Nasser Saber

November 13, 2021

Foundations of Programming, Python

Assignment 05

Creating a Python Program for Managing an Inventory List using Dictionaries

# Introduction

# In this assignment, I will explain the steps I used to create a Python Script that provides the user with menu options to choose from and performs the corresponding tasks following the user input. This script mainly involves working with lists, dictionaries, loops, writing to and reading from text files as well as some related methods.

# Creating the Script

I started creating a Python script in Spyder by adding code to CDInventory\_Starter.py. After declaring the variables, a while loop is added and the user is prompted to make a choice from the menu options. If the choice is ‘x’, the program exits. Once the user has made a choice to load from the file (existing data), a try – except method is used to avoid program break in the case that no txt file exists (no data has been added yet).

The following block is where the user input data is added to a dictionary table. Next block performs the unpacking of the table rows (dictionaries) using values() in order to enable data display as per user’s choice. In the following block, the user is asked to enter the ID of the CD (row) to be deleted and the corresponding dictionary is accessed via the ID key and removed. Finally, the data table is saved to CDInventory.txt if the user chooses to do so.

# Testing the Script

I ran the script in Spyder first. It runs well by asking the user to make the menu choice and performs the expected action based on the chosen menu number as shown in Figure 1.

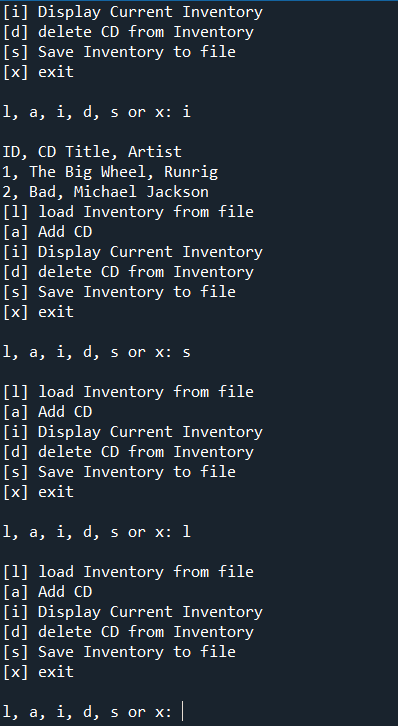


Figure Result of running the AddressBook.py script in Spyder

I then ran the same script in the terminal window, which returned the same results as expected (Figure 2).

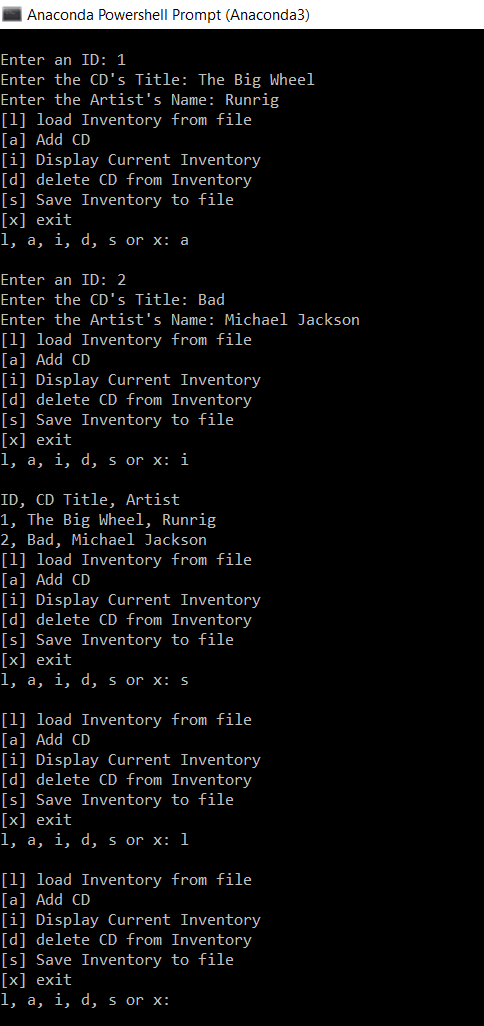


Figure – Result of running the AddressBook.py script in terminal window

Finally, I located the “CDInventory.txt” in the folder and checked that the user inputs were successfully saved in there (Figure 3).

