Tom Williams

Date: 08/13/2020

IT FDN 110: Introduction to Programming (Python)

Assignment\_05

CD Inventory version 2.0

# Introduction

The goal of the assignment is to create a version of the CD Inventory program that allows the user to enter CD data, view the current inventory, save data to a CDInventory.txt data file, load the inventory file into memory, and delete items from inventory. Successful completion requires building and working with dictionaries as inventory items.

# Dictionaries

Having done a little work with JSON and NOSQL databases I am somewhat familiar with key-value pairs but have not used them extensively. For me the most challenging part of the assignment was applying the techniques from the book[[1]](#footnote-1) and tutorial[[2]](#footnote-2), since they mostly used a single dictionary, while this assignment required using multiple dictionaries as rows in a List.

# GitHub

I am familiar with the concept of source control from working in the software industry but have only used GitHub to access documents and examples, never to check in code.

# CD Inventory Version 2.0

The instructions were to start with the provided solution to Assignment4 and adapt it to use dictionaries as the inner loops. There were several clearly marked TODO sections (I like the exclamation point icon these produce) but all sections required modification. My program seems to run successfully and meet expectations.

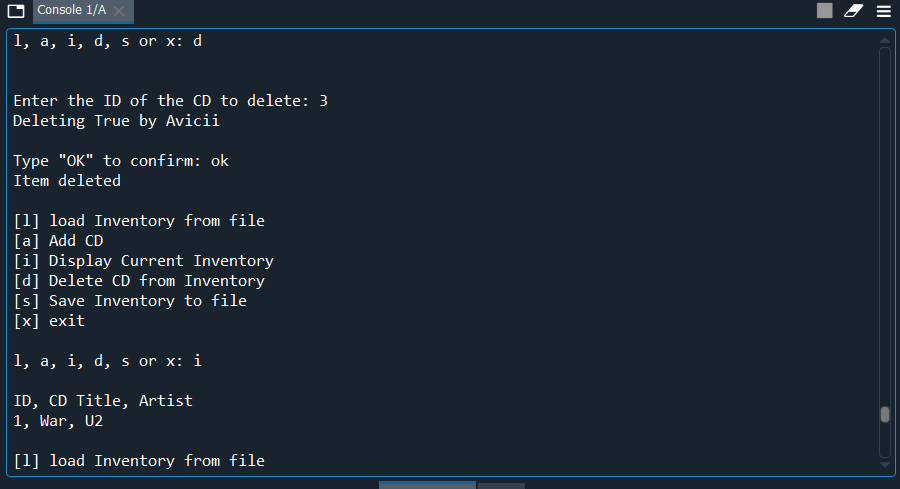


Figure : Spyder Console

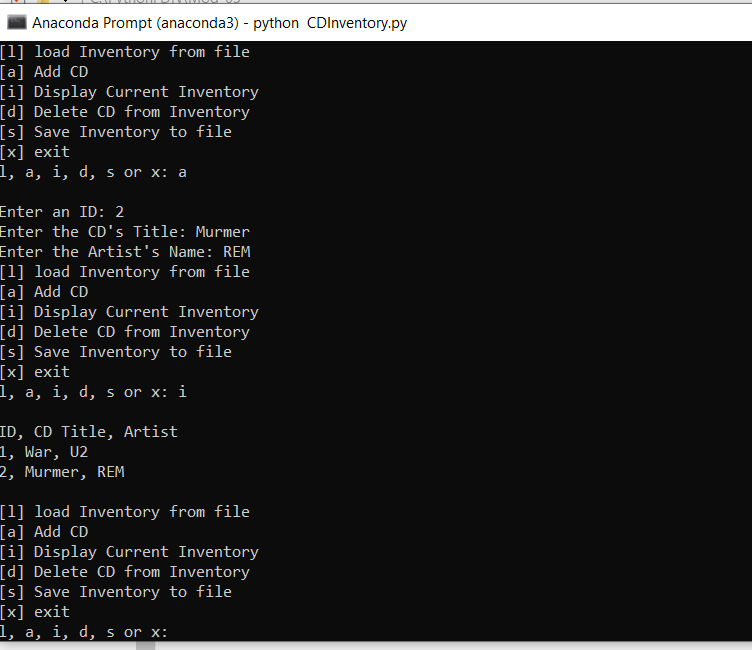


Figure : Anaconda Console

## Adapting Existing Code to use Dictionaries

Changing the Add, Display, and Save modules to use Dictionaries was fairly straightforward. The techniques used in Lab05\_B were mostly sufficient. I did choose to use the .format syntax that had caused me trouble in Assignment04 and was able to do a more straightforward implementation.

1. elif strChoice == 'i':
2. # 3. Display the current data to the user each time the user wants to display the data
3. print('ID, CD Title, Artist')
4. for row in lstTbl:
5. print('{}, {}, {}'.format(row['id'],row['title'], row['artist']))
6. print()

Listing

## Reading from File

The main challenge in the File Read was figuring out how to read the string data into a dictionary. It wasn’t completely clear if that was necessary but I decided it made sense. Combining the row indexing with the dictionary key names was an effective technique.

1. lstTbl.clear()
2. objFile = open(strFileName, 'r')
