Paulo Roberto Barreto Alves

08/13/2025

Foundations of Python Programming

Assignment 05

https://github.com/pagh59258/Python110

Advanced Collections and Error Handling

Introduction

On module 05, I acquired knowledge on some Python programming tools and techniques. With this new knowledge, I created a Python script very similar to the one I have developed in Module 04, but adding some extra techniques, such as: **the use of data processing using dictionaries, JSON files and exception handling.**

Reading Python 100 Module 05 notes.doc file

On module 05 itself, I have acquired knowledge of the following areas:

- Dictionaries Collections
  - Adding and Removing Data
  - Using Dictionaries with File Data
- JSON files
  - JSON vs CSV
  - Working with JSON Files
- Structured Error Handling (Try-Except)
  - Avoiding Errors
- Managing Code Files
  - Network File Sharing
  - Cloud File Sharing
  - GitHub
  - Using GitHub's Website

Performing Module 05 labs

Throughout the module, I was asked to apply the learned knowledge into some practice labs, such as:

- Mod05-Lab01-Working with Dictionaries and files
- Mod05-Lab02-Working with JSON Files
- Mod05-Lab03-Working with Exceptions

- Mod05-Lab04-Saving Files to GitHub

Creating a Python script

I created a new file named Assignment05.py, which as the extension indicates, it is a Python script. This Python script demonstrates using constants, variables, and print statements to display a message about a student's registration for a course. This program is very similar to Assignment04, but it adds the use of data processing using dictionaries and JSON files, and also using exception handling.

Basically, this Python script does this:

- When the program starts, the contents of the "Enrollments.json" file is automatically read into the **students** two-dimensional list of dictionary rows using the json.load() function.
- To avoid errors, we need to have some data already populated on the "Enrollments.json" file before starting script execution.
- Script uses error handling try:Except blocks to verify different kind of errors during execution.
- Display a menu with 4 options:
    - Critique menu option entered to only accept digits 1, 2, 3 or 4;
    - If any other value is entered, an error is displayed, and the menu option is required again.
- Regarding Menu Option “1”:
    - User is allowed to make several enrollments;
    - Script does a critique to not allow invalid fields:
        - Student’s first name cannot be numeric or blank;
        - Students’ last name cannot be numeric or blank;
        - Course name cannot be blank.
    - Data collected is added to a dictionary named student_data and then student_data is added to the students two-dimensional list of dictionaries rows.
- Regarding Menu Option “2”:
    - The program presents a string by formatting the collected data using the print() function;
    - The program uses the print() function to show a string of comma-separated values for each row collected in the students variable.
- Regarding Menu Option “3”:
    - Program opens the "Enrollments.json" file in write mode using the open() function;
    - It writes the contents of the students variable to the file, using the json.dump() function;
    - Next, it closes the file using the close() method;
    - Finally, the program displays which data was written to the file using the students variable.
- Regarding Menu Option “4”:
    - It checks if there is unsaved data and offer the user the option to save it or not;
    - Enrollments.json file is successfully closed;
    - All enrollment data stored on Enrollments.json file is displayed;

- Program is terminated

On Figure 1 below, we can see the initial Enrollment.json file (provided inside assignment zip file), has to contain some data:

Figure 1: Enrollment.csv initial file

On Figure 2 below, we can see the basic coding used for Script header and constants & variables definition:

```python
# ------------------------------------------------------------------------------------------
# Title: Assignment05
# Desc: This assignment demonstrates using dictionaries, files, and exception handling
# Change Log: (Who, When, What)
#   PAlves,8/12/2025,Created Script
#   <Your Name Here>,<Date>, <Activity>
# ------------------------------------------------------------------------------------------

#Import JSON Python module

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
----------------------------------------
'''
# 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.
file = None  # Holds a reference to an opened file.
menu_choice: str  # Hold the choice made by the user.
student_data: dict = {}  # one row of student data
students: list = []  # a table of student data
students_on_file: list = [] #shows current student registrations saved on file
pend_save: str = ''

