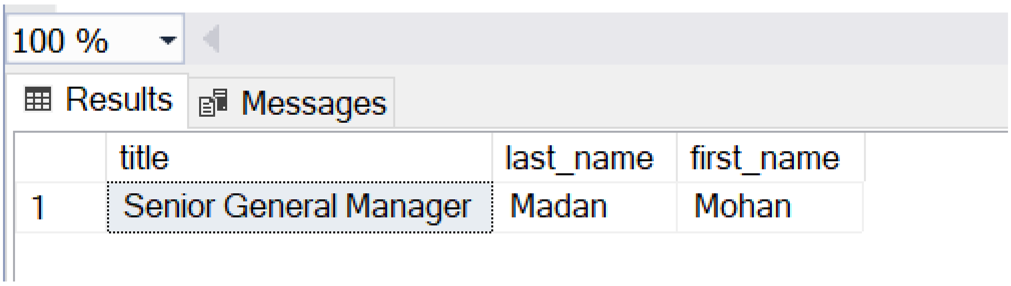
MUSIC STORE SQL QUERIES

SET-1

Q1: Who is the senior most employee based on job title?

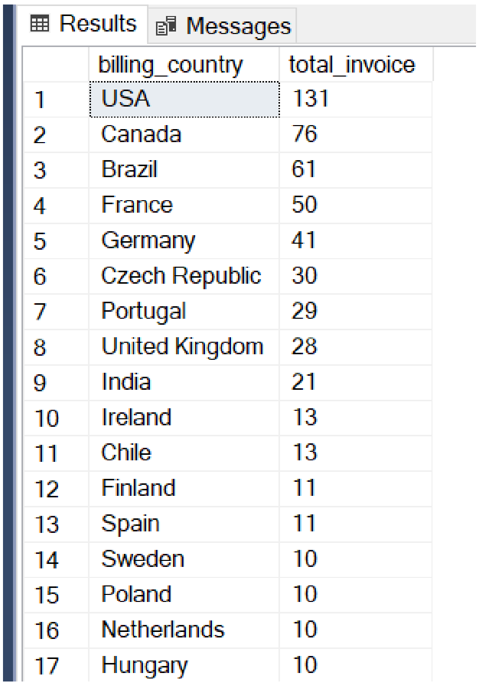
```
SELECT TOP 1 title, last_name, first_name

FROM employee

ORDER BY levels DESC;
```



	title	last_name	first_name
1	Senior General Manager	Madan	Mohan

Q2: Which countries have the most Invoices?

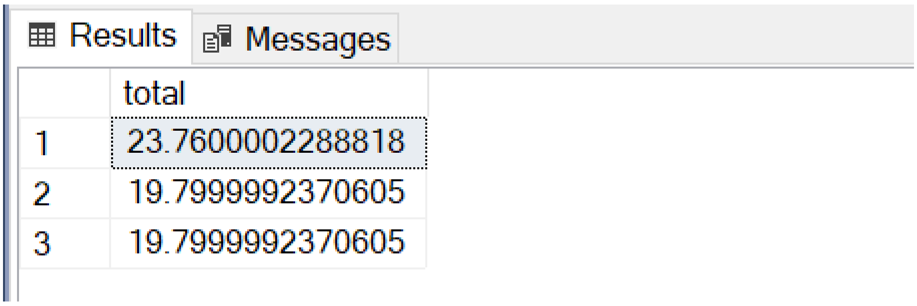
```
SELECT billing_country, COUNT(*) AS total_invoice

FROM invoice

GROUP BY billing_country

ORDER BY total_invoice DESC;
```



	billing_country	total_invoice
1	USA	131
2	Canada	76
3	Brazil	61
4	France	50
5	Germany	41
6	Czech Republic	30
7	Portugal	29
8	United Kingdom	28
9	India	21
10	Ireland	13
11	Chile	13
12	Finland	11
13	Spain	11
14	Sweden	10
15	Poland	10
16	Netherlands	10
17	Hungary	10

Q3: What are top 3 values of total invoice?

