Adnan Dawood

May 29, 2024

Foundations of Programming: Python

Assignment 07

The link to the repository in GitHub: https://github.com/adnandawood/IntroToProg-Python-Mod07

Classes and Objects

Introduction to Classes and Objects in Python

In Python, classes and objects are core concepts of object-oriented programming (OOP). Classes serve as blueprints for creating objects, while objects are instances of classes with their own unique attributes and behaviors.

A class defines the structure and behavior of objects by encapsulating data (attributes) and functionality (methods) into a single entity. When we create an object from a class, we're essentially creating a specific instance of that class, inheriting its properties and behaviors.

Classes and objects provide a powerful way to organize and manage code, promoting modularity, reusability, and scalability in software development. They allow developers to model real-world entities and interactions, making it easier to design and maintain complex systems.

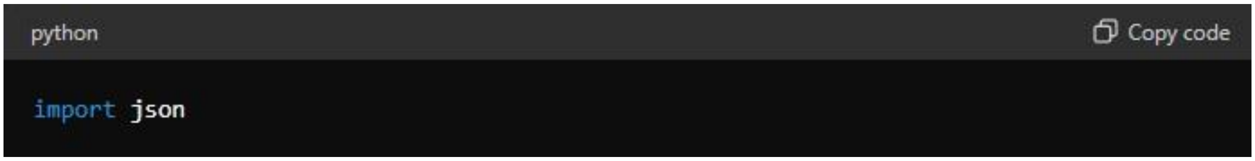
In Python, defining a class is straightforward, using the class keyword followed by the class name and its attributes and methods. Objects are then created by calling the class constructor, which initializes the object's state.

Overall, understanding classes and objects in Python is essential for building robust and flexible applications, enabling developers to create well-structured and maintainable codebases.

Drafting the Code:

This Python code represents a simple course registration program. Let's break it down:

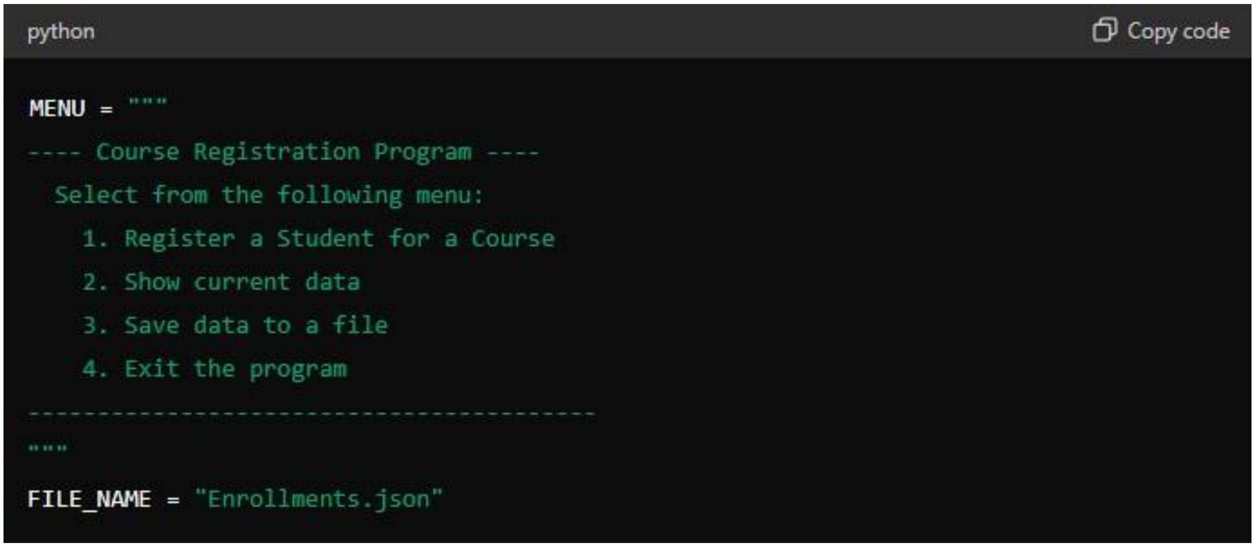
Import JSON Module:



```python
import json
```

This line imports the JSON module, which is used for encoding and decoding JSON data.

Constants

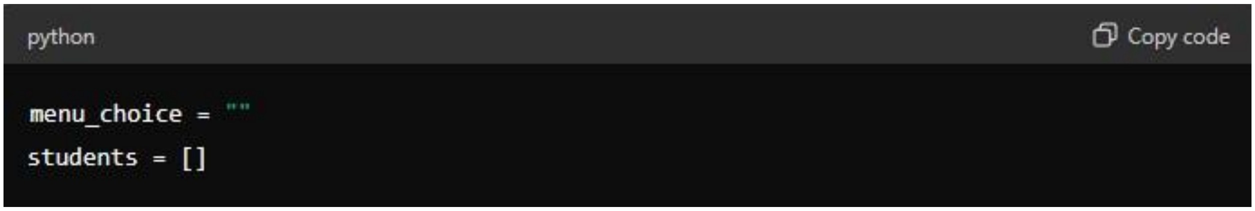


```python
MENU = """
---- Course Registration Program ----
  Select from the following menu:
    1. Register a Student for a Course
    2. Show current data
    3. Save data to a file
    4. Exit the program
------------------------------------------
"""
FILE_NAME = "Enrollments.json"
```

MENU: A multi-line string representing the menu options for the program.

