Nary Hang

February 26, 2022

Introduction to Programming (Python)

Assignment 05

How to modify an existing CD Inventory program

# Introduction

The objective for this fifth assignment is focused on modifying an existing CD Inventory program 2D data structure to use dictionaries as the inner data type. The process to complete the assignment includes modifying the inner structure with dictionaries, adding a function to load existing data and creating a function to delete an entry. This knowledge document will detail the steps taken to modify the program.

# Steps Involved

After opening script in Anaconda Spyder, we were tasked with modifying an existing CD Inventory program, which asks a user to choose from a menu. The first option on the menu allows the user to load the inventory from a file. The other choices on the menu includes adding information on a CD such as ID number, CD title, and artist name as well as displaying current inventory, deleting a CD from inventory, saving the inventory to a file, and exiting.

The following steps were taken:

1. The first step is to modify the inner structure with dictionaries. I created an empty dictionary list variable, dicRow {}. I modified the add CD option menu with the new variable to add keys and values for the dictionary. Once the user add data, it will append to lstTbl. See below for the following code:

strID = input('Enter an ID: ')

strTitle = input('Enter the CD**\'**s Title: ')

strArtist = input('Enter the Artist**\'**s Name: ')

intID = int(strID)

*# Dictionary*

dicRow = {'ID': intID, 'CD TITLE': strTitle, 'ARTIST': strArtist}

stTbl.append(dicRow)

**Figure 1:**

*Results after selecting add CD option from the menu in Spyder*

*Text

Description automatically generated*

1. The next step is to modify the code to display the current data to the user. I updated the for loop to append from rows in lstTbl and used unpacking code to print data. See below for the following code:

print('ID, CD TITLE, ARTIST')

*# Cycling and unpacking*

**for** row **in** lstTbl:

print(\*row.values(), sep = ', ')

**Figure 2:**

*Results after selecting display current inventory from the menu in Spyder*

Graphical user interface, text

Description automatically generated

1. The next step is to modify the code to save the existing data to a text file, CDInventory.txt. First, the text file has to be opened to write. The CDInventory.txt file is in strFileName variable. The for loop is initiated to go through rows from lstTbl to an empty file cabinet, strRow. Another for loop is created for processing to store the items. See below for the following code:

objFile = open(strFileName, 'w')

**for** row **in** lstTbl:

*# Empty file cabinet*

strRow = ''

*# Process*

**for** item **in** row.values():

strRow += str(item) + ','

*# Cutting trailing comma*

strRow = strRow[:-1] + '**\n**'

objFile.write(strRow)

objFile.close()

**Figure 3:**

*Results after selecting saving inventory to file from the menu in Spyder*

Graphical user interface, text

Description automatically generated

**Figure 3.1**

*Results in CDInventory.txt file*

Graphical user interface, text, application

Description automatically generated

1. The next step is to add code when a user selects the option to load inventory from file. The code starts by clearing data from an existing table. Next, it opens the text file to read and cycle through rows stripping and splitting white spaces to form a dictionary list. The dictionary list appends to lstTbl. See code below.

*# Clear existing table*

lstTbl.clear()

*# Read data*

objFile = open(strFileName, 'r')

**for** row **in** objFile:

*# Creating a dictionary out of the file*

lstRow = row.strip().split(',')

*# Columns*

dicRow = {'ID': lstRow[0], 'CD TITLE': lstRow[1], 'ARTIST': lstRow[2]}

lstTbl.append(dicRow)

objFile.close()

**Figure 4:**

*Results after selecting multiple times to load inventory from file in Spyder*

Graphical user interface, text

Description automatically generated

1. The next step is to add a function to delete a CD from the inventory. I printed out the current inventory and created a variable for user to input an ID for deletion. Next, I created a for loop to pull existing data to match ID with deletion input. Unfortunately, this code is not working. See below.

print('ID, CD TITLE, ARTIST')

*# Displaying list*

**for** row **in** lstTbl:

print(\*row.values(), sep = ', ')

*# Creating variable deletion and asking for ID input*

lstDel = input('What CD would you like to delete?')

**for** row **in** lstTbl:

**if** row['ID'] == lstDel:

**del** lstDel

**Figure 5:**

*Results after selecting delete CD from inventory in Spyder.*

*\*Note: Code is not working*

*Graphical user interface, text

Description automatically generated*

1. After successfully running the program in Terminal, see below for screenshots:

**Figure 6:**

*Results after running python script in Terminal*

Text

Description automatically generated

**Figure 6.1:**

*Results after selecting Option a in Terminal*

Text

Description automatically generated

**Figure 6.2:**

*Results after selecting Option s in Terminal*

Text

Description automatically generated

**Figure 6.3:**

*Results after selecting Option i in Terminal*

Text

Description automatically generated

**Figure 6.4:**

*Results after selecting Option l in Terminal*

Text

Description automatically generated

**Figure 6.5:**

*Results after selecting Option d in Terminal*

*\*Note code is not working*

*Graphical user interface, website

Description automatically generated*

1. After completing this assignment, the files are posted in GitHub. Here is the link to the repository:

**https://github.com/naryhang/Assignment\_05**

# Summary

I had a setback in this assignment. I had trouble adding code to delete CD from inventory. I created a new variable, user input and for loop. It looks like I’m missing something crucial.

In completing this assignment, I gained experience in modifying existing code to create a dictionary. Once I created the dictionary, I learned that I had to make adjustments to the other codes for it to work. Also, I learned on how to organize my codes in using the correct indentition and variables are stated at the top.

# Appendix