Emily Martin

03/20/2022

IT FDN 110

Assignment 08

# Introduction

The coursework for Module 08 introduces the concept of object-oriented programming. Module 08 also iterates on previously introduced classes and methods. In this assignment, I used the knowledge gained in Module 08 to iterate on a previously created CD Inventory program by refactoring with object-oriented principles.

# Approach

The following steps document the general approach taken to produce the source code referenced in the Appendix, as well as the reasoning behind a selection of notable syntactical decisions:

1. Based on the course recommendation for IDEs, I created a Python file in Spyder to write my code in. I added a header to capture administrative details.
2. First, I created a list of to-dos in the existing code, and created a separate list of items external to the IDE to track my progress. This allowed me to address each to-do more strategically, and enabled me to add notes to the list to capture additional follow up items.
3. I first added functions to the class CD, including a constructor to instantiate CD objects with.
4. I then brought in the main loop from the previous assignment, knowing that I would need to iteratively make updates to it as I continued to work on the other classes.
5. I then created the class Inventory to manage the CD inventory. I added functions to this class for adding and deleting CDs from the Inventory object.
6. Next, I brought in logic from the previous assignment to populate the FileIO and IO classes with, and updated the logic in both to match the refactored code.
7. As a last step, I surveyed all of the new and existing classes and functions and updated the docstrings where appropriate to accurately capture how each function is intended to perform.
8. Upon completion, I uploaded my assignment to GitHub for review.

# Summary

After iteratively testing the changes applied this code while refactoring with object-oriented principles, I was able to verify that the CD Inventory program maintained the full scope of intended functionality. Additionally, I verified that I was able to both save binary data to a pickle file, and load that data in from the unpickled binary file. In summary, this assignment required knowledge from both current and previous modules to streamline the program.

# Outcome

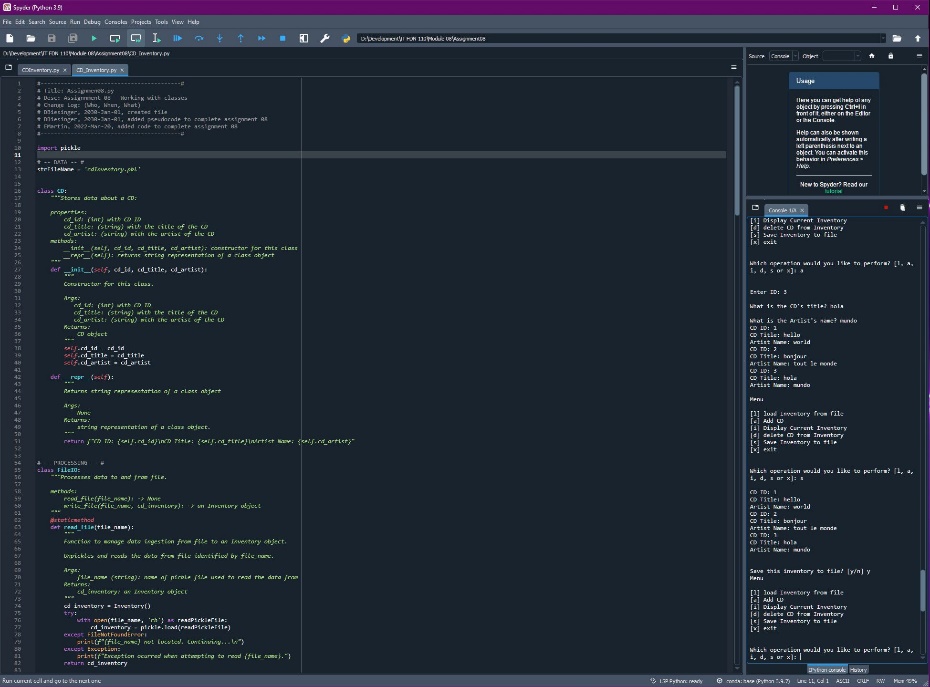


Figure : Capture of code running in Spyder

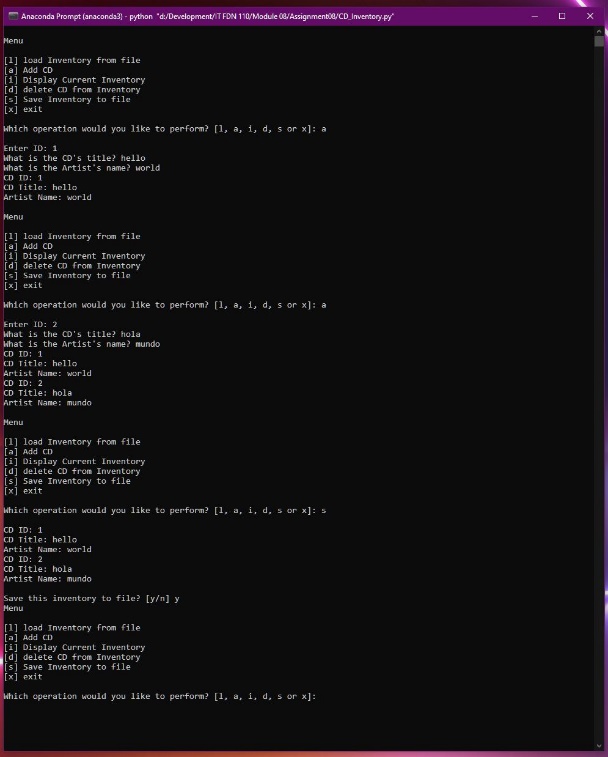


Figure : Capture of code running in a prompt