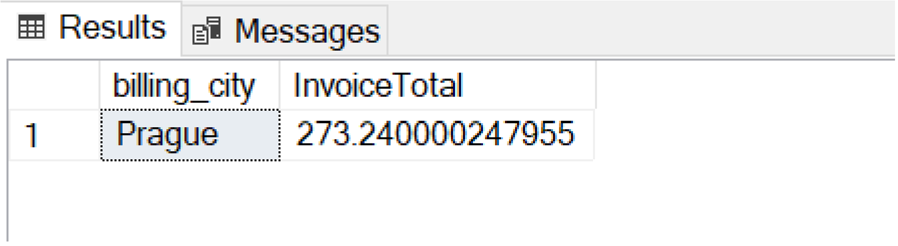
```
SELECT TOP 3 total

FROM invoice

ORDER BY total DESC;
```



	total
1	23.7600002288818
2	19.7999992370605
3	19.7999992370605

Q4: 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 TOP 1 billing_city,SUM(total) AS InvoiceTotal

FROM invoice

GROUP BY billing_city

ORDER BY InvoiceTotal DESC;
```

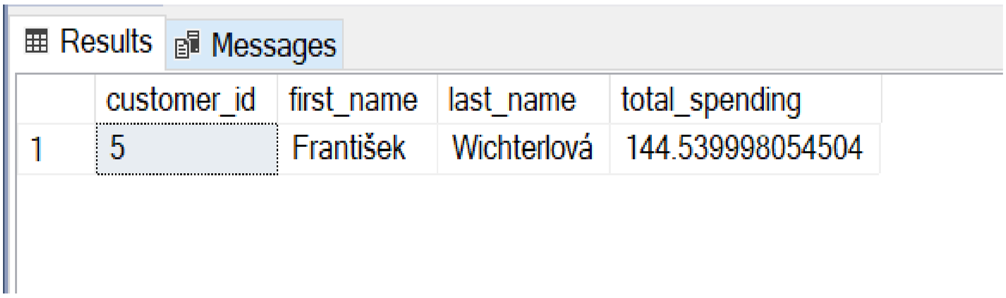


	billing_city	InvoiceTotal
1	Prague	273.240000247955

Q5: 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 TOP 1 c.customer_id, c.first_name, c.last_name, SUM(i.total) AS
total_spending

FROM customer c

JOIN invoice i ON c.customer_id = i.customer_id

GROUP BY c.customer_id, c.first_name, c.last_name
```

```
        ORDER BY total_spending DESC;
```



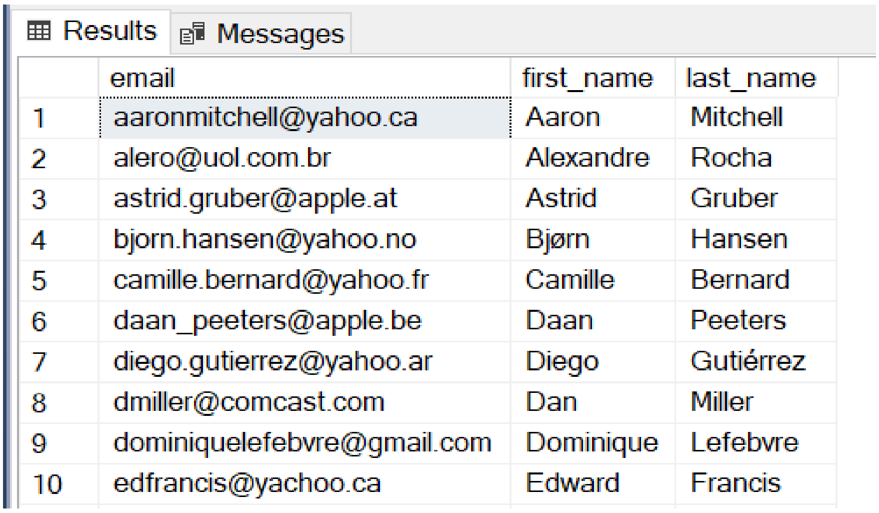
	customer_id	first_name	last_name	total_spending
1	5	František	Wichterlová	144.539998054504

SET-2

Q1: 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.

Method 1 :

