

Shuhua Liang

2022-11-15

IT FDN 110 A

Assignment05

Lists and Dictionaries

Introduction

From this lecture, I learned (1) different file read/write features. (2) How to define a dictionary and a list of dictionaries. (3) How to change values in a Dictionary. (4) How to access the values in a Dictionary.

Read txt file using `open()`

Following the blueprint from the homework instructions, the txt file is declared at the beginning of the file. However, I was confused to see that an object is declared in the beginning instead. I tried to follow the instructions but failed to find a way to directly declare the object by assigning the txt file name to it. In the end, I decided to declare the `objFile` in a different way as in Fig 1b.

```
# declare variables and constants
objFile = "ToDoList.txt" # An object that represents a file
```

Figure 1a, directly declare the object that represents a file.

```
# declare variables and constants
strFile = "ToDoList.txt" # text file to read
objFile = "" # An object that represents a file
```

Figure 2b, declare the object and txt file separately.

Read txt file and convert the data in a “list of dictionaries”

Different from previous homework, this time we need to manually create a txt file with Tasks and Priorities for the code to read. When creating the file, I must type enter to start a new row. That causes a problem. Python somehow picks up the signal as “\n”. I decided to use `row.rstrip('\n')` to fix the problem.

```
C:\_PythonClass\Assignment05\venv\Scripts\python.exe C:\_PythonClass\Assignment05\Assignment05_Shuhua_
[{'Task': 'Learn to type', 'Priority': ' High\n'}, {'Task': 'Learn to drive', 'Priority': ' High\n'},
```

Figure 2, Manually typed txt file shows “\n” sign.

To read a txt file in to a “list of dictionaries” is more complicated than I thought. I must first read each row and assign the row elements into a dictionary and append the dictionary to the list.

```
objFile = open(strFile, "r")
for row in objFile:
    row = row.rstrip('\n') # remove \n in txt file
    strData = row.split(",") # Returns a list!
```

```
dicRow = {"Task":strData[0], "Priority":strData[1]}
lstTable.append(dicRow)
```

Figure 3, Assign txt file data into a list of dictionaries.

Update txt file and Access dictionary items

There are two additional challenges I encountered in the home work are:

(1) how to update the txt file

It will be difficult to edit the txt file without delete/overwrite it. In the previous homework, there is no “delete item” option, so we can simply append a new item to the end of the old txt file. In this homework, there is a “delete task” option. As a result, the whole list need to be write to the txt file and overwrite the old version. I first tried to use `open(strFile, "w")` where “w” overwrite the old version. But since I am writing every dictionary one after another, this will only leave one row of data in the txt file. I also tried to write the whole list to the txt file at once without going through the for loop. But I failed to find out how to do that. In the end, I used a quite dumb way. I first overwrite the old txt file without enter any data. Then, I use `open(strFile, "a")` to append each row one by one to the empty txt file. (2) How to write list of dictionaries into text file.

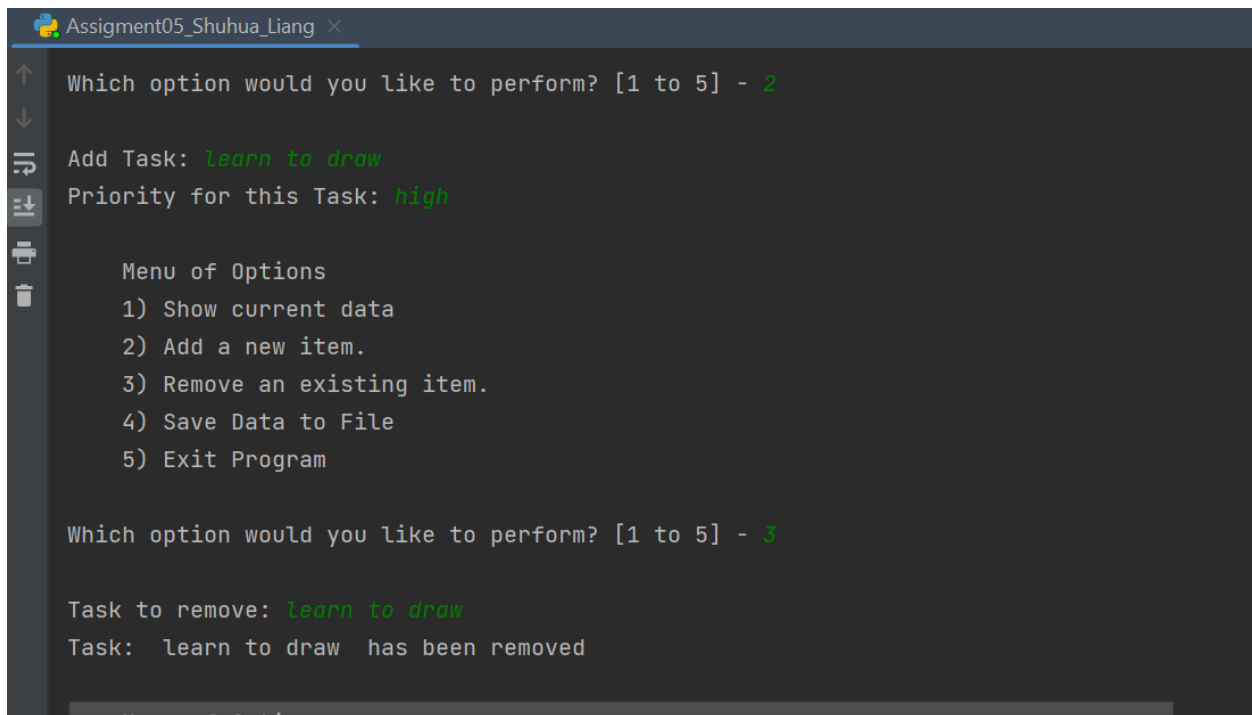
In the previous homework, I write a list of list (table) into a txt file, which is rather straight forward. I used 0 and 1 to access elements in each row. But this method failed in this homework and caused an error. Now we are dealing with a list of dictionaries. So, I can no longer access elements inside a dictionary with 0 and 1. Instead, I need to use keys and values to access each element.

```
elif strChoice.strip() == '4':
    objFile = open(strFile, "w")
    objFile.close()
    # overwrite the old file and start a new file
    objFile = open(strFile, "a")
    for row in lstTable:
        x = str(row["Task"]) + ',' + str(row["Priority"]) + '\n'
        objFile.write(x)
    objFile.close()
```

