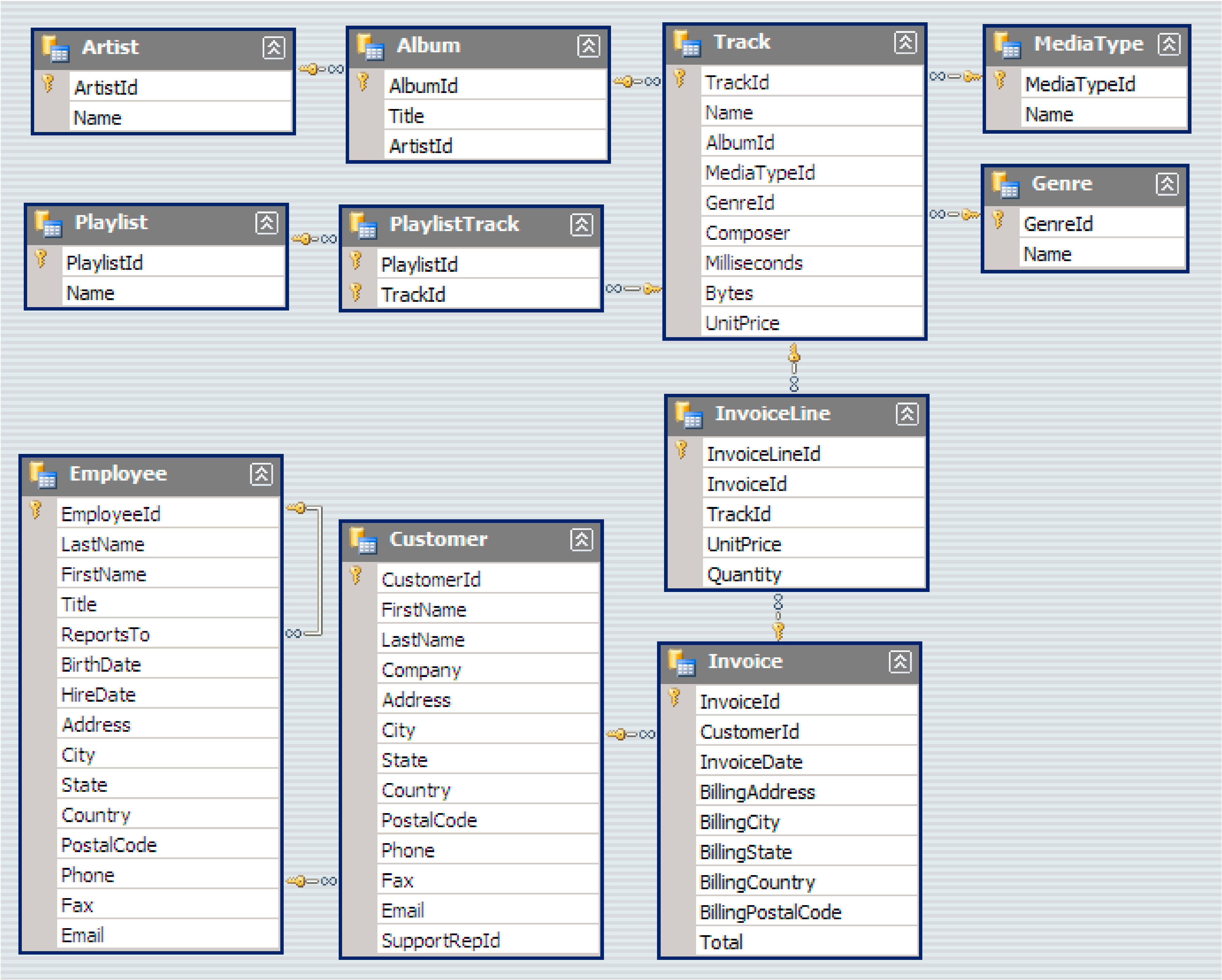
working on PostgreSQL | pgAdmin 16

SQL PROJECT

MUSIC STORE DATA ANALYSIS

Akash Kumar

Schema Diagram



Artist
ArtistId
Name

Album
AlbumId
Title
ArtistId

Track
TrackId
Name
AlbumId
MediaTypeId
GenreId
Composer
Milliseconds
Bytes
UnitPrice

MediaType
MediaTypeId
Name

Genre
GenreId
Name

Playlist
PlaylistId
Name

PlaylistTrack
PlaylistId
TrackId

InvoiceLine
InvoiceLineId
InvoiceId
TrackId
UnitPrice
Quantity

Employee
EmployeeId
LastName
FirstName
Title
ReportsTo
BirthDate
HireDate
Address
City
State
Country
PostalCode
Phone
Fax
Email

Customer
CustomerId
FirstName
LastName
Company
Address
City
State
Country
PostalCode
Phone
Fax
Email
SupportRepId

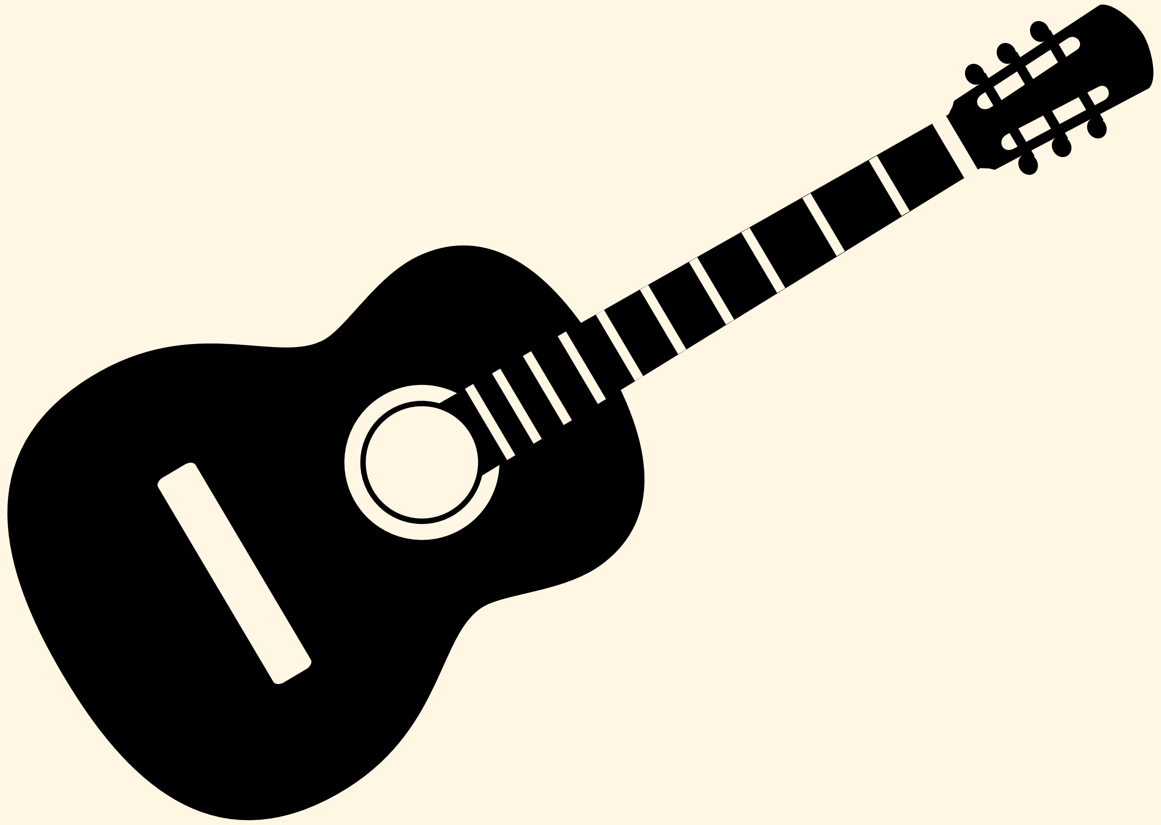
Invoice
InvoiceId
CustomerId
InvoiceDate
BillingAddress
BillingCity
BillingState
BillingCountry
BillingPostalCode
Total

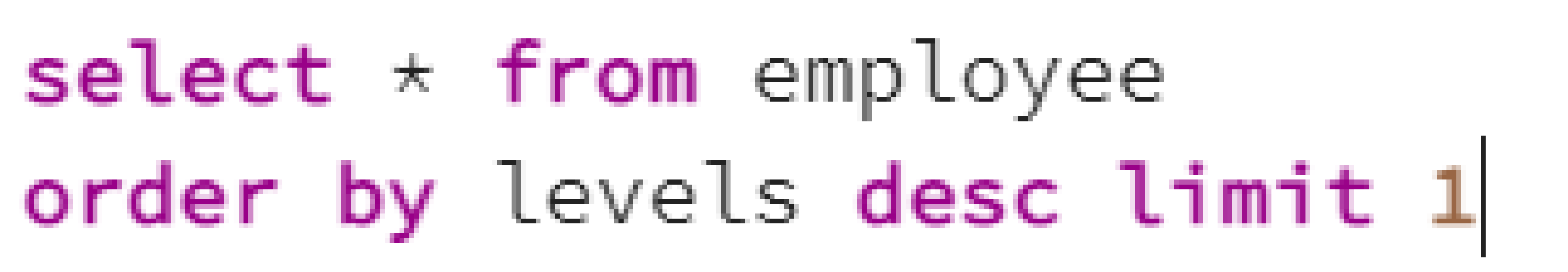
CSV Links of Tables

- employee
- genre
- media type
- playlist
- customer
- track
- album
- playlist_track
- invoice
- invoice_line
- artist

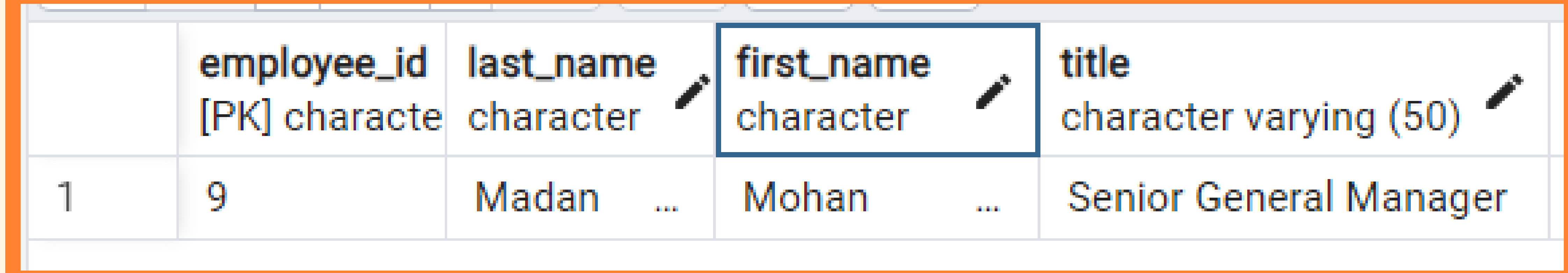
1

Who is the senior most employee based on job title?





