

The Music Player (part 2)

Purpose: To understand how to use other collection types besides arrays.

Learning Objectives:

- Using ArrayLists and HashMaps
- Using Iterators
- Dynamic creation of collections
- Modifying collections in a loop
- Use of the equals() method

Conversions:

- Convert the array of artists in the Song class to an ArrayList of artists.
- Create an equals() method in the Song class which declares a Song to be equal to another Song if the title and release year are the same.
- Convert the array of songs in the Playlist class to a HashMap, with a unique Integer key. (Hint: Utilize a loop counter)
- Create a deleteSong() method in the Playlist class that accepts a Song as a parameter, and will delete any song equal to the passed parameter
- Last time, we created the displayPlaylist() method as a static method of the Driver class. Does this make sense? (No.) So let's improve this design and create a display() method in the Playlist class instead.
- *NOTE: This project should preserve all functionality of The Music Player (part 1). Given the conversions from Arrays to ArrayLists and HashMaps, this will require refactoring. For instance, how will the addArtist method of Song need to change? It is now dealing with an ArrayList, not an Array. Also it needs to return the index position of the added Artist. A new technique is needed. Use the Java API!*

Driver:

- Loop through the **complete** Playlist. For each song, reverse the order of performers. Thus if P!nk was the first artist one on "Just Give Me A Reason" and Nate Ruess was the second, invert the order such that Nate Ruess will become the first artist.
- Loop through the **complete** Playlist several times to build new Playlists. They can maintain the same HashMap key from the **complete** Playlist. Make the following playlists:
 - "Dance Songs" : For any song that is either R&B or Disco.
 - "Mya's Greatest Hits" : For any song that has Mya as a performer
- Iterate through the **complete** Playlist and
 - Remove the song "Roar"
 - Add a new Rock song of your choosing.
- Iterate through the **complete** Playlist again, and add a new acoustic version of every Rock song. The song title should be appended with " [acoustic version]"
- In the driver, create a new Song instance called **songToBeDeleted**. This song should be

Song Title	Artist First Name	Artist Last Name	Genre	Release Year
Crush On You		The Jets	R&B	1985

Note that this song has a different Genre from the original "Crush On You" song by the Jets in your playlist. However, the equals() method should declare that original song equal to **songToBeDeleted**.

Now use **songToBeDeleted** to remove "Crush On You" by the Jets from the complete playlist by invoking your new deleteSong() method from the Playlist class.

- Finally, use the pre-existing display() method in the Driver class to display the results from all of these playlists:
 - "Complete"
 - "Mya's Greatest Hits"
 - "Dance Songs"