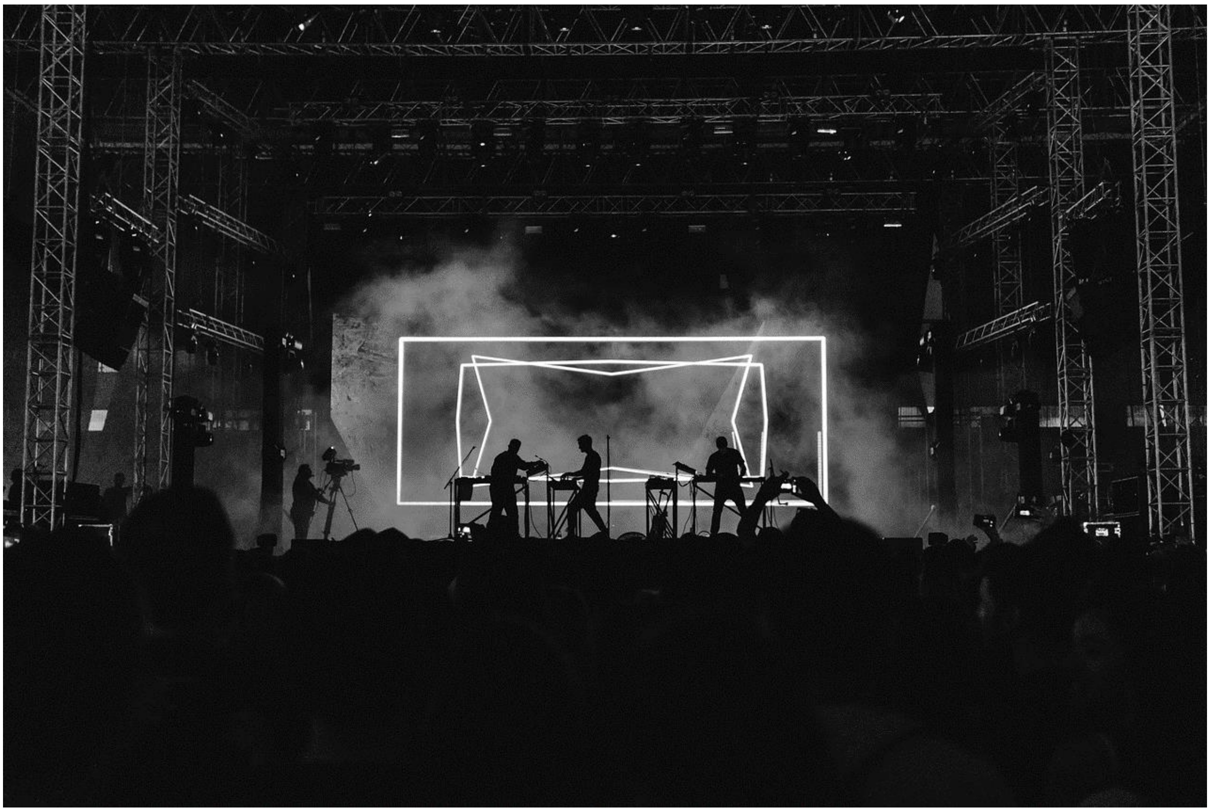
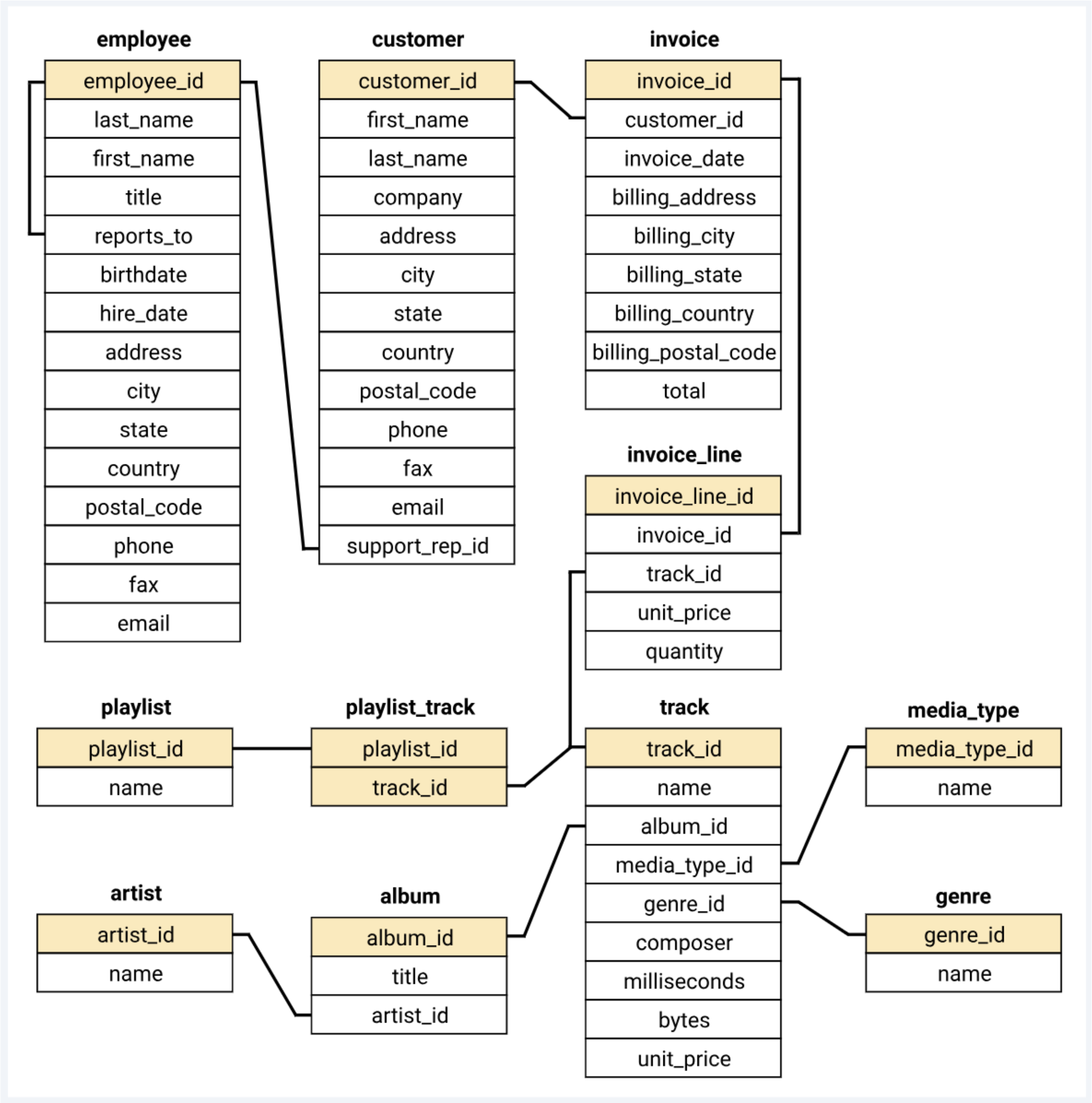
# SQL PROJECT: MUSIC STORE DATA ANALYSIS

## Problem Statement

SQL-driven exploration of a music store database, delving into customer habits, sales patterns, and inventory insights for optimization.



