CS1083 Assignment #5 – Fall 2021

Due: Wednesday, October 20th before 11:00pm (Atlantic time) in the Assignment 5 Desire2Learn submission folder. (See submission instructions below).

The purpose of this assignment is to review and practice file I/O and exception handling.
Preparation:

- Review your notes on Basic File I/O and Exceptions in Chapter 11 of your textbook.
- Review the provided Javadoc files in detail.

Your job:

- Complete the MovieReporter class.
- Create all custom Exception classes.

This assignment is to be done individually. If you have questions, direct them to a tutor/assistant during a help session in the "Faculty of Computer Science Student Success Centre" team or to your course instructor.

Complete the MovieReporter class.

You will complete a class that will answer the age-old question… **what is the best type (or genre) of movie**?

You have been given a data file containing a summary of movie ratings from users of the MovieLens.org website.

You have been given all the classes you need to organize and calculate the average rating by genre. You just need to create the code to read the lines from the file, make use of the classes, and display the genres and their average rating (sorted from most highly rated to lowest rated).

Each line in the file (genrerating.txt) contains a line representing a movie and its ratings. Specifically, a line contains: a movie genre, the total number of points received from all ratings, the total number of ratings received, and the title of a movie. Take note, some movies fall into multiple genres and so will have multiple lines in the file. Don’t read the file in from standard input, instead read the file by opening it in your java program. **Hint**: Also, remember to use the appropriate delimiter for your input file.

Important: File processing to do calculations is an extremely common programming activity; however, data sets such as this one commonly contain some errors. You will need to make sure that your program completes without error (no exceptions should be thrown from the main method). You can ignore any lines containing issues; your program should skip them and move on.

You will also need to create an output file (called errorLog.txt) that describes exactly what lines in the input file were not able to be read (if you encounter a line that does not perfectly follow the format) write the details to an error file. Also, include the error message when there are duplicate movie entries for a particular genre.

Note: do not use redirection on the command line to read or write to the files, instead **for both the input and output files you must open them in your java program**.

Finally, there are two exceptions thrown by the provided classes. You must create the java files for these two exceptions.

Hint: when handling exceptions remember that we can nest try/catch statements (ie: a try block can contain a try/catch block). Also, try adding the genre to the set and if it already exists you should get the existing genre, and then try adding the movie to the genre object. For example:

```
      try{
            //try to add the genre to the set...
            //catch the exception if it can't be added
            genres.addGenre(g);
      }
      catch(DuplicateGenreException e){
            //get the existing genre object
      }

      try{
            //try to add a new movie
      }
      catch(DuplicateMovieException e){
            //print to error log
      }
```

Before you get started!

Look at the source files and Javadoc before you get started. Note that most of the classes have been provided. You will need to create the Exception classes that are required and complete the MovieReporter class. GenreSets contain a list of Genres and Genres contain Movies. Use these classes to organize movies and do the required work. Do not forget to add an author tag containing your name and student ID to the MovieReporter class and include proper Javadoc description and author tag in the Exception classes you create.

Screen Output:

Your output in the terminal should look something like this:

```
TOP MOVIE GENRES
=======================
1. Mystery, contains: 40 movies, avg. rating: 3.34
        Rear Window (1954); average rating: 4.39
2. Animation, contains: 29 movies, avg. rating: 3.32
        Close Shave, A (1995); average rating: 4.49
3. Romance, contains: 170 movies, avg. rating: 3.30
        Casablanca (1942); average rating: 4.46
4. Drama, contains: 471 movies, avg. rating: 3.27
        Entertaining Angels: The Dorothy Day Story (1996); average
rating: 5.00
5. Sci-Fi, contains: 76 movies, avg. rating: 3.24
        Star Kid (1997); average rating: 5.00
6. Thriller, contains: 176 movies, avg. rating: 3.14
        Close Shave, A (1995); average rating: 4.49
7. Comedy, contains: 355 movies, avg. rating: 3.09
        Close Shave, A (1995); average rating: 4.49
8. Action, contains: 188 movies, avg. rating: 3.01
        Star Wars (1977); average rating: 4.36
9. Fantasy, contains: 16 movies, avg. rating: 2.98
        Star Kid (1997); average rating: 5.00
10. Horror, contains: 63 movies, avg. rating: 2.65
        Psycho (1960); average rating: 4.10
```

ErrorLog contents

The contents of your errorLog.txt file should look something like this:

```
Face/Off (1997) is already in the genre Action
Hugo Pool (1997) is already in the genre Romance
Hurricane Streets (1998) is already in the genre Drama
problem reading line: 855
problem reading line: 862
Nightwatch (1997) is already in the genre Horror
Nightwatch (1997) is already in the genre Thriller
That Darn Cat! (1997) is already in the genre Comedy
That Darn Cat! (1997) is already in the genre Mystery
Money Talks (1997) is already in the genre Action
Money Talks (1997) is already in the genre Comedy
```

Submission Instructions on next page...

Your electronic submission (submitted via Desire2Learn) will consist of two files:

i. a written report. This should begin with a title page. As always, your title page should include: the course (CS 1083), the assignment number (Assignment #5 in this case), your full name, and your UNB student number. That should be followed by each part clearly identified with a section heading. Include:

   a. the completed source code for the MovieReporter class
   b. the custom Exception classes you wrote
   c. the output from the completed MovieReporter program (both the screenshot of the terminal window and the error log output).

   This written report should be prepared using a word processor; we recommend using Microsoft Word (i.e. create a .docx file for your report). Copy & paste your java source code & required output into the report document. Add appropriate headings for each part. Fix up the formatting where necessary, adjusting line breaks & page breaks to ensure that your document is easy to read. Use a monospaced font for your code to maintain proper indentation.) Once the report is complete and you've checked it all over, save the .docx file for your own records, and then **save a second copy in pdf format for submission**. (Note: Be sure to open that file in a pdf viewer to verify that the pdf was generated correctly.) The **SINGLE pdf file** containing your report will be submitted to the appropriate assignment submission folder on Desire2Learn. (It is important that you submit a pdf file and NOT the original Word document. This pdf will allow the marker to write comments directly on your work to give you better feedback.)

   Note: Please name this report as follows: **YourName_As5_Report.pdf**

ii. an archive file (.zip) that contains all your work for this assignment. Make sure that your archive includes **all source code** (.java files - in case the marker wishes to compile & run your code), the input file and output files (the error log file and the screenshot of your output in the terminal window). This archive should be submitted as a single file to the appropriate submission folder on Desire2Learn.

   Note: Please name this archive file as follows:
   YourName_As5Archive.zip