Samuel Nunn

May 16, 2020

Foundations of Programming: Python

Assignment 05

<https://github.com/stnunn/IntroToProg-Python>

Working with Dictionaries and Files

# Introduction and Assumptions

In this assignment, we will read/load “Task” and “Priority” values from a text file, and store them as Dictionary “rows” within a List “table” object. Then we will give the user five options: (1) Show current data, (2) Add a new item, (3) Remove an existing item, (4) Save data to a file, and (5) Exit program.

# Instructions

1. Start a new PyCharm project called “Assignment05” within the C:\\_PythonClass\ directory. Copy the Assignment05\_Starter.py file into this directory. Modify the Header section with your information, as seen in Figure 1.

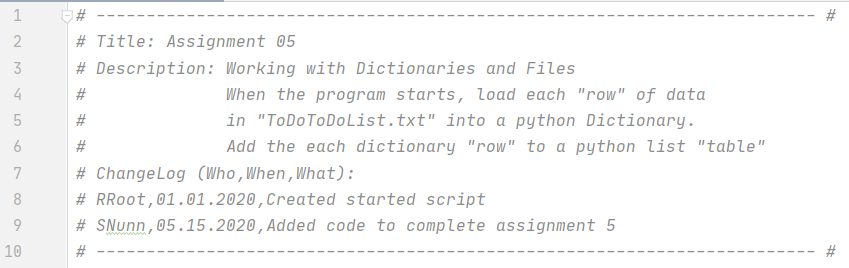


Figure 1. Script Title and Header Section

1. In the “Data” section, declare your variables and constants, as seen in Figure 2.

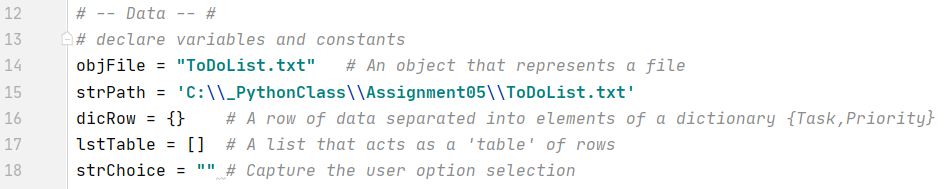


Figure 2. – Data. Declare variables and constants.

1. In the “Processing” section, open the file and use a “for” loop to iterate through each row of the file, using the split() function to split based on “,”. For each iteration, create a new dicRow, and append that dicRow to the lstTable, as seen in Figure 3.

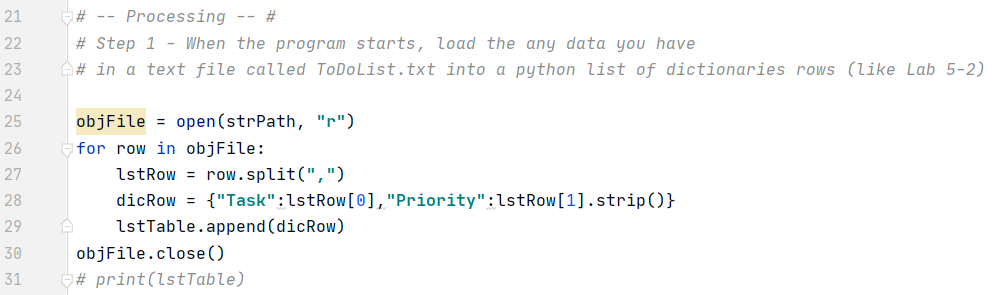


Figure 3. Process the Data

1. In the Input/Output section, the user can choose what they would like to do. If they choose (1), we loop through and unpack the dictionaries in the list, and print the results, separated by a “|”. If the user chooses (2), we append a new dictionary to the list after asking the user what they would like to add, as seen in Figure 4.

