Amy Zhong
2/14/2021
Foundations of Programming: Python
Assignment 5

Week 5 Knowledge Document: CD Inventory

Introduction

For this week's assignment, I'm modifying the CD Inventory app we build from last week so that instead of using lists, I'm going to use dictionaries to record the data. I'm also going to demonstrate my knowledge of how to load existing data, delete data entries and save data into a 2D table. We have also touched upon Git in this module and I have set up my first Git repository[1] on Github. Below is my process documenting how I rewrote the program based off a given template, and my learnings along the way.

Process

Set up the program

For this week's assignment, we are provided with a template as the starting off point. To begin the program, I first added more detail to the header. Since I wasn't the original creator of the code, I executed and read over the script several times to familiar myself with the codes, especially with the variables. While it was nice to not have to start from scratch, there was definitely a learning curve in understanding the initial creator's thought process and code structure. I now have a newfound appreciation for code reviewers and basic coding best practices in general.

Understand the script and tasks

Upon understanding the script, there are 6 total actions the users can perform:
[l]: load the inventory list from a text file
[a]: add a new CD to the inventory
[i]: display the currently inventory
[d]: delete a CD from the inventory
[s]: save inventory to the text file
[x]: exit

For this assignment, my tasks are the following:
[i] and [x]: no code change required, unless I want add affordances to the users about the
[l] and [d]: need to write from scratch
[a] and [s]: need to rewrite the code from lists to dictionaries

Declare the variables

Here I just added dicRow = {} to declare a new dictionary.

[l]: Loading existing data

Since we are only interpreting an existing text file, we only need read action to the file. I used the code I learned from Lab05-B [2] to load the existing data in dictionary form. For each row in the text file, I first converted the rows of data into a list, separating by comma. Then I assigned key-value pairs to each of the data points. We can think of the keys here as the 'heading' of each column and the values are the cell within the column.

[1] https://github.com/ameszhong/assignment05
[2] https://saravji.github.io/saravjis_hut/FDN_Prog/content/Lab_05_B.html

[a]: Add a new entry to the inventory

Here, we'll need to covert the lists to dictionary form. Luckily, the code is fairly simple here. After gathering information from the user, instead of creating a rows of list, we are defining rows of dictionary defined by dicRow with {'ID': intID, 'Title': strTitle, 'Artist': strArtist}. You can tell this is a dictionary because I'm using {}.

[i]: Display current data

The only thing I changed in this section is from print(*row, sep = ', ') to print(*row.values(), sep = ', ') so I can print the values instead of the keys.

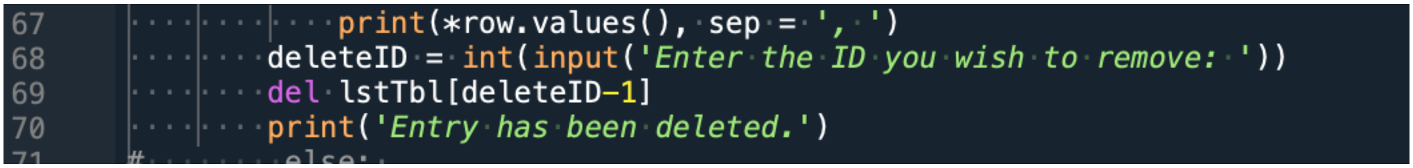
[d]: Delete current inventory

This is the section I had most struggle with. I started with pseudocodes to make sure I have the correct logics.

```
Display existing file
deleteID = Input( which id they would like to delete)
for row in table:
  if deleteID in row:
    Delete the row
  else
    error message
```

Because this is a 2D table, I needed to create a for loop to go through every row of data. If the ID value in the row matches the user input ID, then we can use the del function to delete the entire row.

