SQL DATA OUTPUT

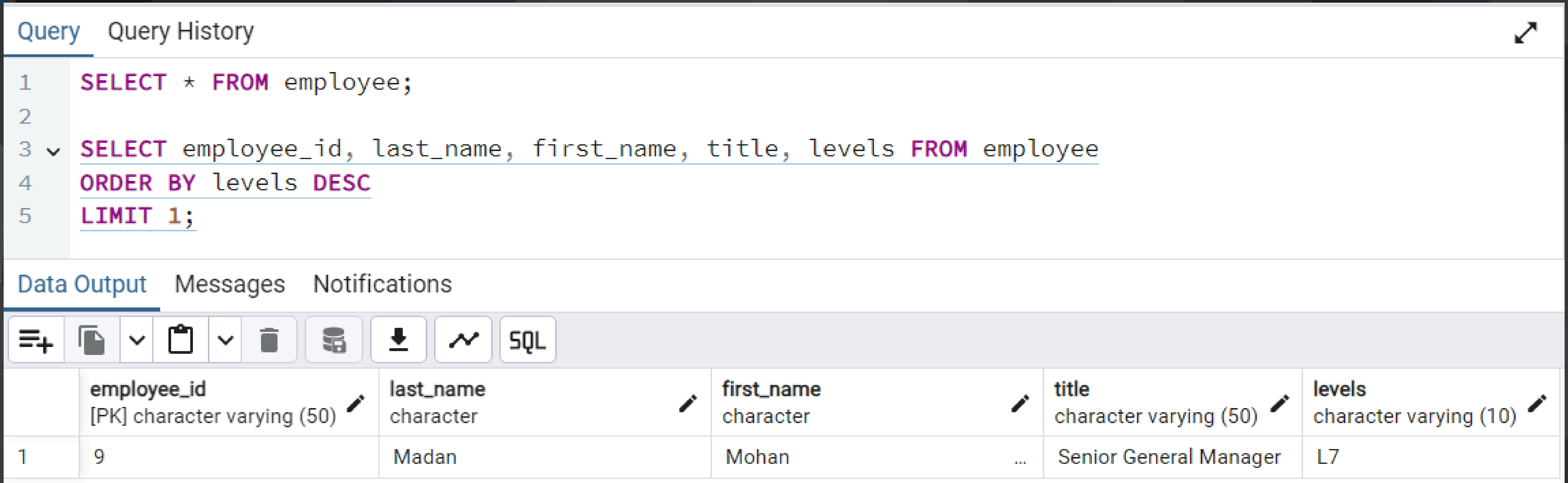
By Jasika Gupta

in Jasika Gupta

jasikagupta04

jasikagupta04@gmail.com

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

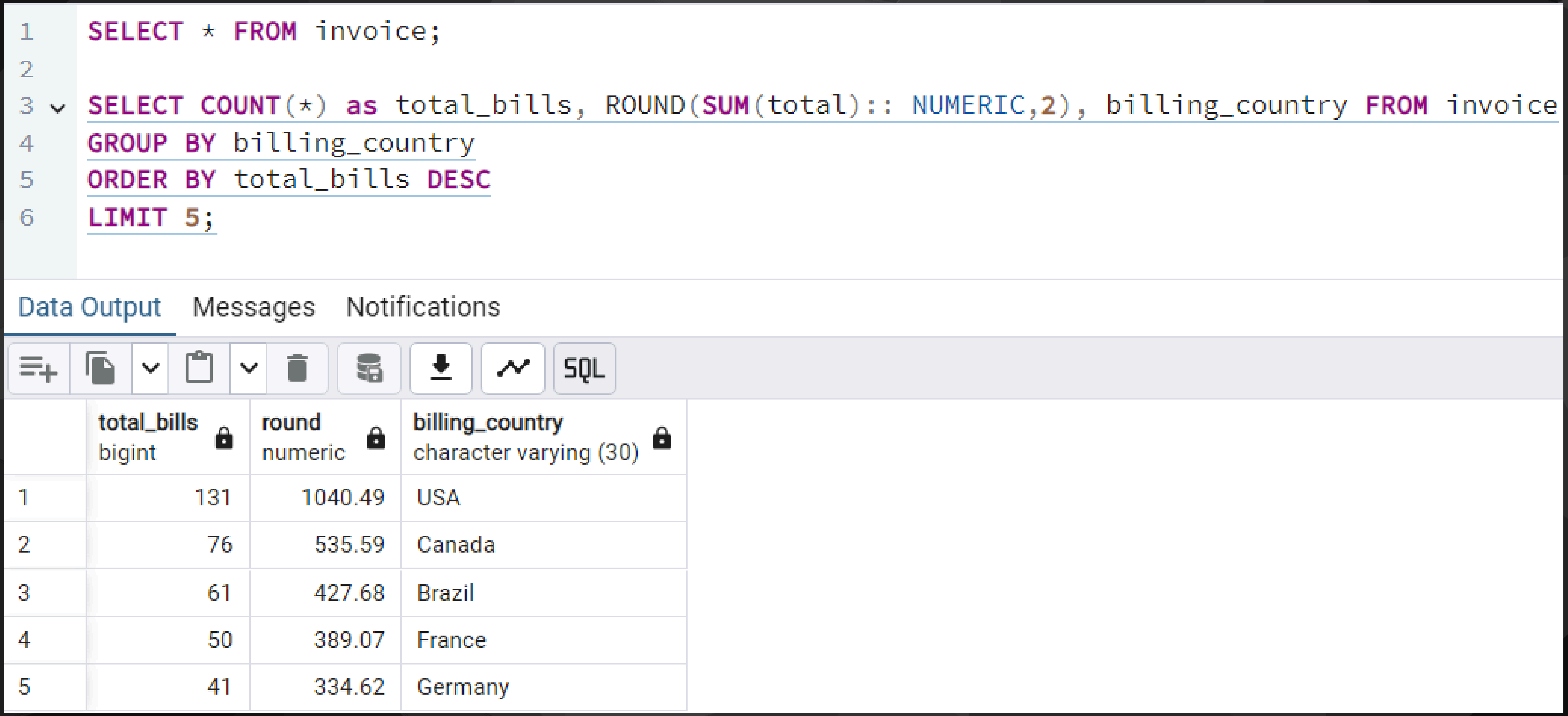


```sql
SELECT * FROM employee;

SELECT employee_id, last_name, first_name, title, levels FROM employee
ORDER BY levels DESC
LIMIT 1;
```

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

Question 2 : Which countries have the most Invoices?