# SCHEMA DIAGRAM:-



# 1ST QUESTION

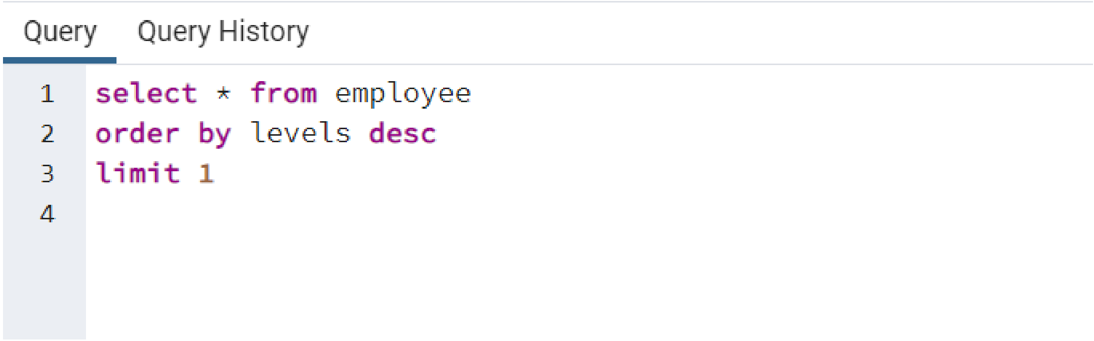
### WHO IS THE SENIOR MOST EMPLOYEE  BASED ON THE JOB TITLE?

**QUERY:-**

select * from employee

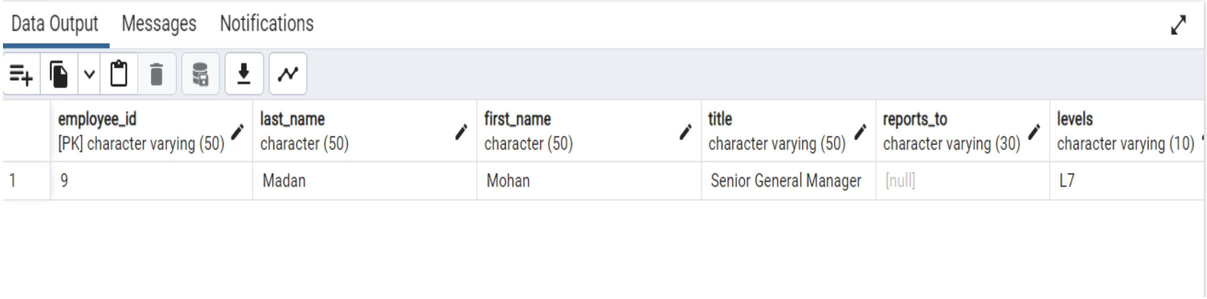
order by levels desc

limit 1



```
select * from employee
order by levels desc
limit 1
```

**OUTPUT:-**



| | employee_id<br>[PK] character varying (50) | last_name<br>character (50) | first_name<br>character (50) | title<br>character varying (50) | reports_to<br>character varying (30) | levels<br>character varying (10) |
|---|---|---|---|---|---|---|
| 1 | 9 | Madan | Mohan | Senior General Manager | [null] | L7 |

# 2ND QUESTION

## WHICH COUNTRIES HAVE THE MOST INVOICES?

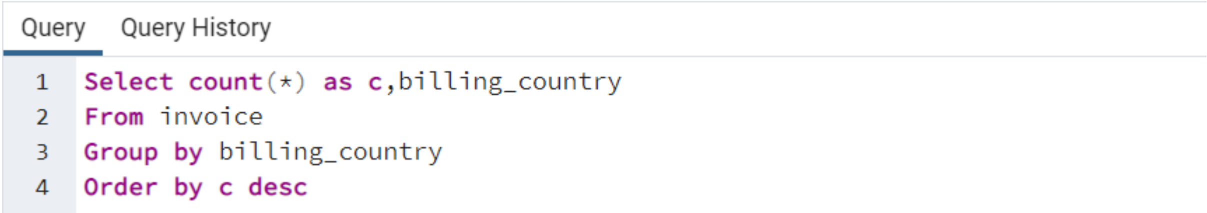
### QUERY:-

Select count(*) as c,billing_country

From invoice

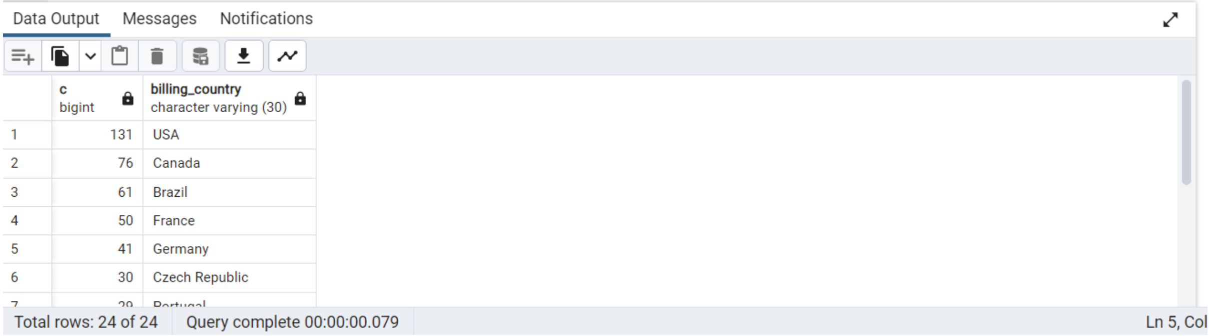
Group by billing_country

Order by c desc



```
Select count(*) as c,billing_country
From invoice
Group by billing_country
Order by c desc
```

