

SQL PROJECT ON MUSIC STORED DATA

PAGE 01

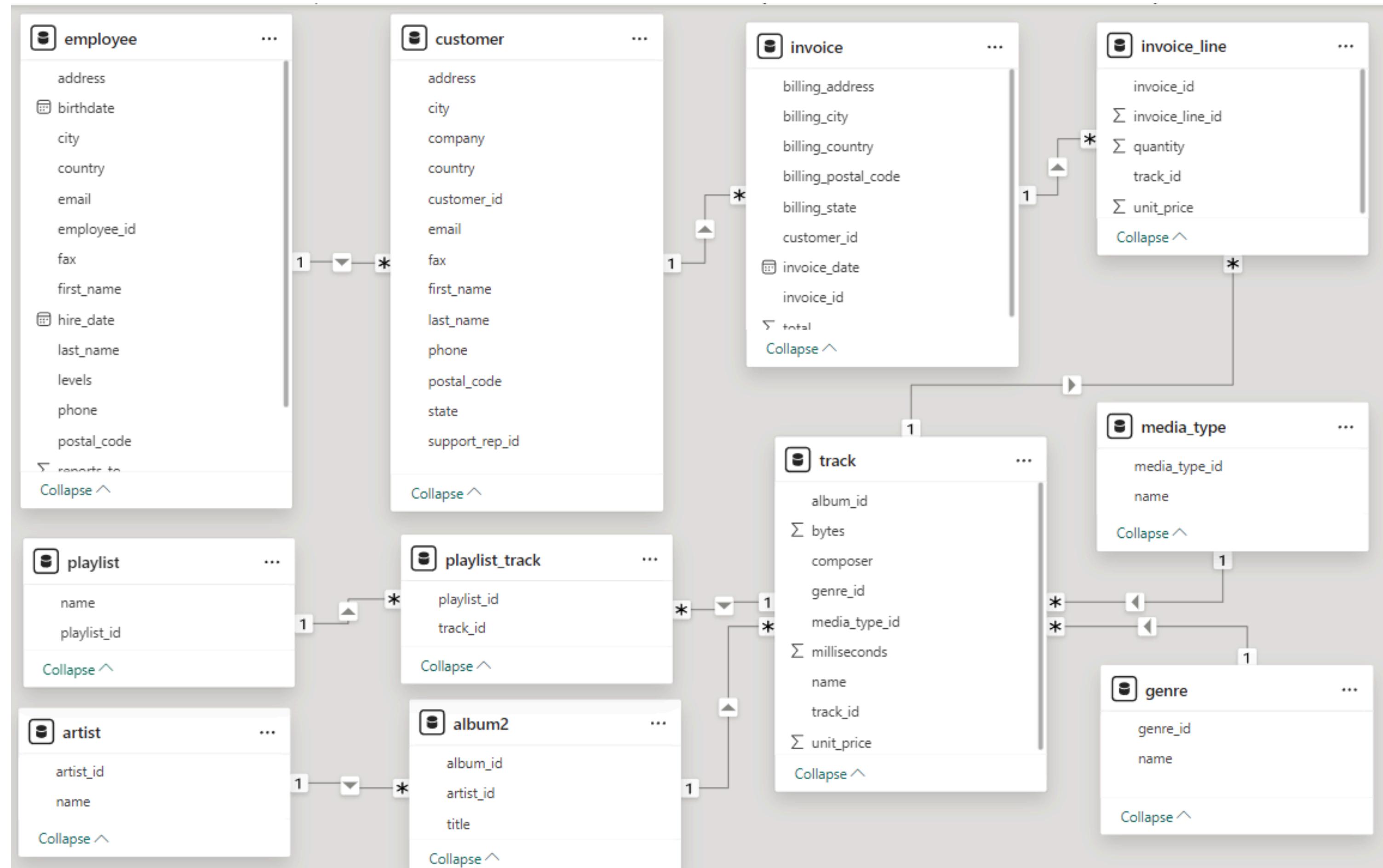
WELCOME TO THE PRESENTATION ON SQL QUERY SOLVING FOR A PROJECT FOCUSED ON MANAGING AND ANALYZING MUSIC-RELATED DATA. IN THIS PROJECT, WE WILL EXPLORE HOW SQL (STRUCTURED QUERY LANGUAGE) IS UTILIZED TO EXTRACT VALUABLE INSIGHTS FROM A MUSIC DATABASE. THE DATA INCLUDES INFORMATION SUCH AS SONGS, ARTISTS, ALBUMS, GENRES, AND PLAYLISTS.



