**Spotify Data Analysis**

rename table `cleaned\_dataset (1)` to Spotify;

describe spotify;

**-- 1. Retrieve the names of all tracks that have more than 1 billion streams.**

select \* from spotify;

SELECT Track as Track\_under\_1B

FROM spotify

WHERE Stream > 1000000000;

**-- 2. List all albums along with their respective artists.**

select Album, Artist from spotify;

**-- 3. Get the total number of comments for tracks where licensed = TRUE.**

SELECT count(Comments) AS total\_comments

FROM spotify

WHERE Licensed = 'true';

**-- 4. Find all tracks that belong to the album type single.**

select \* from spotify

where Album\_type = 'single';

**-- 5. Count the total number of tracks by each artist.**

select Artist, count(Track) as Total\_No\_tracks from spotify

group by 1;

**-- 6. Calculate the average danceability of tracks in each album.**

select Album, avg( Danceability) as Average\_danceablity from spotify

group by 1;

**-- 7. Find the top 5 tracks with the highest energy values.**

select Energy as Highest\_Energy\_level from spotify

order by 1 desc limit 5;

**-- 8. List all tracks along with their views and likes where official\_video = TRUE.**

select Track,Views,Likes from spotify

where official\_video= 'true';

**-- 9. For each album, calculate the total views of all associated tracks.**

select Album,sum(Views) as Total\_views from spotify

group by Album ;

**-- 10. Retrieve the track names that have been streamed on Spotify more than YouTube.**

select \* from spotify;

SELECT Track as Popular\_on\_spotify

FROM spotify

WHERE (SELECT SUM(Track) FROM spotify WHERE most\_playedon = 'Spotify') >

(SELECT SUM(Track) FROM spotify WHERE most\_playedon = 'YouTube');

**-- 11. Write a query to find tracks where the liveness score is above the average.**

select \* from spotify;

select Track, Liveness as liveness\_score

from spotify where(select count(Liveness) from spotify) >

( select avg(Liveness) from spotify)

order by 2 desc limit 5;

-- or

SELECT Track, Liveness

FROM spotify

WHERE Liveness > (SELECT AVG(Liveness) FROM spotify)

order by 2 desc limit 5;

**-- 12. Find tracks where the energy-to-liveness ratio is greater than 1.2.**

select \* from spotify

where EnergyLiveness > 1.2;