

CD Inventory Program (with Dictionaries)

Introduction

In this assignment we were asked to build out an existing CD Inventory program so that it could have the following functionality:

[l] Load Inventory from File - *This function pulls existing lists from the designated text file and converts them to dictionaries within the program's list by assigning values to the existing keys. This function allows the program to manipulate dictionaries created from lists on an external source.*

[a] Add CD - *This function allows the user to input new dictionaries (CDs) into the program.*

[i] Display Current Inventory - *This function displays all dictionaries (inventory) that exists within the program. Note that there could be more inventory that was previously saved to the text file database.*

[d] Delete CD from Inventory - *This function allows the user to delete dictionaries (CDs) from the existing list (inventory) within the program.*

[s] Save Inventory to File - *This function opens the designated text file and reformats the dictionaries before writing them into the text file as lists.*

[x] Exit (l) - *This function ends the program.*

Although similar in many ways to the previous assignment, this latest program modifies the 2D data structure to use dictionaries as the inner data type (list of dictionaries). This update required many changes to the program, including the defining of keys and values. Additionally, significant reformatting was needed to move data back and forth between the program and an external text file.

Defining the Variables

To start the assignment, I opened the existing python file that was shared with us in the Mod_05 zip file and saved a blank text file named, 'CDInventory.txt'. The first update I made was defining the variables, specifically the blank dictionary row that I would be populating and appending to the list of dictionaries. See Figure 1 for the code.

```
8      # Declare variables
9
10     strChoice = '' # User input
11     lstTbl = [] # list of dicts to hold data
12     dicRow = {'intID': 0, 'strTitle': '', 'strArtist': ''} # blank dictionary key
13
14     # Opening Text file
15     strFileName = 'CDInventory.txt' # data storage file
16     objFile = None # file object
```

Figure 1

I also chose to define the ID variable so that it would change each time I input a new CD. The variable first counts how many CD's are within the external database and then begins the integer count based on the existing number of CDs. This feature of the program makes it so the user does not have to define the ID each time a CD is input into the program. Please see the next page (Figure 2) for the code used to define the ID variable.

```
18 # Defining the ID
19 objFile = open(strFileName, 'r')
20 for row in objFile:
21     NewRow = row.strip().split(',')
22     dicRow = {'intID': NewRow[0], 'strTitle': NewRow[1], 'strArtist': NewRow[2]}
23     lstTbl.append(dicRow)
24 ID = int(len(lstTbl))
25 lstTbl = []
26 objFile.close()
```

Figure 2

CD Inventory Program Options

The program prints a list of all available options and requests a user input to execute each option. I will now go through the code for each program:

[l] Load Inventory from File: This portion of the code opens the 'CDInventory' text file which is populated by lists (CD values). For each row within the text file, this function reformats the list so it is readable by the program. It then uses the values within the lists to redefine the values within the blank dictionary ('dicRow'). Once the blank dictionary has the values defined, it is appended to the list of dictionaries. This process is repeated until all rows within the text file have been converted to dictionaries and appended to the list of dictionaries. Once complete, the program can then read and manipulate the dictionaries (CDs) that were sourced from the text file.

```
41 # Load Inventory from file
42 if strChoice == 'l':
43     print('\nLoading Inventory from File...')
44     objFile = open(strFileName, 'r')
45     for row in objFile:
46         NewRow = row.strip().split(',')
47         dicRow = {'intID': int(NewRow[0]), 'strTitle': NewRow[1], 'strArtist': NewRow[2]}
48         lstTbl.append(dicRow)
49         #Clear the values
50         dicRow = {'intID': 0, 'strTitle': '', 'strArtist': ''}
51     objFile.close()
```

Figure 3: Load Inventory from File

[a] Add CD: This function is similar to the previous program completed in Assignment_04. The main difference is that we are defining keys instead of variables. Once the blank dictionary ('dicRow') has been populated by user input, it is appended to the list of dictionaries and then the values are removed. See below for example code:

```
59 # Add CD
60 elif strChoice == 'a':
61     ID = ID + 1
62     dicRow['intID'] = ID
63     dicRow['strTitle'] = input('Enter the CD\'s Title: ')
64     dicRow['strArtist'] = input('Enter the Artist\'s Name: ')
65     lstTbl.append(dicRow)
66     #Clear the values
67     dicRow = {'intID': 0, 'strTitle': '', 'strArtist': ''}
```

