

# CD Inventory Program with Error Handling Technique and Binary Data Storage

## Introduction

In this module, I learned how to save information in a binary format. This is the same format as it is saved in the memory. This way we use less memory and less processing time. The binary format is not easily readable by the user and it is not encrypted, so it is important not to save sensitive information in binary. Another topic from this module is error handling, it is used to prevent programs from crashing. To handle errors try/except statement is used.

Applying this knowledge to assignment 7, I have created the following program.

## Step-by-Step

In this document, I will go over the changes that I have made since the last assignment06.

First of all, to be able to save data into a binary format, I imported a pickle package, from which I used the next functions: load() and dump(). Also, I changed the data storage file format into .dat

```
1  #-----#
2  # Title: CDInventory.py
3  # Desc: Working with classes and functions.
4  # Change Log: (Who, When, What)
5  # DBiesinger, 2030-Jan-01, Created File
6  # MKot, 2022-Nov-18, Modified the code, added functions
7  # MKot, 2022-Nov-24, Modifies code: added error handling, binary data storage
8  #-----#
9
10 import pickle
11
12 # -- DATA -- #
13 strChoice = '' # User input
14 lstTbl = [] # list of lists to hold data
15 dicRow = {} # list of data row
16 strFileName = 'CDInventory.dat' # data storage file
17 objFile = None # file object
18
```

To save and then load the file data I changed two functions from FileProcessor class. The first one is read\_file(). I used a with/as statement to open a file and the character 'rb' for readable mode. Using pickle.load() function I loaded the data from the file to the variable loadFile. (I moved table.clear() function from the top line to line 81, this was needed for error handling, which I am going to explain later.) After that, I started my for loop that goes through the list of dictionaries from the file and assigns the right values to the keys in dicRow variable. The last line adds each dictionary to the table, creating a list of dictionaries.

```

65     @staticmethod
66     def read_file(file_name, table):
67         """Function to manage data ingestion from file to a list of dictionaries
68
69         Reads the data from file identified by file_name into a 2D table
70         (list of dicts) table one line in the file represents one dictionary row in table.
71
72         Args:
73             file_name (string): name of file used to read the data from
74             table (list of dict): 2D data structure (list of dicts) that holds the data during runtime
75
76         Returns:
77             None.
78         """
79         #table.clear() # this clears existing data and allows to load data from file
80         with open(file_name, 'rb') as objFile:
81             loadFile = pickle.load(objFile)
82             table.clear()
83             for i in range(len(loadFile)):
84                 dicRow = {'ID':loadFile[i]['ID'], 'Title':loadFile[i]['Title'], 'Artist': loadFile[i]['Artist']}
85                 table.append(dicRow)
86

```

The second function is write\_file(). Again as it is a binary format, I used the character 'wb' to open the file and rewrite it and I used pickle.dump() to save the data to the file.

```

86     @staticmethod
87     def write_file(file_name, table):
88         """Function to manage data ingestion from list of dictionaries to a file
89
90         Reads the data from file dictionary by table and save line by line into the file
91         file_name.
92
93         Args:
94             file_name (string): name of file where data is saved to
95             table (list of dict): 2D data structure (list of dicts) that holds the data that needs to be saved.
96
97         Returns:
98             None.
99         """
100         # TODO: Add code here
101         with open(file_name, 'wb') as objFile:
102             pickle.dump(table, objFile)
103

```

I also added an error handling try/except statement to handle two types of errors. The first one is when the user enters a string instead of an integer for ID. For that, I added the following while loop that asks the user to enter the ID, checks if the entered value is an integer, in case it is not, prints out "This is not an integer. Try again!" and starts the loop again.

```

159     def add_data():
160         """Gets user input new CD
161
162
163         Args:
164             none
165
166         Returns:
167             strID, strTitle, stArtist: string of the users input
168
169         """
170         while True:
171             strID = input('Enter ID: ').strip()
172             try:
173                 strID = int(strID)
174                 break
175             except:
176                 print('This is not an integer. Try again!')
177             strTitle = input('What is the CD\'s title? ').strip()
178             stArtist = input('What is the Artist\'s name? ').strip()
179             return (strID, strTitle, stArtist)

