Kiem Hollis
3/7/2025
Foundations of Python
Assignment06
GitHubURL

Module 6

Introduction: I am creating a program that demonstrates using constants, variables, the use of functions, classes, using the separation of concerns pattern, and print statements to display a message about a student's registration for a Python course.

I copied and pasted the Script Header provided by the instructor and updated the name and current date section.

I opened up the starter file and read through the tasks in the text module.

I added import json from watching the tutorial so that I can work with the json file.

I added the two 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.
-----------------------------------------
'''
FILE_NAME: str = "Enrollments.json"
```

I added the two variables:

```
students: list = []  # a table of student data
menu_choice: str  # Hold the choice made by the user.
```

I created two classes first:

```
class FileProcessor:
class IO:
```

And proceeded to populate the methods based on what I read from the notes document:

```
class IO:
    @staticmethod
    def output_error_messages(message: str, error: Exception = None):
        """ This function displays the a custom error messages to the user

        ChangeLog: (Who, When, What)
        RRoot,1.3.2030,Created function
```

```
        KHollis, 3/7/2025, Added function

        :return: None
        """
        print(message, end="\n\n")
        if error is not None:
            print("-- Technical Error Message -- ")
            print(error, error.__doc__, type(error), sep='\n')

    @staticmethod
    def output_menu(menu: str):
        """ This function displays the menu of choices to the user

        ChangeLog: (Who, When, What)
        RRoot,1.3.2030,Created function
        KHollis, 3/7/2025, Added function

        :return: None
        """
        print()  # Adding extra space to make it look nicer.
        print(menu)
        print()  # Adding extra space to make it look nicer.

    @staticmethod
    def input_menu_choice():
        """ This function gets a menu choice from the user

        :return: string with the users choice
        """
        choice = "0"
        try:
            choice = input("Enter your menu choice number: ")
            if choice not in ("1", "2", "3", "4"):  # Note these are strings
                raise Exception("Please choose only 1, 2, 3, or 4")
        except Exception as e:
            IO.output_error_messages(e.__str__())  # Not passing e to avoid the
technical message

        return choice

    @staticmethod
    def input_student_data(student_data: list):
        """ This function gets the first name, last name, and GPA from the user

        ChangeLog: (Who, When, What)
        RRoot,1.3.2030,Created function
        KHollis, 3/7/2025, Added function

        :return: str
```

```
        """

        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:
            IO.output_error_messages("That value is not the correct type of
data!", e)
        except Exception as e:
            IO.output_error_messages("There was a non-specific error!", e)
        return student_data

    @staticmethod
    def output_student_courses(student_data: list):
        print("-" * 50)
        for student in student_data:
            print(f'Student {student["FirstName"]} '
                  f'{student["LastName"]} is enrolled in
{student["CourseName"]}')
        print("-" * 50)

    @staticmethod
    def write_data_to_file(file_name: str, student_data: list):
        # global file
        # global students

        try:
            file = open(file_name, "w")
            json.dump(student_data, file)
            file.close()
        except TypeError as e:
            IO.output_error_messages("Please check that the data is a valid JSON
format", e)
        except Exception as e:
            IO.output_error_messages("There was a non-specific error!", e)
        finally:
            if file.closed == False:
```

```
            file.close()

students = FileProcessor.read_data_from_file(FILE_NAME, students)
while (True):
    print(MENU)
    menu_choice = IO.input_menu_choice()
    if menu_choice == "1":
        IO.input_student_data(students)
        continue
    elif menu_choice == "2":
        IO.output_student_courses(students)
        continue
    elif menu_choice == "3":
        IO.write_data_to_file(FILE_NAME, students)
        print("The following data was saved to file!")
        IO.output_student_courses(students)
        continue
    # Stop the loop
    elif menu_choice == "4":
        break  # out of the loop

print("Program Ended")
```

In summation, I created a program that demonstrated using constants, variables, the use of functions, classes, using the separation of concerns pattern, and print statements to display a message about a student's registration for a Python course.