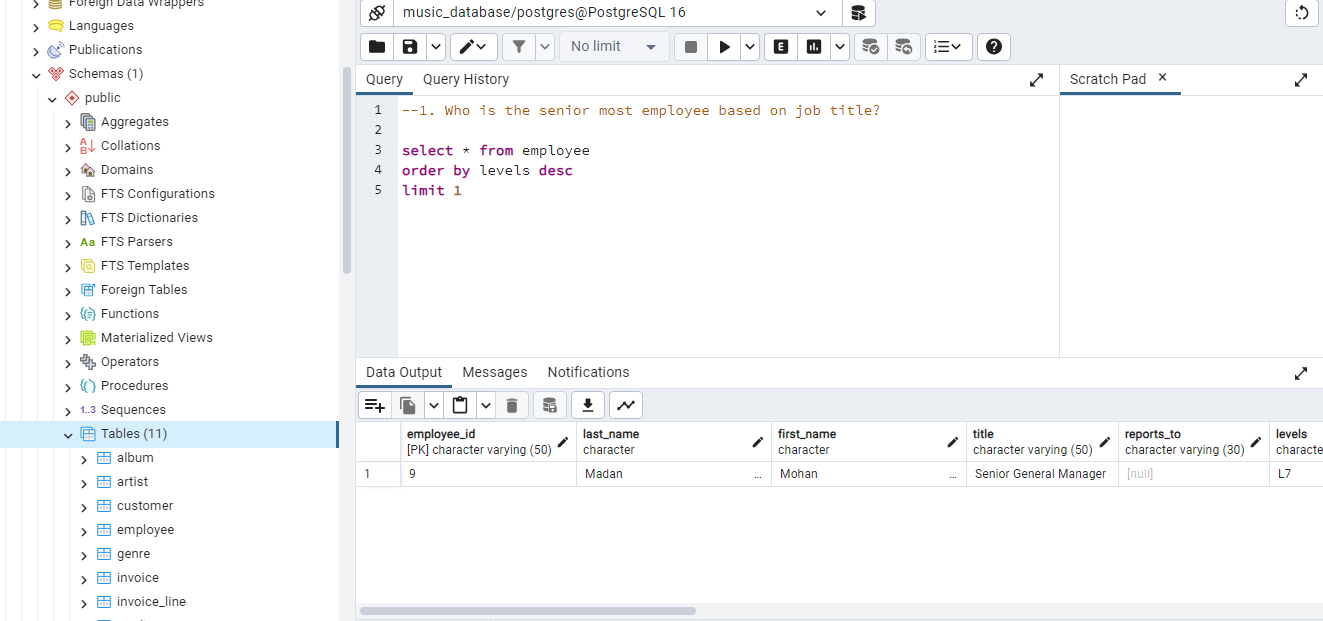
**Question Set 1 – Easy**

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

select \* from employee

order by levels desc

limit 1

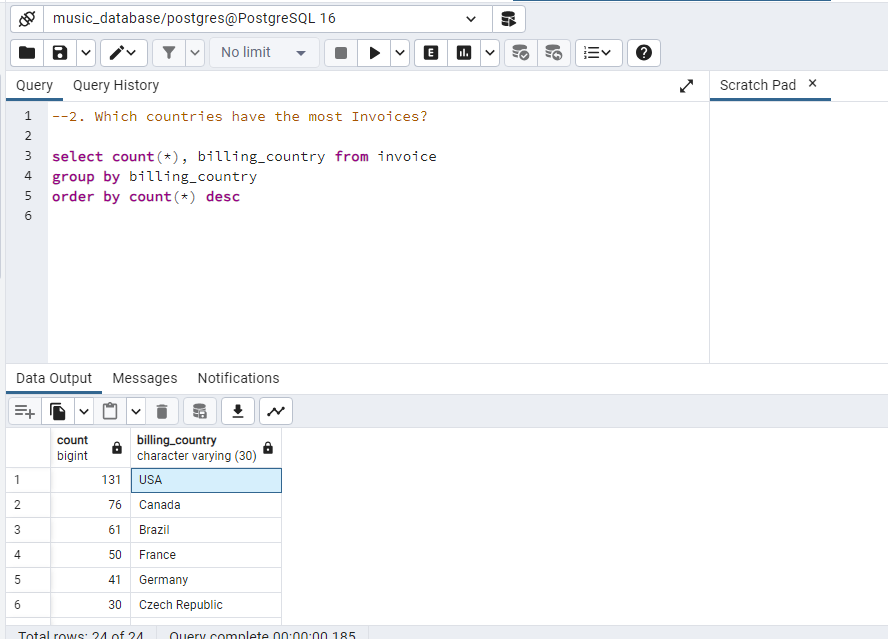


1. Which countries have the most Invoices?

select count(\*), billing\_country from invoice

group by billing\_country

order by count(\*) desc

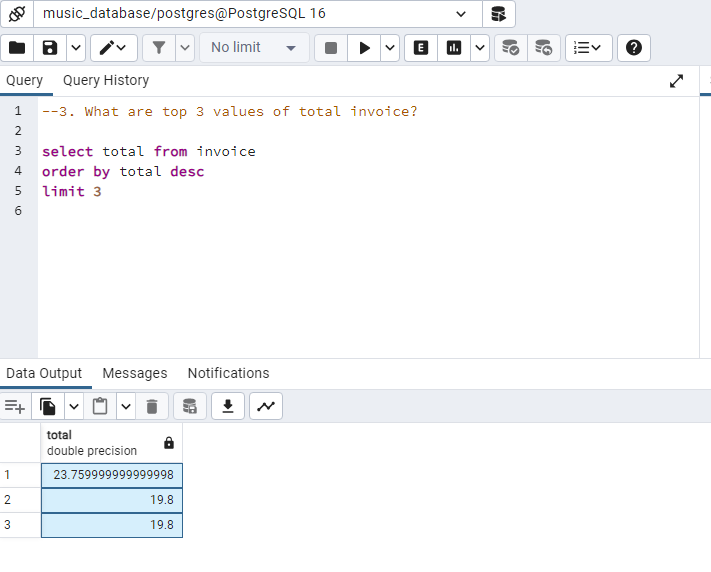


1. What are top 3 values of total invoice?

