

# CDInventory.py Script

## Introduction

This document outlines the steps required for generating a script that has a menu structure and allows user to load CD data from CDInventory.txt file, enter CD data, view the current inventory, delete DC data from inventory, save data to CDInventory.txt data file, and exit the program. The script was written in the Spyder IDE, and its successful operation was shown in Spyder and anaconda terminal. Lastly, the document summarizes my learnings from Module 5.

## Steps:

**Generation of Pseudocode:** The script starts with generation of a pseudocode for the overall script. As shown in the **Listing 1**, the script initially provides multiple options to the user (Load Inventory, Add CD, Display Current Inventory, Delete CD, Save Inventory to File, and Exit). If the user selects one of these options, then that specific part of the script runs.

```
1  #-----#
2  # Title: CDInventory.py
3  # Desc: Script CDInventory to store CD Inventory data
4  # Change Log: (Who, When, What)
5  # DBiesinger, 2030-Jan-01, Created File #
6  # MYokus, 2021-Aug-06, Added Code
7  #-----#
8
9
10 # Declare variables
11
12
13
14 # Get user Input
15
16 ...# 1. Display menu allowing the user to choose:
17 ...
18 ...# 2. Load inventory from file
19 ...
20 ...# 3. Add CD data to list in memory
21 ...
22 ...# 4. Display current inventory
23 ...
24 ...# 5. Delete CD from inventory
25 ...
26 ...# 6. Save inventory to a text file CDInventory.txt
27
28 ...# 7. Exit the program if the user chooses so
```

*Listing 1- Header and layout of the steps in the pseudocode*

## Steps of the Code

### 1. Displaying menu allowing the user to choose: 'Load Inventory', 'Add CD', 'Display Current Inventory', 'Delete CD', 'Save Inventory to file', and 'exit':

The script starts with declaring the variables that will be utilized throughout the code. Afterwards, the user is provided with six different options to choose from with `print()` and `input()` functions as shown in **Listing 2**. The `input()` function was included in the `while` loop so that the menu options are presented to the user again after the user completes a specific option in the menu (**Fig. 1**).

```
10 # Declare variables
11
12 strChoice = '' # User input
13 lstTbl = [] # list of dictionaries to hold data
14 dicRow = {} # a dictionary
15 strFileName = 'CDInventory.txt' # data storage file
16 objFile = None # file object
17
18 # Get user Input
19 print('The Magic CD Inventory\n (Note: There must be CDInventory.txt file in the directory to load the data from the text file!).\n')
20 while True:
21     # 1. Display menu allowing the user to choose:
22     print('[l] Load Inventory from file\n[a] Add CD\n[i] Display Current Inventory')
23     print('[d] delete CD from Inventory\n[s] Save Inventory to file\n[x] exit')
24     strChoice = input('l, a, i, d, s or x: ').lower() # convert choice to lower case at time of input
25     print()
26
```

Listing 2 – Displaying menu options to the user.

```
In [1]: runfile('C:/programming/Assignment05/CDInventory.py', wdir='C:/programming/
Assignment05')
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
```

Figure 1- Displaying menu options to the user (Listing 2).

### 2. Loading Inventory from CDInventory.txt file

The next step is to load the CD inventory data from the CDInventory.txt file. (**Note: There must be CDInventory.txt file in the directory to run this part of the script!**). If the user input is "l", loading inventory from a text file is selected (**Listing 3**). The text file already includes three CD entries, which will be loaded into memory with this step. Within this menu option, the text file is opened using `open()` function and each row in the text file is read using a `for` loop. Within the `for` loop, the code iterates through each row of the text file, removes "\n" at the end of each row using `strip()` function, and separates the elements in each row with a comma using `split()` function. Then, each value in the row is assigned to a `key` in a dictionary, and the dictionary is appended to a list using `append()` function. The text file is then closed with `objFile.close()` function. This step is shown in **Fig. 2**.

```

30 ....if strChoice == 'l': # no elif necessary, as this code is only reached if strChoice is not 'x'
31 .....# 2. Load inventory from file
32 .....lstTbl.clear() # Clear the inventory list in the memory before loading the file
33 .....objFile = open(strFileName, 'r') # Open the CDInventory.txt file
34 .....for row in objFile:
35 .....    lstRow = row.strip().split(',') # Remove "\n" at the end of each row and separates each element with a comma
36 .....    dicRow = {'id': int(lstRow[0]), 'title': lstRow[1], 'artist': lstRow[2]}
37 .....    lstTbl.append(dicRow) # Append the new dictionary to the list
38 .....objFile.close() # Close the CDInventory.txt file

