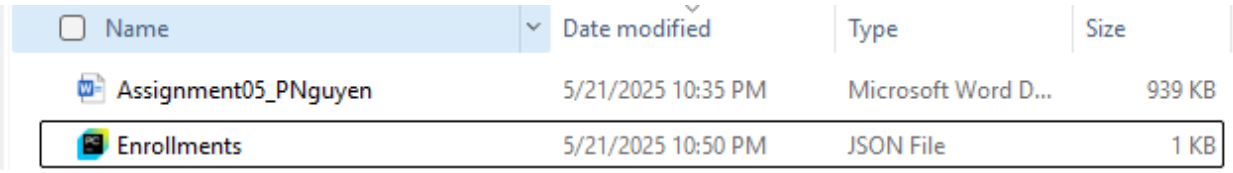

Figure 6: Test in Command Prompt

```
---------------------------------------
What would you like to do: 1
Enter the student's first name: 123
The last name should not contain numbers.
-- Technical Error Message --
Inappropriate argument value (of correct type).
The last name should not contain numbers.
---- 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:
```

Figure 7: Error Message

Summary

This week's assignment was very challenging especially the dictionaries and the error handling. I had to reference back to the notes and videos many times. This module taught me about the GitLab resource and how to store my code. I also learned how to work with json files and make changes to it.