### OUTPUT:-



| | c<br>bigint | billing_country<br>character varying (30) |
|---|---|---|
| 1 | 131 | USA |
| 2 | 76 | Canada |
| 3 | 61 | Brazil |
| 4 | 50 | France |
| 5 | 41 | Germany |
| 6 | 30 | Czech Republic |
| 7 | 29 | Portugal |

Total rows: 24 of 24 Query complete 00:00:00.079 Ln 5, Col

# 3$^{RD}$ QUESTION

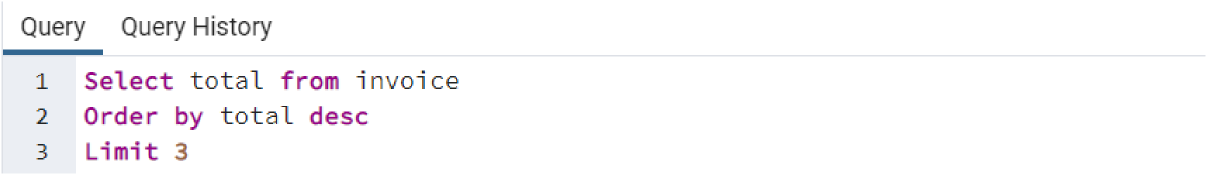
## WHAT ARE THE TOP 3 VALUES OF TOTAL INVOICE?

### QUERY:-

Select total from invoice

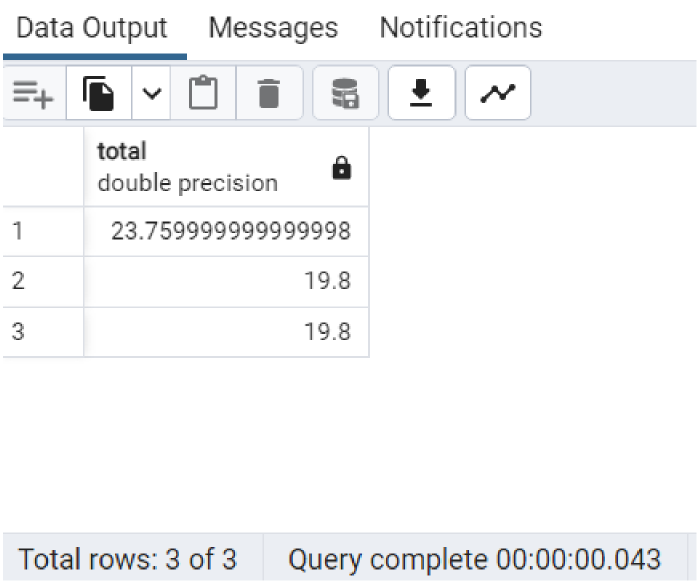
Order by total desc

Limit 3



```
Select total from invoice
Order by total desc
Limit 3
```

### OUTPUT:-



| | total<br>double precision |
|---|---|
| 1 | 23.759999999999998 |
| 2 | 19.8 |
| 3 | 19.8 |

Total rows: 3 of 3 Query complete 00:00:00.043

# 4TH QUESTION

**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. BOTH THE CITY NAME & SUM OF ALL INVOICE TOTALS.**

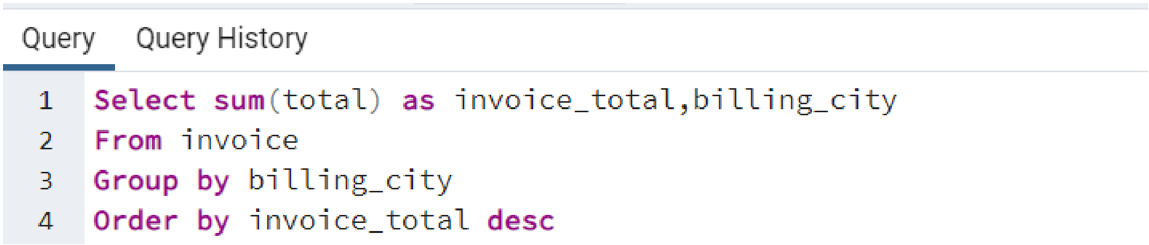
**QUERY:-**

Select sum(total) as invoice_total,billing_city

From invoice

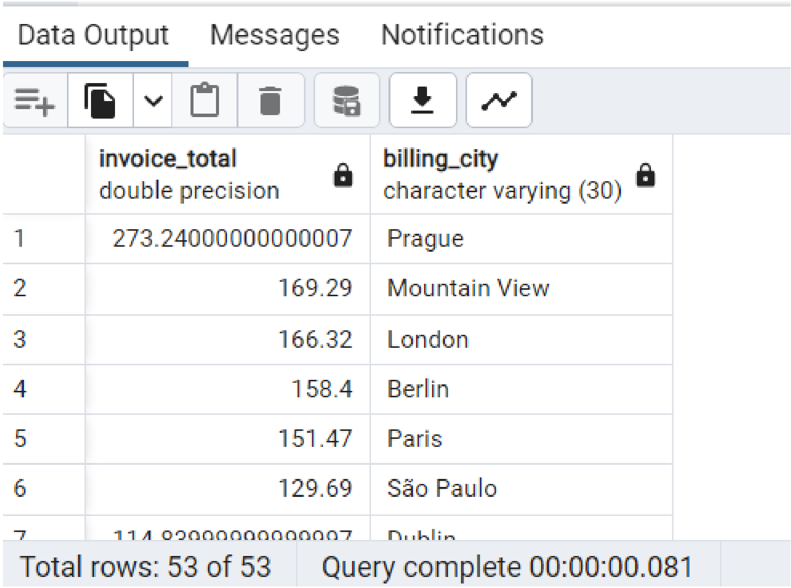
Group by billing_city

Order by invoice_total desc



```
Select sum(total) as invoice_total,billing_city
From invoice
Group by billing_city
Order by invoice_total desc
```

**OUTPUT:-**



| | invoice_total<br>double precision | billing_city<br>character varying (30) |
|---|---|---|
| 1 | 273.24000000000007 | Prague |
| 2 | 169.29 | Mountain View |
| 3 | 166.32 | London |
| 4 | 158.4 | Berlin |
| 5 | 151.47 | Paris |
| 6 | 129.69 | São Paulo |
| 7 | 114.83999999999997 | Dublin |

