

Project 3: GUI for Book Recommendations

CS 1410

Background

The purpose of this project is to get experience with event-driven programming with Graphical User Interfaces (GUIs). You will use the **breezypythongui** module, which is based on Python's **tkinter** module, to design interactive windows for the Book Recommendation System from Project 2. You will also add functionality to the system.

Requirements

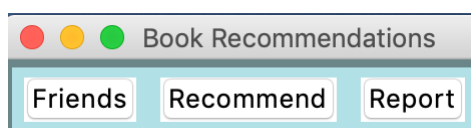
First, you need to make minor changes to your Project 1 code:

- Modify `friends` so that it takes the number of **friends** to consider. You need to change `friends` to determine an arbitrary number of top friends for a reader instead of just 2. Its signature should be `friends(name, nfriends=2)`. Since this function can be called with any number for **nfriends**, you will need to make sure that your program keeps all of the affinity scores between all users, so you can just return the correct number of friends on demand. As in Project 2, this must be done at the module level.
- Modify `recommend` so that it will collect book recommendations from any number of friends of the reader in question. Its signature is `recommend(name, nfriends=2)`.

Provide the following options to users, available through simple buttons in a small main window. The corresponding windows/dialogs appear and disappear independently from the main window.

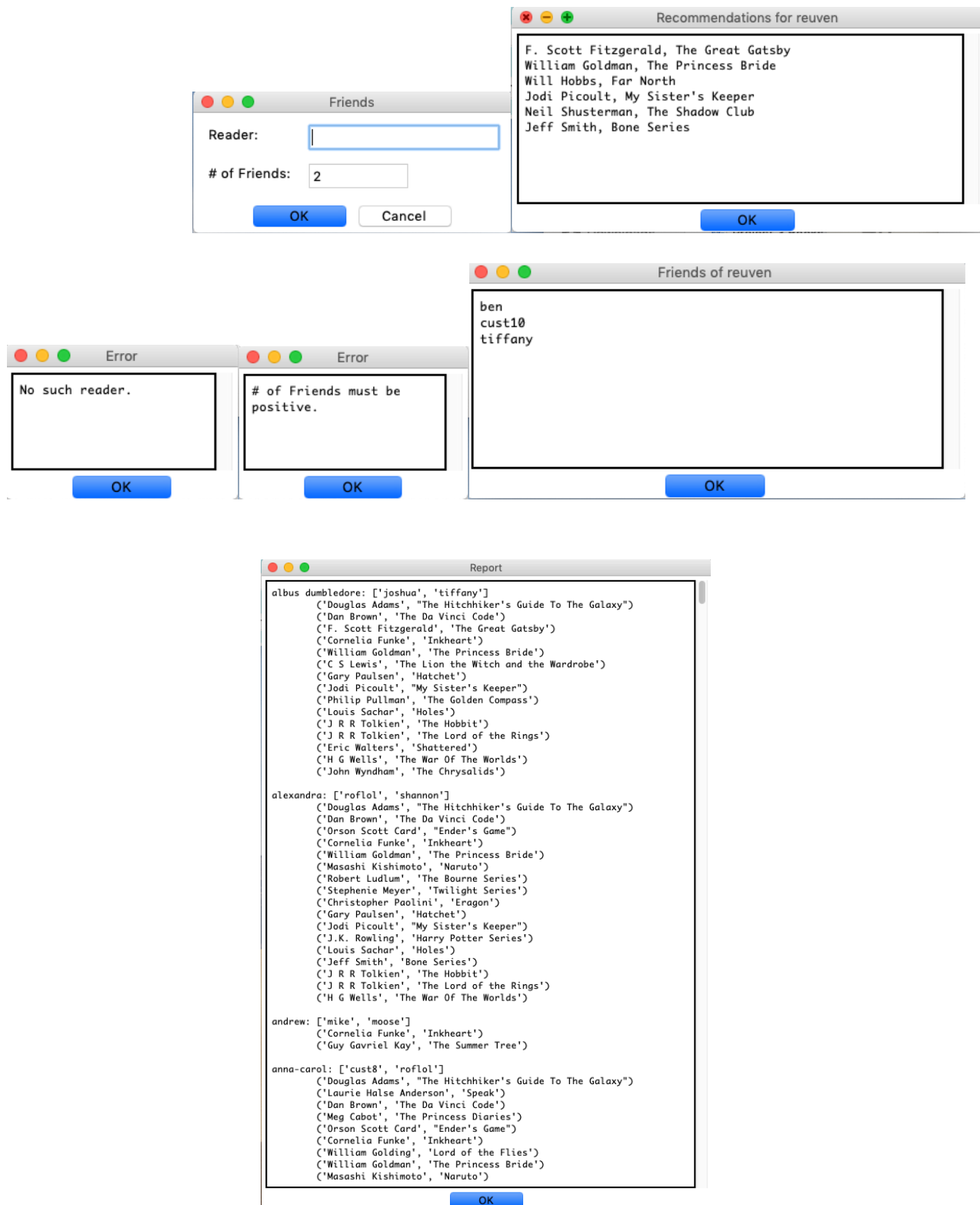
- **Friends.** This option displays a dialog box prompting for a reader name and a number of friends (`nfriends`, the readers with the highest affinity scores to the user of interest; display 2 as the default value in the number field), and displays the `nfriends` friends, one per line, in a message box. This calls your modified **friends** function.
- **Recommend.** This displays a dialog box prompting for a reader name and number of friends, and then displays recommendations for that user in a message box, using the number of top friends requested. Validate that the reader exists in the system. This calls your **recommend** function from Project 1.
- **Report.** This option displays the same output from `bookrecs.main()` as you did in Project 1, but in a message box. For this option, just use the default of `nfriends=2`. This calls your **report** function from Project 2.

Here is a sample display, using “powder blue” (“#B0E0E6”) as the background color:



You can find different color codes at https://www.rapidtables.com/web/color/RGB_Color.html.

Here are some sample screens:



Note that entries are **sorted** by the same criteria as in Project 2.

Implementation Notes

Use `breezypythongui.py` to implement your GUI.

You will need to ensure that the code in your *bookrecs.py* from Project 2 computes the affinity scores *upon import*, so that calls to **friends** and **recommend** will just work when called. This means that you must read the data from *booklist.txt* and *ratings.txt* and compute the affinities at the module/global level before **friends** or **recommend** are ever called. Your **main** from Program 2 will *not* be executed, since your GUI module *imports* your updated *bookrecs.py*.