PRESENTATION

MUSIC STORE ANALYSIS

SQL PROJECT

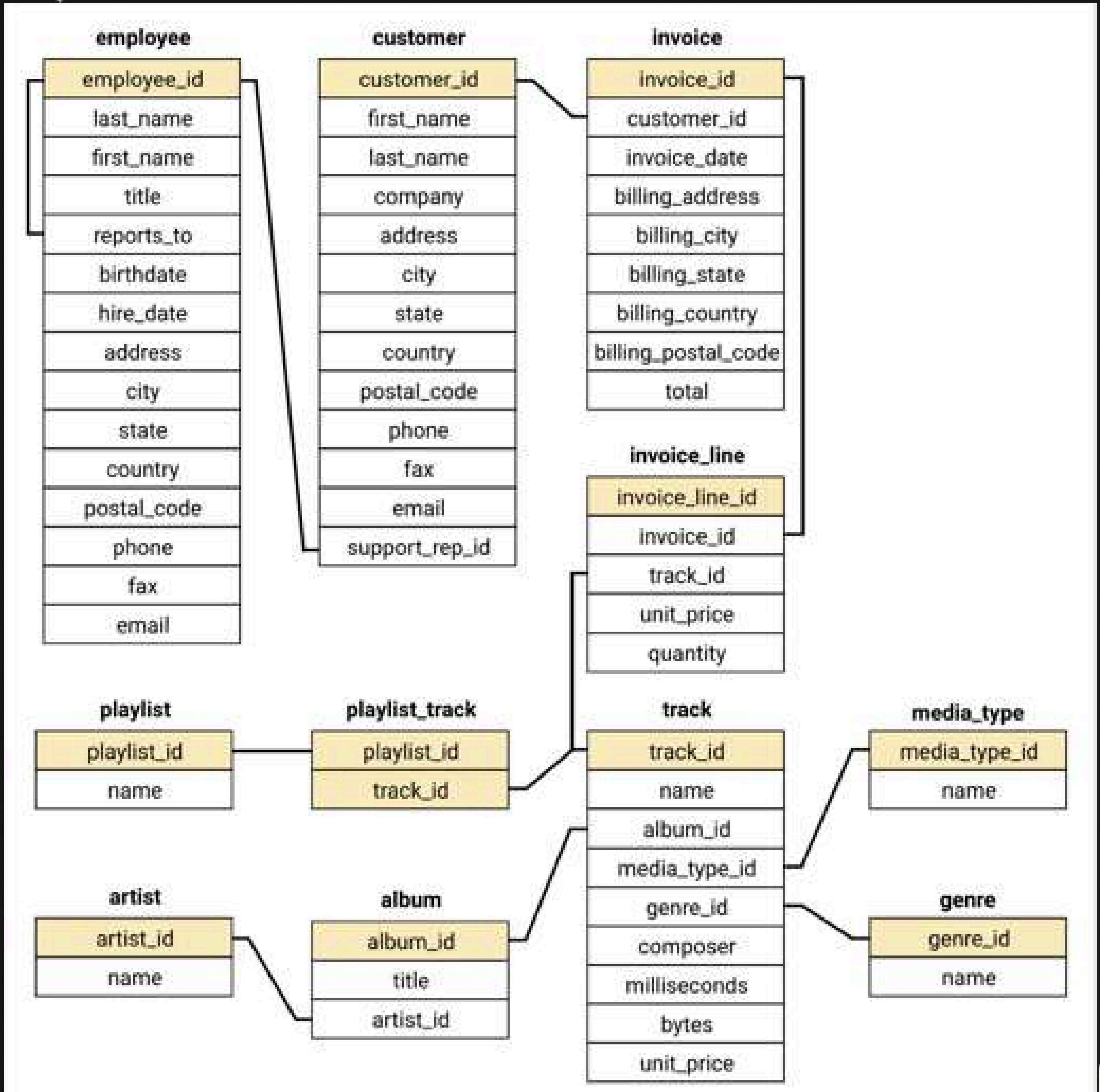
BY: AMIT VERMA

WELCOME TO PROJECT

INTRODUCTION

THIS PROJECT IS BASED ON MUSIC STORE ANALYSIS I HAVE MADE SOME QUESTIONS AND ANSWERS FOR THE PROJECT.WE HAVE GIVEN A DATSET OF MUSIC STORE WITH DIFFERENT COLUMNS WE WILL ANALYZE THE DATA QUESTIONS WITH POSTGRESQL.

SCHEMA_DIAGRAM



employee
employee_id
last_name
first_name
title
reports_to
birthdate
hire_date
address
city
state
country
postal_code
phone
fax
email

customer
customer_id
first_name
last_name
company
address
city
state
country
postal_code
phone
fax
email
support_rep_id

invoice
invoice_id
customer_id
invoice_date
billing_address
billing_city
billing_state
billing_country
billing_postal_code
total

invoice_line
invoice_line_id
invoice_id
track_id
unit_price
quantity

playlist
playlist_id
name

playlist_track
playlist_id
track_id

track
track_id
name
album_id
media_type_id
genre_id
composer
milliseconds
bytes
unit_price

media_type
media_type_id
name

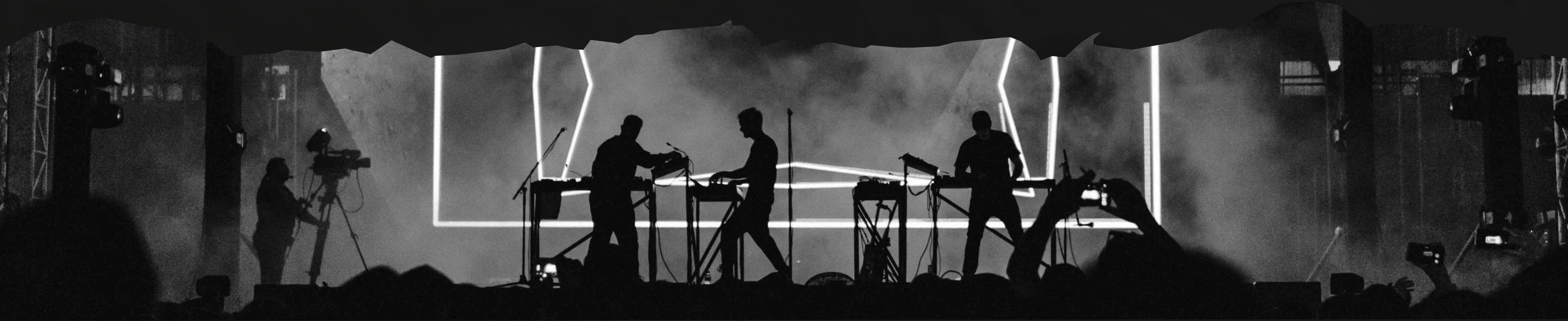
artist
artist_id
name

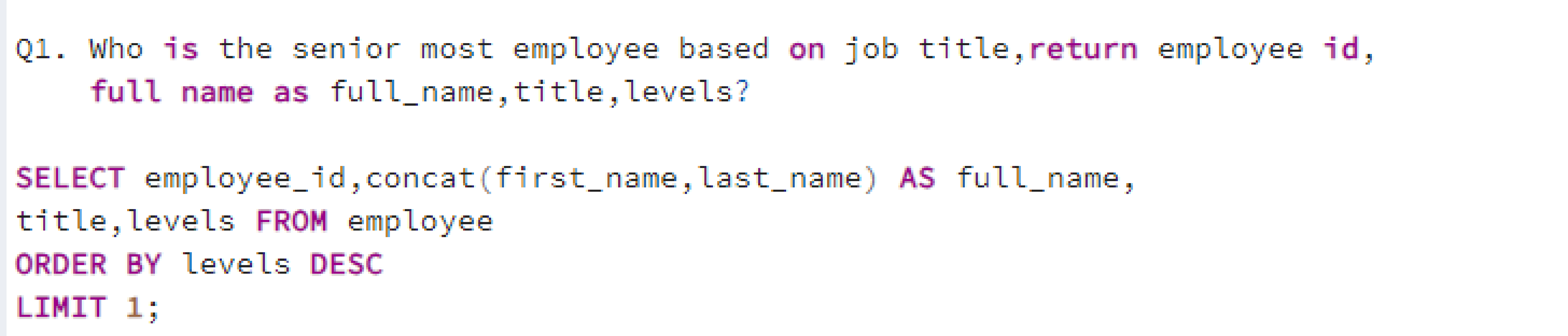
album
album_id
title
artist_id

genre
genre_id
name

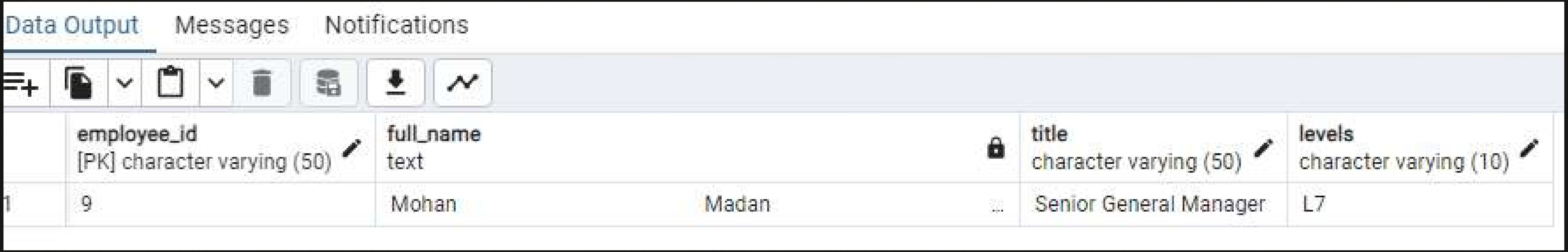
LET’S START WITH OUR QUESTIONS.