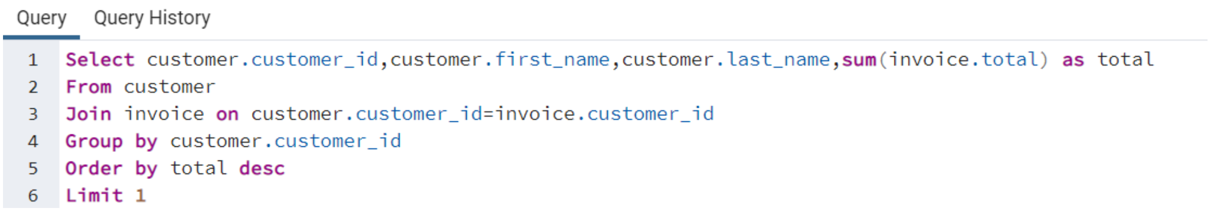
Total rows: 53 of 53 Query complete 00:00:00.081

# 5$^{TH}$ QUESTION

**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.**

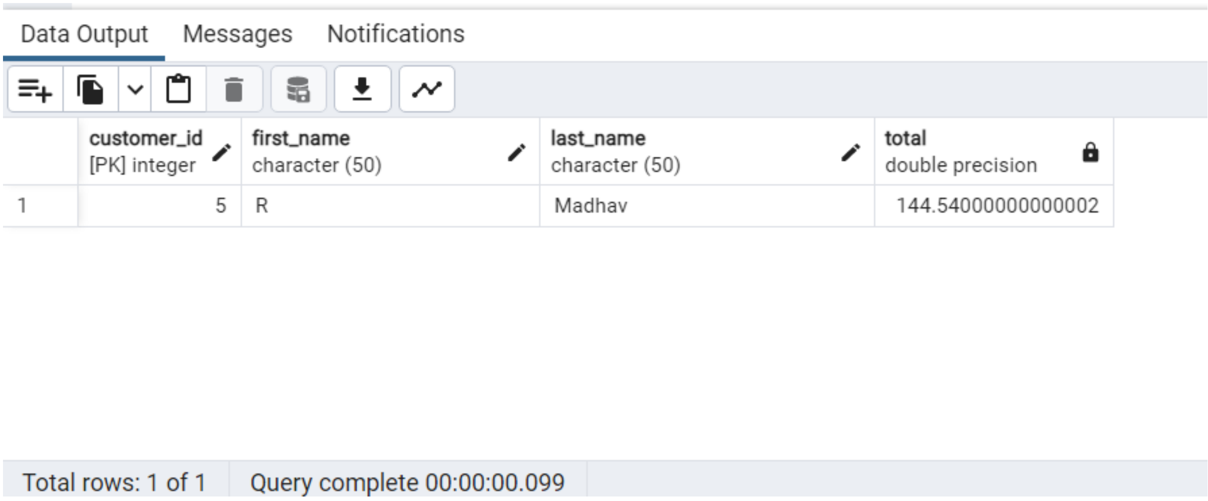
**QUERY:-**

```
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
```



```
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
```

**OUTPUT:-**



| | customer_id<br>[PK] integer | first_name<br>character (50) | last_name<br>character (50) | total<br>double precision |
|---|---|---|---|---|
| 1 | 5 | R | Madhav | 144.54000000000002 |

Total rows: 1 of 1 Query complete 00:00:00.099

# 6$^{TH}$ QUESTION

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

**QUERY:-**

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(

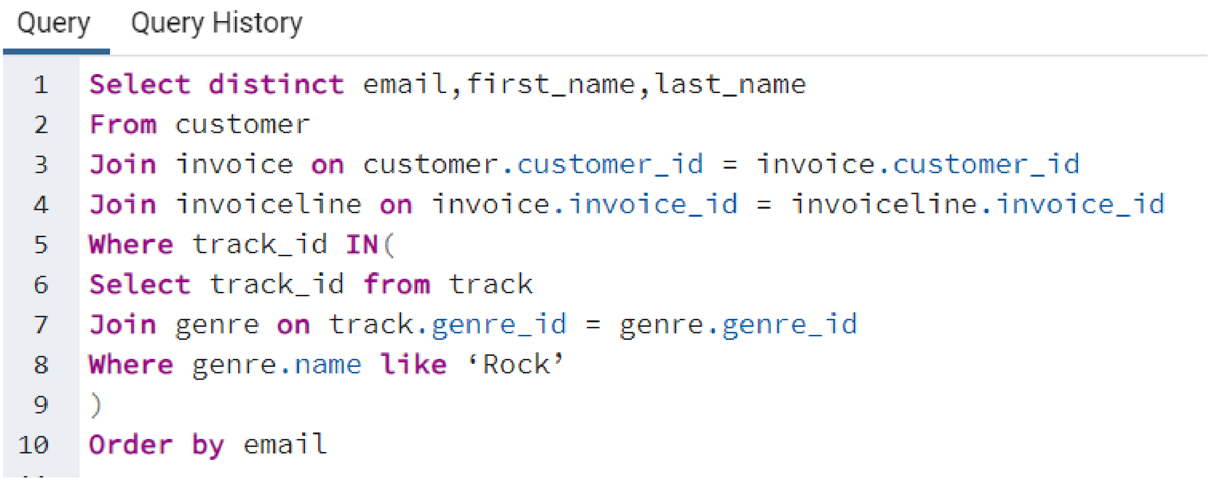
Select track_id from track

Join genre on track.genre_id = genre.genre_id

Where genre.name like 'Rock'

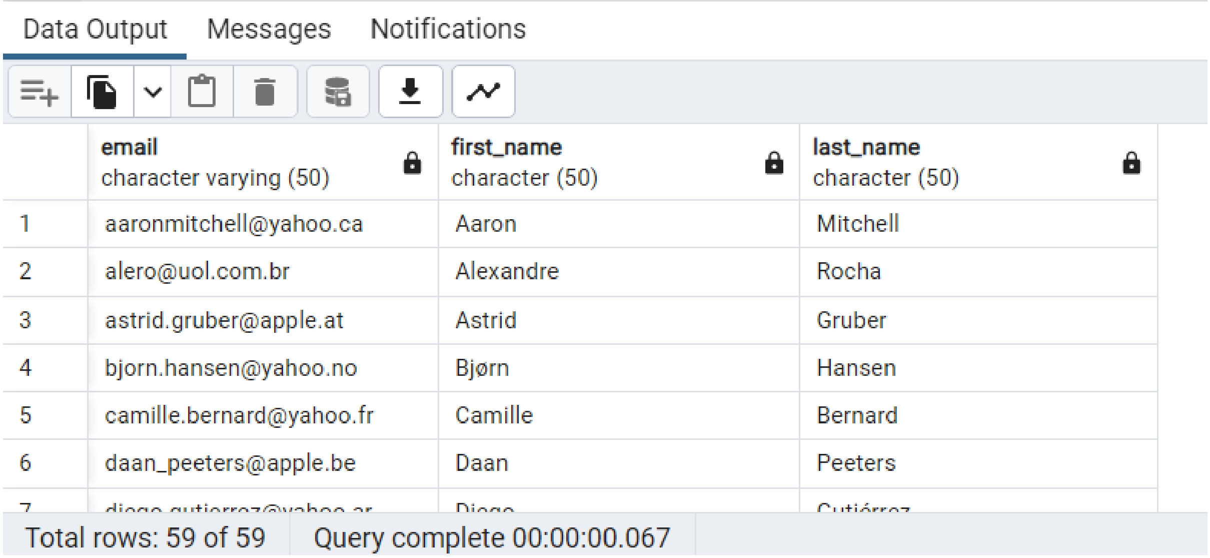
)

