DIGITAL MUSIC STORE ANALYSIS

Using PostgreSQL

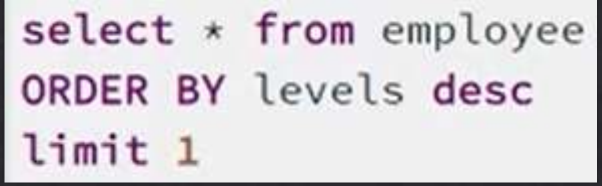
Project Objective:

- It is a digital music store sales analysis project. I analyze and query the data sets using facts, metrics, and data to guide strategic business decisions that align with your goals, objectives, and initiatives.
- For example here I extracted meaningful information such as total invoices, best customers, popular artists, tracks, etc.

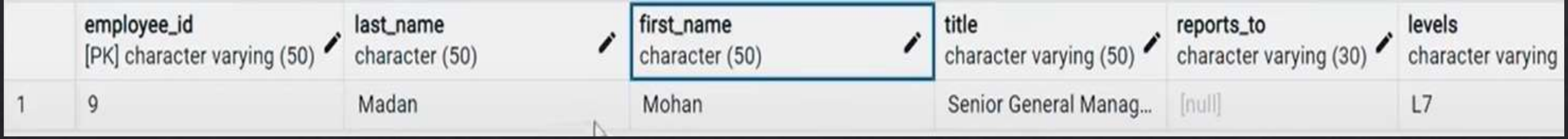
About the project :

- Utilized **PostgreSQL** to extract data from 11 different related tables from music store databases using **JOIN** and **VIEW**
- Transformed and filtered data by using aggregating and filtering functions to improve the reporting process
- Write 11 queries to extract meaningful insights from the datasets and use them to make the final report
- Used **window function**, **joins**, **cte**, **subquery** formed schema or data model using foreign key for query the database appropiately
- Loaded all information into **PowerPoint** to represent key business intelligence that can improve sales performance

Q1: Who is the senior most employee based on job title?

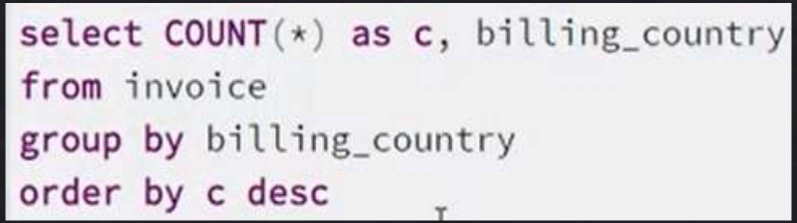


```
select * from employee
ORDER BY levels desc
limit 1
```

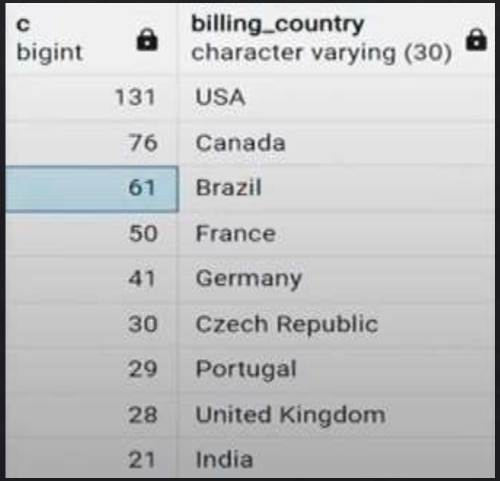


	employee_id [PK] character varying (50)	last_name character (50)	first_name character (50)	title character varying (50)	reports_to character varying (30)	levels character varying
1	9	Madan	Mohan	Senior General Manag...	[null]	L7

Q2: Which countries have the most Invoices?

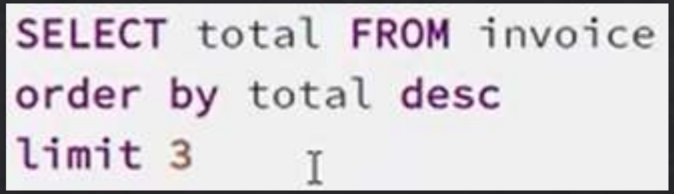


```
select COUNT(*) as c, billing_country
from invoice
group by billing_country
order by c desc
```

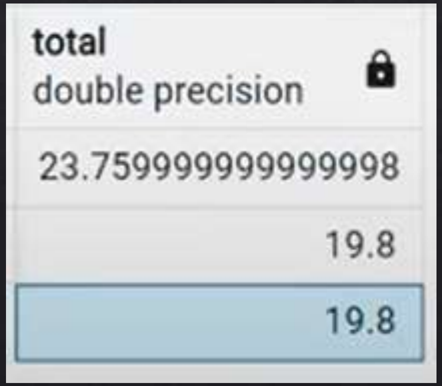


c bigint	billing_country character varying (30)
131	USA
76	Canada
61	Brazil
50	France
41	Germany
30	Czech Republic
29	Portugal
28	United Kingdom
21	India

Q3: What are the top 3 values of the total invoice?