select total from invoice

order by total desc

limit 3



1. 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.

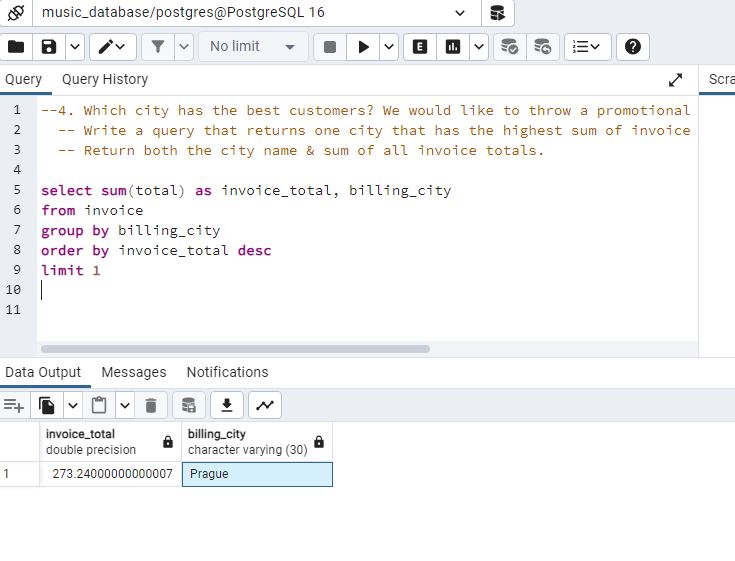
select sum(total) as invoice\_total, billing\_city

from invoice

group by billing\_city

order by invoice\_total desc

limit 1



1. 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 customer.customer\_id, customer.first\_name, customer.last\_name, sum(invoice.total) as total