Order by email



```
Select distinct email,first_name,last_name
From customer
Join invoice on customer.customer_id = invoice.customer_id
Join invoiceline on invoice.invoice_id = invoiceline.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
```

**OUTPUT:-**



| | email<br>character varying (50) | first_name<br>character (50) | last_name<br>character (50) |
|---|---|---|---|
| 1 | aaronmitchell@yahoo.ca | Aaron | Mitchell |
| 2 | alero@uol.com.br | Alexandre | Rocha |
| 3 | astrid.gruber@apple.at | Astrid | Gruber |
| 4 | bjorn.hansen@yahoo.no | Bjørn | Hansen |
| 5 | camille.bernard@yahoo.fr | Camille | Bernard |
| 6 | daan_peeters@apple.be | Daan | Peeters |
| 7 | diego.gutierrez@yahoo.ar | Diego | Gutiérrez |

Total rows: 59 of 59 Query complete 00:00:00.067

# 7TH QUESTION

**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 ARTISTS.**

**QUERY:-**

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

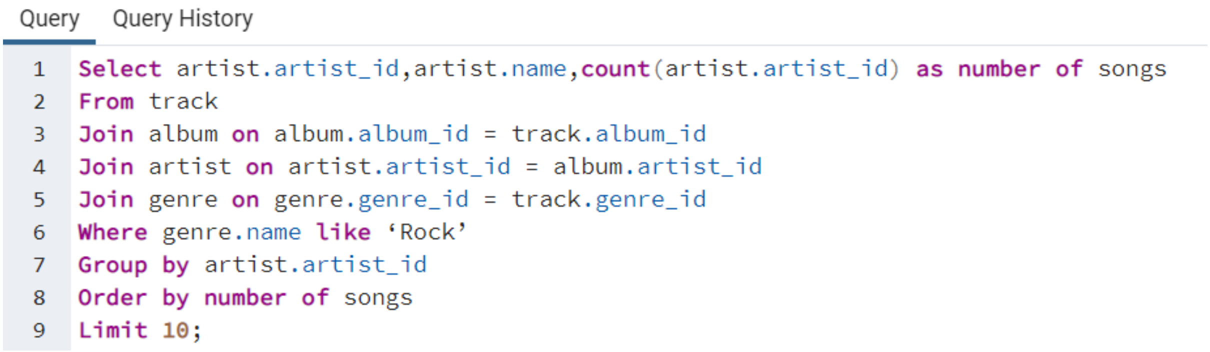
Join genre on genre.genre_id = track.genre_id

Where genre.name like ‘Rock’

Group by artist.artist_id

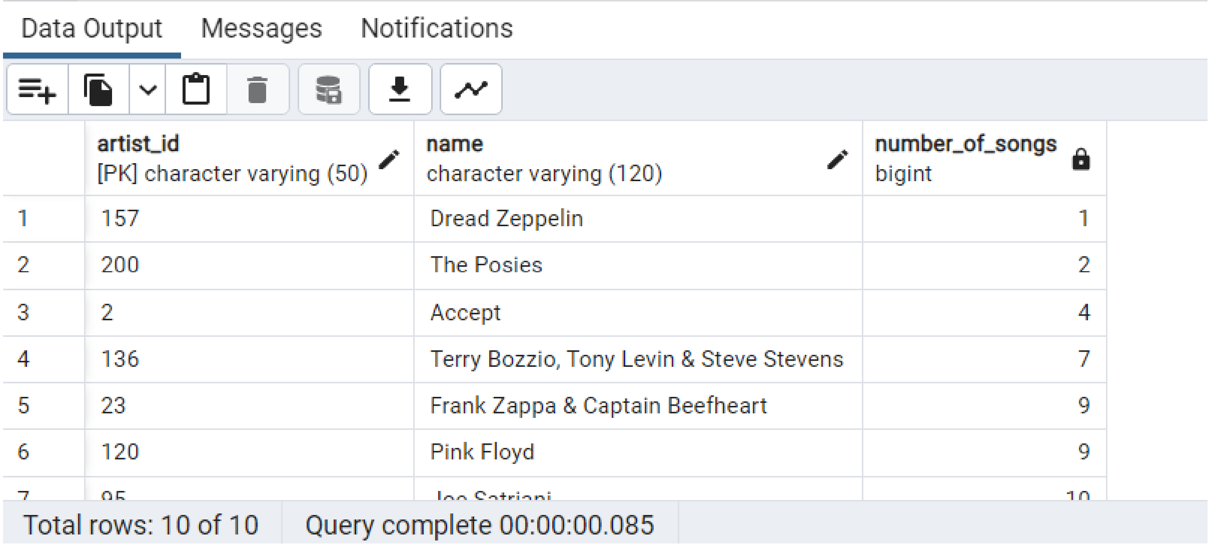
Order by number of songs

Limit 10;



```
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
Limit 10;
```

**OUTPUT:-**



| | artist_id<br>[PK] character varying (50) | name<br>character varying (120) | number_of_songs<br>bigint |
|---|---|---|---|
| 1 | 157 | Dread Zeppelin | 1 |
| 2 | 200 | The Posies | 2 |
| 3 | 2 | Accept | 4 |
| 4 | 136 | Terry Bozzio, Tony Levin & Steve Stevens | 7 |
| 5 | 23 | Frank Zappa & Captain Beefheart | 9 |
| 6 | 120 | Pink Floyd | 9 |
| 7 | 95 | Joe Satriani | 10 |

