Mohammad Ammar Bharmal

March 10, 2024

IT FDN 110 B

Assignment 07 - Attempt2

GitHub URL: https://github.com/bharmaluwedu/IntroToProg-Python-Mod07

Demonstrating the use of the functions, classes, and using the separation of concerns pattern

Introduction

This week is about demonstrating how to use the functions, classes, and using the separation of concerns pattern. The following information is a breakdown on how I wrote this program step-by-step. The program is created in PyCharm 2022.2.1 with Python 3.12.2.

I have the similar program as we did in the class. The additional things are as follows:

1. New json file name as needed.
2. Input from user on course name
3. Additional feature of asking GPA as a fourth element

Since this is my second attempt, I have added all the needed items from rubric to score maximum points which includes complete script, comments and formatting, source control and GitHub page.

Creating the Program

Since there are multiple ways to write a program in Python to achieve the same results, I chose to start it by providing a brief description of what the program is about as a comment in the script. After that, I described what the program does and how to use it. Since the beginning is the same as assignment 1 and 2 so I am directly jumping into the main structure of the program.

Using the *Assignment07_Starter.py*, I completed assignment by creating the classes and functions within the classes. It contains global constants such as FILE_NAME and MENU and other supporting variables including list. It asks the user to choose from four options when the program is run as given below:

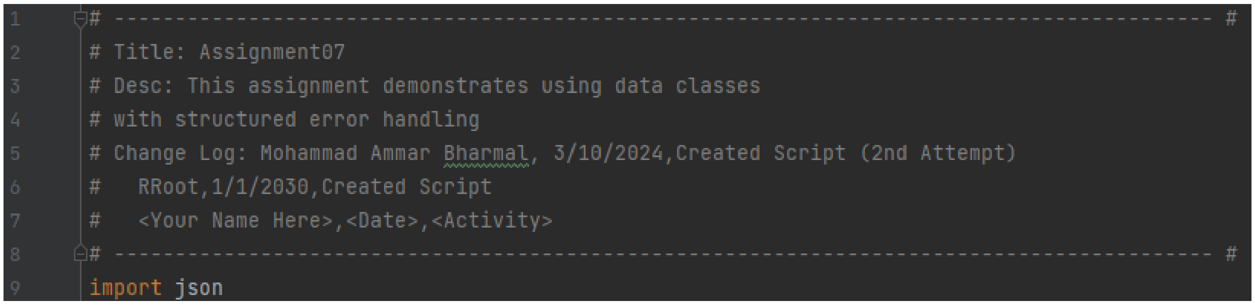
1. Register a Student for a Course
2. Show current data
3. Save data to a file
4. Exit the program

When the program is run, it displays the menu for the user to choose from. On opting menu choice 1 the program prompts the user to enter the student's first name and last name, followed by the course name, using the input() function and stores the inputs in the respective variables. On menu choice 2, the application presents a string by formatting the collected data using the print() function. Data collected for menu choice 1 is added to a two-dimensional list table (list of Student objects). All data in the list is displayed when menu choice 2 is used. When the program starts, the contents of the *Enrollments.json* are automatically read into a two-dimensional list table (a list of Student object rows). On menu choice 3, the program opens a file named *Enrollments.json* in write mode using the open() function. The information is saved in the file and on menu choice 4, the application ends.

In the programs, the application provides structured error handling when the file is read into the list of employee rows, when the user enters a first name and when the user enters a last name. Moreover, the application also provides structured error handling when the user enters a mismatched datatype.

There are two classes: Person and Student. The Person class has only two variables: "*first_name*" and "*last_name*". The Student class borrows the base class from Person and has two more variables: "*course_name*" and "*student_gpa*". The classes also raise value errors if the datatype is not respected as declared. The class *IO* has five functions. They carry out the input and output functionalities in the application. The functions enable menu output, menu choice input, and data input into and data output from the file using json library. Likewise, the class *FileProcessor* reads the file and writes into the file through two functions – one for each functionality.

The libraries used in the program is shown in figure 1.1.



```
# ------------------------------------------------------------------------------------------ #
# Title: Assignment07
# Desc: This assignment demonstrates using data classes
# with structured error handling
# Change Log: Mohammad Ammar Bharmal, 3/10/2024,Created Script (2nd Attempt)
#   RRoot,1/1/2030,Created Script
#   <Your Name Here>,<Date>,<Activity>
# ------------------------------------------------------------------------------------------ #
import json
```