QUESTION SET 1 - EASY

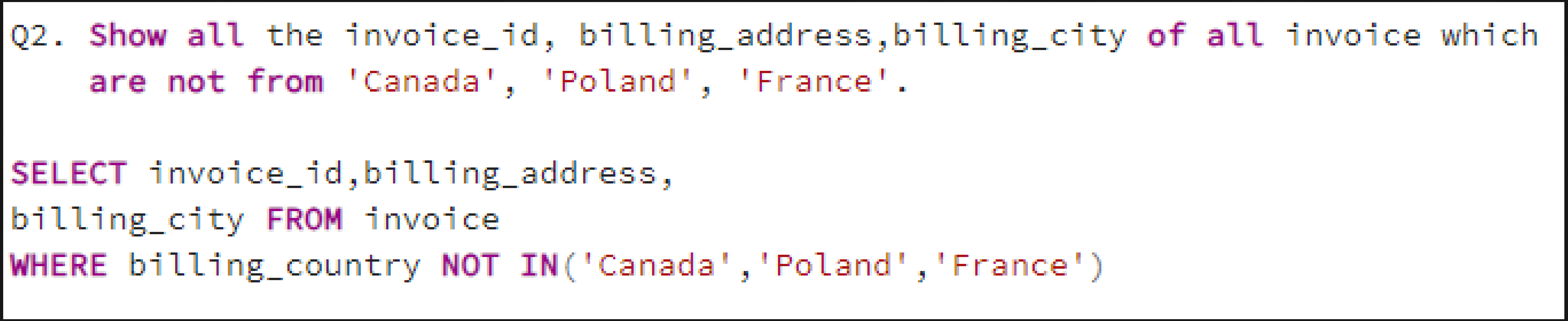




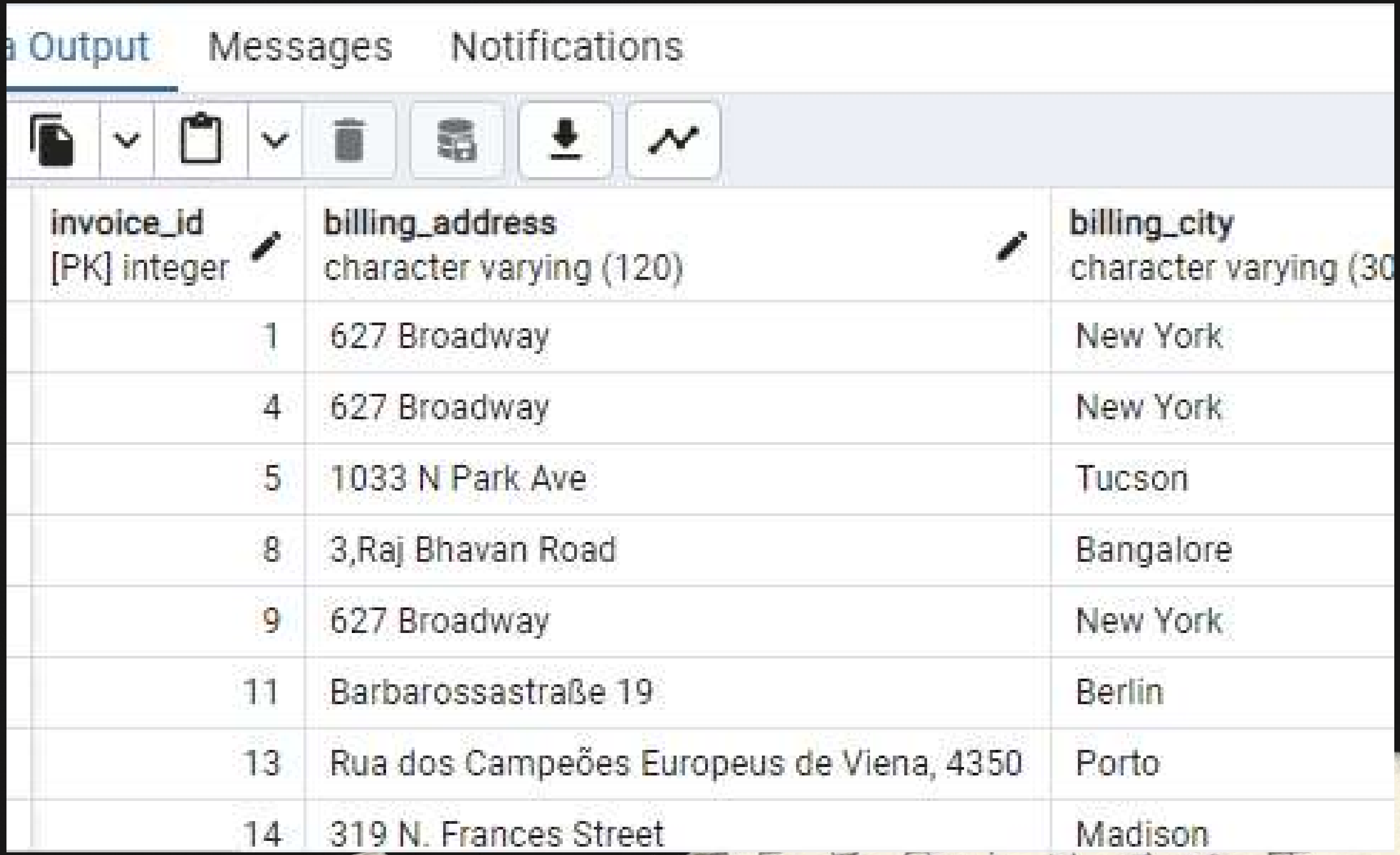
```
Q1. Who is the senior most employee based on job title,return employee id,
    full name as full_name,title,levels?

SELECT employee_id,concat(first_name,last_name) AS full_name,
title,levels FROM employee
ORDER BY levels DESC
LIMIT 1;
```



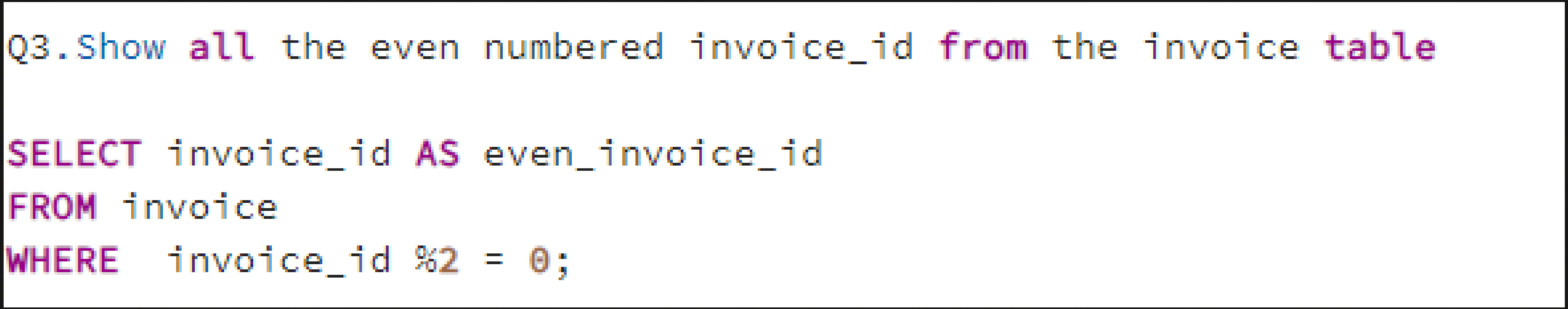
	employee_id [PK] character varying (50)	full_name text	title character varying (50)	levels character varying (10)
1	9	Mohan Madan ...	Senior General Manager	L7



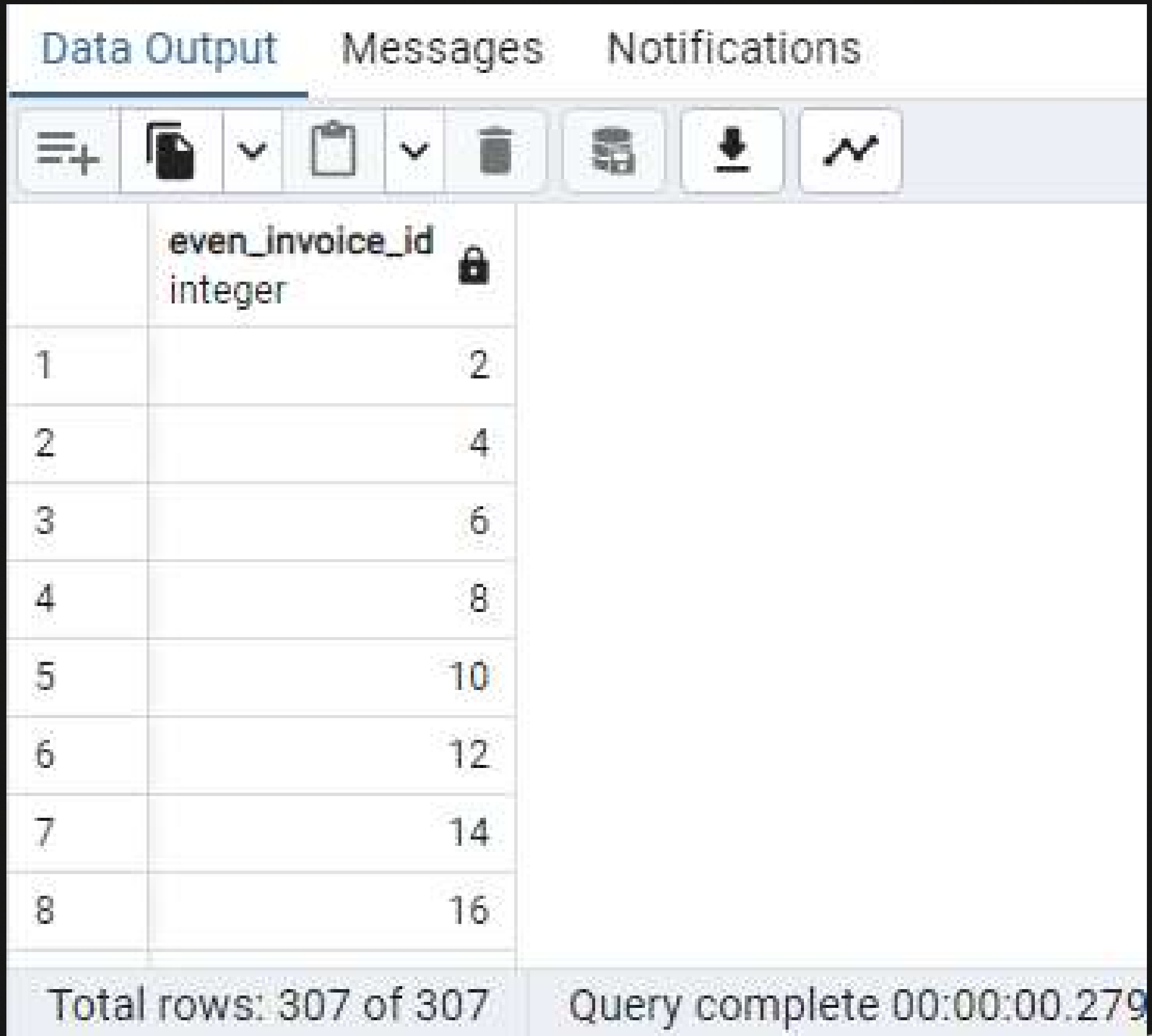
```
Q2. Show all the invoice_id, billing_address,billing_city of all invoice which
    are not from 'Canada', 'Poland', 'France'.

SELECT invoice_id,billing_address,
billing_city FROM invoice
WHERE billing_country NOT IN('Canada','Poland','France')
```



invoice_id [PK] integer	billing_address character varying (120)	billing_city character varying (30
1	627 Broadway	New York
4	627 Broadway	New York
5	1033 N Park Ave	Tucson
8	3,Raj Bhavan Road	Bangalore
9	627 Broadway	New York
11	Barbarossastraße 19	Berlin
13	Rua dos Campeões Europeus de Viena, 4350	Porto
14	319 N. Frances Street	Madison