```

*Listing 3- Loading CD Inventory from CDInventory.txt file.*

```

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

[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- Loading CD Inventory from CDInventory.txt file (Listing 3).*

### 3. Adding CD data to the inventory:

The selection of different options in the menu was done with *if-elif-else* statement. If the user input is “a”, adding CD data to the inventory option in the main menu is selected (**Listing 4**). We already have three CD data in the inventory list and we will be adding new CD data to this inventory. Within this step, three *input()* functions were included to read the ID number (as an string), CD title (as a string), and Artist’s name (as a string). The string for the ID number was converted to an integer using *int()* function. Afterwards, these three inputs were added to a new dictionary, called *dicRow*, which was then appended to the main 2D list of dictionaries (*i.e.*, CD inventory) using *lstTbl.append()* function (**Listing 4**). The successful completion of this step was shown in **Fig. 3**. The figure shows adding a single CD to the previous inventory. The total number of CD data in the inventory is now equal to 4.

```

39 ....elif strChoice == 'a':
40 .....# 3. Add CD data to list in memory
41 .....strID = input('Enter an ID: ') # Read "CD ID" user input
42 .....strName = input('Enter the CD\'s Title: ') # Read "CD Title" user input
43 .....strMail = input('Enter the Artist\'s Name: ') # Read "CD Name" user input
44 .....intID = int(strID)
45 .....dicRow = {'id': intID, 'title': strName, 'artist': strMail} # New CD data dictionary
46 .....lstTbl.append(dicRow) # Append the new CD data dictionary to the list
47 .....print()

```

*Listing 4 – Adding CD data to the inventory*

```

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

Enter an ID: 4

Enter the CD's Title: TitleD

Enter the Artist's Name: ArtistD
[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 3- Adding CD data to the inventory (Listing 4). The image shows adding an ID, title, and artist name to the previous inventory.

#### 4. Displaying the current inventory to the user each time the user wants to display the data

The next step in the script is to display the current inventory to the user. If the user input is “i”, displaying the current inventory option in the main menu is selected (**Listing 5**). Within the *elif* clause, a *for loop* was included to iterate through rows of the CD inventory and to print them on the screen. Within the *print()* function, unpacking of the list was done with ‘\*’. The successful completion of this step is shown in **Fig. 4**. As mentioned previously, we started with three CD entries in the inventory and then added a new CD to the list in the previous step. Now, there are four CD entries in the 2D list as shown in the **Fig. 4**.

```

48 .....elif strChoice == 'i':
49 .....# 4. Display current inventory
50 .....print('ID, Title, Artist')
51 .....for row in lstTbl:
52 .....    print(*row.values(), sep=',') # Unpacking the list with *
53 .....    print()

```

Listing 5 – Displaying the inventory to the user.

```

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

ID, Title, Artist
1, TitleA, ArtistA
2, TitleB, ArtistB
3, TitleC, ArtistC
4, TitleD, ArtistD

[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 4 – Displaying the current inventory to user each time the user wants to see the inventory (Listing 5).

## 5. Deleting CD data from inventory

The next step is to delete an entry in the inventory. If the user input is “d”, deleting a CD entry from the inventory option is selected in the main menu (**Listing 6**). To achieve this action, the user was asked to input which entry they like to remove in the CD inventory using `input()` function. Then, `lstTbl.remove()` function was used to delete a specific row in the 2D list. For instance, the user input was 4, and the 4<sup>th</sup> CD entry in the inventory was deleted (**Fig. 5**). The remaining CD inventory after the deletion of the last entry in the list was displayed in **Fig. 5** to demonstrate successful completion of this step. The remaining number of CD entries in the inventory is now equal to three only.