FILE_NAME: The name of the JSON file where student data will be stored.

Variables:



```python
menu_choice = ""
students = []
```

menu_choice: Stores the user's menu choice.

students: A list to store student objects.

Classes:

FileProcessor: Handles file operations such as reading from and writing to a JSON file.

IO: Handles input/output operations such as displaying menu, getting menu choice, displaying student data, and getting student data from the user.

Person: Base class representing a person with properties for first name, last name, and course name.

Student: Inherits from the Person class. Represents a student with methods to extract data and input student details.

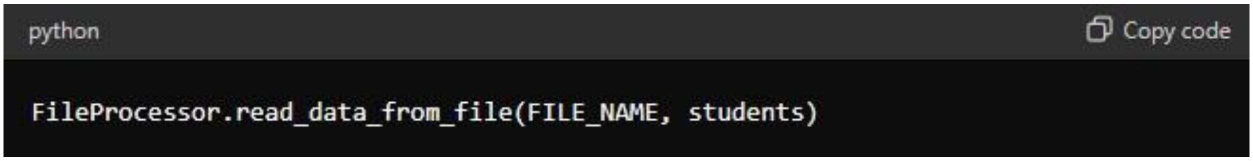
Main Program:

Main Function:

```python
def main():
    ...
```

The main function of the program.

Reading Data from File:



```python
FileProcessor.read_data_from_file(FILE_NAME, students)
```

Reads student data from the JSON file and populates the students list.

Menu Loop:

```python
while True:
    ...
```

Continuously displays the menu and waits for user input until the user chooses to exit.

Menu Choices:

Register a student:

Gets student details from the user and adds the student to the students list.

Show Current Data:

Displays the details of all registered students.

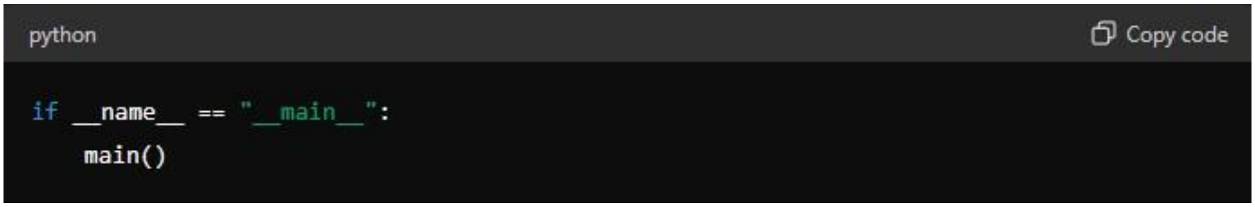
Save Data to a File:

Writes the student data to the JSON file.

Exit the Program:

Exit the program.

Main Program Execution:



```python
if __name__ == "__main__":
    main()
```

Executes the main() function when the script is executed directly.

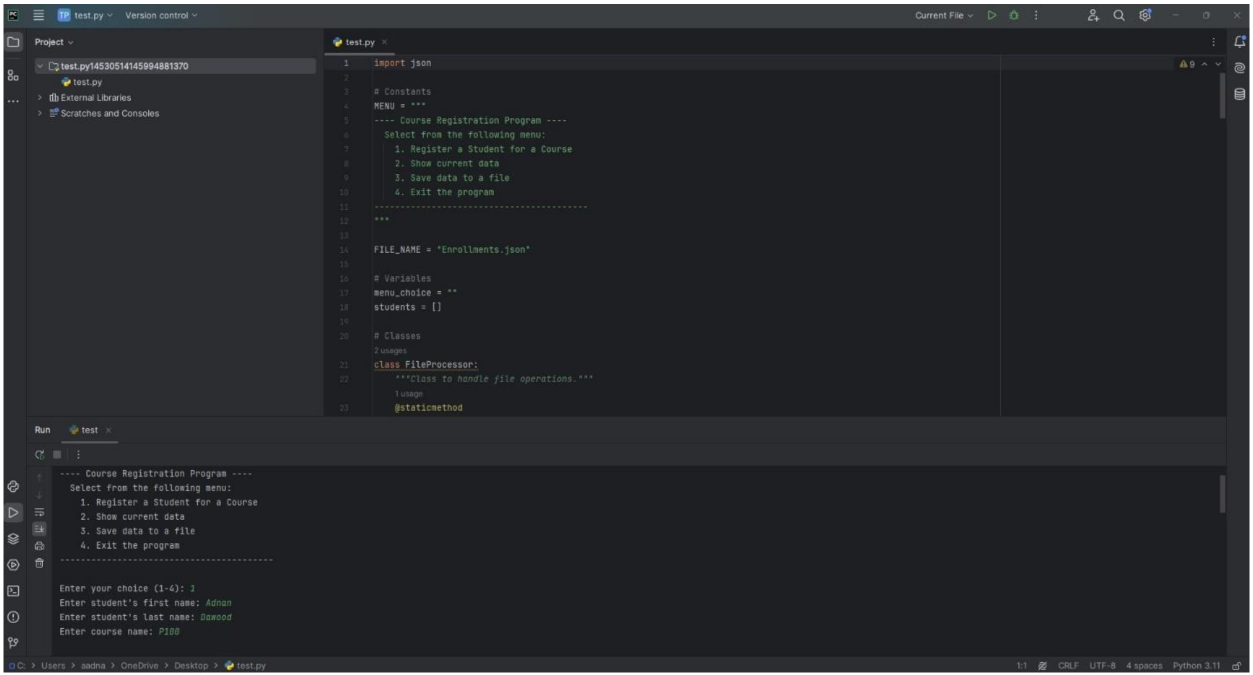
This program provides an interface for registering students for courses, displaying current registrations, saving data to a file, and exiting the program.

