INST 326 Final Project - Spring 2021

Section 1. Overview

This project created a database of 20 different vinyl records from 1980-2020. From viewing the database known as vinylfile, a user may purchase a vinyl record in the store created in the program. The user can be given the price of the vinyl they purchased, and may continue to purchase vinyls from this store (only those corresponding to the vinyl ID's in the CSV file) until they decide to stop by inputting 'no'. The program then produces a final price for the vinyls purchased. 2 py-tests were created (one for total_purchase and another for get_price) that reaffirms the success of the functions, by allowing you to test receiving a price and getting the total of all vinyls from the CSV.

Section 2. Functionality

Class 1: This Class named 'VinylRecord' lists 20 of the records we created in a CSV file. Pandas is used to read and pass the CSV file into our project, in order to do so we import pandas as pd before the class was written.

Method 1: This init method creates an empty shopping cart by allowing users to pass through a vinyl file.

Method 2: This method includes all of the subject matter in our csv file. We called this 'metadata' And this is a string that has all of the characteristics we need to start the function. This method also prompts the user with a question, "pick a number to view your selected metadata." Each metadata is associated with a number, users can choose a number to get information about that vinyl. Metadata = vinylID

Method 3: This method 'purchase' places a function in the cart without the user, essentially allowing the user's shopping cart to open and finalize the purchase before getting a price and total.

Method 4: This method, titled 'make_purchase' prompts the user with a question through an int method, the question asks users to enter what vinyl ID they would like to buy. After the users answer the question with a number corresponding to metadata the choice will be added to their shopping cart using the 'append' operator.

Method 5: The get_price function allows a user to figure out the price of any metadata they chose from method 4, the function will pass in their choice through the dataframe, using 'iloc' to return the row back to 0. The price is read to the user after the user passes their choice through.

Method 6: This total function allows users to get a total price after they have the price and purchased a record. Users can continue to buy records from the shopping cart and the total is returned.

Method 7: This function allows users to continue to purchase records until they no wish to. After they receive a total from method 6, they are asked if they'd like to make a purchase. An if, then statement was added to continue the purchases until the user said 'no'. Once the user says no the function returns to Method 1 with the former information cleared.

To add complexity to this project I created a GUI. This GUI uses Tkinter, and when the user selects the numbers of vinyl id's (methods 6 and 7), they receive the price in the GUI that opens automatically following the command. The user has the option to find out the album names (Just a Vinyl_ID would be of no use to a customer) by clicking on the button in the GUI that says "See the records you chose!"

Section 3. Troubleshoot

How to run the function: enter the function in the command terminal, and answer the questions. Example: In command terminal enter python3 FinalProjectINST326.py Upon first question: Enter '1' Upon second question: Select any number (1-20 ONLY) Upon the third question: Enter 'yes' if you'd like to continue; enter 'no' to break and end the function. If 'no' was entered; you will receive a final total.

Section 4. Citations

We received:

- album_name, artist_name, genre, track_amount, runtime, first_available, discogs_rating, label_name, label_city, label_country from: https://www.discogs.com/
- weight, is_explict, amazon_rating, price in $, from: https://www.amazon.com/
- billboard, occ, aria, gfk, oricon, gold, platinum, multi_platinum, diamond, rollingstone_rating, from: https://en.wikipedia.org/wiki/Main_Page