Luis F Lopez Penunuri
February 15, 2023
IT FDN 110 A
Assignment 05

ToDo LIST.

Introduction

In this practice, we will learn how to manipulate data from a text document (.txt). We will be able to add new information, save the information and remove data from the list.

Practice

Jose was amazed with our job and how we helped him with his moving business. But like every customer, he wants something more. Now, he is asking for a solution that can help him to prioritize his employees' tasks. He needs to create a list of tasks with priorities

This service won't be cheap for you Jose, but I will do my best to provide a solution that fits your needs.

The Script:

To manipulate files, the script uses the module "os.path", which provides functions to check if a file exists ("os.path.exists"), and to manipulate files (*Figure 5.1 and 5.2*).

With function "os.path.exists" the script (*Figure 5.3*) checks if the file exists, and if it does exist, it reads its content into the list of dictionaries using a for loop.

The script displays a menu (*Figure 5.9*) of options to the users, asking the user to select and option by inputting a number from 1 to 5. If the user selects option 1, the script displays the current data in the list of dictionaries. If the user selects option 2, the script prompts the user to enter a new item and then adds the new item to the list of. Dictionaries. If the user selects options 3, the script prompts the user to enter the ID of the task to remove, and then removes the task from the list of dictionaries.

If the user selects option 4, the script saves the data to the set file. If the user selects option 5, the scripts exits the program

Show Current Data

This option displays the current tasks in the list. The tasks are read from the text file “ToDoList.txt”, which should be in the same directory as the script. The tasks are displayed in a table with two columns: ID and Task name (*Figure 5.4*).

Add a New Item (Task)

This option allows the user to add a new task to the list. The user is prompted to enter an ID and task name. The ID should be a unique identifier for the task, and the task name should be a description of the task. This is not working as it should, since I don’t have any validation to make sure the ID is unique (*Figure 5.5*).

Remove an Existing Item

This option allows the user to remove a task from the list. The user is prompted to enter the ID of the task they want to remove. If the task is found in the list, it will be removed (*Figure 5.6*).

Save Data to File

This option saves the current list of tasks to the text file “ToDoList.txt”. Each task is saved as a comma-separated line in the file (*Figure 5.7*).

Exit Program

This option close the interface with the .txt file and exit the application (*Figure 5.8*).

Technical Details

This application is implementing the “os.path” and “typing” modules.

The *os.path* module is used to check if the ToDoLis.txt exists and to save the list of tasks to file. The *typing* module is used to provide type hints for function arguments and return values.

The list of tasks is stored in memory as a list of dictionaries. Each dictionary represents a task and has 2 keys: “ID” and “Task”. The list of dictionaries is loaded from text file ToDoList.txt when the application starts and is updated as the user adds or removes tasks.

Challenges

The whole practice was kind of though. The hard part for me was the “Removed and Existing Item” option. Another problem I had was trying to modify the code that was already there. Honestly, I was not able to fully use it, so I created some local variables.

Pictures:

```
# import necessary modules
from typing import TextIO
import os.path # Functions to manipulate files
```

Figure 5.1 Import modules

Code #1: Use of os.path.exists() method

```
# Python program to explain os.path.exists()

# importing os module
import os

# Specify path
path = '/usr/local/bin/'

# Check whether the specified
# path exists or not
isExist = os.path.exists(path)
print(isExist)
```

Figure 5.2 https://www.geeksforgeeks.org/python-os-path-exists-method/

```
if os.path.exists("ToDoList.txt"):
    with open(objFile, "r") as file:
        for toDos in file:
            lstTable.append(toDos.strip().split(","))
```

Figure 5.3 os.path.exist Implementation

```
if strChoice.strip() == '1':
    print("ID\tTask")
    for toDo in lstTable:
        print(toDo[0], toDo[1])
    continue
```

Figure 5.4 Option 1 Display values from ToDoList.txt

```
elif strChoice.strip() == '2':
    strID = input("Enter ID:")
    strName = input("Enter activity:")
    lstTable.append([strID, strName])
    print("Item added.")
```

Figure 5.5 Option 2 Add a Task

```
elif strChoice.strip() == '3':
    strID = input("Enter the ID of the task you want to delete: ")
    bTaskFound = False
    for toDo in lstTable:
        if toDo[0] == strID:
            lstTable.remove(toDo)
            bTaskFound = True
            print(f"Task with ID {strID} has been deleted.")
            break
    if not bTaskFound:
        print("Task not found.")
    continue
```

Figure 5.6 Option 3 Remove Task

```
elif strChoice.strip() == '4':
    with open("ToDoList.txt", "a") as file:
        for toDo in lstTable:
            file.write(", ".join(toDo) + "\n")
    print("Data saved to file.")
```

Figure 5.7 Option 4 Save data into ToDoList.txt

```
# OPTION 5 - Exit program
elif strChoice.strip() == '5':
    print("Exiting the program")
    file.close()
    break  # and Exit the program
```

Figure 5.8 Option 5 Close file and Exit

PyCharm

```
    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] - 1
ID  Task
1  aaa
2  ccc
1  aaa
2  ccc
3  ttt
4  rrr
```

Figure 5.9 Menu Option 1

```
    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] - 2
Enter ID:5
Enter activity:jjj
Item added.
```

Figure 5.10 Option 2

```
    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
Enter the ID of the task you want to delete: 1
Task with ID 1 has been deleted.
```

Figure 5.11 Option 3

```
    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] - 4
Data saved to file.
```

Figure 5.12 Option 4

```
Which option would you like to perform? [1 to 5] - 5
Exiting the program

Process finished with exit code 0
```

Figure 5.13 Option 5

Terminal

```
    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] - 1
ID      Task
1  aaa
2  ccc
1  aaa
2  ccc
3  ttt
4  rrr
```

Figure 5.14 Option 1

```
    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] - 2
Enter ID:6
Enter activity:yyy
Item added.

    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] - 1
ID      Task
1   aaa
2   ccc
3   ttt
4   rrr
5  jjj
6 yyy
```

Figure 5.15 Option 2

```
Which option would you like to perform? [1 to 5] - 3
Enter the ID of the task you want to delete: 6
Task with ID 6 has been deleted.

    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] - 1
ID      Task
1   aaa
2   ccc
3   ttt
4   rrr
5  jjj
```

Figure 5.16 Option 3

```
    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] - 4
Data saved to file.
```

Figure 5.17 Option 4

```
    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] - 5
Exiting the program
luislopez@Luiss-MacBook-Pro Module05 %
```

Figure 5.18 Option 5

Conclusion

This practice took me three times longer than I thought it would. This is bad because the quote I gave was for three times less. Lesson learned. But Jose has his application running.

In the other hand, this was a good practice, and I can see some real-life application in my mind. However, I can also see that the next one will be harder than this one.