



Presented by Mitali Gupta

Home

Music

About Us

Contact

Search . . .

SPOTIFY

MUSIC

[Home](#)[Music](#)[About Us](#)[Contact](#)

Search . . .



ABOUT US

ABOUT US

Since its launch in 2008, Spotify has revolutionized music listening. Our move into podcasting brought innovation and a new generation of listeners to the medium. In 2022, we took the next leap, entering the fast-growing audiobook market—continuing to shape the future of audio.



ABOUT ME

My name is Mitali Gupta, a data analyst, developed this project to analyze Spotify's music streaming data using PostgreSQL. The objective was to extract meaningful insights into song popularity, artist performance, and user listening behavior.





CREATE TABLE

```
create table spotify (
    artist  varchar(255),
    track   varchar(255),
    album   varchar(255),
    album_type varchar(50),
    dance_ability  float,
    energy   float,
    loudness   float,
    speechiness float,
    acousticness   float,
    instrumentalness   float,
    liveness   float,
    valence float,
    tempo    float,
    duration_min float,
    title   varchar(255),
    channel varchar(255),
    track_VIEWS float,
    track_Likes float,
    track_Comments float,
    licensed   boolean,
    official_video boolean,
    stream   float,
    energy_liveness float,
    most_played_on varchar(50)
```





RETRIEVE THE NAMES OF ALL TRACKS THAT HAVE MORE THAN 1 BILLION STREAMS.

```
select track,  
      stream  
from spotify  
where stream > 1000000000
```

**LIST ALL ALBUMS ALONG WITH THEIR
RESPECTIVE ARTISTS.**

```
select distinct(album),  
artist  
from spotify
```

GET THE TOTAL NUMBER OF COMMENTS FOR TRACKS WHERE LICENSED = TRUE.

```
select licensed,  
       sum(track_comments) as "total_comments"  
  from spotify  
 where licensed = 'true'  
group by 1
```

FIND ALL TRACKS THAT BELONG TO THE ALBUM TYPE SINGLE.

```
select track,  
       album_type  
  from spotify where album_type = 'single'
```

COUNT THE TOTAL NUMBER OF TRACKS BY EACH ARTIST.

```
select artist,  
count(track) as "total_tracks"  
from spotify  
group by 1  
order by 2 desc
```

CALCULATE THE AVERAGE DANCEABILITY OF TRACKS IN EACH ALBUM.

```
select album,  
       round(avg(dance_ability::numeric),1) as "avg_dance_ability"  
  from spotify  
 group by 1
```

FIND THE TOP 5 TRACKS WITH THE HIGHEST ENERGY VALUES.

```
select track,  
       max(energy)  
     from spotify  
   group by 1  
order by 2 desc  
  limit 5
```

LIST ALL TRACKS ALONG WITH THEIR VIEWS AND LIKES WHERE OFFICIAL_VIDEO = TRUE

```
select track,
       sum(track_views) as "total_views",
       sum(track_likes) as "total_likes",
       official_video
  from spotify
 where official_video = 'true'
 group by 1,4
 order by 2 desc
 limit 5
```

**FOR EACH ALBUM, CALCULATE THE
TOTAL VIEWS OF ALL ASSOCIATED
TRACKS.**

```
select album,  
       track,  
       sum(track_views) as "total_views"  
  from spotify  
 group by 1,2  
order by 3 desc
```

RETRIEVE THE TRACK NAMES THAT HAVE BEEN STREAMED ON SPOTIFY MORE THAN YOUTUBE.

```
select * from
(select track,
coalesce(sum(case when most_played_on = 'Youtube' then stream end),0)
as "streamed_on_youtube",
coalesce(sum(case when most_played_on = 'Spotify' then stream end),0)
as "streamed_on_spotify"
from spotify
group by 1) as t1
where streamed_on_spotify > streamed_on_youtube
and
streamed_on_youtube <> 0
```

FIND THE TOP 3 MOST-VIEWED TRACKS FOR EACH ARTIST USING WINDOW FUNCTIONS.

```
select * from
  (select artist,
  track,
  sum(track_views),
  rank() over(partition by artist order by sum(track_views) desc) as "ranking"
  from spotify
  group by 1,2) as t1
where ranking < 4
```

**WRITE A QUERY TO FIND TRACKS WHERE
THE LIVENESS SCORE IS ABOVE THE
AVERAGE.**

```
select track,  
       liveness  
  from spotify  
 where liveness > (select avg(liveness) from spotify )
```

USE A WITH CLAUSE TO CALCULATE THE DIFFERENCE BETWEEN THE HIGHEST AND LOWEST ENERGY VALUES FOR TRACKS IN EACH ALBUM.

```
select album,  
       max(energy) as "max_energy",  
       min(energy) as "min_energy",  
       max(energy) - min(energy) as "difference"  
from spotify  
group by 1  
order by 1
```



Presented by Mitali Gupta

Home

Music

About Us

Contact

Search . . .

THANK

YOU

ALL I AM IS YOURS

ALL I AM IS YOURS