

< Weimin Dang >
< Feb-21-2020>
< IT FDN 100 Winter 2020 >
< Assignment05>

Dictionaries and Its Methods

Introduction

Assignment 5 aims at getting me to practice the following knowledge points:

1. Understand the difference between dictionaries and lists
2. Understand a few dictionary methods such as working with keys, values, key-value pairs, and add or delete them or an entire library
3. Review while and for loops, and if & elif statement
4. Practice working with other people's code
5. Practice peer review, and
6. Practice using github

Developing the Code

Starter code

Below summarizes the initial assessment of the starter code provided:

1. Declare variables
 - a. Minor tweaks to redefine some variables and maybe new variables to add here
2. Get user Input
 - a. Display menu allowing the user to choose: 'load inventory from file', 'Add CD', 'display current inventory', 'delete CD from inventory', 'Save inventory to file' and 'exit'
 - i. no or minimum work required
 - b. Exit the program if the user chooses so
 - i. no or minimum work required
 - c. Add the functionality of loading existing data
 - i. new code will be needed
 - d. Add data to the table each time the user wants to add data
 - i. minor tweaks to accommodate list of dictionaries
 - e. Display the current data to user
 - i. Minor tweaks to accommodate list of dictionaries
 - f. Add the functionality of deleting an entry
 - i. New code will be needed
 - g. Save the data to a txt file
 - i. Minor tweaks may be needed
 - h. What happens if invalid choices
 - i. No or minimum work required

Development of the Modified Code:

1. First, I added to the beginning of the code to import the os.path method for future uses to check if no existing inventory is created

```
import os.path
```

2. Next, in the “declare variable” step, in addition to modify the `lstRow[]` list variable to `lstRow{}` library variable, I added a `savedChoice[]` list variable, which will be used later for switching the mode of saving data to file.

```
12 # Declare global variables
13
14 savedChoice = [] # list of choice user has input
15 strChoice = '' # User input
16 lstTbl = [] # list of Dicts to hold data
17 lstRow = {} # list of data row
18 strFileName = 'CDInventory.txt' # data storage file
19 objFile = None # file object
20
```

3. Next, in the “get user input” section:

- a. There is no change to the menu display and choice of exit section of the code.

```
21 # Get user input
22 print('The Magic CD Inventory\n')
23 while True:
24     # 1. Display menu allowing the user to choose:
25     print('[l] Load Inventory from file\n[a] Add CD\n[i] Display Current Inventory')
26     print('[d] Delete CD from Inventory\n[s] Save Inventory to file\n[x] exit')
27     strChoice = input('l, a, i, d, s or x: ').lower() # convert choice to lower case
28     # at time of input
29     print()
30     if strChoice == 'x':
31         # 5. Exit the program if the user chooses so
32         break
33
```

- b. In the added functionality of loading data from existing saved data, first I used the `os.path.exists()` function to check in case there isn't any saved data, the program will continue by “pass”. If the data file does exist, open the data file in reading mode 'r'.

Then line 38 clears out any temporary data in `lstTbl` saved in memory, this is necessary as otherwise saved data will append to `lstTbl`, then when it is later saved, duplicate records will be created. Just think of playing a game, and loading a saved play, it should just load not append past saves to current status.

Then line 39-45 cycle through the saved data, and add each of them as matching values of the keys of libraries of CDs, which are then appended to the list of libraries `lstTbl`.

```
34 if strChoice == 'l':
35     # Load existing inventory data saved
36     if os.path.exists('CDInventory.txt'): # if file doesn't exist, pass and continue
37         objFile = open('CDInventory.txt', 'r')
38         lstTbl = [] # clear out temporary data saved in memory, otherwise will append saved data
39         # to what is in the memory.
40         for row in objFile: # load the saved CDs and cycle through by each row of the data
41             savedID = row.strip().split(',') # cycle through each comma separated item of each row
42             lstRow = {'ID': savedID[0], 'Title': savedID[1], 'Artist': savedID[2]} # create the 2D-
43             # dictionary, values of the matching keys are from the saved data
44             lstTbl.append(lstRow) # create the list of dictionaries
45             print(lstTbl)
46             objFile.close()
47             print()
48             savedChoice.append(strChoice) # record the user has selected loading from the saved file
49             pass
```

Line 46 is to record if the user has selected the loading choice, which will determine how the new data will be saved. More details will be discussed in the saving to file section.

