James

2/27/2022

IT FDN 110 B: Introduction to Programming Python

Assignment 05

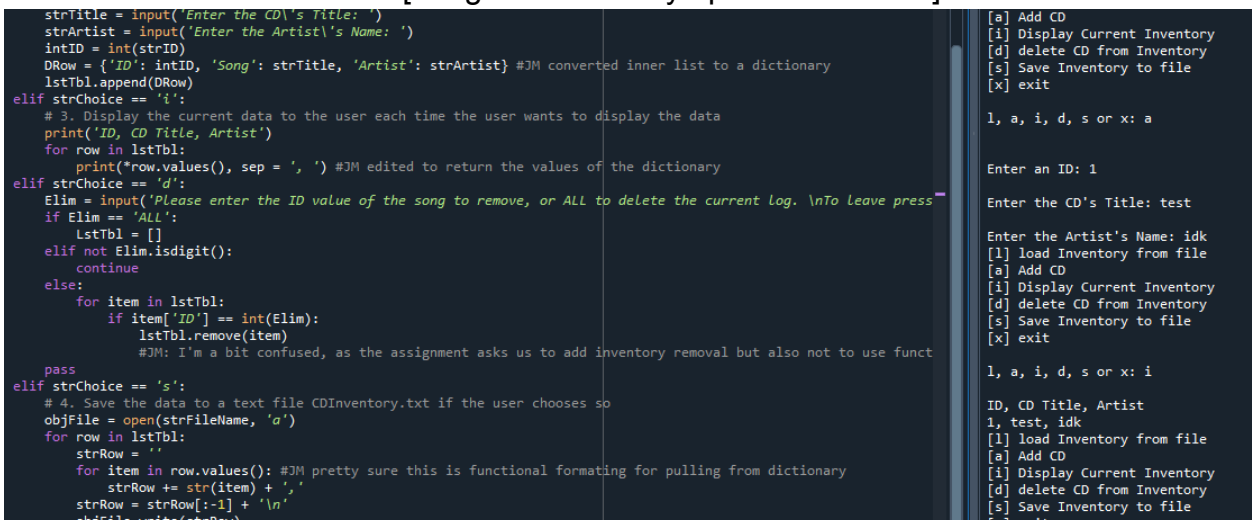
- Overview

This week, we delve into reading files and using dictionaries, with some side work in deleting specific table items.

- Updating to Dictionary

While converting a list entry into a dictionary isn't to difficult, getting python to properly extract data from a dictionary is a radically different format from lists and I had to make several changes to get it to both read the data and display it how I desired. I still don't fully understand how the * prefix to item.values() causes it to drop dictionaries default extra details, but I'm happy with the final result

[Image of dictionary updates in action]