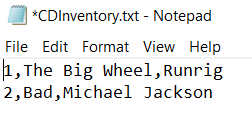


Figure Contents of AddressBook.txt

# Summary

In this assignment, I practiced creating a python program in Spyder, which asks user to choose from several menu options, gets the user input data based on the chosen option and allocates it to a list table using dictionaries, displays and saves the data to a text file and loads data from the file. I tested the script successfully in both Spyder and terminal window.

# Appendix

GitHub

CDInventory.py

1. *#------------------------------------------#*
2. *# Title: CDInventory.py*
3. *# Desc: Assignment 05*
4. *# Change Log: (Who, When, What)*
5. ***# DBiesinger, 2030-Jan-01, Created File***
6. *# NSaber, 2021-Nov-13, Added Functionality*
7. *#------------------------------------------#*
9. *# Declare variabls*
11. strChoice = '' *# User input*
12. lstTbl = [] *# list of lists to hold data*
13. *# TODO replace list of lists with list of dicts*
14. dicRow = [] *# list of data row*
15. **strFileName = 'CDInventory.txt' *# data storage file***
16. objFile = None *# file object*
18. *# Get user Input*
19. **print**('The Magic CD Inventory**\n**')
20. **while True:**
21. *# 1. Display menu allowing the user to choose:*
22. **print**('[l] load Inventory from file**\n**[a] Add CD**\n**[i] Display Current Inventory')
23. **print**('[d] delete CD from Inventory**\n**[s] Save Inventory to file**\n**[x] exit')
24. strChoice = input('l, a, i, d, s or x: ').lower() *# convert choice to lower case at time of input*
25. **print()**
27. **if** strChoice == 'x':
28. *# 5. Exit the program if the user chooses so*
29. **break**
30. **if strChoice == 'l':**
31. *# TODO Add the functionality of loading existing data*
32. **try**:
33. lstTbl.clear()
34. objFile = open(strFileName, 'r')
35. **for row in objFile:**
36. lstRow = row.strip().split(',')
37. dicRow = {'ID': int(lstRow[0]), 'CD title': lstRow[1], 'Artist Name': lstRow[2]}
38. lstTbl.append(dicRow)
39. objFile.close()
40. **except:**
41. **print**('CD Inventory has not been created yet - no file to load inventory**\n**')

44. **elif** strChoice == 'a': *# no elif necessary, as this code is only reached if strChoice is not 'exit'*
45. ***# 2. Add data to the table (2d-list) each time the user wants to add data***
46. strID = input('Enter an ID: ')
47. strTitle = input('Enter the CD**\'**s Title: ')
48. strArtist = input('Enter the Artist**\'**s Name: ')
49. intID = int(strID)
50. **dicRow = {'ID': intID, 'CD title': strTitle, 'Artist Name': strArtist}**
51. lstTbl.append(dicRow)
53. **elif** strChoice == 'i':
54. *# 3. Display the current data to the user each time the user wants to display the data*
55. **print('ID, CD Title, Artist')**
56. **for** row **in** lstTbl:
57. **print**(\*row.values(), sep = ', ')
59. **elif** strChoice == 'd':
60. ***# TODO Add functionality of deleting an entry***
61. intDel = int(input('Enter the ID to be deleted: '))
62. **for** row **in** lstTbl:
63. **if** row['ID'] == intDel:
64. lstTbl.remove(row)
66. **elif** strChoice == 's':
67. *# 4. Save the data to a text file CDInventory.txt if the user chooses so*
68. objFile = open(strFileName, 'a')
69. **for** row **in** lstTbl:
70. **strRow = ''**
71. **for** key, val **in** row.items():
72. strRow += str(val) + ','
73. strRow = strRow[:-1] + '**\n**'
74. objFile.write(strRow)
75. **objFile.close()**
76. **else**:
77. **print**('Please choose either l, a, i, d, s or x!')