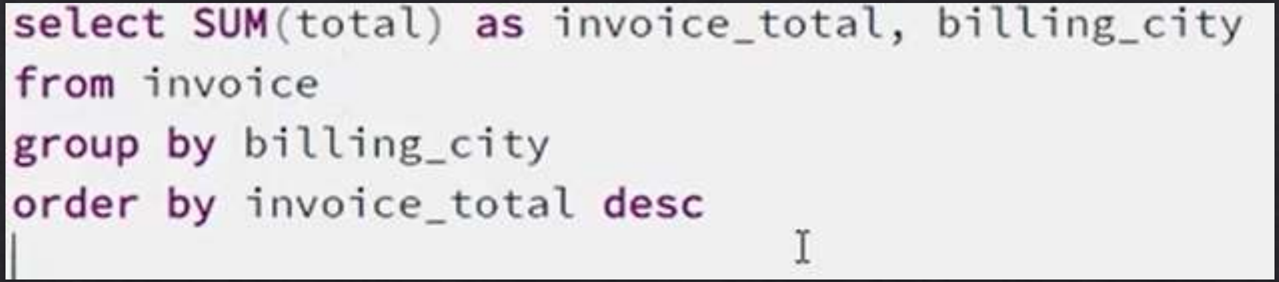


```sql
SELECT total FROM invoice
order by total desc
limit 3
```

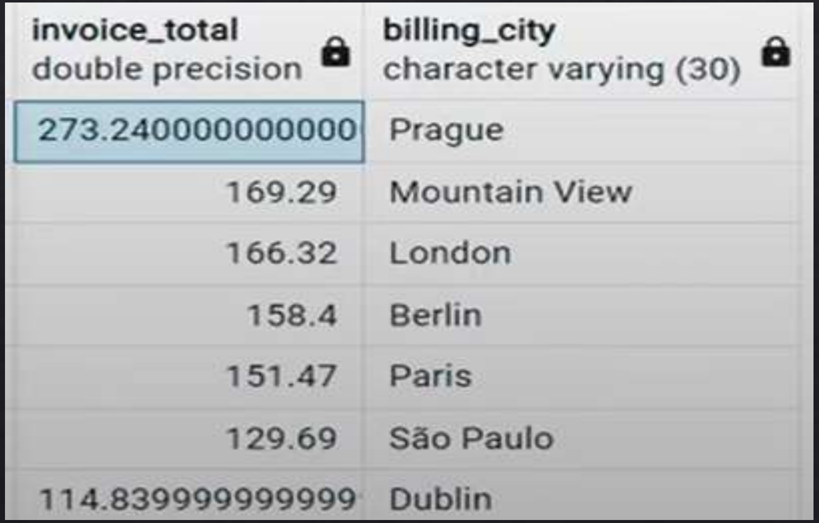


total double precision
23.759999999999998
19.8
19.8

Q4: Which city has the best customers? We want to throw a promotional Music Festival in the town where we make 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

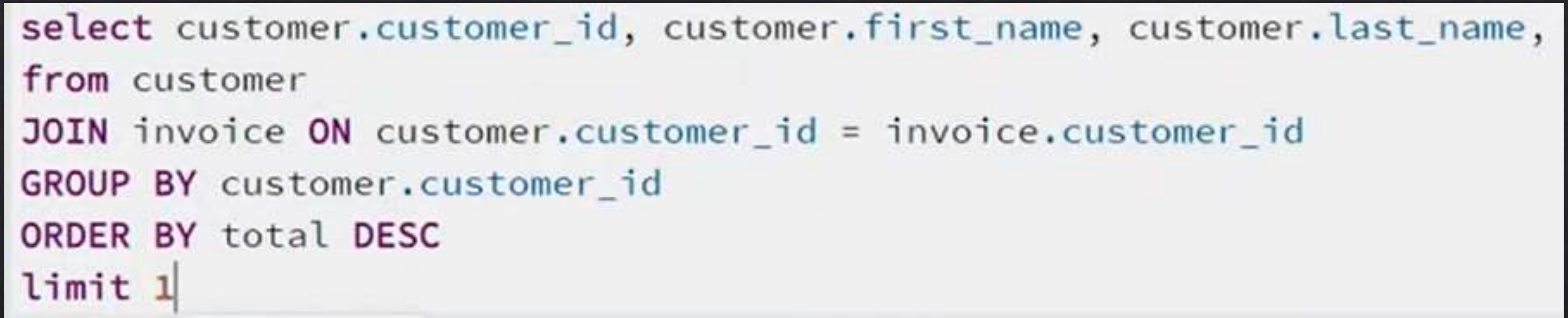


```sql
select SUM(total) as invoice_total, billing_city
from invoice
group by billing_city
order by invoice_total desc
```