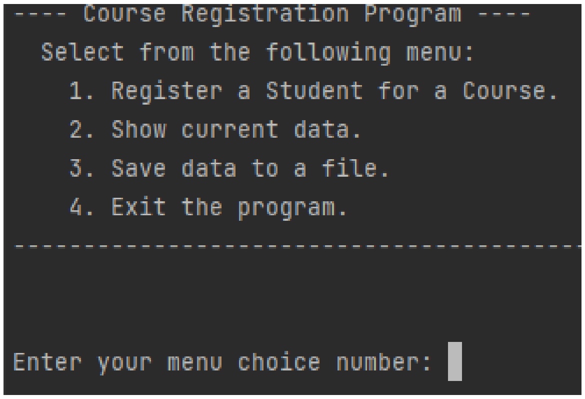
Figure 1.1: Importing library

Presenting and processing the data

When the application is run on IDE PyCharm, the user interface displays the menu for the user to choose what they want to do with the application. Whatever menu choice user inputs, it is collected as a string (string datatype for the input data), and the menu functions from class *IO* is called. Based on the user demand of student registration details into the *json* file or storing the information, the related function is called from the class *FileProcessor*. The *Enrollments.json* related is called back and forth to read or write employee information to or from it. The rest of the program is structured around the same philosophy as did in class.

Testing the Program

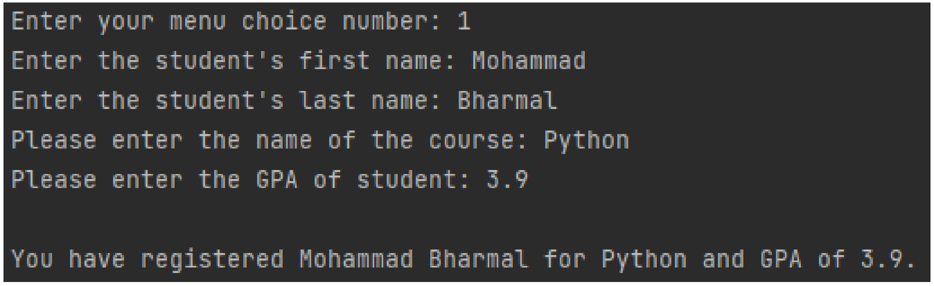
Figure 2.1 shows how the menu is displayed and information is asked by the application. When the user chooses option 2, the employee details are captured and then displayed on the screen as shown in figure 2.2. Figure 2.3 presents the confirmation of saving the data when the user chooses menu option 3.



```
---- 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: 
```

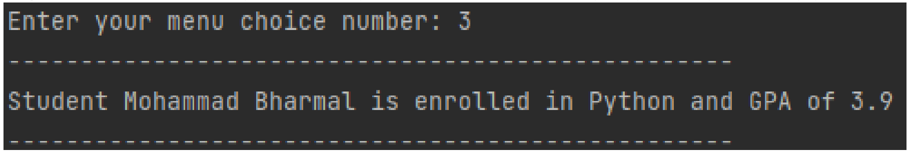
Figure 2.1: Presenting menu to the user and asking to input the desired option



```
Enter your menu choice number: 1
Enter the student's first name: Mohammad
Enter the student's last name: Bharmal
Please enter the name of the course: Python
Please enter the GPA of student: 3.9

You have registered Mohammad Bharmal for Python and GPA of 3.9.
```

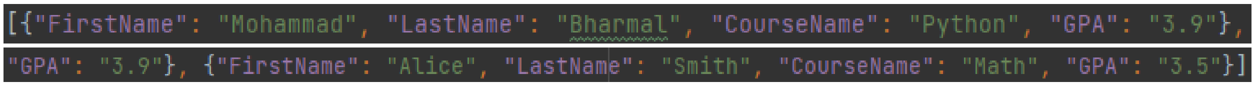
Figure 2.2: User inputs are processed and displayed by the application



```
Enter your menu choice number: 3
--------------------------------------------------
Student Mohammad Bharmal is enrolled in Python and GPA of 3.9
--------------------------------------------------
```

Figure 2.3: User inputs are saved by the application into the json file

The application validation is important so understand the credibility of the application. If two students are registered into the application and are saved using the menu choice 3, the *json* file should store that information. The application is run and validated from the *Enrollments.json* file. Figure 2.4 shows the *json* file as the final validation of the application.



```
[{"FirstName": "Mohammad", "LastName": "Bharmal", "CourseName": "Python", "GPA": "3.9"},
"GPA": "3.9"}, {"FirstName": "Alice", "LastName": "Smith", "CourseName": "Math", "GPA": "3.5"}]
```

Figure 2.4: Validation of the application through the output of the json file

Summary

The assignment 7 demonstrates the use of use the functions, classes, and using the separation of concerns pattern. Using a menu, while loop and data input, writing and saving in json file. Moreover, the opening of the program and commenting to code helps improve the longevity of the program. It makes the program look more professional, well documented, and easy to debug/reuse in future.