IMarshall

02/12/2023

IT FDN 110 A

Assignment 05

IMarshallUW/IntroToProg-Python

Assignment 05 – List and Dictionaries

Introduction

In this paper we will build upon the knowledge we gained in module 04 where we created and saved a list to a file. This time will use a dictionary instead of a list format, and include the ability to delete data from memory. While code will be highlighted showing duplicated efforts from module 04, please refer to the Assignment 04 – Collections report for a more details explanation.

Writing Code

For this project we wanted to build on our knowledge from the previous module where we took input from a user and saved it to a file. While the data was foramtted into a list format, this time we formatted data into dictionary format. Both data types are treated the same way as far as the computer is concerned. The difference being in a list columns are labeled numerically from 0 in an ascending order, whereas in a dictionary the columns are given names by the program to make it easier the user to identify the data contained in the column. For this assignment we chose to name the columns “Tasks” and “Priority” respectively.

We began with a starter code, typical for scenarios where someone is expected to continue a project where someone left off. This code gave the skeleton for a while loop that contained 5 user options and the if/elif statements for said options (*figure 1)*.

```
while (True):
    print("""
    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
    """)
    strChoice = str(input("Which option would you like to perform? [1 to 5] - "))
    print()  # adding a new line for looks
    # Step 3 - Show the current items in the table
    if (strChoice.strip() == '1'):
        # TODO: Add Code Here
        continue
    # Step 4 - Add a new item to the list/Table
    elif (strChoice.strip() == '2'):
        # TODO: Add Code Here
        continue
    # Step 5 - Remove a new item from the list/Table
    elif (strChoice.strip() == '3'):
        # TODO: Add Code Here
        continue
    # Step 6 - Save tasks to the ToDoToDoList.txt file
    elif (strChoice.strip() == '4'):
        # TODO: Add Code Here
        continue
    # Step 7 - Exit program
    elif (strChoice.strip() == '5'):
        # TODO: Add Code Here
        break  # and Exit the program
```

Figure 1: Starter code

To build on this code we wrote a try statement that would attempt to pull data from an existing file by the same name that this program would write to. Using a for loop we made it so the file would be read row by row from the file, stored in local memory, and presented to the user. If the file didn't exist the program would present the user with a message saying as much (*figure 2*).

```
try:
    objFile = open('ToDoList.txt', 'r')
    for row in objFile:
        lstRow = row.split(',')
        dicRow = {'Task': lstRow[0].strip(), 'Priority': lstRow[1].strip()}
        lstTable += [dicRow]
    objFile.close()
    print('Task:'.ljust(40) + 'Priority:')
    for row in lstTable:
        print(row['Task'].ljust(40) + row['Priority'])
    print('^Data currently in file')
# Double print lstTable, investigate later
except:
    print('''
        Attempted to continue data entry from file.
        No file exists, a new file will be created when you save
        ''')
```

Figure 2: Read data in file and write to program memory, notify user is no file exists

Since the menu was written in the starter code we wrote the code for the first option, which would show the user the data currently in the program's memory. To make it easier on the human eye we formatted the print statement to show the dictionary column titles as column header, and justified the "Task" column left with 40 spaces after it before the "Priority" column started. This created the visual appearance of columns to the user (*figure 3 & 4*). We also forced a user interface after the data is presented so that it didn't become lost as the loop continued back to the options menu.

```
    if (strChoice.strip() == '1'):
        # TODO: Add Code Here
        print('The current data is: ')
        print('Task:'.ljust(40) + 'Priority:')
        for row in lstTable:
            print(row['Task'].ljust(40) + row['Priority'])
# Print's in a user friendly format while reminding user of dictionary headers
        input('Press ENTER to continue')
        continue
```

Figure 3: Option 1 code (show current memory)

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

The current data is:
Task:                                   Priority:
Sweep                                   2
laundry                                 5
dishes                                  4
Press ENTER to continue3
```

Figure 4: Option 1 (memory presented to user)

For option 2 we allowed the user to add data. This process was identical to the process laid out in the assignment 04 write-up with the exception of creatin a while loop with a try statement that forced the user to enter a preexisting acceptable value for their priority instead of free-form text (*figure 5 & 6*).