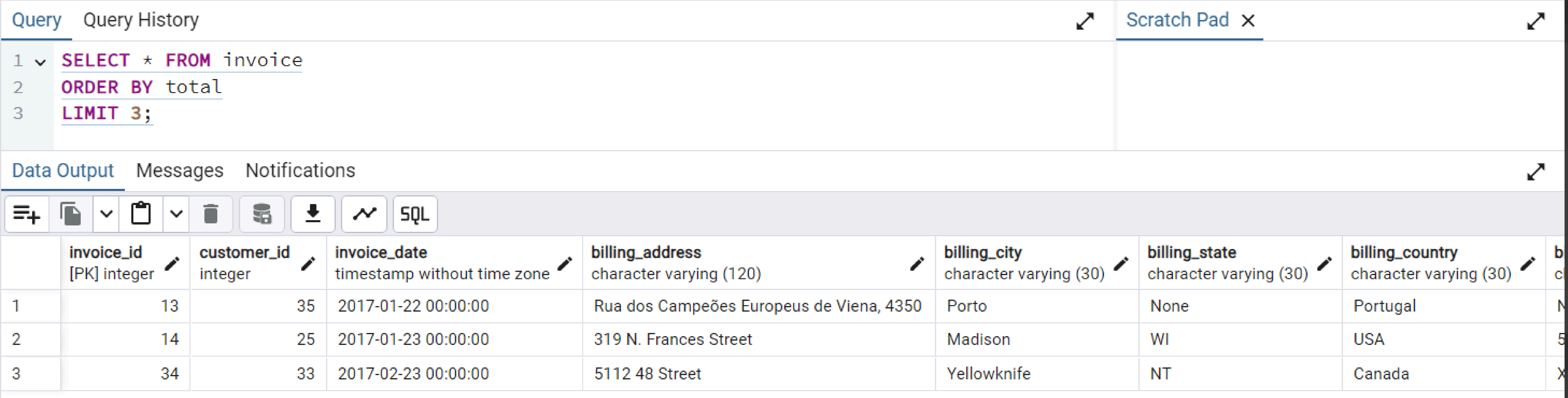


```
SELECT * FROM invoice;

SELECT COUNT(*) as total_bills, ROUND(SUM(total):: NUMERIC,2), billing_country FROM invoice
GROUP BY billing_country
ORDER BY total_bills DESC
LIMIT 5;
```

	total_bills bigint	round numeric	billing_country character varying (30)
1	131	1040.49	USA
2	76	535.59	Canada
3	61	427.68	Brazil
4	50	389.07	France
5	41	334.62	Germany

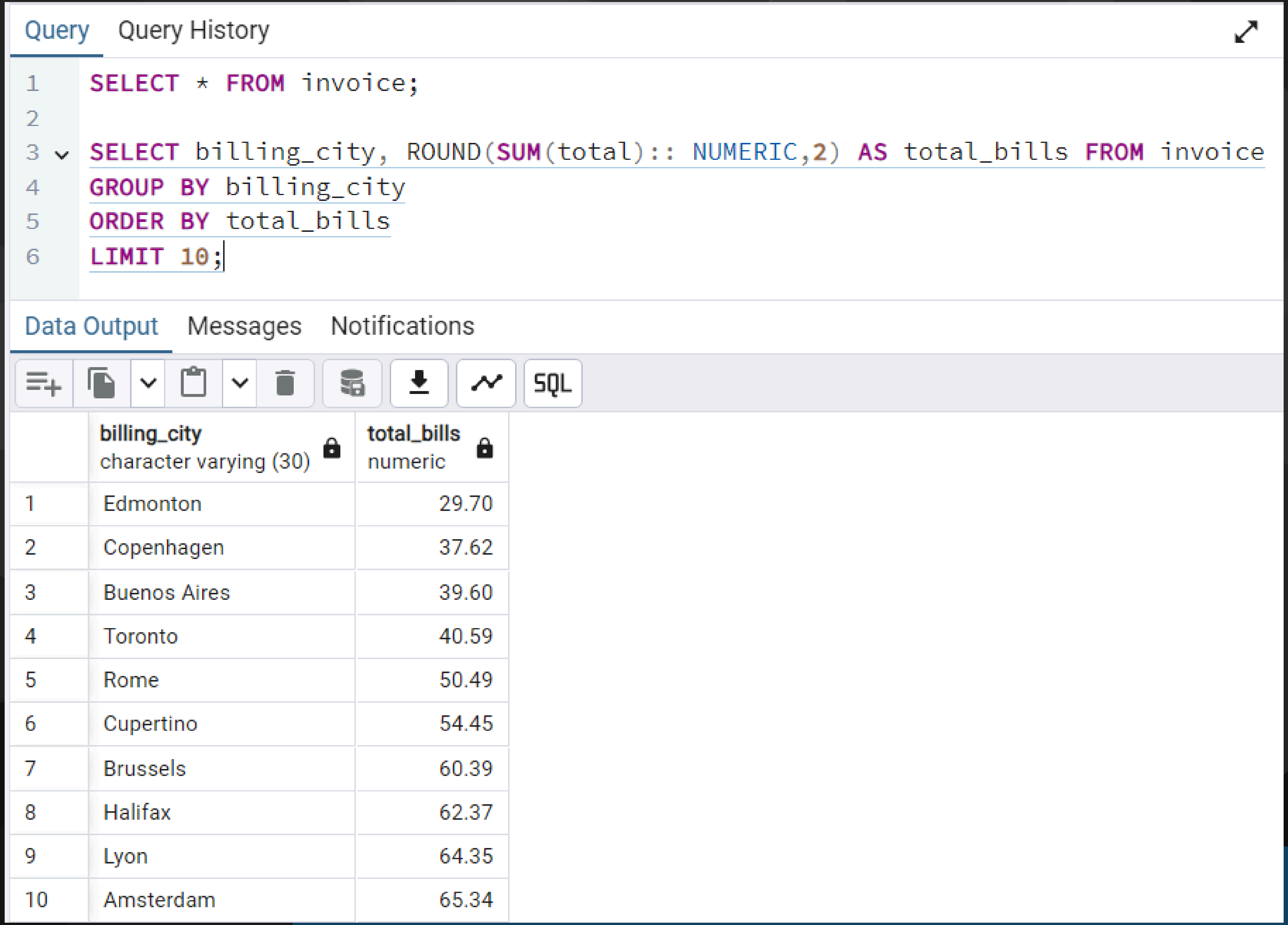
Question 3 : What are top 3 values of total invoice?



```sql
SELECT * FROM invoice
ORDER BY total
LIMIT 3;
```

	invoice_id [PK] integer	customer_id integer	invoice_date timestamp without time zone	billing_address character varying (120)	billing_city character varying (30)	billing_state character varying (30)	billing_country character varying (30)	b c
1	13	35	2017-01-22 00:00:00	Rua dos Campeões Europeus de Viena, 4350	Porto	None	Portugal	N
2	14	25	2017-01-23 00:00:00	319 N. Frances Street	Madison	WI	USA	5
3	34	33	2017-02-23 00:00:00	5112 48 Street	Yellowknife	NT	Canada	X

Question 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.

