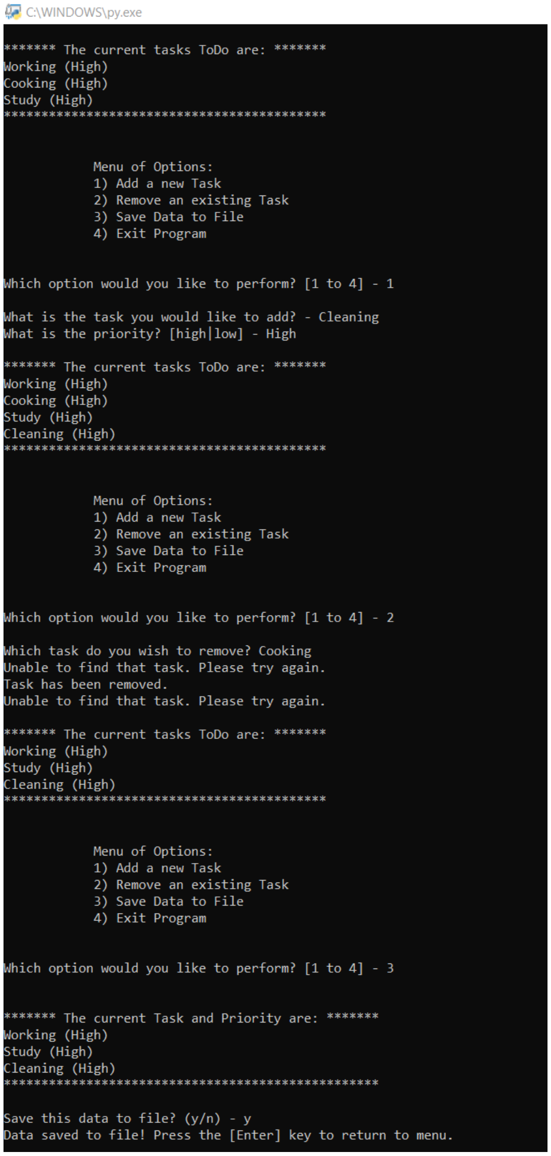
Vu Thai
November 21, 2022
Foundations of Programming Python
Assignment 6
https://github.com/VuThaiT/IntroToProgram_Python

“ToDoFile” Python Script



```
******* The current tasks ToDo are: *******
Working (High)
Cooking (High)
Study (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 you would like to add? - Cleaning
What is the priority? [high|low] - High

******* The current tasks ToDo are: *******
Working (High)
Cooking (High)
Study (High)
Cleaning (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] - 2

Which task do you wish to remove? Cooking
Unable to find that task. Please try again.
Task has been removed.
Unable to find that task. Please try again.

******* The current tasks ToDo are: *******
Working (High)
Study (High)
Cleaning (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] - 3


******* The current Task and Priority are: *******
Working (High)
Study (High)
Cleaning (High)
**************************************************

Save this data to file? (y/n) - y
Data saved to file! Press the [Enter] key to return to menu.
```

