Jaclynn Fallon

Sunday, March 20, 2022

IT FDN 110 B Wi 22: Foundations Of Programming: Python

Assignment 08

Objects & Classes (Part I)

# Introduction

This module introduced Object Oriented Programming (OOP) and the theories and concepts OOP that make it work, including: classes, constructors, attributes, properties, and methods. I found both the idea of an object, and the practical application of working with objects challenging, but I managed to get a script working before the deadline, and the following document outlines how I did it.

# OOP Struggle Bus

This was a very intimidating assignment, and a difficult module. There were lots of pseudocode TODO’s and empty space in the starter script. It was hard to even know where to start, but finally, after reviewing the instructions and the starter code a few times, I started by copying pieces of my code from Assignment 07 into some of the pseudocode TODO’s.

I knew we would be creating objects in this Assignment, and to me, the most logical application of OOP seemed to be to create each CD entry as an object. However, since this was a new concept, I did not try to apply OOP to any other area of the script. Therefore, I decided to leave most of the functions in the Presentation area of the script unchanged from Assignment 07, and I copied them straight over from Assignment 07 with only minor changes.

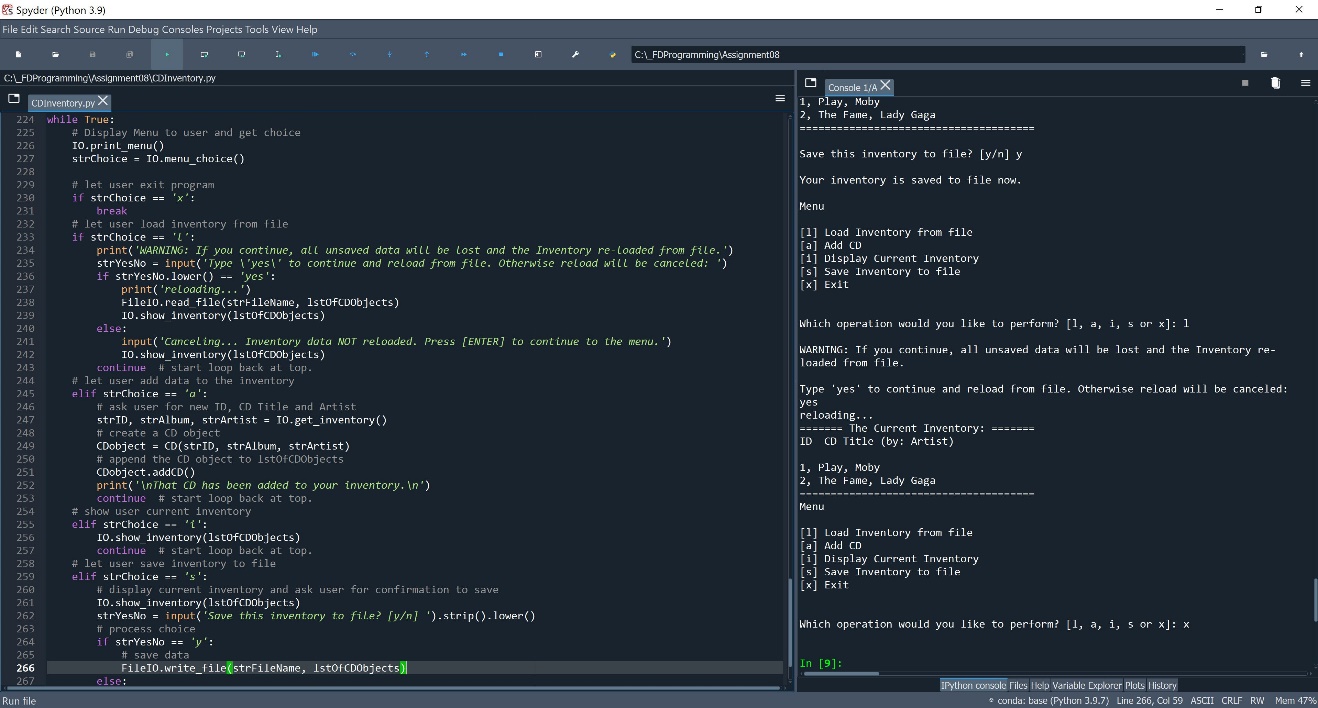


Figure 1 – CD\_Inventory.py running in Spyder for Assignment 08.

However, once that was complete things got more complicated. I decided to move on to working on the CD class, where the main ***object****ive* (haha) of the assignment would be. I figured if I could get that part of the script working, then I could work on writing and reading the list of objects to and from file later. To do this, I actually started by copying some of my code over from Lab 08 E. I had to modify the attributes, properties, and methods, but it was still already configured to instantiate objects with three attributes, similar to what we needed for the assignment.