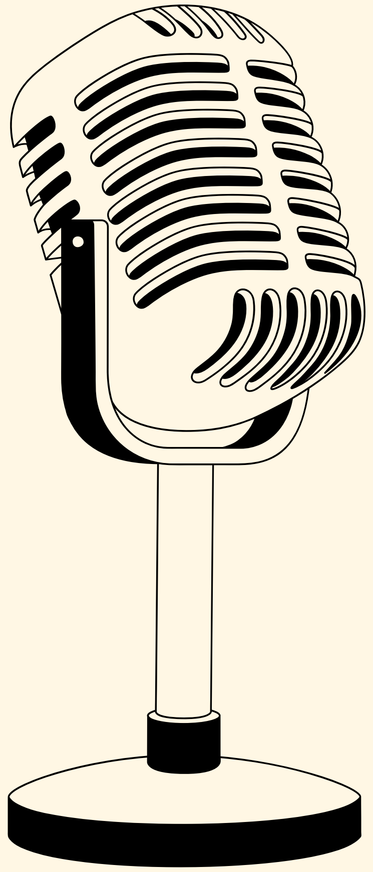
```
select * from employee
order by levels desc limit 1
```

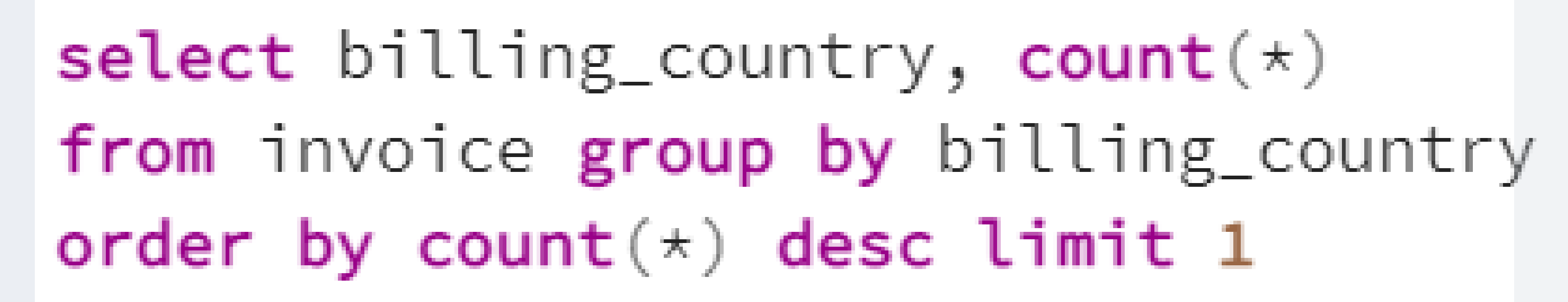


	employee_id [PK] characte	last_name character	first_name character	title character varying (50)
1	9	Madan ...	Mohan ...	Senior General Manager

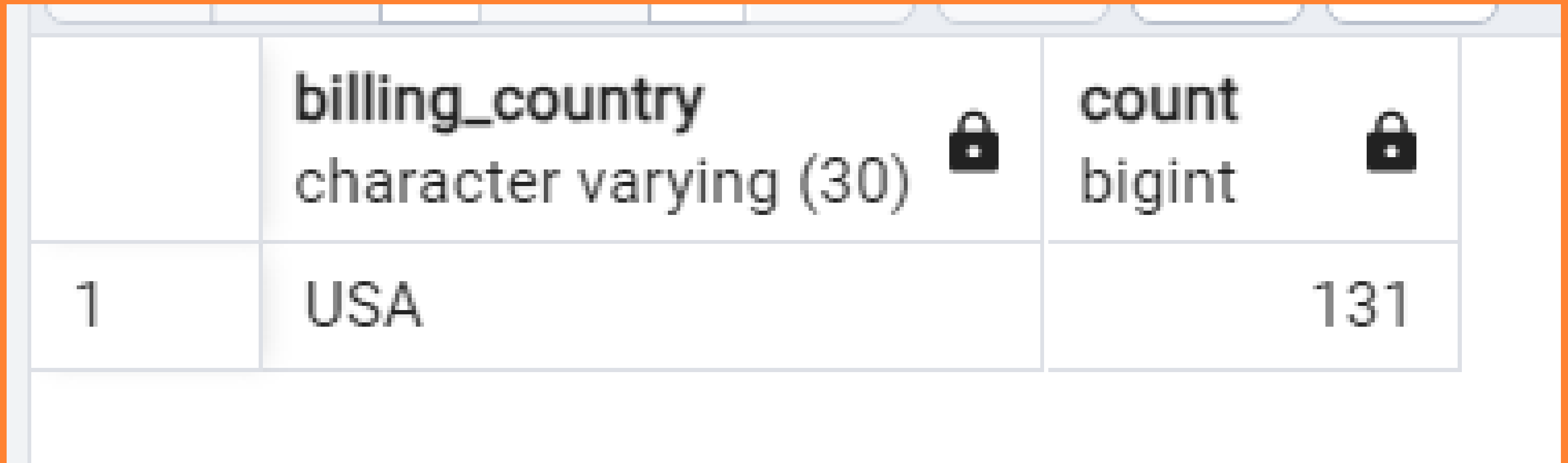
# 2

Which countries have the most Invoices?





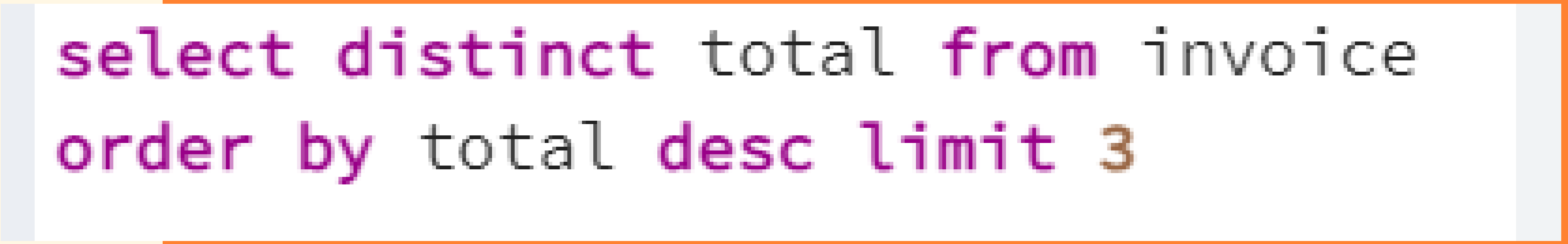
```sql
select billing_country, count(*)
from invoice group by billing_country
order by count(*) desc limit 1
```



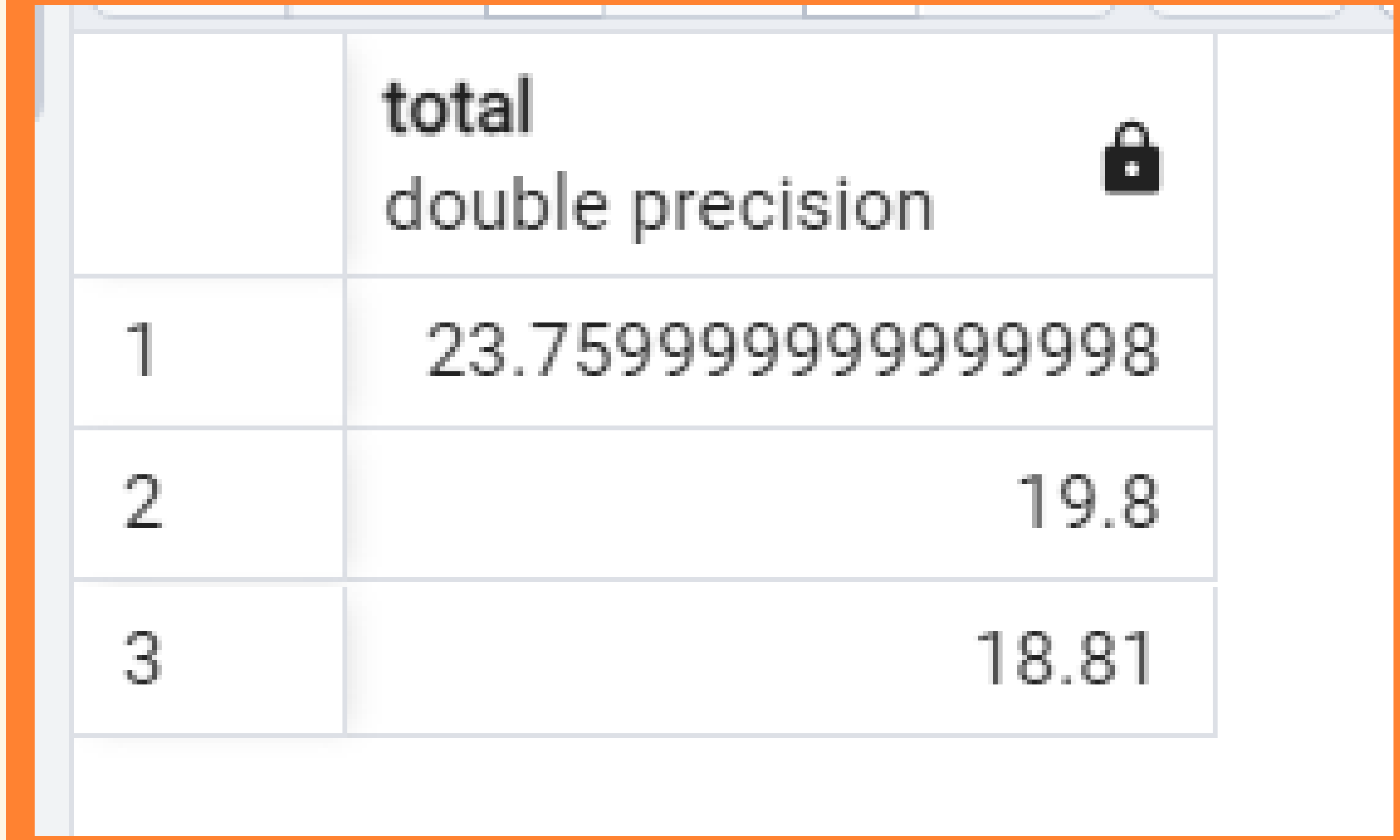
	billing_country character varying (30)	count bigint
1	USA	131

