Josh McMillen

11-24-2020

IT FDN 110 A

Assignment 06

https://github.com/mcmillj2/IntroToProg-Python

Foundations Of Programming: Python

Performing Assignment 06

**Introduction**

This document lays out how to perform assignment 6 in the Foundations of Program: Python course at the University of Washington. In order to complete this assignment I used PyCharm to update a script to use functions that gives the user the options adding a new item to the a to do list, removing an item from the list, saving back to the file, reloading the file and lastly closing the program.

**Creating Third Python Script**

Since PyCharm was already setup on my machine I simply created a new project in a Assignment06 folder on my C drive. This assignment was different where I started with a “starter” script that needed editing. The main task was to: “*Currently the code loadsdata from a file intoa Python List of Dictionaryobjects.However, the code only uses a few functions, and your job is to add more functions toorganize the code.”.* To accomplish this I reviewed the script and started at the top.

**Script header, data section and overall cleanup**

A script header already existed, but simply needed to have my name/date/desc added to record the change (Figure1). Next I reviewed the data section, there appeared to be several variables that were not yet used, but overall no issues were apparent other than a slight update to the text file name. Ultimately I ended up returning to this section and removing several variables I choose not to use in the script. Also there were a few immediate minor errors in the script around using all lower case for variable and function names which I updated.

**Processing section**

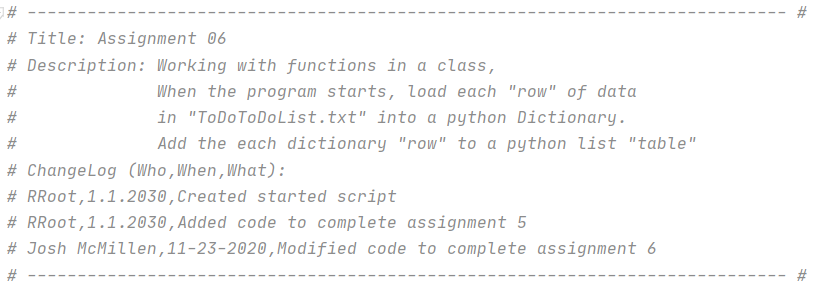
Since the main loop at the bottom of the script was functioning (or so to say the whole script worked as-is), I choose to begin updating the processing section with new functions. For example there was a “write\_data\_to\_file” function that was empty. For this function I used the logic I created in assignment two (see figure 2). The main difference here is that the function collects the file name (text file) and data table (list of rows) during the main loop. The names of the passed variables are not used in the function, but their “place names” are in the function definition (for example “file\_name” collects whatever file name, but not specifically strFileName in this situation. Then file\_name is used within the function instead of the actual file name variable like in assignment 5. The other difference is to close out the function a “return” must be used. In this case there is nothing to be passed back to the main user loop since all the processing already took place. Finally tool tips are added below the function definition to help remember how to use the function later. This function is part of the “processing” class, so it is referenced as processing. write\_data\_to\_file(file\_name, list\_of\_rows).

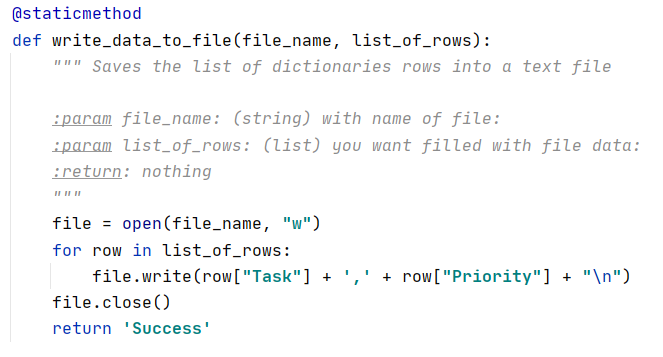
**Presentation section**

Next the presentation section was finalized. For example there is a function called “input\_task\_to\_remove” (see figure 3). The bulk of this function was borrowed from assignment 5 as well, which uses a for loop and another remove row function. The main variable passed in the function is “list of rows” which takes any table and cycles through each row and asks the user if they want to delete it. In this case the main table/list is called “lstTable” but should not be used within the function to make the function reusable. This function is also different where it includes references to other functions, I am not sure if this is correct, however it did simplify the coding.

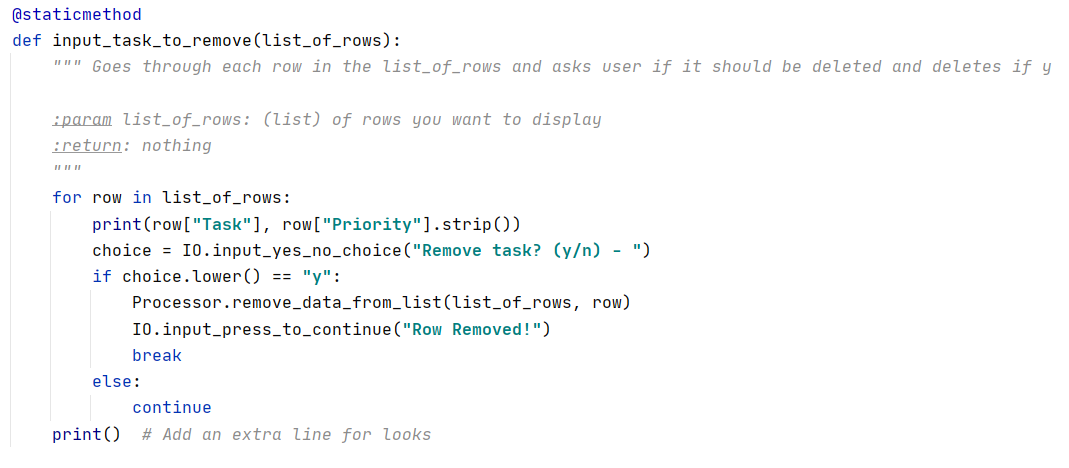
**Main body section and closing**

Finally the main body section of the script was mostly complete as-is. The additions where were to reference the processing and presentation functions as much as possible to make the logic simple in this area. For example the remove task portion is simply an else if checking if option 2 was selected, then referencing the presenting function “input task to remove” which does all the work. Then continue moves the user back to the main menu. The main body allows for the user to add new task items, remove task items, save to the text file, reload from the text file and close the program.