```
54     ...elif strChoice == 'd':
55         ...# 5. Delete CD from inventory
56         ...deleteRow = int(input('Which CD entry (row) do you like to delete?: '))
57         ...lstTbl.remove(lstTbl[deleteRow-1]) # Remove Xth entry in CD data inventory
```

*Listing 6 – Deleting a CD data from inventory.*

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

Which CD entry (row) do you like to delete?: 4
[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, Title, Artist
1, TitleA, ArtistA
2, TitleB, ArtistB
3, TitleC, ArtistC

[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 5 – Deleting a CD from inventory (Listing 6). This picture shows the deletion of 4<sup>th</sup> CD in the inventory. The remaining number of CDs in the inventory is now equal to three.*

## 6. Saving the inventory to a text file and Exiting

Saving the current inventory to the CDInventory.txt option is selected in the main menu if the user input is “s” (**Listing 7**). First, the CDInventory.txt file was opened using *open()* function in “w: write” mode. The “w” mode will erase the existing file with similar name and create a new text file to save the CD inventory data. Then, a nested *for loop* was used to concatenate each item in a row of the 2D CD inventory list into a string. Each string was then saved to the CDInventory.txt file using *objFile.write()* function. The file was then closed with *objFile.close()* function. Saving the CD inventory to the CDInventory.txt file was shown in the **Fig. 6**. Saving data to a text file was successful as shown in **Fig. 8**.

Exiting the *while loop* is accomplished if the user input is “x” in the main menu (**Fig. 6 and 7**). *Break* function gets out of the loop immediately.

```
58 elif strChoice == 's':
59     # 6. Save inventory to a text file CDInventory.txt
60     objFile = open(strFileName, 'w') # Open the CDInventory.txt file.
61     # "w" mode: An existing file with the same name will be erased and it will write over.
62     for row in lstTbl:
63         strRow = ''
64         for item in row.values():
65             strRow += str(item) + ','
66         strRow = strRow[:-1] + '\n' # Remove ',' at the end of each row and add a new line
67         objFile.write(strRow) # Write each DC data row to the CDInventory.txt file
68     objFile.close() # Close the CDInventory.txt file
69 else:
70     print('Please choose either l, a, i, d, s or x!')
```

Listing 7 – Saving the inventory data to CDInventory.txt file

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

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

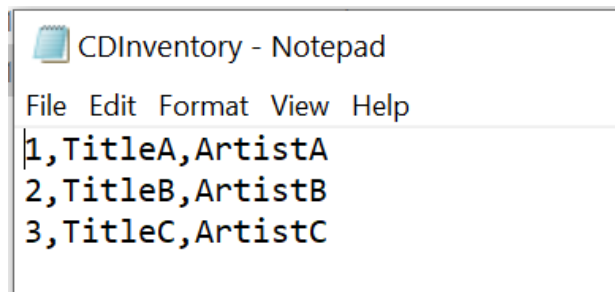
l, a, i, d, s or x: x

In [2]:
```

Figure 6 – Saving the inventory data to CDInventory.txt file and exiting (Listing 7)

```
27 if strChoice == 'x':
28     # 7. Exit the program if the user chooses so
29     break
```

Figure 7- Exiting program (Listing 7)

