# CSCI 2141 ASSIGNMENT PART 2

## Section 0: Adjust Your Dataset (if Necessary)

According to the feedback, I have added information about the primary keys and foreign keys to the database schema.

a) SONG – This table is going to be a table containing data about the various songs available on Spotify.
The primary key is going to be song_id and the foreign key is going to be album_id which references the album_id in ALBUM.

b) CHART- which is a table that contains data about the top songs of the year 2022 on Spotify.
The primary key is going to be chart_id.
The foreign key is going to be artist_id which references the primary key in ARTIST i.e., artist_id.

c) ARTIST- this table is going to contain information about the various artists on Spotify.
The primary key is artist_id.
The foreign key is song_id which references the song_id in SONG.
Each artist can have multiple song and each song can have multiple artists.

d) ALBUM- this table is going to contain information about albums released by various artists.
The primary key is album_id.

Since, I have added foreign keys to the table, I have decided to remove some columns from my database to reduce redundancy, from 2 of my tables:
SONG & CHART:
These tables are not going to have duration_ms & speechiness as attributes anymore. Furthermore, I am removing the attribute weeks_on_chart from CHART.

ARTIST:

This table is not going to have top_song as an attribute, it is going to be replaced by the foreign key song_id.

I have also decided to use ChatGPT to generate the data to increase the number of rows in my dataset and make the process of dataset generation more convenient for ARTIST & ALBUM.
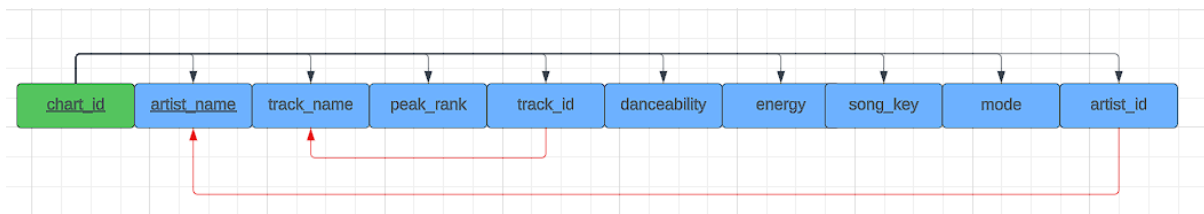
## Section 1: Design Your Database in 3NF

