

CS 115 Homework 10: Recommender+

Overview

In this assignment you will extend the music recommender system from chapter 5 of the textbook. The objectives of this assignment are

- Practice with imperative programming (while- and for-loops), mutable data types (lists, dictionaries).
- Practice designing a program that combines several features and needs to be implemented using several functions.

Suggestions

- Start early, reading the assignment and sketching some ideas. Then set aside for a few hours or days. Review your ideas and think about alternatives and improvements before proceeding with the details.
- Plan for unit testing: Early on, make notes on how to test the main functions. Define the test cases so the functions can be tested separately, before combining them.
- Review at least section 5.6 of the textbook concerning file I/O, and section 5.7 which provides code that you are free to modify and extend.

Requirements

Recommender+ provides a “menu” of the following options.

- When the program starts, it loads the database from the file named “musicrecplus.txt”, which stores the database using the format in the textbook Section 5.7.
- Enter preferences: the user enters their preferences (i.e., the artists they like), as in the original program in the textbook. If this user was already present in the database, the new preferences replace the old ones.
- Ask for recommendations; that should work as in the original program. But getting recommendations is a separate menu item from entering preferences.
- Most popular artists: Print the artist or artists liked by the most users.
- How popular is popular: Print how many users like the most popular artist. If there is a tie for most popular, your program can do whatever you like. (For example, it could just say “sorry”, or it could compute how many users like all, or some, of the most popular.)
- Most likes: print the name of the user with the most artists in their preferences, and print their preferences too. If the user has “opted out” of this feature, don’t print their name. A user opts out by including a dollar sign at the end of their name. For example, “JoeSmith\$” has opted out but “SeanJohnson” has not.
- Quit: When the user chooses to quit, the current database should be written to the file “musicrecplus.txt”, replacing its old contents (if any).

The main program should be structured as a while-loop that repeatedly offers the user the choice to do one of the options. When he or she quits, it should halt after saving the file.

The program doesn't provide a graphical menu. Instead, it prompts the user with the following "menu" text:

Enter a letter to choose an option:

- e - enter preferences
- r - get recommendations
- p - show most popular artists
- h - how popular is the most popular
- m - which user has the most likes
- q - save and quit

After performing the chosen option, the program should print the menu again and repeat except, of course, for quit. If the user enters any letter other than the ones above, the program should print an error message and print the menu again.

Extras

You're welcome to add additional features, but your program is required to include at least these features, associated with the letters listed above.