Noah Cooper
9/4/24
IT FDN 110 B Su 24: Foundations Of Programming: Python
Assignment 06
https://github.com/DrBones2/IntroToProg-Python

Assignment 06 - Functions

Intro

In this module, we learned about functions. Functions are a fundamental aspect to python programming, as they allow for modular code to be called multiple times in the same script, which can massively cut down the complexity and repetitiveness of scripts with recurring concepts.

Learning Fundamentals

Functions are modular, reusable blocks of code. When a script needs to perform a similar task under multiple different conditions, a function can be used to perform the task without needing to re-write the code in multiple places. The script only needs the function to be written once, usually near the top of the script for organizational purposes. After that, the function can be called later in the script.

Functions can store local variables, which are stored differently than global variables. The variables defined outside of a function can be “shadowed” by local variables. This means the function will have the local variable read the global variable, but still allow the local variable to be changed without affecting the global variable.

Parameters are defined along with functions. By declaring parameters, we can provide data to the function as we call it. This allows us to encapsulate the data, and not make it rely on global data, which may otherwise cause bugs. It also provides increased flexibility, allowing us to utilize different variables to reference that are more useful in certain contexts.

Classes are a useful tool to group functions together. By having multiple functions defined under a single class, we provide organization and modularity to our functions. Collaborators on our files will have an easier time identifying functions intended to be used for the purpose signified by its class.

Lastly, the module covered the concept of “Separation of Concerns”, a design principle that aims to allow developers to sort their issues with a software into its relevant field. An issue with presentation should not need to deal with issues related to logic or data storage, for example. An organized and well-documented code base prevents developers from potentially getting bogged down with unrelated aspects.

Assignment

This assignment, while based off of the previous one, had us shaping our scripts in a significantly different way. The code remained similar, but this time we now have functions defined, which allows our code to become modular and succinct.

Each portion of code has now been given its own function. This allows us to define all of our necessary code at the top of the script one time. Even though most functions are only called once throughout the script, it has the benefit of being better future-proofed, allowing us to develop on it without worrying about reusing code.

One tangible benefit we take advantage of is exception handling. Since we check for exceptions in various parts of our code, it allows us to run a function that takes the exception as a local variable, and print out a more detailed error statement. Previously, this code had to be nested into each aspect we wanted to have exception handling for, so we have shortened the necessary amount of code required compared to last assignment.

The script now utilizes the json library, which allows us to import and export our data more easily. Rather than needing to account for the list of dictionaries we have set up and formatting accordingly, the json library provides us with a function to read the data and store it into a list. We can also append our user-input data and write it back into the json file with no problems.

The body of the code is much shorter, with each menu option now calling the functions we defined earlier.

```
================================================================================
===== RESTART: C:\Users\Dr. Bones\Desktop\_Module06\assignment06.py ============
================================================================================
=== 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 program
Select an option: 2
Your data:
Student Bob Smith is enrolled in Python 100
Student Sue Jones is enrolled in Python 100
=== 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 program
Select an option: 1

Enter student's first name: Nathan
Enter student's last name: Steele
Enter course name: Economics 100
Data accepted! You have registered Nathan Steele for Economics 100.
Please remember to save to 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 program
Select an option: 3
Data saved successfully!
Your data:
Student Bob Smith is enrolled in Python 100
Student Sue Jones is enrolled in Python 100
Student Nathan Steele is enrolled in Economics 100
```

Figure 1: The script running in the IDLE terminal. We can read the data & submit our own entry.

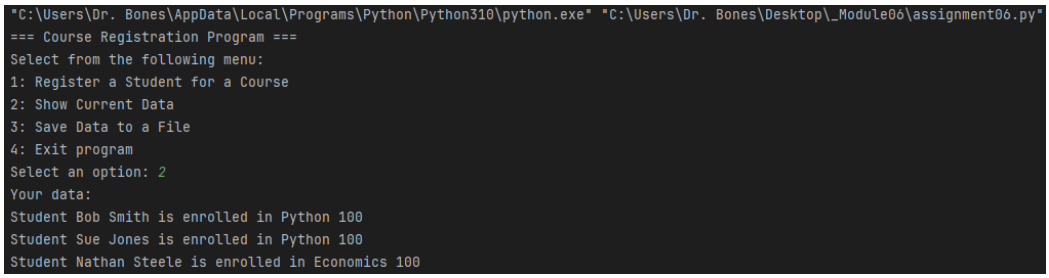

Figure 2: The script running in PyCharm.

Conclusion

This assignment has taught us how to utilize functions, which is a major aspect of creating concise and maintainable python scripts.

This week proved to be quite challenging for me personally, but I feel confident moving forward. For the assignment, I had spent too long hung up on a bug with a confusing error message, only to find it was caused by a missing return statement in one of my functions. It gave me perspective on the necessity for patience and rationality while coding. Being able to stay level headed while coding will allow me to slow down and realize simple errors more effectively, and provide me with more mental stamina for my projects.