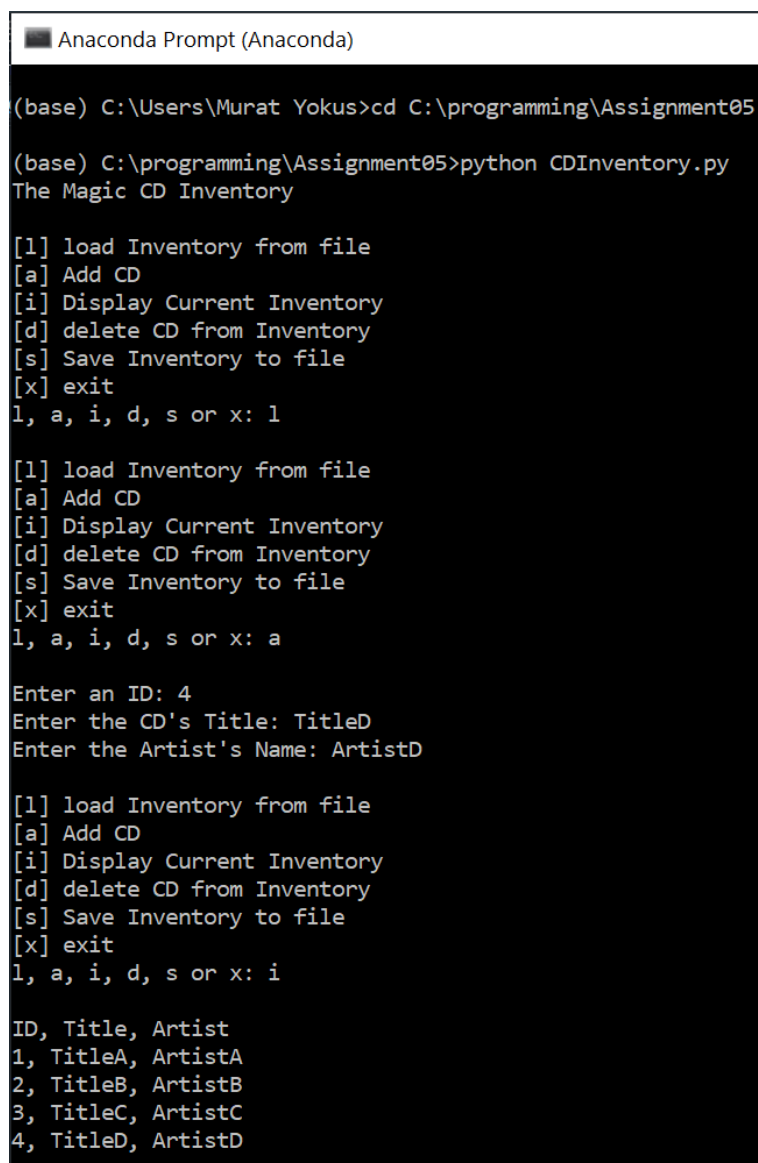


```
CDInventory - Notepad
File Edit Format View Help
1,TitleA,ArtistA
2,TitleB,ArtistB
3,TitleC,ArtistC
```

Figure 8 – Snapshot of the 2D DC inventory data saved to the CDInventory.txt file.

## 7. Verification of the code in the Terminal

The script was written and run in Spyder IDE. Similar code was also verified in Anaconda terminal. The successful operation of the script in the terminal is shown in **Fig. 9** and **10**.



```
Anaconda Prompt (Anaconda)

(base) C:\Users\Murat Yokus>cd C:\programming\Assignment05

(base) C:\programming\Assignment05>python CDInventory.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

[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: TitleD
Enter the Artist's Name: ArtistD

[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, Title, Artist
1, TitleA, ArtistA
2, TitleB, ArtistB
3, TitleC, ArtistC
4, TitleD, ArtistD
```

Figure 9 – Verification of the CDInventory.py script in the terminal.

```
Anaconda Prompt (Anaconda)

[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 entry (row) do you like to delete?: 4
[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: x

(base) C:\programming\Assignment05>
```

Figure 10 – Verification of the CDInventory.py script in the terminal (cont'd).

## 7. GitHub Link

The knowledge document, the script, and CDInventory.txt file were uploaded to GitHub/Assignment\_05 repository. Link: [https://github.com/myokus/Assignment\\_05](https://github.com/myokus/Assignment_05)

## Module 4: Learnings

In the Module 5, I learned and practiced the following topics.