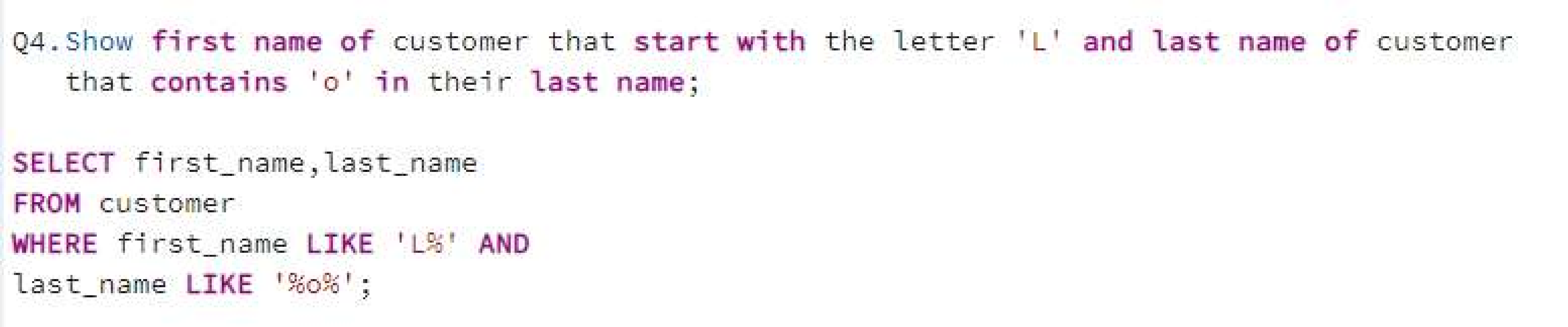


```
Q3.Show all the even numbered invoice_id from the invoice table

SELECT invoice_id AS even_invoice_id
FROM invoice
WHERE  invoice_id %2 = 0;
```

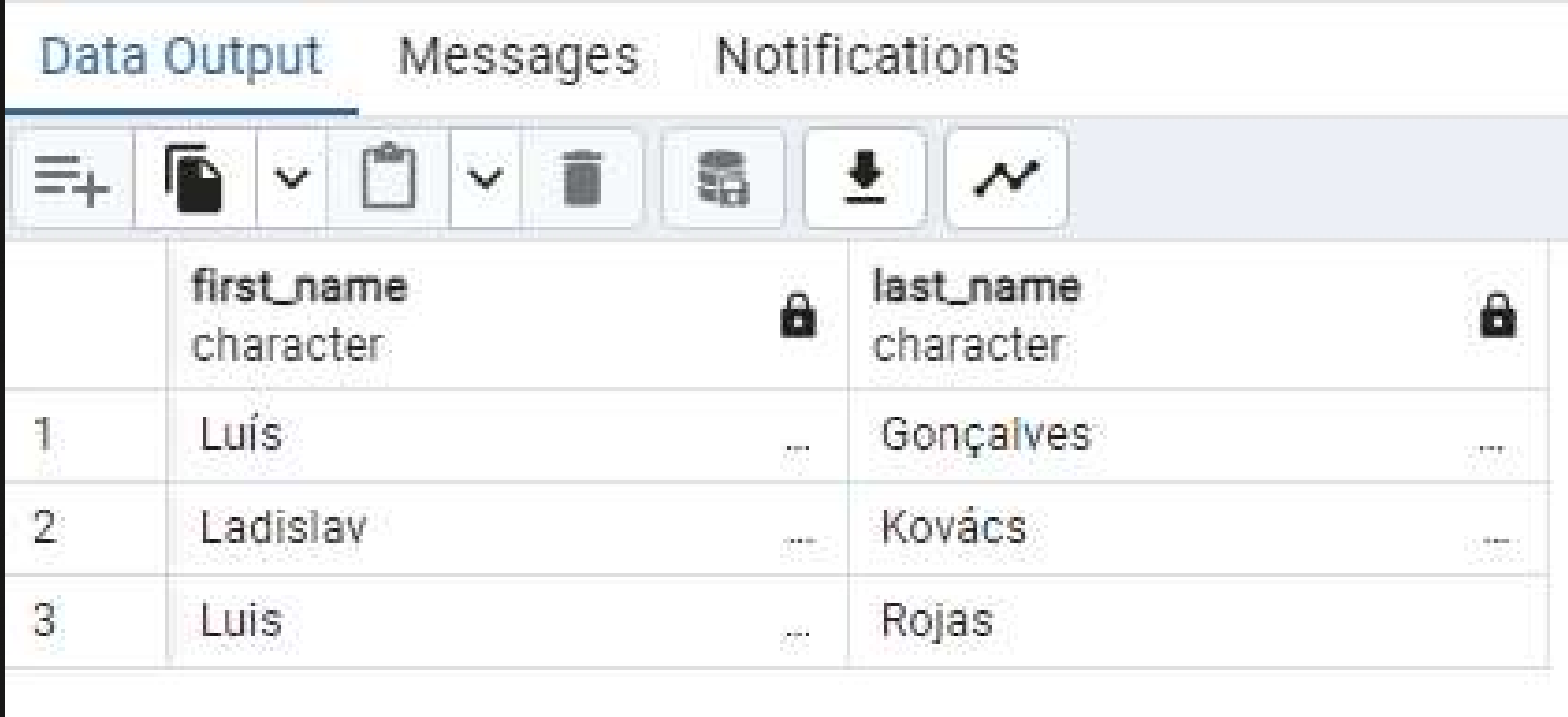


	even_invoice_id integer
1	2
2	4
3	6
4	8
5	10
6	12
7	14
8	16

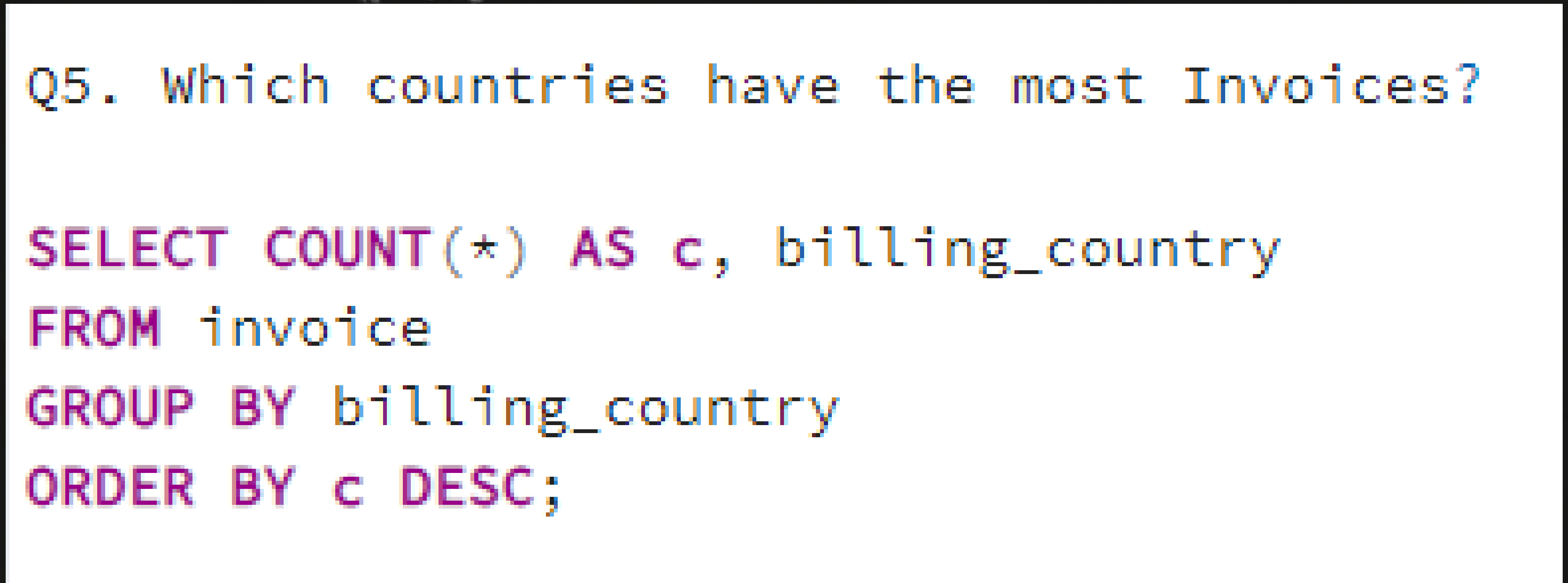
Total rows: 307 of 307   Query complete 00:00:00.279



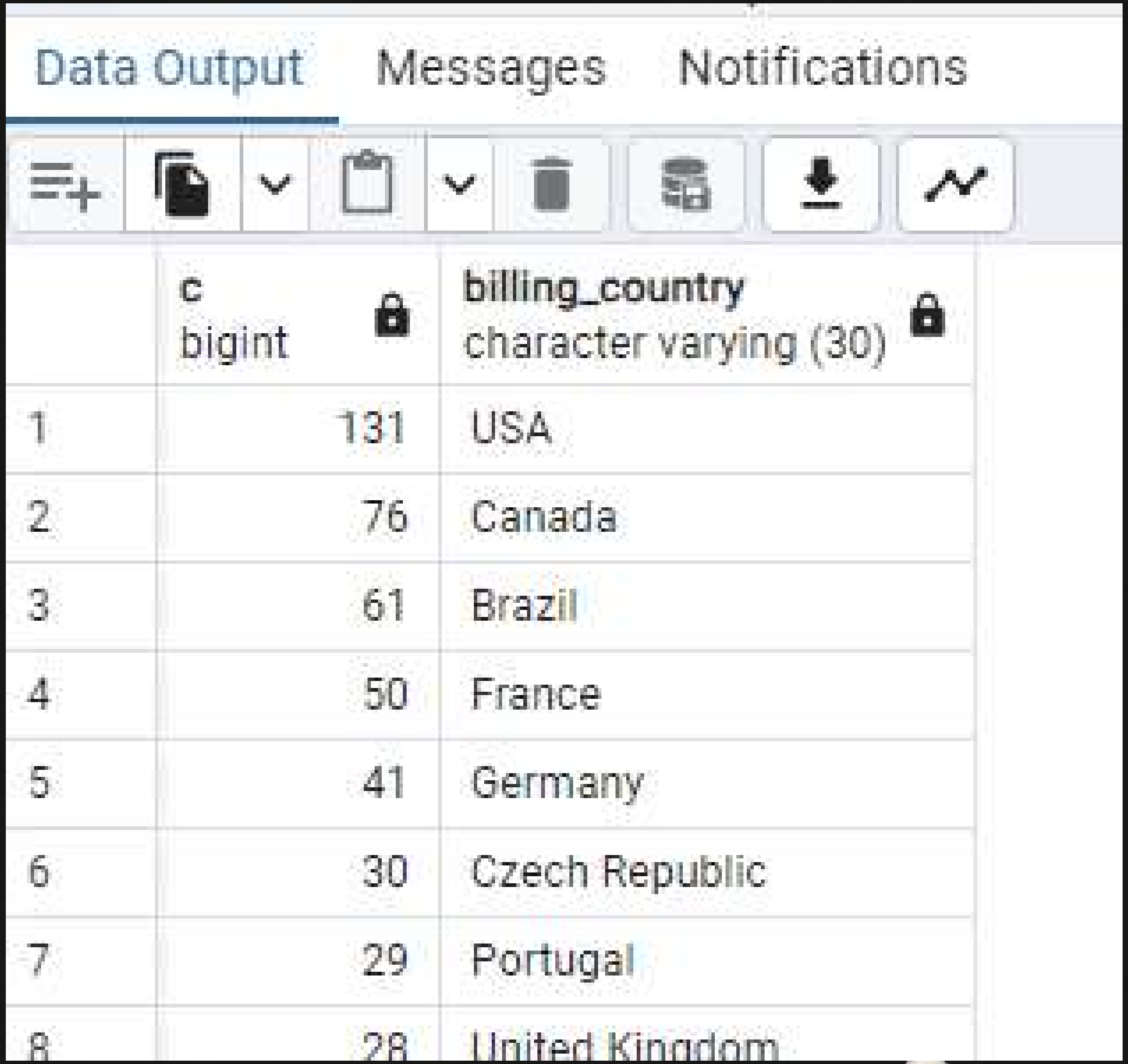
```
Q4.Show first name of customer that start with the letter 'L' and last name of customer
   that contains 'o' in their last name;

SELECT first_name,last_name
FROM customer
WHERE first_name LIKE 'L%' AND
last_name LIKE '%o%';
```