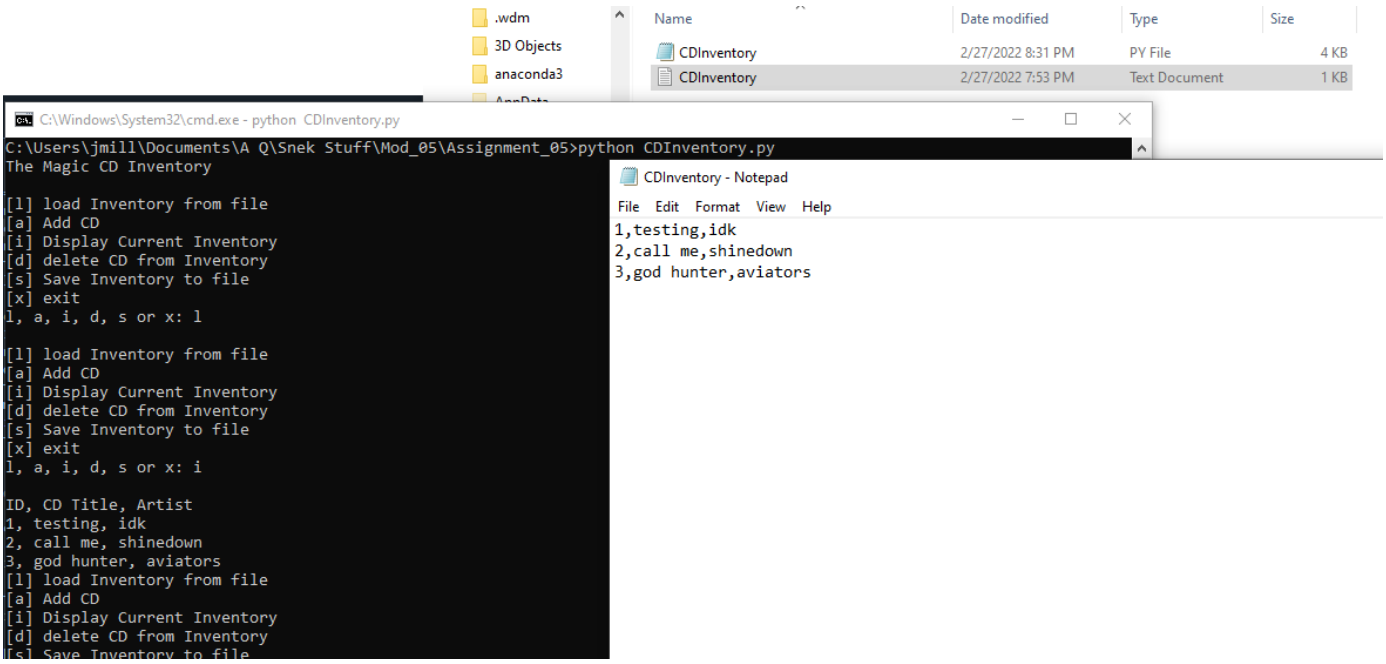
- Reading from File

This, oh gosh did I spend a long time smacking down issues. The first list I made appended the list of dictionaries instead of the dictionaries from the current inventory list, resulting in a list of a list of dictionaries, when the rest of the program is designed around a 2-d list. Used a for loop to fix that little blunder, but I was having issues next converting the str into

a list. I tried to deliminate on ‘,’ and ‘\n’ but do not yet know how, and ended up having to convert all the new lines to ‘,’ separately. I’m still not fully satisfied, as a song with commas in its title would cause issues, but a cursory google search indicates I do not yet know how to use separate tool repositories within python.

This first draft… immediately crashed and burned due to a variety of different issues. Because every entry ends with a new line, this was demininating into an additional ‘’ item in my list that I had to delete. I need to use range to make an integer of steps usable, and the entire time I used (n) for calling from items in a list when I needed to use [n] for python to recognize it. But finally I got it working and now it will read the txt file into a string into a list into a dictionary into a list of dictionaries. I’m still not happy that the program will make mistakes if a song has commas in the title, but I plan on asking for workarounds to that next Wednesday.

[CMD Successfully Pulling from TXT]



- Summary

This week was brutal, in large part due to a busy work week resulting in me not even getting past the reading into coding before the Friday Q & A. Overall I am inclined to say last week was more complicated for the jump in knowledge needed, but this week was no slouch and ended up even rougher on me due to my schedule.

- Appendix

```
strChoice = '' # User input
lstTbl = []  # list of lists to hold data
# TODO replace list of lists with list of dicts
DRow = []  # list of data row
strFileName = 'CDInventory.txt'  # data storage file
objFile = None  # file object
fromfile = [] #JM for use in loading data

# 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':
        objFile = open(strFileName, 'r')
        filestring = objFile.read()
        objFile.close()
        filestring = filestring.replace('\n',',')
        lfile = filestring.split(',')
        lfile = lfile[:-1]
        moving = 0
        reps = len(lfile)//3
        if reps > 0:
            for n in range(reps):
                dfile = {'id':int(lfile[moving+0]), 'Song':lfile[moving+1], 'Artist':lfile[moving+2]}
                fromfile.append(dfile)
                moving +=3
        for row in lstTbl:
            fromfile.append(row)
        lstTbl = fromfile
        fromfile = []
        # TODO Add the functionality of loading existing data
        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)
        DRow = {'ID': intID, 'Song': strTitle, 'Artist': strArtist} #JM converted inner list to a dictionary
        lstTbl.append(DRow)
    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 = ', ') #JM edited to return the values of the dictionary
    elif strChoice == 'd':
        Elim = input('Please enter the ID value of the song to remove, or ALL to delete the current log.
\nTo leave press enter')
        if Elim == 'ALL':
            LstTbl = []
        elif not Elim.isdigit():
            continue
        else:
            for item in lstTbl:
                if item['ID'] == int(Elim):
                    lstTbl.remove(item)
                    #JM: I'm a bit confused, as the assignment asks us to add inventory removal but also not
to use functions
        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 = ''
            for item in row.values(): #JM pretty sure this is functional formating for pulling from
dictionary
                strRow += str(item) + ','
            strRow = strRow[:-1] + '\n'
            objFile.write(strRow)
        objFile.close()
    else:
        print('Please choose either l, a, i, d, s or x!')
```