Ann Yi
11/23/2022
IT FDN 110 A Au 22: Foundations Of Programming: Python
Assignment06
https://github.com/yiy4/IntroToProg-Python-Mod06

Working with Functions

Introduction

In this editable to-do list, the code was divided into three sections—data, processing, and presentation—to organize the script using the design principle separation of concerns.

I. Processing - Adding data to the list

```
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()}
    list_of_rows.append(row)
    return list_of_rows
```

With the task, priority, and list_of_rows given as parameters, I just added task and priority in the list_of_rows using **append()**.

II. Processing - Removing data from the list

```
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
    """
    for row in list_of_rows:
        current_task, priority = row.values()
        if task.lower() == current_task.lower():
            list_of_rows.remove(row)
    return list_of_rows
```

Similarly, given task and list_of_rows, I wrote a for loop asking the program to go through each task in the list_of_rows. Here, **current task, priority = row.values()** line allowed for values of task and priority to be retrieved together because row is a dictionary. I created a new variable current_task so that task in the parameter is differentiated from the task in the list_of_rows.

III. Processing – Write data to file

```
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
    """
```

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

Here, variable **file** was used to be consistent with read_data_from_file function in the previous section of the script. Tasks and priorities in the list_of_rows were written in the ToDoFile.txt using for loop.

IV. Presentation – Asking users what task/priority to add or remove

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

    :return: (string, string) with task and priority
    """
    task = str(input("What is the task? - ")).strip()
    priority = str(input("What is the priority? [high|low] - ")).strip()
    return task, priority
```

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

    :return: (string) with task
    """
    task = str(input("Which TASK would you like removed? - ")).strip()
    return task
```

Simply, the two functions above are asking users input and return those inputs to main body of the script.

Conclusion

Depending on what function was called in the main body of the script, the program read specific functions while disregarding others. It was interesting, but the lack of linear coding process (due to separating the script based on classes) was challenging because I had to keep scrolling up and down to see what and how the variables were referenced.

Running option 1 in PyCharm:

```
        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? - wash the car
What is the priority? [high|low] - low
******* The current tasks ToDo are: *******
Dishes (Low)
read (high)
email (high)
laundry (low)
wash the car (low)
*****************************************
```

Running option 2 in PyCharm:

```
        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

Which TASK would you like removed? - WASH THE CAR
******* The current tasks ToDo are: *******
Dishes (Low)
read (high)
email (high)
laundry (low)
*****************************************
```

Running options 3 & 4 in PyCharm:

```
Which option would you like to perform? [1 to 4] - 3

Data Saved!
******* The current tasks ToDo are: *******
Dishes (Low)
read (high)
email (high)
laundry (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!

Process finished with exit code 0
```