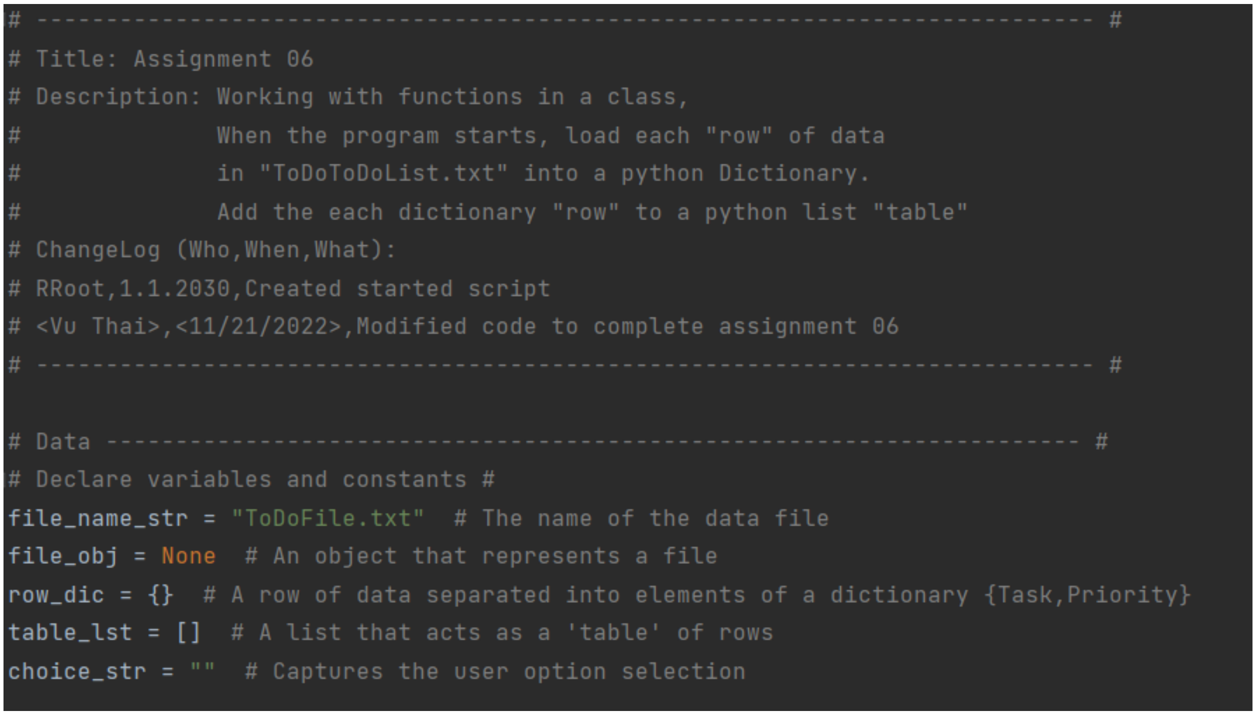
Figure 1.1: Sixth Assignment for Python class: Working with Functions.

Introduction

The objective of this assignment is to incorporate ***functions*** into what we have been learning with open, write, save, and remove input data from an existing text file. Similar to what we have accomplished last week, this week will introduce a new method to define functions and call upon these functions to execute the script's tasks.

Creating the Script with PyCharm

Similar to the previous week, we got a starter .py file with all the outlines and directions listed for us to follow. First of all, I edited all the Script Headers as well as declared extra variables that would be used for the script.

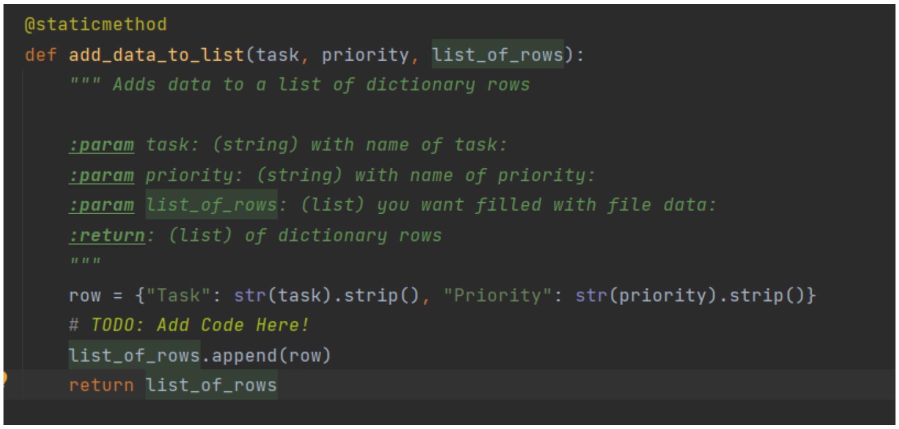


```
# --------------------------------------------------------------------------- #
# Title: Assignment 06
# Description: Working with functions in a class,
#              When the program starts, load each "row" of data
#              in "ToDoToDoList.txt" into a python Dictionary.
#              Add the each dictionary "row" to a python list "table"
# ChangeLog (Who,When,What):
# RRoot,1.1.2030,Created started script
# <Vu Thai>,<11/21/2022>,Modified code to complete assignment 06
# --------------------------------------------------------------------------- #

# Data -------------------------------------------------------------------- #
# Declare variables and constants #
file_name_str = "ToDoFile.txt"  # The name of the data file
file_obj = None  # An object that represents a file
row_dic = {}  # A row of data separated into elements of a dictionary {Task,Priority}
table_lst = []  # A list that acts as a 'table' of rows
choice_str = ""  # Captures the user option selection
```

Figure 2.1: Script Header for ToDoFile Script.