- c. Next in the add data section, first the comment related to no elif needed for line 49 was removed as it is incorrectly carried over from the old code for assignment 4. Then on line 55, the code was modified to fit for dictionary.

```
49     elif strChoice == 'a':
50         # 2. Add data to the table (2d-list) each time the user wants to add data
51         strID = input('Enter an ID: ')
52         strTitle = input('Enter the CD\'s Title: ')
53         strArtist = input('Enter the Artist\'s Name: ')
54         intID = int(strID)
55         lstRow = {'ID':intID, 'Title':strTitle, 'Artist':strArtist} # changed from list to dictionary
56         lstTbl.append(lstRow)
57         print()
```

- d. Next is the code for the added functionality of deleting an entire entry by user specified CD ID. First, a user input was defined on line 68. Then, a counter variable delIdx was created to identify the matching index number of the CD to be deleted. Its initial value starts from -1, its value should return the index or position of the CD in the entire list when the program check user entered CD ID through each CD library of the list. For example, if user entered CD ID matches the very first CD in the inventory, which has a position of 0, and that is the delIdx value when the program finish scanning through the first row. Similarly, if the user entered CD ID matched the 4th CD in the inventory, which has a position of 3, and the delIdx value will be 3. Then this matching positioned CD is instructed to be deleted.

If however, user entered CD ID is not found a match, or the CD inventory is empty, after scanning through every CD library, delIdx value should match one less than the number of items in the lstTbl (the list of the CD libraries) which is calculated by the len() function. Again, this is due to index starts from position 0, but len() returns total number of items.

```
66     elif strChoice == 'd':
67         # Delete an entire entry of a CD, i.e., ID, Title, and Artist based on CD ID entered
68         delID = input('Which CD ID do you want to delete?: ')
69         delIdx = -1 # create an index variable to identify the matching index number of the CD to be
        deleted, it starts from -1, as when the counter starts in the for loop, its value becomes 0, which is
        the index number of the first CD in the list of CD library. The counter will continue until the match
        is identified.
70         for row in lstTbl:
71             delIdx += 1
72             if delID in row.values():
73                 del lstTbl[delIdx]
74                 print('CD', delID, 'is deleted.\n')
75             elif delIdx == len(lstTbl)-1: # if no match is found after cycling through the entire
        inventory list, it is either no matching CD is found or the list is empty.
76                 print('CD', delID, 'doesn\'t exist' 'OR The inventory is empty.\n')
77
```

- e. Next for saving the data, initially I didn't think material change would be required, however, adding a loading function will change how the data should be saved. For example, if a user starts the program and loads saved data, then made some changes. In order to capture all the changes, the saving mode should be using 'w' to overwrite previously saved data. However, if a user starts the program and directly adds a new CD then wants to save the change. This can only be performed using the 'a' mode to append new data to the previously saved data.

```
78 elif strChoice == 's':
79     # 4. Save the data to a text file CDInventory.txt if the user chooses so
80     if len(savedChoice) > 0: # if user has selected loading from the saved data, when save new
data, in order not to create duplicate records, using w mode of open to overwrite the existing file
content
81         objFile = open('CDInventory.txt', 'w')
82         for row in lstTbl:
83             strRow = ''
84             for value in row.values():
85                 strRow += str(value) + ','
86             strRow = strRow[:-1] + '\n'
87             objFile.write(strRow)
88             objFile.close()
89     else: # if user has not yet selected loading from the saved data, only new entries will be
added and no duplicate records will be create.
90         objFile = open('CDInventory.txt', 'a')
91         for row in lstTbl:
92             strRow = ''
93             for value in row.values():
94                 strRow += str(value) + ','
95             strRow = strRow[:-1] + '\n'
96             objFile.write(strRow)
97             objFile.close()
```

- f. Finally if the user input an invalid choice, the program will tell the user to reselect.

