Maricha Friedman

August 8, 2021

IT FDN 110: Introduction to Programming (Python)

Assignment 5

CD Inventory Script with Dictionaries

# Introduction

This assignment uses the same script from the previous assignment to create a CD inventory, but required using a starter template and using a 2D list of dictionaries, not of lists. I was not able to complete all the tasks in this assignment as I would have liked, but the basic functionality required is there.

# Loading existing data

I opened the file with ‘r’ mode and used a for loop to bring in each separate line of text. This works fine if there is text in the .txt file, and I wrote some data in dictionary format matching the dictionary structure of the script to test this. However, I was not able to successfully pull that data into a variable in memory, I can only print each line. That means that the user isn’t able to delete or modify that data, only view it. I tried to use part of an example from the lab that read text and separated values by comma - row.strip().split(',') - but I was not able to separate each item and assign it to the ID, Title, and Artist variables to then add them to the dictionary as another row. I would like to figure out how to do this.

# Converting list to dictionary, appending data

I was able to change the 2D list of lists to a list of dictionaries simply by replacing lstRow with dicRow and adding keys for each value, then appending each dictionary row to the table:

**elif** strChoice **==** 'a'**:** # no elif necessary, as this code is only reached if strChoice is not 'exit'

# 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**)**

dicRow **=** **{**'id'**:** strID**,** 'CD Title'**:** strTitle**,** 'Artist Name'**:** strArtist**}**

lstTbl**.**append**(**dicRow**)**

Since this is part of a for loop that was already in the template, this is executed every time the user wants to enter a new CD and is stored in the memory of the program. For the next step, displaying in-memory data, I simply used a for loop of the table to print each row. I did not format the output though, so it is a default dictionary output. Since I was not able to bring in the data that may have been in the file, the display only shows what the user has entered in the current instance of the program.

# Deleting an entry – revisiting loops again

After a lot of trial and error and some Googling I was finally able to allow the user to delete an entry. I used the “geek\_translator.py” example from the textbook as a guide, but that example only includes how to delete a key and value pair from one dictionary, not a list of dictionaries. While testing that method, I found it did work, but I got an error that the “dictionary changed size during iteration” because I was only removing a key/value pair, and not the whole row. Furthermore, I was using a row loop evaluating rows, but ran into issues trying to use “if [search term] in row”, because since my rows were dictionaries, nothing matched even though it was clearly there. I’m still not entirely sure why this is the case, since this is the first time I’ve used dictionaries.

I had to use another for loop within that loop to iterate over each value in key/value pairs, since I was specifically looking for a match in values. This finally worked. I wanted to use “del” to delete the row, but ran into errors presumably because that is meant for removing items from a dictionary, not list items. This [forum post](https://stackoverflow.com/questions/33190779/how-to-delete-a-dictionary-from-a-list-of-dictionaries)[[1]](#footnote-1) on Stack Overflow that reminded me of simply using “remove” to remove an item from a list, which we have seen in previous modules. At this point I was able to identify and delete the correct row, but it was also printing my “else” statement for non-matches as many times as there were key/value pairs:

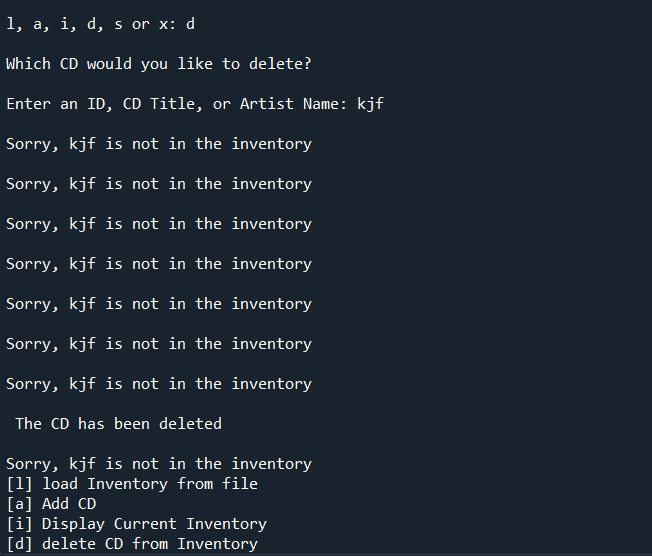


Figure - Loop output telling me each time a value didn't match the search term

Thanks to Google, I found [this site](https://www.geeksforgeeks.org/python-removing-dictionary-from-list-of-dictionaries/)[[2]](#footnote-2), which helped by reminding me of how I can use the length of a list to limit what happens in each iteration, so I added a counter = 0 outside of the loop, and then instead of printing the no-match notification in my “else” statement, I added 1 to the counter. Then, at the end of that process, I added an if statement that if the counter is equal to the length of the table (number of rows) multiplied by 3 (number of key/value pairs per row), then it could print the “not found” statement. Multiplying by 3 is admittedly an inelegant and non-scalable solution, but it worked for this assignment.

# Writing data to file

This last step took up a lot of my time on this assignment, and I still wasn’t able to get the outcome I wanted. Using the example from Listing 6 from the module, and the feedback from last week’s assignment, I tried to create a string per row of only the values in each dictionary (row), separated by commas. This sort of worked, except that I couldn’t separate the strings at each row. I could only add new lines after each separate value or keep all values from all rows in one long line in the document. I would like to figure out where I went wrong on this as well.

Rather than write a long comma-separate string to the file, I reverted to writing each row (dictionary in the list table) to the file on its own line.

# Summary

This assignment took more time than the previous ones for me because it built on things we’ve briefly seen in previous modules, and it’s not easy to remember them all and where they could be useful. While I had the starter template, it was difficult to adapt the script to using dictionaries. I am still trying to wrap my head around 2D tables, especially when working with loops. I’m also still struggling with loops in general and some of the logic around each step.

# Appendix

## Listing CDInventory.py

#------------------------------------------#

# Title: CDInventory.py

# Desc: Starter Script for Assignment 05

# Change Log: (Who, When, What)

# DBiesinger, 2030-Jan-01, Created File

# Maricha Friedman, Aug 8 2021, updated to complete assignment

#------------------------------------------#

# Declare variables

strChoice **=** '' # User input

lstTbl **=** **[]** # list of dictionaries to hold data

# TODO replace list of lists with list of dicts

dicRow **=** **{}** # dictionary 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'**:**

# TODO Add the functionality of loading existing data

objFile **=** **open(**strFileName**,** 'r'**)**

**for** line **in** objFile**:**

**print(**line**)**

**pass**

**elif** strChoice **==** 'a'**:** # no elif necessary, as this code is only reached if strChoice is not 'exit'

# 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**)**

dicRow **=** **{**'id'**:** strID**,** 'CD Title'**:** strTitle**,** 'Artist Name'**:** strArtist**}**