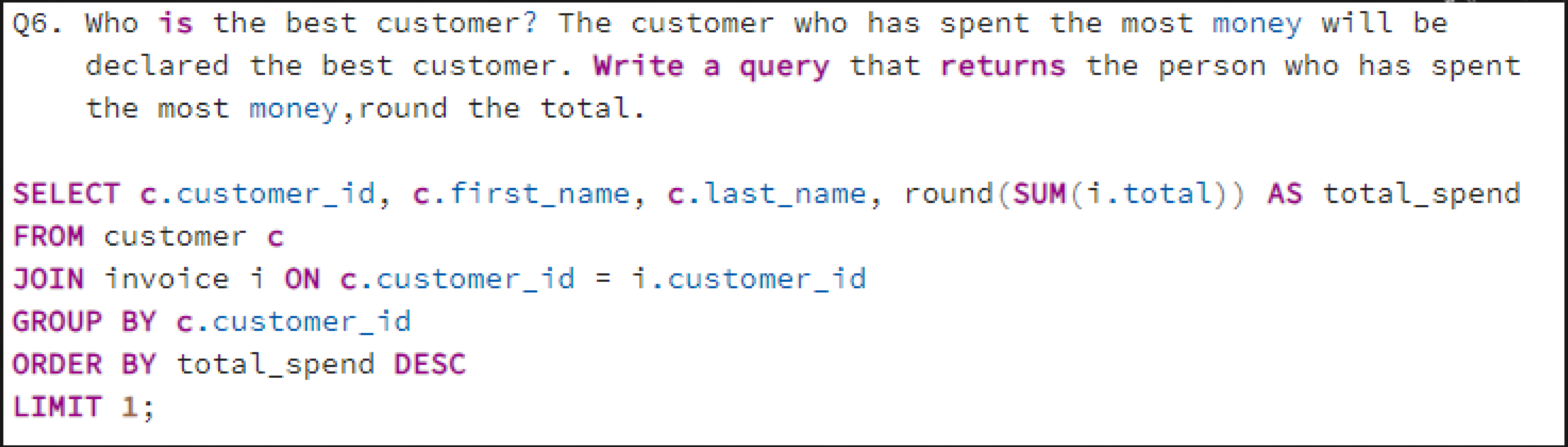
	first_name character	last_name character
1	Luís	Gonçalves
2	Ladislav	Kovács
3	Luis	Rojas



```
Q5. Which countries have the most Invoices?

SELECT COUNT(*) AS c, billing_country
FROM invoice
GROUP BY billing_country
ORDER BY c DESC;
```

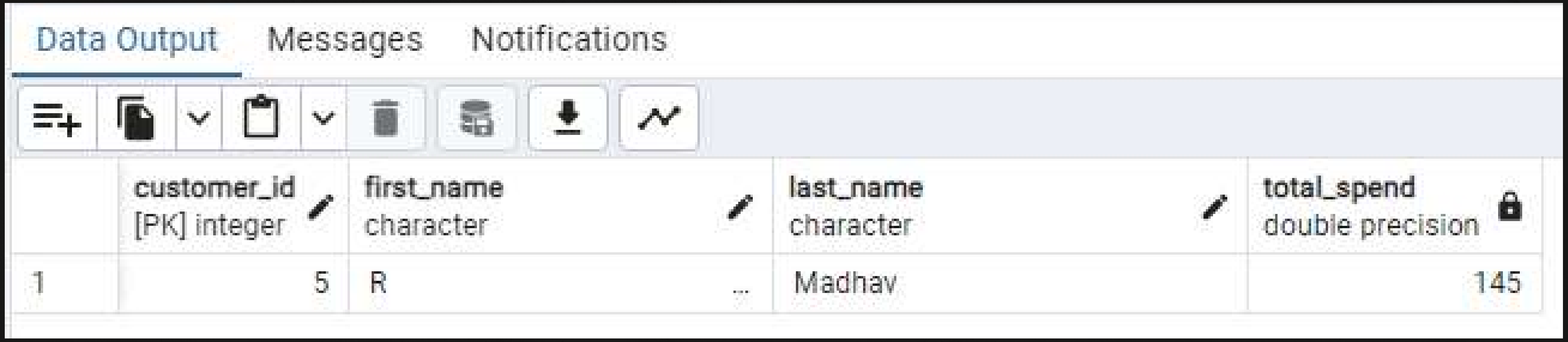


	c bigint	billing_country character varying (30)
1	131	USA
2	76	Canada
3	61	Brazil
4	50	France
5	41	Germany
6	30	Czech Republic
7	29	Portugal
8	28	United Kingdom



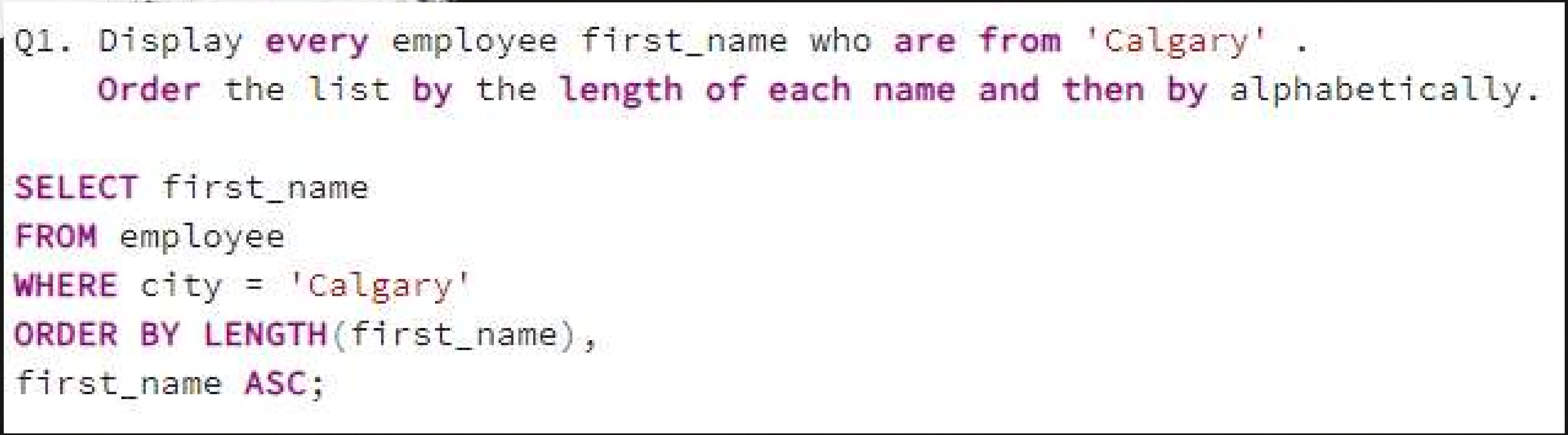
Q6. Who is the best customer? The customer who has spent the most money will be declared the best customer. Write a query that returns the person who has spent the most money,round the total.

```
SELECT c.customer_id, c.first_name, c.last_name, round(SUM(i.total)) AS total_spend
FROM customer c
JOIN invoice i ON c.customer_id = i.customer_id
GROUP BY c.customer_id
ORDER BY total_spend DESC
LIMIT 1;
```