```

A similar approach is used to capture the same error with the delete function when the user enters a string instead of an integer.

```

222     elif strChoice == 'd':
223         # 3.5.1 get Userinput for which CD to delete
224         # 3.5.1.1 display Inventory to user
225         IO.show_inventory(lstTbl)
226         # 3.5.1.2 ask user which ID to remove
227         while True:
228             try:
229                 intIDDel = int(input('Which ID would you like to delete? ').strip())
230                 break
231             except:
232                 print('\nPlease enter only integers. Try again!\n')
233         # 3.5.2 search thru table and delete CD
234         # TODO: move processing code into function
235         DataProcessor.delete_data(intIDDel, lstTbl)
236         IO.show_inventory(lstTbl)
237         continue # start loop back at top.

```

Another error is when there is no file yet created, but the user chooses 'l' to load the data from the file. To handle this error, I added a try/except statement for the FileProcessor.read\_file() function, if the file is not yet created the program prints out "No data has been saved yet. There is nothing to load." and returns to the main menu without clearing out the memory as I moved table.clear() function (that I mentioned above) below the potential error moment.

```

194     # 3.2 process load inventory
195     if strChoice == 'l':
196         print('WARNING: If you continue, all unsaved data will be lost and the Inventory will be reloaded from file. Otherwise reload will be canceled.')
197         strYesNo = input('type \'yes\' to continue and reload from file. otherwise reload will be canceled. ')
198         if strYesNo.lower() == 'yes':
199             print('reloading...')
200             try:
201                 FileProcessor.read_file(strFileName, lstTbl)
202                 IO.show_inventory(lstTbl)
203             except:
204                 print('\nNo data has been saved yet. There is nothing to load.\n')
205         else:
206             input('canceling... Inventory data NOT reloaded. Press [ENTER] to continue to main menu. ')
207             IO.show_inventory(lstTbl)
208         continue # start loop back at top.

```

## Results

Here are some results from running this program in Spyder and Terminal.

```

[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]: l

WARNING: If you continue, all unsaved data will be lost and the Inventory will be reloaded from file. Otherwise reload will be canceled.
type 'yes' to continue and reload from file. otherwise reload will be canceled.

No data has been saved yet. There is nothing to load.

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]: a

Enter ID: qw
This is not an integer. Try again!
Enter ID: q
This is not an integer. Try again!Enter ID: 1

What is the CD's title? The Big Wheel
What is the Artist's name? Runrig
===== The Current Inventory: =====
ID  CD Title (by: Artist)
1   The Big Wheel (by:Runrig)
=====

Which operation would you like to perform? [l, a, i, d, s or x]: a
Enter ID: 3
What is the CD's title? MammaMia
What is the Artist's name? ABBA
===== The Current Inventory: =====
ID  CD Title (by: Artist)
1   The Big Wheel (by:Runrig)
2   Bad (by:Michael Jackson)
=====

Save this inventory to file? [y/n] y

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]: a
Enter ID: 3
What is the CD's title? MammaMia
What is the Artist's name? ABBA
===== The Current Inventory: =====
ID  CD Title (by: Artist)
1   The Big Wheel (by:Runrig)
2   Bad (by:Michael Jackson)
3   MammaMia (by:ABBA)
=====

Save this inventory to file? [y/n] y

Menu

```

```

ID  CD Title (by: Artist)

1  The Big Wheel (by:Runrig)
2  Bad (by:Michael Jackson)
3  MammaMia (by:ABBA)
=====
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]: l

WARNING: If you continue, all unsaved data will be lost and the Inventory
will be reloaded. type 'yes' to continue and reload from file. otherwise reload will be
reloading...
===== The Current Inventory: =====
ID  CD Title (by: Artist)

1  The Big Wheel (by:Runrig)
2  Bad (by:Michael Jackson)
=====
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]: a

Enter ID: 3
What is the CD's title? MammaMia
What is the Artist's name? ABBA
===== The Current Inventory: =====