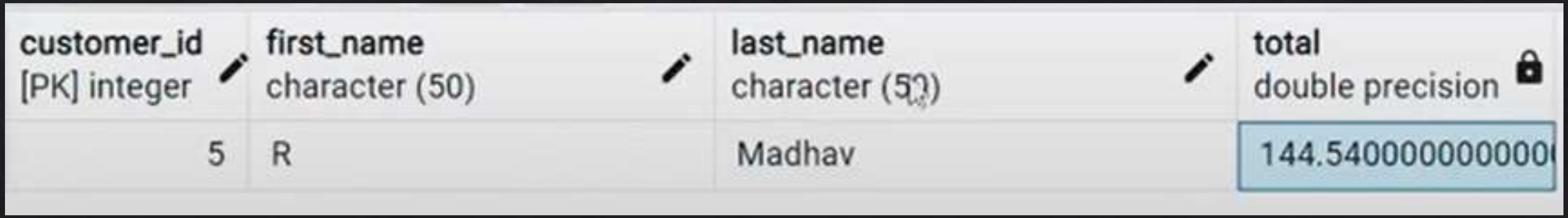


invoice_total double precision	billing_city character varying (30)
273.240000000000	Prague
169.29	Mountain View
166.32	London
158.4	Berlin
151.47	Paris
129.69	São Paulo
114.839999999999	Dublin

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.

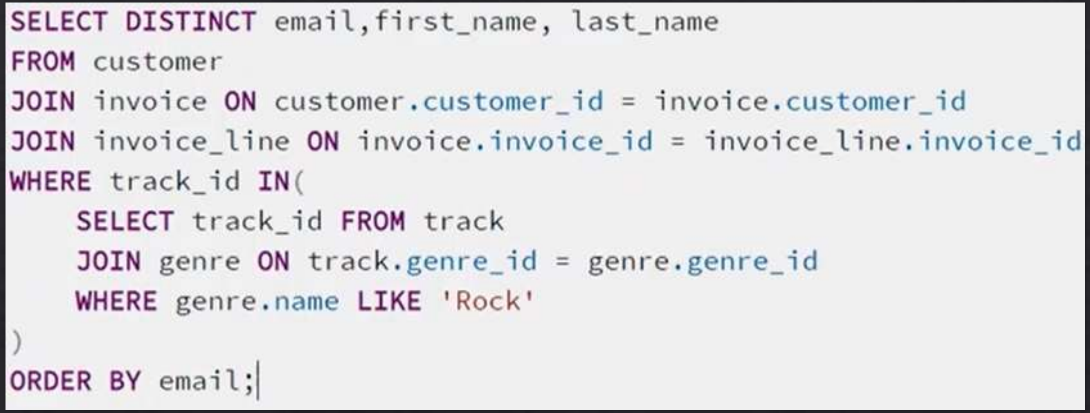


```sql
select customer.customer_id, customer.first_name, customer.last_name,
from customer
JOIN invoice ON customer.customer_id = invoice.customer_id
GROUP BY customer.customer_id
ORDER BY total DESC
limit 1
```