3

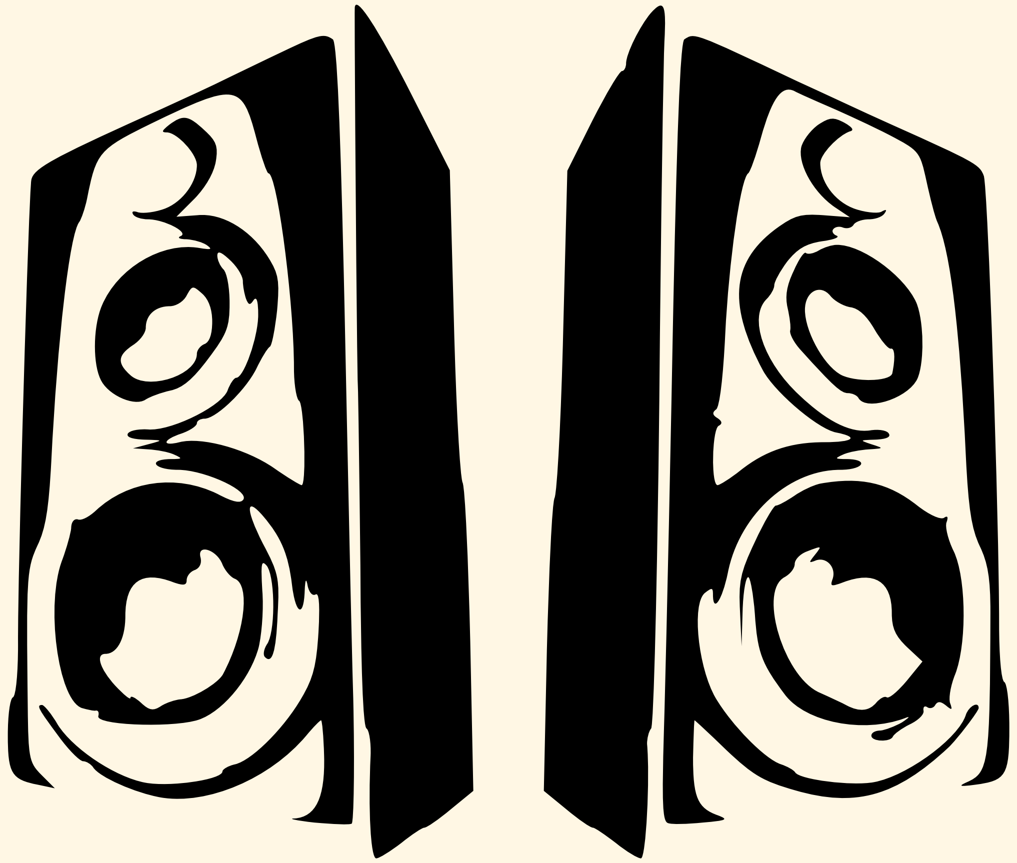
What are top 3 values of total invoice?



```
select distinct total from invoice
order by total desc limit 3
```



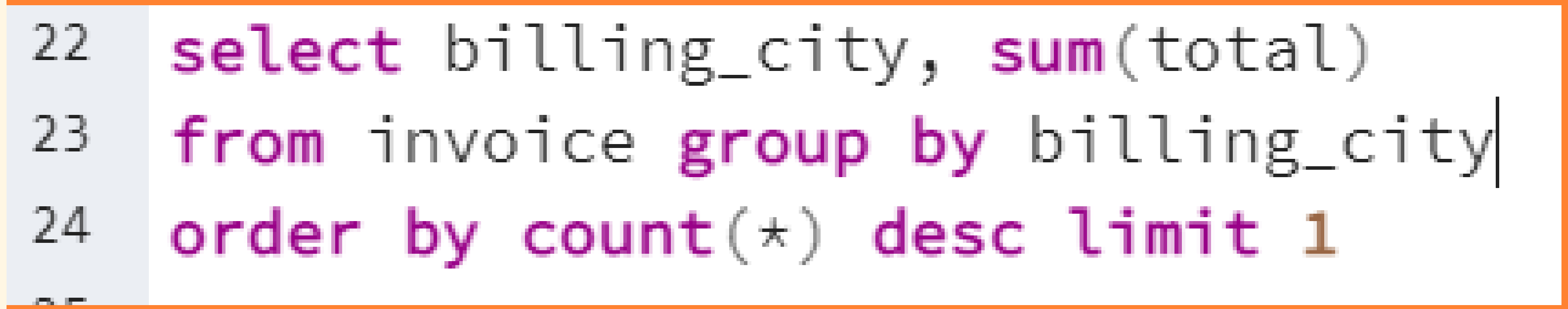
	total double precision
1	23.7599999999999998
2	19.8
3	18.81



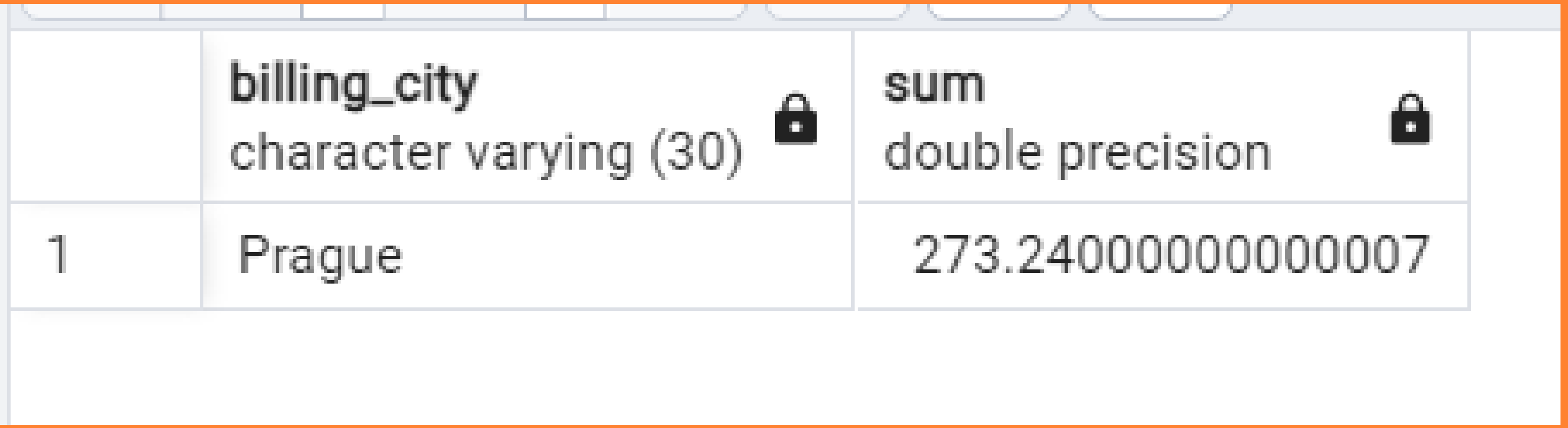
4

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



```sql
select billing_city, sum(total)
from invoice group by billing_city
order by count(*) desc limit 1
```

