**MUSIC STORE**

**Created by : Vaibhav Rohidas Kamble.**

**Contact no : +91 9890969840**

**Email : vaibhavrk017@gmail.com**

In this document, we have saved the different queries used in SQL to solve the music store dataset problem statement.

Which includes CTE, Recursive CTE, Joins, and Sub-Queries.

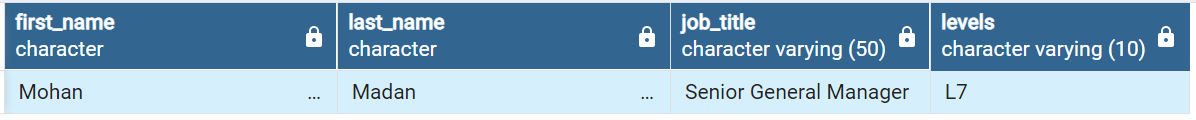
**SECTION 1 : EASY QUESTIONS**

1. Who is the senior most employee based on job title?

select first\_name, last\_name, title as job\_title, levels

from employee

order by levels desc limit 1;

****

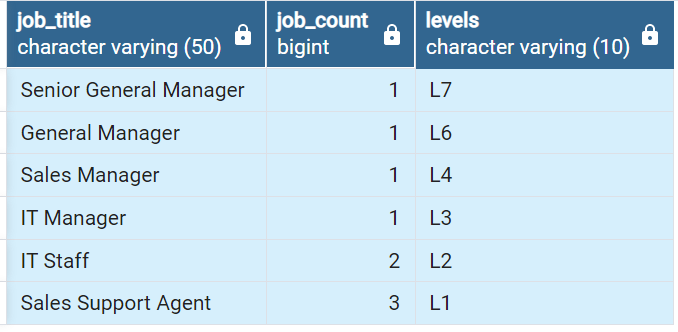
1. What are the job titles offered in this dataset?

select title as job\_title, count(\*) as job\_count, levels

from employee

group by 1,3

order by 3 desc;



1. Top 10 countries having the most Invoices?

select \* from invoice;

select count(\*) as count, billing\_country

from invoice

group by billing\_country

order by count desc limit 10;



1. what are the top 3 values of invoices?

select \* from invoice

order by total desc limit 3;



1. Which city has the best customers?

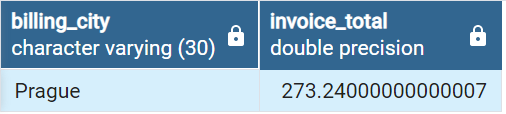
we would like to through the promotional music festival in the city where we made the most money. write a query that returns one city that has the highest sum of invoice totals. return both the city name and the sum of all invoice totals.

select billing\_city, sum(total) as invoice\_total

from invoice

group by billing\_city

order by invoice\_total desc limit 1;



1. Who is the best customer?

The customer who has spent the most money will be declared the best customer. Write the query that returns the person who has spent the most money.

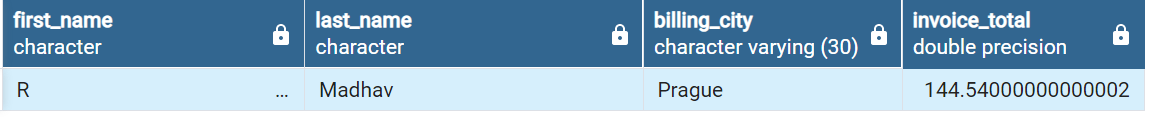
select c.first\_name, c.last\_name, i.billing\_city, sum(i.total) as invoice\_total

from customer c

left join invoice i on i.customer\_id = c.customer\_id

group by first\_name, last\_name, billing\_city

order by invoice\_total desc limit 1;



**SECTION 2 : MODERATE QUESTIONS**

1. Write a query to return the email, first name, last name, and genre of all rock music listeners

return your list ordered alphabetically by email starting with A.

select \* from customer;

select c.first\_name, c.last\_name, c.email, g.name

from customer c

join invoice i on c.customer\_id = i.customer\_id

join invoice\_line il on il.invoice\_id = i.invoice\_id

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

join genre g on g.genre\_id = t.genre\_id

group by c.first\_name, c.last\_name, c.email, g.name

having g.name like 'Rock'

order by c.email asc;

* same question using subquery / Query optimization:

select c.email, c.first\_name, c.last\_name

from customer c

join invoice i on i.customer\_id = c.customer\_id

join invoice\_line il on il.invoice\_id = i.invoice\_id

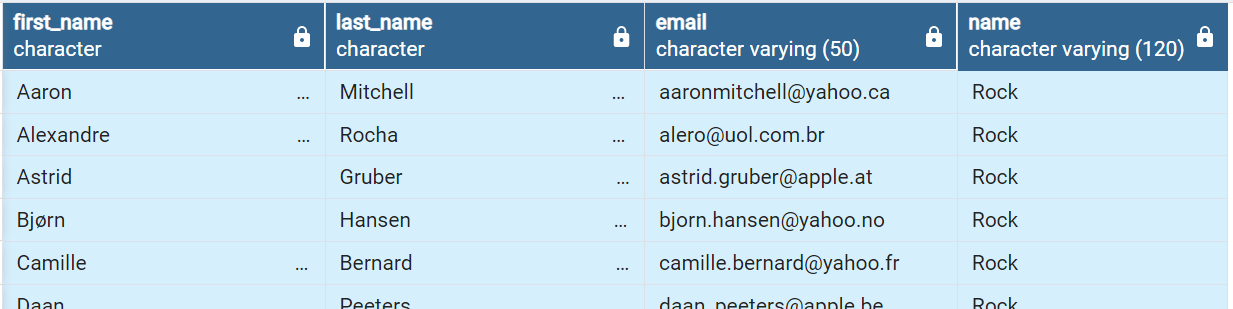
where track\_id in (select track\_id

from track t

join genre g on t.genre\_id = g.genre\_id

where g.name like 'Rock')

order by email;



1. Let's invite the artists who have written the most rock music in our dataset. Write the query that returns the artist name and total track count of the top 10 rock bands.

select a.artist\_id, a.name, count(t.name) as track\_count

from artist a

join album m on a.artist\_id = m.artist\_id

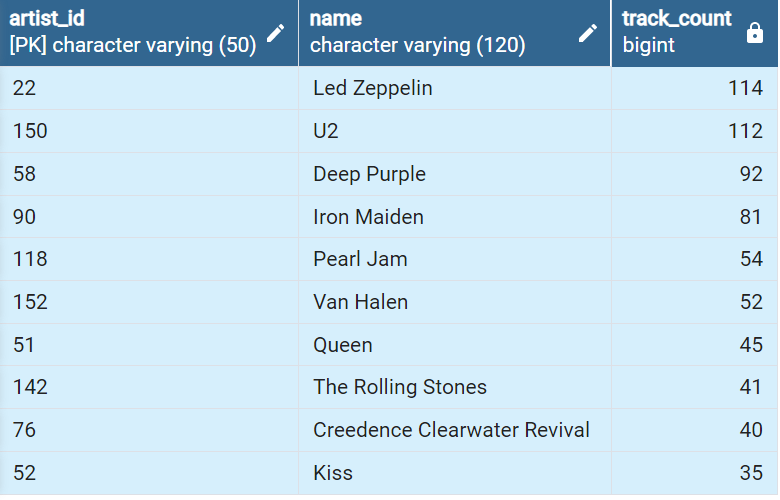
join track t on m.album\_id = t.album\_id

join genre g on g.genre\_id = t.genre\_id

where g.name like 'Rock'

group by a.artist\_id, a.name

order by track\_count desc limit 10;



1. Written all the track names that have song lengths longer than the average song length.

Return the Name and Milliseconds for each track. Order by the song length with the longest

songs listed first.

select name, milliseconds

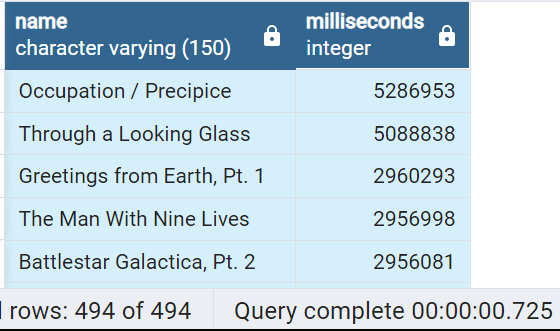
from track

where milliseconds > (

select avg(milliseconds) as avg\_song\_length

from track)

order by milliseconds desc;



**SECTION 3 : CTE - ADVANCED QUESTIONS**

1. Find how much amount spent by each customer on artists? Write a query to return the customer

name, artist name, and total spend.

with best\_selling\_artist as(

select a.artist\_id, a.name as artist\_name,

sum(il.unit\_price\*il.quantity) as total\_sales

from invoice\_line il

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

join album on album.album\_id = t.album\_id

join artist a on a.artist\_id = album.artist\_id

group by 1

order by 3 desc limit 1

)

select c.customer\_id, c.first\_name, c.last\_name,bsa.artist\_name,

sum(il.unit\_price\*il.quantity) as total\_spend

from invoice i

join customer c on c.customer\_id = i.customer\_id

join invoice\_line il on il.invoice\_id = i.invoice\_id

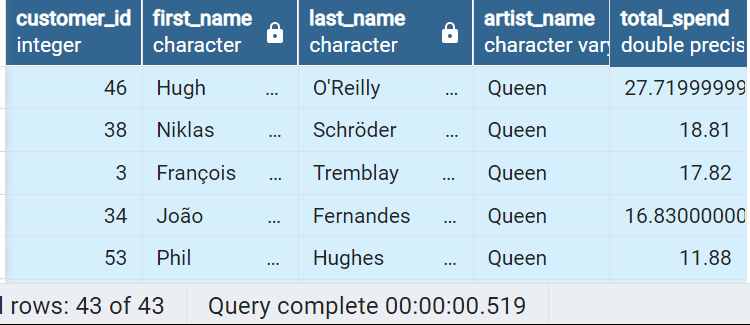
join track t on t.track\_id = il.track\_id

join album on album.album\_id = t.album\_id

join best\_selling\_artist bsa on bsa.artist\_id = album.artist\_id

group by 1,2,3,4

order by 5 desc;



1. We want to find out the most popular music genre for each country. We determine the most

popular genre as the genre with the highest amount of purchases. Write a query that returns

each country along with the top genre for the country where a maximum number of purchases is shared

and returns all genres.