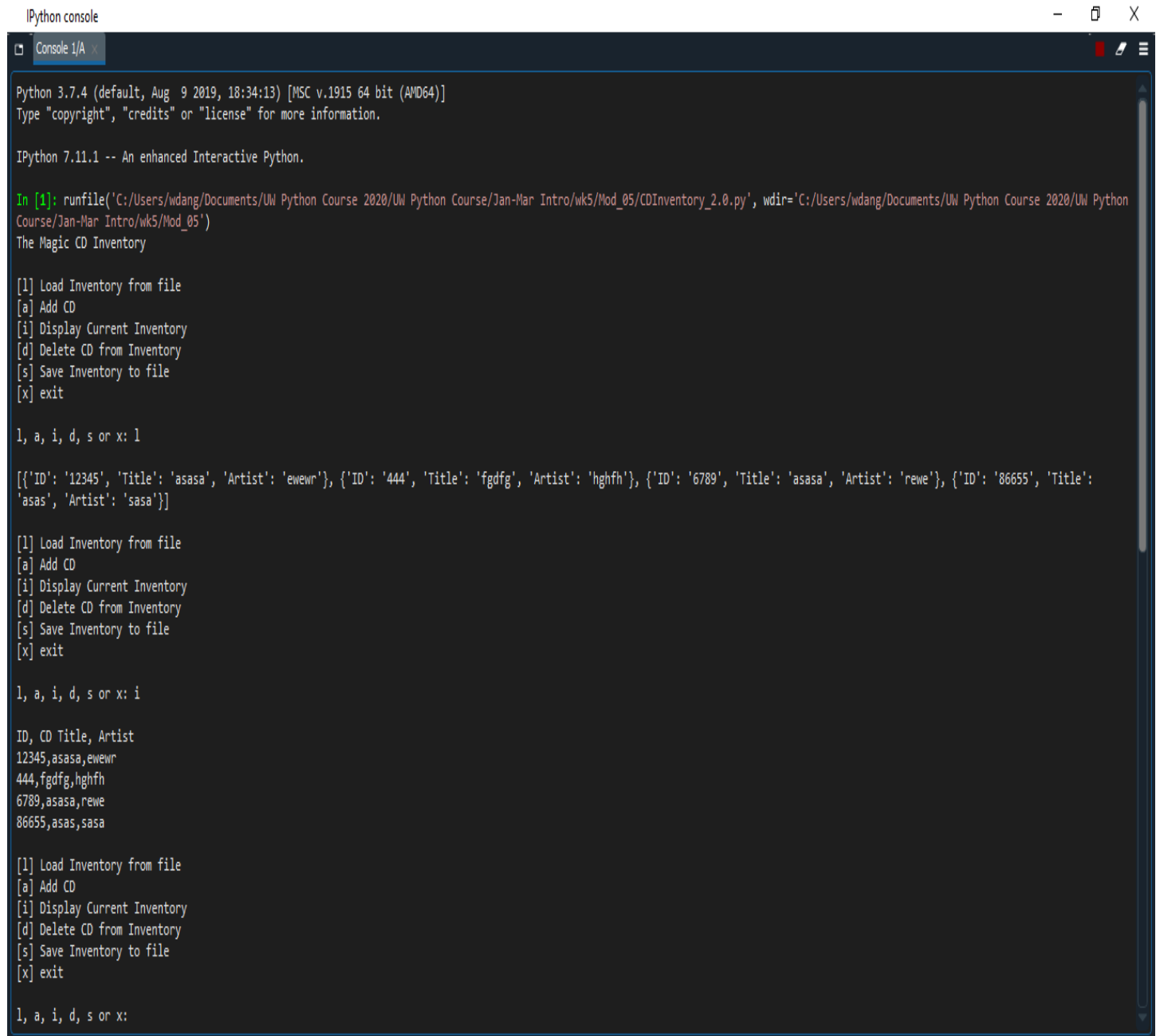
```
98
99 else:
100     print('Please choose either l, a, i, d, s or x!')
101
```