Testing the script and the findings:

I used PyCharm to evaluate my script. Also, the script was evaluated in terminal.

Here is how my script looks like and its output in PyCharm:

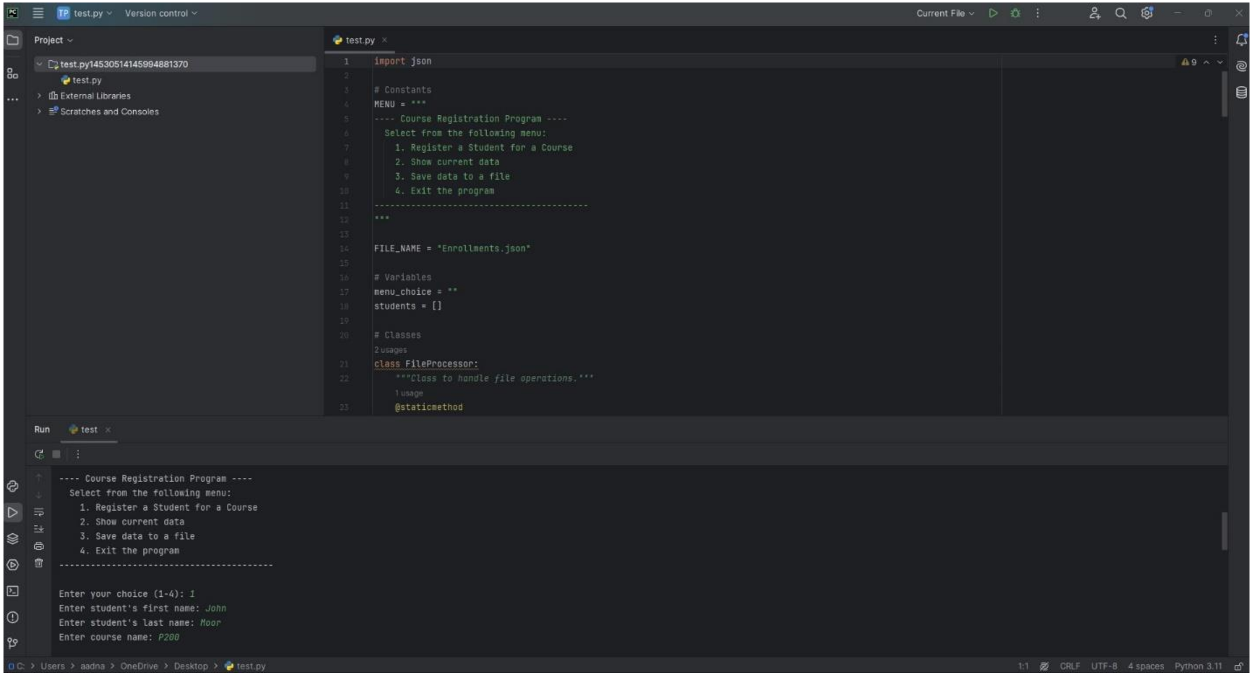
a- Running the program and entering 1 for option one “Register a Student for a Course” as shown in the picture below:



test.py
Version control
Current File
Project
test.py14530514145994881370
test.py
External Libraries
Scratches and Consoles
Run
test
C: > Users > aadna > OneDrive > Desktop > test.py
1:1 CRLF UTF-8 4 spaces Python 3.11

```
import json

# Constants
MENU = """
---- 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 = "Enrollments.json"

# Variables
menu_choice = ""
students = []

# Classes
class FileProcessor:
    """Class to handle file operations."""
    @staticmethod
```

```
---- 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 choice (1-4): 1
Enter student's first name: Adnan
Enter student's last name: Dawood
Enter course name: P100
```

