

Java Lab 16: Collections

This lab uses Collections. Create a project named Lab16.

Problem statement: The application will let users display a list of movies in several orderings and search for specific movies by several criteria.

The tab-delimited file "movies.tsv" has information about movies, one per line. The data's format is

id#<tab>movie name<tab>year<tab>countries<tab>genre1<tab>genre2 ...

where the genre list is some number of tab-delimited strings. You'll probably want to strip the strings of spaces; some of the movie names and countries are surrounded by quotes, which should be replaced with ""; there may be multiple countries listed for a movie, but treat it as one string – we won't be looking for specific countries; and finally, for some movies, the table display's formats may not be large enough, but you can ignore that – just use the given formats.

Your program will present these choices in a menu: sort by ID, sort by name, sort by year, sort by year (reverse), search by name, search by year, search by genre, quit. The sort options will use the method `sortBy(String)`, where the parameter is either "ID", "Name", "Year", or "Reverse Year", where the last one sorts from largest (most recent year) to smallest. The search options will use methods `searchByName(String)`, `searchByYear(int)`, and `searchByGenre(String)`. These methods return at most 5 movies. The first returns a Movie objects that match exactly on name, the second matches on year, and the third matches on genre. Each returns an `ArrayList<Movie>`; use `displayMovies()` to display the results. **For consistency, call `sortBy("ID")` before searching – otherwise, your results may differ.**

To do the sorting, you need three things. First, `Movie` implements the interface `Comparable<Movie>`; you'll need to implement its `compareTo()` method to compare movies on `movieID`. Second, use an anonymous inner class (recall that this is similar to an `EventHandler` in JavaFX) that implements `Comparator<Movie>`; you'll need to implement its `compare()` method to compare movies on `movieName`. Third, create a class named `SortByYear` that also implements `Comparator<Movie>`; you'll again need to implement its `compare()` method, but this time, compare movies on the year field. You should try out (not required) Eclipse's wizard: select New->Class, name it, then in the Interface box, click Add, then find `Comparator<T>` and click OK, then change T to `Movie` and click Finish. Here's the code for sorting by year in reverse order (from the most recent year), assuming you've written the `SortByYear` code:

```
Comparator c = Collections.reverseOrder(new SortByYear());
Collections.sort(movieList, c);
```

Screen shots (not all output shown for long results):

1. Display by ID 2. Display by name 3. Display by year 4. Display by year in reverse 5. Search by name 6. Search by year 7. Search by Genre 8. Quit Enter your choice: 1				1. Display by ID 2. Display by name 3. Display by year 4. Display by year in reverse 5. Search by name 6. Search by year 7. Search by Genre 8. Quit Enter your choice: 2			
ID	Name	Year	Countries	ID	Name	Year	Country Genres
2	Identity	2003	USA	40	21 Grams	2003	USA crime drama thriller
4	Planet of the Apes	2001	USA	223	50 First Dates	2004	USA comedy romance
6	Antitrust	2001	USA	133	A Beautiful Mind	2001	USA biography drama
7	Dirty Pretty Things	2002	UK	63	A Boy and His Dog	1975	USA comedy drama thriller
8	Bringing Down the House	2003	USA	150	A Bug's Life	1998	USA adventure animation comedy
9	Cradle 2 the Grave	2003	USA	128	AVP: Alien vs. Predator	2004	USA,UK,Czech Republic,Canada,Germany action adventure thriller
10	Death to Smoochy	2002	USA,UK,Germany	125	Adaptation.	2002	USA comedy drama
11	Helboy	2004	USA	185	After the Sunset	2004	USA action comedy crime
13	Jui kuen II	1994	Hong Kong	126	Agents secrets	2004	France,Italy,Spain action crime thriller
14	Shi mian mai fu	2004	China,Hong Kong	197	Akira	1988	Japan action animation
15	Femme Fatale	2002	France,Switzerland	127	Alfie	2004	UK,USA comedy drama romance
16	Lu dino ii	1992	Hono Kono	130	Alien: Resurrection	1997	USA action horror
				129	Aliens	1986	USA,UK action horror
				93	Amarcord	1973	Italy,France comedy drama
				66	American Beauty	1999	USA drama romance
				78	Angel Heart	1987	UK,Canada,USA horror mystery thriller
				90	Anger Management	2003	USA comedy

```
1. Display by ID
2. Display by name
3. Display by year
4. Display by year in reverse
5. Search by name
6. Search by year
7. Search by Genre
8. Quit
Enter your choice: 3

ID      Name      Year      Country Genres
88      La grande illusion 1937      France drama war
98      Persona 1966      Sweden drama thriller
93      Amarcord 1973      Italy,France comedy drama
145     Enter the Dragon 1973      Hong Kong,USA action crime drama
63      A Boy and His Dog 1975      USA comedy drama thriller
282     Eraserhead 1977      USA fantasy horror
88      Hi no tori 1978      Japan action adventure animation
92      Revenge of the Pink Panther 1978      UK,USA comedy crime mystery
131     Blade Runner 1982      USA,Hong Kong,UK thriller
62      Brainstorm 1983      USA thriller
184     Kaze no tani no Naushika 1984      Japan adventure animation fantasy
132     Nineteen Eighty-Four 1984      UK drama romance thriller
56      This Is Spinal Tap 1984      USA comedy music
18      Back to the Future 1985      USA adventure comedy
129     Aliens 1986      USA,UK action horror
78      Angel Heart 1987      UK,Canada,USA horror mystery thriller
226     Predator 1987      USA action horror
171     The Untouchables 1987      USA crime drama thriller
197     Akira 1988      Japan action animation
27      Dead Ringers 1988      Canada,USA drama thriller

1. Display by ID
2. Display by name
3. Display by year
4. Display by year in reverse
5. Search by name
6. Search by year
7. Search by Genre
8. Quit
Enter your choice: 4
Enter your choice: 4

ID      Name      Year      Country Genres
219     Assault on Precinct 13 2005      USA,France action crime drama
218     Constantine 2005      USA,Germany drama fantasy horror
223     50 First Dates 2004      USA comedy romance
128     AVP: Alien vs. Predator 2004      USA,UK,Czech Republic,Canada,Germany action adventure thriller
185     After the Sunset 2004      USA action comedy crime
126     Agents secrets 2004      France,Italy,Spain action crime thriller
127     Alfie 2004      UK,USA comedy drama romance
137     Blade: Trinity 2004      USA action adventure fantasy
228     Collateral 2004      USA crime drama thriller
138     Cube Zero 2004      Canada,USA mystery thriller
170     Diarios de motocicleta 2004      Argentina,USA,Chile,Peru,Brazil,UK,Germany,France adventure biography drama
144     Dodgeball: A True Underdog Story 2004      USA,Germany comedy sport
146     Eternal Sunshine of the Spotless Mind 2004      USA drama romance
87      Exorcist: The Beginning 2004      USA horror mystery thriller
142     Fahrenheit 9/11 2004      USA comedy documentary drama
113     Finding Neverland 2004      USA,UK biography drama family
186     Garden State 2004      USA comedy drama romance
```

```
1. Display by ID
2. Display by name
3. Display by year
4. Display by year in reverse
5. Search by name
6. Search by year
7. Search by Genre
8. Quit
Enter your choice: 5
Enter the movie name: Blade Runner

ID      Name      Year      Country Genres
131     Blade Runner 1982      USA,Hong Kong,UK thriller

1. Display by ID
2. Display by name
3. Display by year
4. Display by year in reverse
5. Search by name
6. Search by year
7. Search by Genre
8. Quit
Enter your choice: 6
Enter the year: 1999

ID      Name      Year      Country Genres
19      Hei kek ji wong 1999      Hong Kong comedy drama romance
39      The Boondock Saints 1999      Canada,USA action crime thriller
66      American Beauty 1999      USA drama romance
162     Office Space 1999      USA comedy
```

```
1. Display by ID
2. Display by name
3. Display by year
4. Display by year in reverse
5. Search by name
6. Search by year
7. Search by Genre
8. Quit
Enter your choice: 7
Enter the genre: action

ID      Name      Year      Country Genres
4      Planet of the Apes 2001      USA action adventure thriller
6      Antitrust 2001      USA action crime drama
9      Cradle 2 the Grave 2003      USA action crime drama
11     Hellboy 2004      USA action fantasy horror
13     Jui kuen II 1994      Hong Kong action comedy

1. Display by ID
2. Display by name
3. Display by year
4. Display by year in reverse
5. Search by name
6. Search by year
7. Search by Genre
8. Quit
Enter your choice: 7
Enter the genre: kiddie
Nothing to display
```