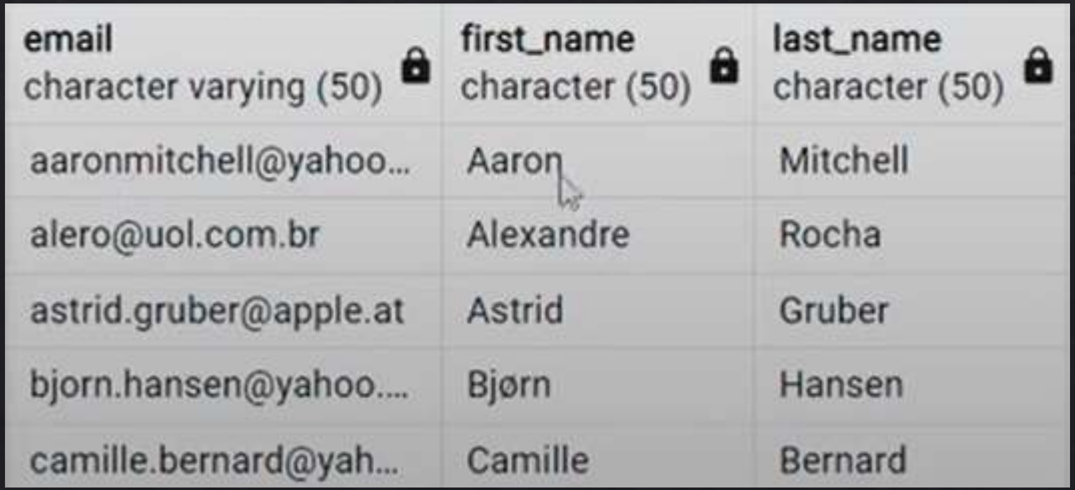


customer_id [PK] integer	first_name character (50)	last_name character (50)	total double precision
5	R	Madhav	144.540000000000

Q6: Write a 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.

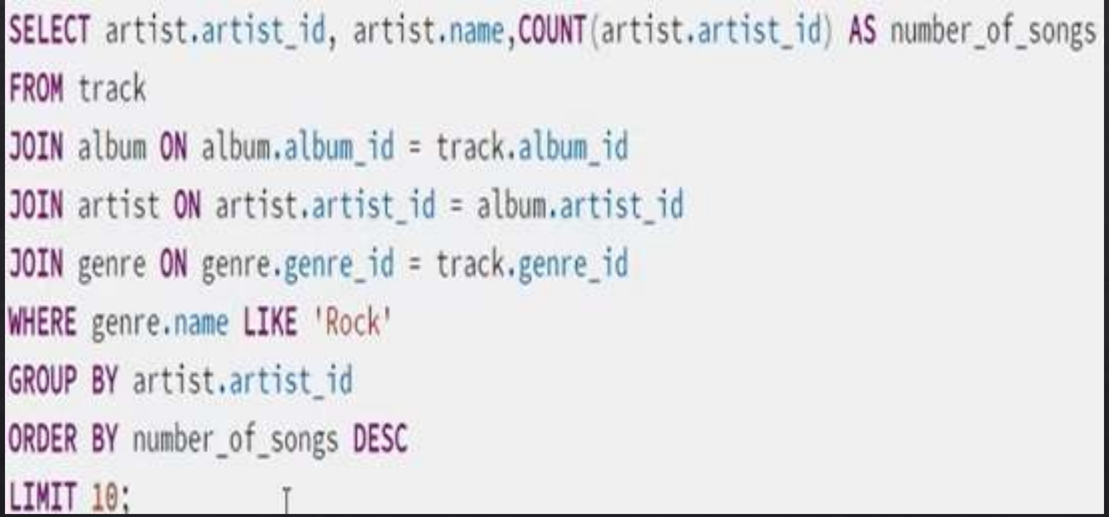


```sql
SELECT DISTINCT email,first_name, last_name
FROM customer
JOIN invoice ON customer.customer_id = invoice.customer_id
JOIN invoice_line ON invoice.invoice_id = invoice_line.invoice_id
WHERE track_id IN(
    SELECT track_id FROM track
    JOIN genre ON track.genre_id = genre.genre_id
    WHERE genre.name LIKE 'Rock'
)
ORDER BY email;
```