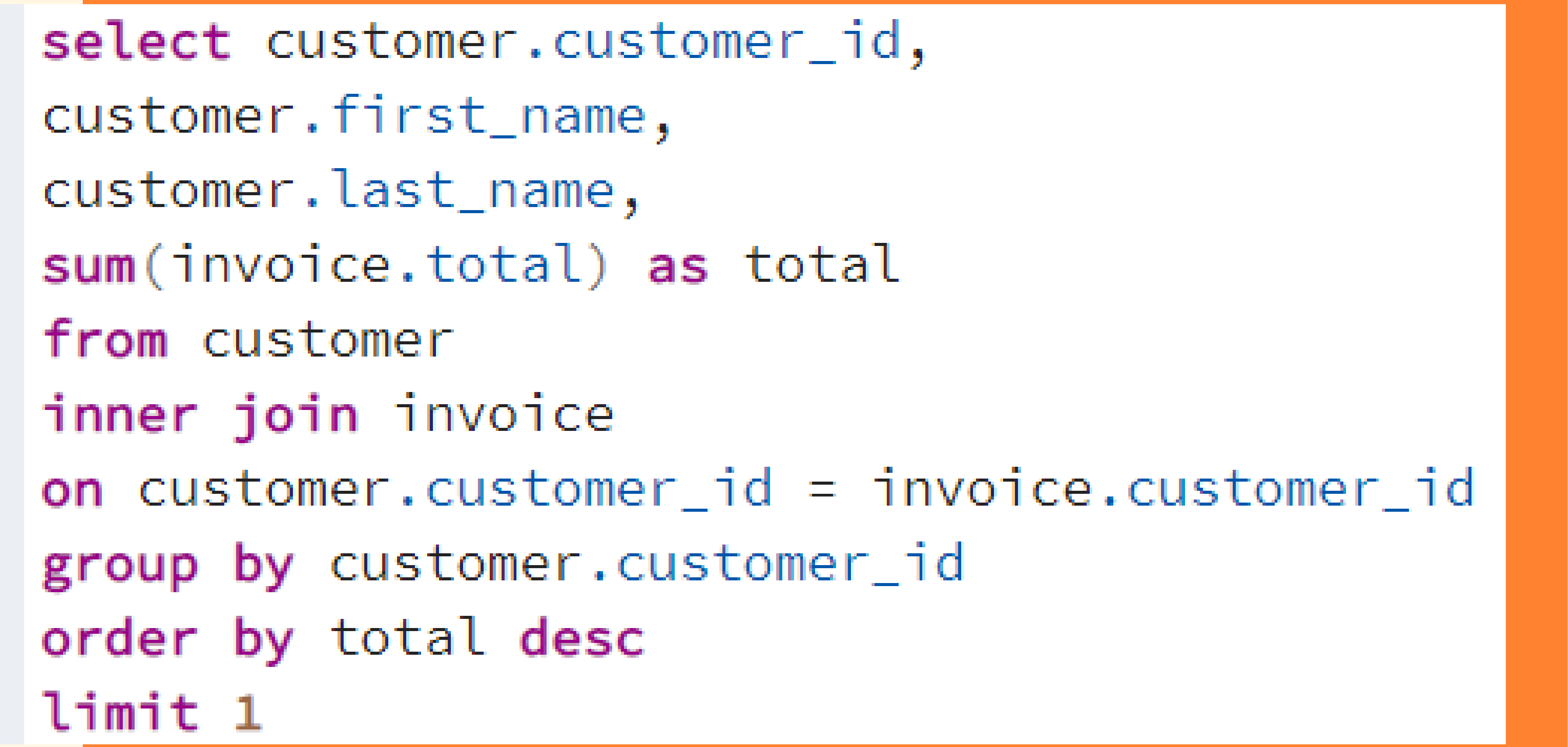


	billing_city character varying (30)	sum double precision
1	Prague	273.2400000000007

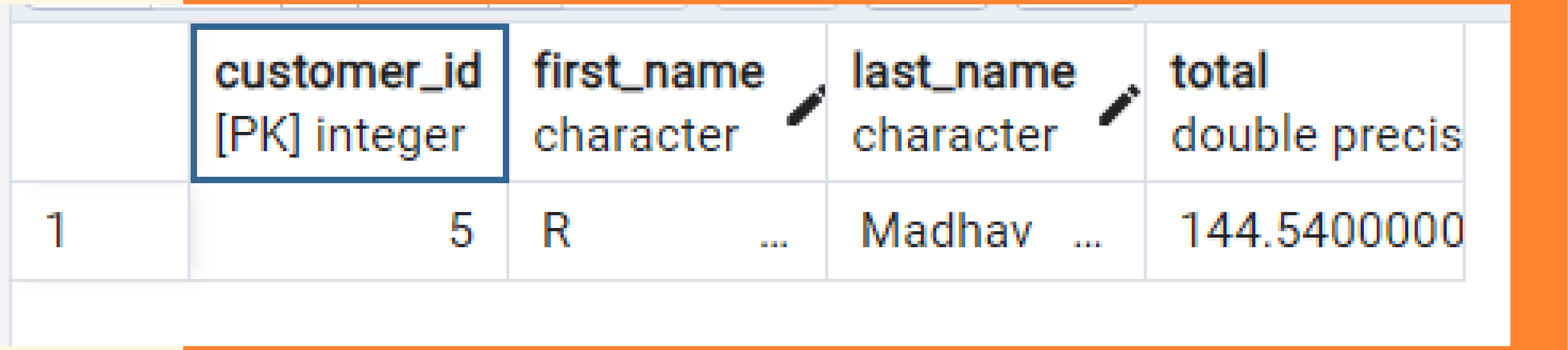
5

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

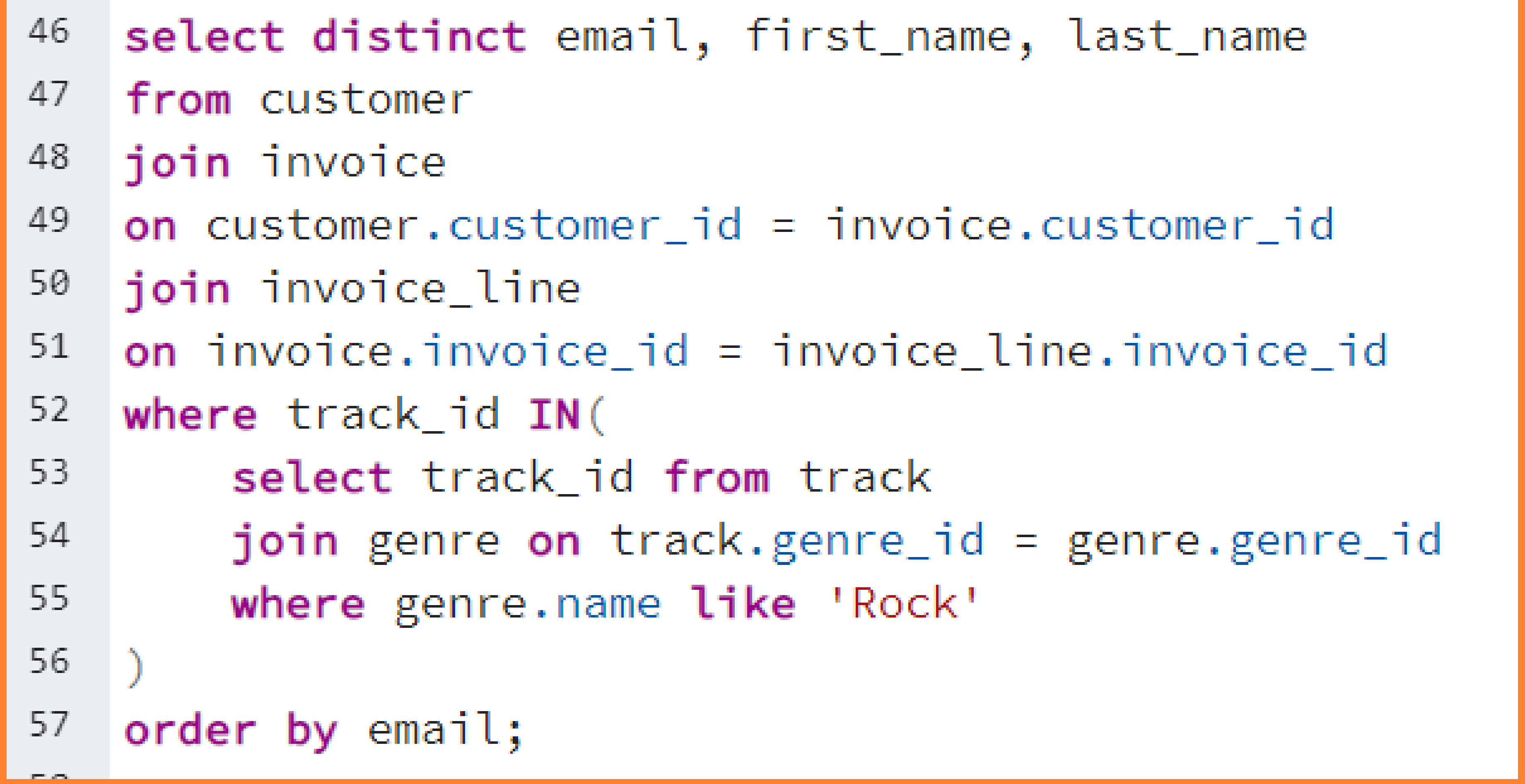


	customer_id [PK] integer	first_name character	last_name character	total double precis
1	5	R ...	Madhav ...	144.5400000

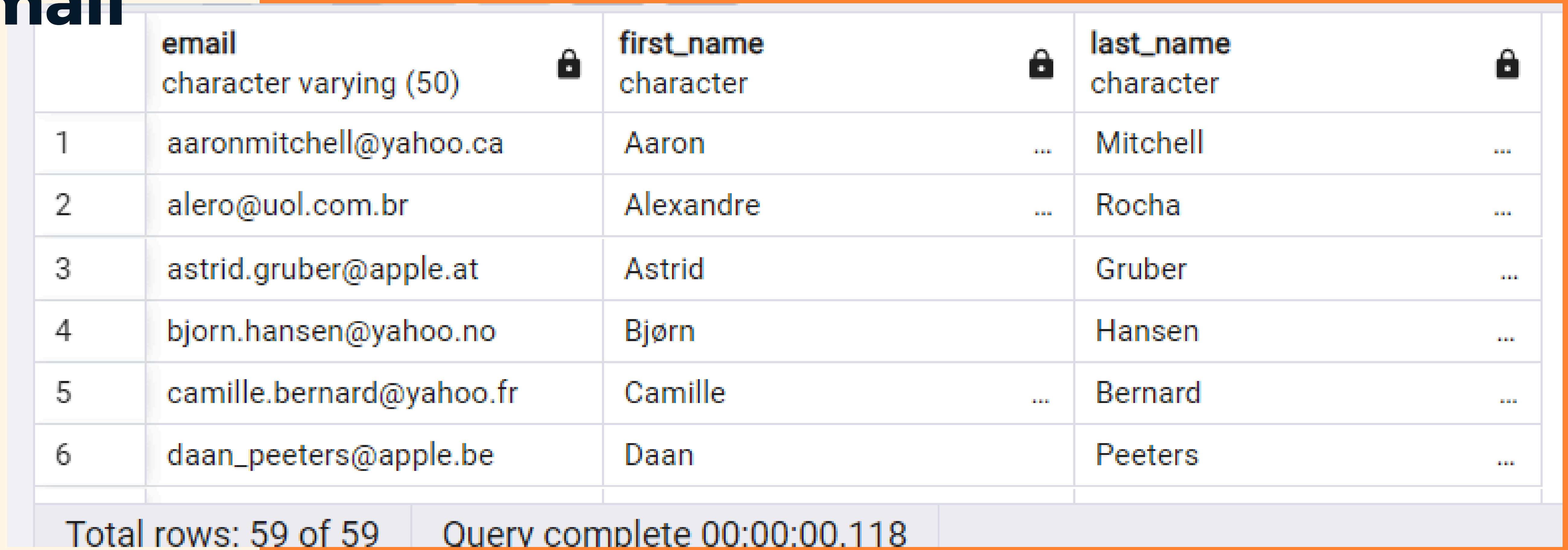
6

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



```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;
```