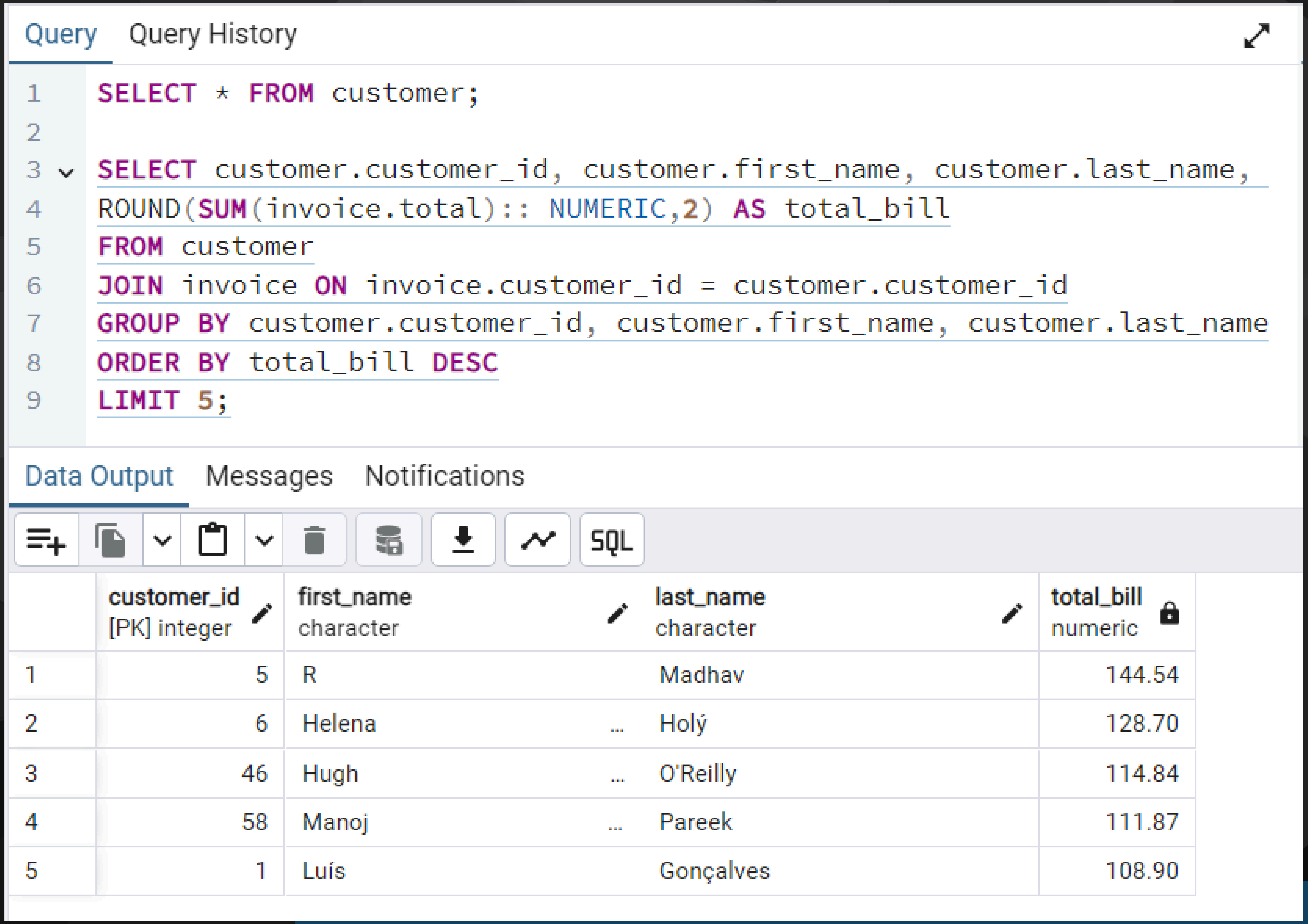


```sql
SELECT * FROM invoice;

SELECT billing_city, ROUND(SUM(total):: NUMERIC,2) AS total_bills FROM invoice
GROUP BY billing_city
ORDER BY total_bills
LIMIT 10;
```

	billing_city character varying (30)	total_bills numeric
1	Edmonton	29.70
2	Copenhagen	37.62
3	Buenos Aires	39.60
4	Toronto	40.59
5	Rome	50.49
6	Cupertino	54.45
7	Brussels	60.39
8	Halifax	62.37
9	Lyon	64.35
10	Amsterdam	65.34

Question 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.