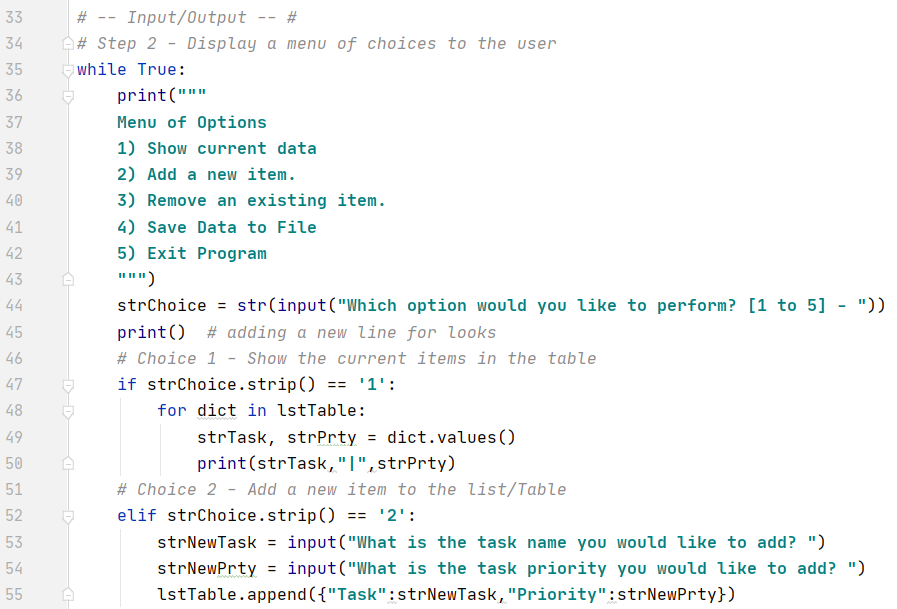


Figure 4. Choices (1) and (2).

1. If the user chooses (3), we prompt for which “Task” they would like to remove. We then loop through the lstTable and check if the dictionary value for key of ”Task” is == the “Task” the user would like to be removed. If so, we use the “remove” method to remove the dictionary from the list. We also use a numRemoved counter to keep track of how many instances of the “Task” were removed, in case there were more than one, as seen in Figure 5.

If the user chooses (4), we use the “w” argument when opening the .txt file to overwrite the pre-existing data. We use another “for” loop to unpack the dictionary values and write them to the file, separated by “,”, as seen in Figure 5.

Lastly, if the user chooses (5), we “break” to exit the loop, and ask the user to “Press Enter key to exit the program., as seen in Figure 5.

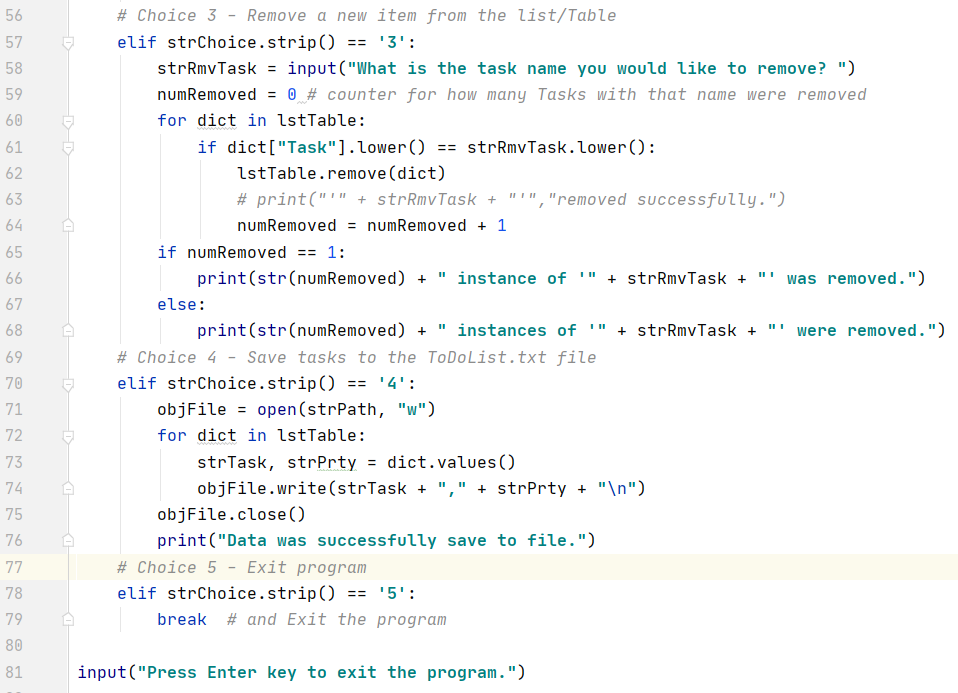


Figure . Handling Options (3), (4), and (5)