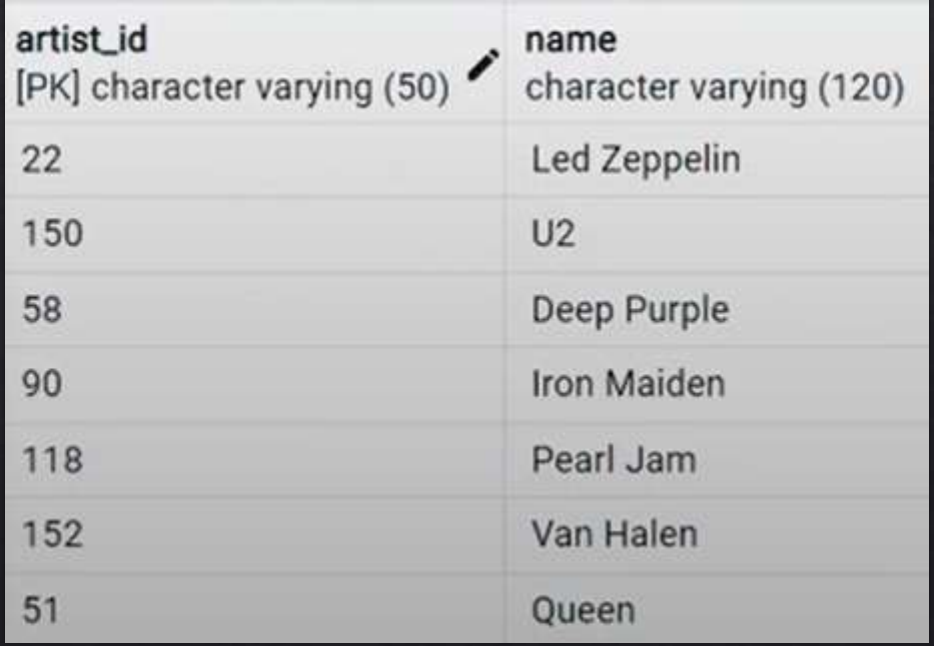


email character varying (50)	first_name character (50)	last_name character (50)
aaronmitchell@yahoo...	Aaron	Mitchell
alero@uol.com.br	Alexandre	Rocha
astrid.gruber@apple.at	Astrid	Gruber
bjorn.hansen@yahoo....	Bjørn	Hansen
camille.bernard@yah...	Camille	Bernard

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.

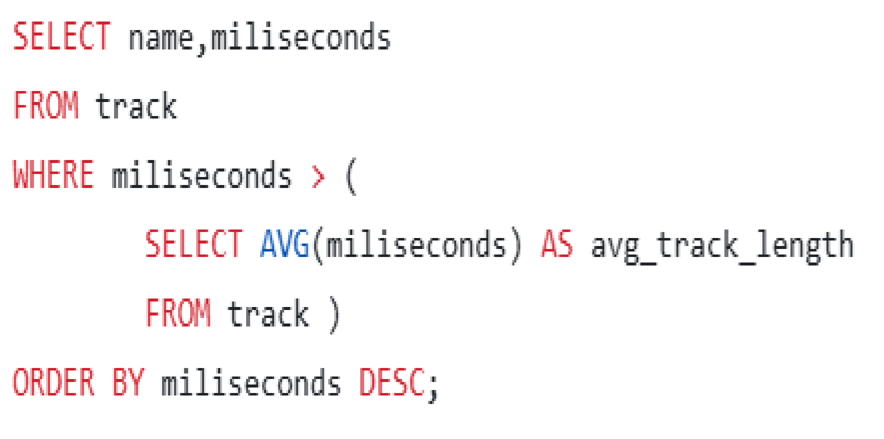


```
SELECT artist.artist_id, artist.name,COUNT(artist.artist_id) AS number_of_songs
FROM track
JOIN album ON album.album_id = track.album_id
JOIN artist ON artist.artist_id = album.artist_id
JOIN genre ON genre.genre_id = track.genre_id
WHERE genre.name LIKE 'Rock'
GROUP BY artist.artist_id
ORDER BY number_of_songs DESC
LIMIT 10;
```