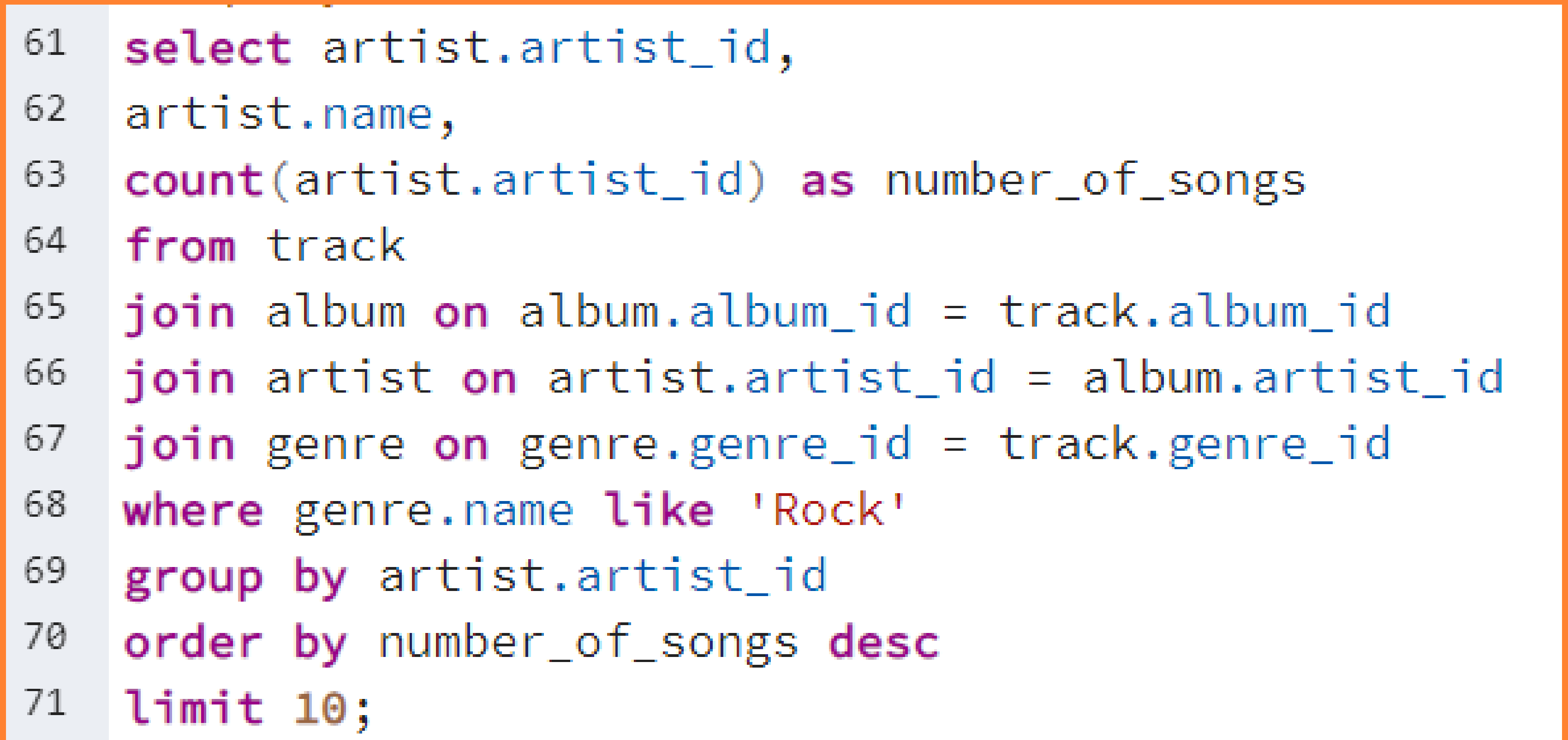
	email character varying (50)	first_name character	last_name character
1	aaronmitchell@yahoo.ca	Aaron	Mitchell
2	alero@uol.com.br	Alexandre	Rocha
3	astrid.gruber@apple.at	Astrid	Gruber
4	bjorn.hansen@yahoo.no	Bjørn	Hansen
5	camille.bernard@yahoo.fr	Camille	Bernard
6	daan_peeters@apple.be	Daan	Peeters

Total rows: 59 of 59   Query complete 00:00:00.118

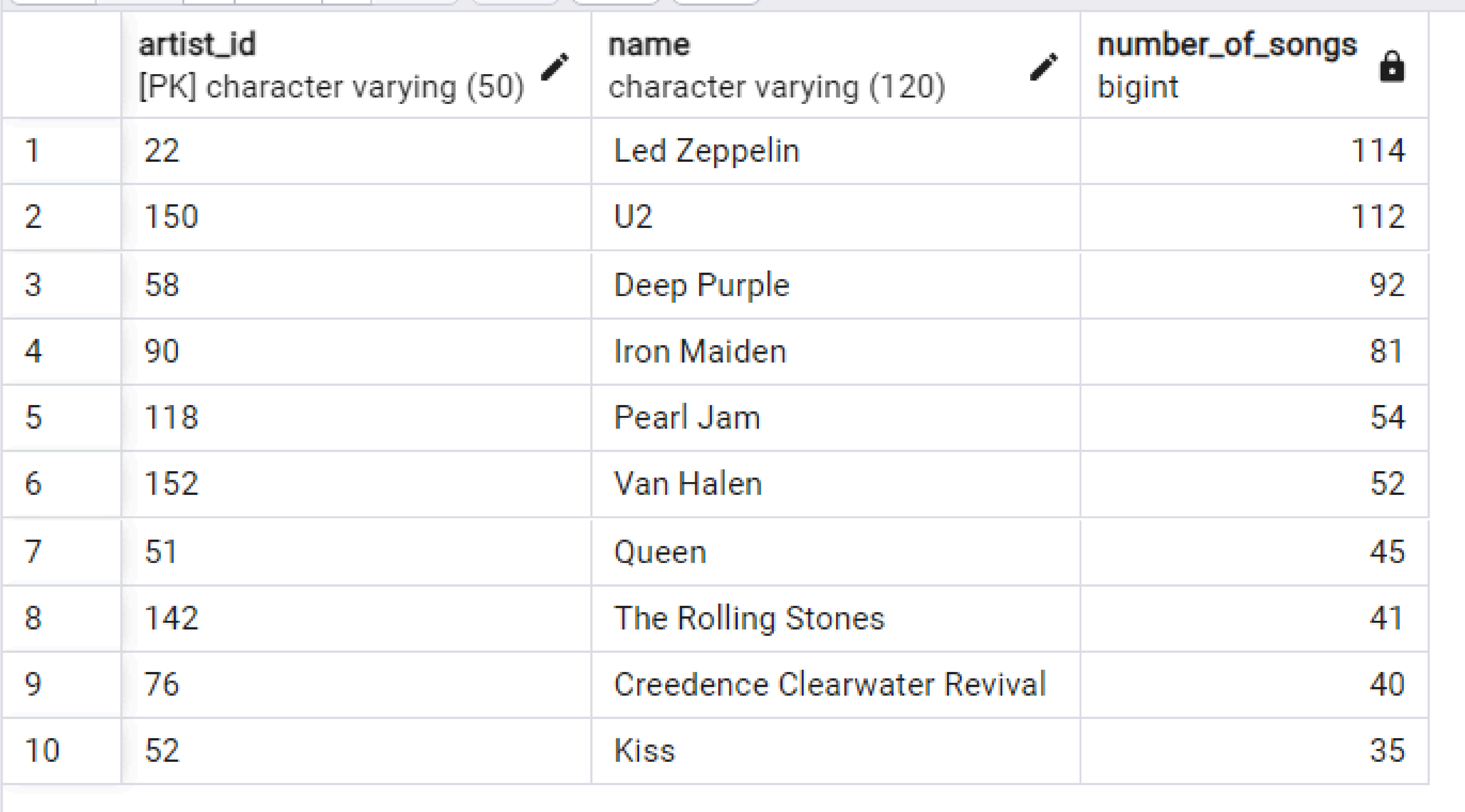
7

Let's invite the artists who have written the most rock music in our dataset.