```

Figure 2: Assignment05.py Python script header, constants and variables

On Figure 3 below, we can see the activities performed at script startup and before entering menu loop.

```
40
41  ##file = open(FILE_NAME, "r")
42  ##for row in file.readlines():
43  ##    # Transform the data from the file
44  ##    student_data = row.split(',')
45  ##    student_data = [student_data[0], student_data[1], student_data[2].strip()]
46  ##    # Load it into our collection (list of lists)
47  ##    students.append(student_data)
48  ##file.close()
49
50  # Loading JSON file content into students list
51
52  try:
53      file = open(FILE_NAME, "r")
54      students = json.load(file)
55      file.close()
56
57  except FileNotFoundError as e:
58      print("Text file must exist before running this script!\n")
59      print("-- Technical Error Message -- ")
60      print(e, e.__doc__, type(e), sep='\n')
61
62  except Exception as e:
63      print("There was a non-specific error!\n")
64      print("-- Technical Error Message -- ")
65      print(e, e.__doc__, type(e), sep='\n')
66
67  finally:
68      if file.closed == False:
69          file.close()
70
```

Figure 3: PyCharm – Assignment05.py Python script startup activities

On Figure 4 below, we can see the script menu handling (option 1).

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

    # Present the menu of choices
    print(MENU)
    menu_choice = input("Enter your menu choice number: ")

    # 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("Input Error: Student's first name must be alphanumeric!")
            else:
                student_last_name = input("Enter the student's last name: ")
                if not student_last_name.isalpha():
                    raise ValueError("Input Error: Student's last name must be alphanumeric!")
                else:
                    course_name = input("Please enter the name of the course: ")
                    if course_name == "":
                        raise ValueError("Input Error: Course Name cannot be blank!")

        except ValueError as e:
            print("-" * 65)
            print(e)
            print("-" * 65)
            continue

        except Exception as e:
            print("-" * 65)
            print("There was a non-specific error!\n")
            print("-- Technical Error Message -- ")
            print(e, e.__doc__, type(e), sep='\n')
            print("-" * 65)

        student_data = {'FirstName':student_first_name,
                        'LastName':student_last_name,
                        'CourseName':course_name}
        students.append(student_data)
        print("-" * 65)
        print(f"You have registered {student_first_name} {student_last_name} for {course_name}.")
        print("-" * 65)
        continue
```

Figure 4: PyCharm – Assignment05.py Python script menu handling (option 1)

On Figure 5 below, we can see the script menu handling (option 2).

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

        # Process the data to create and display a custom message
        print("-"* 65)
        print("Here is the list of students currently registered for courses:")
        print("-"* 65)
        for student in students:
            print(f"{student['FirstName']} {student['LastName']} is enrolled in {student['CourseName']}.")
        print("-"* 65)
        print("IMPORTANT")
        print("- Some of these registrations might not be yet saved to file")
        print("- Make sure you use menu option 3 to save all registrations to file")
        print("-"* 65)
        continue
```

Figure 5: PyCharm – Assignment05.py Python script menu handling (option 2)

On Figure 6 below, we can see the script menu handling (option 3).

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

        try:
            #  Save the data to the file
            file = open(FILE_NAME, "w")
            json.dump(students, file, indent=2)
            file.close()

        except TypeError as e:
            print("Please check that the data is a valid JSON format\n")
            print("-- Technical Error Message -- ")
            print(e, e.__doc__, type(e), sep='\n')

        except Exception as e:
            print("-- Technical Error Message -- ")
            print("Built-In Python error info: ")
            print(e, e.__doc__, type(e), sep='\n')

        finally:
            if file.closed == False:
                file.close()

        print("-"* 65)
        print("The following data was saved to file:")
        print("-"* 65)
        for student in students:
            print(f"{student['FirstName']} {student['LastName']} ,"
                  f"is enrolled in {student['CourseName']}.")
        print("-"* 65)
        continue
```

Figure 6: PyCharm – Assignment05.py Python script menu handling (option 3)

On Figure 7 below, we can see the script menu handling (option 4) and final activity.

```
    # Stop the loop
    elif menu_choice == "4":
        #Check if there are information not yet saved on file
        file = open(FILE_NAME, "r")
        students_on_file = json.load(file)
        file.close()
        if(len(students) != len(students_on_file)):
            print("-" * 65)
            print("Warning: There are students registrations not yet saved on file.")
            print("-" * 65)
            pend_save=input("Do you want to save the data? (y/n): ")
            if(pend_save == "y"):
                file = open(FILE_NAME, "w")
                json.dump(students, file, indent=2)
                file.close()
                print("-" * 65)
                print("Thank you for saving the data!")
                print("-" * 65)
            else:
                print("-" * 65)
                print("As per your choice, pending registrations will not be saved to file!")
                print("-" * 65)
        break  # closes loop and terminate script execution
    else:
        print("-"* 65)
        print("Input Error: Please only choose option 1, 2, 3, or 4")
        print("-"* 65)