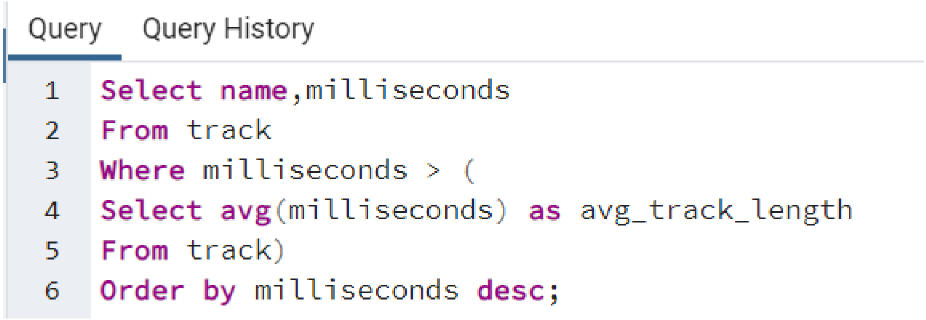
Total rows: 10 of 10 Query complete 00:00:00.085

# 8TH QUESTION

**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.**

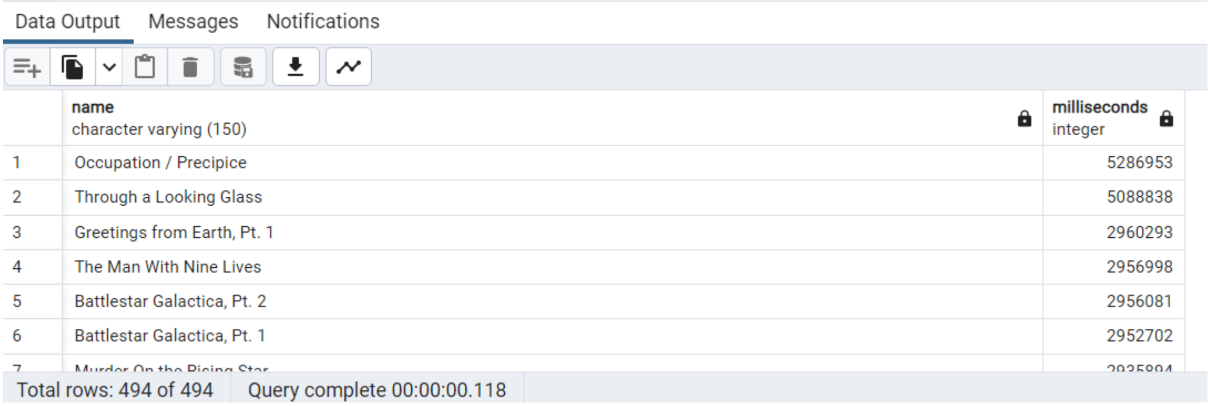
**QUERY:-**

```
Select name,milliseconds

From track

Where milliseconds > (

Select avg(milliseconds) as avg_track_length

From track)

Order by milliseconds desc;
```



```
Select name,milliseconds
From track
Where milliseconds > (
Select avg(milliseconds) as avg_track_length
From track)
Order by milliseconds desc;
```

**OUTPUT:-**



| | name<br>character varying (150) | milliseconds<br>integer |
|---|---|---|
| 1 | Occupation / Precipice | 5286953 |
| 2 | Through a Looking Glass | 5088838 |
| 3 | Greetings from Earth, Pt. 1 | 2960293 |
| 4 | The Man With Nine Lives | 2956998 |
| 5 | Battlestar Galactica, Pt. 2 | 2956081 |
| 6 | Battlestar Galactica, Pt. 1 | 2952702 |
| 7 | Murder On the Rising Star | 2935894 |

Total rows: 494 of 494 Query complete 00:00:00.118

# 9$^{TH}$ QUESTION

**FIND HOW MUCH AMOUNT SPENT BY EACH CUSTOMER ON ARTISTS? WRITE A QUERY TO RETURN CUSTOMER NAME, ARTIST NAME AND TOTAL SPENT.**

**QUERY:-**

select c.customer_id,c.first_name,c.last_name,artist.name,sum(invoice_line .unit_price*

invoice_line.quantity) as s

from customer c

join invoice on c.customer_id = invoice.customer_id

join invoice_line on invoice.invoice_id = invoice_line.invoice_id

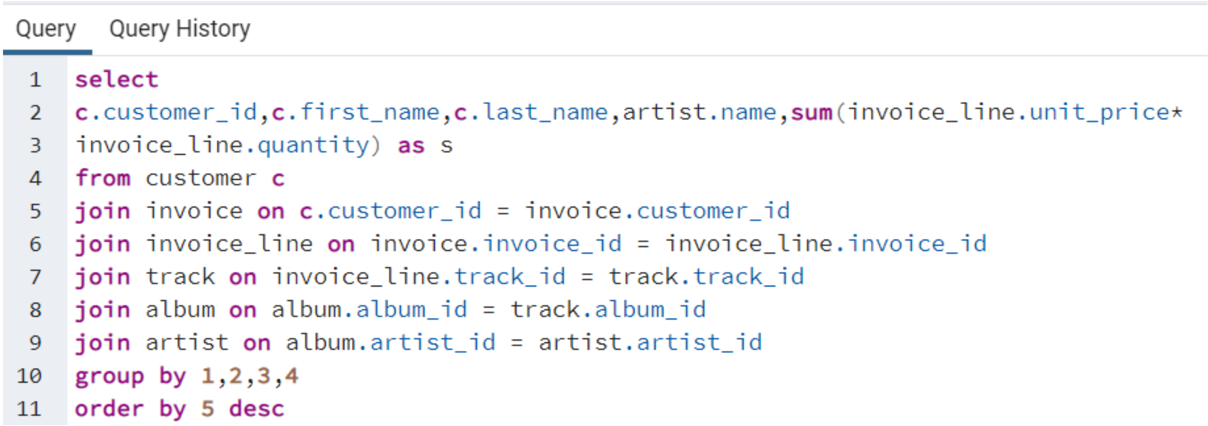
join track on invoice_line.track_id = track.track_id

