Leila Mehrabi

06/05/2023

IT FDN 110 A Sp 23: Foundations of Programming: Python

Assignment05

Github: https://github.com/mehrabileila/IntroToProg-Python

Working with dictionaries in Python

Introduction

In this lesson, we learned about dictionaries in Python and how to work with them. We made a program that creates a table of tasks and priorities. The user can add, delete, display, or save the information to the file.

Input Data

When the program starts, the contents of the file "ToDoList.txt" is loaded to the memory and held in a list called "lstTable," which acts as a table of dictionaries.

Next, the user gets prompted with a menu to choose an item (Figure 1). There are five different options available:

```
    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] -
```

Figure 1: Menu options

1- Show Current data

If the user chooses number 1, the current data which has already been loaded from the txt file to "lstTable" at the beginning of the program is displayed (Figure 2)

```
   TASK    |  PRIORITY
------------------------------
Assignment | 1
video | 4
call | 5
studying | 6
doing lab | 5
Press any key to go back to the main menu
```

Figure 2: Choice 1

2- Add a new Item

If the user makes choice number 2, they will be given the option to add a new item to the tasks list. If the task already exists, the user will be warned that the task already exists, and the program doesn't let them add a duplicate entry (Figure 3)

```
Which option would you like to perform? [1 to 5] - 2

Please input the task name: writing code
Please input the task's priority: 4
```

Figure 3: Choice 2

3- Remove an Existing Item

If the user chooses number 3, they will be asked what task they want to remove. And if the task exists in the list, it will be deleted. If the task doesn't exist, the user will get warned that the task doesn't exist (Figure 4)

```
Which option would you like to perform? [1 to 5] - 3

Please enter the task you want to delete: writing code
The task 'writing code' deleted
```

Figure 4: Choice 3

4- Save Data to the file

If the user makes choice number 4, they will be asked if they are sure to save the data to the file, and if they press “y,” the current data in “lstTable” will be saved to the “ToDoList.txt” file, and the user will be informed. If the user press “n,” the program returns to the main menu.

```
Which option would you like to perform? [1 to 5] - 4

Are you sure you want to save the current data to the file? y or n? y
Data saved to the file!
 Press any key to continue
```

Figure 5: Choice 4

5- Exit Program

If the user makes choice number 5, they will be asked if they are sure they want to exit the program, and if they press “y,” the program will end. If they press any other key, the program will return to the main menu.

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

Are you sure you want to exit? y or n: y
```

Figure 6: Choice 5

Choice 1 Code in Detail

If the user chooses 1, they want to see the current data. When The program starts, it loads the current data from the txt file to the lstTable list. “lstTable” will get edited later during the program when the user tries to add or remove anything from the current data. When the user chooses 1, the content of “lstTable” will be displayed.

```
if (strChoice.strip() == '1'):
    # TODO: Add Code Here
    print("   TASK   "+"|"+"  PRIORITY  ")
    print("------------------------------")
    for row in lstTable:#reads all rows in the list and displays the value
        print(row["Task"]+" | "+row["Priority"])
    input("Press any key to go back to the main menu ")
    continue
```

Figure 7: Choice 1 code

Choice 2 Code in Detail

If the user chooses number 2 from the main menu, they want to add data to the list. So, they are prompted to enter the task into variables *strTask.* The program then checks to see if the task already exists; if so, the user will be warned, and the program goes back to the main menu. A helping variable called “helpstrExist” determines if the task exists. If the task doesn’t exist, the program will ask for the priority and keep it in a variable called “strPriority.” Then two variables, “strTask” and strPriority,” are used to create a dictionary. This dictionary will be appended to the “lstTable” nested list, which in fact, acts as a table of dictionaries. (Figure 8)

```
elif (strChoice.strip() == '2'):
    # TODO: Add Code Here
    strTask=str(input("Please input the task name: "))
    helpstrExist=0 #helping variable which helps to determine if the added task already exists?
    for row in lstTable:#checks to see if the task already exists. If so, it doesn't let the user enter the task
        if(row["Task"].lower()==strTask.lower()):
            print("The task already exists")
            helpstrExist = 1
            break
    #End For
    if(helpstrExist!=1):#if the task doesn't already exist it goes to next step and asks for the priority and adds it to the table
        strPriority=str(input("Please input the task's priority: "))
        dicRow={"Task":strTask,"Priority":strPriority}
        lstTable.append(dicRow)
        continue
    else:
        input("Press any key to continue ")
        continue
```

Figure 8: Choice 2 code

Choice 3 Code in Detail

If the user chooses 3, they want to remove an existing item (Figure 9). The program prompts the user to enter the task they want to remove from the list and stores it in the “strTask”

variable. Then it'll check whether the entered task exists in the list. The program will warn the user and return to the main menu if it doesn't. If it does, the program will remove the task and its relevant priority from the list. The helping variable "helpstrFound" allows the program to check if the entered task exists. (Figure 9)