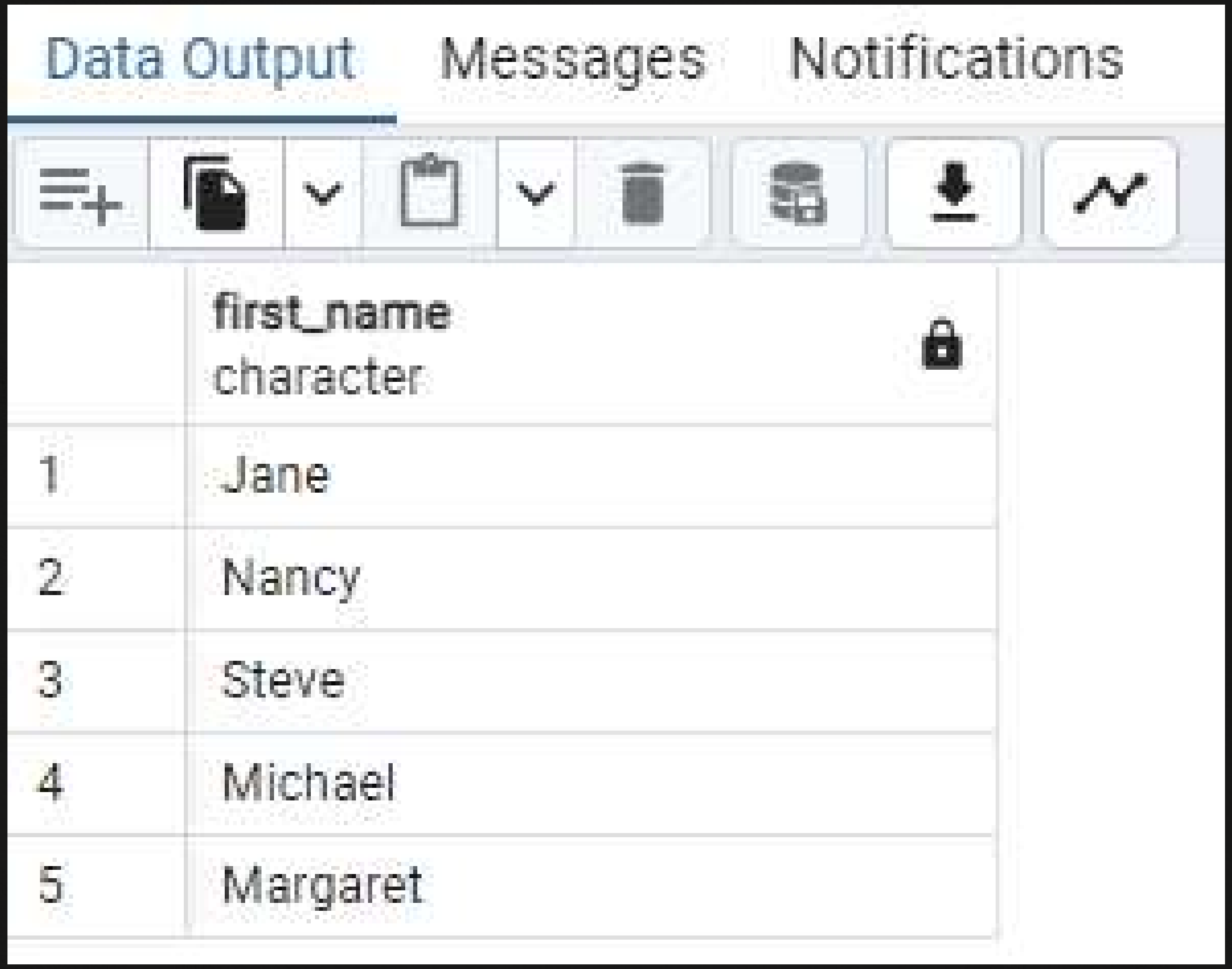


	customer_id [PK] integer	first_name character	last_name character	total_spend double precision
1	5	R ...	Madhav	145

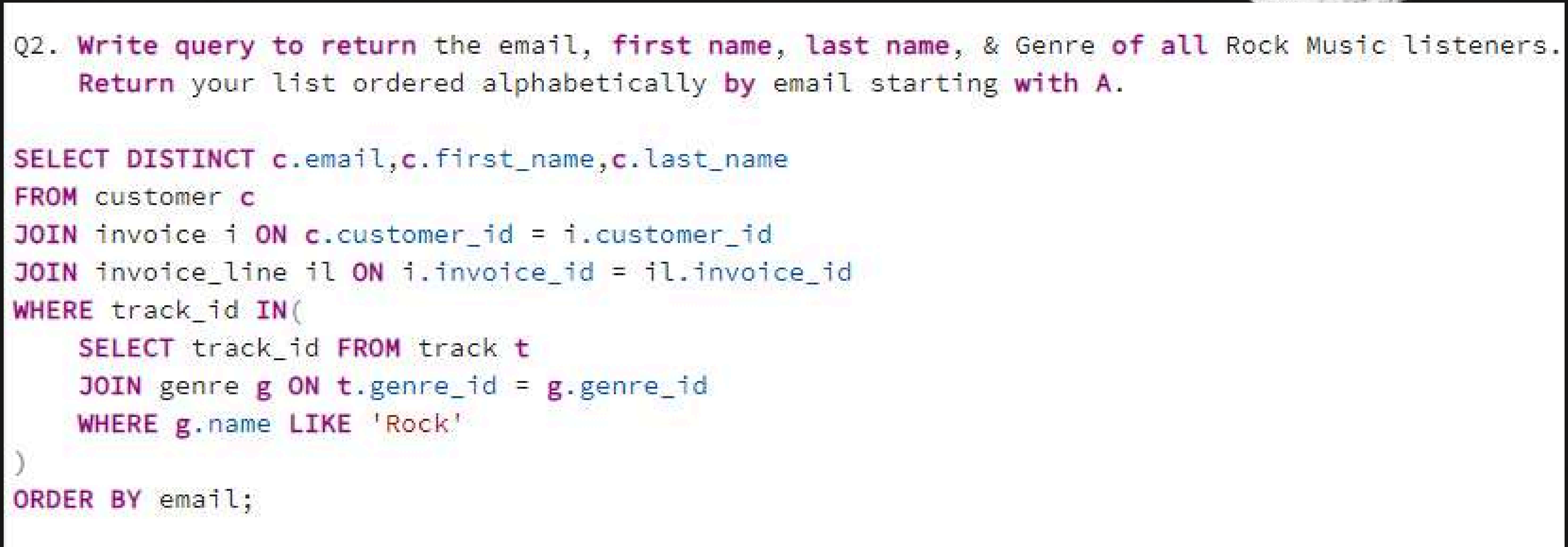
QUESTION SET 2 - MEDIUM



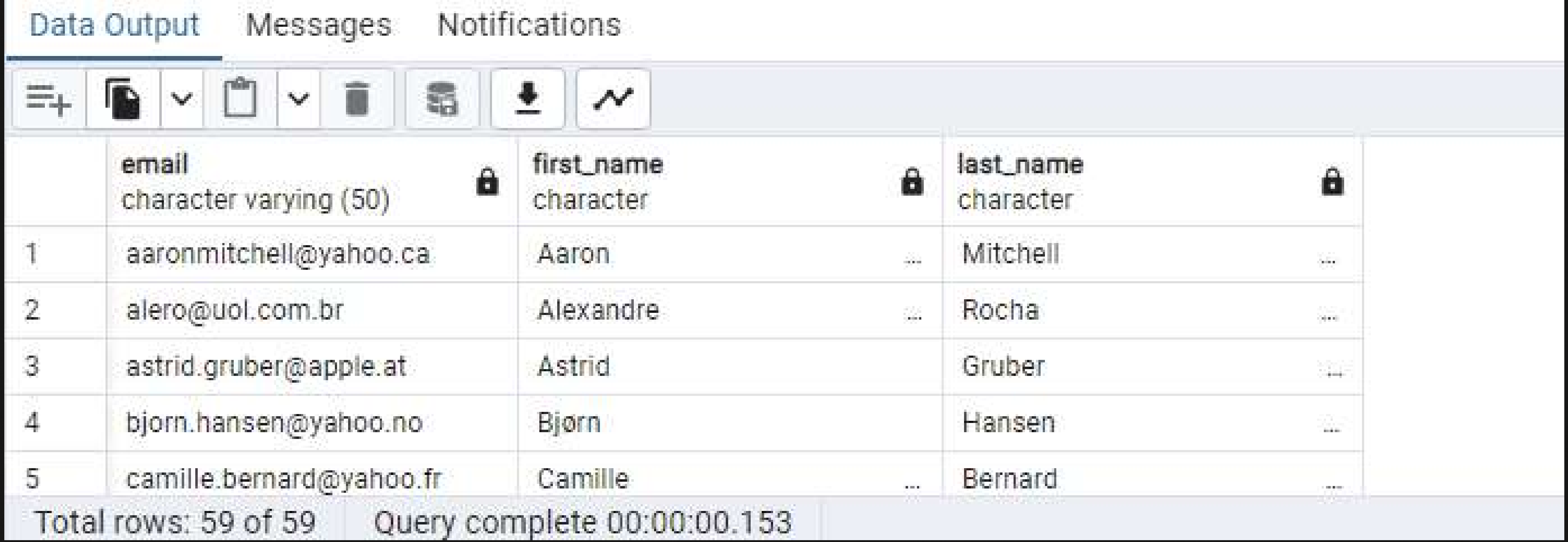
```
Q1. Display every employee first_name who are from 'Calgary' .
    Order the list by the length of each name and then by alphabetically.

SELECT first_name
FROM employee
WHERE city = 'Calgary'
ORDER BY LENGTH(first_name),
first_name ASC;
```



```
   first_name
   character
1  Jane
2  Nancy
3  Steve
4  Michael
5  Margaret
```

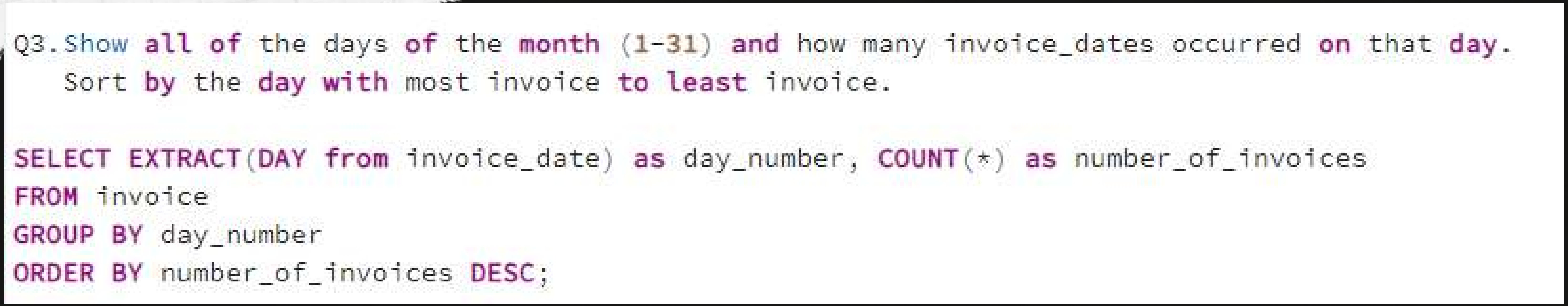


```
Q2. Write query to return the email, first name, last name, & Genre of all Rock Music listeners.
    Return your list ordered alphabetically by email starting with A.

SELECT DISTINCT c.email,c.first_name,c.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;
```