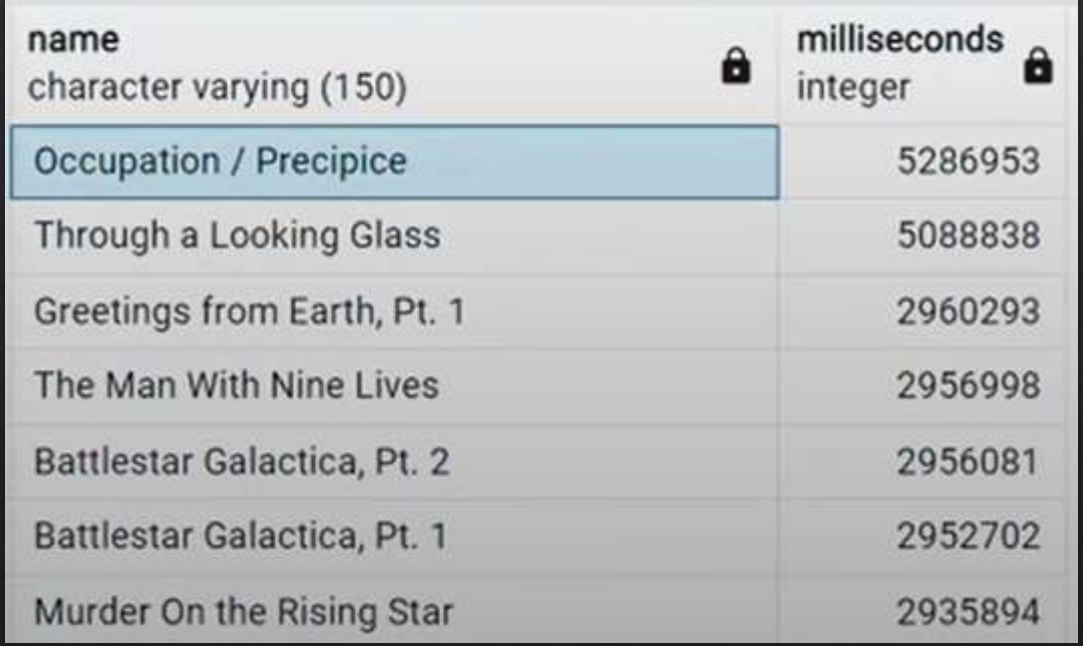


artist_id	name
[PK] character varying (50)	character varying (120)
22	Led Zeppelin
150	U2
58	Deep Purple
90	Iron Maiden
118	Pearl Jam
152	Van Halen
51	Queen

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.

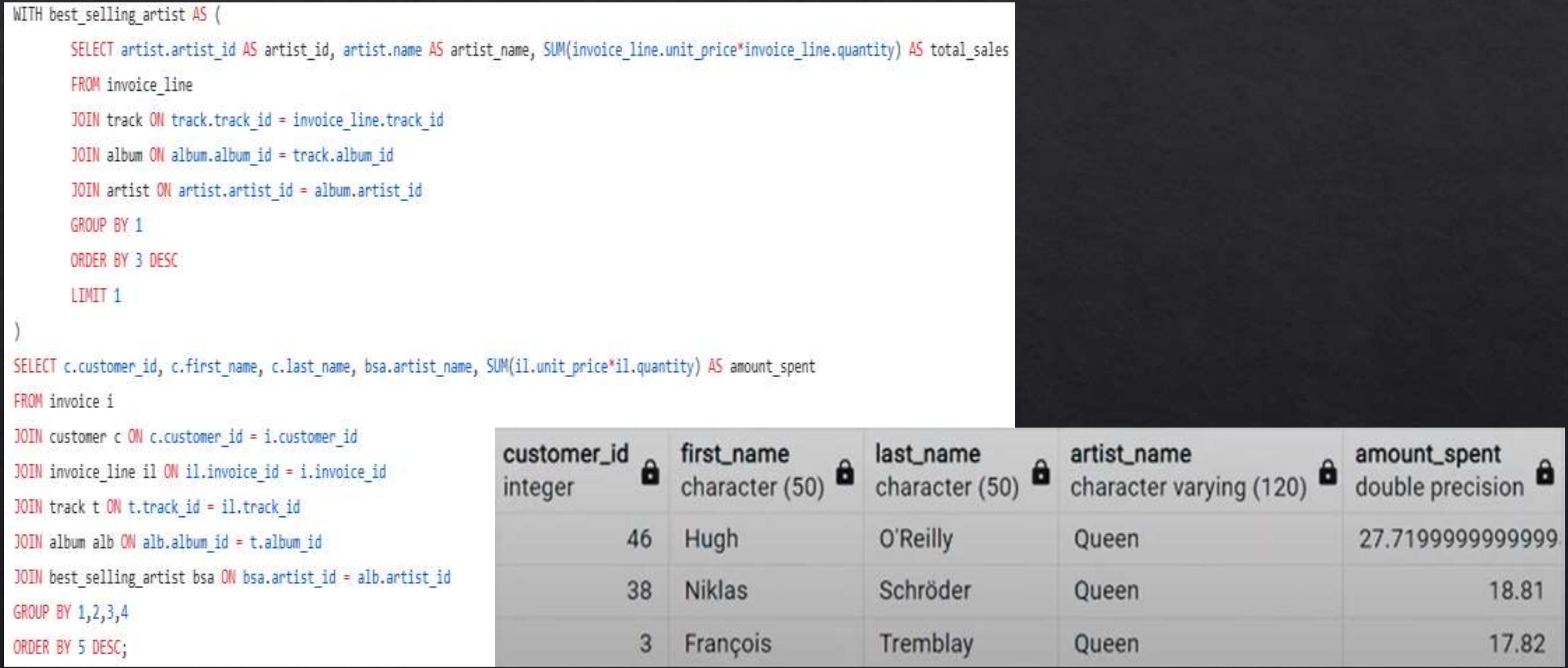


```
SELECT name,miliseconds
FROM track
WHERE miliseconds > (
        SELECT AVG(miliseconds) AS avg_track_length
        FROM track )
ORDER BY miliseconds DESC;
```



name character varying (150)	milliseconds integer
Occupation / Precipice	5286953
Through a Looking Glass	5088838
Greetings from Earth, Pt. 1	2960293
The Man With Nine Lives	2956998
Battlestar Galactica, Pt. 2	2956081
Battlestar Galactica, Pt. 1	2952702
Murder On the Rising Star	2935894