Figure 4: Add CD

[i] Display Current Inventory: This function is also very similar to the previous program in Assignment_04. The only difference is that it is printing dictionaries. To make the data more presentable, it only prints the values of the dictionaries.

```
69 # Display Current Inventory
70 elif strChoice == 'i':
71     print()
72     lstTbl
73     for row in lstTbl:
74         print(row.values())
75
```

Figure 5: Display Current Inventory:

[d] Delete CD from Inventory: This function prints all the dictionaries (CDs) within the current inventory (lstTbl). The function then asked the user to input which ID they would like to delete. Using a for loop and if function, the program is able to eliminate the corresponding row based on the ID inputted by the user.

```
76 # Delete CD from Inventory
77 elif strChoice == 'd':
78     print()
79     for row in lstTbl:
80         print(row.values())
81     IDRemove = int(input('Select which ID # you would like to delete: ')) - 1
82     for row in lstTbl:
83         if row['intID'] == IDRemove:
84             lstTbl.remove(row)
85     print(row)
```

Figure 6: Delete CD from Inventory

[s] Save Inventory to File - This function first opens the text file ('CDInventory'). It then opens each value in each row, converts them to strings and then reformats them to be in list form. By doing this, the values can be written into the text document in a presentable way. Additionally, the values can be imported back into the program using the 'Load Inventory from File' function.

```
87 # Save Inventory to File
88 elif strChoice == 's':
89     # 4. Save the data to a text file CDInventory.txt if the user chooses so
90     objFile = open(strFileName, 'a')
91     for row in lstTbl:
92         list_of_values = list(row.values())
93         list_of_values = str(list_of_values).replace(', ', ',').replace('[', '').replace(']', '').replace('\\', '') + '\n'
94         objFile.write(list_of_values)
95         list_of_values = ''
96     objFile.close()
97     print('\nSaved!')
98     #Clear table not that info has been exported
99     lstTbl = []
```

Figure 7: Save Inventory to File

[x] Exit (I): This function simply ends the program.

```
36 # Exit
37 if strChoice == 'x':
38     # 5. Exit the program if the user chooses so
39     break
```

Figure 8: Exit

Running the Program (in Spyder)

1) To be efficient in this section of the summary, I populated the text file with some CD values:

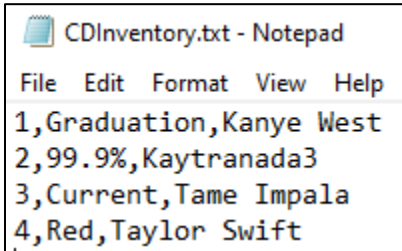


Figure 9: Initial Text Document

2) I then ran the program and loaded the text document:

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

Loading Inventory from File...
Items in list now:
{'intID': 1, 'strTitle': 'Graduation', 'strArtist': 'Kanye West'}
{'intID': 2, 'strTitle': '99.9%', 'strArtist': 'Kaytranada3'}
{'intID': 3, 'strTitle': 'Current', 'strArtist': 'Tame Impala'}
{'intID': 4, 'strTitle': 'Red', 'strArtist': 'Taylor Swift'}
```

Figure 10: Loading values from text file in dictionary format

3) I then added a CD:

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

Enter the CD's Title: Yellow Submarine

Enter the Artist's Name: The Beatles
```

Figure 11: Adding CD

4) I then viewed the current inventory within the program:

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

dict_values([1, 'Graduation', 'Kanye West'])
dict_values([2, '99.9%', 'Kaytranada3'])
dict_values([3, 'Current', 'Tame Impala'])
dict_values([4, 'Red', 'Taylor Swift'])
dict_values([5, 'Yellow Submarine', 'The Beatles'])
```

Figure 12: Viewing Program Current Inventory

5) I then removed the dictionary with an ID of 3 from the program's current inventory:

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

dict_values([1, 'Graduation', 'Kanye West'])
dict_values([2, '99.9%', 'Kaytranada3'])
dict_values([3, 'Current', 'Tame Impala'])
dict_values([4, 'Red', 'Taylor Swift'])
dict_values([5, 'Yellow Submarine', 'The Beatles'])

Select which ID # you would like to delete: 3
{'intID': 1, 'strTitle': 'Graduation', 'strArtist': 'Kanye West'}
{'intID': 2, 'strTitle': '99.9%', 'strArtist': 'Kaytranada3'}
{'intID': 4, 'strTitle': 'Red', 'strArtist': 'Taylor Swift'}
{'intID': 5, 'strTitle': 'Yellow Submarine', 'strArtist': 'The Beatles'}
```