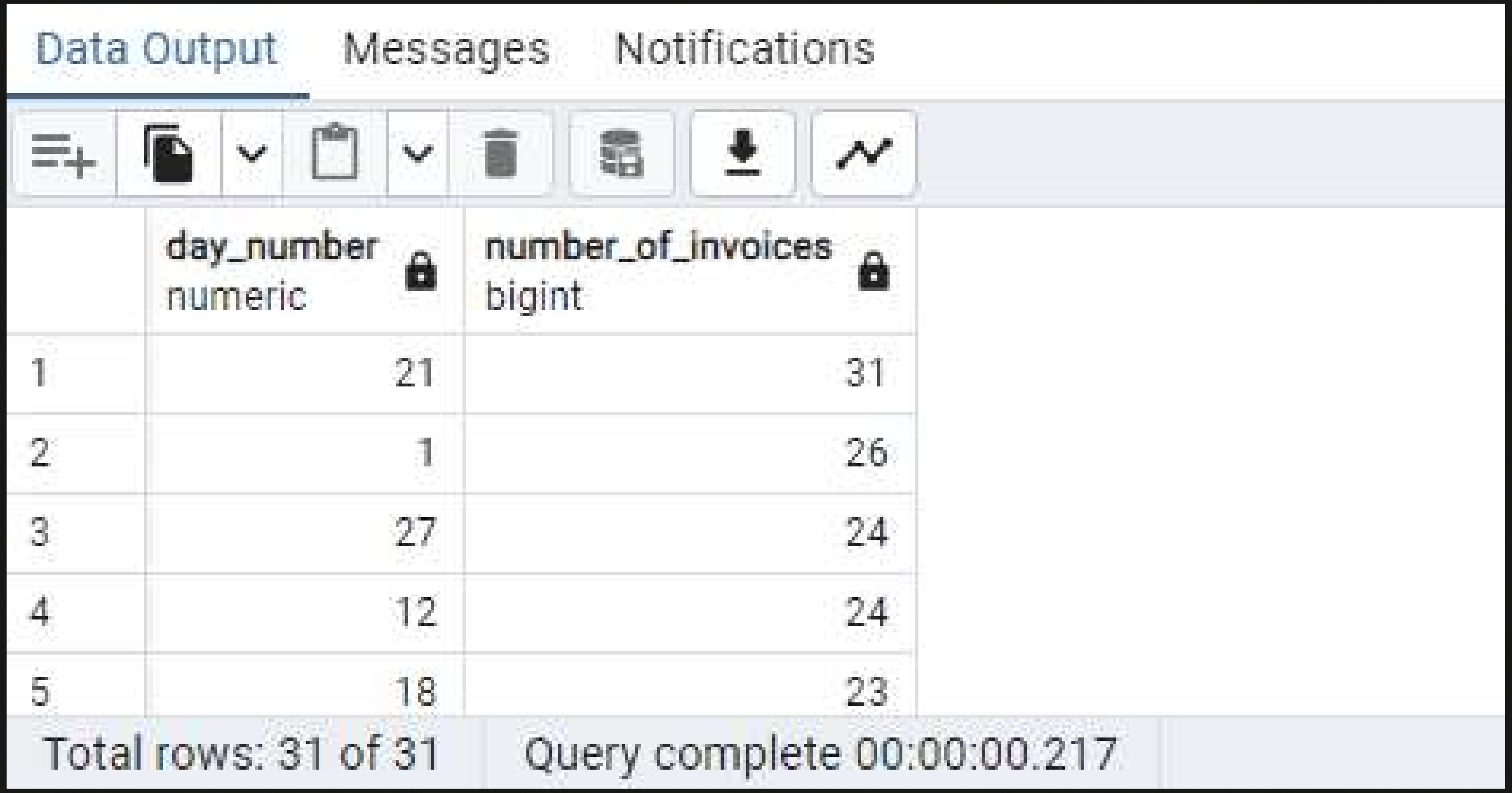


	email character varying (50)	first_name character	last_name character
1	aaronmitchell@yahoo.ca	Aaron	Mitchell
2	alero@uol.com.br	Alexandre	Rocha
3	astrid.gruber@apple.at	Astrid	Gruber
4	bjorn.hansen@yahoo.no	Bjørn	Hansen
5	camille.bernard@yahoo.fr	Camille	Bernard

Total rows: 59 of 59   Query complete 00:00:00.153

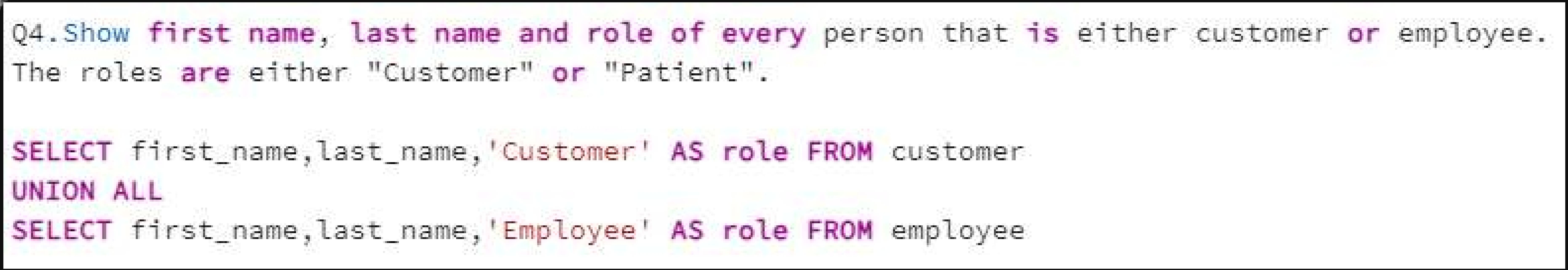


```
Q3.Show all of the days of the month (1-31) and how many invoice_dates occurred on that day.
   Sort by the day with most invoice to least invoice.

SELECT EXTRACT(DAY from invoice_date) as day_number, COUNT(*) as number_of_invoices
FROM invoice
GROUP BY day_number
ORDER BY number_of_invoices DESC;
```

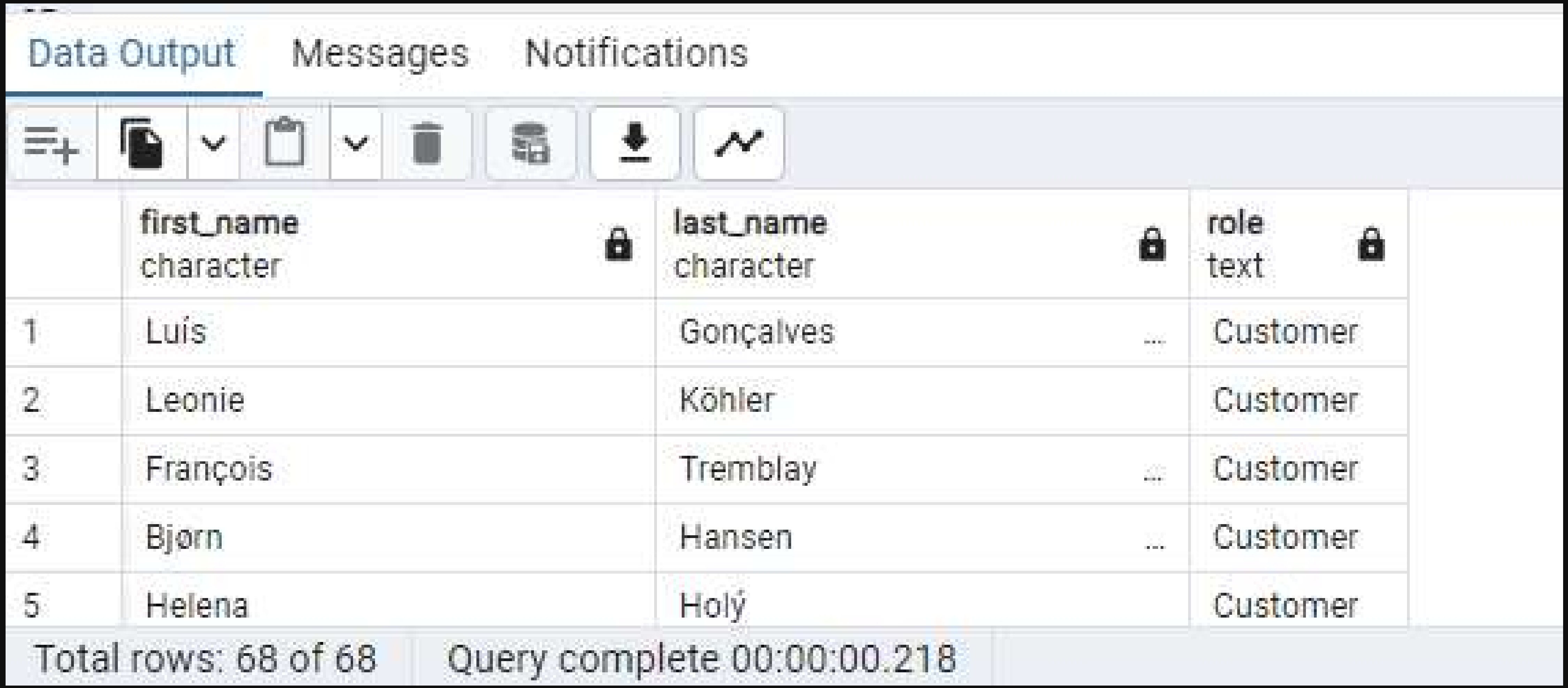


	day_number numeric	number_of_invoices bigint
1	21	31
2	1	26
3	27	24
4	12	24
5	18	23

Total rows: 31 of 31 Query complete 00:00:00.217

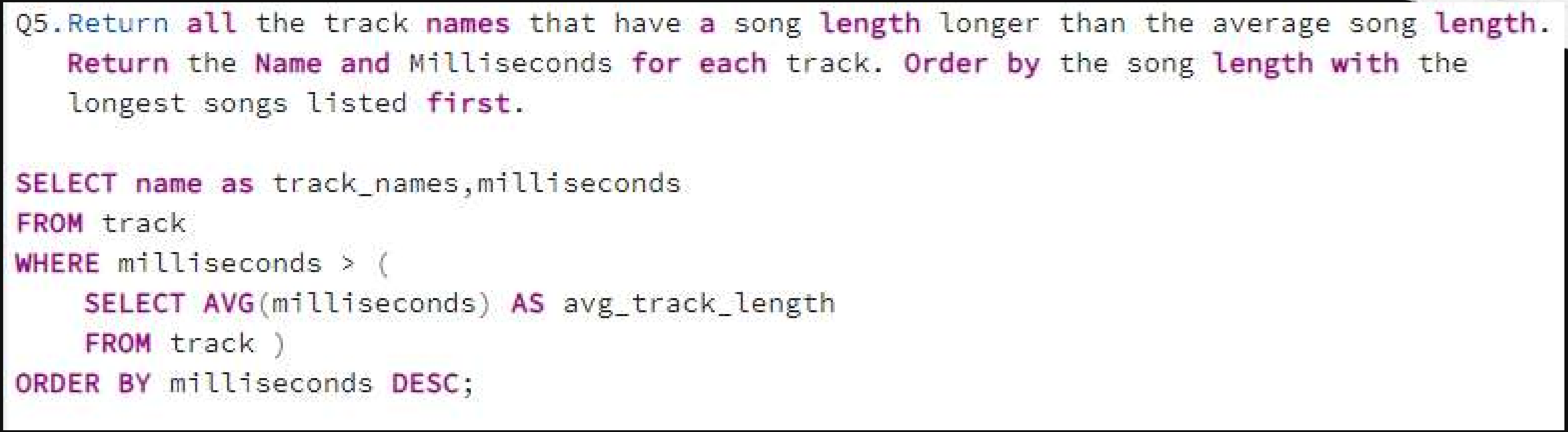


```
Q4.Show first name, last name and role of every person that is either customer or employee.
The roles are either "Customer" or "Patient".

SELECT first_name,last_name,'Customer' AS role FROM customer
UNION ALL
SELECT first_name,last_name,'Employee' AS role FROM employee
```