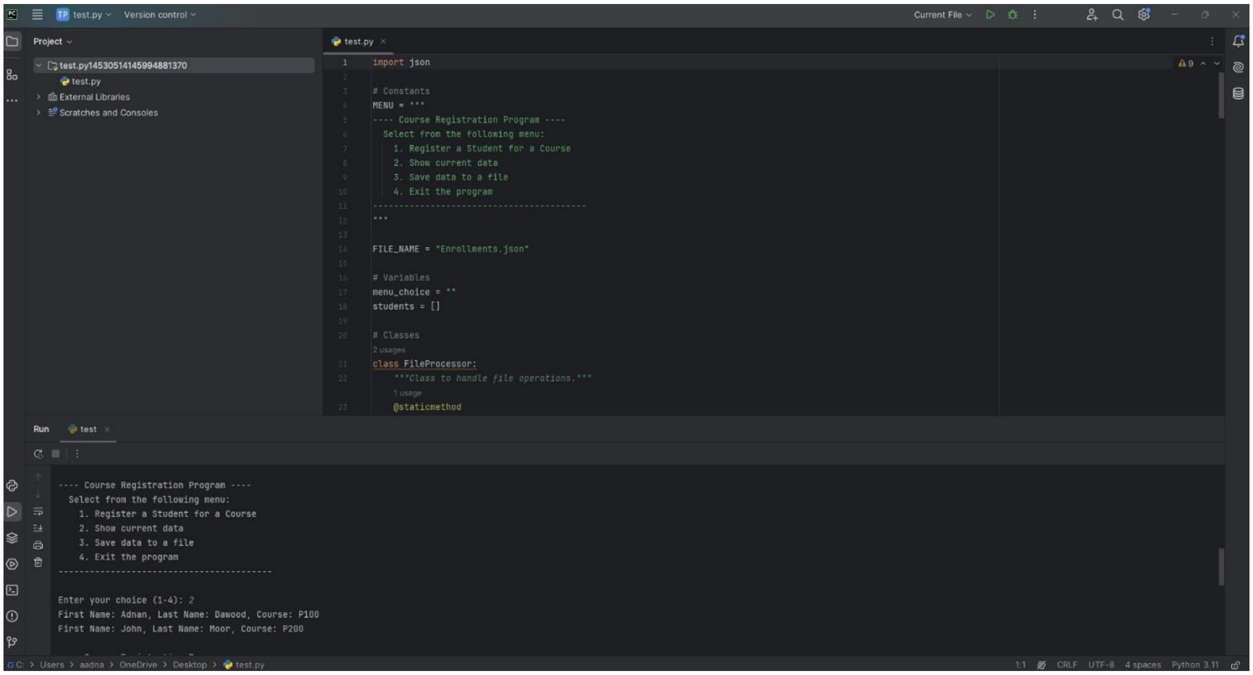


test.py
Version control
Current File
Project
test.py14530514145994881370
test.py
External Libraries
Scratches and Consoles
Run
test
C: > Users > aadna > OneDrive > Desktop > test.py
1:1 CRLF UTF-8 4 spaces Python 3.11

```
import json

# Constants
MENU = """
---- 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 = "Enrollments.json"

# Variables
menu_choice = ""
students = []

# Classes
class FileProcessor:
    """Class to handle file operations."""
    @staticmethod
```

```
---- 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 choice (1-4): 1
Enter student's first name: John
Enter student's last name: Moor
Enter course name: P200
```

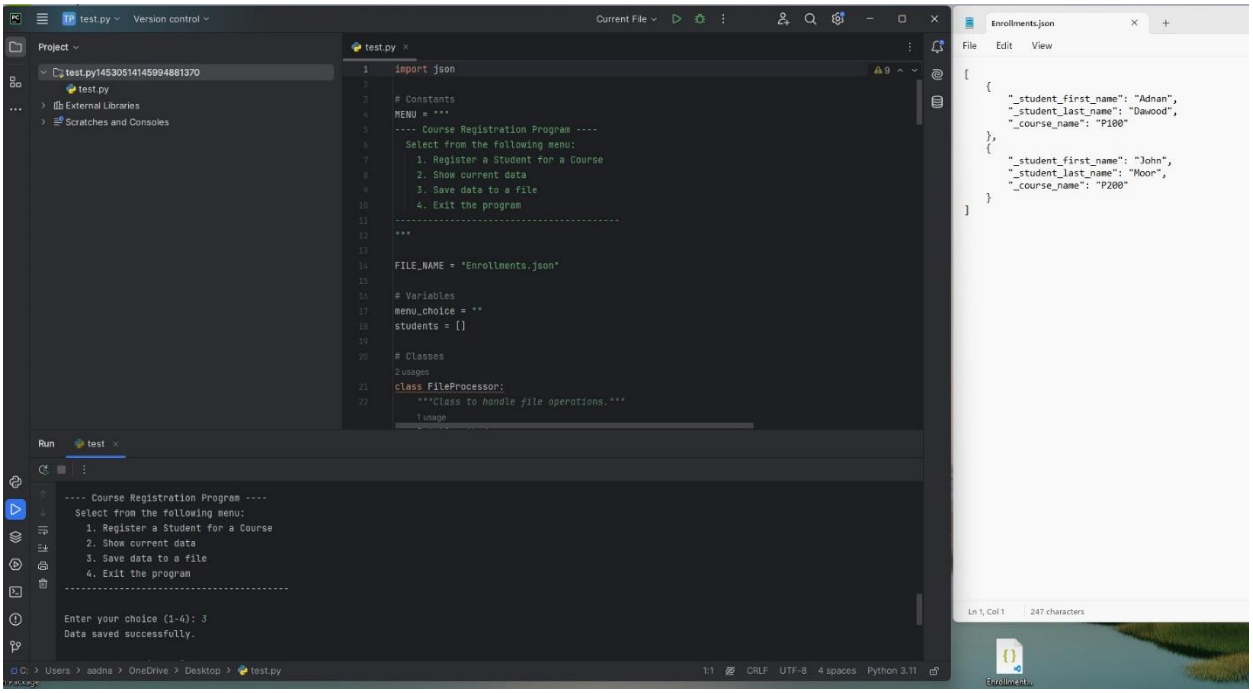
b- Entering 2 for option two “Show current data” as shown in the picture below:



```python
import json

# Constants
MENU = """
---- 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 = "Enrollments.json"

# Variables
menu_choice = ""
students = []

# Classes
class FileProcessor:
    """Class to handle file operations."""
    @staticmethod
```

```
---- 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 choice (1-4): 2
First Name: Adnan, Last Name: Dawood, Course: P100
First Name: John, Last Name: Moor, Course: P200
```