print("-"* 65)
print("Assignment05.py Python Script Execution Successfully Terminated")
print("-" * 65)
```

Figure 7: PyCharm – Assignment05.py Python script menu handling (option 4) and final activiy)

Then I executed the script via PyCharm and via Windows command shell.

Executing script on PyCharm

Figure 8 shown below displays the Assignment05.py Python script menu using PyCharm.

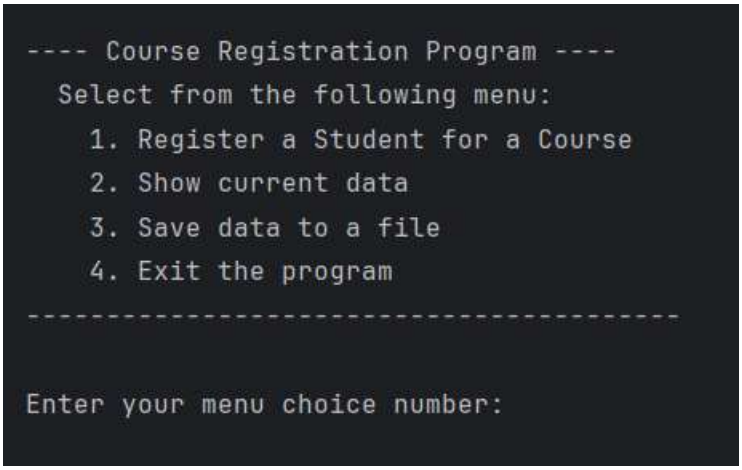

Figure 8: PyCharm – Assignment05.py script menu

Figure 9 shown below displays the critique if an invalid menu option is chosen (either alphabetic options or integers different from 1, 2, 3 or 4):

```
---- 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:
----------------------------------------------------------------
Input Error: Please only choose option 1, 2, 3, or 4
----------------------------------------------------------------

---- 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: a
----------------------------------------------------------------
Input Error: Please only choose option 1, 2, 3, or 4
----------------------------------------------------------------
```

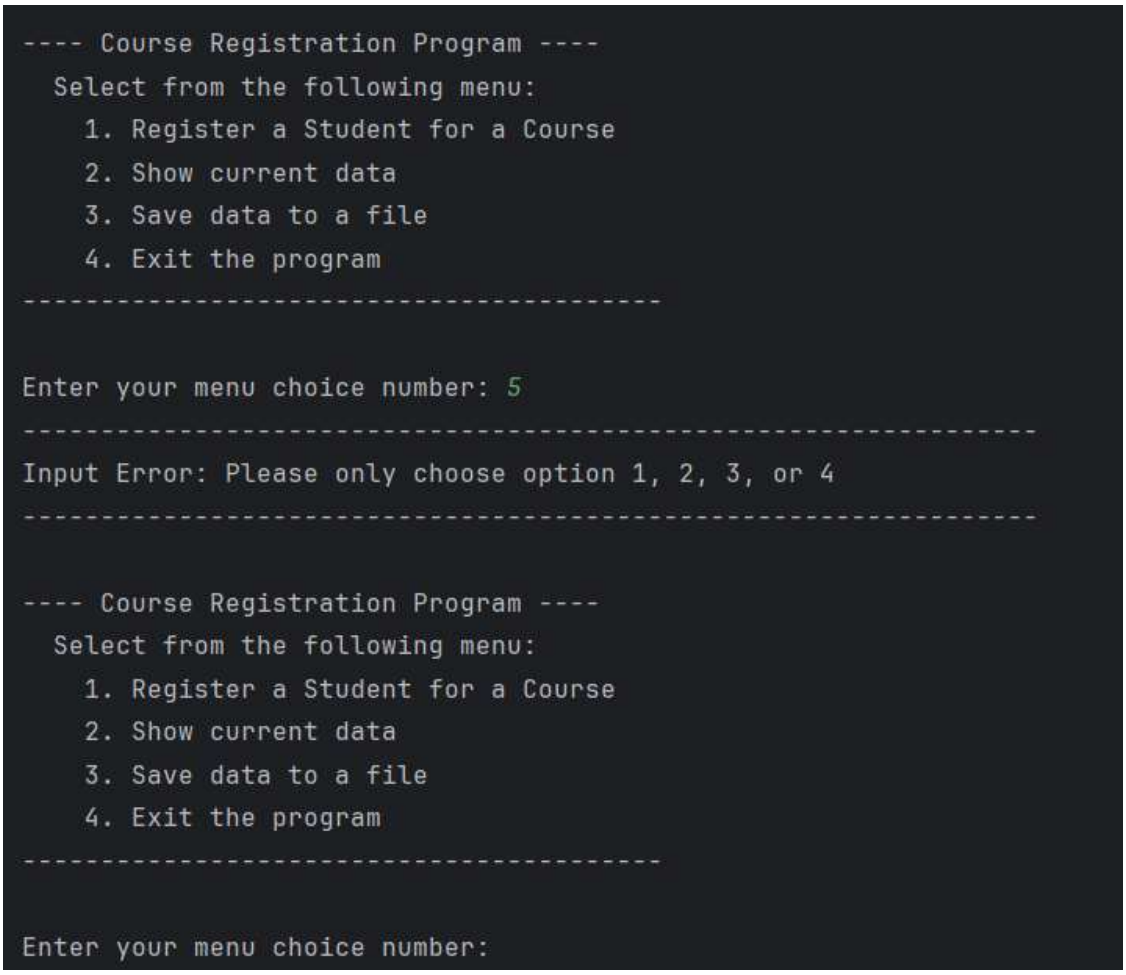

Figure 9: PyCharm – Assignment05.py script menu invalid option critique

Figure 10 below shows error validation on Student's First and Last name and Course name invalid entries:

