Jon Bennefeld
May 22, 2024
Foundations of Programming: Python
IT FDN 110 A
Assignment06
bentleybear17/IntroToProg-Python-Mod06 (github.com)

Working with functions using a Python script

Introduction

For this assignment, I wrote a Python script that reads from a JSON file, collects data from the user, and writes the collected data back to the JSON file. This week's script introduces functions and classes. It is the first Python script I have written that makes use of functions for organizing and simplifying code.

Starting from an existing script

As the scripts for this course have gotten longer, with more features and techniques used, I have been copying, recycling, and editing pieces of code from earlier assignments and from the labs. For this assignment, I did this in a more complete way. The note at the end of the assignment document says, "The process and code needed to complete this assignment task is very similar to Modul06-Lab03!" I started my assignment script by directly copying the code from that lab, and proceeded to edit it for the assignment. As I went through the code and edited it for the assignment, I edited the change logs to list my name, and that I created the script. This seemed reasonable to me, as I had created a new file and populated it. However, I thought about how I had not created something so much as completely borrowed it. I don't have enough programming experience to know if there is a point at which code is copyrighted or needs to be cited.

Reading and writing to an existing JSON file

The Python script, as its first step, reads the data already stored in the student information JSON file, into a list of dictionary rows. The script handles the possibility that there is not a pre-existing file with the correct name using a try-except construct. If the correct file does not exist, or if the data in the file is not in a form the script expects, the script gives an error message but keeps running. As the script runs, if the user chooses option 3 , the script writes the user-entered student information to the JSON file. Any data that is in the file when the script starts is read in to the program data, and then re-written (overwritten) in the file if the write-to-file menu option is chosen. The script will give an error message if there is an issue writing to the file, and then continue running. It will close the file whether or not it is able to write data to the file.

Collecting data from the user and handling input exceptions

The script presents a menu to the user, in the same way as the last several assignments. Menu option 1 prompts the user for student first name, last name, and course name. If the first or last name information entered included any non-alpha characters, the script displays an error message, does not save that set

of student information, and continues. The script also handles if the user enters an invalid menu option: it prints an error message and re-prints the menu and asks again for a menu choice.

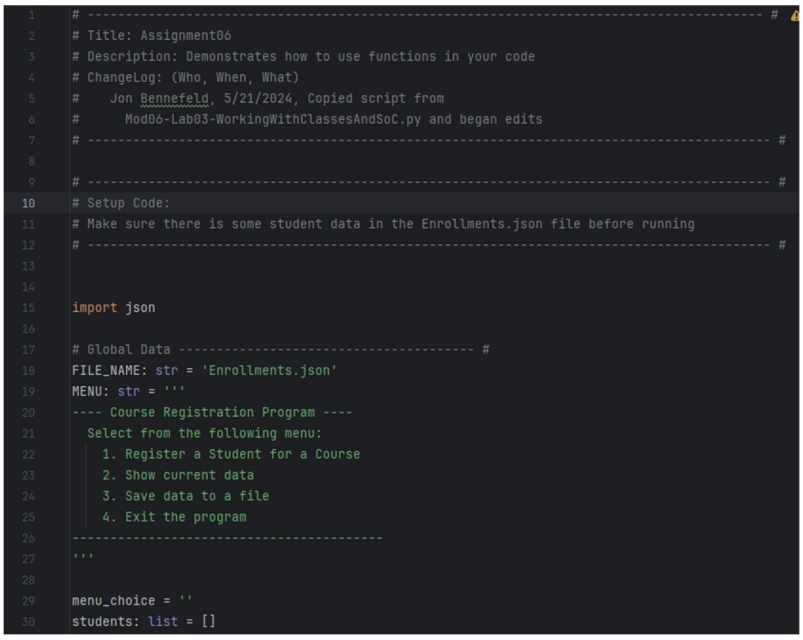
Functions in Python

In my limited past programming experience, I have learned about and worked with functions, and I understand the efficiency and reusability they provide. I am also comfortable with parameters, global and local variables, and returned values. The idea of classes is entirely new to me. I have to say, I read the "Classes and Functions" section in this week's Notes document multiple times, and I didn't grasp its purpose fully, and I feel it must be useful for larger projects (as an update, the classes and objects information did connect with me more as I started to read the notes for Week 7). Also, the "@staticmethod" decorator seems like an unusual concept for small programs, but appears to foreshadow interesting knowledge about non-static functions to come.