Summary

Running the script in Spyder:

Figure 1 below shows when user starts the program and loaded the existing inventory, also display it by showing the CD values only i.e. the ID, Title, and Artist.

Figure 1



```
Python 3.7.4 (default, Aug 9 2019, 18:34:13) [MSC v.1915 64 bit (AMD64)]
Type "copyright", "credits" or "license()" for more information.

IPython 7.11.1 -- An enhanced Interactive Python.

In [1]: runfile('C:/Users/wdang/Documents/UW Python Course 2020/UW Python Course/Jan-Mar Intro/wk5/Mod_05/CDInventory_2.0.py', wdir='C:/Users/wdang/Documents/UW Python Course 2020/UW Python Course/Jan-Mar Intro/wk5/Mod_05')
The Magic CD Inventory

[1] Load Inventory from file
[a] Add CD
[i] Display Current Inventory
[d] Delete CD from Inventory
[s] Save Inventory to file
[x] exit

l, a, i, d, s or x: l

[{'ID': '12345', 'Title': 'asasa', 'Artist': 'ewewr'}, {'ID': '444', 'Title': 'fgdfg', 'Artist': 'hghfh'}, {'ID': '6789', 'Title': 'asasa', 'Artist': 'rewe'}, {'ID': '86655', 'Title': 'asasa', 'Artist': 'sasa'}]

[1] Load Inventory from file
[a] Add CD
[i] Display Current Inventory
[d] Delete CD from Inventory
[s] Save Inventory to file
[x] exit

l, a, i, d, s or x: i

ID, CD Title, Artist
12345,asasa,ewewr
444,fgdfg,hghfh
6789,asasa,rewe
86655,asasa,sasa

[1] Load Inventory from file
[a] Add CD
[i] Display Current Inventory
[d] Delete CD from Inventory
[s] Save Inventory to file
[x] exit

l, a, i, d, s or x:
```

Figure 2 shows a new CD 0001, Bad, MJ is added, and an existing CD 444 is deleted, and the new view of the cd inventory list.

Figure 2

```
l, a, i, d, s or x: a

Enter an ID: 0001

Enter the CD's Title: Bad

Enter the Artist's Name: MJ

[l] Load Inventory from file
[a] Add CD
[i] Display Current Inventory
[d] Delete CD from Inventory
[s] Save Inventory to file
[x] exit

l, a, i, d, s or x: d

Which CD ID do you want to delete?: 444
CD 444 is deleted

CD 444 deosn't existOR The inventory is empty

[l] Load Inventory from file
[a] Add CD
[i] Display Current Inventory
[d] Delete CD from Inventory
[s] Save Inventory to file
[x] exit

l, a, i, d, s or x: i

ID, CD Title, Artist
12345,asasa,ewewr
6789,asasa,rew
86655,asas,sasa
1,Bad,MJ
```

Figure 3 below shows the new data is saved and using “L” and “I” menu option to confirm the new data is saved as expected. Then user exited the program.