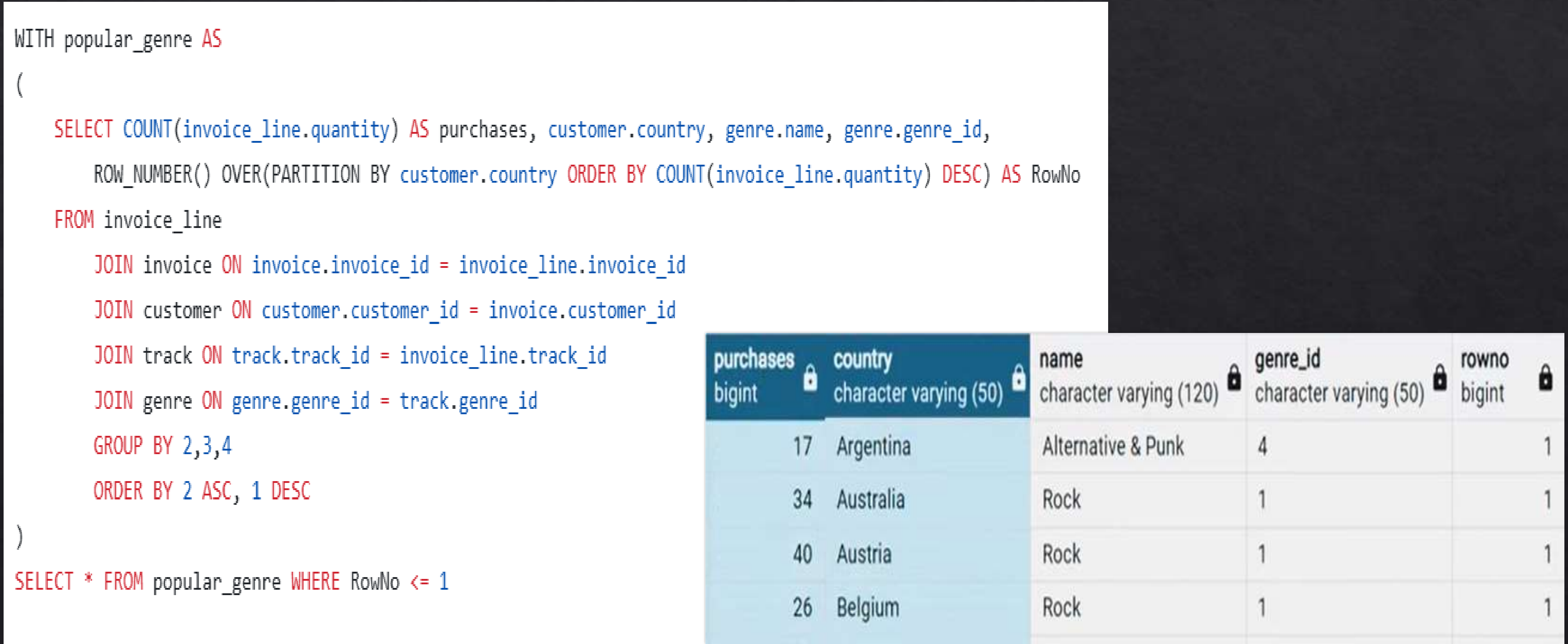
Q9: Find how much amount spent by each customer on artists. Write a query to return the customer name, artist name, and total spent



```
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
	JOIN track ON track.track_id = invoice_line.track_id
	JOIN album ON album.album_id = track.album_id
	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_spent
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 alb ON alb.album_id = t.album_id
JOIN best_selling_artist bsa ON bsa.artist_id = alb.artist_id
GROUP BY 1,2,3,4
ORDER BY 5 DESC;
```

customer_id integer	first_name character (50)	last_name character (50)	artist_name character varying (120)	amount_spent double precision
46	Hugh	O'Reilly	Queen	27.7199999999999
38	Niklas	Schröder	Queen	18.81
3	François	Tremblay	Queen	17.82

