

Programming Assignment 5

Note: When you turn in an assignment to be graded in this class you are making the claim that you neither gave nor received assistance on the work you turned in (except, of course, assistance from the instructor).

Programming Project 5 builds on the work you completed in Programming Project 4. Begin by writing a subclass of your **Song** class and call it **MySong**. The **MySong** class includes an additional data field, **playcount**.

If the playcount is not present in the input data file, then assign a value of 0 to the instance variable.

Refactor the **SongReader** class to read **MySong** data from the input file.

Write the class **MusicManager** that contains the **main** method and instantiates a **SongReader** object and call the methods of the **SongReader** class to read in the input data file

The **MusicManager** class will provide a variety of functions based on input to the program from command line arguments. The arguments have a very specific ordering. The first argument is the name of the file that contains all of the song data. The format of this file is the same as described for Project 4 with an additional tag for the playcount, <playcount>. When your **MusicManager** program reads this input file, it will be creating **MySong** objects.

The second command line argument is the number that refers to the desired functionality for the **MusicManager** program, 1, 2, or 3. Here is the specified behavior for each of the three options:

1. Display the top 10 songs with the highest playcount ordered from highest to lowest playcount. If the playlist contains less than 10 songs, display all the songs.
2. Reads the third command line argument as an artist name and displays a message if the playlist contains any songs by that artist. Note: a multi-word artist name can be passed as a single command line argument if it is enclosed in double quotes. For example,

```
java MusicManager myPlaylist.data 2 "Iron & Wine"
```

the command line argument array is interpreted as

```
String[] args = {"myPlaylist.txt", "2", "Iron & Wine"};
```

3. Reads the third command line argument as an artist name and displays all songs by that artist. The songs are grouped by album and for each album the songs are listed in alphabetical order by title.

It is expected that your program will be well documented, and you are required to write a private helper method called **printHeading** in the **MusicManager** class that outputs the following information to the console in an easy-to-read format: your name, the project number, the course identifier (CMSC 256), and the current semester. You will call this method as the first statement in your main method. All files must contain a comment block at the beginning that includes the file name; all of the same information that was specified for the helper method **printHeading**; and a brief description of the file's purpose.

You will submit the project files (**Song.java**, **SongReader.java**, **MySong.java**, and **MusicManager.java** – along with any other Java files needed to run your program) to the assignment link in Blackboard.