Eliana Arias-Dotson
Created: 2020, February 24
Final Editing: February 24 2020
Foundations of Programming, Python
Assignment 05

Introduction

On this module the goal was to implement the use of list, dictionaries into an user menu.
I continue to divide each routine into individual ones to later incorporate it into the bigger codes, however I struggle making it all work. I am not satisfied with the code as it is but I think I need to dedicate time to understand it better.

What we are learning

Learning more about listing and dictionaries provides options to create versatility on the code. Exploring the material and the labs helped understanding the possibility of use when coding.

Defining the template for Spider and starting to implement organization into the code makes easy to not only read the code but also helps on splitting the assignment into simple parts.

Approach to the assignment

Previously, I have split the tasks into individual parts to make it less overwhelming, that seemed to help me work and make progress. This particular assignment seemed simpler however I had difficulty implementing what I was hoping to do specially for the "deleting" script

Troubles making it work

I struggle with this assignment. When looping through the lstTbl I printed the items, and look at the values store on the dictionary running several ideas and try to implement it but just did not work

Concepts and Applications

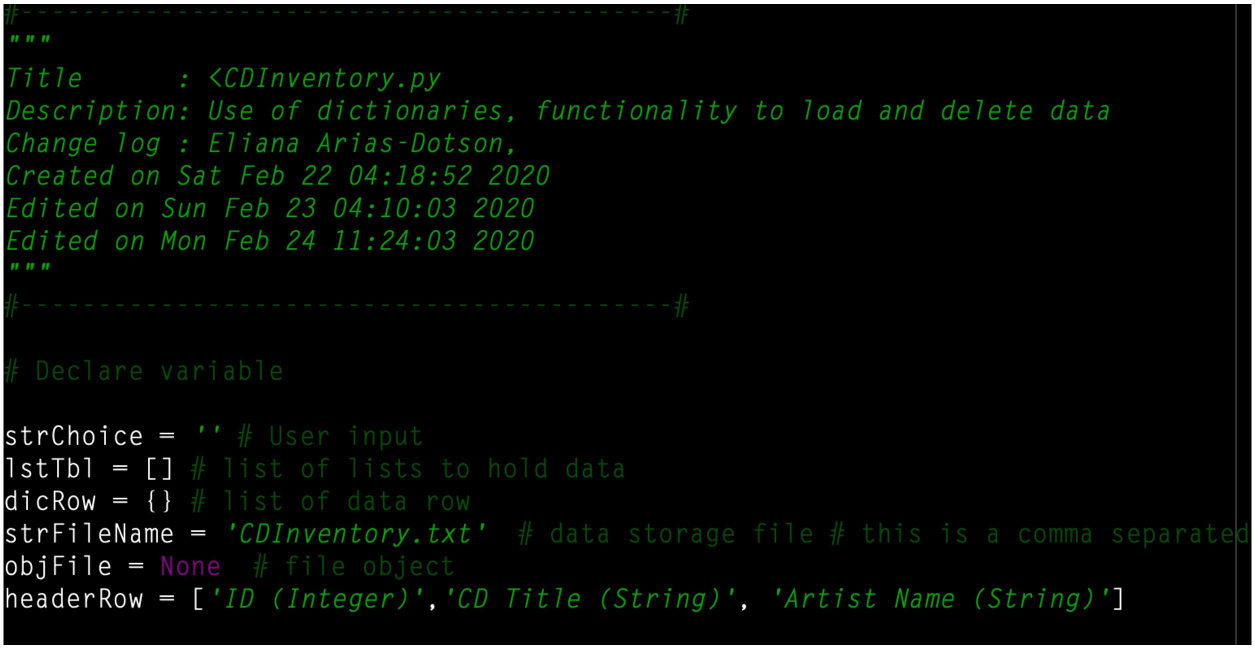
I believe practice is a "must" on coding like anything else. Being new at coding, even though the concepts seem simple, the application is something that I hope will come easier.

The code

Considering the main body of the code was given I thought implementing the small parts required would have been easier for me, however I could not get the code to work when deleting the item.

Header and Variables

In this part of the code we used the template customized and defined the list of data row as a dictionary instead of a list. The rest of the variables were predefined and provided on the starter script.