Much of the codes were already written for this script so I will go straight to the point and list the following 5 "TODO: Add Code Here!" from this assignment.

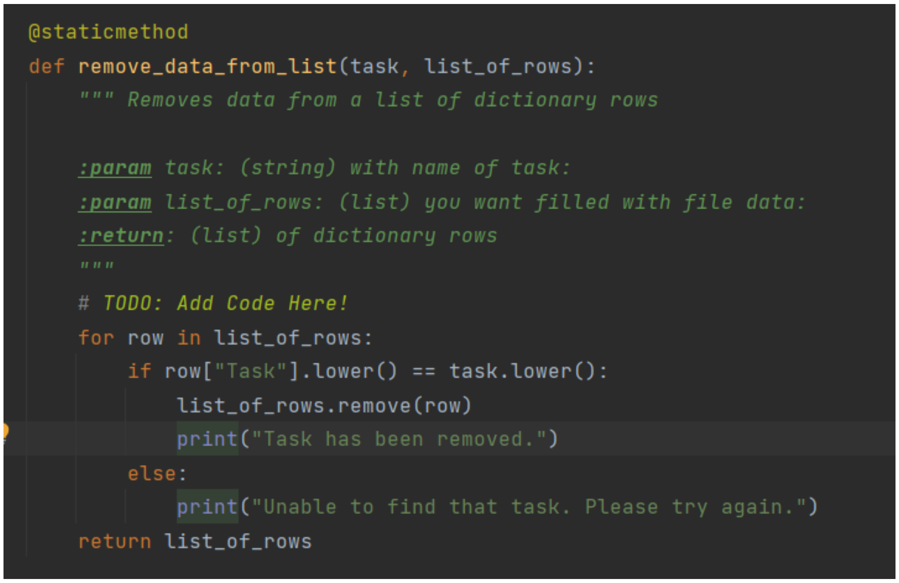
For the very first TODO for add_data_to_list function, since the instructor already gave the format for the variable row, all I need to do is add list_of_rows append(row) from what I have learned from the previous week to successfully add new input data into the existing list.



```
@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: Add Code Here!
    list_of_rows.append(row)
    return list_of_rows
```

Figure 2.2: Add data to list function with append.

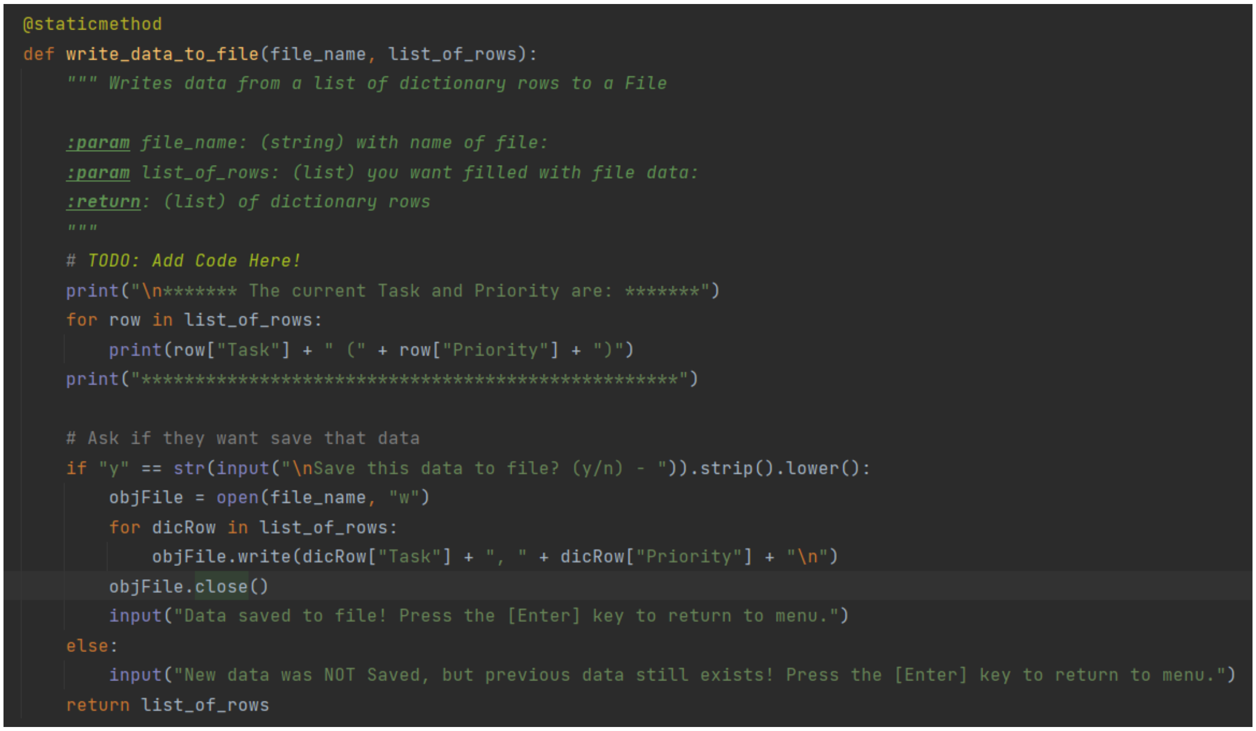
Next, the TODO for remove_data_from_list function is similar to what I used to finish last week’s assignment.



```
@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: Add Code Here!
    for row in list_of_rows:
        if row["Task"].lower() == task.lower():
            list_of_rows.remove(row)
            print("Task has been removed.")
        else:
            print("Unable to find that task. Please try again.")
    return list_of_rows
```