```
---- 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:
------------------------------------------------------------------
Input Error: Student's first name must be alphanumeric!
------------------------------------------------------------------

---- 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: 123
------------------------------------------------------------------
Input Error: Please only choose option 1, 2, 3, or 4
------------------------------------------------------------------

---- 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:
```

```
---- 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: Vic
Enter the student's last name:
------------------------------------------------------------------
Input Error: Student's last name must be alphanumeric!
------------------------------------------------------------------

---- 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: Vic
Enter the student's last name: 123
------------------------------------------------------------------
Input Error: Student's last name must be alphanumeric!
------------------------------------------------------------------

---- 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:
```

```
---- 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: Vic
Enter the student's last name: Vu
Please enter the name of the course:
------------------------------------------------------------------
Input Error: Course Name cannot be blank!
------------------------------------------------------------------

---- 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 10: PyCharm – Assignment05.py script validation on Student's First and Last name and Course name invalid entries

Figure 11 below shows a couple of valid registration made via Menu Option 1:

```
---- 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: Vic
Enter the student's last name: Vu
Please enter the name of the course: Java 303
------------------------------------------------------------------
You have registered Vic Vu for Java 303.
------------------------------------------------------------------

---- 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: John
Enter the student's last name: Doe
Please enter the name of the course: Arts 101
------------------------------------------------------------------
You have registered John Doe for Arts 101.
------------------------------------------------------------------
```

Figure 11: PyCharm – Assignment05.py script shows a couple of valid registration via menu option1

Figure 12 shown below displays the output of Menu option 2, when the current data is shown:

```
---- 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: 2
-------------------------------------------------------------------
Here is the list of students currently registered for courses:
-------------------------------------------------------------------
Bob Smith is enrolled in Python 100.
Sue Jones is enrolled in Python 100.
Vic Vu is enrolled in Java 303.
John Doe is enrolled in Arts 101.
-------------------------------------------------------------------
IMPORTANT
- Some of these registrations might not be yet saved to file
- Make sure you use menu option 3 to save all registrations 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
----------------------------------------

Enter your menu choice number:
```

Figure 12: PyCharm – Assignment05.py script menu option 2 output showing current data

Figure 13 shown below displays the output of Menu option 3, when data is saved to JSON file and displayed:

```
---- 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: 3
-------------------------------------------------------------------
The following data was saved to file:
-------------------------------------------------------------------
Bob Smith ,is enrolled in Python 100.
Sue Jones ,is enrolled in Python 100.
Vic Vu ,is enrolled in Java 303.
John Doe ,is enrolled in Arts 101.
-------------------------------------------------------------------

---- 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 13: PyCharm – Assignment05.py script menu options 3 output

Figure 14 shown below displays the menu options 4 (when there is pending data to be 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
----------------------------------------

Enter your menu choice number: 1
Enter the student's first name: John
Enter the student's last name: Tester
Please enter the name of the course: Python 110
------------------------------------------------------------------
You have registered John Tester for Python 110.
------------------------------------------------------------------

---- 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: 4
------------------------------------------------------------------
Warning: There are students registrations not yet saved on file.
------------------------------------------------------------------
Do you want to save the data? (y/n): y
------------------------------------------------------------------
Thank you for saving the data!
------------------------------------------------------------------
------------------------------------------------------------------
Assignment05.py Python Script Execution Successfully Terminated
------------------------------------------------------------------

Process finished with exit code 0
```