```
    # Step 4 - Add a new item to the list/Table
    elif (strChoice.strip() == '2'):
        # TODO: Add Code Here
        print('Type in a task and priority from 1 (low) to 5 (high)')
        strTask = str(input('Enter a task: ').strip())
        while True:
            try:
                strPri = str(input('Enter a priority: '))
            except ValueError:
                print('Please enter a valid integer between 1-5')
                continue
            if (strPri == '1'):
                break
            elif (strPri == '2'):
                break
            elif (strPri == '3'):
                break
            elif (strPri == '4'):
                break
            elif (strPri == '5'):
                break
            else:
                print('Please enter a value between 1 and 5')
                # Holds user to defined priority range
        dicRow = {'Task': strTask, 'Priority': strPri}
        lstTable += [dicRow]
        continue
```

Figure 5: Option 2 (Only accepting 1-5 for priority)

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

Type in a task and priority from 1 (low) to 5 (high)
Enter a task: Sweep
Enter a priority: 7
Please enter a value between 1 and 5
Enter a priority: 2
```

Figure 6: Option 2 user interface

Option 3 was to delete a row from memory. This was done with a for loop to check every row in the memory against a user input and delete said row if the string matched. To help ensure that the entries weren't case sensitive we also added a .lower() function that forced both the user input and the data in the "Task" column being compared were treated as being entirely in lower case letters (*figure 7 & 8*).

```
    # Step 5 - Remove a new item from the list/Table
    elif (strChoice.strip() == '3'):
        # TODO: Add Code Here
        strTask = str(input('What task would you like to remove? '))
        for row in lstTable:
            if row['Task'].lower() == strTask.lower():
                lstTable.remove(row)
                print('row removed')
                break
                # Without break statement will print for every row in instead of single message
            else:
                print('data not found in row')
        continue
```

Figure 7: Option 3 (remove row from memory for loop)

```
The current data is:
Task:                                   Priority:
Sweep                                   2
laundry                                 5
dishes                                  4
Press ENTER to continue3

    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

What task would you like to remove? sweep
row removed

    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

The current data is:
Task:                                   Priority:
laundry                                 5
dishes                                  4
Press ENTER to continue
```

Figure 8: Option 3 (user interface)

Option 4 and option 5 were treated the same way as option 3 in assignment 04, except the write to file and program exit were split into two separate options (*figure 9 & 10*).

```
    # Step 6 - Save tasks to the ToDoToDoList.txt file
    elif (strChoice.strip() == '4'):
        # TODO: Add Code Here
        objFile = open('ToDoList.txt', 'w')
        for row in lstTable:
            objFile.write(str(row['Task']) + ',' + str(row['Priority']) + '\n')
        objFile.close()
        print(' Data was saved to file')
        continue
    # Step 7 - Exit program
    elif (strChoice.strip() == '5'):
        # TODO: Add Code Here
        print('''
            Thank you for using our services.
            Have a nice day
            ''')
        break  # and Exit the program
```

Figure 9: Option 4 & 5 (save memory to file and exit program respectively)

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

 Data was saved to file

    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


            Thank you for using our services.
            Have a nice day


Process finished with exit code 0
```

Figure 10: Option 4 & 5 (user interface)

Conclusion

In this paper we will built upon the knowledge we gained in module 04 where we created and saved a list to a file, using a dictionary instead of a list format, and include the ability to delete data from memory. Please reference *figures 11 & 12* to see program demonstrated in Command Prompt.

Command Prompt

```
Microsoft Windows [Version 10.0.19044.2486]
(c) Microsoft Corporation. All rights reserved.

C:\Users\nezum>cd C:\_PythonClass\Assignment05

C:\_PythonClass\Assignment05>Python C:\_PythonClass\Assignment05\Assigment05.py
Task:                                   Priority:
laundry                                 5
dishes                                  4
Clean dog area                          3
^Data currently in file

    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

Type in a task and priority from 1 (low) to 5 (high)
Enter a task: mop
Enter a priority: 8
Please enter a value between 1 and 5
Enter a priority: 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] - 1

The current data is:
Task:                                   Priority:
laundry                                 5
dishes                                  4
Clean dog area                          3
mop                                     1
Press ENTER to continue

    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

What task would you like to remove? clean dog ArEa
data not found in row
data not found in row
row removed
```

Figure 11: Program running in Command Prompt 1/2

Command Prompt

```
    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

The current data is:
Task:                                   Priority:
laundry                                 5
dishes                                  4
mop                                     1
Press ENTER to continue

    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 was saved to file

    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


           Thank you for using our services.
           Have a nice day


C:\_PythonClass\Assignment05>
```

Figure 12: Program running in Command Prompt 2/2