Figure 2.3: Remove data from list function with remove.

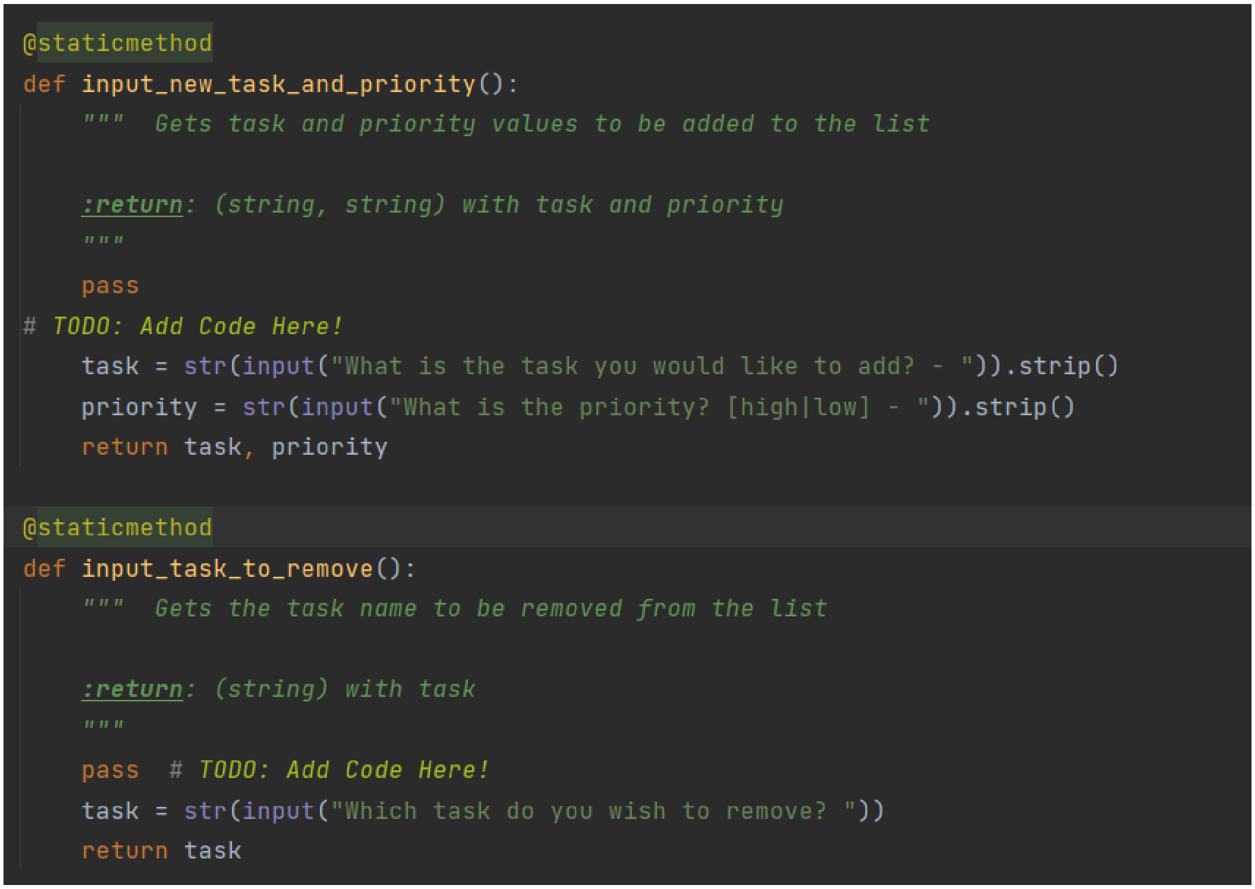
Since this week's assignment is pretty much a copy and paste from the last assignment, I will go ahead and re-use my codes to complete the rest of the assignments.



```python
@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: Add Code Here!
    print("\n******* The current Task and Priority are: *******")
    for row in list_of_rows:
        print(row["Task"] + " (" + row["Priority"] + ")")
    print("**************************************************")

    # Ask if they want save that data
    if "y" == str(input("\nSave this data to file? (y/n) - ")).strip().lower():
        objFile = open(file_name, "w")
        for dicRow in list_of_rows:
            objFile.write(dicRow["Task"] + ", " + dicRow["Priority"] + "\n")
        objFile.close()
        input("Data saved to file! Press the [Enter] key to return to menu.")
    else:
        input("New data was NOT Saved, but previous data still exists! Press the [Enter] key to return to menu.")
    return list_of_rows
```