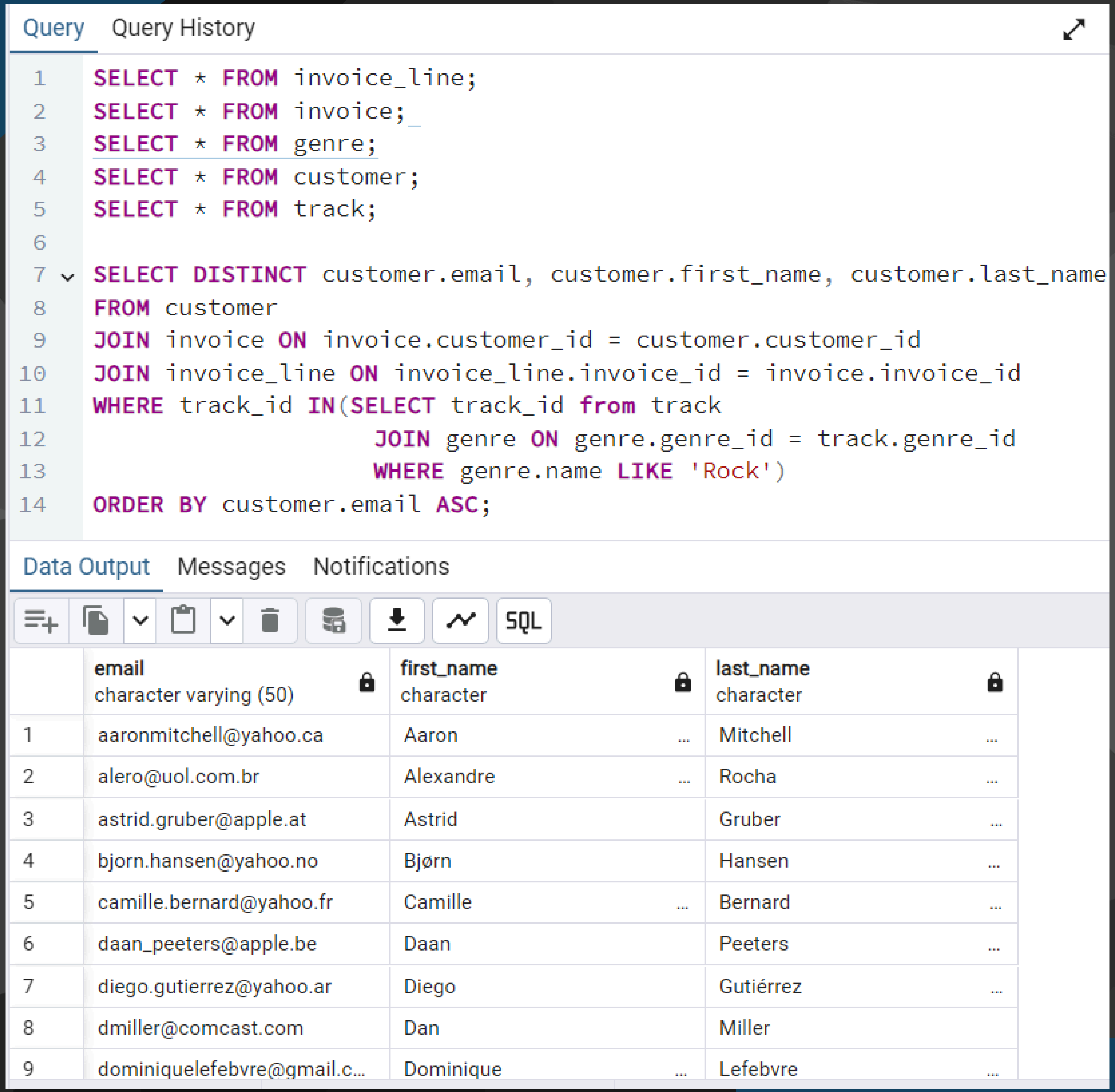


```
SELECT * FROM customer;

SELECT customer.customer_id, customer.first_name, customer.last_name,
ROUND(SUM(invoice.total):: NUMERIC,2) AS total_bill
FROM customer
JOIN invoice ON invoice.customer_id = customer.customer_id
GROUP BY customer.customer_id, customer.first_name, customer.last_name
ORDER BY total_bill DESC
LIMIT 5;
```

	customer_id [PK] integer	first_name character	last_name character	total_bill numeric
1	5	R	Madhav	144.54
2	6	Helena ...	Holý	128.70
3	46	Hugh ...	O'Reilly	114.84
4	58	Manoj ...	Pareek	111.87
5	1	Luís	Gonçalves	108.90

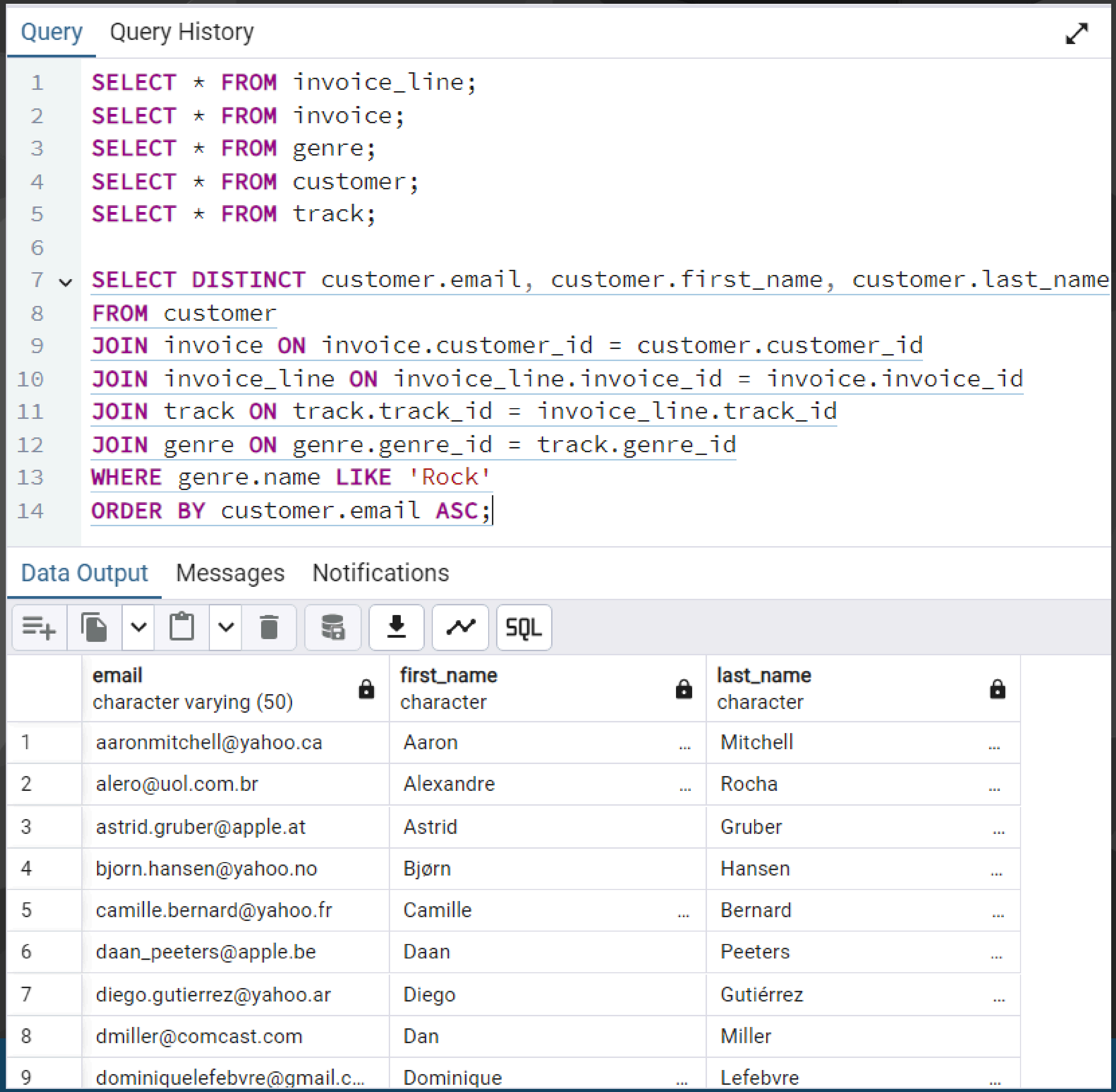
Question 6 : Write query to return the email, first name, last name, & Genre of all Rock Music listeners. Return your list ordered alphabetically by email starting with A.



```sql
SELECT * FROM invoice_line;
SELECT * FROM invoice;
SELECT * FROM genre;
SELECT * FROM customer;
SELECT * FROM track;

SELECT DISTINCT customer.email, customer.first_name, customer.last_name
FROM customer
JOIN invoice ON invoice.customer_id = customer.customer_id
JOIN invoice_line ON invoice_line.invoice_id = invoice.invoice_id
WHERE track_id IN(SELECT track_id from track
                    JOIN genre ON genre.genre_id = track.genre_id
                    WHERE genre.name LIKE 'Rock')
ORDER BY customer.email ASC;
```