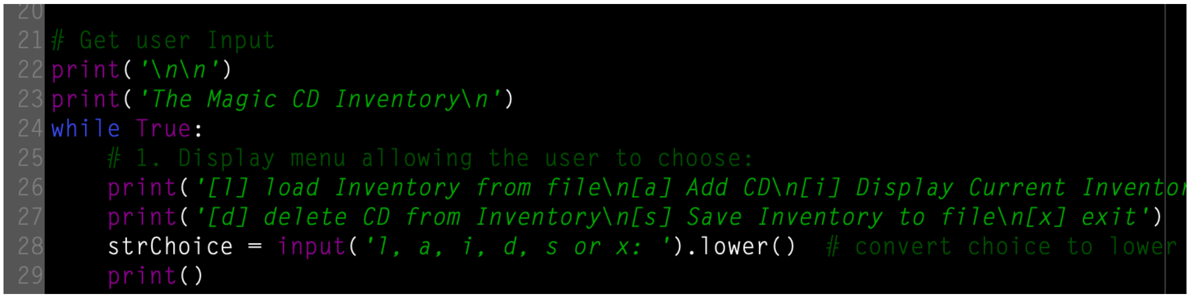


```
#------------------------------------------#
"""
Title       : <CDInventory.py
Description: Use of dictionaries, functionality to load and delete data
Change log : Eliana Arias-Dotson,
Created on Sat Feb 22 04:18:52 2020
Edited on Sun Feb 23 04:10:03 2020
Edited on Mon Feb 24 11:24:03 2020
"""
#------------------------------------------#

# Declare variable

strChoice = '' # User input
lstTbl = [] # list of lists to hold data
dicRow = {} # list of data row
strFileName = 'CDInventory.txt'  # data storage file # this is a comma separated
objFile = None  # file object
headerRow = ['ID (Integer)','CD Title (String)', 'Artist Name (String)']
```

Figure 1. Template created in Spyder and Declaration of variables for code

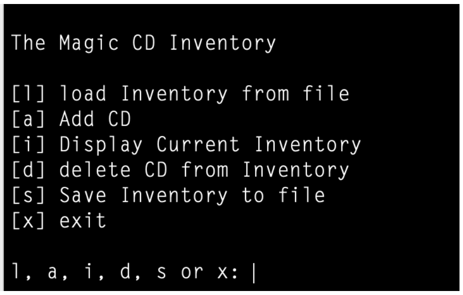
Welcome to the Menu and User Options

In this part of the starter code the script displays the options to choose from in a simplified way by using the “\n separator. Choices for user input are simplified to letters and formatted to “lower” case.



```
# Get user Input
print('\n\n')
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 Invento
    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
    print()
```

Figure 2. User input and display of options to chose from .

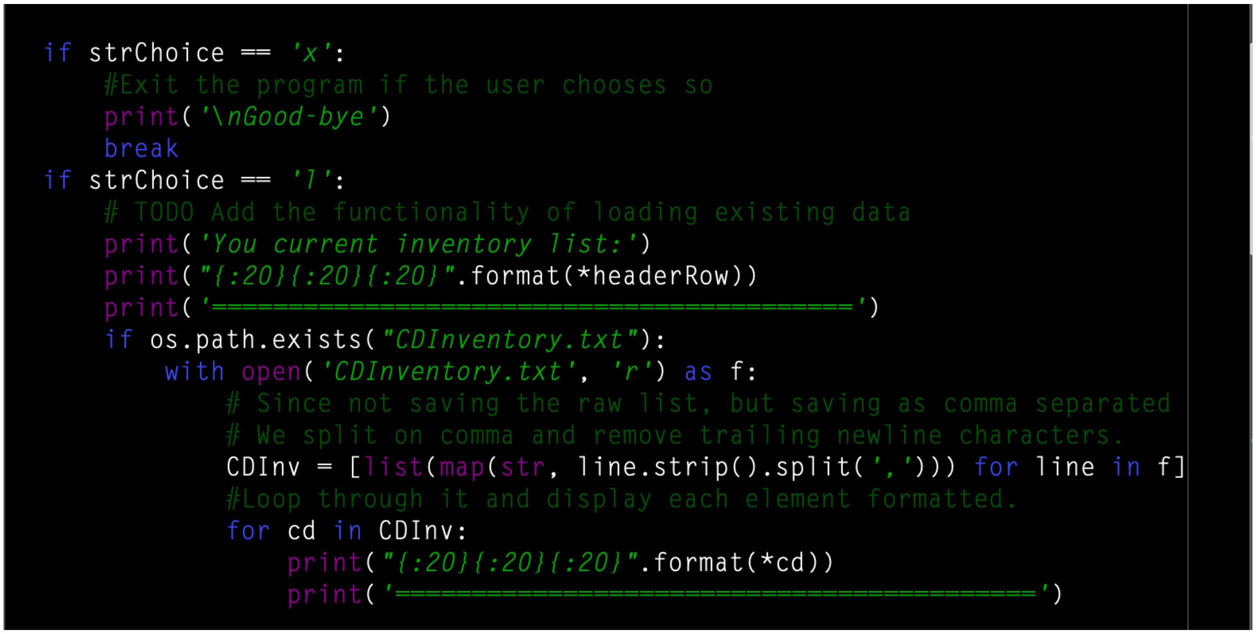


```
The Magic CD Inventory

[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: |
```