What I didn't understand but got help during the office hour [3] was the need increase the row number by one every time in the loop to keep track where we are in the row. I originally did not have that in my code. While the script worked the first time, when I tried to delete another entry, it failed to perform. In my initial attempt, I was using the input as the index for the list but because the ID value could be in any order, my code failed to perform when the IDs were not sorted starting with 1.



```
            print(*row.values(), sep = ', ')
        deleteID = int(input('Enter the ID you wish to remove: '))
        del lstTbl[deleteID-1]
        print('Entry has been deleted.')
#       else:
```

Figure 1: My initial implementation - failed to work if IDs are out of numerical order

[s]: Saving entries to text file

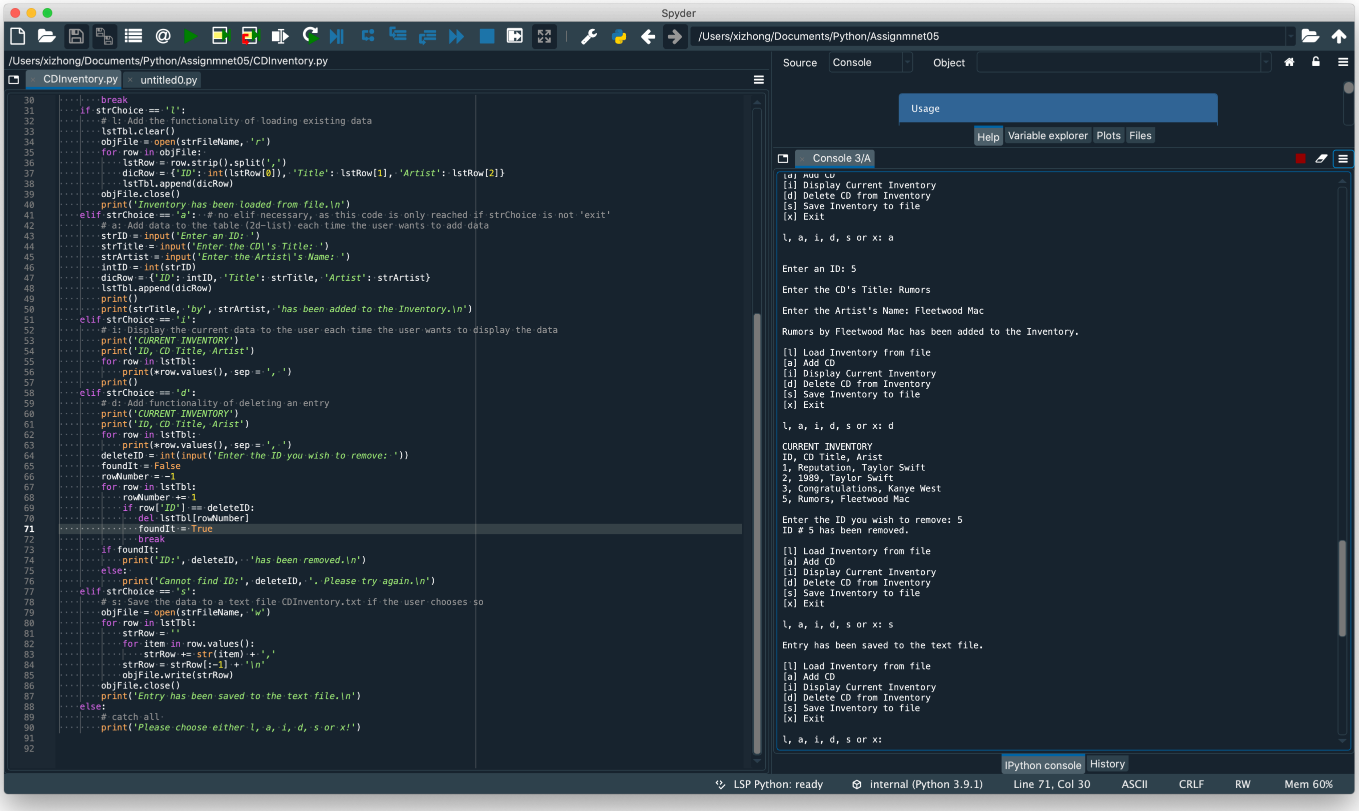
Similar to displaying entries, the only thing I changed was from for `item in row:` to `for item in row.values():` so that the values are saved.

