Erin Leggett
November 16, 2021
IT FDN 110A
Module 05 - Assignment 05

EMPLOYING DICTIONARIES AND LISTS IN PYTHON

OBJECTIVE

Describe the steps taken to create the Python script attached to this assignment.

INTRODUCTION

This week we expanded upon our knowledge of collections - or sequences - of data by exploring the differences between and applications of dictionaries and lists, particularly as pertains to organizing data into tables and writing those tables to a file. We also gained experience reading a text file into memory prior to accessing, manipulating, and saving its content via a user-facing menu within the program.

FIRST PASS

Using the pseudo-code provided, I first went about executing the functional aspects of the code: reading the ToDoList.txt file into memory in order to interact with its contents and fleshing out the provided menu options. My first pass was simple, but functional: I managed to get read and write working properly, though it took a little debugging to get the file loaded into memory as a table from a .txt file. In order to do so, not only did the text file need to be *read*; the program also needed to be told how to *organize* the data it was being presented. To accomplish such, I called in a lstData variable and used that to create a dictionary, which was then used to construct a table:

```
objFile = open(strFile, "r")
for row in objFile:
    lstData = row.split(",")
    dicRow = {"Task": lstData[0].strip(), "Priority":
lstData[1].strip()}
    lstTable.append(dicRow)
objFile.close()
```

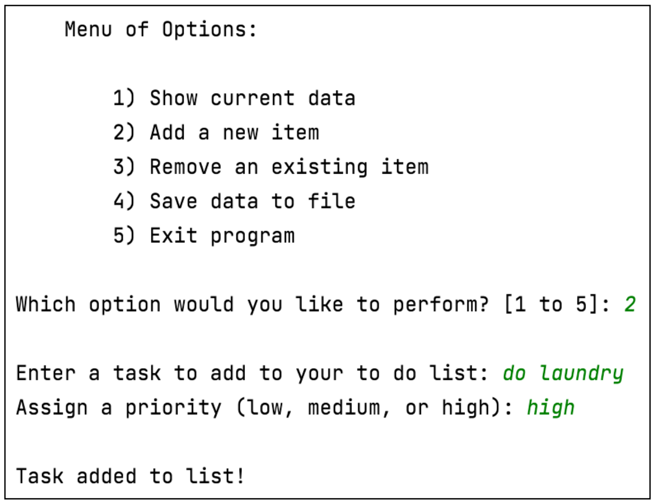
SECOND PASS

I ended up taking a second pass at my code prior to turning it in, as some elements that I had struggled to include on my own were covered in greater detail during the lecture. In particular, I wanted the user to be alerted when they tried to remove a task that did not appear on their current to do list; I hadn't though to assign a Boolean operator. After running the script a few times, I also spent some time adding line breaks for readability and finding opportunities to re-print the list for ease of use. For example, I decided to display the list twice in the following conditional: once at the front end, to show the user what could be removed, and again to confirm that the task in question had been removed as requested. Here is what that looked like:

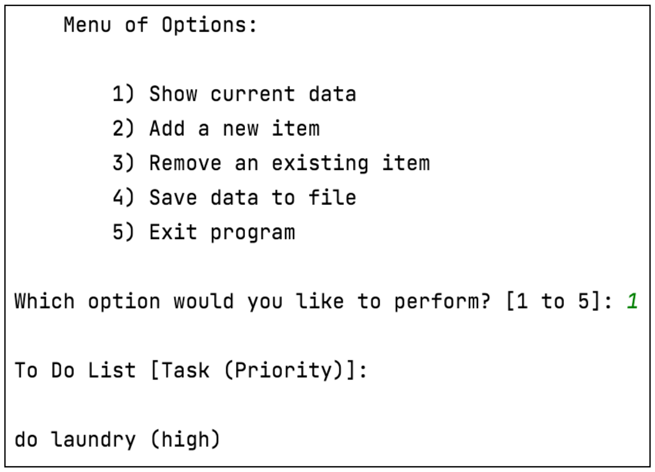
```
elif strChoice.strip() == "3":
    print("To Do List [Task (Priority)]:\n")
    for row in lstTable:
        print(row["Task"] + " (" + row["Priority"] + ")")
    strRemove = input("\nEnter a task to remove from your to do list:
")
    blnFlag = False
    intRow = 0
    for row in lstTable:
        task, priority = dict(row).values()
        if task.lower() == strRemove.lower():
            lstTable.remove(row)
            blnFlag = True
        intRow += 1
    if blnFlag == True:
        print("\nTask successfully removed from list!\n")
        print("To Do List [Task (Priority)]:\n")
        for row in lstTable:
            print(row["Task"] + " (" + row["Priority"] + ")")
    else:
        input("\nTask not found. Press Enter to return to main menu.")
    continue
```

