Peer-graded Assignment: Step Five

Review criteria

You will submit your project by copying and pasting the link to the web page that runs your recommender. Reviewers will look at your project to confirm that all the required parts are complete.

Basic Requirements Checklist

- Does your program display a reasonable number of movies for the user to rate?
- Does your program successfully recommend at least one movie to the user?
- Does your program correctly display the results in an HTML table?

Instructions

For the last part of the capstone, you will integrate your recommender program with our course site. This will enable you to let a user run your program interactively on the internet. The user will be presented with a list of movies to rate, they will submit their ratings, and then a list of movies recommended for them by your program will be displayed.

You will be able to choose which movies the user is presented to rate, and how to display the recommended movies.

You will have to write a class that will allow the course site to run your recommender program, and then you will have to create a zip file of the code to be uploaded.

Write a Class

For this assignment you will be given one interface **Recommender**. You will have to write a class **RecommendationRunner** that implements **Recommender**. The two methods you will need to implement are:

- **getItemsToRate()**
- **printRecommendationsFor()**

When the user first visits the recommender site, our code will call the method **getItemsToRate()** to get a list of movies to display on the web page for users to rate.

After the user submits their ratings, our code will call the method **printRecommendationsFor()** to get your recommendations based on the user's ratings and display them.

Specifically, you should:

- Create a new class named **RecommendationRunner** that implements **Recommender**.
- Write the method **getItemsToRate()**. It returns a list of strings representing movie IDs that will be used to present movies to the user for them to rate.

  You can choose how to select movies for this list, for example, you could select recent movies, movies from a specific genre, randomly chosen movies, or something else.

  The movies returned by this method will be displayed on a web page, so the number of movies you choose to return may affect how long the page takes to load, and how willing users will be to rate the movies. 10-20 movies should be fine to get a good profile of the user's opinions, but 50 may be too many.
- Write the void method **printRecommendationsFor()**. It prints out an HTML table of movies recommended by your program for the user based on the movies they rated.
  It has one parameter **webRaterID**, a String that is the ID of the user, who has been added by our code to the **RaterDatabase** with the ratings they entered.
  To get the movies recommended by your program, you may want to use your **FourthRatings** class.
  Because the HTML printed by this method will be displayed on a webpage, the number of recommended movies you choose to display may affect how long the page takes to load.

  For example, you may want to display only the top 10-20 recommended movies, or to not include movies the user rated. In some rare cases there may not be any recommended movies, for example, if no movies were rated by the number of minimal raters specified in the recommender.

  If no movies are recommended, you should notify the user with a message. Whatever is printed by this method will be displayed on the web page: HTML, plain text, or debugging information.

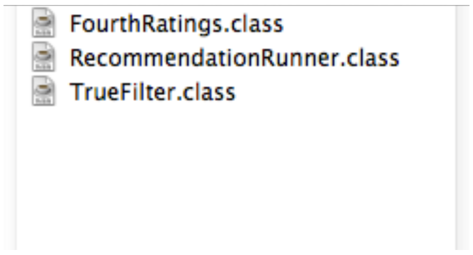
  If you want to style this HTML page, please include the CSS styling directly within the page using the **<style>** tag.

Create the zip file

To run your program on our course site, you will upload a single zip file containing some of your Java class files. You will need to include:

- **FourthRatings**
- **RecommendationRunner**
- Any other supporting classes you wrote and want to use (for example, if you use any of the filters you wrote, you will need to include them here).

