Serena del Banco

11/17/25

Foundations of Python Programming

Assignment05

Assignment05

Introduction

In this assignment, I demonstrated the use of dictionaries, files, and exception handling.

Import Python modules

First, I imported json and _io modules:

```
import json
from json import JSONDecodeError
import _io
```

Defining constants and variables

Using the Assignment05-Starter.py, I defined the constants and variables below: A couple edits were changing student_data from a list to a dictionary, and changing file = None to file = _io.TextIOWrapper. Brackets [] were used to create empty lists, and {} were used to create empty dictionaries.

```
# Define 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
student_first_name: str = ''
```

```
student_last_name: str = “
course_name: str = ''
student_data: dict = {}
students: list = []  # a table of student data
file = _io.TextIOWrapper  # Holds a reference to an opened file. (TODO: Change this to use _io.TextIOWrapper instead of None)
menu_choice: str  # Hold the choice made by the user.
```

Read “Enrollments.json” into students list

First, I used the try: command to handle errors that may be raised by the code:

```
try:
```

Under that, I wrote code to open the json file and load the data into the students list:

```
file = open(FILE_NAME, "r")
students=json.load(file)
file.close()
```

Next, I included an exception in case a FileNotFoundError is raised. This code displays the error that is raised. If this error is raised, the code then creates the Enrollments.json file:

```
except FileNotFoundError as e:
    print(f"File {FILE_NAME} not found")
    print(e,e.__doc__, type(e), sep='\n')
    print("Creating file since it doesn't exist")
    file=open(FILE_NAME, 'w')
```

Then, I included an exception in case the data in the json file is not valid. The code displays the type of error, then resets the file:

```
except JSONDecodeError as e:
    print(e, e.__doc__, type(e), sep='\n')
    print("Data in file not valid. Resetting it.")
    file = open(FILE_NAME, 'w')
    json.dump(students, file)
```

If the error raised is not FileNotFoundError or JSONDecodeError, then the code prints that the error is an unhandled exception, and prints the type of error thrown:

```
except Exception as e:
    print("Unhandled exception")
    print(e, e.__doc__, type(e), sep='\n')
```

Lastly, if the file is not closed, the code under finally closes the file:

```
finally:
    if not file.closed:
        file.close()
```

Present menu choices

To present the menu choices and prompt the user to enter their information, I wrote the following script:

```
while (True):
    print(MENU)
    menu_choice = input("What would you like to do: ")
```

Menu choice 1

To collect user information, such as first name, last name, and course name, I wrote the following script. The try command was added to handle exceptions, and exceptions added were .isalpha(), which raises a ValueError if the string entered includes non-alphabetic values.

```
if menu_choice == "1”:
    try:
        student_first_name = input("Enter the student's first name: ")
        if not student_first_name.isalpha():
            raise ValueError('First name must be alphabetic')
        student_last_name = input("Enter the student's last name: ")
        if not student_last_name.isalpha():
            raise ValueError('Last name must be alphabetic')
        course_name = input("Please enter the name of the course: ")
```

To store this data in the student_data dictionary, I wrote the following script:

```
        student_data = {"FirstName": student_first_name, "LastName":
student_last_name,"CourseName": course_name}
```

To append this data to the students list and display the data entered into the students list, I wrote the following script:

```
        students.append(student_data)
        print(f"You have registered {student_first_name} {student_last_name} for
{course_name}.")
        continue
```

Menu choice 2

To present the data that has been entered, I prompted the program to print the dictionary elements FirstName, LastName, and CourseName for each row/student:

```
elif menu_choice == "2":
    print("-"*50)
    for student in students:
        print(f"Student {student["FirstName"]} {student["LastName"]} is enrolled in {student['CourseName']}")
    print("-"*50)
    continue
```