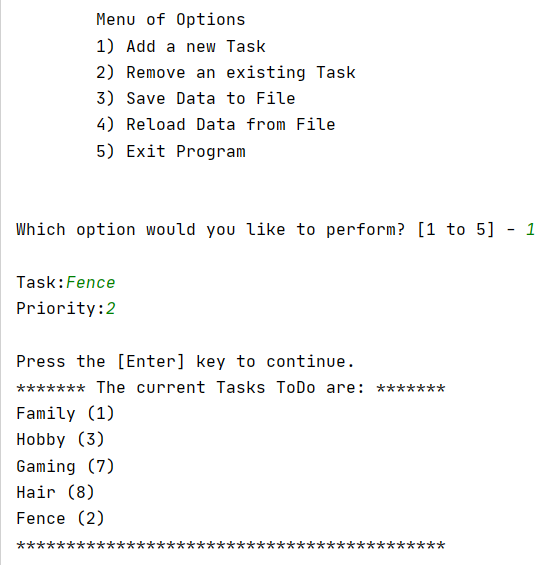
**Figure 1: Header**

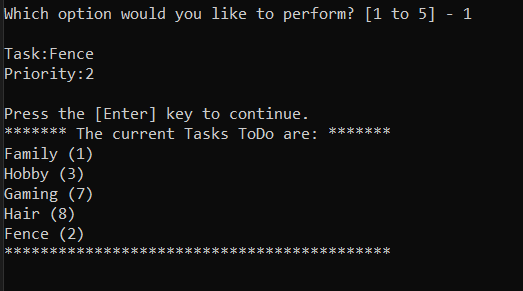


**Figure 2: Write data to text file processing function example**

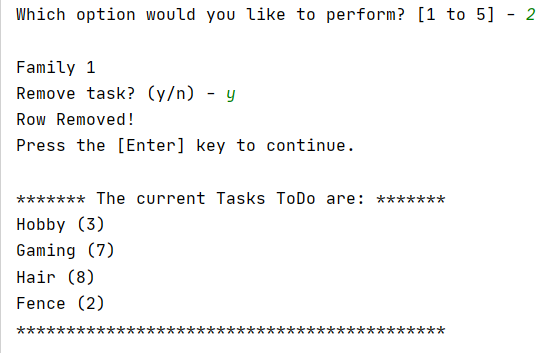


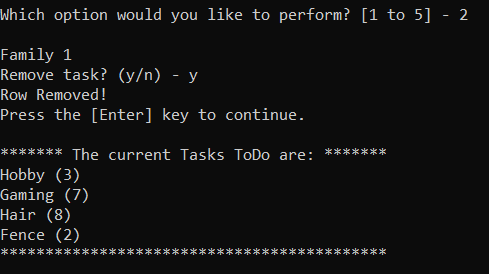
**Figure 3: Removing a task presentation function example**



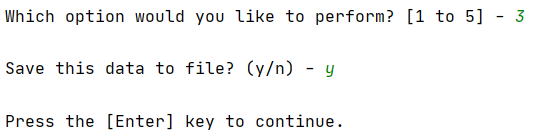


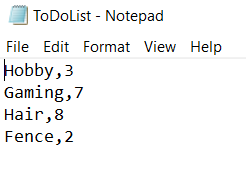
**Figure 4: Adding a task in PyCharm and Command Line**



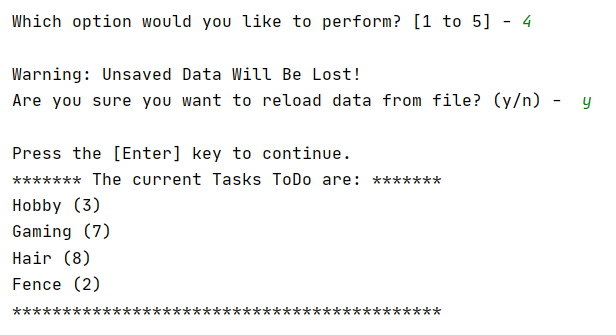


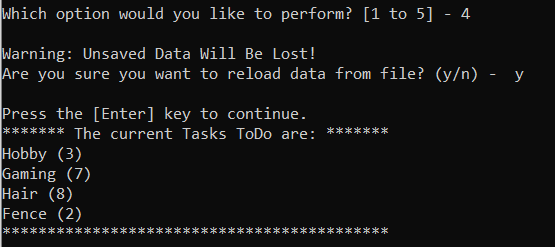
**Figure 5: Deleting a task in PyCharm and Command Line**





**Figure 6: Saving the file in PyCharm and Text Output**





**Figure 7: Reloading the file in PyCharm and Command Line**

**Summary**

To fulfil the requirements of assignment 6 I modified the starter script. The main changes made where to add in several functions to the processing and presentation sections. These functions were then referenced in the main body (loop) for the user. The functions enable the user to add new tasks, remove tasks, save to a text file, reload from the text file and close the program.