Figure 14: PyCharm – Assignment05.py script menu option 4 output (when there is pending data)

Figure 15 shown below displays the menu option 4, which exit the program (and no pending enrollment):

```
---- 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: 4
-------------------------------------------------------------------
Assignment05.py Python Script Execution Successfully Terminated
-------------------------------------------------------------------

Process finished with exit code 0
```

Figure 15: PyCharm – Assignment05.py script menu option 4 with no pending enrollment

Figure 16 shown below displays the menu option 4, which exit the program (without any pending enrollment):

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

Enter your choice: 4
-------------------------------------------------------------------------
Exiting...Thanks for using the enrollment program!

Process finished with exit code 0
```

Figure 16: PyCharm – Assignment05.py script menu option 4 without any pending enrollment

Figure 17 shown below displays the final content of the Enrollments.json file:

```
[
  {
    "FirstName": "Bob",
    "LastName": "Smith",
    "CourseName": "Python 100"
  },
  {
    "FirstName": "Sue",
    "LastName": "Jones",
    "CourseName": "Python 100"
  },
  {
    "FirstName": "Vic",
    "LastName": "Vu",
    "CourseName": "Java 303"
  },
  {
    "FirstName": "John",
    "LastName": "Doe",
    "CourseName": "Arts 101"
  },
  {
    "FirstName": "John",
    "LastName": "Tester",
    "CourseName": "Python 110"
  },
  {
    "FirstName": "Mary",
    "LastName": "Jones",
    "CourseName": "Cooking 303"
  },
  {
    "FirstName": "Susan",
    "LastName": "Boyle",
    "CourseName": "Singing 201"
  }
]
```


Figure 17: PyCharm – Assignment05.py script – final content on Enrollment.json file

Executing script on Windows command shell (cmd)

Figure 18 shown below, displays the successful execution of Assignment05.py Python script using Windows command shell (cmd).

```
---- 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: Mary
Enter the student's last name: Jones
Please enter the name of the course: Cooking 303
----------------------------------------------------------------
You have registered Mary Jones for Cooking 303.
----------------------------------------------------------------

---- 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: Susan
Enter the student's last name: Boyle
Please enter the name of the course: Singing 201
----------------------------------------------------------------
You have registered Susan Boyle for Singing 201.
----------------------------------------------------------------

---- 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: 2
----------------------------------------------------------------
Here is the list of students currently registered for courses:
----------------------------------------------------------------
Bob Smith is enrolled in Python 100.
Sue Jones is enrolled in Python 100.
Vic Vu is enrolled in Java 303.
John Doe is enrolled in Arts 101.
John Tester is enrolled in Python 110.
Mary Jones is enrolled in Cooking 303.
Susan Boyle is enrolled in Singing 201.
----------------------------------------------------------------
```

```
IMPORTANT
- Some of these registrations might not be yet saved to file
- Make sure you use menu option 3 to save all registrations 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
----------------------------------------

Enter your menu choice number: 3
-------------------------------------------------------------------
The following data was saved to file:
-------------------------------------------------------------------
Bob Smith ,is enrolled in Python 100.
Sue Jones ,is enrolled in Python 100.
Vic Vu ,is enrolled in Java 303.
John Doe ,is enrolled in Arts 101.
John Tester ,is enrolled in Python 110.
Mary Jones ,is enrolled in Cooking 303.
Susan Boyle ,is enrolled in Singing 201.
-------------------------------------------------------------------

---- 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: 4
-------------------------------------------------------------------
Assignment05.py Python Script Execution Successfully Terminated
-------------------------------------------------------------------
```

Figure 18: Windows command shell (CMD) - Execution of Assignment05.py script

Summary

The creation and execution of this third Python script was a great way to enhance my Python programming using some of the new knowledge I learned on Module 05, including **the use of data processing using dictionaries, JSON files and error/exception handling**.