



ITCS 209 Object Oriented Programming	Name:	Lab	Challenge Bonus	
	ID:			
	Section:			

Lab11: RegEx, File I/O, Exception (again)

Continue from Lab10, you have done importing movies from the files. In this lab, you will be importing movies' rating and exporting movies to the file. You can add any variables and methods on your own, however, do not modify the code in the "DO NOT MODIFY" section.

Task 1: import rating and update movies' rating

- 1.1 Modify **Movie** class to accept the rating and calculate average rating score. The movie with no rating is not the same as movie with 0 point rating. As a result, if there is no rating yet, the **getAverageRating()** will throws **MovieException** with the message "no_rating" without returning any specific rating score. (note. You may see how this Exception work in the toString() method)
- 1.2 Modify **BasicMovieDatabase** class by adding some method from your lab 10 and implementing a new method called **public void importRating(String filename)**: to read data from file that contains list of movie rating. A rating is a tuple of (mid, score, timestamp). The mid is the IDs of the rated movie, score is a double numeric value. If given, a rating score can range from [0.0, 5] inclusive. Finally, timestamp¹ is a long value indicating the time at which this rating was given.

Task 2: exporting movies and write to a new file

- 2.1 Implementing **exportMovies(String filename)** method in the **BasicMovieDatabase** class to print all movies in this class in the valid format <mid>,<title>,<tag1>|<tag2>|...|<tagN> and then return number of exported movies.

Expected output for Task 1 & Task 2:

```
Done importing 48 movies
-- before adding ratings --
Retrieving movie ID 139747: [mid:139747->Before We Go [Romance] (rating:no_rating)]
Retrieving movie ID 141432: [mid:141432->Sweet Red Bean Paste [Drama] (rating:no_rating)]
Retrieving movie ID 139415: [mid:139415->Irrational Man [Drama] (rating:no_rating)]
Retrieving movie ID 139620: [mid:139620->Everything's Gonna Be Great [Adventure, Drama, Comedy,
Children] (rating:no_rating)]
Retrieving movie ID 141305: [mid:141305->Round Trip to Heaven [(no genres listed)] (rating:no_rating)]
Retrieving movie ID 140715: [mid:140715->Straight Outta Compton [Drama] (rating:no_rating)]
Retrieving movie ID 148626: [mid:148626->The Big Short [Drama] (rating:no_rating)]
-- after adding ratings --
Retrieving movie ID 139747: [mid:139747->Before We Go [Romance] (rating:2.00/5)]
Retrieving movie ID 141432: [mid:141432->Sweet Red Bean Paste [Drama] (rating:2.75/5)]
Retrieving movie ID 139415: [mid:139415->Irrational Man [Drama] (rating:no_rating)]
Retrieving movie ID 139620: [mid:139620->Everything's Gonna Be Great [Adventure, Drama, Comedy,
Children] (rating:3.00/5)]
Retrieving movie ID 141305: [mid:141305->Round Trip to Heaven [(no genres listed)] (rating:4.25/5)]
Retrieving movie ID 140715: [mid:140715->Straight Outta Compton [Drama] (rating:3.38/5)]
Retrieving movie ID 148626: [mid:148626->The Big Short [Drama] (rating:2.80/5)]
-- exporting movies --
Done exporting 50 movies
```

¹ <https://en.wikipedia.org/wiki/Timestamp>

Challenge Bonus (Optional):

Similar to lab 10, you can search the movie and return only the movie that match with the given condition. In addition, to the keyword search on the movie's title, you will have to match the rating score condition too. There are five types of operators: OPERATOR {EQ, LT, GT, LTE, GTE}; that can be operated on the rating score. (EQ = Equal, LT = Less than, GT = Greater than, LTE = Less than or equal, and GTE = Greater than or equal). Implement the following method

public List<Movie> searchMoviesWithRating(String query, OPERATOR op, **double** rating): Return a list of movies, each of whose title contains "query" and rating scores satisfies the condition.

```
//////////////////// BONUS //////////////////////
Results for movies that match: [america] and have rating EQ 0.0

Results for movies that match: [america] and have rating EQ 2.5
    [mid:86290->American: The Bill Hicks Story [Documentary, Comedy]
    (rating:2.50/5)]

Results for movies that match: [america] and have rating EQ 4.0

Results for movies that match: [america] and have rating GT 0.0
    [mid:86290->American: The Bill Hicks Story [Documentary, Comedy]
    (rating:2.50/5)]
    [mid:140711->American Ultra [Action, Comedy] (rating:3.33/5)]
    [mid:39398->C.S.A.: The Confederate States of America [Drama, Comedy]
    (rating:4.25/5)]
    [mid:114662->American Sniper [Action, War] (rating:3.20/5)]

Results for movies that match: [america] and have rating GT 2.5
    [mid:140711->American Ultra [Action, Comedy] (rating:3.33/5)]
    [mid:39398->C.S.A.: The Confederate States of America [Drama, Comedy]
    (rating:4.25/5)]
    [mid:114662->American Sniper [Action, War] (rating:3.20/5)]

Results for movies that match: [america] and have rating GT 4.0
    [mid:39398->C.S.A.: The Confederate States of America [Drama, Comedy]
    (rating:4.25/5)]

Results for movies that match: [america] and have rating GTE 0.0
    [mid:86290->American: The Bill Hicks Story [Documentary, Comedy]
    (rating:2.50/5)]
    [mid:140711->American Ultra [Action, Comedy] (rating:3.33/5)]
    [mid:39398->C.S.A.: The Confederate States of America [Drama, Comedy]
    (rating:4.25/5)]
    [mid:114662->American Sniper [Action, War] (rating:3.20/5)]

Results for movies that match: [america] and have rating GTE 2.5
    [mid:86290->American: The Bill Hicks Story [Documentary, Comedy]
    (rating:2.50/5)]
    [mid:140711->American Ultra [Action, Comedy] (rating:3.33/5)]
    [mid:39398->C.S.A.: The Confederate States of America [Drama, Comedy]
    (rating:4.25/5)]
    [mid:114662->American Sniper [Action, War] (rating:3.20/5)]

Results for movies that match: [america] and have rating GTE 4.0
    [mid:39398->C.S.A.: The Confederate States of America [Drama, Comedy]
    (rating:4.25/5)]
```