Figure 2.4: Write data to file function with write.

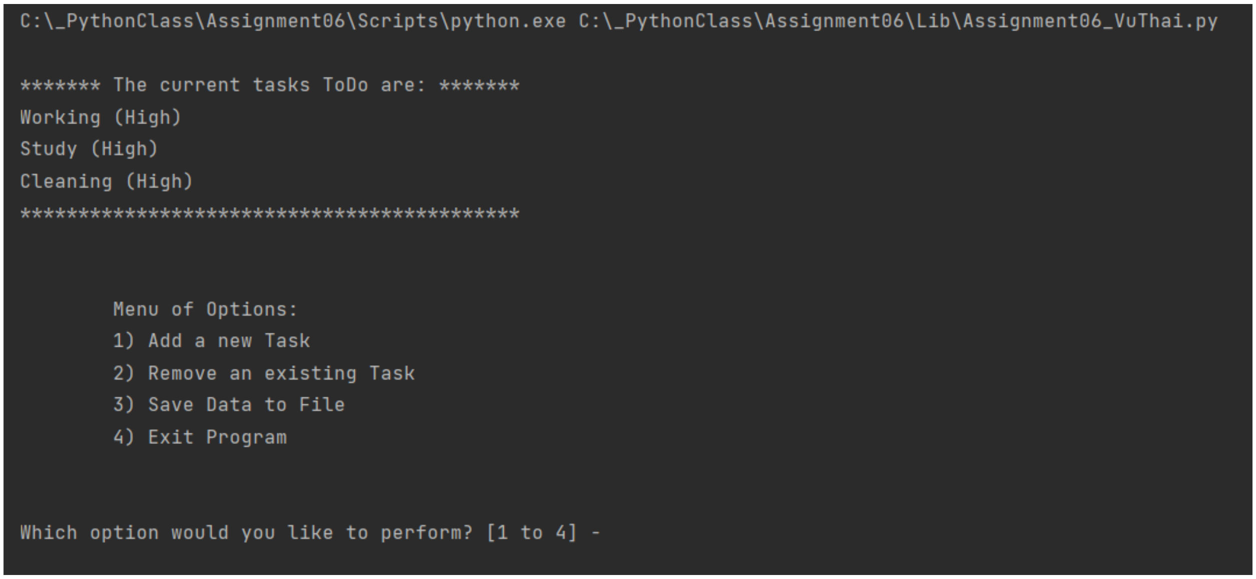
Next, for the last 2 TODO is for Input/Output functions, by following exercises from the books as well as labs provided by the lecture, I come up with the following:



```
@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: Add Code Here!
    task = str(input("What is the task you would like to add? - ")).strip()
    priority = str(input("What is the priority? [high|low] - ")).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 Code Here!
    task = str(input("Which task do you wish to remove? "))
    return task
```