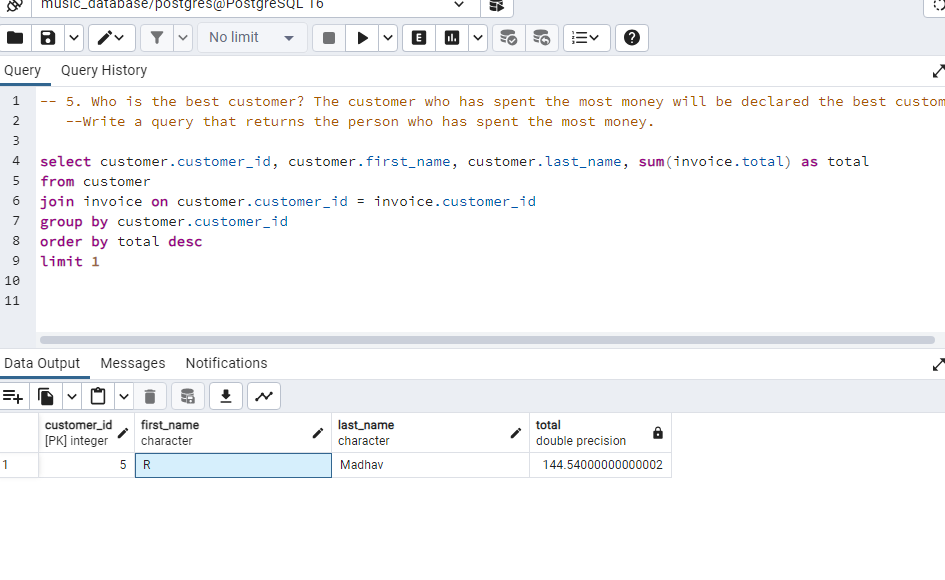
from customer

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

group by customer.customer\_id

order by total desc

limit 1



**Question Set 2 – Moderate**

1. 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 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(

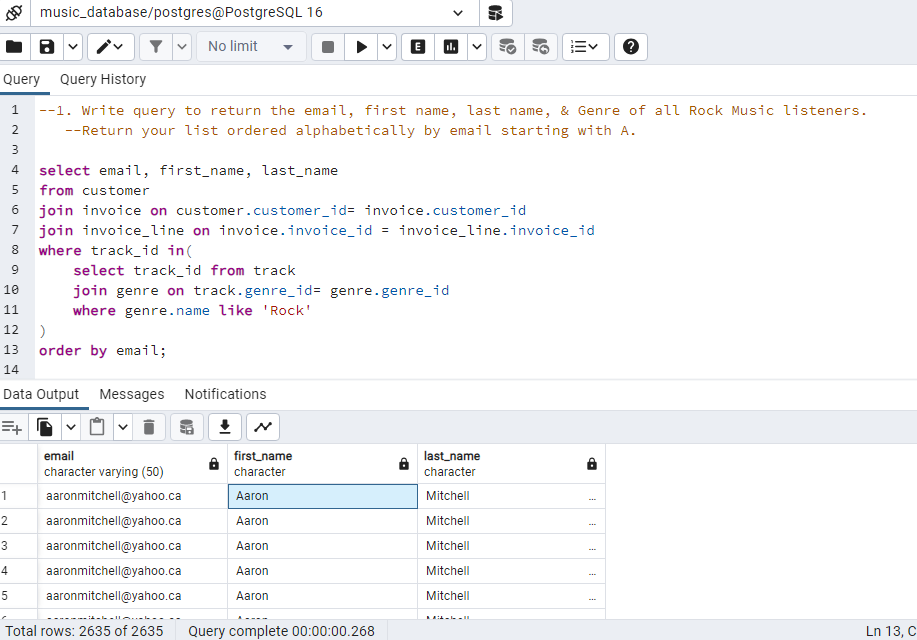
select track\_id from track

join genre on track.genre\_id= genre.genre\_id

where genre.name like 'Rock'

)

order by email;



1. 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

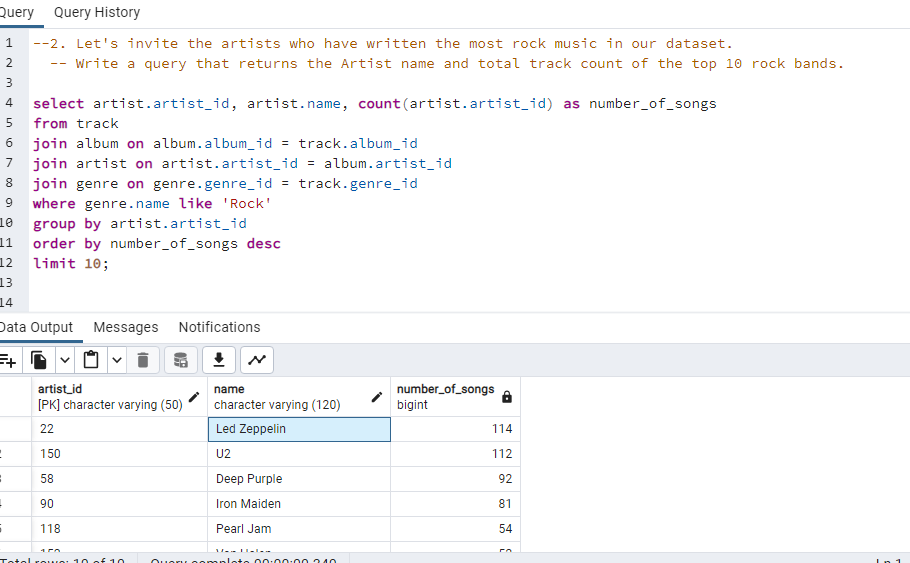
join genre on genre.genre\_id = track.genre\_id

where genre.name like 'Rock'

group by artist.artist\_id

order by number\_of\_songs desc

limit 10;