	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
7	diego.gutierrez@yahoo.ar	Diego	Gutiérrez
8	dmiller@comcast.com	Dan	Miller
9	dominiquelefebvre@gmail.c...	Dominique	Lefebvre

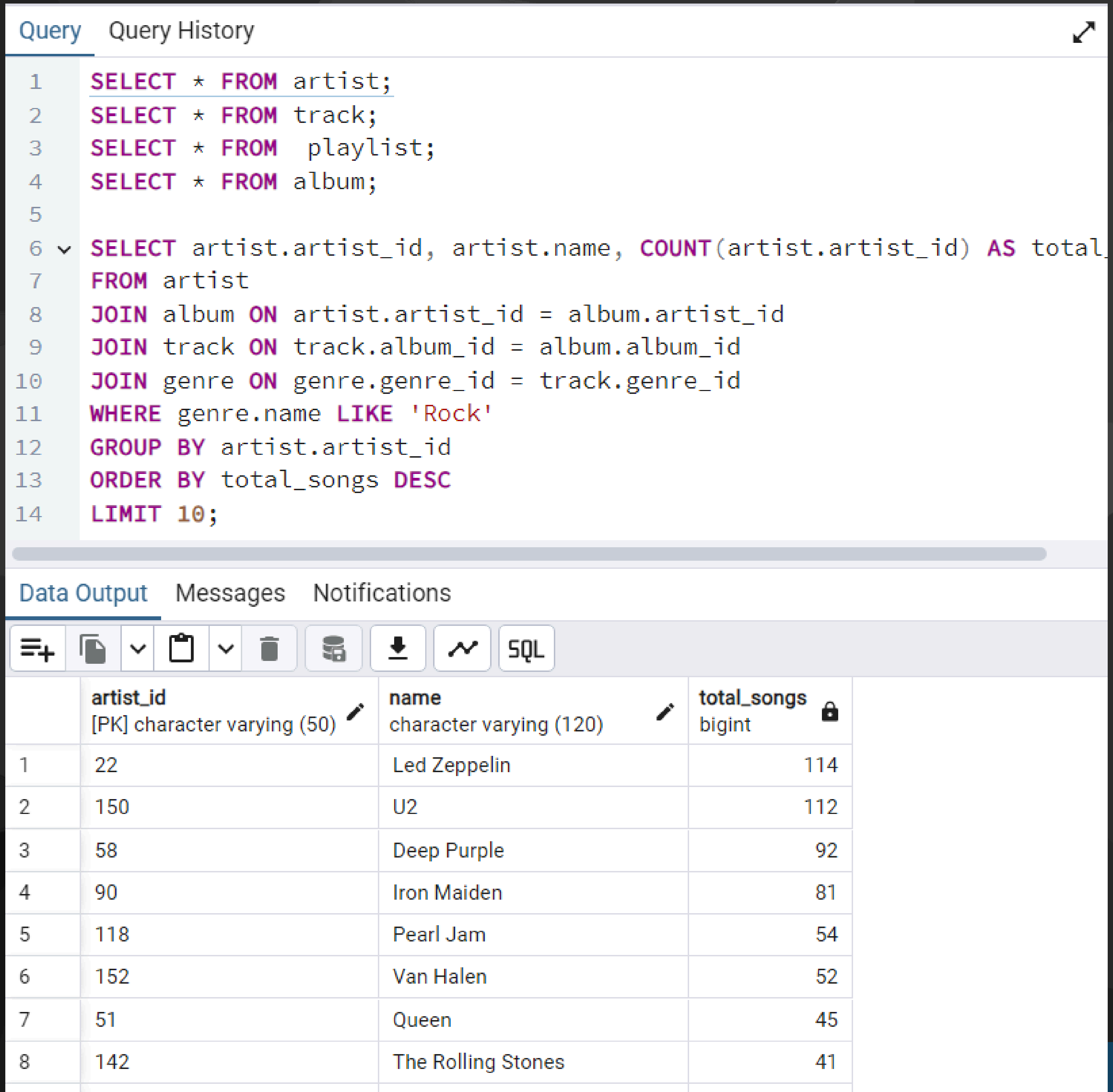


```sql
SELECT * FROM invoice_line;
SELECT * FROM invoice;
SELECT * FROM genre;
SELECT * FROM customer;
SELECT * FROM track;

SELECT DISTINCT customer.email, customer.first_name, customer.last_name
FROM customer
JOIN invoice ON invoice.customer_id = customer.customer_id
JOIN invoice_line ON invoice_line.invoice_id = invoice.invoice_id
JOIN track ON track.track_id = invoice_line.track_id
JOIN genre ON genre.genre_id = track.genre_id
WHERE genre.name LIKE 'Rock'
ORDER BY customer.email ASC;
```

	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
7	diego.gutierrez@yahoo.ar	Diego	Gutiérrez
8	dmiller@comcast.com	Dan	Miller
9	dominiquelefebvre@gmail.c...	Dominique	Lefebvre

Question 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.