3. for row in objFile:
4. lstRow = row.strip().split(',')
5. dicRow = {'id':lstRow[0],'title':lstRow[1], 'artist':lstRow[2]} # create rows as dictionaries
6. lstTbl.append(dicRow)
7. objFile.close()
8. print('Inventory loaded from file.')
9. print()

Listing

## Deleting an Item

By far the most difficult section was the deletion, and my solution, while providing the needed functionality, may be more complex than necessary. It seemed that the delete needed to work whether the user was working from the list already loaded in memory, or chose delete as the first command of a session, so I wanted to check to see if the inventory existed and, if not, load from file. This seemed an idea use case for a function, but we were told not to use them. My very inelegant solution was simply to reuse the load code.

The next challenge was finding the correct item to delete based on user input. I decided that rather than just deleting the row based on a single entry, I wanted to provide more a verbose description for the user. This required me to both choose the correct dictionary to delete and to select elements from within that particular dictionary. This was very difficult for me to achieve and I look forward to seeing a more elegant solution.

1. strID = input('Enter the ID of the CD to delete: ')
2. dicDel = {} # Initialize empty dictionary to hold delete target
3. # loop through dictionaries to find delete target by matching ID
4. for dic in lstTbl:
5. if dic['id'] == strID:
6. dicDel = dic # hold item to be deleted
7. # Access dictionary items to print confirmation message
8. print ('Deleting ' + dicDel['title']+' by '+dicDel['artist'])
9. strDel = input('Type "OK" to confirm: ')
10. if strDel.lower() != 'ok':
11. print('')
12. break
13. elif strDel.lower() == 'ok':
14. lstTbl.remove(dic)
15. print('Item deleted')
16. print()
17. if dicDel == {}: # if no matching ID found print message
18. print('Sorry no CD with that ID in inventory.')
19. print()

Listing

I am disappointed I couldn’t figure out how to use the get() command as part of my solution. The two dimensionality was a barrier.

# Summary

The goal of the assignment was to create a version of the CD Inventory program that allows the user to enter CD data, view the current inventory, save data to a CDInventory.txt data file, load the inventory file into memory, and delete items from inventory. The pedagogical goal was to use dictionaries, and while I struggled at times, I was able to use them in several ways throughout my solution.