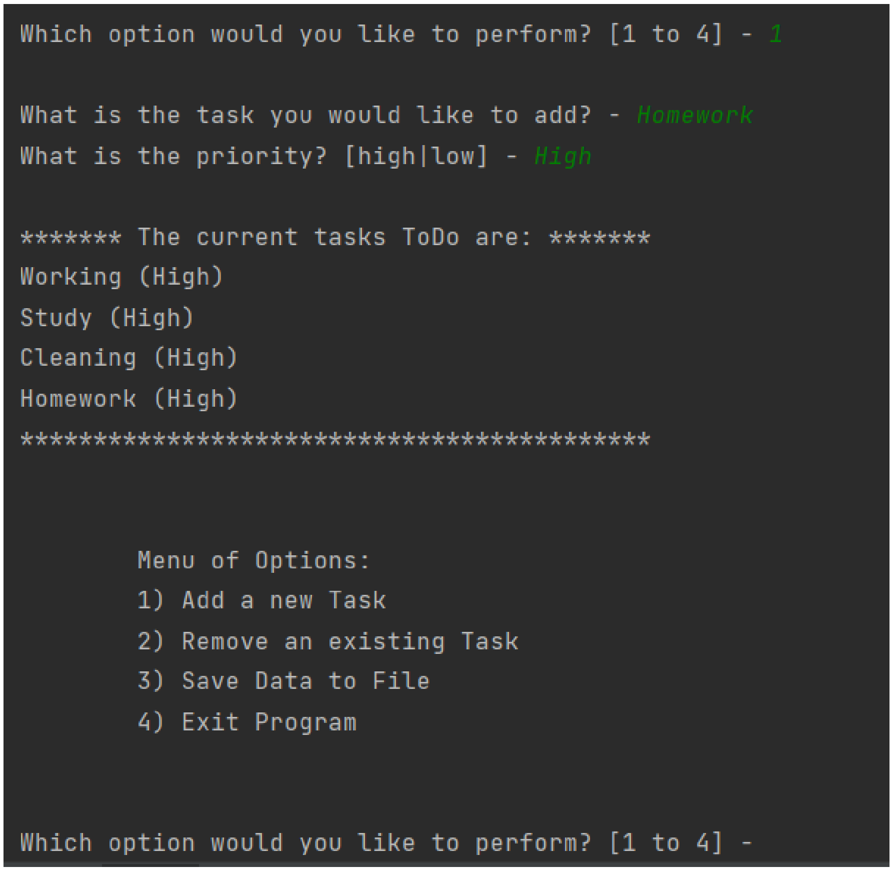
Figure 2.5: Input/Output functions for this script.

This assignment is more like a puzzle than actual coding since all I did for this script is re-using codes from the last assignment, with a little bit of improvement from looking at the instructor's codes, to make my script a little bit better. Since the instructor recommended splitting the codes into pieces and testing each function as I go, I made multiple test scripts for each of the TODO above and piece them together at the end. Thankfully, the instructor provided the green comments to help me better understand what should be returned since that one tripped me up the most... Below will be a series of screenshots of my script running on PyCharm.



