Implement a `MusicLibrary` class in Java that will manage a collection of music tracks. The class should only use `ArrayList<T>` collections to store both tracks and playlists. Implement basic operations on collections, including adding and removing elements, searching, and displaying content. In the `main` method, show example usage of the class.

- `ArrayList<String>` for storing music tracks
- `ArrayList<Playlist>` for storing playlists

Main solution elements:

1. **Helper class `Playlist`**
   - Stores the playlist name and a list of tracks
2. **Operations on the track collection:**
   - `addTrack` - adds a new track to the main list (checks if the track already exists using the `contains` method)
   - `removeTrack` - removes a track from the main list and from all playlists (uses the `remove` method)
   - `displayTracks` - displays all tracks using a `for` loop
   - `searchTracks` - uses loops and find (filter method) to find tracks containing the given phrase
3. **Operations on the playlist collection:**
   - `createPlaylist` - creates a new playlist (checks existence using a loop)
   - `findPlaylist` - helper method to find a playlist by name (uses a loop)
   - `addTrackToPlaylist` - adds a track to a specific playlist
   - `displayPlaylist` - displays the contents of a playlist
   - `displayAllPlaylists` - displays a list of all playlists using a `for-each` loop
   - In the example in the `main` method, we show how to:
4. Create a library through the parameterized constructor
5. Add tracks to the main collection
6. Create playlists
7. Add tracks to playlists
8. Search for tracks containing a specific phrase
9. Remove tracks (also from playlists)

Example main method:

```java
public class Program {
    public static void main(String[] args) {
```

```java
        // Create a new music library
        MusicLibrary myLibrary = new MusicLibrary("Rock Collection",
"John Smith");

        // Add tracks to the library
        myLibrary.addTrack("Led Zeppelin - Stairway to Heaven");
        myLibrary.addTrack("Queen - Bohemian Rhapsody");
        myLibrary.addTrack("Pink Floyd - Comfortably Numb");
        myLibrary.addTrack("AC/DC - Back in Black");
        myLibrary.addTrack("Metallica - Nothing Else Matters");

        // Display all tracks
        myLibrary.displayTracks();

        // Create playlists
        myLibrary.createPlaylist("Favorites");
        myLibrary.createPlaylist("Party");

        // Add tracks to playlists
        myLibrary.addTrackToPlaylist("Queen - Bohemian Rhapsody",
"Favorites");
        myLibrary.addTrackToPlaylist("Pink Floyd - Comfortably Numb",
"Favorites");
        myLibrary.addTrackToPlaylist("AC/DC - Back in Black", "Party");
        myLibrary.addTrackToPlaylist("Metallica - Nothing Else Matters",
"Party");

        // Display all playlists
        myLibrary.displayAllPlaylists();

        // Display the contents of the "Favorites" playlist
        myLibrary.displayPlaylist("Favorites");

        // Search for tracks with the phrase "Queen"
        myLibrary.searchTracks("Queen");

        // Remove a track from the library
        myLibrary.removeTrack("AC/DC - Back in Black");

        // Check if the track was removed from the playlist
        myLibrary.displayPlaylist("Party");

        System.out.println("Total number of tracks in the library: " +
myLibrary.getTrackCount());
        System.out.println("Total number of playlists: " +
myLibrary.getPlaylistCount());
```

```
    }
}
```