I learned that the convention for Python script layout is to have the functions created above the main body of the code, since the script runs from top to bottom and the functions need to be declared before called, and for the purpose of clean organization. Also, I found it interesting that if you use the Python help function with a function name, it will return the doc string for the function.

Code Excerpts

As the scripts now getting longer than 200 lines, I will be including excerpts of code here rather than the full scripts. Figure 1 shows excerpts of myPython script for this week's assignment.



```
# ------------------------------------------------------------------------------------------ #
# Title: Assignment06
# Description: Demonstrates how to use functions in your code
# ChangeLog: (Who, When, What)
#    Jon Bennefeld, 5/21/2024, Copied script from
#      Mod06-Lab03-WorkingWithClassesAndSoC.py and began edits
# ------------------------------------------------------------------------------------------ #

# ------------------------------------------------------------------------------------------ #
# Setup Code:
# Make sure there is some student data in the Enrollments.json file before running
# ------------------------------------------------------------------------------------------ #


import json

# Global Data ------------------------------------- #
FILE_NAME: str = 'Enrollments.json'
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
-------------------------------------
'''

menu_choice = ''
students: list = []
```

```python
# Processing ------------------------------------- #
class FileProcessor:
    """
    A collection of processing layer functions that work with Json files

    ChangeLog: (Who, When, What)
    Jon Bennefeld, 5/21/2024, Created Class
    """

    @staticmethod
    def read_data_from_file(file_name: str, student_data: list):
        try:
            file = open(file_name, "r")
            student_data = json.load(file)
            file.close()
        except FileNotFoundError as e:
            IO.output_error_messages( message: "Text file must exist before running this script!", e)
        except Exception as e:
            IO.output_error_messages( message: "There was a non-specific error!", e)
        finally:
            if file.closed == False:
                file.close()
        return student_data

    @staticmethod
    def write_data_to_file(file_name: str, student_data: list):

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

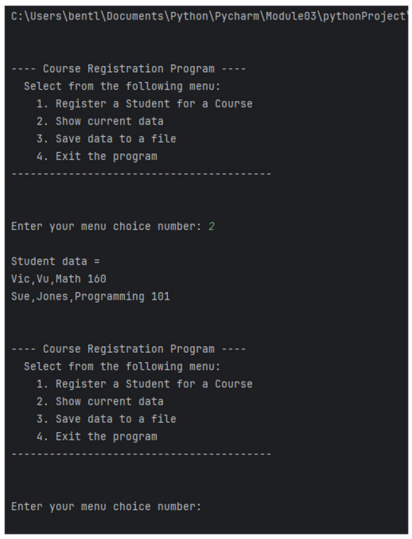
```python
    @staticmethod
    def input_student_data(student_data: list):
        """ This function gets the first name, last name, and course name from the user

        ChangeLog: (Who, When, What)
           Jon Bennefeld, 5/21/2024, Created function

        :return: None
        """

        try:
            # input the data
            # if the student's first or last name does not contain all letters, raise an error
            student_first_name = input("What is the student's first name? ")
            if not student_first_name.isalpha():
                raise ValueError("The first name should contain letters only.")

            student_last_name = input("What is the student's last name? ")
            if not student_last_name.isalpha():
                raise ValueError("The last name should contain letters only.")

            course_name = input("What is the student's course? ")

            student = {"FirstName": student_first_name,
                       "LastName": student_last_name,
                       "CourseName": course_name}
            student_data.append(student)

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

# End of function definitions
```

```python
# Beginning of the main body of this script
students = FileProcessor.read_data_from_file(file_name=FILE_NAME, student_data=students)

# Repeat the follow tasks
while True:
    IO.output_menu(menu=MENU)

    menu_choice = IO.input_menu_choice()

    if menu_choice == "1":  # get new data
        students = IO.input_student_data(student_data=students)
        continue

    elif menu_choice == "2":  # display current data (including what was read from the file upon start)
        IO.output_student_courses(student_data=students)
        continue

    elif menu_choice == "3":  # save data to the file, and also display what was written to the file
        FileProcessor.write_data_to_file(file_name=FILE_NAME, student_data=students)
        print("The following data is written to the file:")
        IO.output_student_courses(student_data=students)
        continue

    elif menu_choice == "4":  # end the program
        break  # out of the while loop

```

Figure 1: Python script excerpts

Testing script in Pycharm

The assignment requires that I successfully run my script in both Pycharm and from the command shell. Figure 2 shows the successful run of my program in Pycharm, with each menu selection chosen at least once, and some of the error handling code tested.