Polish and test

Now that I tested the code and made sure everything ran properly, I added the final polishes: provided more context, checked spelling, etc. I ran the script one more time in Spyder, and in the Terminal to make sure it's working as expected.

Here's how the script looked in Spyder:

[3] https://washington.zoom.us/rec/play/Xj_jPvuJSteGNv93JudYgF6xGS9-0VOWdeg30mpNY1kgn51uw2ltS_q_E5HwOd6aaU4g7TTVnoFPtJnm.7Gork_nbJJEDeiae?continueMode=true starting from 00:44min

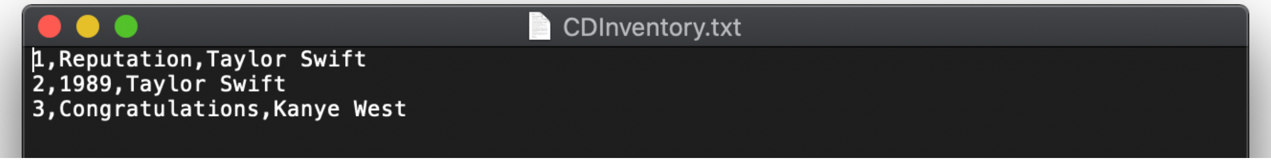


```python
            break
    if strChoice == 'l':
        # l: Add the functionality of loading existing data
        lstTbl.clear()
        objFile = open(strFileName, 'r')
        for row in objFile:
            lstRow = row.strip().split(',')
            dicRow = {'ID': int(lstRow[0]), 'Title': lstRow[1], 'Artist': lstRow[2]}
            lstTbl.append(dicRow)
        objFile.close()
        print('Inventory has been loaded from file.\n')
    elif strChoice == 'a':  # no elif necessary, as this code is only reached if strChoice is not 'exit'
        # a: 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': intID, 'Title': strTitle, 'Artist': strArtist}
        lstTbl.append(dicRow)
        print()
        print(strTitle, 'by', strArtist, 'has been added to the Inventory.\n')
    elif strChoice == 'i':
        # i: Display the current data to the user each time the user wants to display the data
        print('CURRENT INVENTORY')
        print('ID, CD Title, Artist')
        for row in lstTbl:
            print(*row.values(), sep = ', ')
        print()
    elif strChoice == 'd':
        # d: Add functionality of deleting an entry
        print('CURRENT INVENTORY')
        print('ID, CD Title, Arist')
        for row in lstTbl:
            print(*row.values(), sep = ', ')
        deleteID = int(input('Enter the ID you wish to remove: '))
        foundIt = False
        rowNumber = -1
        for row in lstTbl:
            rowNumber += 1
            if row['ID'] == deleteID:
                del lstTbl[rowNumber]
                foundIt = True
                break
        if foundIt:
            print('ID:', deleteID,  'has been removed.\n')
        else:
            print('Cannot find ID:', deleteID, '. Please try again.\n')
    elif strChoice == 's':
        # s: Save the data to a text file CDInventory.txt if the user chooses so
        objFile = open(strFileName, 'w')
        for row in lstTbl:
            strRow = ''
            for item in row.values():
                strRow += str(item) + ','
            strRow = strRow[:-1] + '\n'
            objFile.write(strRow)
        objFile.close()
        print('Entry has been saved to the text file.\n')
    else:
        # catch all
        print('Please choose either l, a, i, d, s or x!')
```

```
[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: a


Enter an ID: 5

Enter the CD's Title: Rumors

Enter the Artist's Name: Fleetwood Mac

Rumors by Fleetwood Mac has been added to the 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: d

CURRENT INVENTORY
ID, CD Title, Arist
1, Reputation, Taylor Swift
2, 1989, Taylor Swift
3, Congratulations, Kanye West
5, Rumors, Fleetwood Mac

Enter the ID you wish to remove: 5
ID # 5 has been removed.

[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

Entry has been saved to the text file.

[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:
```