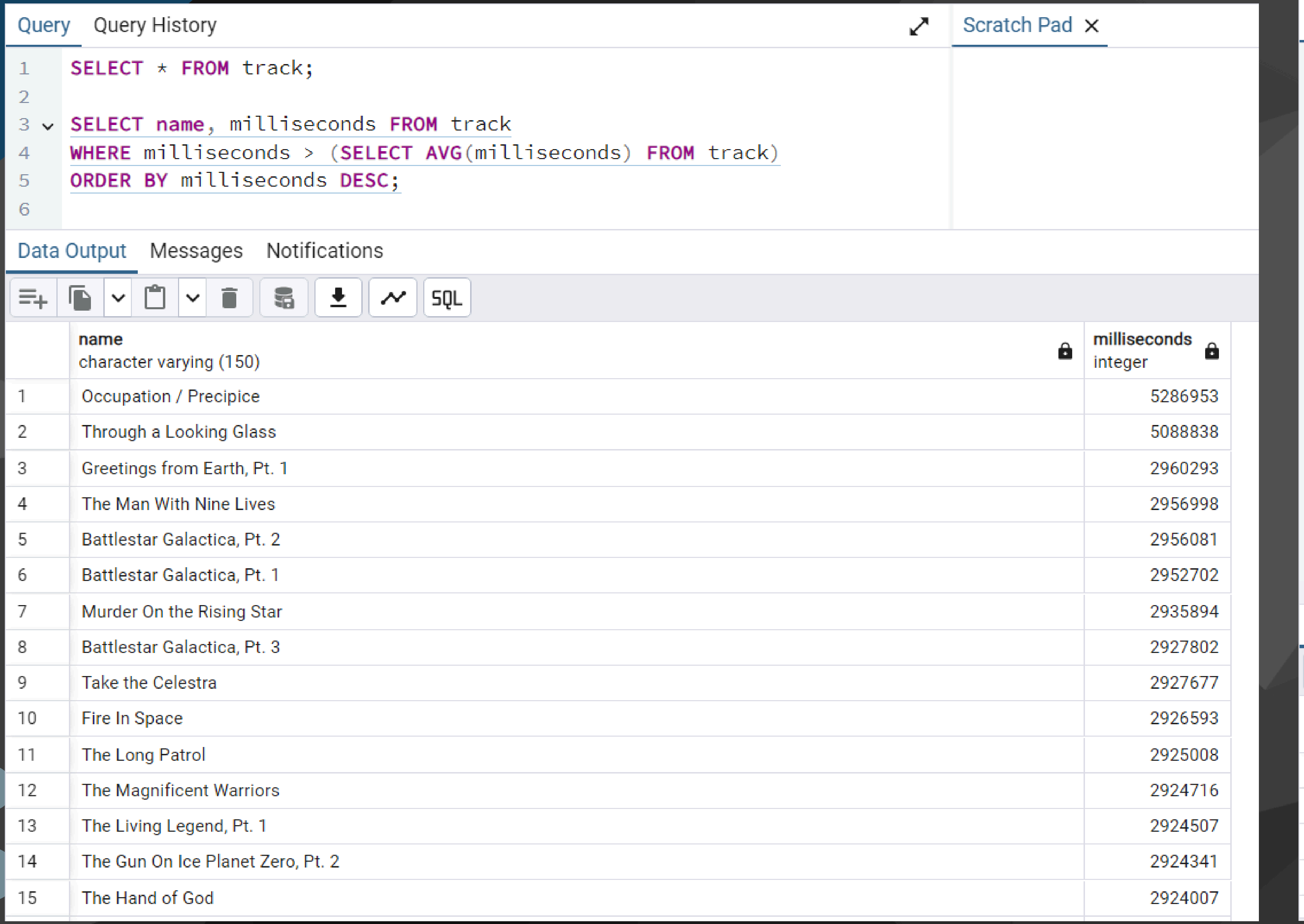


```sql
SELECT * FROM artist;
SELECT * FROM track;
SELECT * FROM  playlist;
SELECT * FROM album;

SELECT artist.artist_id, artist.name, COUNT(artist.artist_id) AS total_
FROM artist
JOIN album ON artist.artist_id = album.artist_id
JOIN track ON track.album_id = album.album_id
JOIN genre ON genre.genre_id = track.genre_id
WHERE genre.name LIKE 'Rock'
GROUP BY artist.artist_id
ORDER BY total_songs DESC
LIMIT 10;
```

	artist_id [PK] character varying (50)	name character varying (120)	total_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

Question 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.

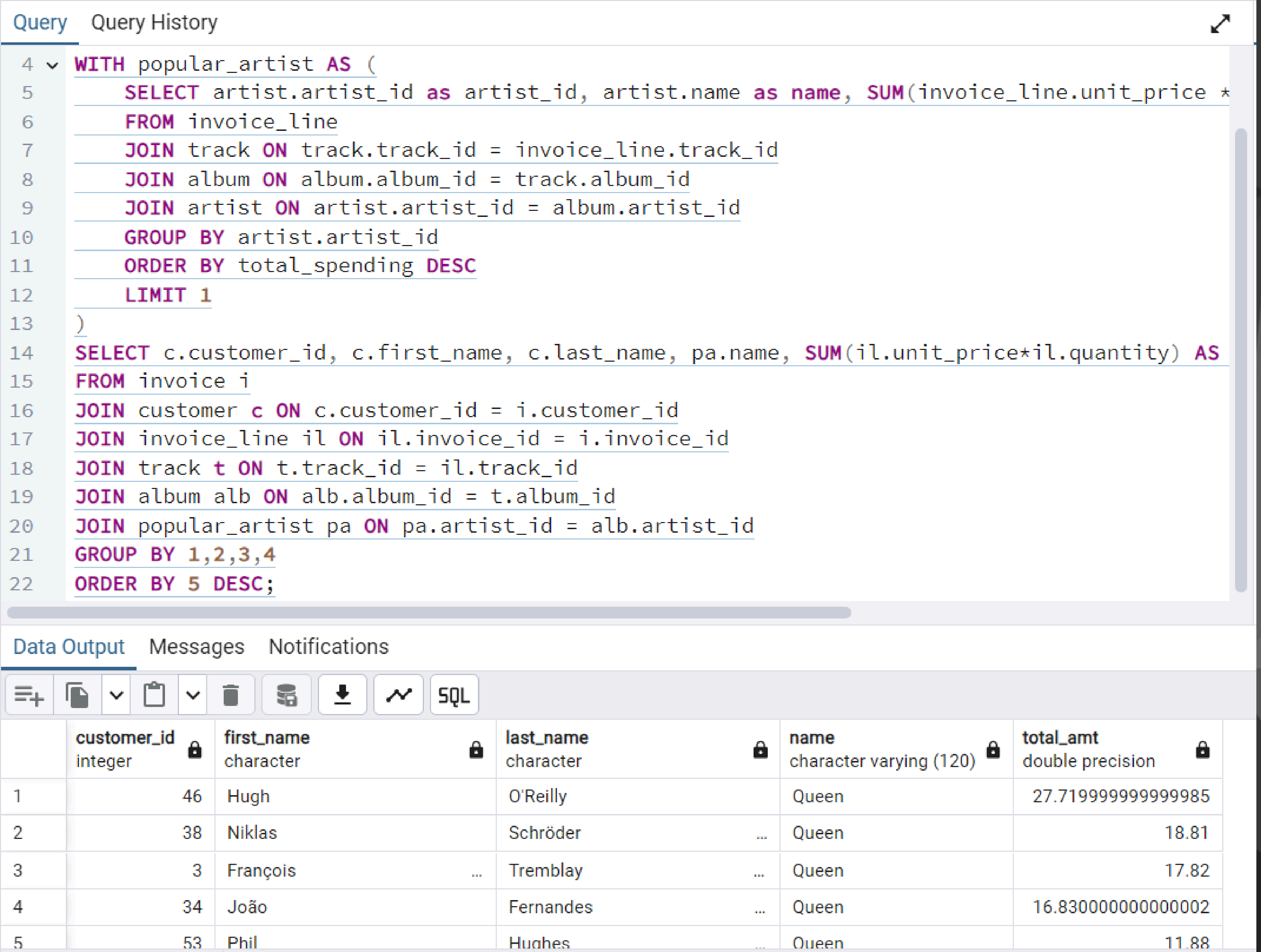


```sql
SELECT * FROM track;

SELECT name, milliseconds FROM track
WHERE milliseconds > (SELECT AVG(milliseconds) FROM track)
ORDER BY milliseconds DESC;