- Dictionaries and lists
- Reading from a file into a list
- Reading from a file into a dictionary
- Writing to a file
- Brief introduction to functions



## Summary

Overall, the objective of this assignment is to create a CD inventory program that has a menu structure and allows user to load CD data from CDInventory.txt file, enter CD data, view the current inventory, delete DC data from inventory, save data to CDInventory.txt data file, and exit the program. This document demonstrated the successful completion of the individual steps of the assignment along with my learnings from the Module 5.

## Appendix

### Listing CDInventory.py

```
1. #-----#
2. # Title: CDInventory.py
3. # Desc: Script CDInventory to store CD Inventory data
4. # Change Log: (Who, When, What)
5. # DBiesinger, 2030-Jan-01, Created File #
6. # MYokus, 2021-Aug-06, Added Code
7. #-----#
8.
9.
10. # Declare variables
11.
12. strChoice = '' # User input
13. lstTbl = [] # List of dictionaries to hold data
14. dicRow = {} # a dictionary
15. strFileName = 'CDInventory.txt' # data storage file
16. objFile = None # file object
17.
18. # Get user Input
19. print('The Magic CD Inventory\n (Note: There must be CDInventory.txt file in the directory to load the
    data from the text file!).\n')
20. while True:
21.     # 1. Display menu allowing the user to choose:
22.     print('[l] load Inventory from file\n[a] Add CD\n[i] Display Current Inventory')
23.     print('[d] delete CD from Inventory\n[s] Save Inventory to file\n[x] exit')
24.     strChoice = input('l, a, i, d, s or x: ').lower() # convert choice to lower case at time of input
25.     print()
26.
27.     if strChoice == 'x':
28.         # 7. Exit the program if the user chooses so
29.         break
30.     if strChoice == 'l': # no elif necessary, as this code is only reached if strChoice is not 'x'
31.         # 2. Load inventory from file
32.         lstTbl.clear() # Clear the inventory list in the memory before Loading the file
33.         objFile = open(strFileName, 'r') # Open the CDInventory.txt file
34.         for row in objFile:
35.             lstRow = row.strip().split(',') # Remove "\n" at the end of each row and separates each
                element with a comma
36.             dicRow = {'id': int(lstRow[0]), 'title': lstRow[1], 'artist': lstRow[2]}
37.             lstTbl.append(dicRow) # Append the new dictionary to the list
38.         objFile.close() # Close the CDInventory.txt file
39.     elif strChoice == 'a':
40.         # 3. Add CD data to List in memory
41.         strID = input('Enter an ID: ') # Read "CD ID" user input
42.         strName = input('Enter the CD\'s Title: ') # Read "CD Title" user input
43.         strMail = input('Enter the Artist\'s Name: ') # Read "CD Name" user input
44.         intID = int(strID)
45.         dicRow = {'id': intID, 'title': strName, 'artist': strMail} # New CD data dictionary
46.         lstTbl.append(dicRow) # Append the new CD data dictionary to the list
47.         print()
48.     elif strChoice == 'i':
49.         # 4. Display current inventory
50.         print('ID, Title, Artist')
```

```

51.         for row in lstTbl:
52.             print(*row.values(), sep = ', ') # Unpacking the list with *
53.         print()
54.     elif strChoice == 'd':
55.         # 5. Delete CD from inventory
56.         deleteRow = int(input('Which CD entry (row) do you like to delete?: '))
57.         lstTbl.remove(lstTbl[deleteRow-1]) # Remove Xth entry in CD data inventory
58.     elif strChoice == 's':
59.         # 6. Save inventory to a text file CDInventory.txt
60.         objFile = open(strFileName, 'w') # Open the CDInventory.txt file.
61.         # "w" mode: An existing file with the same name will be erased and it will write over.
62.         for row in lstTbl:
63.             strRow = ''
64.             for item in row.values():
65.                 strRow += str(item) + ','
66.             strRow = strRow[:-1] + '\n' # Remove ',' at the end of each row and add a new line
67.             objFile.write(strRow) # Write each DC data row to the CDInventory.txt file
68.         objFile.close() # Close the CDInventory.txt file
69.     else:
70.         print('Please choose either l, a, i, d, s or x!')

```