

ECE 325 LAB Assignment 5: Sorting your song collection

(20 pts total) After the **release** of your first album, and **playing** your first show, you finally have the time to **improve** the **management system** for your collection of songs. In this assignment, you'll **implement** a **new version** of the **SongCollection** class which **allows** **sorting songs** in different ways.

In this version of the application, a Song has **a title, release date** and a **popularity score**. The **popularity score** **corresponds** to the **number of plays** of the song on a Spotify-like application.

The information about the songs is currently **stored in songs.txt** (one line for each song). You must **load** the **songs into** a **SongCollection**. Because you **assume that** there were **some mistakes** when **entering** the **information** in the **songs.txt file**, you decide to **use a TreeSet** to **store the SongCollection** to **avoid** **storing** duplicate songs. Two songs are **considered duplicates** if they have the **same title** and **release date**. The TreeSet **should order** the songs alphabetically by title and then from new to old.

- 1) (4pts) Let Song implement the correct interface for usage in a TreeSet and finish the implementation for the sorting.
 - 2) (8pts) Implement the loadSongs() method in the SongCollection class. You can load the songs directly into the collection. Your implementation must use a BufferedReader and a Scanner.
 - 3) (4 pts) Sometimes, we want to see the songs ordered by their popularity (and after they are sorted by popularity, in their natural ordering). Finish the implementation of the sort() method.
 - 4) Print the SongCollection (just use its toString() method) directly after loading the songs into the collection.
 - 5) Print the SongCollection after sorting it by popularity.
- (4 pts) Overall code quality.

Some hints for your implementation:

- You can parse a string into a LocalDate object using the provided SongCollection.parseLocalDate(String str) method.
- Make sure to deal with exceptions were necessary and close your stream correctly.
- Make sure not to load garbage from the file... some lines may be corrupted.

Please submit:

1) A zip file containing your code and a PDF with the answers to the questions above.

Name the file 'FirstName_ID_lab_asg5.zip' and keep the exact same file structure as the zip that was provided for the assignment. For example,

Filename: Cor-Paul_1234567_lab_asg5.zip

```
|----- solution.pdf (if applicable for this assignment)
|----- src
|         |----- ece325_lab_assignment5
|         |         |----- *.java
```

2) A screencast/movie that shows the following steps:

- Open your eClass with your name shown
- Open your IDE
- Show your code briefly
- Execute your code and show the results of the execution of 4) and 5).

Please do not modify any of the names/methods we've defined in the provided *.java files, unless explicitly mentioned that you are allowed to do so.