Figure 3. Display of options for menu

Choice Exit and Loading Data (x and l)

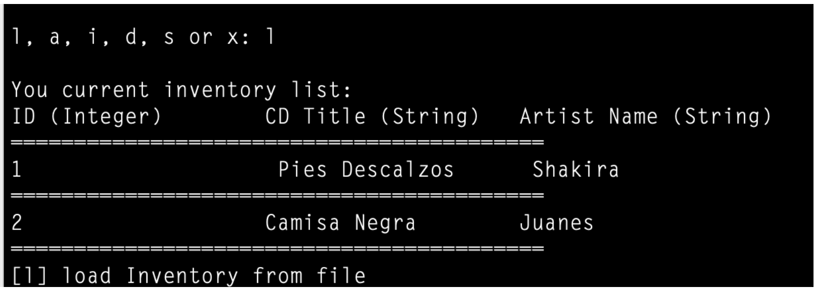
The first option provided is a conditional IF to select 'x' to exit the code. I simply add the print ('\Good bye') incorporated from the solution on Assignment 04.



```
if strChoice == 'x':
    #Exit the program if the user chooses so
    print('\nGood-bye')
    break
if strChoice == 'l':
    # TODO Add the functionality of loading existing data
    print('You current inventory list:')
    print("{:20}{:20}{:20}".format(*headerRow))
    print('==========================================')
    if os.path.exists("CDInventory.txt"):
        with open('CDInventory.txt', 'r') as f:
            # Since not saving the raw list, but saving as comma separated
            # We split on comma and remove trailing newline characters.
            CDInv = [list(map(str, line.strip().split(','))) for line in f]
            #Loop through it and display each element formatted.
            for cd in CDInv:
                print("{:20}{:20}{:20}".format(*cd))
                print('==========================================')
```

Figure 4. Python code for options "x" and "l" for exiting and loading data

For the conditional IF the choice provided was 'l' to load the data , I used the improved version of my Assignment 04 attending all sugestions from Mr Kloss. My output from Loading file:

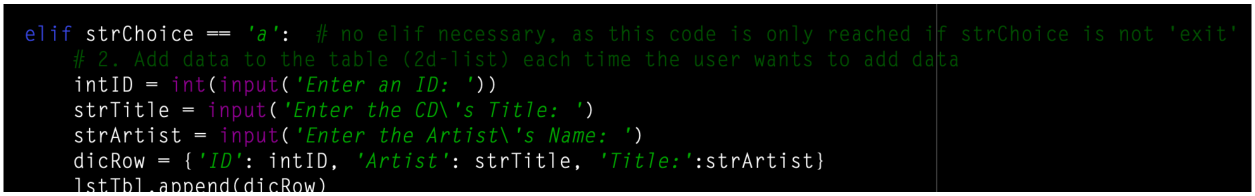


```
l, a, i, d, s or x: l

You current inventory list:
ID (Integer)        CD Title (String)   Artist Name (String)
==========================================
1                    Pies Descalzos      Shakira
==========================================
2                   Camisa Negra        Juanes
==========================================
[l] load Inventory from file
```

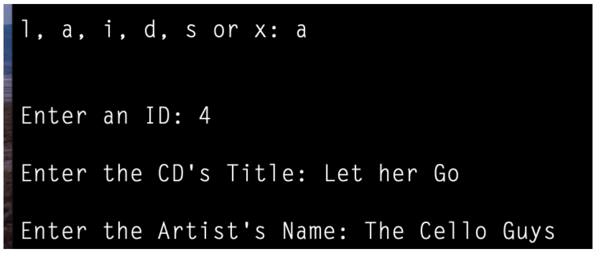
Figure 5. Example of output when loading data from inventory.