```
Enter your menu choice number: 1
What is the student's first name? Ann
What is the student's last name? Smith44
That value is not the correct type of data!

-- Technical Error Message --
The last name should contain letters only.
Inappropriate argument value (of correct type).
<class 'ValueError'>


---- 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
What is the student's first name? Ann
What is the student's last name? Smith
What is the student's course? Biology 202


---- 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:
```

```
Enter your menu choice number: 3
The following data is written to the file:

Student data =
Vic,Vu,Math 160
Sue,Jones,Programming 101
Ann,Smith,Biology 202


---- 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: 5
Please, choose only 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: 4

Process finished with exit code 0
```

Figure 2: Script run in Pycharm

Testing script the command shell

Figure 3 shows the successful run of my program in the command shell.

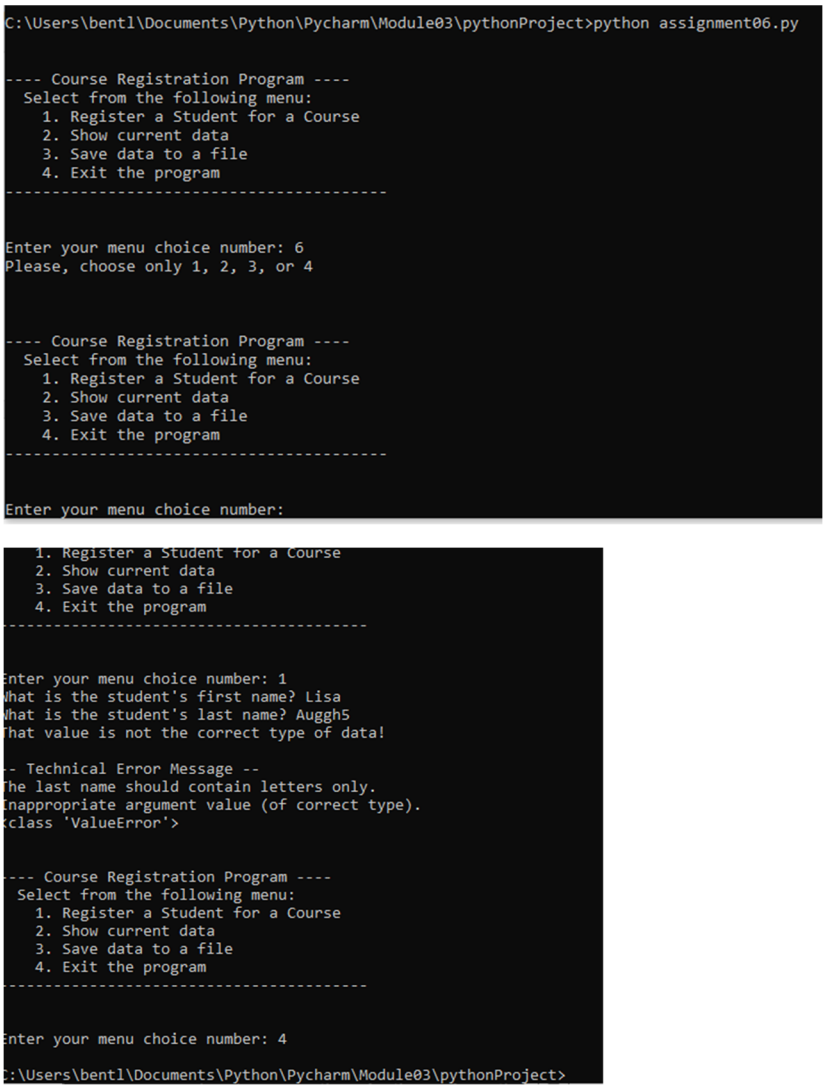


Figure 3: Script run in Command Shell

Output file

Figure 4 shows that my script wrote student information into a JSON file.

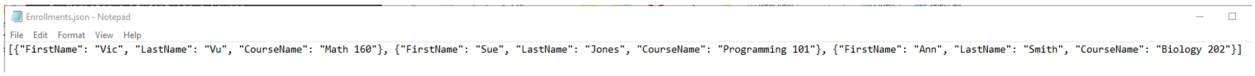


Figure 4: Output file opened in Notepad

Summary

This assignment introduced me to using functions in Python, and included a mix of ideas I already knew and ideas that were fully new to me. I learned about the modularity and efficiency that functions provide, and I learned about grouping functions into classes. I learned the first part of what objects are and expect to learn much more in the coming weeks.