1. 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

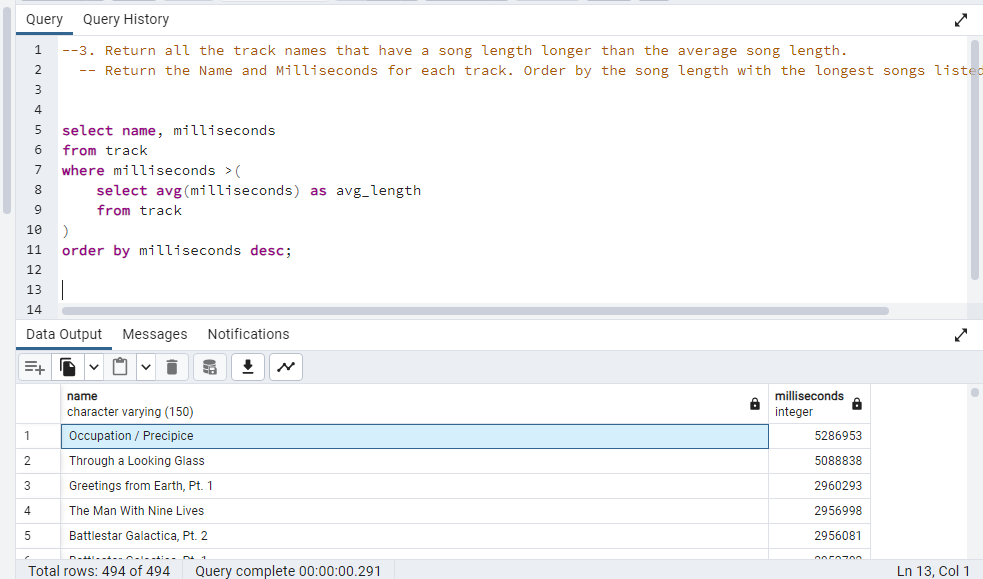
where milliseconds >(

select avg(milliseconds) as avg\_length

from track

)

order by milliseconds desc;



**Question Set 3 – Advance**

1. 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 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 customer.customer\_id, customer.first\_name, customer.last\_name, best\_selling\_artist.artist\_name,

sum(invoice\_line.unit\_price\*invoice\_line.quantity) AS amount\_spent

from invoice

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

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

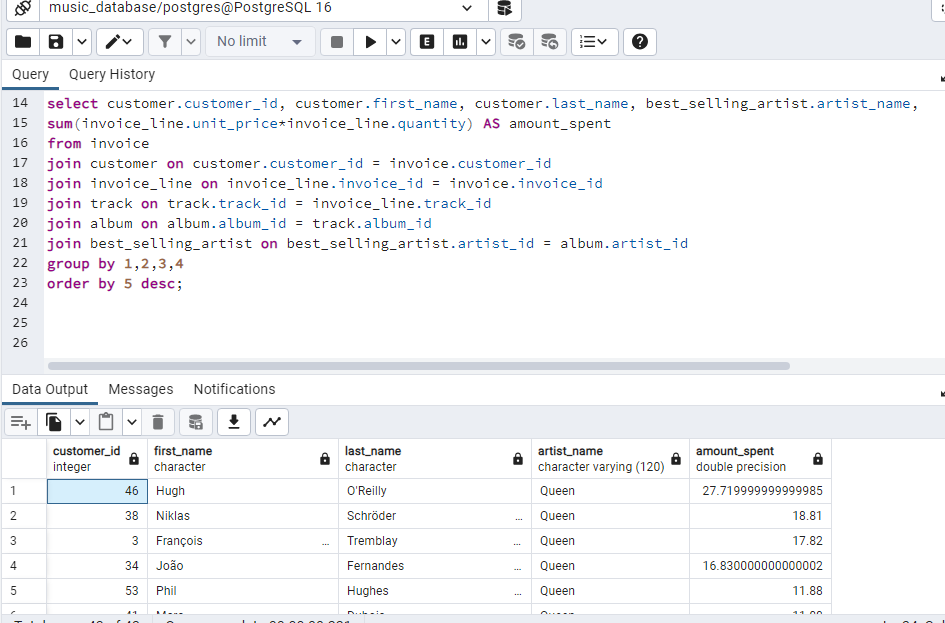
join track on track.track\_id = invoice\_line.track\_id

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

join best\_selling\_artist on best\_selling\_artist.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 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