START SLIDE

SCHEMA

PAGE 02



```
3 • select * from employee  
order by levels desc  
5 limit 1;
```



Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: | Fetch rows: |

	employee_id	last_name	first_name	title	reports_to	levels	birthdate	hire_date	address	city	state	country	postal_code
▶	9	Madan	Mohan	Senior General Manager	NULL	L7	1961-01-26	2016-01-14	1008 Vrinda Ave MT	Edmonton	AB	Canada	T5K 2N1
●	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL



- ```
select count(*) as c,billing_country
from invoice
group by billing_country
order by c desc;
```



|   | c   | billing_country |
|---|-----|-----------------|
| ▶ | 131 | USA             |
|   | 76  | Canada          |
|   | 61  | Brazil          |
|   | 50  | France          |
|   | 41  | Germany         |
|   | 30  | Czech Republic  |
|   | 29  | Portugal        |
|   | 28  | United Kingdom  |
|   | 21  | India           |
|   | 13  | Chile           |
|   | 13  | Ireland         |



- `select total from invoice  
order by total desc  
limit 3;`



|   | total              |
|---|--------------------|
| ▶ | 23.759999999999998 |
|   | 19.8               |
|   | 19.8               |

Q4: WHICH CITY HAS THE BEST CUSTOMERS? WE WOULD LIKE TO THROW A PROMOTIONAL MUSIC FESTIVAL IN THE CITY 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 & SUM OF ALL INVOICE TOTALS

PAGE 06

- ```
select sum(total) as invoice_total,billing_city
from invoice
group by billing_city
order by invoice_total desc;
```



Result Grid | Filter Rows:

	invoice_total	billing_city
▶	273.24000000000007	Prague
	169.29000000000002	Mountain View
	166.31999999999996	London
	158.39999999999998	Berlin
	151.46999999999997	Paris
	129.69000000000003	SÃ£o Paulo
	114.83999999999999	Dublin
	111.86999999999998	Delhi

Q5: WHO IS THE BEST CUSTOMER? THE CUSTOMER WHO HAS SPENT THE MOST MONEY WILL BE DECLARED THE BEST CUSTOMER. WRITE A QUERY THAT RETURNS THE PERSON WHO HAS SPENT THE MOST MONEY.

PAGE 07



```
select C.customer_id, C.first_name, C.last_name, sum(I.total) as total  
from customer as C  
inner join invoice as I on C.customer_id = I.customer_id  
group by C.customer_id  
order by total desc  
limit 1;
```