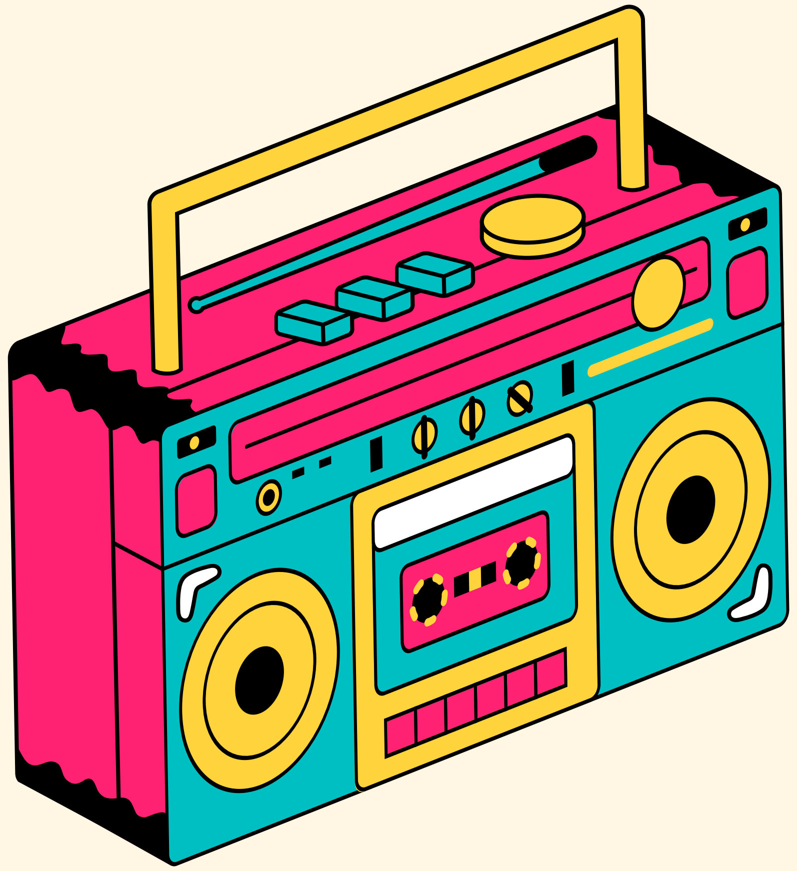
Write a query that returns the Artist name and total track count of the top 10 rock bands



```sql
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 [PK] character varying (50)	name character varying (120)	number_of_songs bigint
1	22	Led Zeppelin	114
2	150	U2	112
3	58	Deep Purple	92
4	90	Iron Maiden	81
5	118	Pearl Jam	54
6	152	Van Halen	52
7	51	Queen	45
8	142	The Rolling Stones	41
9	76	Creedence Clearwater Revival	40
10	52	Kiss	35

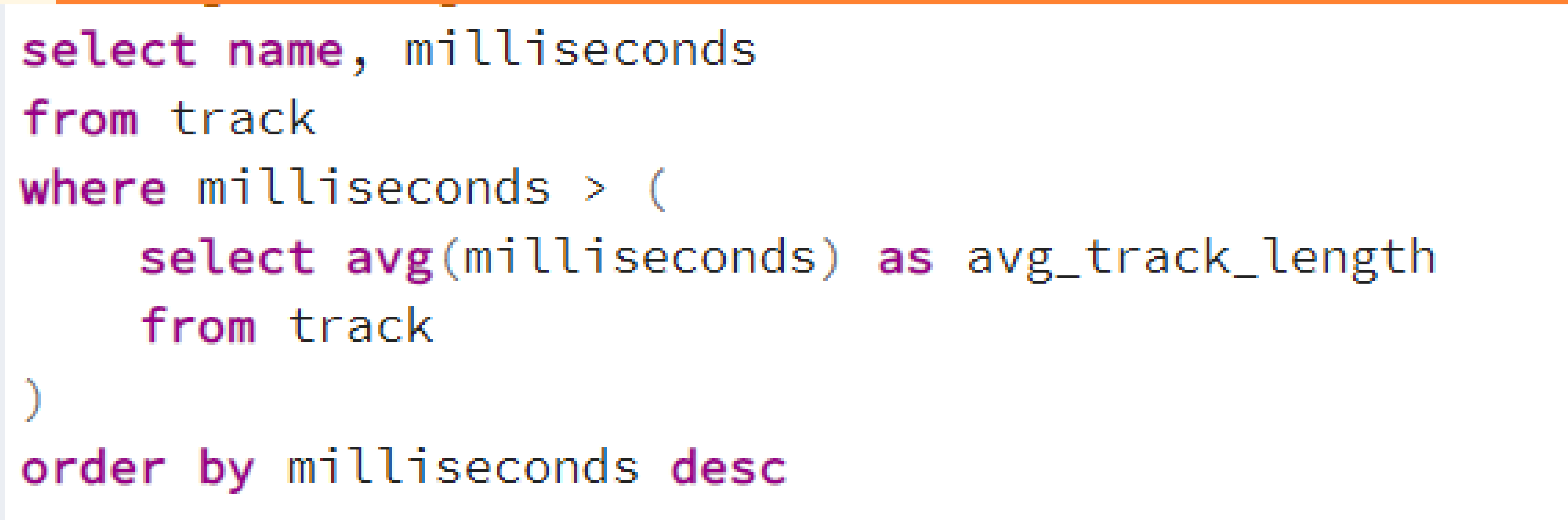


8

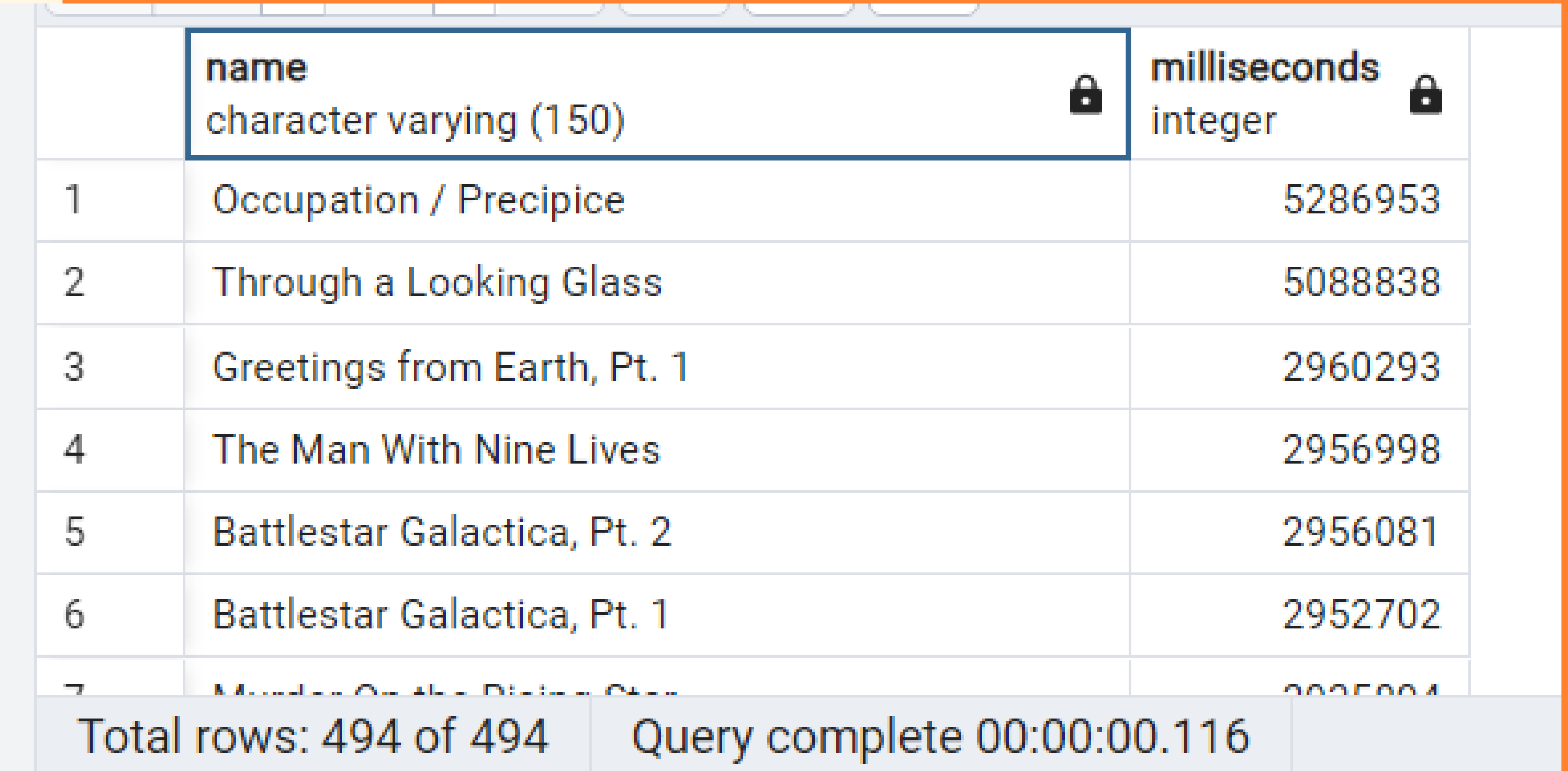
Return all the track names that have a song length longer than the average song length.

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
```

