

Christian Todhunter

05/25/2020

IT FDN 100 A

Assignment06

FUNCTIONS!!!!

INTRODUCTION TO MODULE 05

Module 6 introduced us to functions, how to use them, why they are nice, and how they can help keep code organized. From returning values to re-using your code in later assignments, functions are an invaluable feature in any programming language.

THE ASSIGNMENT REQUIREMENTS

The follow figure outlines the task at hand from R. Roots "Assignment05" document:

Step 5 - Apply your knowledge

Now that you have reviewed the websites and videos, **modify** a script that manages a "ToDo list." This project is like the last one, but different enough to be a challenge.

I have provided a starting template to I want you to modify and use for your program called "Assignment06_Starter.py". Currently the code loads data from a file into a Python List of Dictionary objects. However, the code only uses a few functions, and **your job is to add more functions to organize the code**. You may note that it is both easier and harder to work with someone else's code, and that is part of the assignment.


- Create** a new sub-folder called Assignment06 inside the _PythonClass folder.
- Create** a new project in PyCharm that uses the _PythonClass\Assignment06 folder as its location
- Add** the starter file, "Assignment06_Starter.py," to your project.
- Add** code to your script that performs the assignment's task. Don't forget to update the changelog in the script's header.
- Run** the script in BOTH PyCharm and an OS command/shell window and capture images of it working on your computer.
- Verify** that it worked, by locating the text file and opening it in a text editor. The file should be in the same folder as your script if you used the correct, **relative** file path!

Figure 1: Screenshot from *Assignment06* by R. Root

In addition to the above, we were also tasked with uploading THIS DOCUMENT to GitHub, along with the corresponding python file.

MY APPROACH

This week I watched the videos, read the text, visited the websites and watched the additional external video explaining functions in a different manner. I also watched some of the office hours video with Mr. Root because I was struggling on an issue. My issue was properly unpacking the tuple via the code below:



```
# Step 4 - Process user's menu choice
if strChoice.strip() == '1': # Add a new Task
    # TODO: Add Code Here
    strTask, strPriority = IO.input_new_task_and_priority()

    Processor.add_data_to_list(strTask, strPriority, lstTable)

    IO.input_press_to_continue(strStatus)
    continue # to show the menu
```

Figure 2: The Troublesome Tuple

I did end up phoning a friend for some help because I spend a lot of time on it and wasn't getting anywhere. As it turns out I was misunderstanding how strTask and strPriority were populated from the function input_new_task_and_priority(). I was under the impression that the proper way to populate the variables was by entering them as the parameter for the function, then the function would take care of it automatically, see below for how I was trying to write the code prior to fixing the issue:

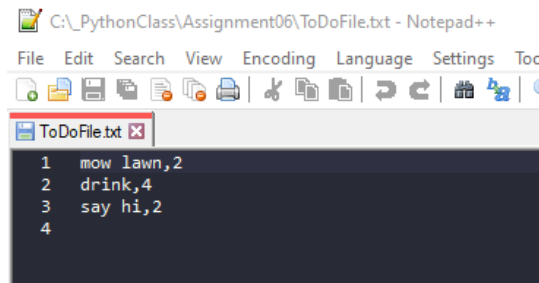
```
IO.input_new_task_and_priority(strTask, strPriority)
```

After the above line in my program, I thought strTask and strPriority would be assigned the input() values from within the function, and I could then use them in the following Processor.add_data_to_list() function. I knew this was where the issue was because I was printing the lstTable and list_of_rows and noticed it wasn't being filled properly. The main issue is the difference between mutable and immutable types, and how the function handles local and global variables. The input_new_task_and_priority() function returns the task and priority, but because these were from a tuple (which is not mutable), I was unable to directly assign the new values outside of the function, without adding and unpacking the tuple within the function via strTask, strPriority being assigned to the function. This is something I plan to have expanded on in the next office hours session if I can't sort it out after reviewing the notes, videos, and text again.

After that single line, I was able to sort the rest out via plugging in the right functions and assigning or 'tying' together the correct parameters from IO function to Processor function. The rest of the code was straight forward and I was able to complete the run the program in both the command line and PyCharm successfully.