Figure 3

```
l, a, i, d, s or x: s

[l] Load Inventory from file
[a] Add CD
[i] Display Current Inventory
[d] Delete CD from Inventory
[s] Save Inventory to file
[x] exit

l, a, i, d, s or x: l

[{'ID': '12345', 'Title': 'asasa', 'Artist': 'ewe
'Artist': 'MJ'}]

[l] Load Inventory from file
[a] Add CD
[i] Display Current Inventory
[d] Delete CD from Inventory
[s] Save Inventory to file
[x] exit

l, a, i, d, s or x: i

ID, CD Title, Artist
12345,asasa,ewewr
6789,asasa,rew
86655,asas,sasa
1,Bad,MJ

[l] Load Inventory from file
[a] Add CD
[i] Display Current Inventory
[d] Delete CD from Inventory
[s] Save Inventory to file
[x] exit

l, a, i, d, s or x: x

In [2]: |
```

Figure 4 and 5 shows the code running in the terminal.

Figure 4

```
Anaconda Prompt (anaconda3)
(base) C:\Users\wdang>python CDInventory_2.0.py
The Magic CD Inventory

[l] Load Inventory from file
[a] Add CD
[i] Display Current Inventory
[d] Delete CD from Inventory
[s] Save Inventory to file
[x] exit
l, a, i, d, s or x: l

[{'ID': '12345', 'Title': 'asasa', 'Artist': 'ewewr'}, {'ID': '6789', 'Title': 'asasa', 'Artist': 'rewe'}, {'ID': '86655', 'Title': 'asas', 'Artist': 'sasa'}, {'ID': '1', 'Title': 'Bad', 'Artist': 'MJ'}]

[l] Load Inventory from file
[a] Add CD
[i] Display Current Inventory
[d] Delete CD from Inventory
[s] Save Inventory to file
[x] exit
l, a, i, d, s or x: i

ID, CD Title, Artist
12345,asasa,ewewr
6789,asasa,rewe
86655,asas,sasa
1,Bad,MJ
```

Figure 5

```
Anaconda Prompt (anaconda3)
[l] Load Inventory from file
[a] Add CD
[i] Display Current Inventory
[d] Delete CD from Inventory
[s] Save Inventory to file
[x] exit
l, a, i, d, s or x: d

Which CD ID do you want to delete?: 12345
CD 12345 is deleted

CD 12345 deosn't existOR The inventory is empty

[l] Load Inventory from file
[a] Add CD
[i] Display Current Inventory
[d] Delete CD from Inventory
[s] Save Inventory to file
[x] exit
l, a, i, d, s or x: s

[l] Load Inventory from file
[a] Add CD
[i] Display Current Inventory
[d] Delete CD from Inventory
[s] Save Inventory to file
[x] exit
l, a, i, d, s or x: l

[{'ID': '6789', 'Title': 'asasa', 'Artist': 'rewe'}, {'ID': '86655', 'Title': 'asas', 'Artist': 'sasa'}, {'ID': '1', 'Title': 'Bad', 'Artist': 'MJ'}]

[l] Load Inventory from file
[a] Add CD
[i] Display Current Inventory
[d] Delete CD from Inventory
[s] Save Inventory to file
[x] exit
l, a, i, d, s or x: i

ID, CD Title, Artist
6789,asasa,rewe
86655,asas,sasa
1,Bad,MJ
```


Lessons learned:

Debugging:

For a script similar to this assignment, it is very important to test the code by changing the sequence and selection of the menu options to see if everything works as expected.

When modification of code such as changing from list to dictionary, it can impact many other parts of the code.

Work from other's code:

Working from other's code is realistic work situation and can be both beneficial and challenging as I can learn from how others think of a solution for a problem, but can easily fall into traps or encounter issues especially when modifications and new functions are needed.

Peer review:

Peer review is also a very good way of learning and help each other in the study community.

Github:

I have heard of Github many times and feel fortunate to start utilizing this great resource and platform.