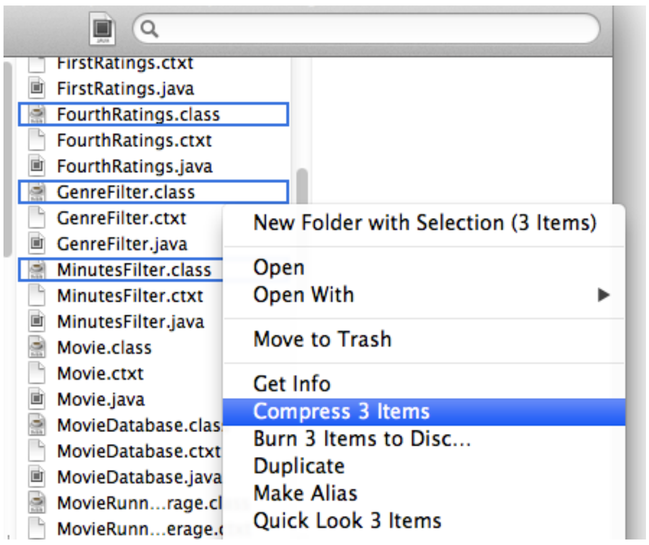
Note that you only need to include class files, not the original java files, so all files you include should have a .class extension. You do not need to include files we gave you such as **MovieDatabase**. Here is an example of the contents of the zip file:



```
FourthRatings.class
RecommendationRunner.class
TrueFilter.class
```

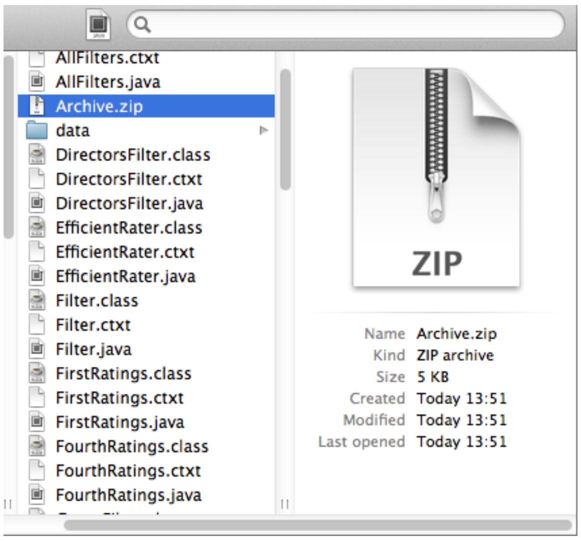
In this case, the learner has chosen to include their **TrueFilter**. You may wish to include other filters you wrote.

To create a zip file on a Mac, select the files you would like to include in the zip file while holding the command key. Once you have selected all the files you would like to include, release the command key, and right-click on one of the selected items. You should see something like this:



```
FirstRatings.ctxt
FirstRatings.java
FourthRatings.class
FourthRatings.ctxt
FourthRatings.java
GenreFilter.class
GenreFilter.ctxt
GenreFilter.java
MinutesFilter.class
MinutesFilter.ctxt
MinutesFilter.java
Movie.class
Movie.ctxt
Movie.java
MovieDatabase.clas
MovieDatabase.ctxt
MovieDatabase.java
MovieRunn...rage.cl
MovieRunn...erage.c
```

Select ‘Compress n Items’ (n will be the number of items you have selected). You should now see a file Archive.zip in the same folder:



```
AllFilters.ctxt
AllFilters.java
Archive.zip
data
DirectorsFilter.class
DirectorsFilter.ctxt
DirectorsFilter.java
EfficientRater.class
EfficientRater.ctxt
EfficientRater.java
Filter.class
Filter.ctxt
Filter.java
FirstRatings.class
FirstRatings.ctxt
FirstRatings.java
FourthRatings.class
FourthRatings.ctxt
FourthRatings.java
```

Name	Archive.zip
Kind	ZIP archive
Size	5 KB
Created	Today 13:51
Modified	Today 13:51
Last opened	Today 13:51

Run Your Code Online

To run your code online, upload your zip file at http://www.dukelearntoprogram.com/capstone/upload.php and press ‘Submit Code’. If there were no problems with your file, you should see a page with a link to click to run your program interactively.

Clicking this link calls the **getItemsToRate()** method, so you will see a page with movies to rate.
To share your program with others, copy the URL of this page with the list of movies to rate (not the upload page) and send it to whoever you would like to share your program with.