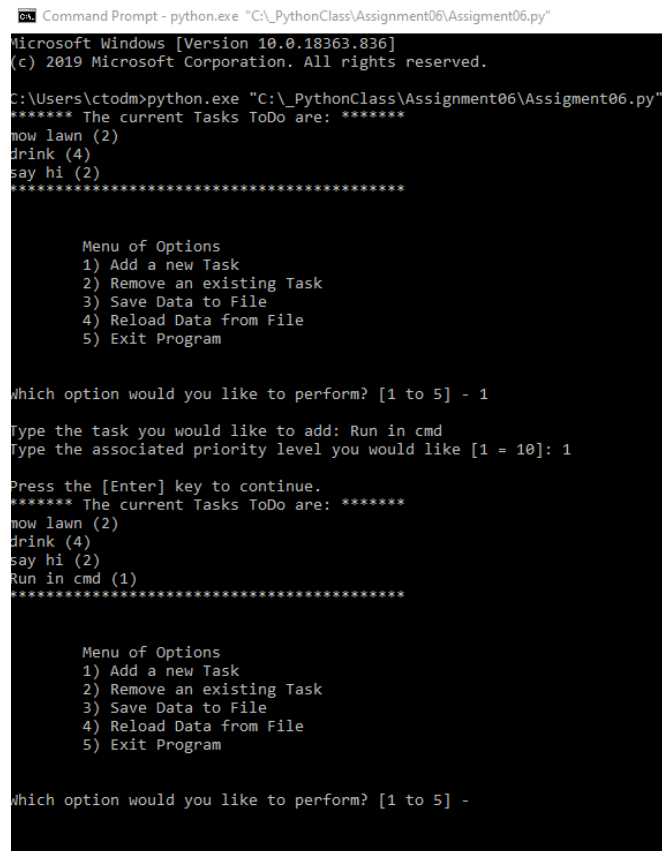
THE SOLUTION ARRIVES!

Please see below the text file output from the program in Figure 4, the program running in Command-Line via Figure 5, and the program running in PyCharm via Figure 6:



```
C:\_PythonClass\Assignment06\ToDoFile.txt - Notepad++
File Edit Search View Encoding Language Settings Too
ToDoFile.txt
1 mow lawn,2
2 drink,4
3 say hi,2
4
```

Figure 3: Assignment05.py output .txt file via Notepad++



```
Command Prompt - python.exe "C:\_PythonClass\Assignment06\Assignment06.py"
Microsoft Windows [Version 10.0.18363.836]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\ctodm>python.exe "C:\_PythonClass\Assignment06\Assignment06.py"
***** The current Tasks ToDo are: *****
mow lawn (2)
drink (4)
say hi (2)
*****

Menu of Options
1) Add a new Task
2) Remove an existing Task
3) Save Data to File
4) Reload Data from File
5) Exit Program

Which option would you like to perform? [1 to 5] - 1

Type the task you would like to add: Run in cmd
Type the associated priority level you would like [1 = 10]: 1

Press the [Enter] key to continue.
***** The current Tasks ToDo are: *****
mow lawn (2)
drink (4)
say hi (2)
Run in cmd (1)
*****

Menu of Options
1) Add a new Task
2) Remove an existing Task
3) Save Data to File
4) Reload Data from File
5) Exit Program

Which option would you like to perform? [1 to 5] -
```

Figure 4: Command-line running Assignment05.py (larger image available in submission)

```
C:\Python38\python.exe C:/_PythonClass/Assignment06/Assignment06.py
***** The current Tasks ToDo are: *****
mow lawn (2)
drink (4)
*****

Menu of Options
1) Add a new Task
2) Remove an existing Task
3) Save Data to File
4) Reload Data from File
5) Exit Program

Which option would you like to perform? [1 to 5] - 1

Type the task you would like to add: say hi
Type the associated priority level you would like [1 = 10]: 2

Press the [Enter] key to continue.
***** The current Tasks ToDo are: *****
mow lawn (2)
drink (4)
say hi (2)
*****
```

Figure 5: PyCharm running Assignment06.py

SUMMARY

This assignment was very similar to Assignment05, my main issue was pulling the new task and priority data from the function and storing it so I could add it to the processor. This class continues to challenge me, and I will be reviewing the material from this module prior to jumping into the next one.

Link to my Github: <https://github.com/CtodGit/IntroToProg-Python-Mod06>