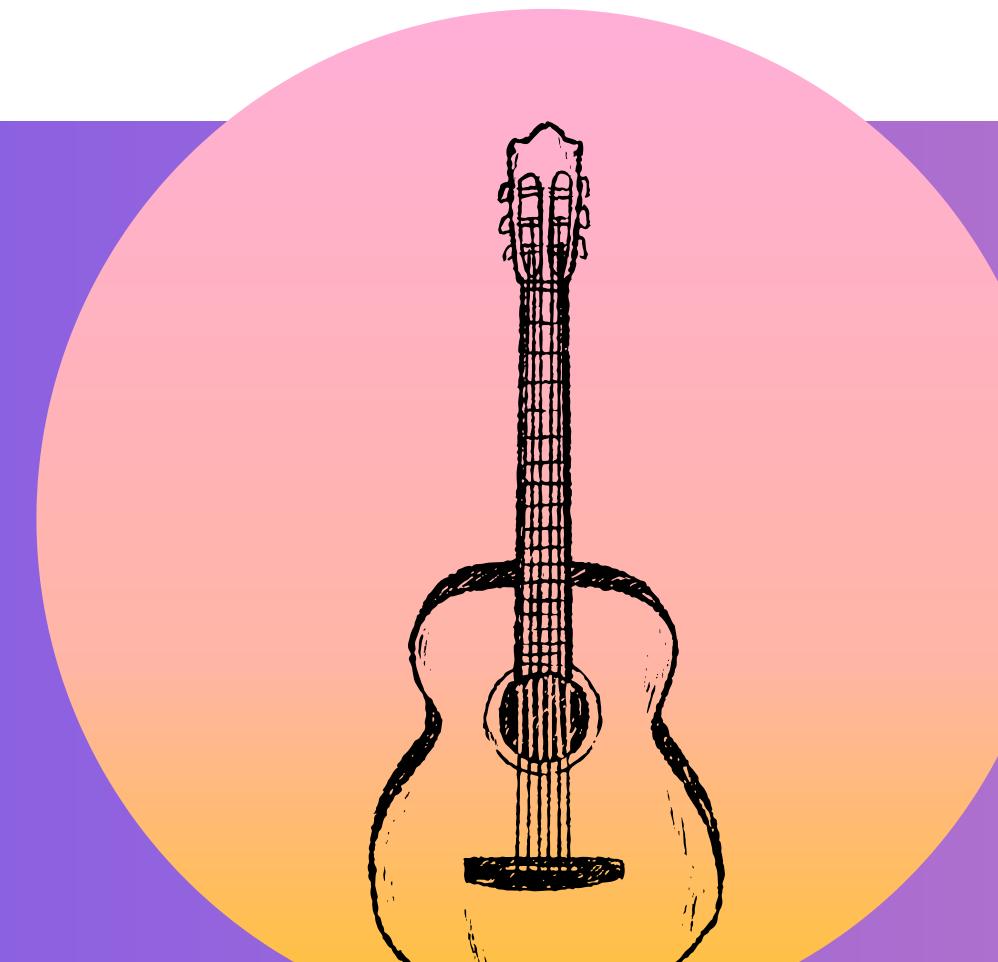
Result Grid | Filter Rows: Export:

	customer_id	first_name	last_name	total
▶	5	František	Widterlová	144.54

Q6: WRITE QUERY TO RETURN THE EMAIL, FIRST NAME, LAST NAME, & GENRE OF ALL ROCK MUSIC LISTENERS. RETURN YOUR LIST ORDERED ALPHABETICALLY BY EMAIL STARTING WITH A

PAGE 08

- ```
select distinct email, first_name, last_name
from customer
inner join invoice on customer.customer_id = invoice.customer_id
inner join invoice_line on invoice.invoice_id = invoice_line.invoice_id
where track_id in(
 select track_id from track
 inner join genre on track.genre_id = genre.genre_id
 where genre.name like "Rock"
)
Order by email Asc;
```



|  | email                       | first_name | last_name |
|--|-----------------------------|------------|-----------|
|  | aaronmitchell@yahoo.ca      | Aaron      | Mitchell  |
|  | alero@uol.com.br            | Alexandre  | Rocha     |
|  | bjorn.hansen@yahoo.no       | Bjørn      | Hansen    |
|  | camille.bernard@yahoo.fr    | Camille    | Bernard   |
|  | daan_peeters@apple.be       | Daan       | Peeters   |
|  | diego.gutierrez@yahoo.ar    | Diego      | Gutiérrez |
|  | dmiller@comcast.com         | Dan        | Miller    |
|  | dominiquelefrevre@gmail.com | Dominique  | Lefebvre  |
|  | edfrancis@yahoo.ca          | Edward     | Francis   |
|  | eduardo@woodstock.com.br    | Eduardo    | Martins   |
|  | ellie.sullivan@shaw.ca      | Ellie      | Sullivan  |
|  | emma_jones@hotmail.com      | Emma       | Jones     |
|  | enrique_munoz@yahoo.es      | Enrique    | Muñoz     |