Figure 13: Remove CD

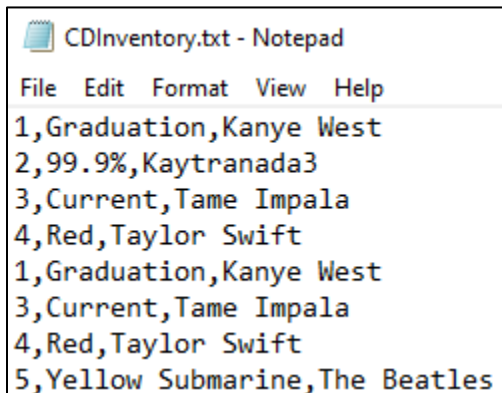
6) I then saved the inventory to the file:

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

Saved!
```

Figure 14: Saving current inventory to file

If you open the text file, you will see the updated current inventory was added (in addition to what was previously in the text document). I could have added a function to clear the text document, however that was not specified in the assignment so I opted out:



```
CDInventory.txt - Notepad
File Edit Format View Help
1,Graduation,Kanye West
2,99.9%,Kaytranada3
3,Current,Tame Impala
4,Red,Taylor Swift
1,Graduation,Kanye West
3,Current,Tame Impala
4,Red,Taylor Swift
5,Yellow Submarine,The Beatles
```

Figure 15: Updated text file

7) I then exited the program:

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

In [166]: |
```

Figure 16: Exit program.

Running the Program (in Terminal)

I then completed the exact same steps in Terminal.

1) To be efficient in this section of the summary, I populated the text file with some CD values.

2) I then ran the program and loaded the text document:

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

Loading Inventory from File...
Items in list now:
{'intID': 1, 'strTitle': 'Graduation', 'strArtist': 'Kanye West'}
{'intID': 2, 'strTitle': '99.9%', 'strArtist': 'Kaytranada3'}
{'intID': 3, 'strTitle': 'Current', 'strArtist': 'Tame Impala'}
```

Figure 17

3) I then added a CD:

```
l, a, i, d, s or x: a
Enter the CD's Title: Yellow Submarine
Enter the Artist's Name: The Beatles
```

Figure 18

4) I then viewed the current inventory within the program:

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

dict_values([1, 'Graduation', 'Kanye West'])
dict_values([2, '99.9%', 'Kaytranada3'])
dict_values([3, 'Current', 'Tame Impala'])
dict_values([4, 'Yellow Submarine', 'The Beatles'])
```

Figure 19

5) I then removed the dictionary with an ID of 3 from the program's current inventory:

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

dict_values([1, 'Graduation', 'Kanye West'])
dict_values([2, '99.9%', 'Kaytranada3'])
dict_values([3, 'Current', 'Tame Impala'])
dict_values([4, 'Yellow Submarine', 'The Beatles'])
Select which ID # you would like to delete: 3
{'intID': 1, 'strTitle': 'Graduation', 'strArtist': 'Kanye West'}
{'intID': 2, 'strTitle': '99.9%', 'strArtist': 'Kaytranada3'}
{'intID': 4, 'strTitle': 'Yellow Submarine', 'strArtist': 'The Beatles'}
```

Figure 20

6) I then saved the inventory to the file:

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

Saved!
```

Figure 21

The text document was updated in the same way as before.

7) I then exited the program:

```
l, a, i, d, s or x: x  
(base) C:\_FDProgramming\Mod_05\Assignment_05>
```

Figure 22

Summary

Using the textbook, the Module 04 documentation and videos, and the supplemental websites and video, I was able to successfully create the assignment program. The program demonstrates my newly developed knowledge of working with dictionaries.