c- Entering 3 for option three “Save data to a file (check the right side of the picture)” as shown in the picture below:

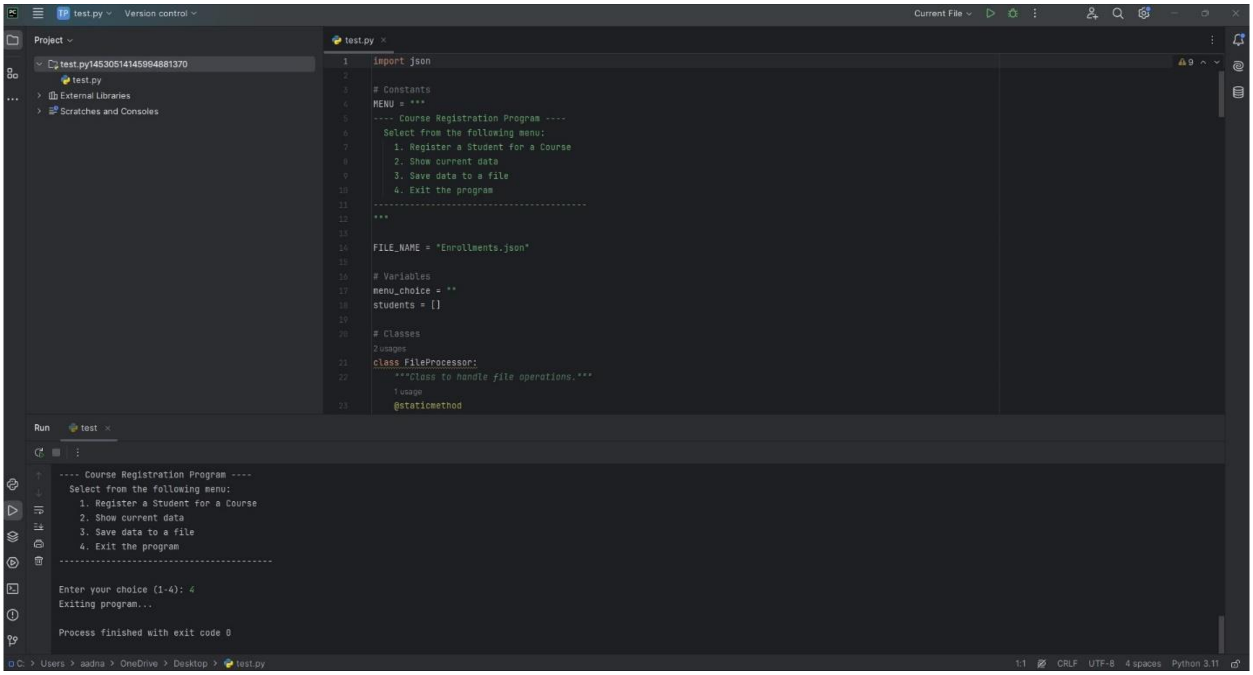


```python
import json

# Constants
MENU = """
---- 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 = "Enrollments.json"

# Variables
menu_choice = ""
students = []

# Classes
class FileProcessor:
    """Class to handle file operations."""
```

```
---- 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 choice (1-4): 3
Data saved successfully.
```

```json
[
    {
        "_student_first_name": "Adnan",
        "_student_last_name": "Dawood",
        "_course_name": "P100"
    },
    {
        "_student_first_name": "John",
        "_student_last_name": "Moor",
        "_course_name": "P200"
    }
]
```

d- Entering 4 for option four “Exit the program” as shown in the picture below:

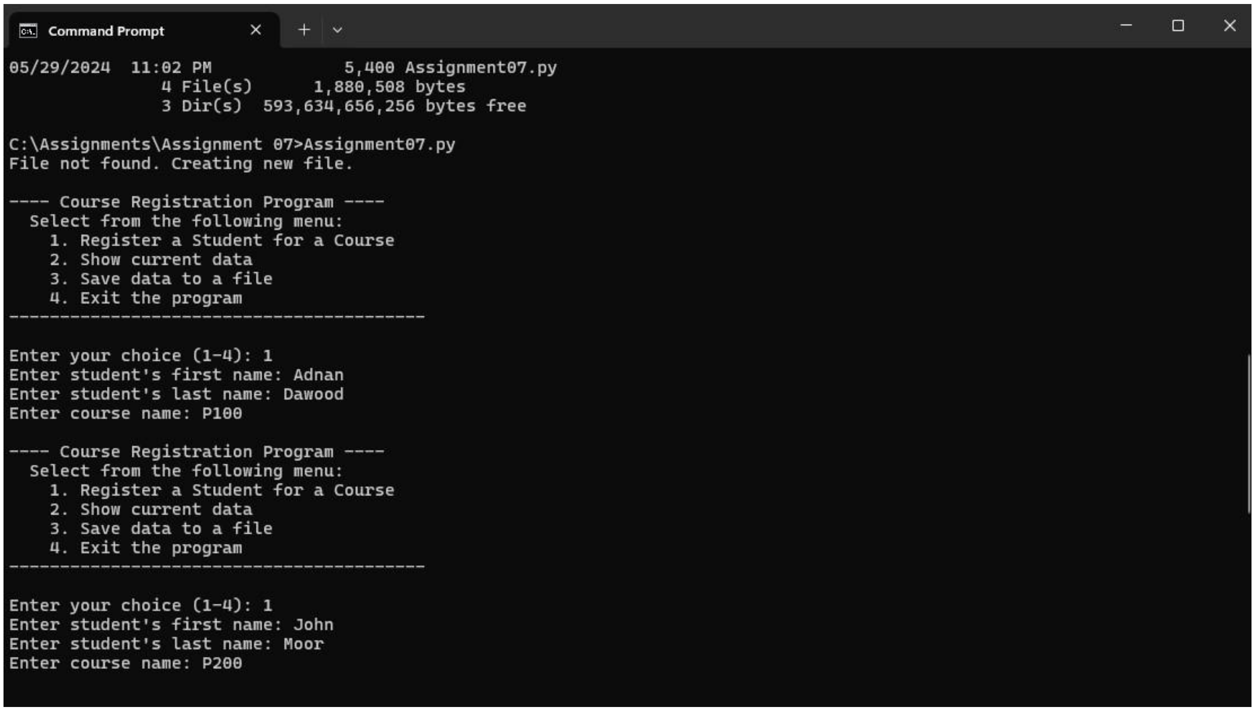


```python
import json

# Constants
MENU = """
---- 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 = "Enrollments.json"

# Variables
menu_choice = ""
students = []

# Classes
class FileProcessor:
    """Class to handle file operations."""
    @staticmethod
```

```
---- 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 choice (1-4): 4
Exiting program...

Process finished with exit code 0
```

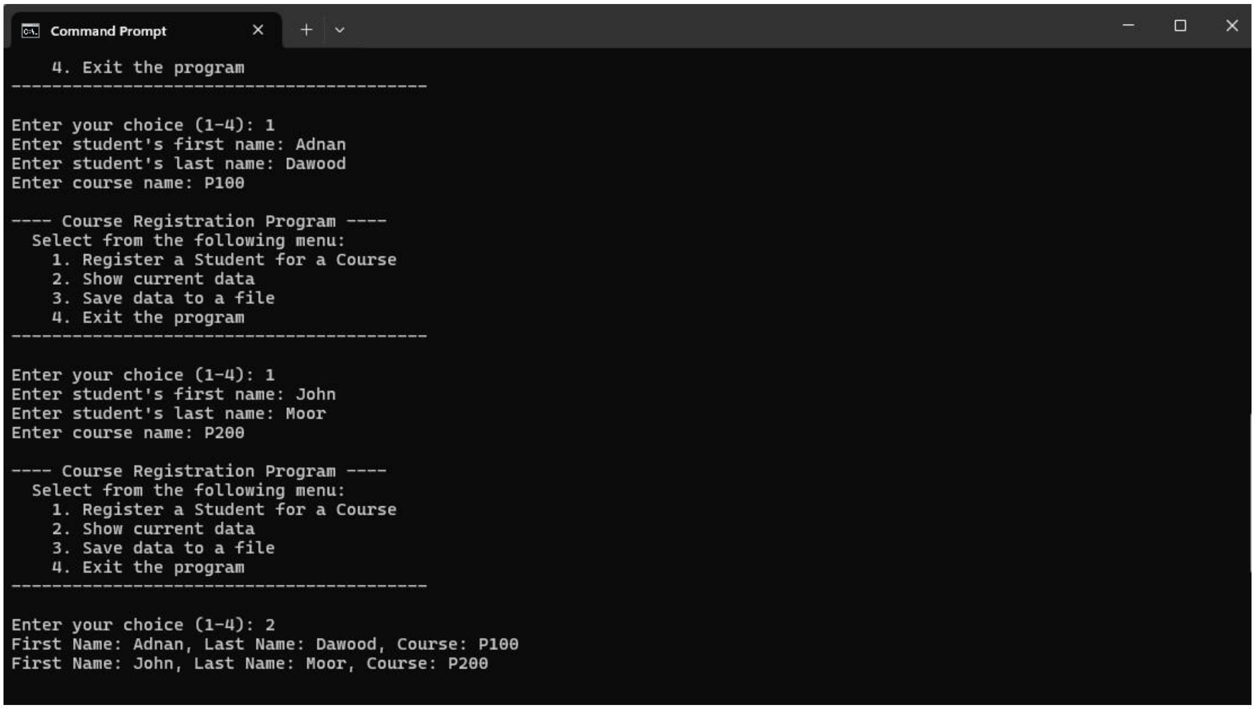
Here is how my script looks like and its output in terminal:

a- Running the program in terminal and entering 1 for option one “Register a Student for a Course” as shown in the picture below:



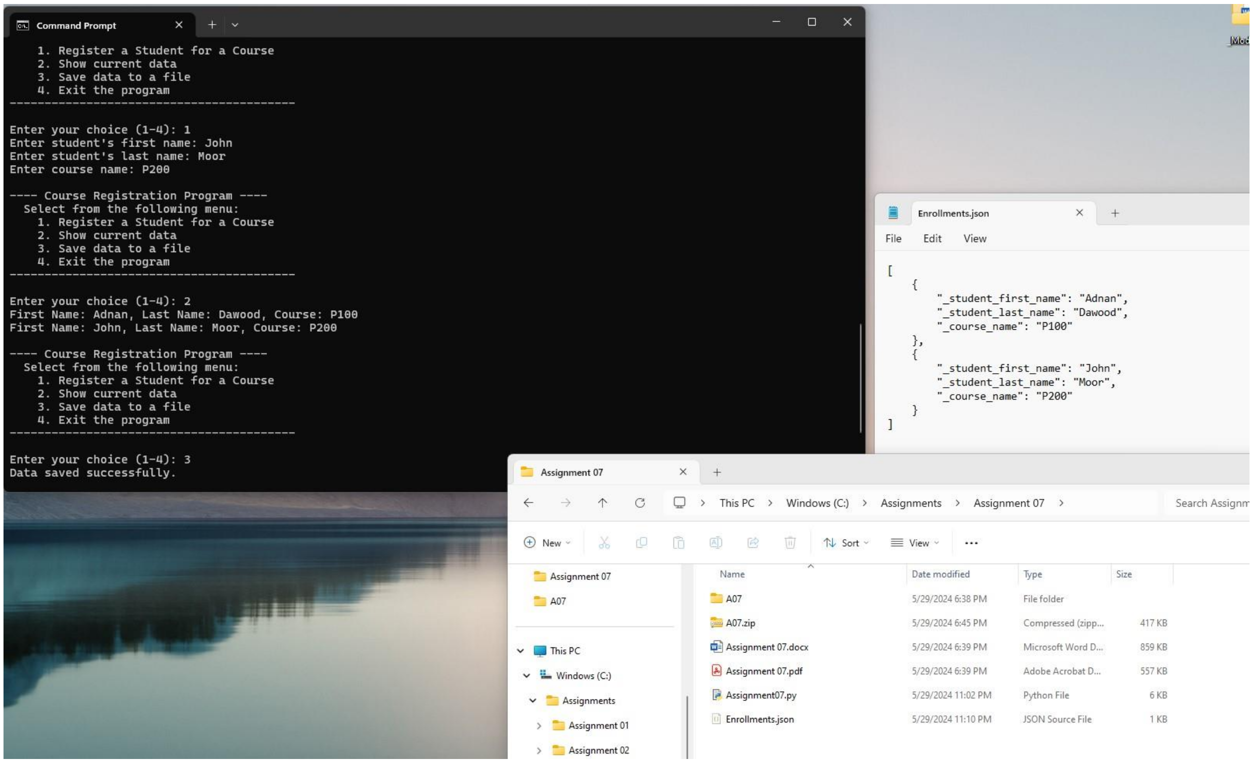
```
05/29/2024  11:02 PM             5,400 Assignment07.py
               4 File(s)      1,880,508 bytes
               3 Dir(s)  593,634,656,256 bytes free

C:\Assignments\Assignment 07>Assignment07.py
File not found. Creating new 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 choice (1-4): 1
Enter student's first name: Adnan
Enter student's last name: Dawood
Enter course name: P100

---- 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 choice (1-4): 1
Enter student's first name: John
Enter student's last name: Moor
Enter course name: P200
```

b- Entering 2 for option two “Show current data” as shown in the picture below:



```
    4. Exit the program
-----------------------------------------

Enter your choice (1-4): 1
Enter student's first name: Adnan
Enter student's last name: Dawood
Enter course name: P100

---- 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 choice (1-4): 1
Enter student's first name: John
Enter student's last name: Moor
Enter course name: P200

---- 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 choice (1-4): 2
First Name: Adnan, Last Name: Dawood, Course: P100
First Name: John, Last Name: Moor, Course: P200
```

c- Entering 3 for option three “Save data to a file (check the right side of the picture)” as shown in the picture below:

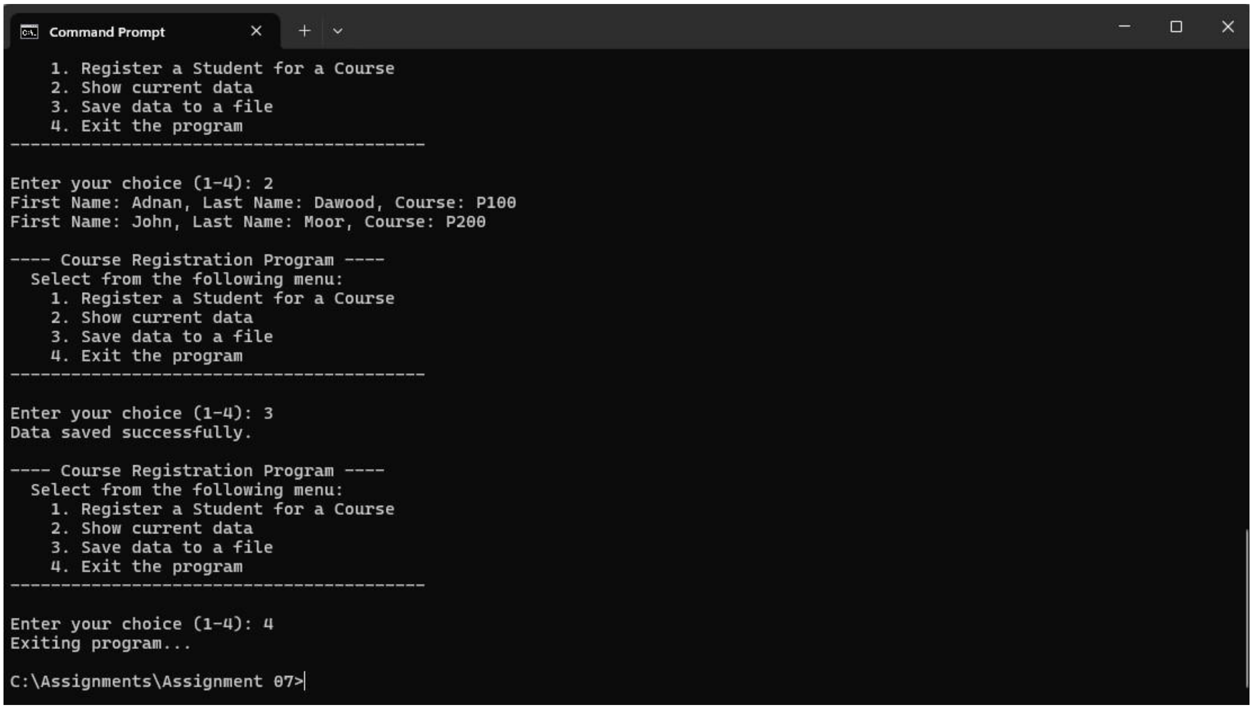


```
    1. Register a Student for a Course
    2. Show current data
    3. Save data to a file
    4. Exit the program
----------------------------------------

Enter your choice (1-4): 1
Enter student's first name: John
Enter student's last name: Moor
Enter course name: P200

---- 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 choice (1-4): 2
First Name: Adnan, Last Name: Dawood, Course: P100
First Name: John, Last Name: Moor, Course: P200

---- 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 choice (1-4): 3
Data saved successfully.
```

```
[
    {
        "_student_first_name": "Adnan",
        "_student_last_name": "Dawood",
        "_course_name": "P100"
    },
    {
        "_student_first_name": "John",
        "_student_last_name": "Moor",
        "_course_name": "P200"
    }
]
```

Name	Date modified	Type	Size
A07	5/29/2024 6:38 PM	File folder	
A07.zip	5/29/2024 6:45 PM	Compressed (zipp...	417 KB
Assignment 07.docx	5/29/2024 6:39 PM	Microsoft Word D...	859 KB
Assignment 07.pdf	5/29/2024 6:39 PM	Adobe Acrobat D...	557 KB
Assignment07.py	5/29/2024 11:02 PM	Python File	6 KB
Enrollments.json	5/29/2024 11:10 PM	JSON Source File	1 KB

d- Entering 4 for option four “Exit the program” as shown in the picture below:



```
    1. Register a Student for a Course
    2. Show current data
    3. Save data to a file
    4. Exit the program
----------------------------------------

Enter your choice (1-4): 2
First Name: Adnan, Last Name: Dawood, Course: P100
First Name: John, Last Name: Moor, Course: P200

---- 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 choice (1-4): 3
Data saved successfully.

---- 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 choice (1-4): 4
Exiting program...

C:\Assignments\Assignment 07>
```

Summary:

This Python code implements a course registration program. Here's a summary of its functionality:

The program consists of several components:

Constants defining the program's menu and the name of the file where student data is stored.

Variables to store the user's menu choice and a list to hold student data.

Classes to handle file operations (FileProcessor), input/output operations (IO), and define a base class for a person (Person) and a derived class for a student (Student).

The FileProcessor class provides methods to read data from and write data to a JSON file.

The IO class handles input/output operations, such as displaying the menu, getting user input for menu choice, and displaying student data.

The Person class serves as a base class with properties for a person's first name, last name, and course name. The student class inherits from Person and adds methods for inputting student data and extracting data from a student object.

The main program loop in the main() function:

Reads student data from a JSON file.

Displays the menu and waits for the user to input a choice.

Based on the user's choice, it either registers a student, shows current student data, saves data to a file, or exits the program.

In summary, this code creates a command-line interface for managing student registrations for courses, allowing users to add new students, view existing registrations, save data to a file, and exit the program.