Q7: LET'S INVITE THE ARTISTS WHO HAVE WRITTEN THE MOST ROCK MUSIC IN OUR DATASET.  
WRITE A QUERY THAT RETURNS THE ARTIST NAME AND TOTAL TRACK COUNT OF THE TOP 10  
ROCK BANDS

PAGE 09

- ```
select artist.artist_id, artist.name, count(artist.artist_id) as number_of_songs
from track
inner join album on track.album_id = album.album_id
inner join artist on album.artist_id = artist.artist_id
inner join genre on genre.genre_id = track.genre_id
where genre.name like "Rock"
group by artist.artist_id
order by number_of_songs desc
limit 10;
```



Result Grid | Filter Rows:

	artist_id	name	number_of_songs
▶	3	Aerosmith	15
	8	Audioslave	14
	22	Led Zeppelin	14
	4	Alanis Morissette	13
	5	Alice In Chains	12
	1	AC/DC	8
	2	Accept	4

Q8: RETURN ALL THE TRACK NAMES THAT HAVE A SONG LENGTH 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.

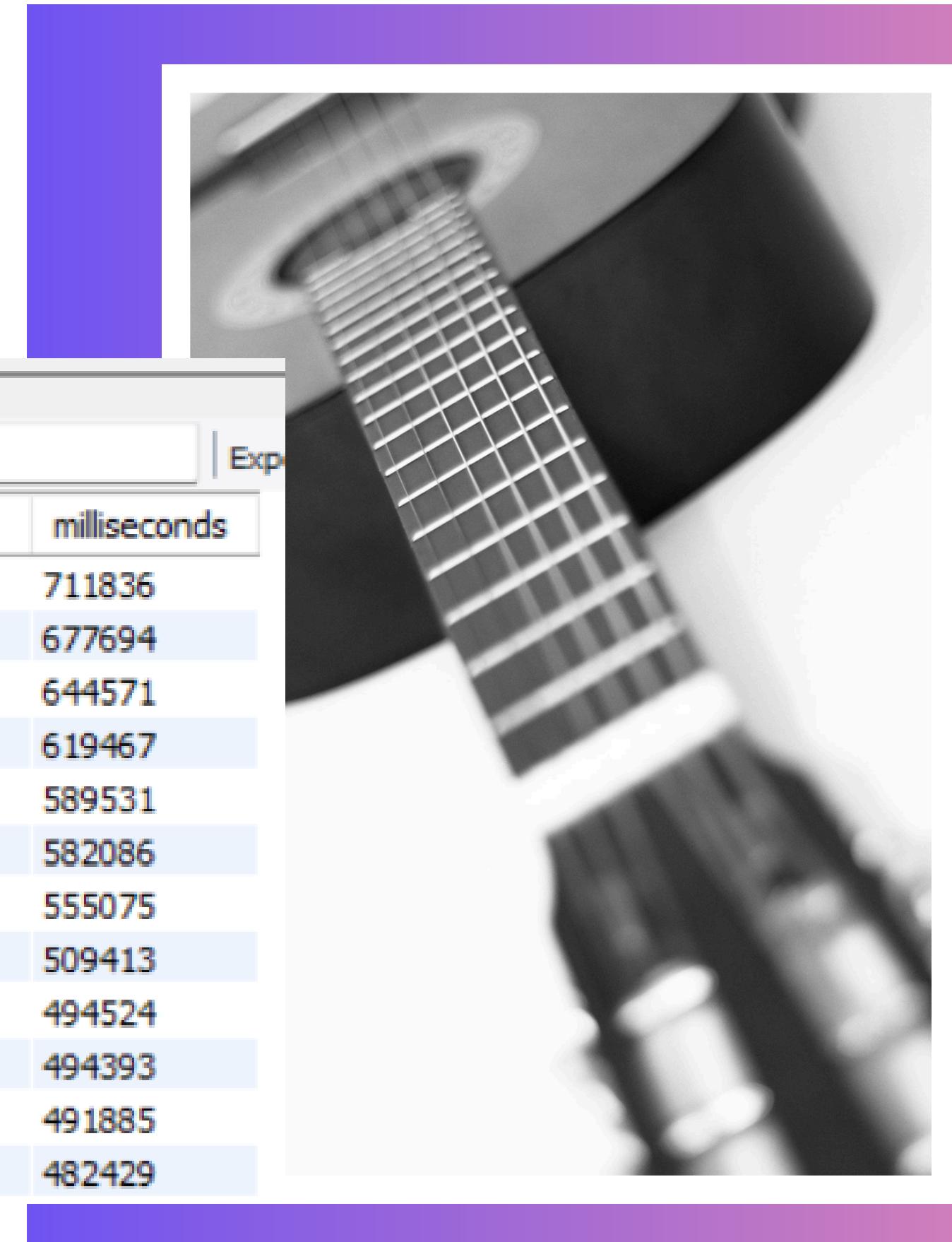
PAGE 10

- ```
select name, milliseconds
 from track
 where milliseconds > (
 select avg(milliseconds) as average_track_length
 from track
)
 order by milliseconds desc;
```