1. Run the program in the command prompt, as seen in Figures 6 – 9, below.

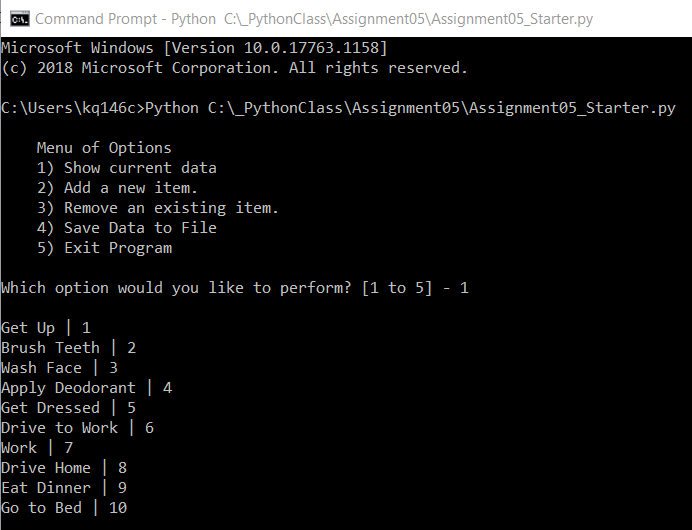


Figure - Option 1

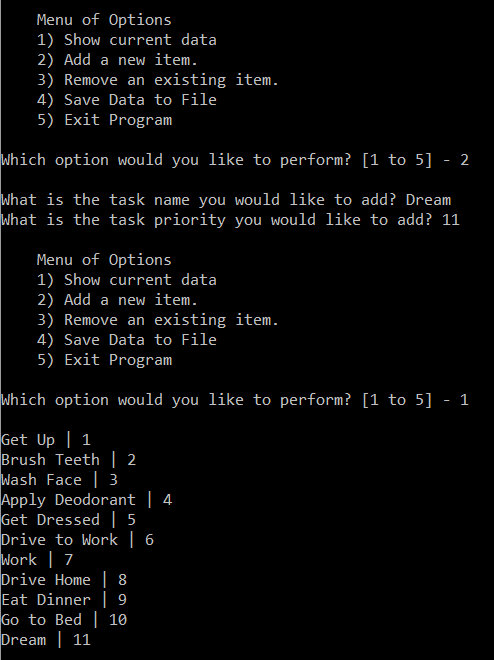


Figure - Option 2

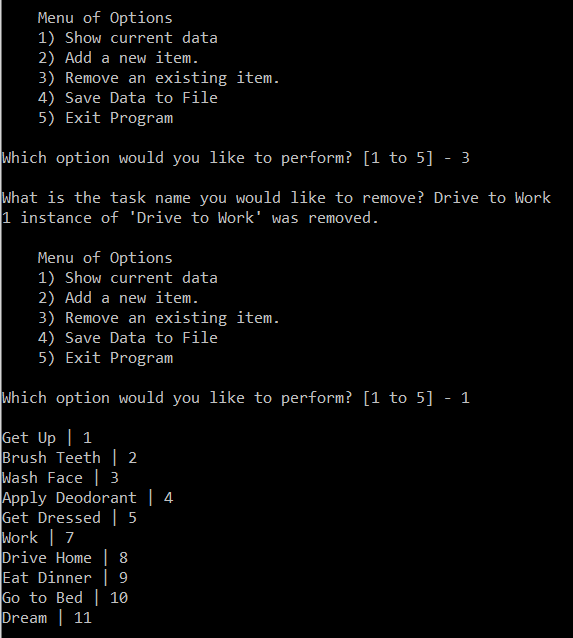


Figure - Option 3

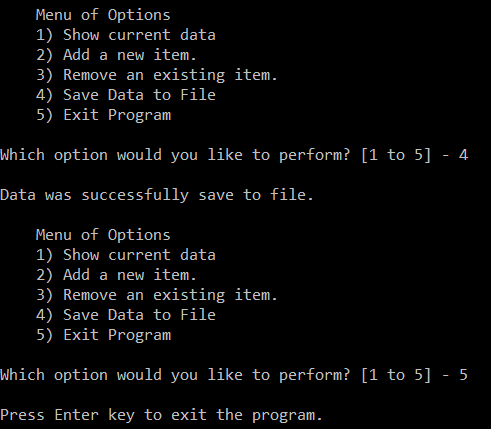


Figure – Option 4 and Option 5

1. Open the .txt file to see the Tasks and Priorities which were changed by the user, as seen in Figure 10. Note that “Drive to Work” was removed, while “Dream” was added.

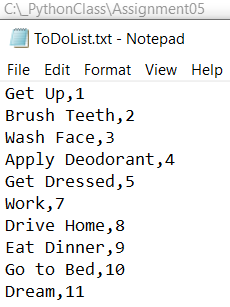


Figure . Open HomeInventory.txt to see the saved output

# Summary

In this assignment, we learned how to read/load “Task” and “Priority” values from a text file, and store them as Dictionary “rows” within a List “table” object. Then we learned how to give the user five options: (1) Show current data, (2) Add a new item, (3) Remove an existing item, (4) Save data to a file, and (5) Exit program.

We utilized “for” loops, unpacking, and conditional logic to accomplish the list and dictionary modifications. Lastly, writing the results to a .txt file allows the user to save the results to a file, so it can be reviewed at a later time.