Choice Adding data:

The script to add data was provided and here the values on list were matched to the keys that exist in the Magic Inventory. the Values added to the each key from user input are appended to the lstTable by using the append function.
Example of the options asked to user are simply inputs as strings, except ID, which is defined as integer:



```
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
    intID = int(input('Enter an ID: '))
    strTitle = input('Enter the CD\'s Title: ')
    strArtist = input('Enter the Artist\'s Name: ')
    dicRow = {'ID': intID, 'Artist': strTitle, 'Title:':strArtist}
    lstTbl.append(dicRow)
```

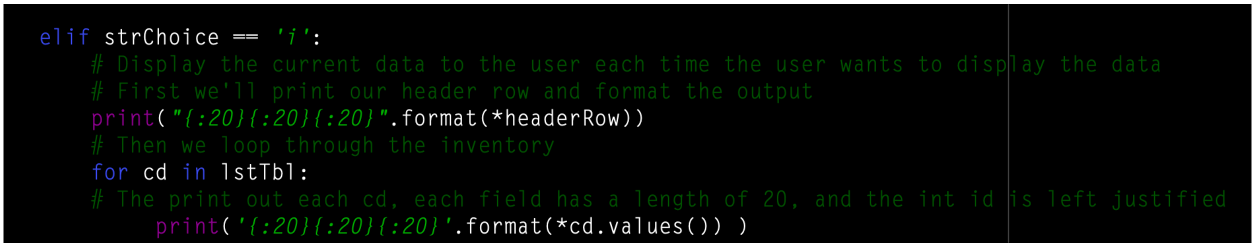


```
l, a, i, d, s or x: a


Enter an ID: 4

Enter the CD's Title: Let her Go

Enter the Artist's Name: The Cello Guys
```

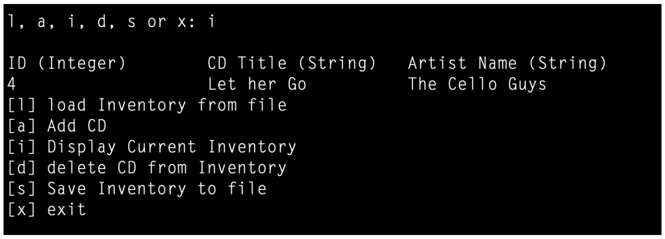
Figure 6. Display of code for ‘adding - a’ new data to the list and example of user input when asked to add the new data.

Choice Displaying data:

When chosing to display the data, I followed the example and suggestions provided in assignment 04 by formating the display using the function .format and adding some indentation and to display the entries in headerRow and the values stored in my lstTbl.



```
elif strChoice == 'i':
    # Display the current data to the user each time the user wants to display the data
    # First we'll print our header row and format the output
    print("{:20}{:20}{:20}".format(*headerRow))
    # Then we loop through the inventory
    for cd in lstTbl:
    # The print out each cd, each field has a length of 20, and the int id is left justified
        print('{:20}{:20}{:20}'.format(*cd.values()) )
```



