Johnathan Luu
Intro to Python
Assignment 5
5/24/2022
https://github.com/LofiLogan/

Functions

Intro

This week's assignment calls for editing a script that does the same tasks as last week's assignment, but using functions in classes instead. This adds to the formatting of Data, Processing, Presentation, and Main Body of Script because it organizes the script in a way where the process is more organized and sectioned off. It appears that this would make the writing and editing process of the script more straightforward and standardized for multiple reasons, primarily making it more straightforward for collaborations.

Creating the program

The script under the "Processing" section is very similar to that of the script from Assignment 5. The main difference in the formatting. The functions allow for the processes to be consolidated into specific sections that separate them from other tasks and refers to the specific function when called for. As a result, other than defining the class and functions, the script will look very similar:

Defining class:

```
class Processor:
    """  Performs Processing tasks """
```

Defining Functions, followed by similar script commands as Assignment 5:

```python
@staticmethod
def read_data_from_file(file_name, list_of_rows):
    """ Reads data from a file into a list of dictionary rows

    :param file_name: (string) with name of file:
    :param list_of_rows: (list) you want filled with file data:
    :return: (list) of dictionary rows
    """
    list_of_rows.clear()  # clear current data
    file = open(file_name, "r")
    for line in file:
        task, priority = line.split(",")
        row = {"Task": task.strip(), "Priority": priority.strip()}
        list_of_rows.append(row)
    file.close()
    return list_of_rows

@staticmethod
def add_data_to_list(task, priority, list_of_rows):
    """ Adds data to a list of dictionary rows

    :param task: (string) with name of task:
    :param priority: (string) with name of priority:
    :param list_of_rows: (list) you want filled with file data:
    :return: (list) of dictionary rows
    """
    row = {"Task": str(task).strip(), "Priority": str(priority).strip()}
    # TODO:
    list_of_rows.append(row)
    return list_of_rows

@staticmethod
def remove_data_from_list(task, list_of_rows):
    """ Removes data from a list of dictionary rows

    :param task: (string) with name of task:
    :param list_of_rows: (list) you want filled with file data:
    :return: (list) of dictionary rows
    """
    # TODO:
    for  row  in  list_of_rows:
        if  row["Task"].lower() == task.lower():
            list_of_rows.remove(row)
    return list_of_rows

@staticmethod
def write_data_to_file(file_name, list_of_rows):
    """ Writes data from a list of dictionary rows to a File

    :param file_name: (string) with name of file:
    :param list_of_rows: (list) you want filled with file data:
    :return: (list) of dictionary rows
    """
    # TODO:
    objFile = open(file_name, "w")
    for row in list_of_rows:
        objFile.write(row["Task"] + ',' + row["Priority"] + '\n')
    objFile.close()
    return list_of_rows
```

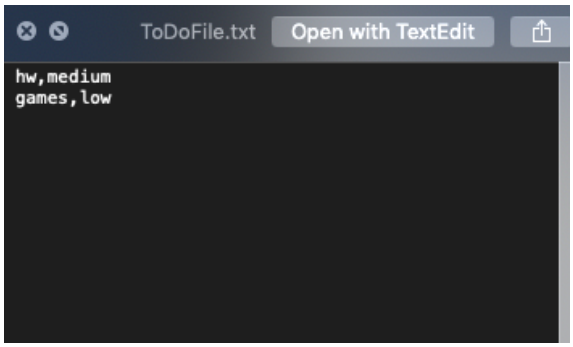
This is also the case for when we are defining the class and functions for the main menu. The code that we had to input simply were input functions to ask for the task and priority, or an input function to remove a task. This is the same as the previous assignment except we would be returning the variables of the “task” and/or “priority” at the end of the function:

```
@staticmethod
def input_new_task_and_priority():
    """  Gets task and priority values to be added to the list

    :return: (string, string) with task and priority
    """
    pass  # TODO:
    task = str(input("What is the task?")).strip()
    priority = str(input("What is the priority?")).strip()
    return task, priority

@staticmethod
def input_task_to_remove():
    """  Gets the task name to be removed from the list

    :return: (string) with task
    """
    pass  # TODO:
    # Add an extra line for looks
    task = str(input("What is the name of task you wish to remove? - ")).strip()
    print()
    return task
```

The “Main Body” of the script is actually the section for combining the functions from the two classes together. It is more streamlined than the script in assignment 5 because the functions are already defined, so it would simply call for the function to show the menu from class “IO”, followed by the task to do from the class “processing”.

Results

```
Last login: Tue May 24 10:58:28 on ttys003
StudioRig1:Assignment06 coding$ python3 Assignment06.py
******* The current tasks ToDo are: *******
eat (high)
*****************************************


        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

What is the task?hw
What is the priority?medium
******* The current tasks ToDo are: *******
eat (high)
hw (medium)
*****************************************


        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

What is the task?games
What is the priority?low
******* The current tasks ToDo are: *******
eat (high)
hw (medium)
games (low)
*****************************************


        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

What is the name of task you wish to remove? - eat

******* The current tasks ToDo are: *******
hw (medium)
games (low)
*****************************************


        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] - 3

Data Saved!
******* The current tasks ToDo are: *******
hw (medium)
games (low)
*****************************************


        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!
StudioRig1:Assignment06 coding$
```



Summary

In summary, functions are a great way to streamline code for more organized writing. In addition, defining the classes will group functions (as methods) to allow for more straightforward and compartmentalized organization.