select \* from release;

select\* from chart;

select \* from track;

select \* from artist;

-- Display all the song and how long they stay in the chart;

select c.Track\_Id, t.track\_name, max(c.weeks\_on\_board) as total\_weeks

from chart c

join track t on t.track\_id = c.track\_id

group by c.track\_id, t.track\_name

order by [total\_weeks] desc;

-- 1. Exmple 1: Display all of the songs in the list with their audio\_features;

--This is for the circle visulaization that I have showed you.

select t.track\_name, af.danceability, af.energy, af.loudness, af.mode, af.speechiness, af.acousticness, af.instrumentalness, af.liveness, af.valence, af.tempo

from audio\_feature af

join track t on t.Track\_Id = af.af\_track\_id;

--2 Example 2: Compare the average duration of songs over the years.

select Year(al.release\_date) as release\_year, avg(af.duration\_ms) as avg\_duration

from audio\_feature af

join album\_info ai on ai.Track\_Id = af.af\_track\_id

join album al on al.album\_id = ai.Album\_Id

group by Year(al.release\_date)

ORDER by release\_year;

--3 Example 3: Compare the average loudness of songs over the years.

SELECT

CASE

WHEN release\_year >= 1940 AND release\_year <= 1949 THEN '1940s'

WHEN release\_year >= 1950 AND release\_year <= 1959 THEN '1950s'

WHEN release\_year >= 1960 AND release\_year <= 1969 THEN '1960s'

WHEN release\_year >= 1970 AND release\_year <= 1979 THEN '1970s'

WHEN release\_year >= 1980 AND release\_year <= 1989 THEN '1980s'

WHEN release\_year >= 1990 AND release\_year <= 1999 THEN '1990s'

WHEN release\_year >= 2000 AND release\_year <= 2009 THEN '2000s'

WHEN release\_year >= 2010 AND release\_year <= 2019 THEN '2010s'

-- Add more cases for other decades as needed

ELSE 'Current Decade'

END AS decade,

release\_year, loudness

FROM (SELECT af.loudness as loudness, YEAR(al.release\_date) as release\_year

FROM audio\_feature af

join album\_info ai on ai.Track\_Id = af.af\_track\_id

join album al on al.album\_id = ai.Album\_Id) as sub\_query

ORDER BY release\_year ASC;

--4 Example 4: Compare the avg\_stream, popularity, followers

select ar.artist\_name, ar.followers, ar.popularity, avg(s.NO\_OF\_STREAMS) as avg\_stream

from artist as ar

join release r on r.Artist\_Id = ar.artist\_id

join SptfyWWDSongRanking as s on s.Track\_Id = r.Track\_Id

GROUP by ar.artist\_name, ar.followers, ar.popularity;

A picture containing screenshot, colorfulness, art

Description automatically generated

A picture containing text, screenshot, diagram, plot

Description automatically generated

A picture containing text, screenshot, diagram, plot

Description automatically generated

A picture containing plot, line, screenshot, diagram

Description automatically generatedA picture containing text, screenshot, diagram

Description automatically generated

A picture containing diagram, plan, rectangle, square

Description automatically generated