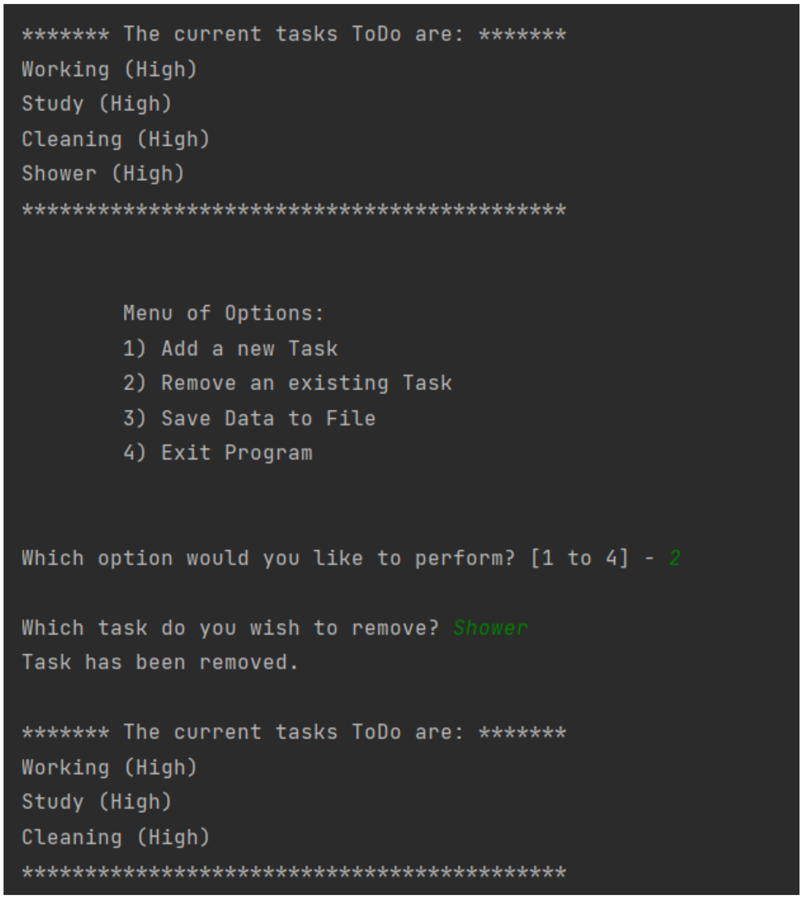
```
C:\_PythonClass\Assignment06\Scripts\python.exe C:\_PythonClass\Assignment06\Lib\Assignment06_VuThai.py

******* The current tasks ToDo are: *******
Working (High)
Study (High)
Cleaning (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] -
```

Figure 2.6: Display current data and main menu selection to the user.



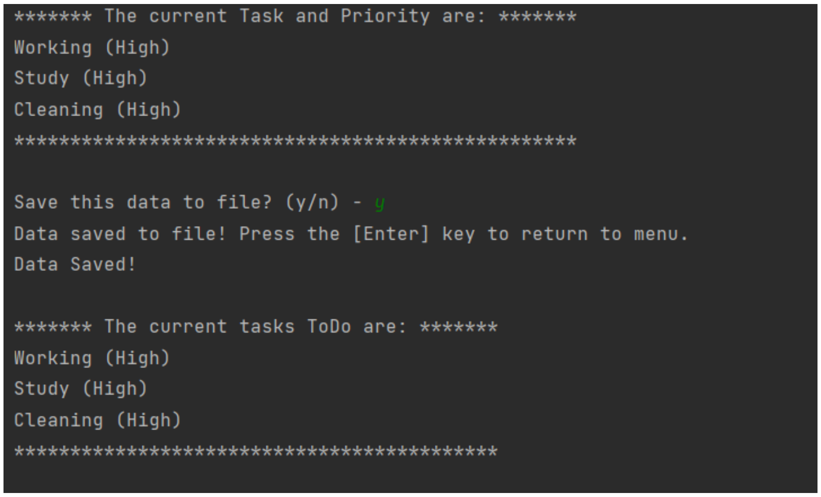
```
Which option would you like to perform? [1 to 4] - 1

What is the task you would like to add? - Homework
What is the priority? [high|low] - High

******* The current tasks ToDo are: *******
Working (High)
Study (High)
Cleaning (High)
Homework (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] -
```

Figure 2.7: Option 1 to add new task + priority to the existing list.