Figure 4: Writing a list of dictionaries to a txt file is more complex than writing a list of list (table)

Results

I overcame all obstacles and finished the homework. Some other challenging parts are adding/removing tasks from the list. As you can see in the figure below, the code was able to conduct operations as instructed.

A terminal window titled 'Assigment05_Shuhua_Liang' showing the execution of a program. The user enters '2' for the first menu option, adding the task 'learn to draw' with 'high' priority. Then, the user enters '3' for the second menu option, removing the task 'learn to draw'. The terminal output shows the menu of options and the confirmation of task removal.

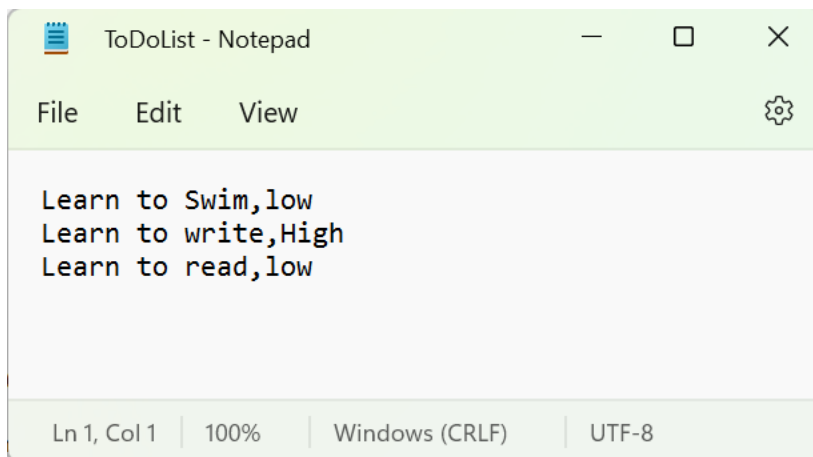
```
Assigment05_Shuhua_Liang x
Which option would you like to perform? [1 to 5] - 2
Add Task: learn to draw
Priority for this Task: high

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

Task to remove: learn to draw
Task: learn to draw has been removed
```

Figure 5: The code responded to inputs as designed.

A Notepad window titled 'ToDoList - Notepad' showing the contents of a text file. The file contains three lines of text, each representing a task with its priority: 'Learn to Swim,low', 'Learn to write,High', and 'Learn to read,low'. The status bar at the bottom indicates the cursor is at line 1, column 1, the font size is 100%, the encoding is Windows (CRLF), and the character set is UTF-8.

```
ToDoList - Notepad
File Edit View
Learn to Swim,low
Learn to write,High
Learn to read,low
Ln 1, Col 1 | 100% | Windows (CRLF) | UTF-8
```

Figure 6: Final results with tasks added/removed are saved in the ToDoList.txt file

Summary

In this project, I learned how to construct a list of dictionaries, access elements in the dictionaries and edit them. I also learned how to write data into txt files with different modes, such as read only, write only and append only.