```

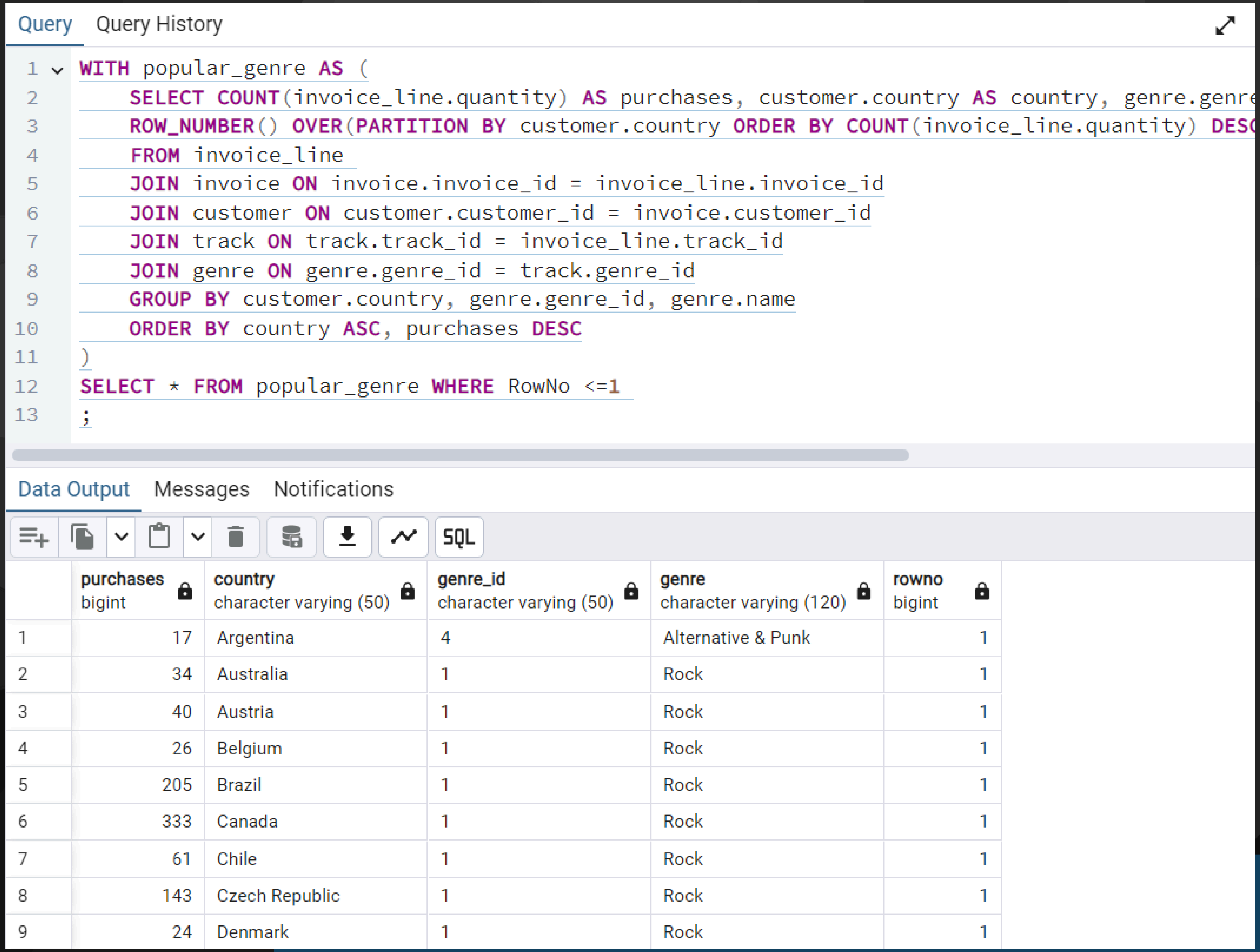
	name character varying (150)	milliseconds integer
1	Occupation / Precipice	5286953
2	Through a Looking Glass	5088838
3	Greetings from Earth, Pt. 1	2960293
4	The Man With Nine Lives	2956998
5	Battlestar Galactica, Pt. 2	2956081
6	Battlestar Galactica, Pt. 1	2952702
7	Murder On the Rising Star	2935894
8	Battlestar Galactica, Pt. 3	2927802
9	Take the Celestra	2927677
10	Fire In Space	2926593
11	The Long Patrol	2925008
12	The Magnificent Warriors	2924716
13	The Living Legend, Pt. 1	2924507
14	The Gun On Ice Planet Zero, Pt. 2	2924341
15	The Hand of God	2924007



```sql
WITH popular_artist AS (
    SELECT artist.artist_id as artist_id, artist.name as name, SUM(invoice_line.unit_price *
    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 artist.artist_id
    ORDER BY total_spending DESC
    LIMIT 1
)
SELECT c.customer_id, c.first_name, c.last_name, pa.name, SUM(il.unit_price*il.quantity) AS
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
JOIN popular_artist pa ON pa.artist_id = alb.artist_id
GROUP BY 1,2,3,4
ORDER BY 5 DESC;
```

	customer_id integer	first_name character	last_name character	name character varying (120)	total_amt double precision
1	46	Hugh	O'Reilly	Queen	27.719999999999985
2	38	Niklas	Schröder	Queen	18.81
3	3	François	Tremblay	Queen	17.82
4	34	João	Fernandes	Queen	16.830000000000002
5	53	Phil	Hughes	Queen	11.88

