

Lab6B-MyFlix

Due No Due Date **Points** None

This lab has been designed to help you learn the use of ArrayLists, Comparable, hashCode(), and equals().

Problem statement: MoviesDB.tsv has tab-separate-values (TSVs). A tab is represented as "\t". This file has data about 20 movies. It has movie name, year, and genres. A movie may have one or more genres.

Fig.1 MoviesDB.tsv

Identity	2003	mystery	thriller	
Planet of the Apes	2001	action	adventure	thriller
...				
Lu ding ji II: Zhi shen long jiao	1992	action	comedy	
I Spy	2002	action	adventure	comedy
Dead Ringers	1988	drama	thriller	

Your program should perform two things:

- Provide a **search functionality** to allow users to search for a string in movie names. When found, it should print the movies, arranged in alphabetical order, as shown in Figure 2,3,4. If nothing is provided in the search string n input, i.e. simply pressed enter, then display the entire list of movies as shown in Figure 4. The search is case-insensitive.
- Provide a **list of genres with number of movies** in each. The list should be sorted in descending order of the number of movies in each genre as shown in Figure 5. Genres with same count should be sorted alphabetically.

<pre>*** Welcome to MyFlix *** 1. Search for a movie 2. List of genres 3. Exit 1 Enter the string to search in movie names future 1. Back to the Future Year: 1985 2. Back to the Future Part II Year: 1989</pre> <p>Fig.2 : Search Found</p>	<pre>*** Welcome to MyFlix *** 1. Search for a movie 2. List of genres 3. Exit 1 Enter the string to search in movie names abc Sorry! No movie found!</pre> <p>Fig.3: Search not found</p>
<pre>*** Welcome to MyFlix *** 1. Search for a movie 2. List of genres 3. Exit 1 Enter the string to search in movie names</pre>	<pre>*** Welcome to MyFlix *** 1. Search for a movie 2. List of genres 3. Exit 2 1. action Number of movies: 10 2. comedy Number of movies: 9</pre>

1. Antitrust	Year: 2001
2. Back to the Future	Year: 1985
3. Back to the Future Part II	Year: 1989
4. Bringing Down the House	Year: 2003
5. Cradle 2 the Grave	Year: 2003
...	
19. Resident Evil	Year: 2002
20. Shi mian mai fu	Year: 2004

Fig.4: Blank search string (simply pressed Enter)

3. drama	Number of movies:	8
4. adventure	Number of movies:	5
5. crime	Number of movies:	5
6. thriller	Number of movies:	4
7. horror	Number of movies:	3
8. mystery	Number of movies:	3
9. fantasy	Number of movies:	2
10. romance	Number of movies:	1

Fig.5: List of Genres (genres with same count sorted alphabetically)

Solution Design

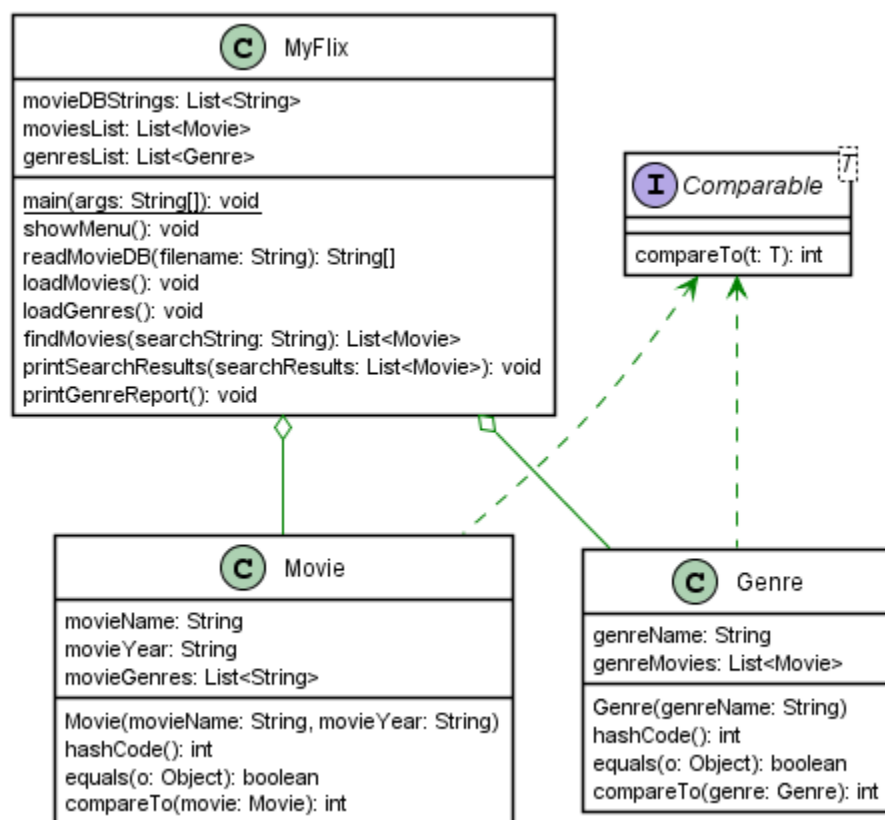


Fig. 6. Class diagram

As shown in Figure 6, the program has three classes: MyFlix, Movie, and Genre. MyFlix has 8 methods, of which three are fully coded. Refer to the comments in code file for description on other five methods.

Movie and Genre classes

- They both implement Comparable interface. The Movie comparable is based on movieName and the Genre comparable is based on the number of movies in its genreMovies list.
- They both override hashCode() and equals() methods. For Movie, the hashCode() and equals() use the movieName and movieYear as the unique identifier for a movie. For Genre, the genreName is used as the identifier. Depending on your logic, you may or may not use equals() and hashCode() in your code but equals() is tested in test-cases.

Instructions

- Create a package called lab6
- Download following files and copy/import them in lab6
 - [MyFlix.java](#) ⬇ (https://canvas.cmu.edu/courses/25253/files/7173444/download?download_frd=1)
 - [TestMyFlix.java](#) ⬇ (https://canvas.cmu.edu/courses/25253/files/7172945/download?download_frd=1)
- Download MoviesDB.tsv file and import it into your project folder
 - [MoviesDB.tsv](#) ⬇ (https://canvas.cmu.edu/courses/25253/files/7172944/download?download_frd=1)
- Complete MyFlix.java, and create Movie and Genre classes as required.
- Write your name in all three java files, zip them into Andrew-id-lab6.zip. Submit the zip file

Rubric

- Console outputs: 5 points
- Test-cases: 5 points
- Submission issues may cause a loss of up to two points