Appendix – copy of the script

```
#-----#
# Title: CDInventory_2.0.py
# Desc: Script for Assignment 05
# Change Log: (Who, When, What)
# DBiesinger, 2030-Jan-01, Created File
# WDang, 2020-Feb-20, Modified File
#-----#

# use os.path to check if file exist
import os.path

# Declare glable variabls

savedChoice = [] # list of choice user has input
strChoice = '' # User input
lstTbl = [] # list of Dicts to hold data
lstRow = {} # list of data row
strFileName = 'CDInventory.txt' # data storage file
objFile = None # file object

# Get user Input
print('The Magic CD Inventory\n')
while True:
    # 1. Display menu allowing the user to choose:
    print('[l] Load Inventory from file\n[a] Add CD\n[i] Display Current Inventory')
    print('[d] Delete CD from Inventory\n[s] Save Inventory to file\n[x] exit')
    strChoice = input('l, a, i, d, s or x: ').lower() # convert choice to lower case at time of input
    print()

    if strChoice == 'x':
        # 5. Exit the program if the user chooses so
        break

    if strChoice == 'l':
        # Load existing inventory data saved
        if os.path.exists('CDInventory.txt'): #if file doesn't exist, pass and continue
            objFile = open('CDInventory.txt', 'r')
            lstTbl = [] #clear out temporary data saved in memory, otherwise will append saved data to what is in the
memory.
            for row in objFile: #load the saved CDs and cycle through by each row of the data
                savedID = row.strip().split(',') #cycle through each comma separated item of each row
                lstRow = {'ID':savedID[0], 'Title':savedID[1], 'Artist':savedID[2]} #create the 2D-dictionary, values of the
matching keys are from the saved data
                lstTbl.append(lstRow) # create the list of dictionaries
            print(lstTbl)
            objFile.close()
            print()
            savedChoice.append(strChoice) # record the user has selcted loading from the saved file
```

pass

elif strChoice == 'a':

2. Add data to the table (2d-list) each time the user wants to add data

strID = input('Enter an ID: ')

strTitle = input('Enter the CD\'s Title: ')

strArtist = input('Enter the Artist\'s Name: ')

intID = int(strID)

lstRow = {'ID':intID, 'Title':strTitle, 'Artist':strArtist} # changed from list to dictionary

lstTbl.append(lstRow)

print()

elif strChoice == 'i':

3. Display the current data to the user each time the user wants to display the data

print('ID, CD Title, Artist')

for row in lstTbl:

print(*row.values(), sep = ',') # use values() attribute to display only the values of the CD libraries

print()

elif strChoice == 'd':

Delete an entire entry of a CD, i.e., ID, Title, and Artist based on CD ID entered

delID = input('Which CD ID do you want to delete?: ')

delIdx = -1 # create an index variable to identify the matching index number of the CD to be deleted, it starts from -1, as when the counter starts in the for loop, its value becomes 0, which is the index number of the first CD in the list of CD library. The counter will continue until the match is identified.

for row in lstTbl:

delIdx +=1

if delID in row.values():

del lstTbl[delIdx]

print('CD', delID, 'is deleted \n')

elif delIdx == len(lstTbl)-1: # if no match is found after cycling through the entire inventory list, it is either no matching CD is found or the list is empty.

print('CD', delID, 'doesn\'t exist' 'OR The inventory is empty \n')

elif strChoice == 's':

4. Save the data to a text file CDInventory.txt if the user chooses so

if len(savedChoice) > 0: # if user has selected loading from the saved data, when save new data, in order not to create duplicate records, using w mode of open to overwrite the existing file content

objFile = open('CDInventory.txt', 'w')

for row in lstTbl:

strRow = ''

for value in row.values():

strRow += str(value) + ','

strRow = strRow[:-1] + '\n'

objFile.write(strRow)

objFile.close()

else: # if user has not yet selected loading from the saved data, only new entries will be added and no duplicate records will be create.

objFile = open('CDInventory.txt', 'a')

```
for row in lstTbl:
    strRow = ""
    for value in row.values():
        strRow += str(value) + ','
    strRow = strRow[:-1] + '\n'
    objFile.write(strRow)
objFile.close()
```

else:

```
print('Please choose either l, a, i, d, s or x!')
```