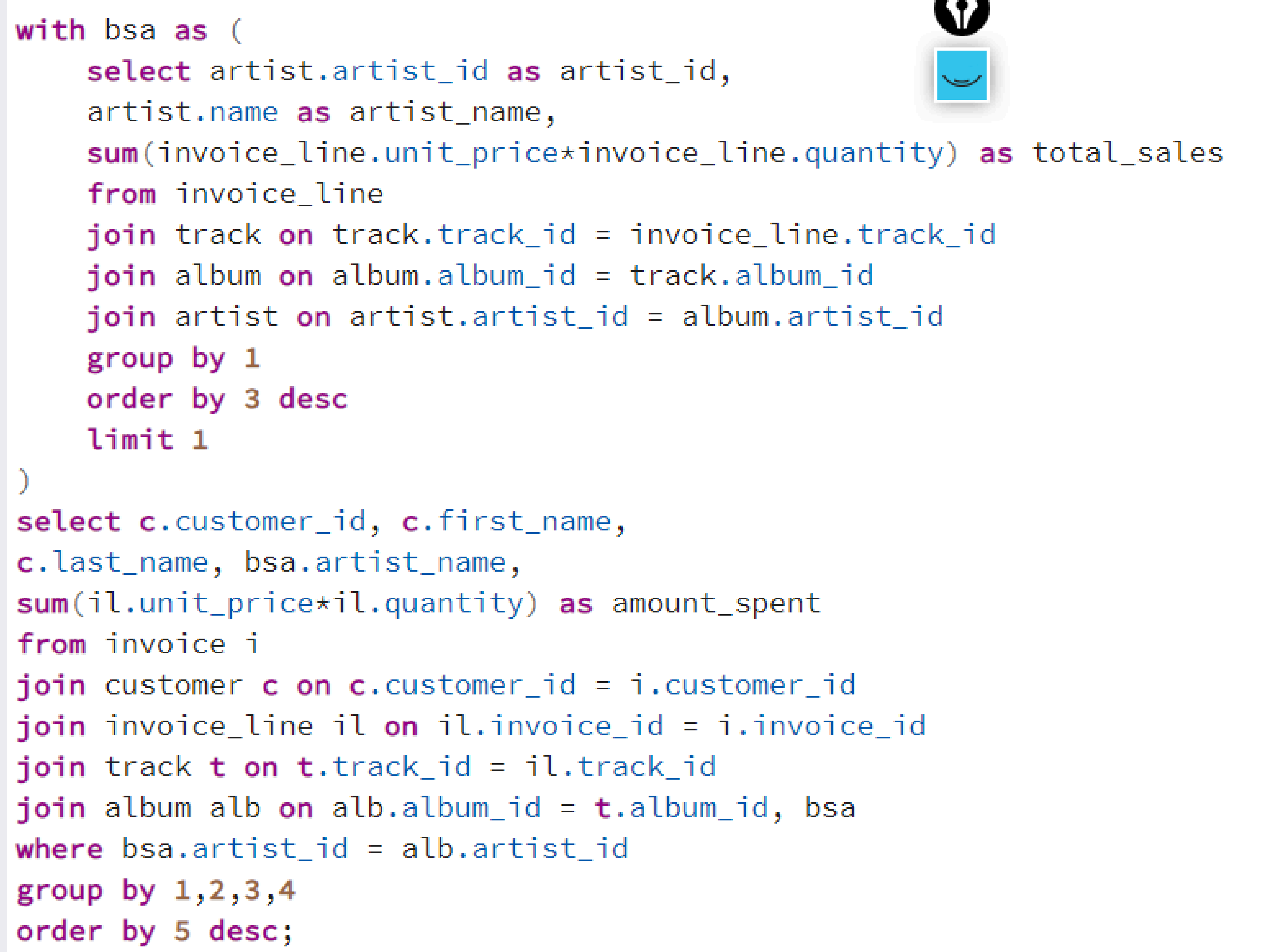


	name character varying (150)	milliseconds integer
1	Occupation / Precipice	5286953
2	Through a Looking Glass	5088838
3	Greetings from Earth, Pt. 1	2960293
4	The Man With Nine Lives	2956998
5	Battlestar Galactica, Pt. 2	2956081
6	Battlestar Galactica, Pt. 1	2952702
7	Murder On the Rising Star	2935894

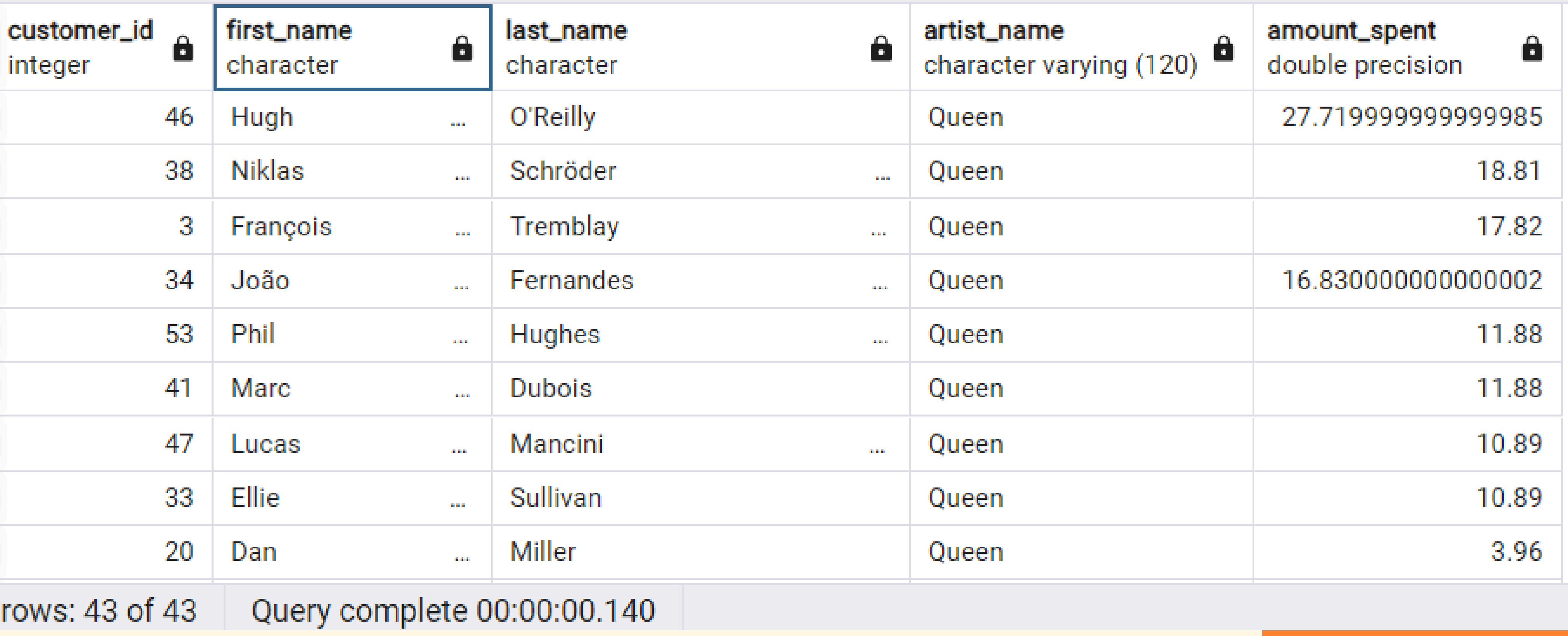
Total rows: 494 of 494 Query complete 00:00:00.116

9

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



```
with bsa 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, bsa
where bsa.artist_id = alb.artist_id
group by 1,2,3,4
order by 5 desc;
```



customer_id integer	first_name character	last_name character	artist_name character varying (120)	amount_spent double precision
46	Hugh ...	O'Reilly	Queen	27.719999999999985
38	Niklas ...	Schröder ...	Queen	18.81
3	François ...	Tremblay ...	Queen	17.82
34	João ...	Fernandes ...	Queen	16.830000000000002
53	Phil ...	Hughes ...	Queen	11.88
41	Marc ...	Dubois	Queen	11.88
47	Lucas ...	Mancini ...	Queen	10.89
33	Ellie ...	Sullivan	Queen	10.89
20	Dan ...	Miller	Queen	3.96

rows: 43 of 43 Query complete 00:00:00.140

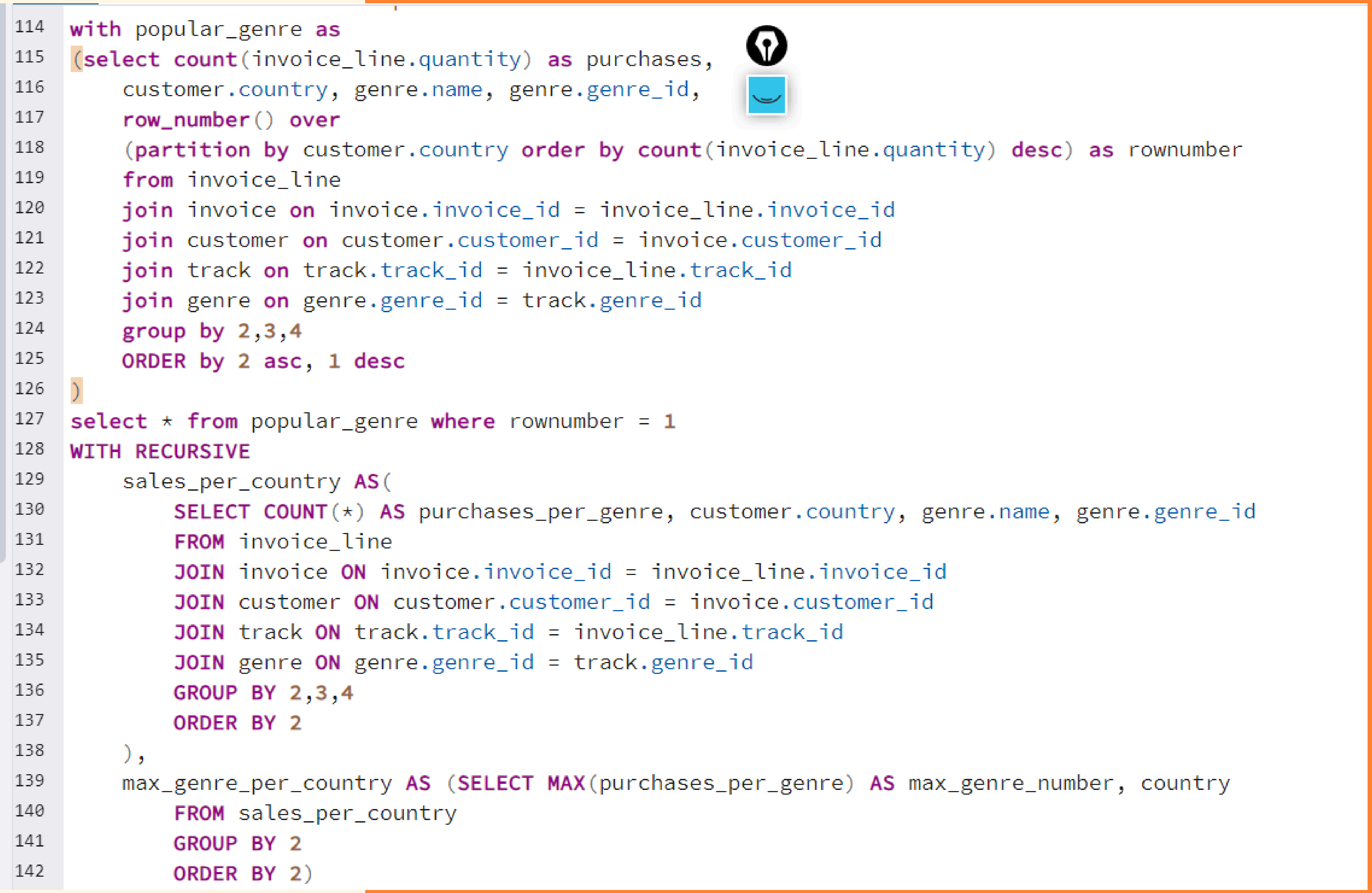
10

We want to find out the most popular music Genre for each country.