lstTbl**.**append**(**dicRow**)**

**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**)**

**elif** strChoice **==** 'd'**:**

# TODO Add functionality of deleting an entry

**print(**'Which CD would you like to delete?'**)**

delSrch **=** **str(input(**'Enter an ID, CD Title, or Artist Name: '**))**

counter **=** 0

**for** row **in** lstTbl**:**

**for** key**,** val **in** row**.**items**():**

**if** delSrch **in** val**:**

lstTbl**.**remove**(**row**)**

**print(**'\n The CD has been deleted'**)**

**continue**

**elif** delSrch **not** **in** val**:**

counter **+=** 1

**continue**

**if** counter **==** **(len(**lstTbl**)\***3**):**

**print(**'\nSorry,'**,** delSrch**,** 'is not in the inventory'**)**

**pass**

**elif** strChoice **==** 's'**:**

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

objFile **=** **open(**strFileName**,** 'a'**)**

**for** row **in** lstTbl**:**

strRow **=** **str(**row**)**

strRow **=** '\n' **+** strRow

objFile**.**write**(**strRow**)**

objFile**.**close**()**

**else:**

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

# Script running in Spyder

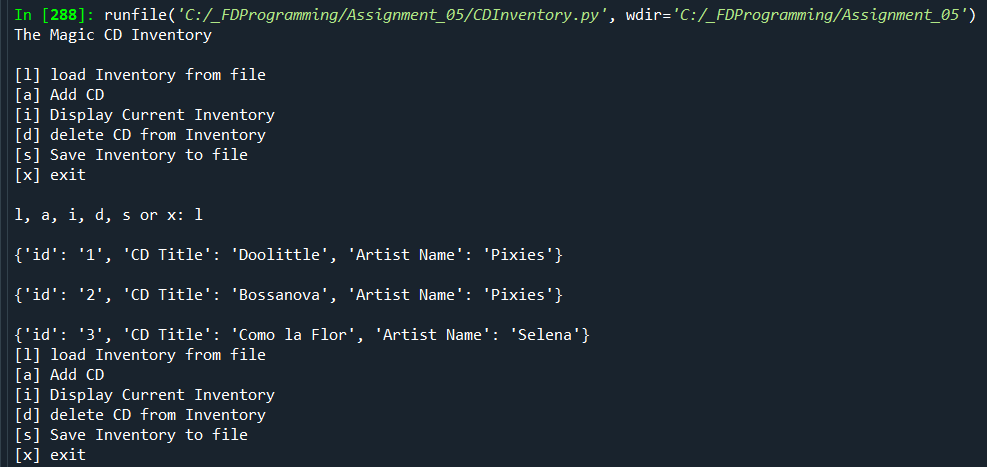
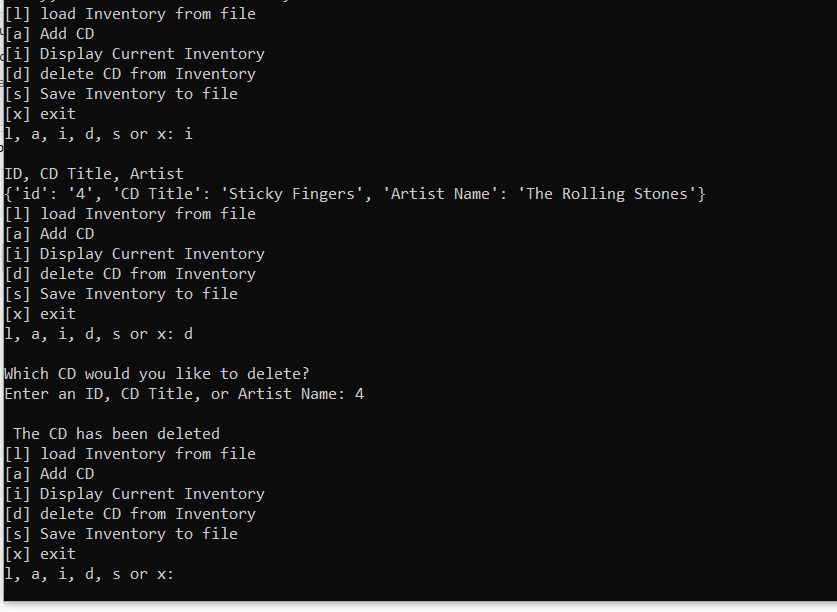


Figure - Script running in Spyder, step 1



Figure - Script running in Spyder - Step 2

# Script running in Terminal



# GitHub Repository

Here is the link for my repository for Assignment 5:

<https://github.com/marichaf/Assignment_05>

1. [https://stackoverflow.com/questions/33190779/how-to-delete-a-dictionary-from-a-list-of-dictionaries](https://stackoverflow.com/questions/33190779/how-to-delete-a-dictionary-from-a-list-of-dictionaries%20Retrieved%20August%208), Retrieved August 8, 2021 [↑](#footnote-ref-1)
2. <https://www.geeksforgeeks.org/python-removing-dictionary-from-list-of-dictionaries/> , Retrieved August 8, 2021. [↑](#footnote-ref-2)