Nicco Fortes

12/4/22

Foundations of Programming

Module 7

Module 7

# Intro

This week we covered different ways to work with text files, how to read lines from them, and using a “with…as” construct when using them. We then learned about saving and loading binary data, and how we might implement this in our current CD inventory code. Last we covered structured error handling and the advantages of it.

# Module 7

The beginning of this module covered working with text files. Although we have been working with text files for a number of weeks it was a good review of the subject. In particular using the “with…as” construct with the files was an illuminating section as I had seen this implemented in some of my classmates code.

The pickling method made saving and loading files much more straightforward. I see the value in learning the way we saved the data in earlier iterations. Pickling makes the process much easier to follow.

Structured error handling was also introduced. The advantage to handling these errors to control the program and not lose data from the code kicking out is evident. I thought the implementation to be relatively easy as well as the structure can be placed around the existing code.

# Assignment 7

In this assignment we were tasked with adding structured error handling into our existing code and also changing the load/save file functions to handle binary files. For the structured error handling portion, adjustment were made to the following sections:

* Function for loading data to text file
* Function for exporting data to text file
* User input of CD data (integer value error)
* User selection of line to delete (integer value error)

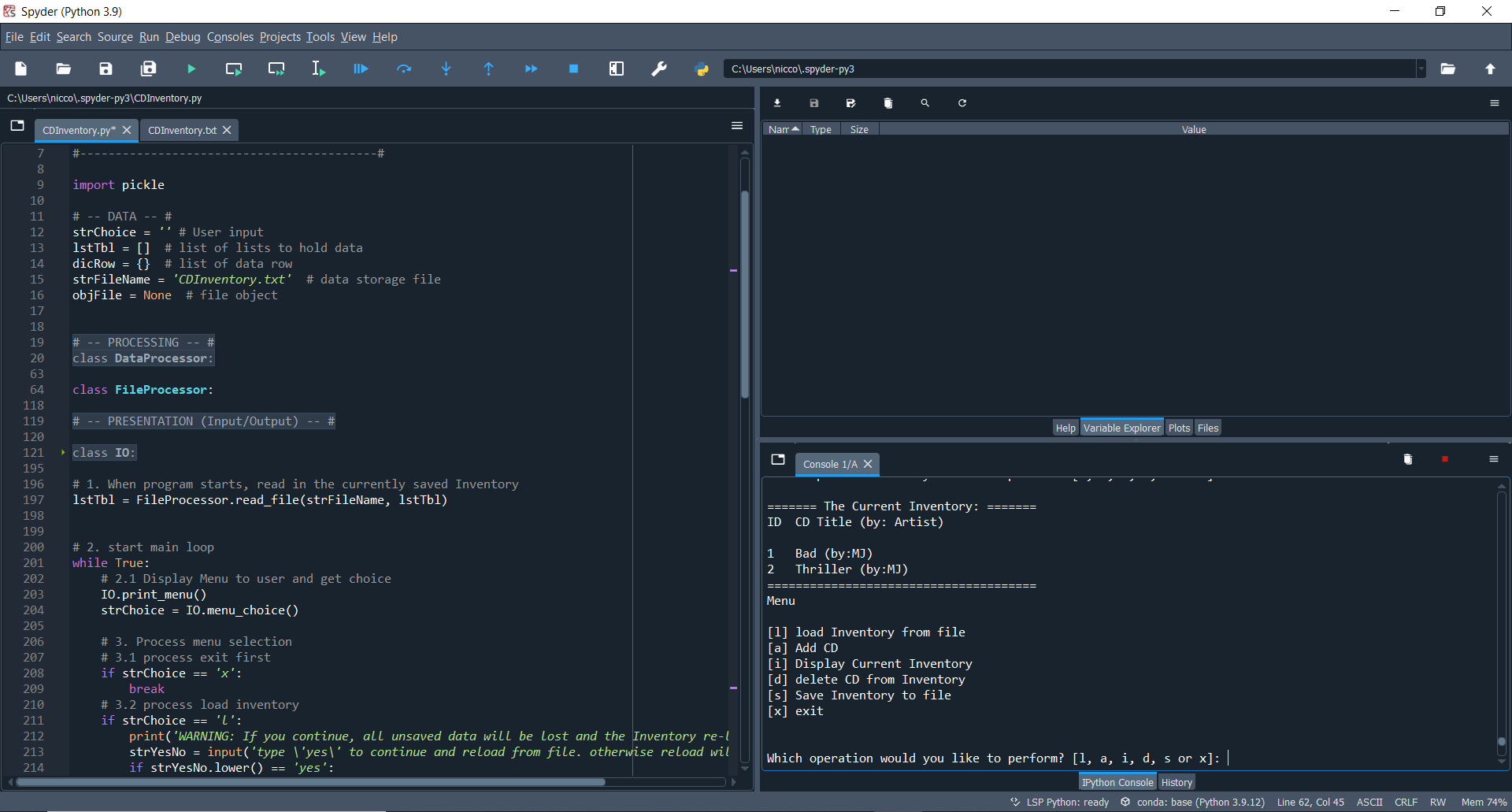


Figure : Script running in Spyder

Graphical user interface, text

Description automatically generated

# Summary

Module 7 was a good introduction into a couple of subjects I was interested in leading up to this week. In particular learning about pickling data was interesting and really take out a lot of the complexity of some of the older iterations of this program. Structured handling was also an interesting subject. In the beginning weeks, before we started working in Spyder, it was very inefficient to have to restart the program after each error. This is a

# GitHub Link

<https://github.com/NiccoFortes/Assignment_07.git>

# Appendix/Search

None