

## Assignment 7

### Introduction

Assignment 7 is an extension of our CD inventory script. In this assignment however, we save and load data using binary instead of plain text files. We also added structured error handling so that if the user inputs something that would normally end the program, the program instead prints an error message to the user and allows them to 'try again.' For the assignment, I first tried to add the structured error handling using the try-except method. I tried to add a try-except in most locations where a user is asked to input data. These included the functions 'add\_newcd', 'delete\_cd', 'read\_file' (in case no file is available to be read at the directory). Note that I don't think that an error could ever be raised under 'add\_newcd' because as far as I know any character can be entered at each of the requests. However, I believe I heard in class that it's often not a bad idea to include more 'off ramps' (my phrase) than less in case something happens that you haven't thought of that the user does. Note further I'm not sure I have the exact syntax correct for some of my exception handling because when I test the script and purposefully type something I know to be incorrect it doesn't necessarily print my error message (then again, the program doesn't crash). Finally, I didn't add error handling for the function 'menu\_choice' because that block of code already uses "while choice not in" which seems like it is already taken care of.

Besides error handling, we also are trying to transition from saving in 'human readable' text files to saving using binary data files. In this I think I struggled and will need to revisit. I first changed the strFileName from .txt to .dat in the Data portion of the program. And I changed a text file I had already saved to a .dat file in the directory. I changed any 'r' or 'w' mode to 'rb' and 'wb'. I tried to modify code to simplify the reading and writing using binary, however when I go to 'save,' my program crashes. When I load inventory the program doesn't crash, but I don't see it loading what I would expect.

For the assignment, we performed our own searches for structured error handling and saving in binary. My search terms are in the Appendix below.

Below is a view of the program running in Spyder. When I try to save, I get an error:

```
Console 1/A x

Which operation would you like to perform? [l, a, i, d, s or x]: a

Enter ID: 2

What is the CD's title? Boy

What is the Artist's name? U2
===== The Current Inventory: =====
ID  CD Title (by: Artist)

2   Boy (by:U2)
=====
Menu

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

Which operation would you like to perform? [l, a, i, d, s or x]: s

===== The Current Inventory: =====
ID  CD Title (by: Artist)

2   Boy (by:U2)
=====

Save this inventory to file? [y/n] y
===== The Current Inventory: =====
ID  CD Title (by: Artist)

2   Boy (by:U2)
=====
Traceback (most recent call last):

  File "C:\_FDNProgramming\Assignment07\CDInventoryAlt1.py", line 84, in <module>
    FileProcessor.write_file(strFileName, lstTbl)

  File "C:\_FDNProgramming\Assignment07\CDInventoryAlt1.py", line 112, in write_file
    pickle.dump(lstTable, objFile)

NameError: name 'lstTable' is not defined
```

Figure 1. Running in Spyder. The script rarely crashes except when I attempt to save.

Below is a screenshot of the program running in Anaconda console:

```
Anaconda Prompt (Anaconda3) - python CDInventory.py
(base) C:\_FDNProgramming\Assignment07>python CDInventory.py
Menu

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

Which operation would you like to perform? [l, a, i, d, s or x]: i

===== The Current Inventory: =====
ID      CD Title (by: Artist)

=====
Menu

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

Which operation would you like to perform? [l, a, i, d, s or x]: d

===== The Current Inventory: =====
ID      CD Title (by: Artist)

=====
Which ID would you like to delete? 1
===== The Current Inventory: =====
ID      CD Title (by: Artist)

=====
Menu

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

Which operation would you like to perform? [l, a, i, d, s or x]: _
```

Figure 2. Program running in Anaconda console.

## GitHub

For this assignment we are required to upload our python file and this knowledge document to GitHub. The link for my assignment is:

<https://github.com/rolflekve/Assignment07>

## Summary

Assignment 7 was a challenging assignment. I tried to add some error handling, although I'm not sure it's working in every instance. I tried to change to save data in a binary data file (.dat file). I was able to create a data file

## Appendix

Below are web searches based on errors I was seeing, and also for researching error handling and saving using binary data files.

- EOF error: ran out of input

- I searched for this error after I got it after trying my first pickle attempt in the function `read_file`
- What is a dat file
- How do I open a dat file
- Python error handling
  - <https://docs.python.org/3/tutorial/errors.html> The Python.org website actually has some pretty straightforward descriptions of error handling.
  - <https://www.geeksforgeeks.org/python-exception-handling/> Geeks for Geeks similarly has descriptions that are accessible to newbies. There is a 3 minute video at the bottom of the link that reinforces the text.
- Python pickling
  - <https://docs.python.org/3/library/pickle.html> Once again I started with the Python.org site which was not unhelpful. It included a warning, in addition to the warning in Mod07 (pickled data is not encrypted), that indicates, “The pickle module is not secure. Only unpickle data you trust.” A good lesson! Scrolling through the document, it is quickly obvious that the website has much more information than is necessary for beginners.
  - <https://realpython.com/python-pickle-module/> This website has useful information (at least in the top ‘half’ of the page). The website first discusses that pickling is one of three or four serialization options in Python; the other two being ‘json’ and ‘marshal’ (and XML). The point is made that pickle is the option to use for most users.