Result Grid | Filter Rows:  | Export

| name                                   | milliseconds |
|----------------------------------------|--------------|
| How Many More Times                    | 711836       |
| Advance Romance                        | 677694       |
| Sleeping Village                       | 644571       |
| You Shook Me(2)                        | 619467       |
| Talkin' 'Bout Women Obviously          | 589531       |
| Stratus                                | 582086       |
| No More Tears                          | 555075       |
| The Alchemist                          | 509413       |
| Wheels Of Confusion / The Straightener | 494524       |
| Book Of Thel                           | 494393       |
| You Oughta Know (Alternate)            | 491885       |
| Terra                                  | 482429       |



# Q9: FIND HOW MUCH AMOUNT SPENT BY EACH CUSTOMER ON ARTISTS? WRITE A QUERY

PAGE 06

TO RETURN CUSTOMER NAME, ARTIST NAME AND TOTAL SPENT

```
• WITH best_selling_artist AS (
 select artist.artist_id AS artist_id, artist.name AS artist_name,
 SUM(invoice_line.unit_price*invoice_line.quantity) AS total_sales
 from invoice_line
 inner join track on track.track_id = invoice_line.track_id
 inner join album on album.album_id = track.album_id
 inner join artist on artist.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 amount_spend
from invoice as i
inner join customer as c on c.customer_id = i.customer_id
inner join invoice_line as il on il.invoice_id = i.invoice_id
inner join track as t on t.track_id = il.track_id
inner join album as alb on alb.album_id = t.album_id
inner join best_selling_artist as bsa on bsa.artist_id = alb.artist_id
group by 1,2,3,4
order by 5 desc;
```

|   | customer_id | first_name | last_name | artist_name | amount_spend       |
|---|-------------|------------|-----------|-------------|--------------------|
| ▶ | 15          | Jennifer   | Peterson  | Aerosmith   | 14.850000000000001 |
|   | 55          | Mark       | Taylor    | Aerosmith   | 14.850000000000001 |
|   | 57          | Luis       | Rojas     | Aerosmith   | 2.969999999999998  |
|   | 43          | Isabelle   | Mercier   | Aerosmith   | 2.969999999999998  |
|   | 59          | Rishabh    | Mishra    | Aerosmith   | 1.98               |
|   | 3           | FranÃ§ois  | Tremblay  | Aerosmith   | 1.98               |
|   | 8           | Daan       | Peeters   | Aerosmith   | 1.98               |
|   | 38          | Niklas     | SchrÃ¶der | Aerosmith   | 1.98               |
|   | 6           | Helena     | HolÃ½     | Aerosmith   | 1.98               |
|   | 30          | Edward     | Francis   | Aerosmith   | 1.98               |
|   | 12          | Roberto    | Almeida   | Aerosmith   | 1.98               |
|   | 42          | Wyatt      | Girard    | Aerosmith   | 0.99               |
|   | 41          | Marc       | Dubois    | Aerosmith   | 0.99               |
|   | 51          | Joakim     | Johansson | Aerosmith   | 0.99               |

Q10: 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 PAGE 08 ALONG WITH THE TOP GENRE. FOR COUNTRIES WHERE THE MAXIMUM NUMBER OF PURCHASES IS SHARED RETURN ALL GENRES.