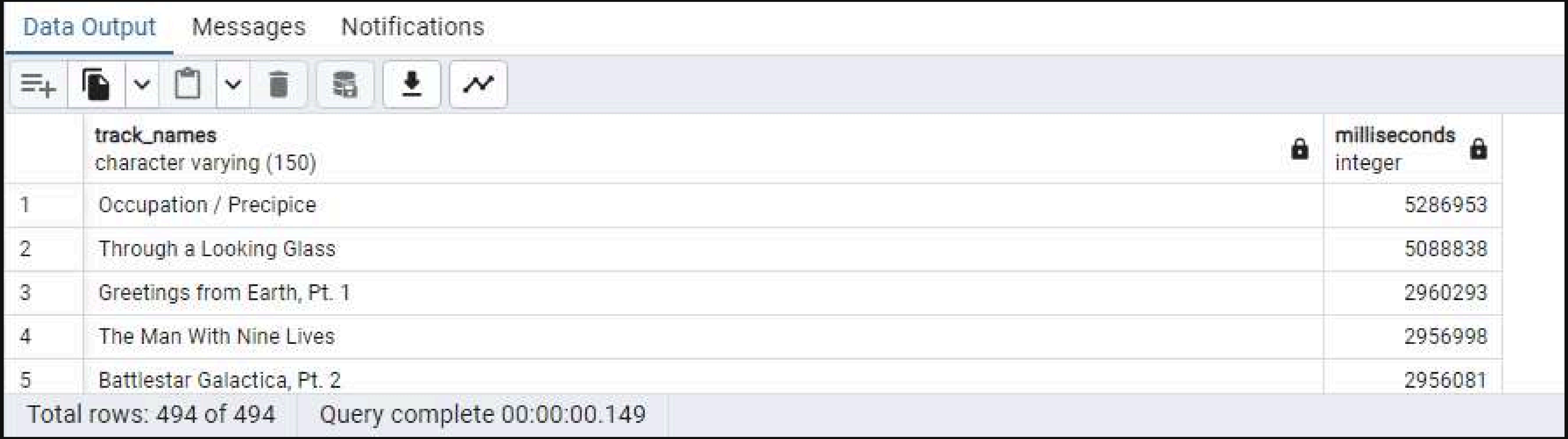


	first_name character	last_name character	role text
1	Luís	Gonçalves …	Customer
2	Leonie	Köhler	Customer
3	François	Tremblay …	Customer
4	Bjørn	Hansen …	Customer
5	Helena	Holý	Customer

Total rows: 68 of 68 Query complete 00:00:00.218



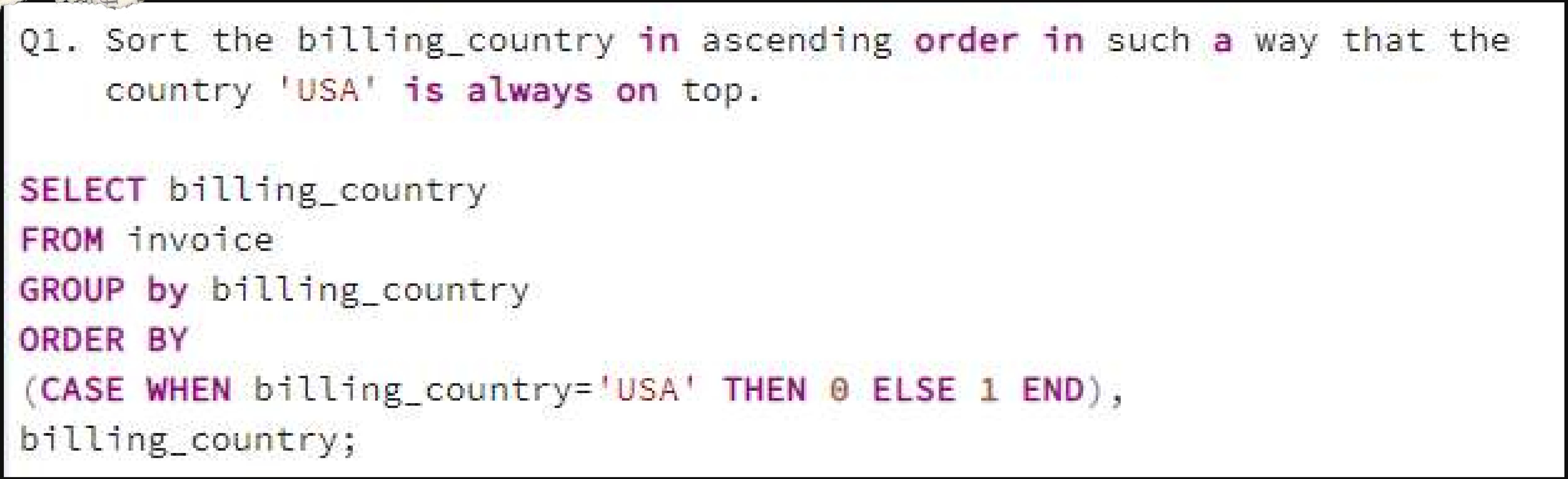
```
Q5.Return all the track names that have a song length longer than the average song length.
   Return the Name and Milliseconds for each track. Order by the song length with the
   longest songs listed first.

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



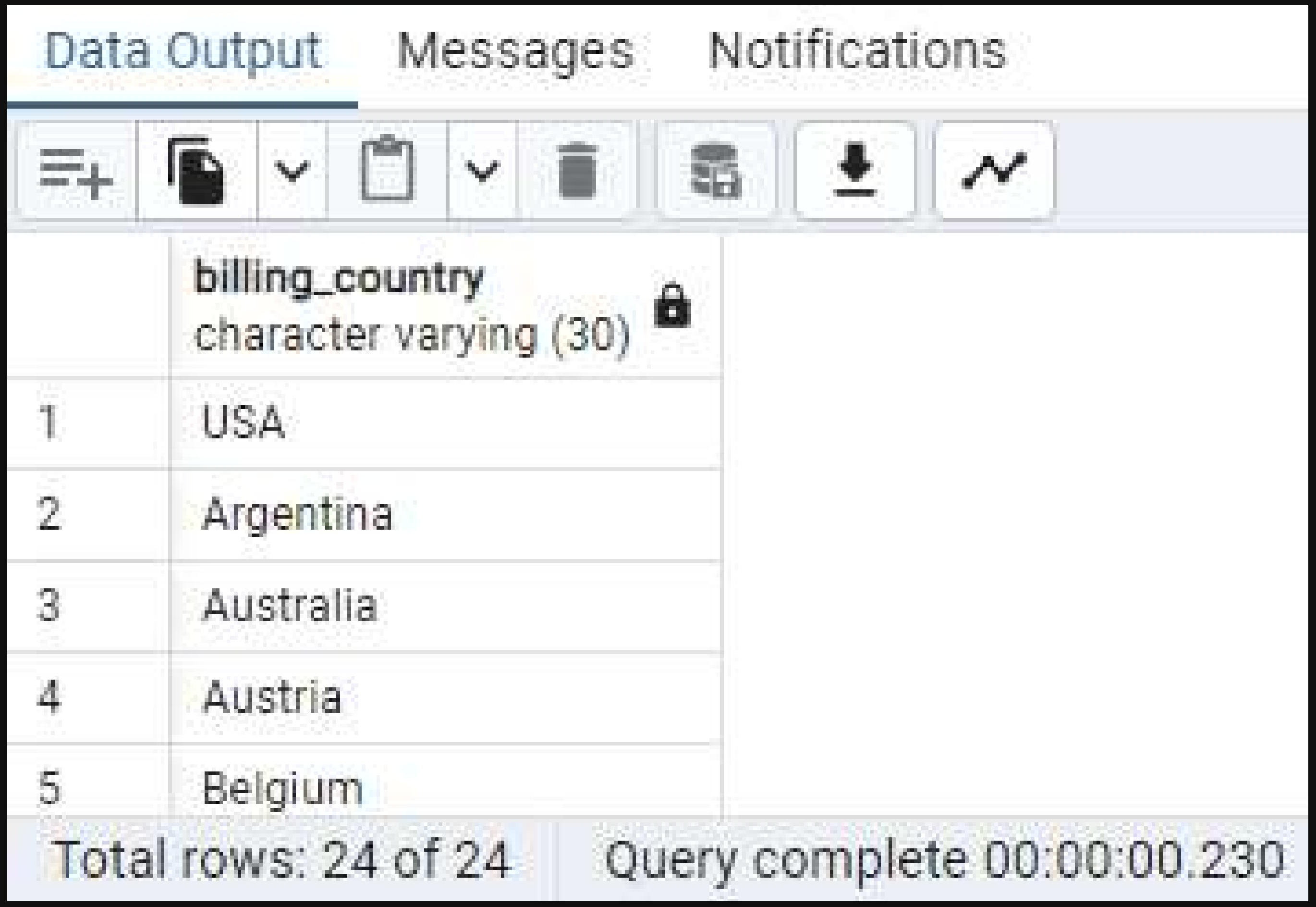
	track_names character varying (150)	milliseconds integer
1	Occupation / Precipice	5286953
2	Through a Looking Glass	5088838
3	Greetings from Earth, Pt. 1	2960293
4	The Man With Nine Lives	2956998
5	Battlestar Galactica, Pt. 2	2956081

Total rows: 494 of 494 Query complete 00:00:00.149

QUESTION SET 3 - HARD

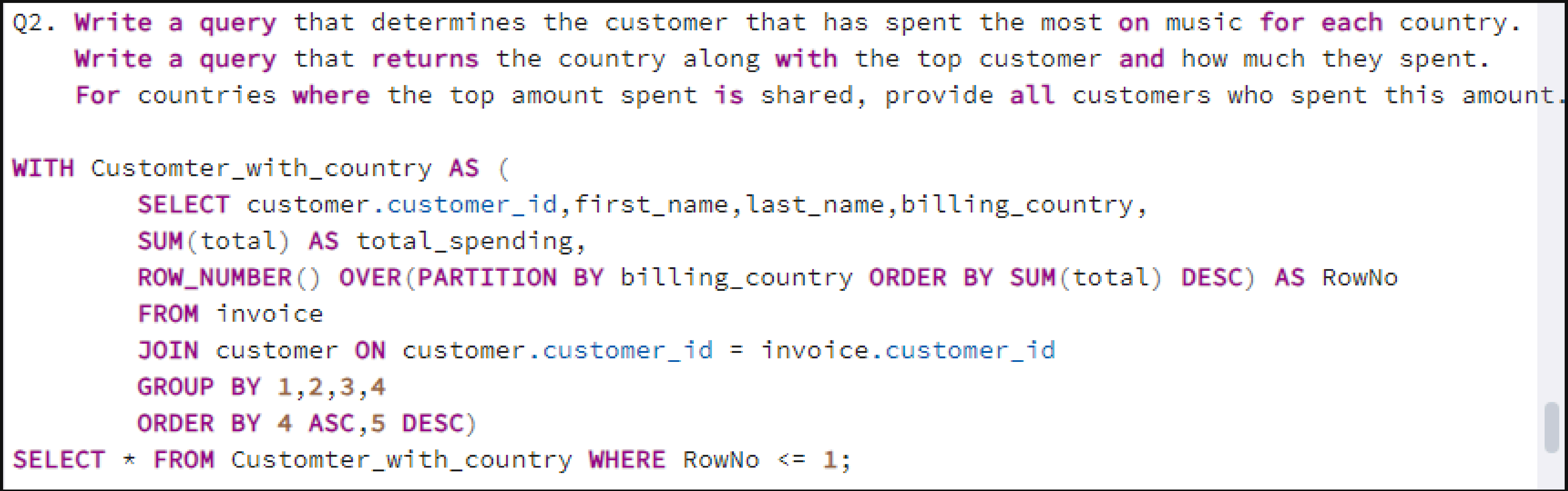


```
Q1. Sort the billing_country in ascending order in such a way that the
    country 'USA' is always on top.