Solution Design:

Steps	
<ul style="list-style-type: none">Download Lab16Main.java, Movie.java, movies.tsv	<u>Lab16 instructions</u> <ul style="list-style-type: none">Complete all classes listed below. Check output with movies.tsv.Write your name and Andrew Id at the top in all Java files to be submittedZip all the java files in a zip-file named as your Andrew ID. Submit it on Canvas.

Class	Member variables	Methods
Lab16Main	<ul style="list-style-type: none">movieList: ArrayList<Movie>	<p>main(): Creates a Lab16Main object, calls its readMovies() method with "movies.tsv"; calls getMenuChoice() in a loop; calls Lab4Main methods to do the other work. <i>Partially coded.</i></p> <p>public static final int</p> <p>static int getMenuChoice(): display the menu, get a valid user choice. <i>Already coded.</i></p>

		<p>void readMovies(String filename): open the movie file, read each line and split it on \t, call toMovie() to parse the Strings, store the new movie in movieList. <i>Partially coded.</i></p> <p>Movie toMovie(String[]): get the movieID (convert to int), movieName (strip it and replace "\"" with ""), year (convert to int), country (strip it and replace "\"" with ""), and the genres (split again on ",", add the genre strings to an ArrayList<String>). New up a new Movie object from these parts and return it. <i>Already coded.</i></p> <p>private void displayMovies(ArrayList<Movie>): display the parameter as a table. This is parameterized so it can be used with any ArrayList<Movie> argument. <i>Already coded.</i></p> <p>void sortBy(String): use Collections.sort() to sort the movie on the appropriate field, as indicated by the parameter (Name, ID, Year, or ReverseYear). See the code above for ReverseYear. <i>Partially coded.</i></p> <p>ArrayList<Movie> searchByName(String name): returns up to 5 moves whose name match the parameter, or null if not found. <i>Already coded.</i></p> <p>ArrayList<Movie> searchByYear(int year): returns up to 5 movies from that year. Use displayMovies() to display what it returns.</p> <p>ArrayList<Movie> searchByGenre(String genre): returns up to 5 movies matching that genre. Use displayMovies() to display what it returns.</p> <p>ArrayList<Movie> getList(): returns movieList, breaks encapsulation, but okay for now. <i>Partially coded.</i></p>
Movie implements Comparable<Movie>	<ul style="list-style-type: none"> • int movieID: • String movieName • int year • String country • ArrayList<String> genres 	<p>Overloaded constructor, toString(), all getters. <i>Already coded.</i></p> <p>int compareTo(Movie): compares this movie to the parameter on the movieID field</p>
SortByYear implements Comparator<Movie>		<p>public static int compare(Movie, Movie): compares two movies on year</p>