```

```

1  The Big Wheel (by:Runrig)
2  Bad (by:Michael Jackson)
3  MammaMia (by:ABBA)
=====
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)

1  The Big Wheel (by:Runrig)
2  Bad (by:Michael Jackson)
3  MammaMia (by:ABBA)
=====
Save this inventory to file? [y/n] y
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)

1  The Big Wheel (by:Runrig)
2  Bad (by:Michael Jackson)
3  MammaMia (by:ABBA)
=====
Which ID would you like to delete? 3

```

```

The CD was removedWhich operation would you like to perform? [l, a, i, d, s or x]: s
===== The Current Inventory: =====
ID  CD Title (by: Artist)

1  The Big Wheel (by:Runrig)
2  Bad (by:Michael Jackson)
=====
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)

1  The Big Wheel (by:Runrig)
2  Bad (by:Michael Jackson)
=====
Save this inventory to file? [y/n] y
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]: x

```

[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]: l

WARNING: If you continue, all unsaved data will be lost and the Inventory re-load type 'yes' to continue and reload from file. otherwise reload will be canceledyes reloading...

No data has been saved yet. There is nothing to load.

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]: a

```
Enter ID: 1
What is the CD's title? The Big Wheel
What is the Artist's name? Runrig
===== The Current Inventory: =====
ID      CD Title (by: Artist)
```

```
1      The Big Wheel (by:Runrig)
=====
```

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]: a

```
Enter ID: 2
What is the CD's title? Bad
What is the Artist's name? Michael Jackson
===== The Current Inventory: =====
ID      CD Title (by: Artist)
```

```
1      The Big Wheel (by:Runrig)
2      Bad (by:Michael Jackson)
=====
```

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

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

```
1      The Big Wheel (by:Runrig)
2      Bad (by:Michael Jackson)
3      MammaMia (by:ABBA)
4      Mamma (by:Mia)
=====
```

Which ID would you like to delete? 4  
The CD was removed

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

```
1      The Big Wheel (by:Runrig)
2      Bad (by:Michael Jackson)
3      MammaMia (by:ABBA)
=====
```

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)
```

```
1      The Big Wheel (by:Runrig)
2      Bad (by:Michael Jackson)
3      MammaMia (by:ABBA)
=====
```

Save this inventory to file? [y/n] y

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]: l

WARNING: If you continue, all unsaved data will be lost and the Inventory re-load type 'yes' to continue and reload from file. otherwise reload will be canceledyes reloading...

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

```
1      The Big Wheel (by:Runrig)
2      Bad (by:Michael Jackson)
3      MammaMia (by:ABBA)
=====
```

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]: x

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

Enter ID: 3

What is the CD's title? MammaMia

What is the Artist's name? ABBA

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

```
1      The Big Wheel (by:Runrig)
2      Bad (by:Michael Jackson)
3      MammaMia (by:ABBA)
=====
```

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)
```

```
1      The Big Wheel (by:Runrig)
2      Bad (by:Michael Jackson)
3      MammaMia (by:ABBA)
=====
```

Save this inventory to file? [y/n] y

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]: a

Enter ID: qqqq

This is not an integer. Try again!

Enter ID: /

This is not an integer. Try again!

Enter ID: 4

What is the CD's title? Mamma

What is the Artist's name? Mia

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

```
1      The Big Wheel (by:Runrig)
2      Bad (by:Michael Jackson)
3      MammaMia (by:ABBA)
4      Mamma (by:Mia)
=====
```

# Summary

In this assignment, I worked with binary code and understood how to save data in binary format. I also explored and wrote try/except statements for error handling, so that my program does not crush.

My GitHub: <https://github.com/mustbekot/Assignment07>

Online resources for exception handling:

1. <https://realpython.com/python-exceptions/>
2. [https://www.tutorialspoint.com/python/python\\_exceptions.htm?key=try+except](https://www.tutorialspoint.com/python/python_exceptions.htm?key=try+except)

Online resources for pickling:

1. <https://www.pythonforthelab.com/blog/storing-binary-data-and-serializing/>
2. <https://www.afternerd.com/blog/python-pickle/>