Figure 2: final code in Spyder

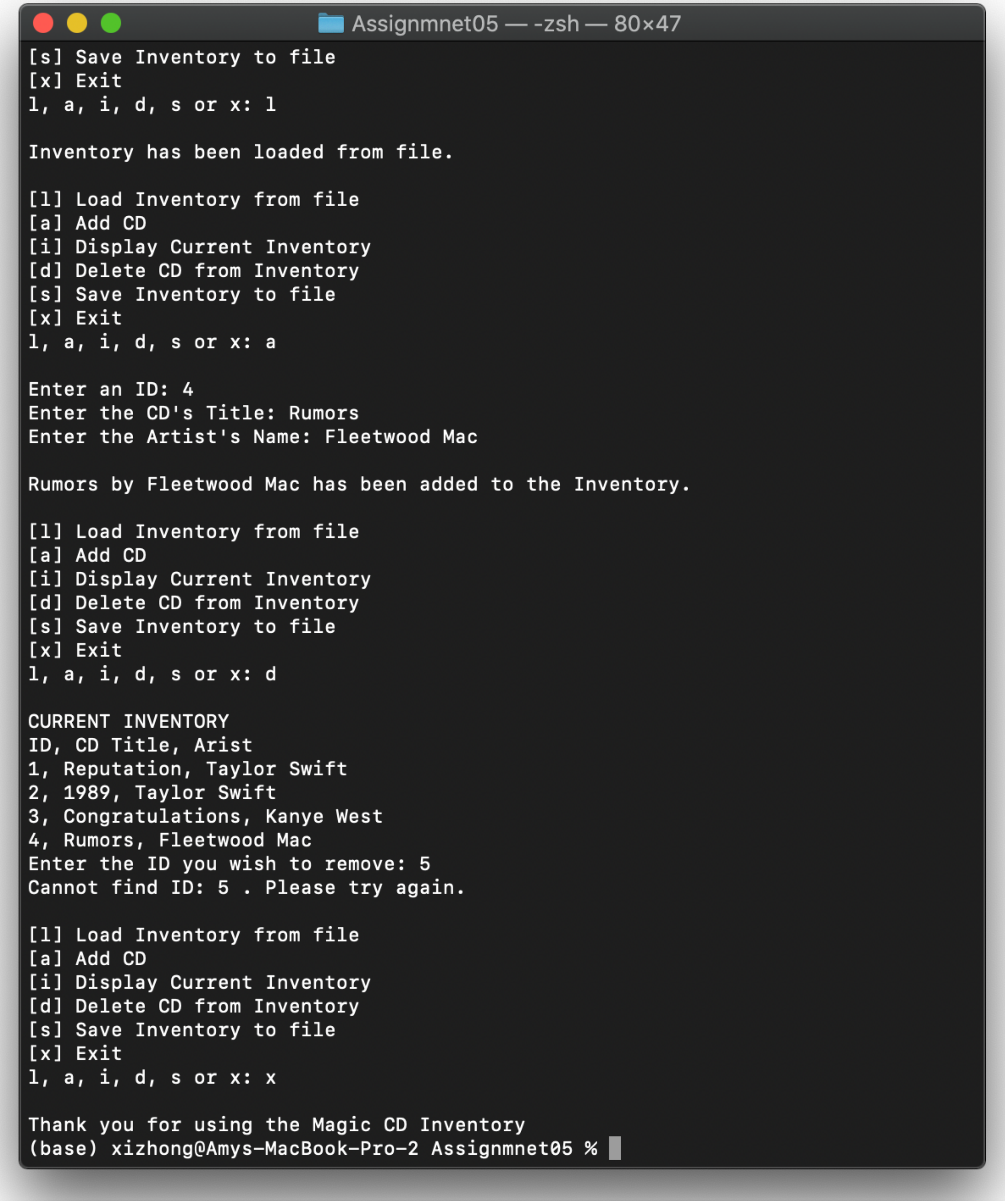
With the corresponding text file:



```
1,Reputation,Taylor Swift
2,1989,Taylor Swift
3,Congratulations,Kanye West
```