Menu choice 3

To open and write in Enrollments.json, I wrote the following code. I added the try command to handle exceptions, and used the json.dump() command to add the contents of students to the json file.

```
elif menu_choice == "3":
    try:
        file = open(FILE_NAME, "w")
        json.dump(students, file)
        file.close()
```

To print what was written in the file, I wrote the following script:

```
        for student in students:
            print(f"Registration info for {student["FirstName"]} {student["LastName"]} has been saved to {FILE_NAME}")
        continue
```

I also included exceptions for TypeError, as well as unhandled exceptions:

```
    except TypeError as e:
        print("Check that data is in valid JSON format")
        print(e, e.__doc__, type(e), sep='\n')
    except Exception as e:
        print("Unhandled exception")
        print(e, e.__doc__, type(e), sep='\n')
    finally:
        if not file.closed:
            file.close()
```

Menu choice 4 and else statement

To stop the loop, I wrote the following script:

```
elif menu_choice == "4":
    break
```

If a menu_choice does not equal 1, 2, 3, or 4, the program prompts the user to enter another option.

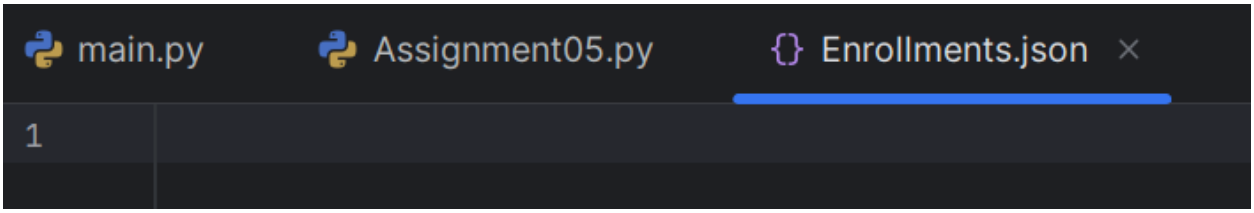
```
else:
    print("Please only choose option 1, 2, 3, or 4")
```

Running the program

I first wanted to test my exceptions code if the Enrollments.json file did not exist. When I delete the Enrollments.json file, this is the result:

```
File Enrollments.json not found
[Errno 2] No such file or directory: 'Enrollments.json'
File not found.
<class 'FileNotFoundError'>
Creating file since it doesn't exist
```

An empty Enrollments.json file is also created with this code:



Next, the menu choices are displayed. If I enter menu choice 1, and enter non-alphabetic characters, this is the result:

```
---- 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: Serena1
First name must be alphabetic
Inappropriate argument value (of correct type).
<class 'ValueError'>
```

If I select menu choice 1 and enter all alphabetic characters, then select menu choice 2, this is the result:

```
What would you like to do: 1
Enter the student's first name: Serena
Enter the student's last name: delBanco
Please enter the name of the course: Python 101
You have registered Serena delBanco for Python 101.
```

```
What would you like to do: 2
------------------------------------------------
Student Serena delBanco is enrolled in Python 101
------------------------------------------------
```

If I enter additional students details under menu choice 1, and then select menu choice 2, this is the result:

```
What would you like to do: 2
-------------------------------------------------
Student Serena delBanco is enrolled in Python 101
Student Galinda Upland is enrolled in Sorcery 101
Student Elphaba Thropp is enrolled in Sorcery 101
-------------------------------------------------
```

If I enter menu choice 3, this is the result:

```
What would you like to do: 3
Registration info for Serena delBanco has been saved to Enrollments.json
Registration info for Galinda Upland has been saved to Enrollments.json
Registration info for Elphaba Thropp has been saved to Enrollments.json
```

If I enter menu choice 4, this is the result:

```
What would you like to do: 4
Program Ended


Process finished with exit code 0
```

In the Enrollments.json file, this is what has been written:

Lastly, this script was uploaded to a GitHub repository here:

[Sdelbanco/IntroToProg-Python-Mod05: This Rep. will be used for reviewing homework files]