Arun Naidu

February 18, 2023

IT FDN 110 A Wi 23: Foundations of Programming: Python

Assignment 06

https://github.com/arunnaidu2021/IntroToProg-Python-Mod06

Priority List and File I/O using Functions

Introduction

In this assignment a fairly large amount of code has already been written using functions. The code needs additions to make it fully operational.

Process

Once again, the script is divided up as follows:

1. Data
2. Processing
3. Input/Output

The **data** are the global variables used throughout the program. This includes the name of the data file and the variables used in manipulating the data.

The **processing** includes the Processor class which contains read and write to file and add and delete operations.

Since the row was already extracted, to add items to the list required a single line of code with the append method:

```
list_of_rows.append(row)
```

The deletion function was slightly trickier because the code needed to walk through the table to remove the appropriate row:

```
for row in list_of_rows:
    local_task, priority = dict(row).values()
    if local_task == task:
        list_of_rows.remove(row)
```

The final piece was to write the table out to a file, which we have done many times now and was actually a copy and paste from a previous assignment:

```
file = open(file_name, "w")
for row in list_of_rows:
    file.write(row["Task"] + "," + row["Priority"] + "\n")
file.close()
```

The IO class covers **input/output**. This includes the display of the menu, selection of items from the menu, display of the list, and the selection of items to add and remove from the list.

Most of this was already in place, with the only code being to input the tasks to add and remove dictionary entries.

Code operating in PyCharm:

```
******* The current tasks ToDo are: *******
Do homework (1)
Vacuum floor (5)
Organize desk (2)
****************************************


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


Which option would you like to perform? [1 to 4] - 1
```

Figure 1. PyCharm. The initial list is displayed.

```
Please enter a new task: Wash car
Please enter the task priority: 3
******* The current tasks ToDo are: *******
Do homework (1)
Vacuum floor (5)
Organize desk (2)
Wash car (3)
****************************************
```

Figure 2. PyCharm. The result of adding a new task.

```
*******************************************


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



Which option would you like to perform? [1 to 4] - 2

Please enter task to remove: Vacuum floor
******* The current tasks ToDo are: *******
Do homework (1)
Organize desk (2)
Wash car (3)
*******************************************
```
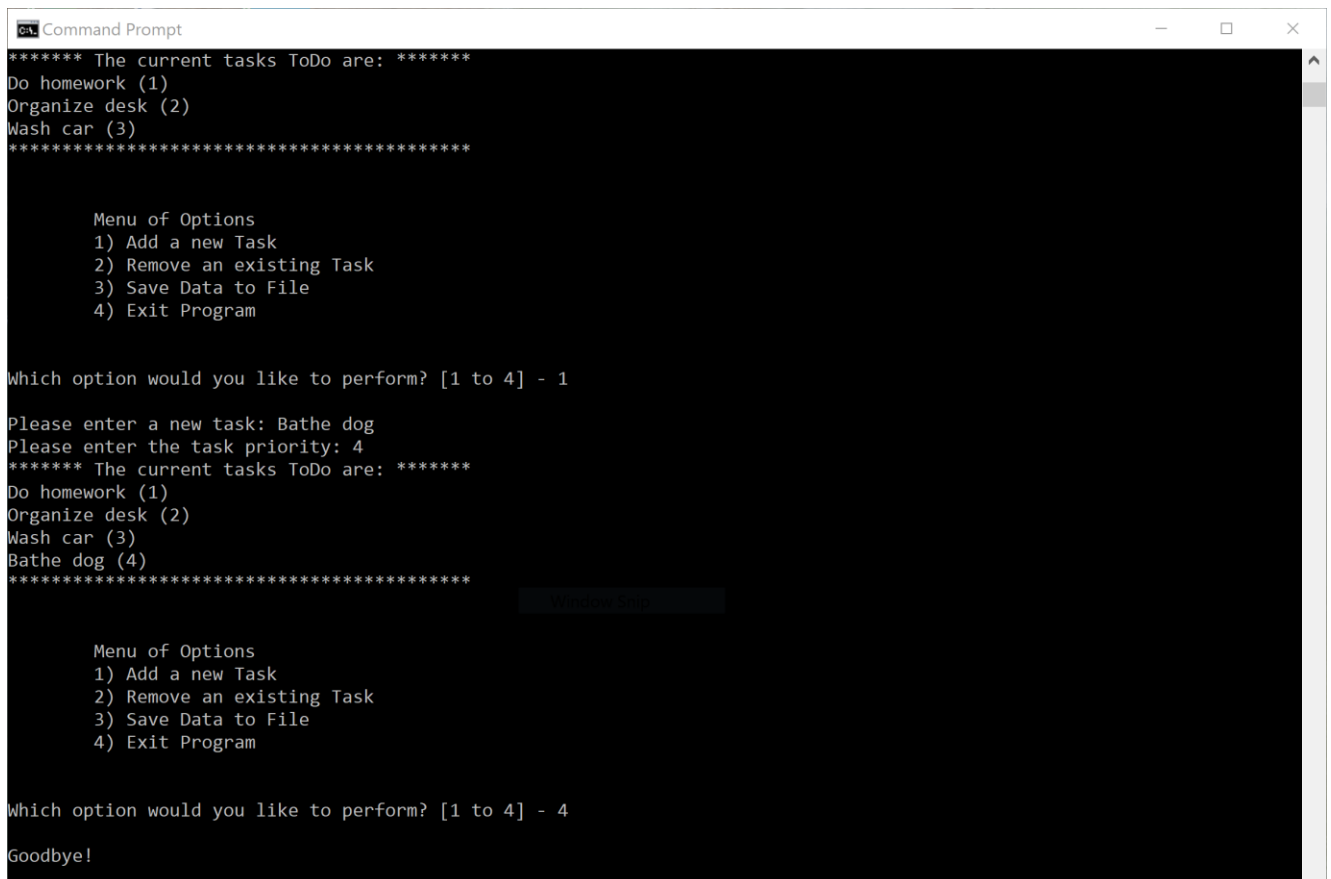
Figure 3. PyCharm. The result of removing a task.


```
Command Prompt                                                              —    □    ×
******* The current tasks ToDo are: *******
Do homework (1)
Organize desk (2)
Wash car (3)
*******************************************


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


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

Please enter a new task: Bathe dog
Please enter the task priority: 4
******* The current tasks ToDo are: *******
Do homework (1)
Organize desk (2)
Wash car (3)
Bathe dog (4)
*******************************************
                              Window Snip

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


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

Goodbye!
```

Figure 4: Python Code Operating in the Console

Summary

In this assignment much of the structure of the code was in place, making it much easier to follow than in the previous assignment where I had to build it up from scratch. I

 went through the sections where TODO markers were placed and added the necessary steps for the tasks to complete correctly. The documentation strings for each function defined the return data very clearly.