Nathan Fenske

3/20/2022

Intro To Python

Assignment 8

**Introduction**

The goal of this assignment is to allow the user to create a CD inventory list by presenting a menu he/she may interact with by using object-oriented programming. The menu allows the user to add a CD with the artist, view the current inventory list, save it, and exit. Error handling is added to ensure the user has a working inventory list and prevents erroneous data or unclear errors and crashes.

**Summary**

The following program, part of it shown below in Figure 1, uses class definitions to keep the code organized and clean. Error handling is placed in defined functions where user input is required. This allows the user to know exactly what he or she did incorrectly as well as prevents the program from closing or crashing without any explanation. Static methods are employed to associate objects with a class.

Text

Description automatically generatedFigure 1: Part of the program

**Details**

**The CD Class**

The code below in Figure 2 stores data about the cd using constructors, properties, and methods within the class “class CD”. The constructor automatically invokes right after a new CD is created. This sets up the initial attribute values of the CD such as ID, title, and artist. The properties of each attribute is then defined which will interact with the attributes. The particular method of description formats the data in such a way that is clean with no trailing commas.

Text

Description automatically generated

Figure 2: Class CD

**The File IO Class**

The FileIO class definition contains static method procedures of what is to execute when called upon. In like 85, reading the file is defined. First, the inventory will clear and then the program will open the file. The order is determined by the values added to the variables cd\_id, title, and artist. The object of the CD is created and appended to the current inventory and the file closes. In the event there is nothing in the file, error handling displays this as seen in Figures 3 and 4. Starting in line 105, the method to write to the file is defined. The cd\_input calls for the rows to invoke the description definition in the CD Class to ensure the data is saved in the desired format. The program writes to the file, adds a new line and closes it. A message let’s you know once it’s been saved as seen in figure 5.

Text

Description automatically generated

Figure 3: Class File IO

Text

Description automatically generated

Figure 4: Loading From File

Text

Description automatically generated

Figure 5: Saving To File

**The IO Class**

Lines 114-182 lay out the blueprint for the menu in the class IO. This working code can be seen in Figures 6 and 7.

Text

Description automatically generated

Figure 6: Add a CD to Inventory

Graphical user interface, text

Description automatically generated

Figure 7: Display Inventory

**Running the Code**

Lines 189 to the end run through the user’s choices. For example, in line 220 (Figure 8), when the user chooses “i” to display the current inventory, the method show\_inventory is called upon to execute and displays the lstOfCDObjects, which would be the cd\_id, title, and artist.

The rest of the choices can be seen in Figures 9 and 10. With each option selected, the code calls the defined method to execute.

Graphical user interface, text

Description automatically generated

Figure 8: Show Inventory Method

Text

Description automatically generated

Figure 9: Command Prompt Demo

**GitHub Link**

The complete program can be found at the link below.

Graphical user interface, text

Description automatically generated

Figure 10: Notepad Screenshot