- **WITH popular\_genre AS**

```
 (
 select COUNT(invoice_line.quantity) as purchases, customer.country, genre.name, genre.genre_id,
 ROW_NUMBER() OVER(PARTITION BY customer.country ORDER BY COUNT(invoice_line.quantity) DESC) AS RowNo
 from invoice_line
 inner join invoice on invoice.invoice_id = invoice_line.invoice_id
 inner join customer on customer.customer_id = invoice.customer_id
 inner join track on track.track_id = invoice_line.track_id
 inner join genre on genre.genre_id = track.genre_id
 group by 2,3,4
 order by 2 asc, 1 desc
)
 select * from popular_genre where RowNo <= 1;
```



|   | purchases | country        | name | genre_id | RowNo |
|---|-----------|----------------|------|----------|-------|
| ▶ | 1         | Argentina      | Rock | 1        | 1     |
|   | 18        | Australia      | Rock | 1        | 1     |
|   | 5         | Belgium        | Rock | 1        | 1     |
|   | 14        | Brazil         | Jazz | 2        | 1     |
|   | 57        | Canada         | Rock | 1        | 1     |
|   | 7         | Chile          | Rock | 1        | 1     |
|   | 14        | Czech Republic | Rock | 1        | 1     |
|   | 6         | Denmark        | Rock | 1        | 1     |
|   | 6         | Finland        | Rock | 1        | 1     |
|   | 26        | France         | Rock | 1        | 1     |
|   | 28        | Germany        | Rock | 1        | 1     |
|   | 4         | Hungary        | Rock | 1        | 1     |
|   | 13        | India          | Rock | 1        | 1     |
|   | 2         | Ireland        | Rock | 1        | 1     |

Q11: 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 TOP CUSTOMER AND HOW MUCH THEY SPENT. FOR COUNTRIES WHERE THE TOP AMOUNT SPENT IS SHARED, PROVIDE ALL CUSTOMERS WHO SPENT THIS AMOUNT

```
• with Customer_with_country as (
 select customer.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 RowNo
 from invoice
 inner join customer on customer.customer_id = invoice.customer_id
 group by 1,2,3,4
 order by 4 asc,5 desc)
select * from Customer_with_country where RowNo <= 1;
```

|   | customer_id | first_name | last_name   | billing_country | total_spending    | RowNo |
|---|-------------|------------|-------------|-----------------|-------------------|-------|
| ▶ | 56          | Diego      | Gutiérrez   | Argentina       | 39.6              | 1     |
|   | 55          | Mark       | Taylor      | Australia       | 81.18             | 1     |
|   | 8           | Daan       | Peeters     | Belgium         | 60.38999999999999 | 1     |
|   | 13          | Fernanda   | Ramos       | Brazil          | 106.9199999999997 | 1     |
|   | 3           | François   | Tremblay    | Canada          | 99.99             | 1     |
|   | 57          | Luis       | Rojas       | Chile           | 97.0200000000001  | 1     |
|   | 5           | František  | Wichterlová | Czech Republic  | 144.54            | 1     |
|   | 9           | Kara       | Nielsen     | Denmark         | 37.62             | 1     |
|   | 44          | Terhi      | Hämäläinen  | Finland         | 79.2              | 1     |
|   | 42          | Wyatt      | Girard      | France          | 99.99             | 1     |
|   | 37          | Fynn       | Zimmermann  | Germany         | 94.0500000000001  | 1     |
|   | 45          | Ladislav   | Kovács      | Hungary         | 78.21             | 1     |

**Thank You  
for Tuning In!**