join album on album.album_id = track.album_id

join artist on album.artist_id = artist.artist_id

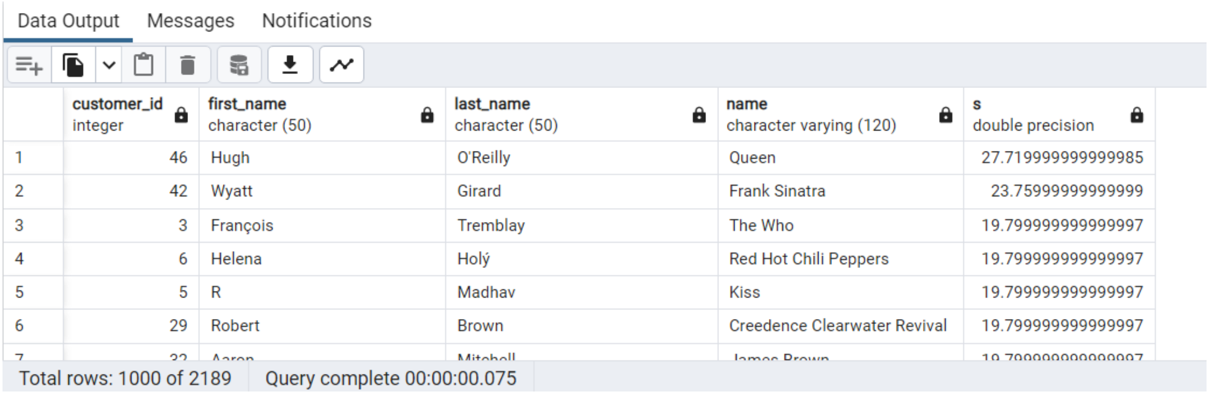
group by 1,2,3,4

order by 5 desc



```
select
c.customer_id,c.first_name,c.last_name,artist.name,sum(invoice_line.unit_price*
invoice_line.quantity) as s
from customer c
join invoice on c.customer_id = invoice.customer_id
join invoice_line on invoice.invoice_id = invoice_line.invoice_id
join track on invoice_line.track_id = track.track_id
join album on album.album_id = track.album_id
join artist on album.artist_id = artist.artist_id
group by 1,2,3,4
order by 5 desc
```

## OUTPUT:-



| | customer_id<br>integer | first_name<br>character (50) | last_name<br>character (50) | name<br>character varying (120) | s<br>double precision |
|---|---|---|---|---|---|
| 1 | 46 | Hugh | O'Reilly | Queen | 27.719999999999985 |
| 2 | 42 | Wyatt | Girard | Frank Sinatra | 23.75999999999999 |
| 3 | 3 | François | Tremblay | The Who | 19.799999999999997 |
| 4 | 6 | Helena | Holý | Red Hot Chili Peppers | 19.799999999999997 |
| 5 | 5 | R | Madhav | Kiss | 19.799999999999997 |
| 6 | 29 | Robert | Brown | Creedence Clearwater Revival | 19.799999999999997 |
| 7 | 32 | Aaron | Mitchell | James Brown | 19.799999999999997 |

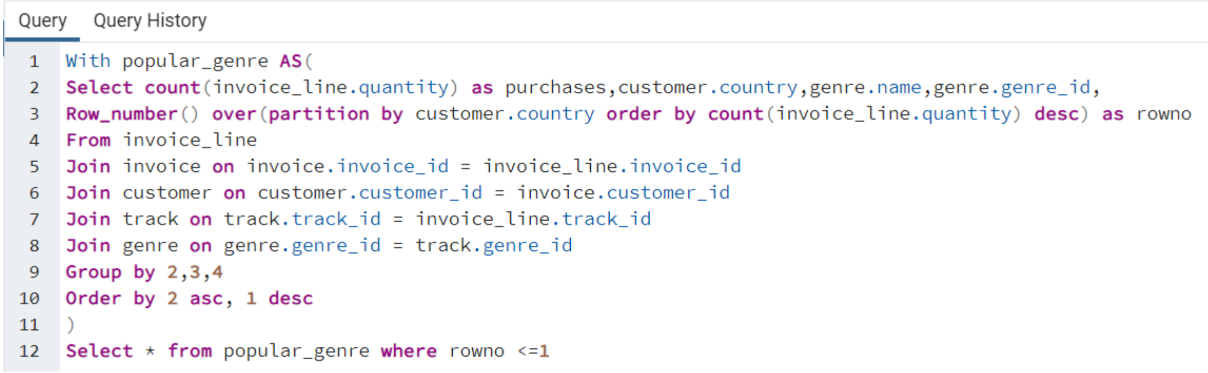
Total rows: 1000 of 2189 Query complete 00:00:00.075

# 10TH QUESTION

**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.**

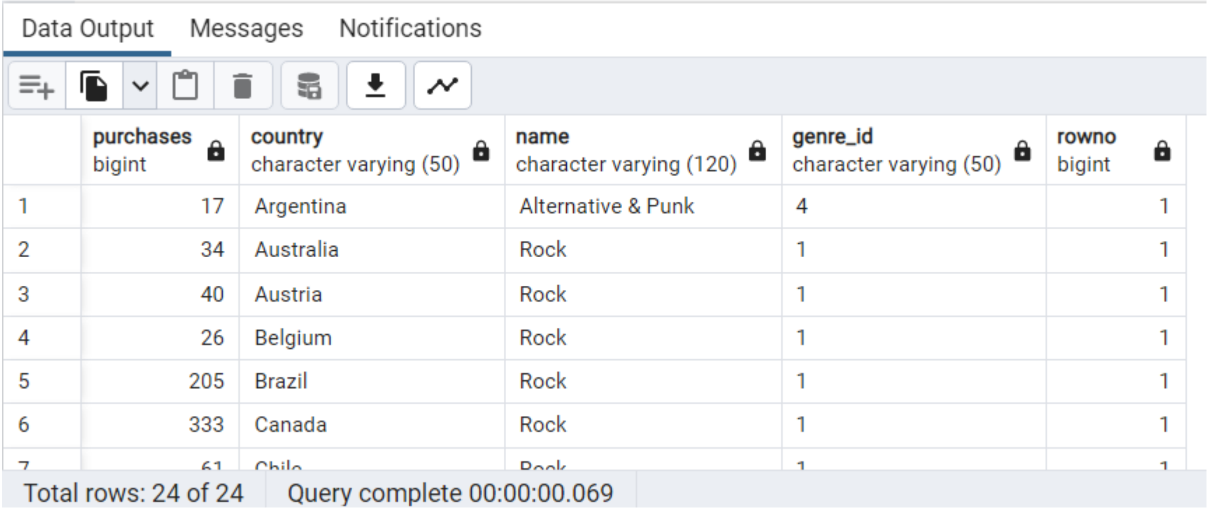
**QUERY:-**

```
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
```



```
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
```