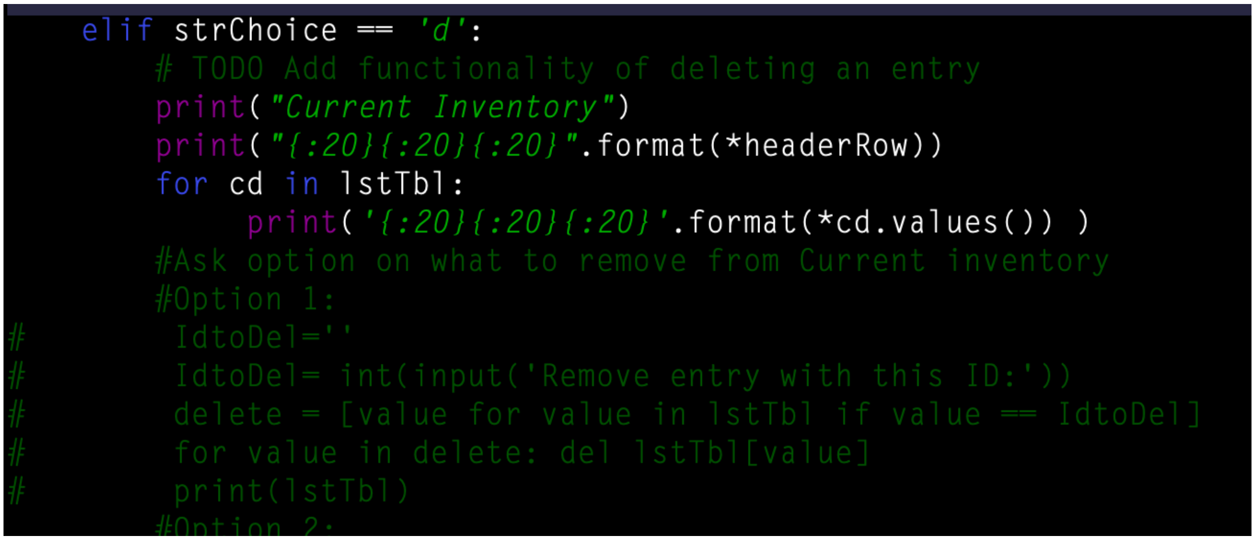
```
l, a, i, d, s or x: i

ID (Integer)        CD Title (String)   Artist Name (String)
4                   Let her Go          The Cello Guys
[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 7. Display of code for ‘displaying - i’ new data added to the list and example of outputs when doing so.

Deleting input

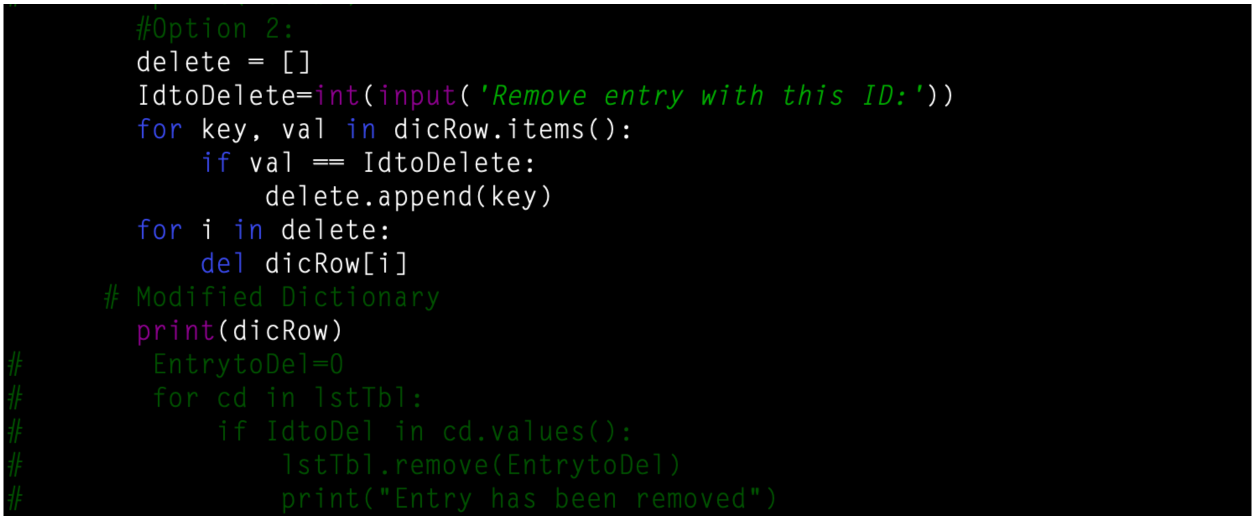
This turned out to be the task I had more trouble with. I tried several options but none of them worked to actually delete the entry. I'm obviously missing or missunderstanding how to "find "a value from the row and remove the row associated to it.

This part was unsuccessful for me, but I wanted to show a couple of possibilities I tried:



```
    elif strChoice == 'd':
        # TODO Add functionality of deleting an entry
        print("Current Inventory")
        print("{:20}{:20}{:20}".format(*headerRow))
        for cd in lstTbl:
             print('{:20}{:20}{:20}'.format(*cd.values()) )
        #Ask option on what to remove from Current inventory
        #Option 1:
#        IdtoDel=''
#        IdtoDel= int(input('Remove entry with this ID:'))
#        delete = [value for value in lstTbl if value == IdtoDel]
#        for value in delete: del lstTbl[value]
#        print(lstTbl)
        #Option 2:
```

Figure 8.a. Attempt 1 to delete values from the lstTbl .



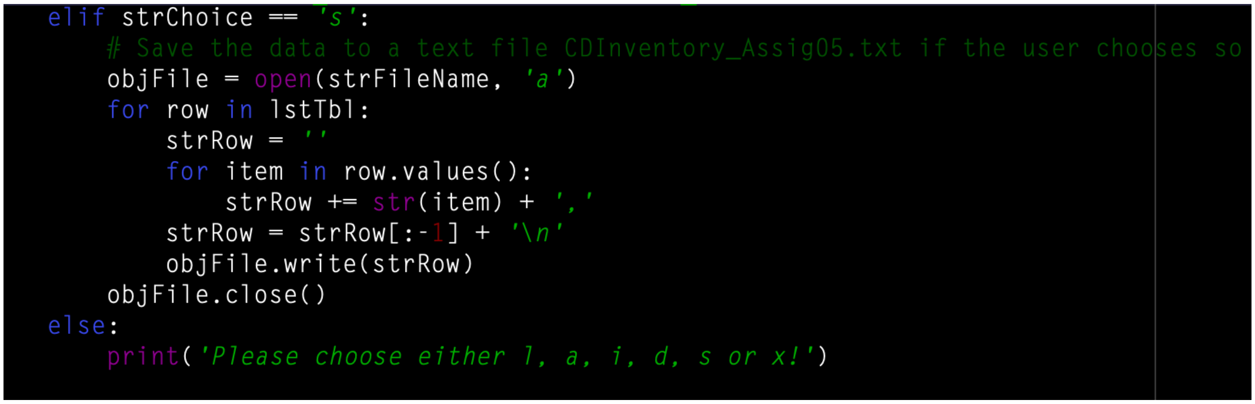
```
        #Option 2:
        delete = []
        IdtoDelete=int(input('Remove entry with this ID:'))
        for key, val in dicRow.items():
            if val == IdtoDelete:
                delete.append(key)
        for i in delete:
            del dicRow[i]
      # Modified Dictionary
        print(dicRow)
#        EntrytoDel=0
#        for cd in lstTbl:
#            if IdtoDel in cd.values():
#                lstTbl.remove(EntrytoDel)
#                print("Entry has been removed")
```

Figure 8.b. Attempt 2 to delete values from the lstTbl .

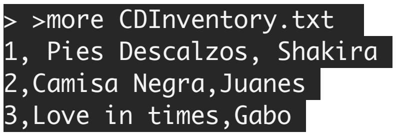
Neither attempt solve my problem. I tried , the code does not provide errors but when requested to display the entry that was supposed to be removed nothing had changed.

Saving to file

this part of the code was also provided, we access the file to append the new data using the ‘a’ option, access each row on that lstTbl and iterate over the items on the row.values since they are located on the dictionary, the iteration goes through each strRow, accounting for each item separated by ‘,’/ then it writes to the row excluding the last ‘,’ and adding a space finally it writes to file and closes it (since in this particular option the os.path was not used



```
    elif strChoice == 's':
        # Save the data to a text file CDInventory_Assig05.txt if the user chooses so
        objFile = open(strFileName, 'a')
        for row in lstTbl:
            strRow = ''
            for item in row.values():
                strRow += str(item) + ','
            strRow = strRow[:-1] + '\n'
            objFile.write(strRow)
        objFile.close()
    else:
        print('Please choose either l, a, i, d, s or x!')
```



```
> >more CDInventory.txt
1, Pies Descalzos, Shakira
2,Camisa Negra,Juanes
3,Love in times,Gabo
```

GitHub

LInk to the repository with the assignment and document are located under:
https://github.com/elidot/Assignment_05

Lessons learned:

Coding takes focus and understanding of what each new concept does and how it works. Visualizing the variables on spyder and using the print input I understood what was getting accomplished by the provided code. Unfortunately I did not succed deleting the input as requested but I am hopeful to clarify this concept .