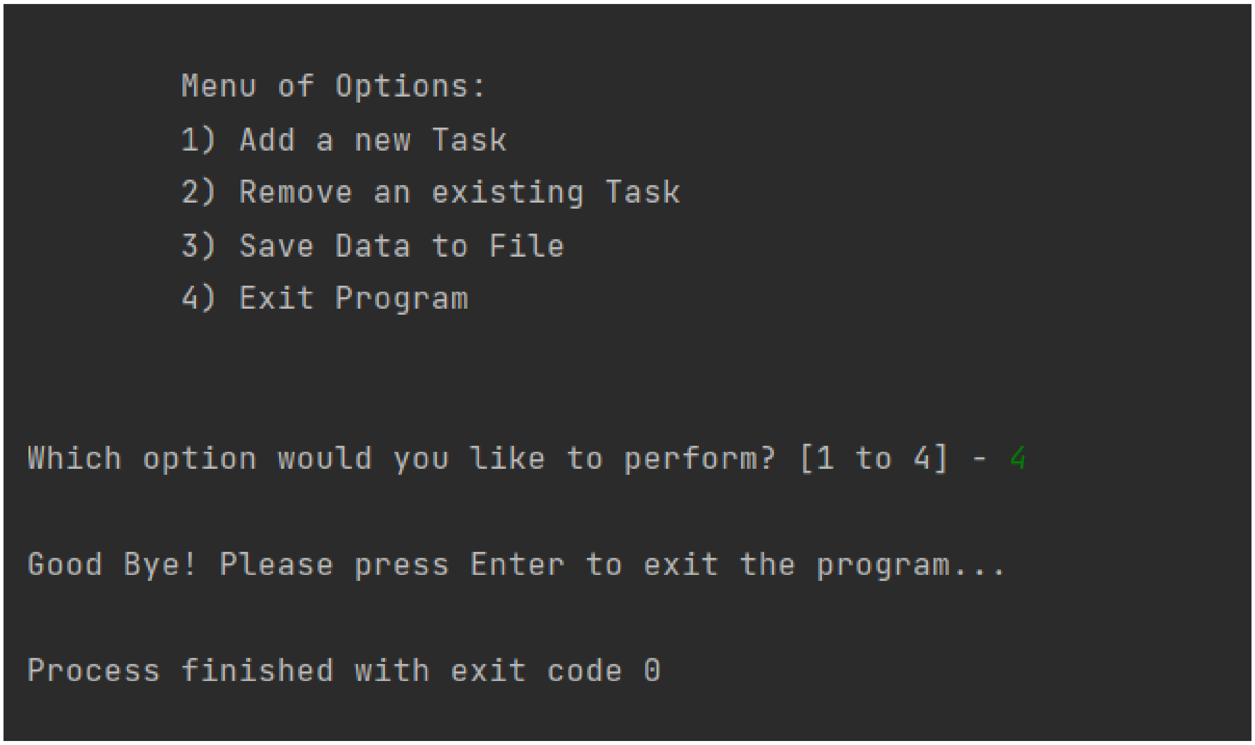
```
******* The current tasks ToDo are: *******
Working (High)
Study (High)
Cleaning (High)
Shower (High)
*******************************************


        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] - 2

Which task do you wish to remove? Shower
Task has been removed.

******* The current tasks ToDo are: *******
Working (High)
Study (High)
Cleaning (High)
*******************************************
```

Figure 2.8: Option 2 to remove a task from the existing list.



```
******* The current Task and Priority are: *******
Working (High)
Study (High)
Cleaning (High)
**************************************************

Save this data to file? (y/n) - y
Data saved to file! Press the [Enter] key to return to menu.
Data Saved!

******* The current tasks ToDo are: *******
Working (High)
Study (High)
Cleaning (High)
*******************************************
```

Figure 2.9: Option 3 to save all currently displayed data into ToDoFile.txt file.



```
        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

Good Bye! Please press Enter to exit the program...

Process finished with exit code 0
```

Figure 2.10: Option 4 to exit the program.

Conclusion

Overall, this assignment was definitely a challenging one at first since I don't even know how or where to get started. However, as I go through several and I really meant SEVERAL examples online as well as the labs and exercises from the books to finally get a hang of it. The trickiest part for me is to know what should be returning… Thankfully, we are re-using all the codes from the previous week so it saves a lot of time trying to figure out what should I write here and there. I can see how functions can be useful in a bigger project with several lines of code since we can recall the functions anytime we need them but for a small assignment like this, it was used only once or twice so I found it more complicated than the previous assignment.