# OUTPUT:-



| | purchases<br>bigint | country<br>character varying (50) | name<br>character varying (120) | genre_id<br>character varying (50) | rowno<br>bigint |
|---|---|---|---|---|---|
| 1 | 17 | Argentina | Alternative & Punk | 4 | 1 |
| 2 | 34 | Australia | Rock | 1 | 1 |
| 3 | 40 | Austria | Rock | 1 | 1 |
| 4 | 26 | Belgium | Rock | 1 | 1 |
| 5 | 205 | Brazil | Rock | 1 | 1 |
| 6 | 333 | Canada | Rock | 1 | 1 |
| 7 | 61 | Chile | Rock | 1 | 1 |

Total rows: 24 of 24 Query complete 00:00:00.069

# 11TH QUESTION

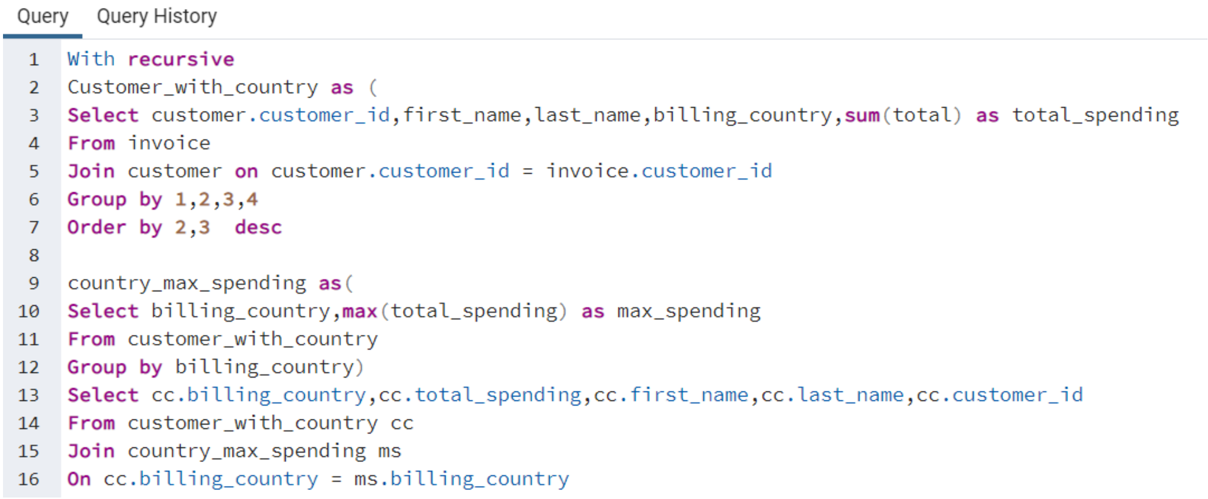
**WRITE A QUERY THAT DETERMINES THE CUSTOMER THAT HAS SPENT THE MOST ON MUSIC FOR EACH COUNTRY. WRITE A QUERY THAT RETURNS THE COUNTRY ALONG WITH THE TOP CUSTOMER AND HOW MUCH THEY SPENT. FOR COUNTRIES WHERE THE TOP AMOUNT SPENT IS SHARED, PROVIDE ALL CUSTOMERS WHO SPENT THIS AMOUNT.**

**QUERY:-**

```
With recursive

Customer_with_country as (

Select customer.customer_id,first_name,last_name,billing_country,sum(total) as total_spending

From invoice

Join customer on customer.customer_id = invoice.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 customer_with_country

Group by billing_country)

Select cc.billing_country,cc.total_spending,cc.first_name,cc.last_name,cc.customer_id

From customer_with_country cc
```

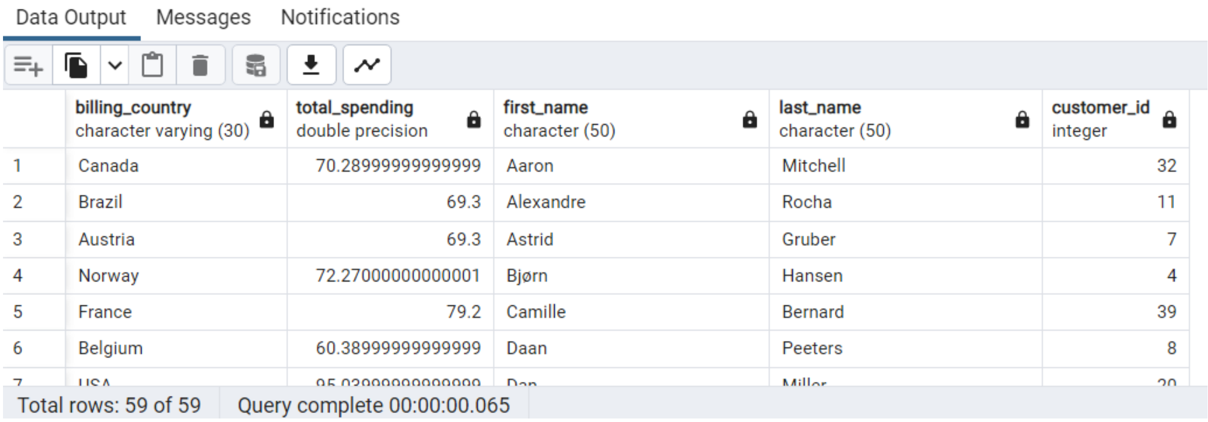
Join country_max_spending ms

On cc.billing_country = ms.billing_country



```sql
With recursive
Customer_with_country as (
Select customer.customer_id,first_name,last_name,billing_country,sum(total) as total_spending
From invoice
Join customer on customer.customer_id = invoice.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 customer_with_country
Group by billing_country)
Select cc.billing_country,cc.total_spending,cc.first_name,cc.last_name,cc.customer_id
From customer_with_country cc
Join country_max_spending ms
On cc.billing_country = ms.billing_country
```

## OUTPUT:-



| | billing_country<br>character varying (30) | total_spending<br>double precision | first_name<br>character (50) | last_name<br>character (50) | customer_id<br>integer |
|---|---|---|---|---|---|
| 1 | Canada | 70.28999999999999 | Aaron | Mitchell | 32 |
| 2 | Brazil | 69.3 | Alexandre | Rocha | 11 |
| 3 | Austria | 69.3 | Astrid | Gruber | 7 |
| 4 | Norway | 72.27000000000001 | Bjørn | Hansen | 4 |
| 5 | France | 79.2 | Camille | Bernard | 39 |
| 6 | Belgium | 60.38999999999999 | Daan | Peeters | 8 |
| 7 | USA | 95.03999999999999 | Dan | Miller | 20 |

Total rows: 59 of 59    Query complete 00:00:00.065