```
elif (strChoice.strip() == '3'):
    # TODO: Add Code Here
    strTask=str(input("Please enter the task you want to delete: "))
    helpstrFound = 0 #helping variable that helps determoine if the entered task exists?
    for row in lstTable:
        if(row["Task"].lower()==strTask.lower()):#if the task exists, it deletes the relevant row from the table and informs the user
            lstTable.remove(row)
            print("The task"+" '"+strTask+"' "+"deleted\n")
            input("Press any key to continue ")
            helpstrFound=1
            break
```

Figure 9: Choice 3 code

Choice 4 Code in Detail

If the user makes choice 4, then they want to save the current data in "lstTable" to the file "ToDoList.txt." The program then asks again if they're going to save the data. If they press "y," the current will be saved, and the user will be informed. If they press "n," the user will be warned that the data didn't save, and nothing will be saved. If they press anything else other than "y" or "n," then the program will keep asking to press only "y" or "n" (Figure 10)

```
elif (strChoice.strip() == '4'):
    # TODO: Add Code Here
    while (True):
        strSave = str(input("Are you sure you want to save the current data to the file? y or n? "))
        if (strSave.lower() == 'y'):  # saves the current data to the file
            objfile = open("ToDoList.txt", "w")
            for row in lstTable:
                objfile.write(row["Task"] + "," + row["Priority"]+"\n")#writes current data to the file
            input("Data saved to the file! \n Press any key to continue")
            objfile.close()
            break
        elif (strSave.lower() == 'n'):
            input("Data didn't save to the file \n Press any key to continue ")
            break
        else:
            print("Please enter only 'y' or 'n': ")  # checks if any key other than y or n is entered by the user
    continue
```

Figure 10: Choice 4 code

Choice 5 Code in Detail

If the user makes choice 5, then they want to exit the program. The code then will ask them if they are sure to exit, and if they press “y,” the program will end. If they press any other key, the program will return to the main menu.

```
# Step 7 - Exit program
elif (strChoice.strip() == '5'):
        strExit=str(input("Are you sure you want to exit? y or n: "))#double checks if the user really wants to exit?
        if(strExit.lower()=="y"):
            print("End of Program")
            break # and Exit the program
        # TODO: Add Code Here
        else:
            print("Please enter any key to go back to the main menu ")
            continue
```

Figure 11: Choice 5 code

```
C:\_PythonClass\Assignment05\venv>python Assigment05_LeilaMehrabi.py

    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

   TASK   |  PRIORITY
------------------------------
Assignment | 1
video | 4
call | 5
studying | 6
doing lab | 5
Press any key to go back to the main menu
```

Figure 12: Running the program in Windows command prompt

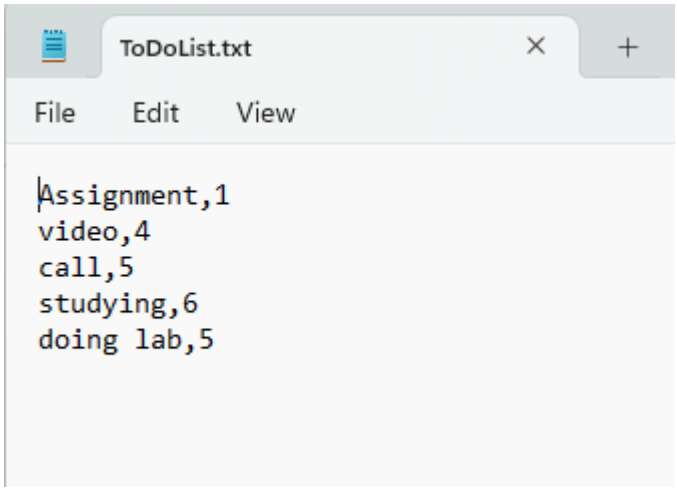

Figure 13: The output file "HomeInventory.txt

Summary

In this assignment, we learned how to create an interactive program to read data from the user, edit it and write it to the file. We also learned how to work with the dictionary object and use it with loops to read, write or display items.