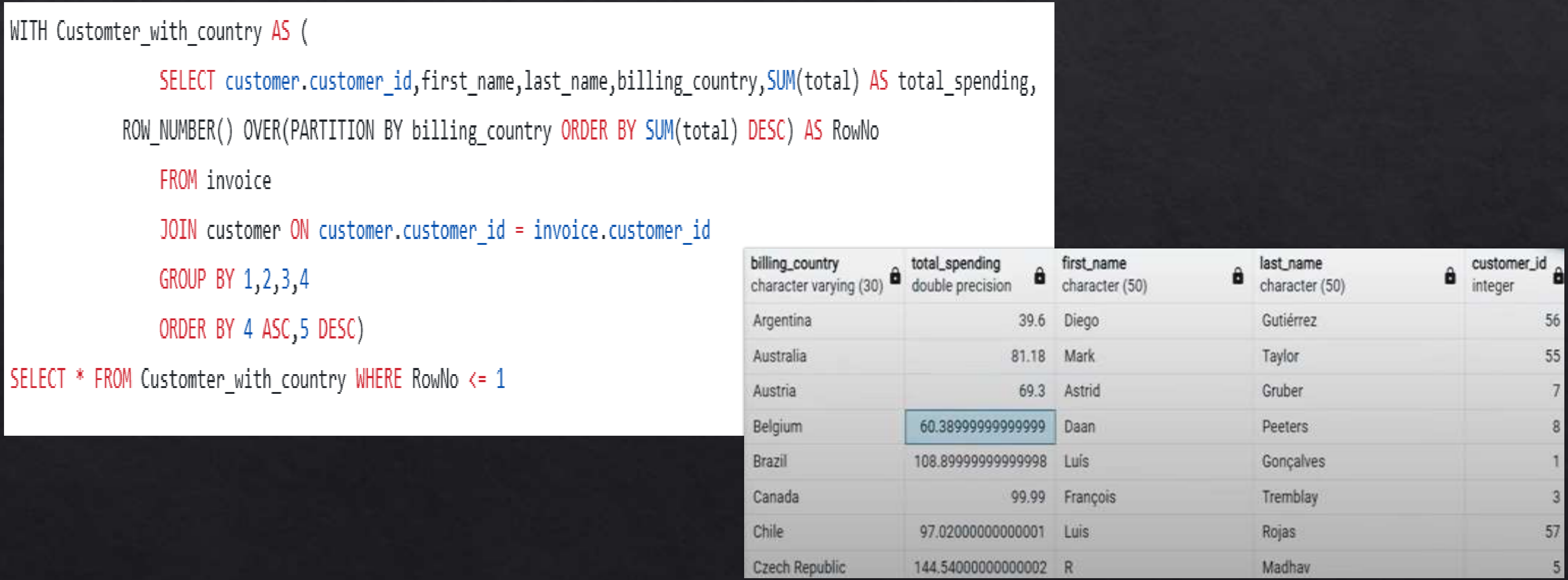
Q10: We want to find out the most popular music Genre for each country. We determine the most popular genre as the genre with the highest amount of purchases. Write a query that returns each country along with the top Genre. For countries where the maximum number of purchases is shared return all Genres.



```
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
        JOIN invoice ON invoice.invoice_id = invoice_line.invoice_id
        JOIN customer ON customer.customer_id = invoice.customer_id
        JOIN track ON track.track_id = invoice_line.track_id
        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 bigint	country character varying (50)	name character varying (120)	genre_id character varying (50)	rowno bigint
17	Argentina	Alternative & Punk	4	1
34	Australia	Rock	1	1
40	Austria	Rock	1	1
26	Belgium	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 Customter_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

            JOIN customer ON customer.customer_id = invoice.customer_id

            GROUP BY 1,2,3,4

            ORDER BY 4 ASC,5 DESC)

SELECT * FROM Customter_with_country WHERE RowNo <= 1
```

billing_country character varying (30)	total_spending double precision	first_name character (50)	last_name character (50)	customer_id integer
Argentina	39.6	Diego	Gutiérrez	56
Australia	81.18	Mark	Taylor	55
Austria	69.3	Astrid	Gruber	7
Belgium	60.38999999999999	Daan	Peeters	8
Brazil	108.89999999999998	Luís	Gonçalves	1
Canada	99.99	François	Tremblay	3
Chile	97.02000000000001	Luis	Rojas	57
Czech Republic	144.54000000000002	R	Madhav	5

THANK YOU