Figure 3: Corresponding text file with new data

And here's how it looks in Terminal:



```
[s] Save Inventory to file
[x] Exit
l, a, i, d, s or x: l

Inventory has been loaded from file.

[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: a

Enter an ID: 4
Enter the CD's Title: Rumors
Enter the Artist's Name: Fleetwood Mac

Rumors by Fleetwood Mac has been added to the 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: d

CURRENT INVENTORY
ID, CD Title, Arist
1, Reputation, Taylor Swift
2, 1989, Taylor Swift
3, Congratulations, Kanye West
4, Rumors, Fleetwood Mac
Enter the ID you wish to remove: 5
Cannot find ID: 5 . Please try again.

[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

Thank you for using the Magic CD Inventory
(base) xizhong@Amys-MacBook-Pro-2 Assignmnet05 %
```

Figure 4: Final script in Terminal

Summary

In this module, I learned about dictionaries and how to convert lists to dictionaries. I also learned how to read and write files directly through an IDE. As we progress into the quarter, the scripts we are writing are getting more and more complicated. The need for us to keep our scripts efficient and effective also increases. While not directly shown in the assignment, we also learned about ways to improve our scripts and briefly touched upon git and Github. Personally speaking, it has become significantly more difficult for me to troubleshoot codes and understand what went wrong. I'm hoping we'll be learning ways to practice more efficient and clearer codes than what we are doing now.

Appendix

Github repo

https://github.com/ameszhong/assignment05

Final code

```
#------------------------------------------#
# Title: CDInventory.py
# Desc: Starter Script for Assignment 05
# Change Log: (Who, When, What)
# DBiesinger, 2030-Jan-01, Created File
# AZhong, 2021-Feb-14, Modifyied the script fro using lists to dictionaries
#------------------------------------------#

# Declare variabls

strChoice = '' # User input
lstTbl = []  # list of lists to hold data
dicRow = {} # dictionary of 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':
        # x: Exit the program if the user chooses so
        print('Thank you for using the Magic CD Inventory')
        break
    if strChoice == 'l':
        # l: Add the functionality of loading existing data
        lstTbl.clear()
        objFile = open(strFileName, 'r')
        for row in objFile:
            lstRow = row.strip().split(',')
            dicRow = {'ID': int(lstRow[0]), 'Title': lstRow[1], 'Artist': lstRow[2]}
            lstTbl.append(dicRow)
        objFile.close()
        print('Inventory has been loaded from file.\n')
    elif strChoice == 'a':  # no elif necessary, as this code is only reached if strChoice is not 'exit'
        # a: 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': intID, 'Title': strTitle, 'Artist': strArtist}
        lstTbl.append(dicRow)
        print()
        print(strTitle, 'by', strArtist, 'has been added to the Inventory.\n')
    elif strChoice == 'i':
        # i: Display the current data to the user each time the user wants to display the data
        print('CURRENT INVENTORY')
        print('ID, CD Title, Artist')
        for row in lstTbl:
            print(*row.values(), sep = ', ')
        print()
    elif strChoice == 'd':
        # d: Add functionality of deleting an entry
        print('CURRENT INVENTORY')
        print('ID, CD Title, Arist')
        for row in lstTbl:
            print(*row.values(), sep = ', ')
        deleteID = int(input('Enter the ID you wish to remove: '))
        foundIt = False
        rowNumber = -1
        for row in lstTbl:
            rowNumber += 1
            if row['ID'] == deleteID:
               del lstTbl[rowNumber]
               foundIt = True
               break
        if foundIt:
            print('ID:', deleteID,  'has been removed.\n')
        else:
            print('ID:', deleteID, 'cannot be found. Please try again.\n')
    elif strChoice == 's':
        # s: Save the data to a text file CDInventory.txt if the user chooses so
        objFile = open(strFileName, 'w')
        for row in lstTbl:
            strRow = ''
            for item in row.values():
                strRow += str(item) + ','
            strRow = strRow[:-1] + '\n'
            objFile.write(strRow)
        objFile.close()
        print('New entries have been saved to the text file.\n')
    else:
        # catch all
        print('Please choose either l, a, i, d, s or x!')
```