* Method 1: using CTE

with popular\_genre as

( select count(il.quantity) as purchases, c.country, g.name as genre\_name, g.genre\_id,

row\_number() over(partition by c.country order by count(il.quantity) desc) as row\_number

from invoice\_line il

join invoice i on i.invoice\_id = il.invoice\_id

join customer c on c.customer\_id = i.customer\_id

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

join genre g on g.genre\_id = t.genre\_id

group by 2,3,4

order by 2 asc, 1 desc

)

select \* from popular\_genre where row\_number <= 1 order by purchases desc;

* Method 2: using recursive CTE

with recursive

sales\_per\_country as(

select count(\*) as purchase\_per\_genre, c.country, g.name as genre\_name, g.genre\_id

from invoice\_line il

join invoice i on i.invoice\_id = il.invoice\_id

join customer c on c.customer\_id = i.customer\_id

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

join genre g on g.genre\_id = t.genre\_id

group by 2,3,4

order by 2 asc

),

max\_genre\_per\_country as (select max(purchase\_per\_genre) as max\_genre\_number, country

from sales\_per\_country

group by 2

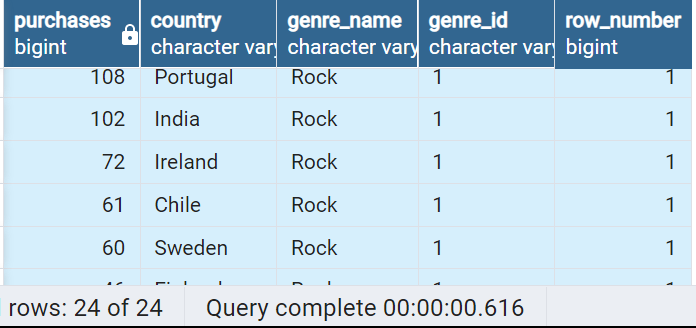
order by 2)

select sales\_per\_country.\*

from sales\_per\_country

join max\_genre\_per\_country on max\_genre\_per\_country.country = sales\_per\_country.country

where sales\_per\_country.purchase\_per\_genre = max\_genre\_per\_country.max\_genre\_number;



1. Write a query that determines the customer that has spent the most on music for each country.

Write a query that returns the country along with the customers and how much they spent. for countries

where the top amount spent is shared, provide all customers who spent this amount.

* Method 1: Using Recursive CTE

with recursive

customer\_with\_country as(

select c.customer\_id, c.first\_name, c.last\_name, i.billing\_country, sum(i.total) as total\_spending

from invoice i

join customer c on c.customer\_id = i.customer\_id

group by 1,2,3,4

order by 2,3 desc),

country\_max\_spending as(

select max(total\_spending) as max\_spending, billing\_country

from customer\_with\_country

group by billing\_country)

select cc.first\_name, cc.last\_name, cc.billing\_country, cc.total\_spending, cc.customer\_id

from customer\_with\_country cc

join country\_max\_spending ms on cc.billing\_country = ms.billing\_country

where cc.total\_spending = ms.max\_spending

order by 3;

* Method 2: Using CTE

with customer\_with\_country as(

select c.customer\_id, first\_name, last\_name, billing\_country, sum(total) as total\_spending,

row\_number() over(partition by billing\_country order by sum(total) desc ) as row\_number

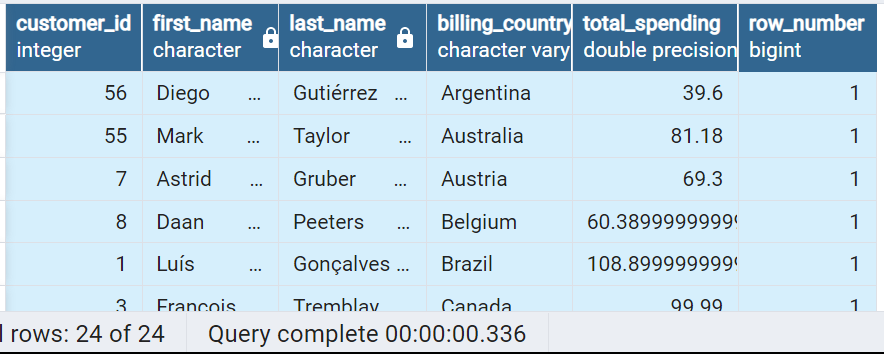
from invoice i

join customer c on c.customer\_id = i.customer\_id

group by 1,2,3,4

order by 4 asc, 5 desc)

select \* from customer\_with\_country where row\_number <= 1;



**Keep Learning New Things 😊**