Design the 3NF for the database.
The database is already in 1 NF form since, there are no comma separated lists in the database.
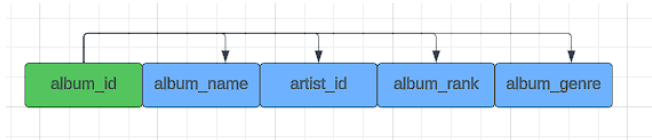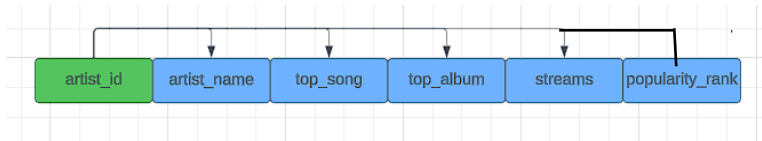Dependency diagram before normalization:
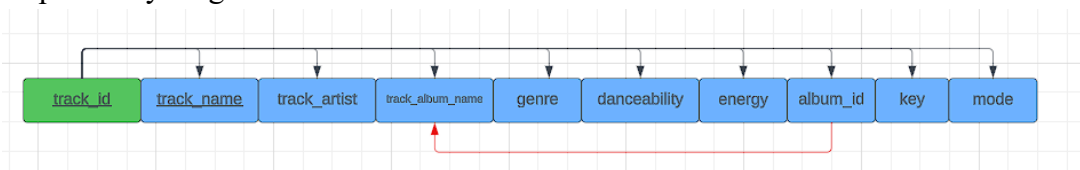
Dependency diagram after normalization:
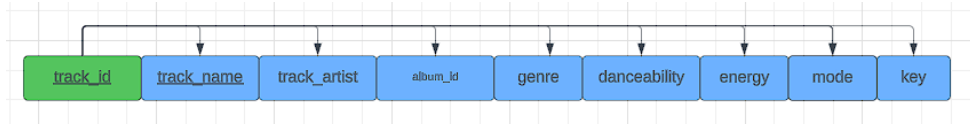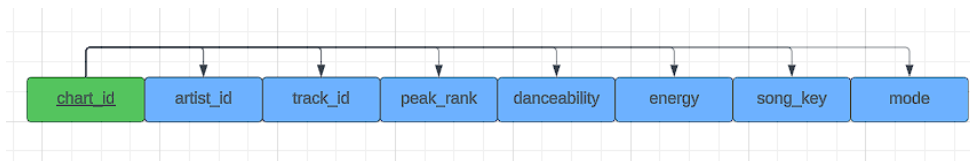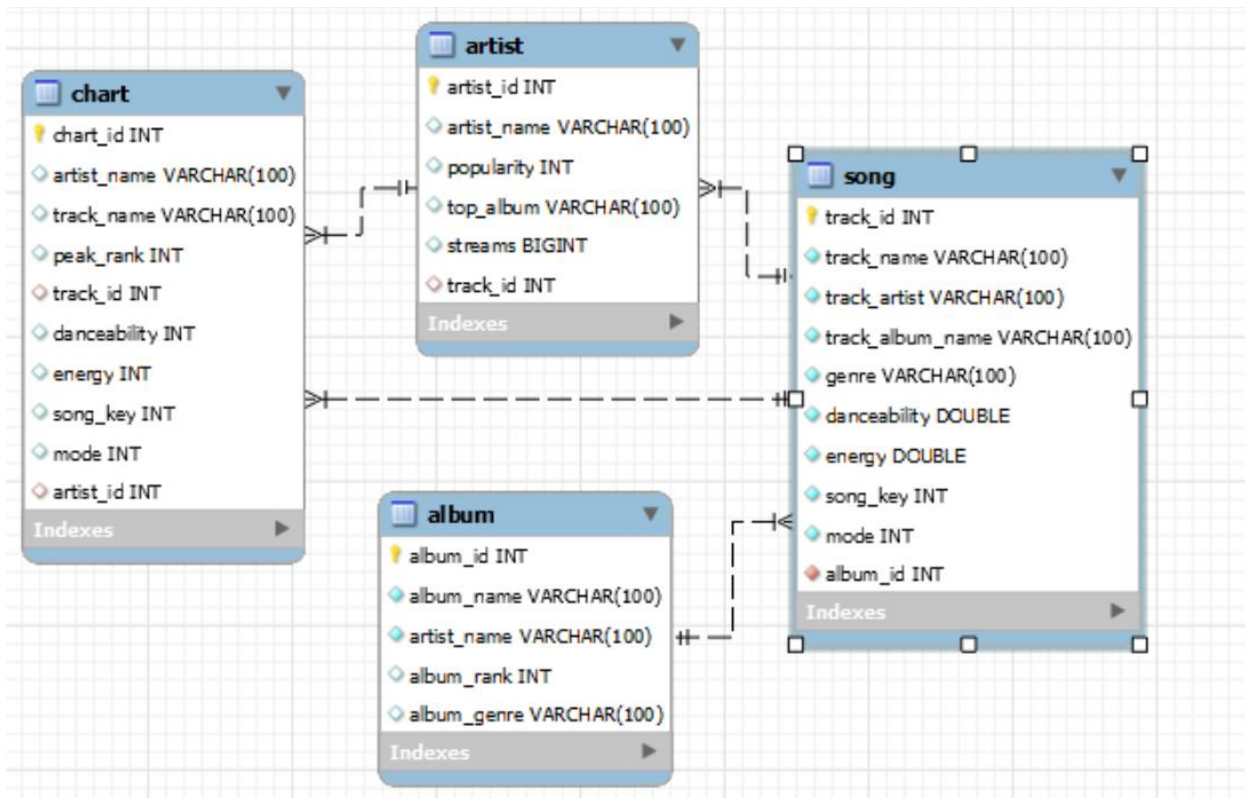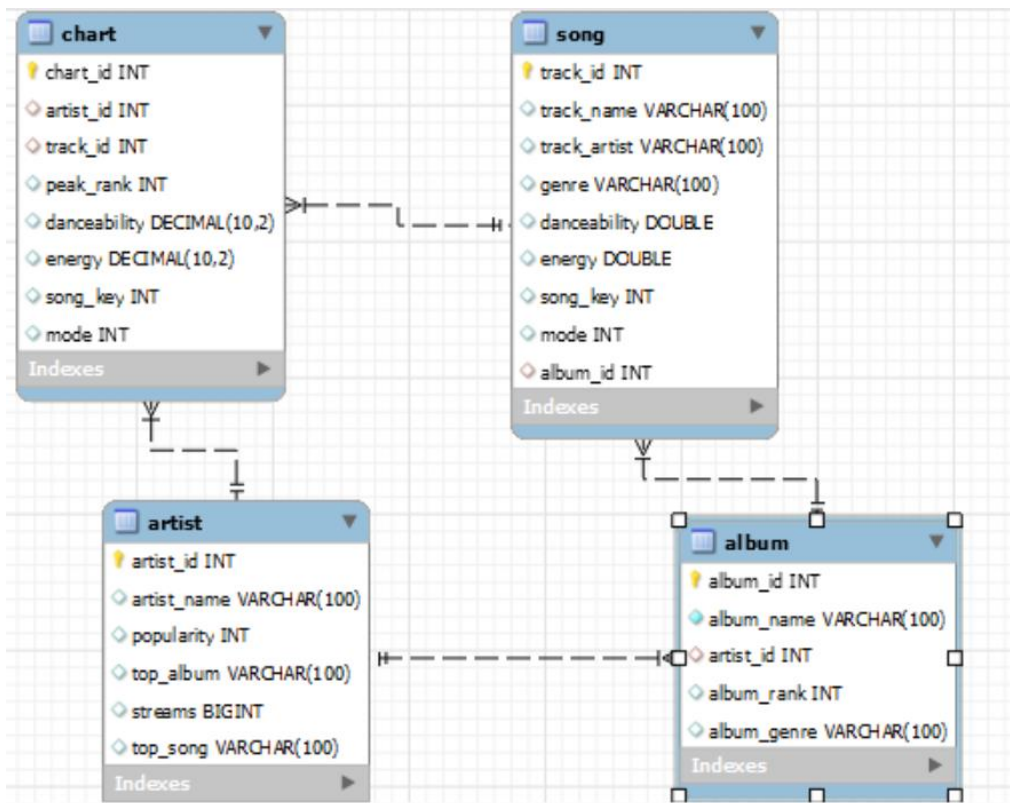
Album:



Artist:



Song:



Chart:



Internal Schema before normalization:



Internal schema after normalization:

There are no partial dependencies in the database because there is only one primary in in every single table i.e., the id of each entity. Therefore, they are in 2NF.

However, there are some transitive dependencies in the tables SONG & CHART due to the addition of foreign key attributes to the table. Which can be removed by the process of normalization, but this would not increase the number of tables in the database because, they are foreign keys which means that a table defining these attributes already exists in the database.

Which implies that the dependent non-prime attributes only need to be removed from SONG & CHART.

## SECTION 2:

Please refer the corresponding sql file.

## SECTION 3:

Where clause:

```
mysql> SELECT artist_name, popularity, top_album, top_song, COUNT(*) FROM artist WHERE streams > 300000 GROUP BY art
ist_name, popularity, top_album, top_song;
+-----------------+------------+------------+-----------------+----------+
| artist_name     | popularity | top_album  | top_song        | COUNT(*) |
+-----------------+------------+------------+-----------------+----------+
| Ed Sherran      |         20 | Divide     | Beautiful people |        1 |
| Taylor Swift    |          7 | Reputation | Lover           |        1 |
| Justin Bieber   |         29 | Believe    | Baby            |        1 |
| Zedd            |         37 | Loud       | Diamonds        |        1 |
| The Chainsmokers|         56 | Sweetner   | Thank you next  |        1 |
| Zara Larrson    |         45 | Uncover    | Venus           |        1 |
+-----------------+------------+------------+-----------------+----------+
6 rows in set (0.01 sec)
```

Inner join:

```
mysql> SELECT artist.artist_name, artist.popularity, album.album_name, album.album_genre, COUNT(*) FROM artist INNER
 JOIN album USING (artist_id) GROUP BY artist.artist_name, artist.popularity, album.album_name, album.album_genre;
+---------------+------------+--------------------------+------------+----------+
| artist_name   | popularity | album_name               | album_genre | COUNT(*) |
+---------------+------------+--------------------------+------------+----------+
| Ed Sherran    |         20 | I dont care              | Pop        |        1 |
| Taylor Swift  |          7 | The Dark Side of the Moon | Rock       |        1 |
| Justin Bieber |         29 | Thriller                 | Pop        |        1 |
| Zedd          |         37 | Abbey Road               | Rock       |        1 |
| Zedd          |         37 | Back in Black            | Hard Rock  |        1 |
+---------------+------------+--------------------------+------------+----------+
5 rows in set (0.01 sec)
```

Group by:

```
mysql> SELECT chart.peak_rank, song.track_name, song.track_artist, AVG(chart.danceability) as avg_danceability FROM
chart INNER JOIN song USING (track_id) GROUP BY chart.peak_rank, song.track_name, song.track_artist;
+-----------+---------------------------------------------------+--------------+------------------+
| peak_rank | track_name                                        | track_artist | avg_danceability |
+-----------+---------------------------------------------------+--------------+------------------+
|         1 | I Don't Care (with Justin Bieber) - Loud Luxury Remix | Ed Sheeran   |         0.730000 |
|         3 | Memories - Dillon Francis Remix                   | Maroon 5     |         0.670000 |
+-----------+---------------------------------------------------+--------------+------------------+
2 rows in set (0.01 sec)
```