CODE VALIDATION

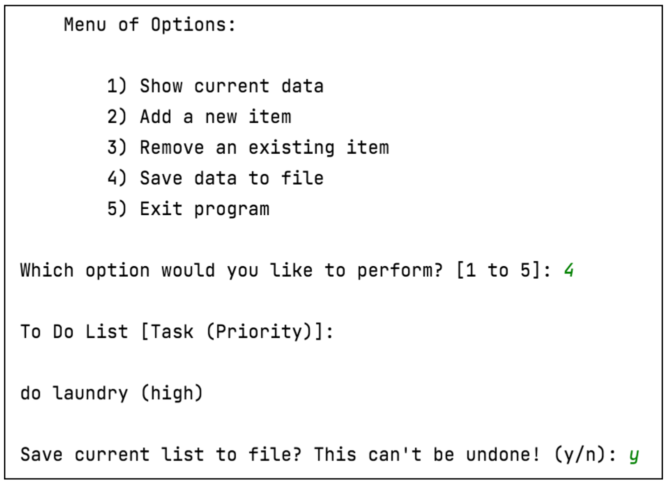
To validate my script, I ran it once in PyCharm to add, display, and write a task to my list - and then moved over to Terminal to display my list, remove the item I just added, and re-save my list, in theory producing a blank file. Here is that first sequence of events playing out in PyCharm:

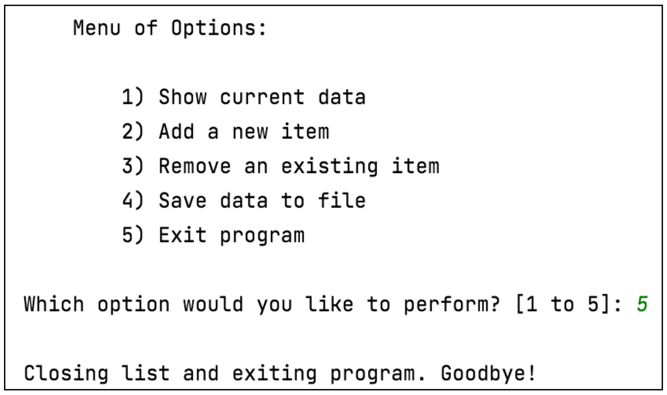


```
    Menu of Options:

        1) Show current data
        2) Add a new item
        3) Remove an existing item
        4) Save data to file
        5) Exit program

Which option would you like to perform? [1 to 5]: 2

Enter a task to add to your to do list: do laundry
Assign a priority (low, medium, or high): high

Task added to list!
```

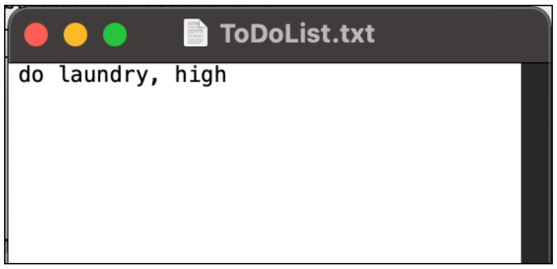


```
    Menu of Options:

        1) Show current data
        2) Add a new item
        3) Remove an existing item
        4) Save data to file
        5) Exit program

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

To Do List [Task (Priority)]:

do laundry (high)
```


```
    Menu of Options:

        1) Show current data
        2) Add a new item
        3) Remove an existing item
        4) Save data to file
        5) Exit program

Which option would you like to perform? [1 to 5]: 4

To Do List [Task (Priority)]:

do laundry (high)

Save current list to file? This can't be undone! (y/n): y
```

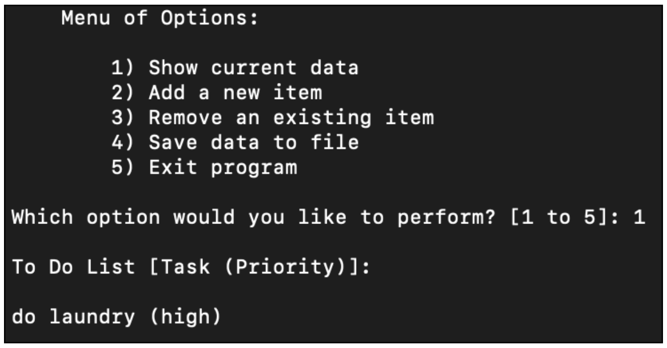

```
    Menu of Options:

        1) Show current data
        2) Add a new item
        3) Remove an existing item
        4) Save data to file
        5) Exit program

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

Closing list and exiting program. Goodbye!
```

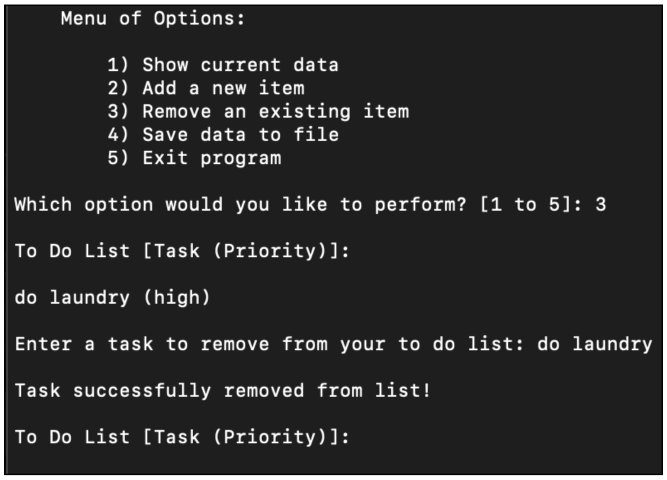
As well as the ToDoList.txt file, showing my high-priority to do list item:



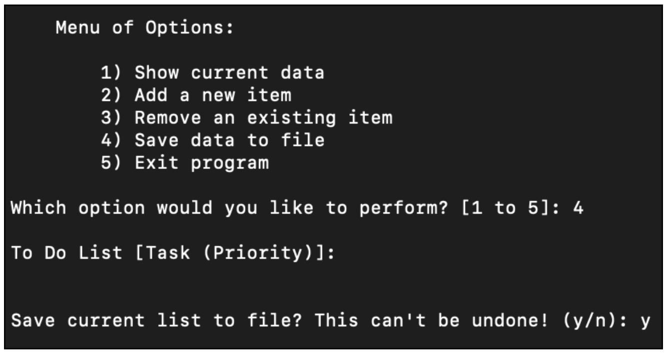
```
do laundry, high
```

I then moved over to Terminal and displayed my list, removed the task I'd just saved, and re-saved the file to verify hat my script would write the deletion back:

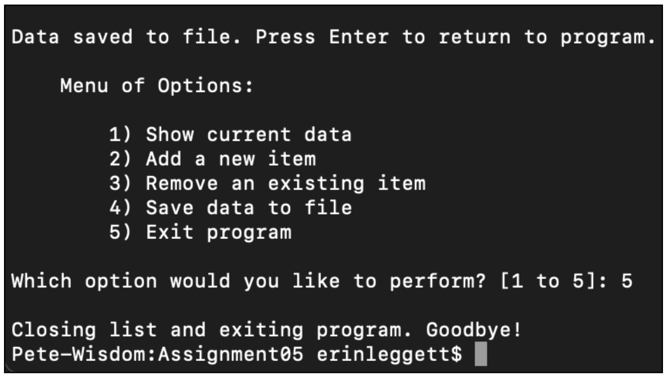

```
    Menu of Options:

        1) Show current data
        2) Add a new item
        3) Remove an existing item
        4) Save data to file
        5) Exit program

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

To Do List [Task (Priority)]:

do laundry (high)
```



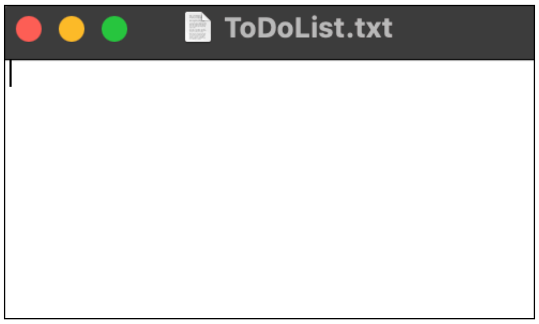
```
    Menu of Options:

        1) Show current data
        2) Add a new item
        3) Remove an existing item
        4) Save data to file
        5) Exit program

Which option would you like to perform? [1 to 5]: 3

To Do List [Task (Priority)]:

do laundry (high)

Enter a task to remove from your to do list: do laundry

Task successfully removed from list!

To Do List [Task (Priority)]:
```



```
    Menu of Options:

        1) Show current data
        2) Add a new item
        3) Remove an existing item
        4) Save data to file
        5) Exit program

Which option would you like to perform? [1 to 5]: 4

To Do List [Task (Priority)]:


Save current list to file? This can't be undone! (y/n): y
```



```
Data saved to file. Press Enter to return to program.

    Menu of Options:

        1) Show current data
        2) Add a new item
        3) Remove an existing item
        4) Save data to file
        5) Exit program

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

Closing list and exiting program. Goodbye!
Pete-Wisdom:Assignment05 erinleggett$
```



LESSONS LEARNED

Unlike last week, I had issues getting PyCharm and Terminal to create the companion .txt file on their own while running the script; instead, I had to add an appropriately-named .txt file to my project folder. As yet, I haven't been able to figure out where that configuration lies. However, I was able to work out a different roadblock I'd been hitting with Terminal: instead of entering the Python environment immediately, I pointed the directory to my _PythonClass/Assignment05 folder, which allowed me to run my script without having to paste the full file path in.

I also encountered a few features that I wanted (and tried) to add to my script file, but at this point didn't have the ability to fully realize. For example, I tried to limit the user to inputting only a small range of priorities (low, medium, and high) - but I encountered difficulty getting the user-inputted string to accurately compare to a list of predetermined values. I also attempted to write header columns back to the .txt file - if anything, for the sake of formatting - but ended up just defining those key values instead, as that process was giving me less quarrel.

GITHUB LINK

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