SELECT billing_country
FROM invoice
GROUP by billing_country
ORDER BY
(CASE WHEN billing_country='USA' THEN 0 ELSE 1 END),
billing_country;
```

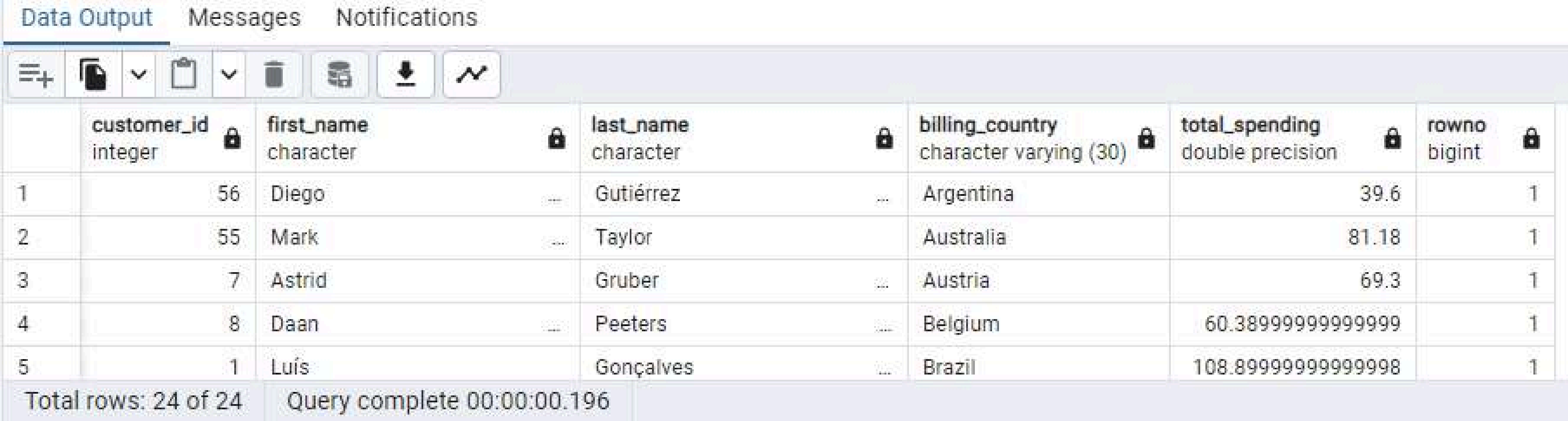


	billing_country character varying (30)
1	USA
2	Argentina
3	Australia
4	Austria
5	Belgium

Total rows: 24 of 24 Query complete 00:00:00.230

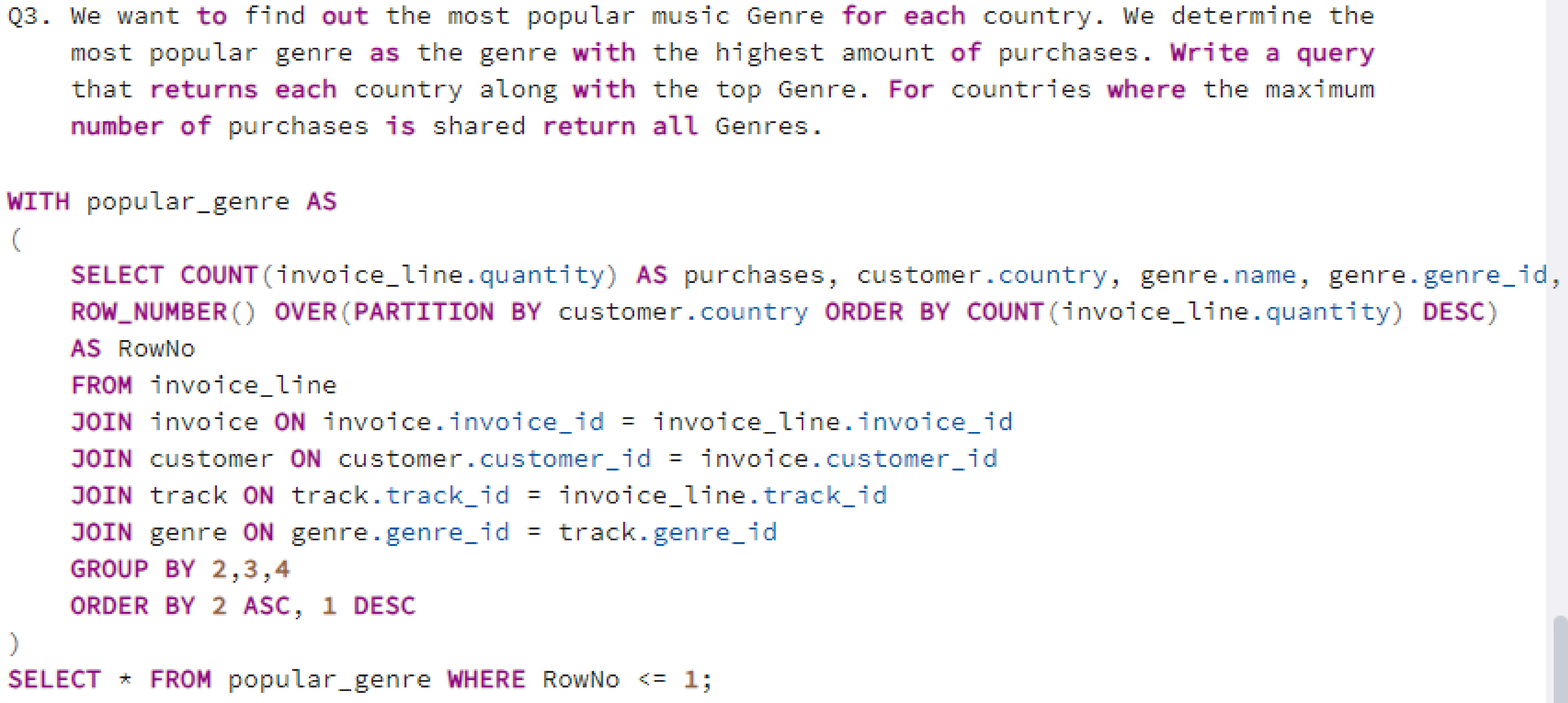


```
Q2. Write a query that determines the customer that has spent the most on music for each country.
    Write a query that returns the country along with the top customer and how much they spent.
    For countries where the top amount spent is shared, provide all customers who spent this amount.

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;
```

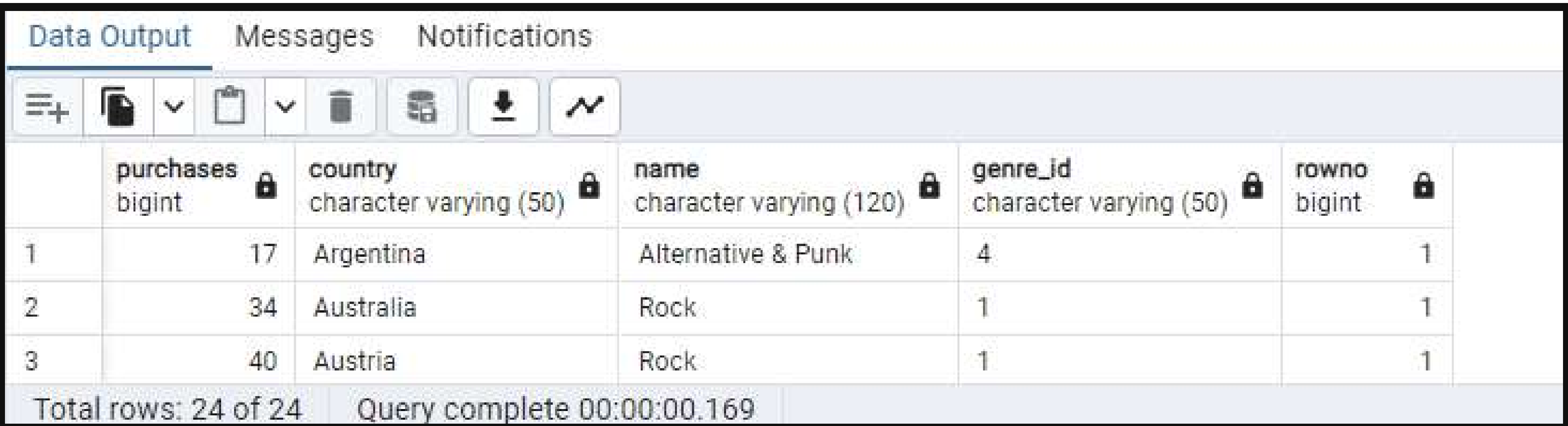


	customer_id integer	first_name character	last_name character	billing_country character varying (30)	total_spending double precision	rowno bigint
1	56	Diego	Gutiérrez	Argentina	39.6	1
2	55	Mark	Taylor	Australia	81.18	1
3	7	Astrid	Gruber	Austria	69.3	1
4	8	Daan	Peeters	Belgium	60.38999999999999	1
5	1	Luís	Gonçalves	Brazil	108.89999999999998	1

Total rows: 24 of 24   Query complete 00:00:00.196



```
Q3. 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
1	17	Argentina	Alternative & Punk	4	1
2	34	Australia	Rock	1	1
3	40	Austria	Rock	1	1

Total rows: 24 of 24 Query complete 00:00:00.169

THANKYOU