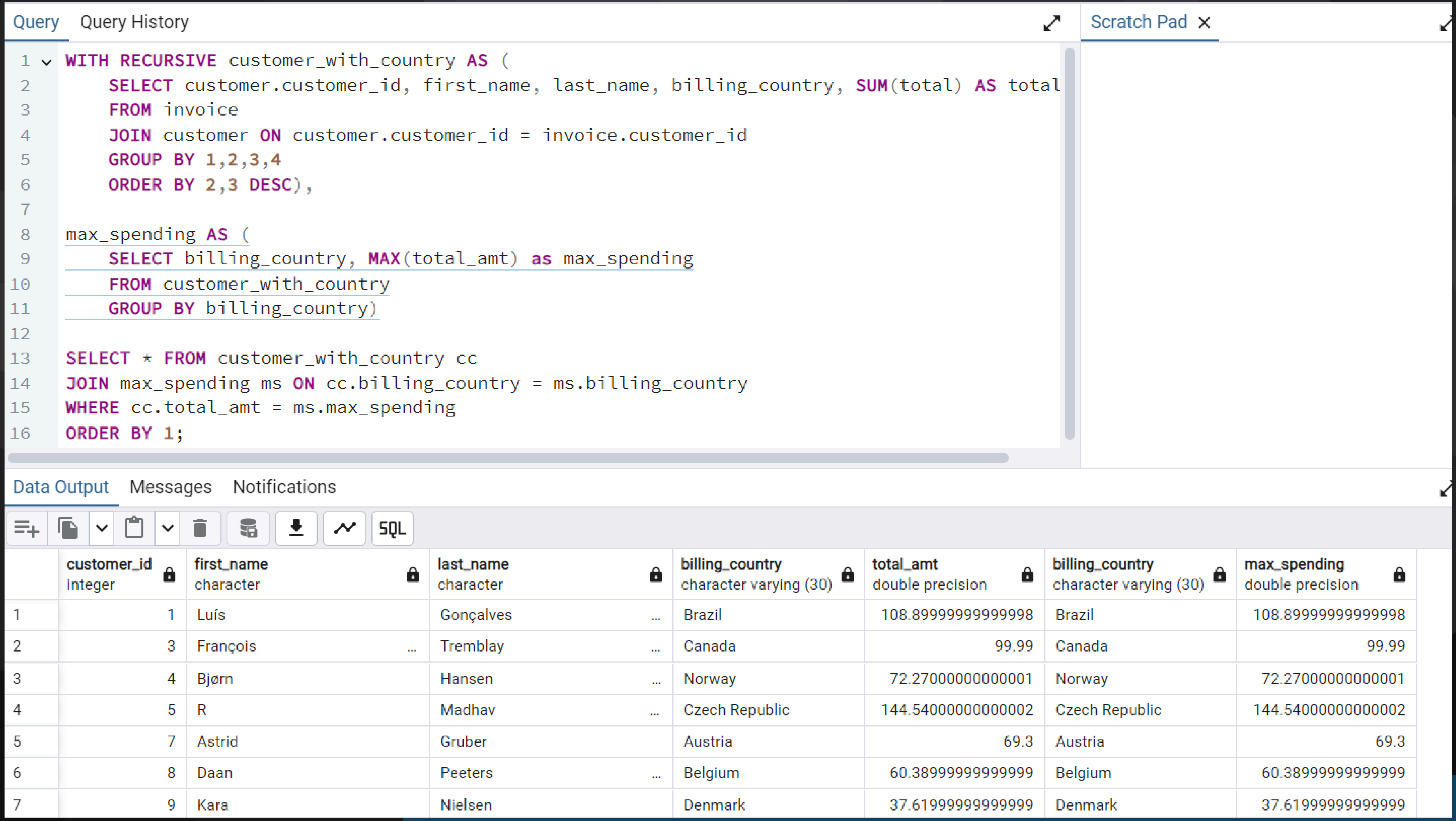
Question 9 : We want to find out the most popular music Genre for each country. We determine the most popular genre as the genre with the highest amount of purchases. Write a query that returns each country along with the top Genre. For countries where the maximum number of purchases is shared return all Genres.



```sql
WITH popular_genre AS (
    SELECT COUNT(invoice_line.quantity) AS purchases, customer.country AS country, genre.genre
    ROW_NUMBER() OVER(PARTITION BY customer.country ORDER BY COUNT(invoice_line.quantity) DESC
    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 customer.country, genre.genre_id, genre.name
    ORDER BY country ASC, purchases DESC
)
SELECT * FROM popular_genre WHERE RowNo <=1
;
```

	purchases bigint	country character varying (50)	genre_id character varying (50)	genre character varying (120)	rowno bigint
1	17	Argentina	4	Alternative & Punk	1
2	34	Australia	1	Rock	1
3	40	Austria	1	Rock	1
4	26	Belgium	1	Rock	1
5	205	Brazil	1	Rock	1
6	333	Canada	1	Rock	1
7	61	Chile	1	Rock	1
8	143	Czech Republic	1	Rock	1
9	24	Denmark	1	Rock	1

Question 10: 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.



```sql
WITH RECURSIVE customer_with_country AS (
    SELECT customer.customer_id, first_name, last_name, billing_country, SUM(total) AS total
    FROM invoice
    JOIN customer ON customer.customer_id = invoice.customer_id
    GROUP BY 1,2,3,4
    ORDER BY 2,3 DESC),

max_spending AS (
    SELECT billing_country, MAX(total_amt) as max_spending
    FROM customer_with_country
    GROUP BY billing_country)

SELECT * FROM customer_with_country cc
JOIN max_spending ms ON cc.billing_country = ms.billing_country
WHERE cc.total_amt = ms.max_spending
ORDER BY 1;
```

	customer_id integer	first_name character	last_name character	billing_country character varying (30)	total_amt double precision	billing_country character varying (30)	max_spending double precision
1	1	Luís	Gonçalves ...	Brazil	108.89999999999998	Brazil	108.89999999999998
2	3	François ...	Tremblay ...	Canada	99.99	Canada	99.99
3	4	Bjørn	Hansen ...	Norway	72.27000000000001	Norway	72.27000000000001
4	5	R	Madhav ...	Czech Republic	144.54000000000002	Czech Republic	144.54000000000002
5	7	Astrid	Gruber	Austria	69.3	Austria	69.3
6	8	Daan	Peeters ...	Belgium	60.38999999999999	Belgium	60.38999999999999
7	9	Kara	Nielsen	Denmark	37.61999999999999	Denmark	37.61999999999999

THANK YOU