# Appendix

Full Code

1. # Declare variabls
2. strChoice = '' # User input
3. lstTbl = [] # list of lists to hold data
4. dicRow = {} # dictionary data row
5. strFileName = 'CDInventory.txt' # data storage file
6. objFile = None # file object
7. # Get user Input
8. print('The Magic CD Inventory\n')
9. while True:
10. # 1. Display menu allowing the user to choose:
11. print('[l] load Inventory from file\n[a] Add CD\n[i] Display Current Inventory')
12. print('[d] Delete CD from Inventory\n[s] Save Inventory to file\n[x] exit')
13. strChoice = input('l, a, i, d, s or x: ').lower() # convert choice to lower case at time of input
14. print()
15. if strChoice == 'x':
16. # 5. Exit the program if the user chooses so
17. print('Goodbye!')
18. break
20. if strChoice == 'l':
21. # 1. load existing data from text file CDInventory.txt if the user chooses so
22. lstTbl.clear()
23. objFile = open(strFileName, 'r')
24. for row in objFile:
25. lstRow = row.strip().split(',')
26. dicRow = {'id':lstRow[0],'title':lstRow[1], 'artist':lstRow[2]} # create rows as dictionaries
27. lstTbl.append(dicRow)
28. objFile.close()
29. print('Inventory loaded from file.')
30. print()
32. elif strChoice == 'a': # no elif necessary, as this code is only reached if strChoice is not 'exit'
33. # 2. Add data to the table (2d-list) each time the user wants to add data
34. strID = input('Enter an ID: ')
35. strTitle = input('Enter the CD\'s Title: ')
36. strArtist = input('Enter the Artist\'s Name: ')
37. dicRow = {'id': strID,'title':strTitle,'artist':strArtist} # create dictionary
38. lstTbl.append(dicRow) # add dictionary to list as row
40. elif strChoice == 'i':
41. # 3. Display the current data to the user each time the user wants to display the data
42. print('ID, CD Title, Artist')
43. for row in lstTbl:
44. print('{}, {}, {}'.format(row['id'],row['title'], row['artist']))
45. print()
47. elif strChoice == 'd':
48. # 4. Delete an entry from the inventory if the user chooses so
49. # Check if inventory is in memory
50. if lstTbl == []:
51. # If not, load it (this should be a function call but instead repeating load code)
52. objFile = open(strFileName, 'r')
53. for row in objFile:
54. lstRow = row.strip().split(',')
55. dicRow = {'id':lstRow[0],'title':lstRow[1], 'artist':lstRow[2]}
56. lstTbl.append(dicRow)
57. objFile.close()
58. # Ask user for ID of CD to delete
59. strID = input('Enter the ID of the CD to delete: ')
60. dicDel = {} # Initialize empty dictionary to hold delete target
61. # loop through dictionaries to find delete target by matching ID
62. for dic in lstTbl:
63. if dic['id'] == strID:
64. dicDel = dic # hold item to be deleted
65. # Access dictionary items to print confirmation message
66. print ('Deleting ' + dicDel['title']+' by '+dicDel['artist'])
67. strDel = input('Type "OK" to confirm: ')
68. if strDel.lower() != 'ok':
69. print('')
70. break
71. elif strDel.lower() == 'ok':
72. lstTbl.remove(dic)
73. print('Item deleted')
74. print()
75. if dicDel == {}: # if no matching ID found print message
76. print('Sorry no CD with that ID in inventory.')
77. print()
79. elif strChoice == 's':
80. # 4. Save the data to a text file CDInventory.txt if the user chooses so
81. objFile = open(strFileName, 'w')
82. for row in lstTbl:
83. strRow = ''
84. for item in row.values():
85. strRow += str(item) + ','
86. strRow = strRow[:-1] + '\n'
87. objFile.write(strRow)
88. objFile.close()
89. print('Inventory saved to file.')
90. print()
92. else:
93. print('Please choose either l, a, i, d, s or x!')

Listing

1. Dawson, Python Programming for the Absolute Beginner, pp. 140-148 [↑](#footnote-ref-1)
2. <https://realpython.com/python-dicts> last accessed 08/13/2020 [↑](#footnote-ref-2)