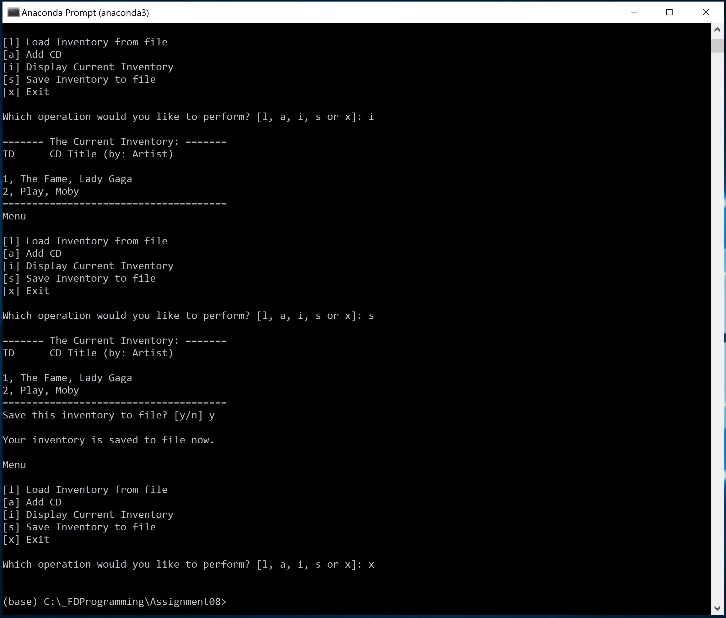


Figure 2 – CD\_Inventory.py running in a terminal.

Once that was complete, I needed to figure out how to receive input from the user and pass those values into the CD class where they would become attributes of a CD object. I contemplated this for a long time, but in the end, what I came up with wasn’t much different from what we did in Assignment 07. Once I had the values from the user, I created a CD object to pass the values to as attributes with the following line of code in the main body of the script:

|  |  |
| --- | --- |
| 1  2 | # create a CD object  CDobject = CD(strID, strAlbum, strArtist) |

Within the CD class I created a method that appended the object to the list of objects, and then I called that method in the main body with the following code:

|  |  |
| --- | --- |
| 1  2 | # append the CD object to lstOfCDObjects  CDobject.addCD() |

I quickly realized I would need to change my ‘display inventory’ code to work with an object instead of a dictionary, but I just created a simple for loop for this, and relied on the custom \_\_str\_\_() method I created to display a presentable string representation of the object.

The next step of the process would be to save the list of objects to file. Since the strFileName variable was set to cdInventory.txt, I figured we should save the attributes of the objects to a text file, rather than pickling the list of objects. To do this, I created another method in the CD class called ‘csv’ to format the object’s attributes in csv format. Then I modified my code in the write\_file function (copied from a previous assignment) to a simple for loop that would loop through the list of objects, invoke the csv method on each one, and same them to the text file in csv format.

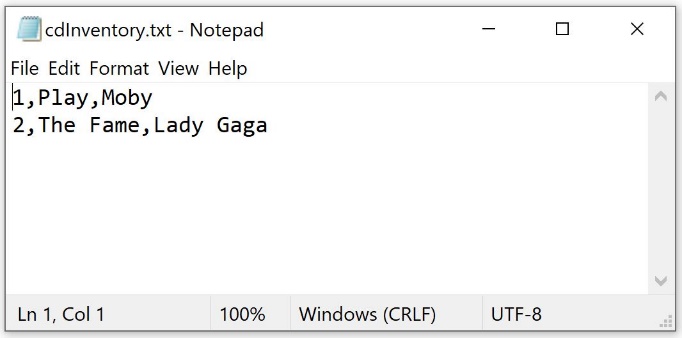


Figure 3 – Some results of cdInventory.py saving to text file.

The final piece of the puzzle was to load the data from the text file, and I ended up following the same strategy I used to get values from user input. Once I read the data from file, I used a for loop to assign each value per line to the variables cd\_id, title, and artist, then passed those values to the CD() class to be used as attributes for a new object, and appended the object to the list. I threw in a couple print statements and displayed the cd inventory to verify the objects were being created successfully and working in other parts of the script.

And finally, I posted my script and knowledge document to GitHub, and here is the link to it: <https://github.com/falloj/Assignment_08>

# Summary

To me, this was by far the most challenging module we’ve had yet. Although in my final script I could verify that I was successfully creating objects, and I was able successfully assign and access their attributes, I’m still not sure I’m using properties and methods completely correctly, and I’m still fuzzy on private vs public attributes and methods, and when to use which one. I think I have a grasp on the abstraction of what an object is, but I’m not fully confident in my application of the theory. I hope that, like we did with functions, we’ll have more opportunities to apply the concepts of OOP in next week’s module.