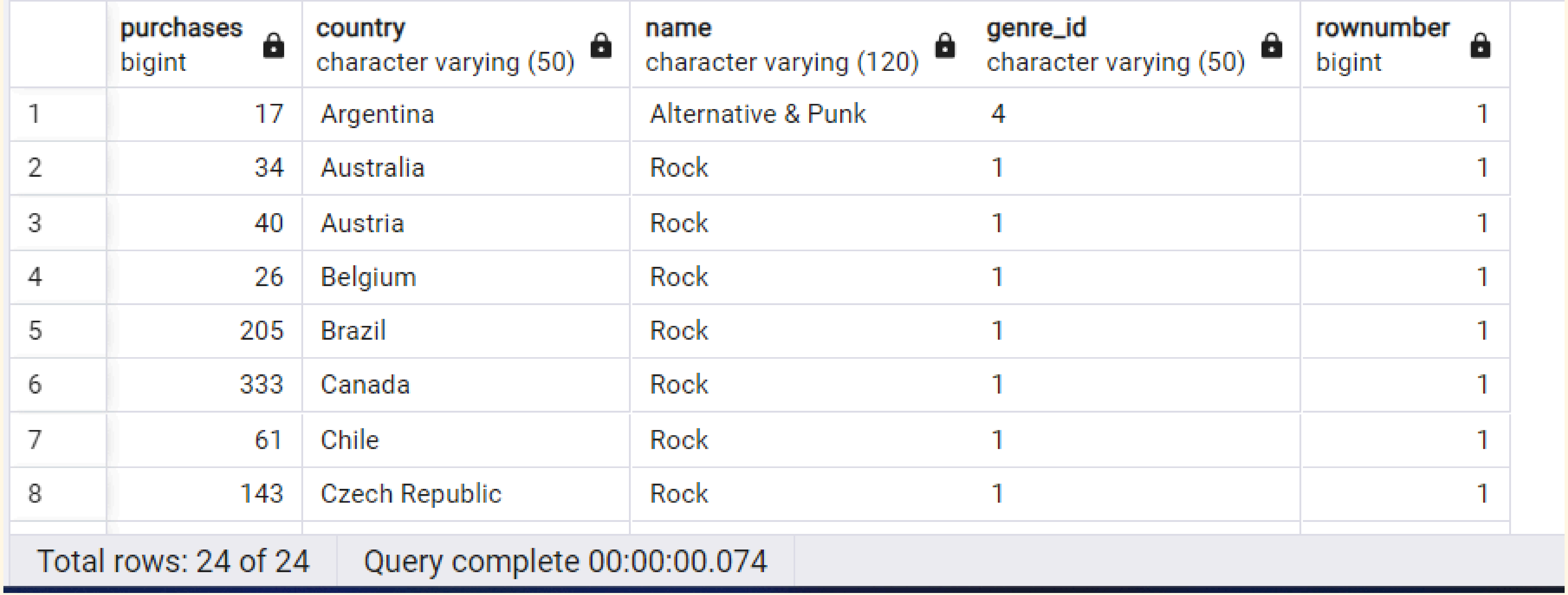
We determine the most popular genre as the genre with the highest amount of purchases.

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(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 rownumber
    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 rownumber = 1
WITH RECURSIVE
    sales_per_country AS(
        SELECT COUNT(*) AS purchases_per_genre, customer.country, genre.name, genre.genre_id
        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
    ),
    max_genre_per_country AS (SELECT MAX(purchases_per_genre) AS max_genre_number, country
        FROM sales_per_country
        GROUP BY 2
        ORDER BY 2)
```



	purchases bigint	country character varying (50)	name character varying (120)	genre_id character varying (50)	rownumber 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
8	143	Czech Republic	Rock	1	1

Total rows: 24 of 24

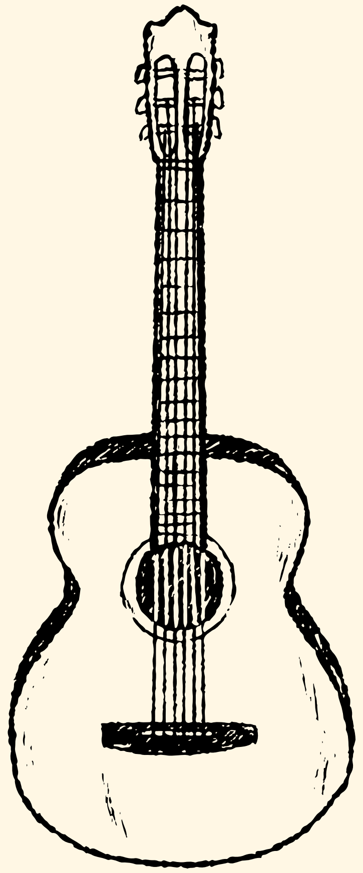
Query complete 00:00:00.074

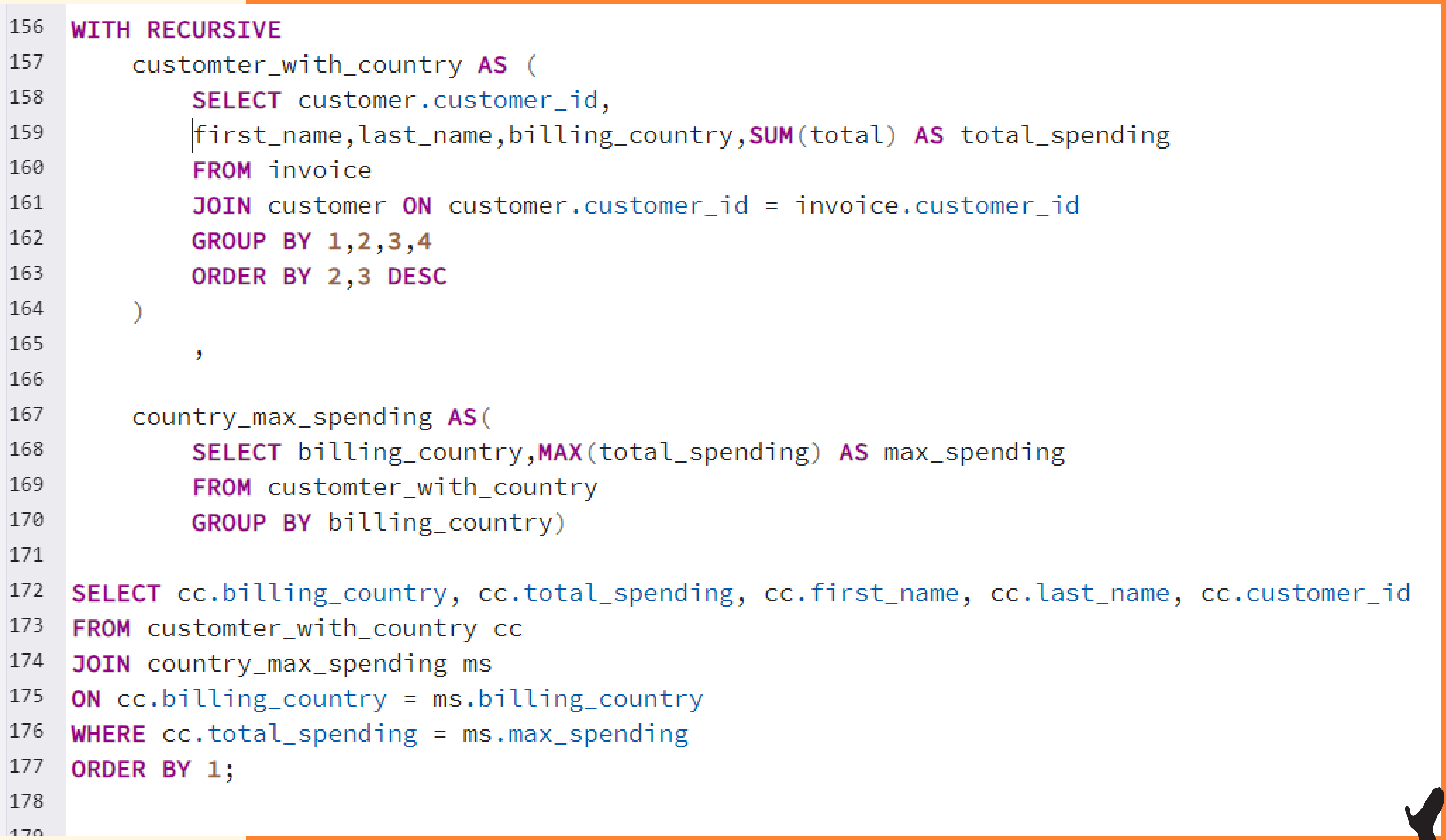
11

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 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 customter_with_country
        GROUP BY billing_country)

SELECT cc.billing_country, cc.total_spending, cc.first_name, cc.last_name, cc.customer_id
FROM customter_with_country cc
JOIN country_max_spending ms
ON cc.billing_country = ms.billing_country
WHERE cc.total_spending = ms.max_spending
ORDER BY 1;
```

Thank you!