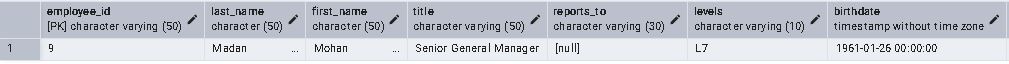
-- 1) Who is the senior most employee based on job title?

SELECT \* FROM employee

ORDER BY levels DESC

LIMIT 1;



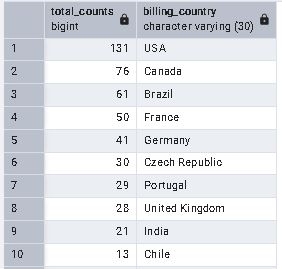
-- 2) Which countries have the most Invoices?

SELECT COUNT(\*) total\_counts, billing\_country

FROM invoice

GROUP BY billing\_country

ORDER BY total\_counts desc;

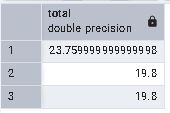


-- 3) What are top 3 values of total invoice?

SELECT total FROM invoice

ORDER BY total DESC

LIMIT 3;



-- 4) 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.

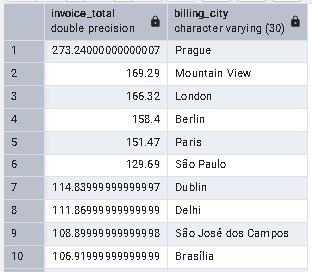
-- Return both the city name & sum of all invoice totals

SELECT SUM(total) AS invoice\_total, billing\_city

FROM invoice

GROUP BY billing\_city

ORDER BY invoice\_total DESC;



-- 5) 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

SELECT c.customer\_id, c.first\_name, c.last\_name, SUM(i.total) AS Total

FROM customer c

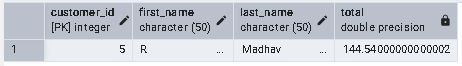
JOIN invoice i

ON c.customer\_id = i.customer\_id

GROUP BY c.customer\_id

ORDER By Total DESC

LIMIT 1;



-- 6) 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

SELECT DISTINCT email,first\_name, last\_name

FROM customer c

JOIN invoice i ON c.customer\_id = i.customer\_id

JOIN invoice\_line il ON i.invoice\_id = il.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;



-- 7) 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 ar.artist\_id, ar.name,COUNT(ar.artist\_id) AS number\_of\_song

FROM track t

JOIN album a ON a.album\_id = t.album\_id

JOIN artist ar ON ar.artist\_id = a.artist\_id

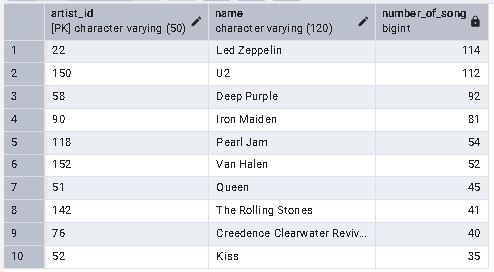
JOIN genre g ON g.genre\_id = t.genre\_id

WHERE g.name LIKE 'Rock'

GROUP By ar.artist\_id

ORDER By number\_of\_song DESC

LIMIT 10;



-- 8) 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, milliseconds

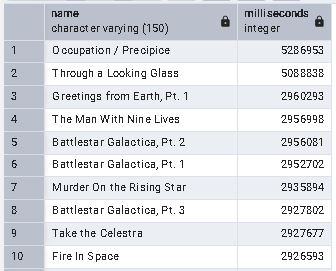
FROM track

WHERE milliseconds > (

SELECT AVG (milliseconds) AS avg\_track\_length

FROM track)

ORDER BY milliseconds DESC;



-- 9) Find how much amount spent by each customer on artists?

-- Write a query to return customer name, artist name and total spent

WITH best\_selling\_artist AS(

SELECT ar.artist\_id AS artist\_id, ar.name AS artist\_name,

SUM(il.unit\_price\*il.quantity) AS total\_sale

FROM invoice\_line il

JOIN track t ON t.track\_id = il.track\_id

JOIN album a ON a.album\_id = t.album\_id

JOIN artist ar ON a.artist\_id = a.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

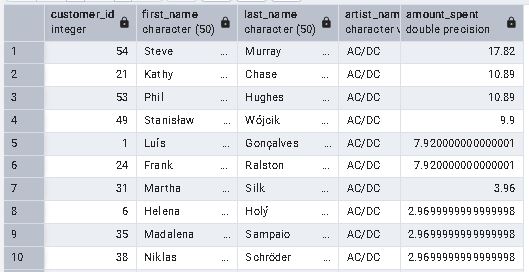
JOIN track t ON t.track\_id = il.track\_id

JOIN album a ON a.album\_id = t.album\_id

JOIN best\_selling\_artist bsa ON bsa.artist\_id = a.artist\_id

GROUP BY 1,2,3,4

ORDER BY 5 DESC;



-- 10) 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(il.quantity) AS purchase, c.country, g.name, g.genre\_id,

ROW\_NUMBER() OVER(PARTITION BY c.country ORDER BY COUNT(il.quantity) DESC) AS row\_no

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\_no <= 1;



-- 11) 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 RECURSIVE

customter\_with\_country AS (

SELECT c.customer\_id,first\_name,last\_name,billing\_country,SUM(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 billing\_country,MAX(total\_spending) AS max\_spending

FROM customter\_with\_country

GROUP BY billing\_country)

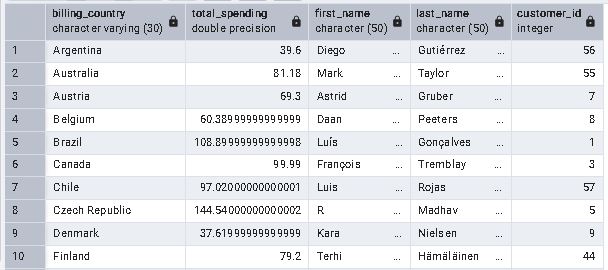
SELECT cwc.billing\_country, cwc.total\_spending, cwc.first\_name, cwc.last\_name, cwc.customer\_id

FROM customter\_with\_country cwc

JOIN country\_max\_spending cms ON cwc.billing\_country = cms.billing\_country

WHERE cwc.total\_spending = cms.max\_spending

ORDER BY 1;



-- Method 2

WITH customter\_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\_no

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 customter\_with\_country WHERE row\_no <= 1;