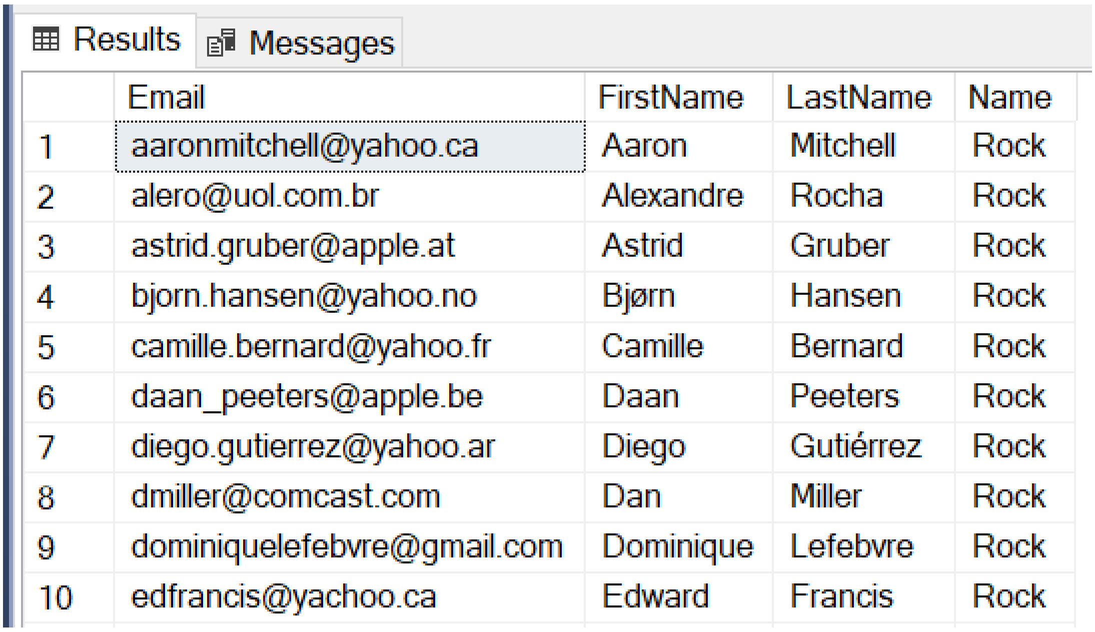
```
        SELECT DISTINCT email,first_name, last_name
        FROM customer c
        JOIN invoice i ON c.customer_id = i.customer_id
        JOIN invoice_line l ON i.invoice_id =l.invoice_id
        WHERE track_id IN(
                SELECT track_id FROM track t
                JOIN genre g ON t.genre_id = g.genre_id
                WHERE g.name LIKE 'Rock')
        ORDER BY email;
```



	email	first_name	last_name
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.com	Dominique	Lefebvre
10	edfrancis@yachoo.ca	Edward	Francis

Method 2 :

```
SELECT DISTINCT email AS Email,first_name AS FirstName, last_name AS LastName, g.name AS Name

FROM customer c

JOIN invoice i ON i.customer_id = c.customer_id

JOIN invoice_line l ON l.invoice_id = i.invoice_id

JOIN track t ON t.track_id =l.track_id

JOIN genre g ON g.genre_id = t.genre_id

WHERE g.name LIKE 'Rock'

ORDER BY email;
```



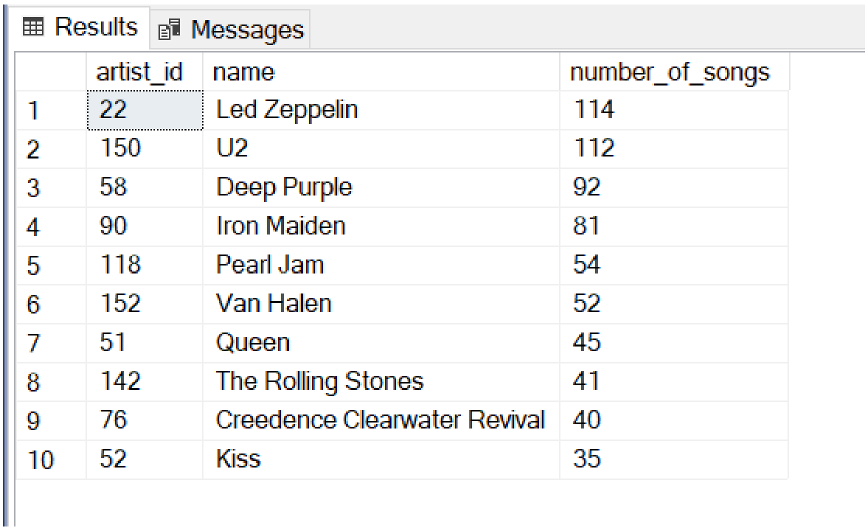
	Email	FirstName	LastName	Name
1	aaronmitchell@yahoo.ca	Aaron	Mitchell	Rock
2	alero@uol.com.br	Alexandre	Rocha	Rock
3	astrid.gruber@apple.at	Astrid	Gruber	Rock
4	bjorn.hansen@yahoo.no	Bjørn	Hansen	Rock
5	camille.bernard@yahoo.fr	Camille	Bernard	Rock
6	daan_peeters@apple.be	Daan	Peeters	Rock
7	diego.gutierrez@yahoo.ar	Diego	Gutiérrez	Rock
8	dmiller@comcast.com	Dan	Miller	Rock
9	dominiquelefebvre@gmail.com	Dominique	Lefebvre	Rock
10	edfrancis@yachoo.ca	Edward	Francis	Rock

Q2: 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 TOP 10 s.artist_id, s.name,COUNT(s.artist_id) AS number_of_songs

FROM track t

JOIN album a ON a.album_id = t.album_id

JOIN artist s ON s.artist_id = a.artist_id

JOIN genre g ON g.genre_id = t.genre_id
```

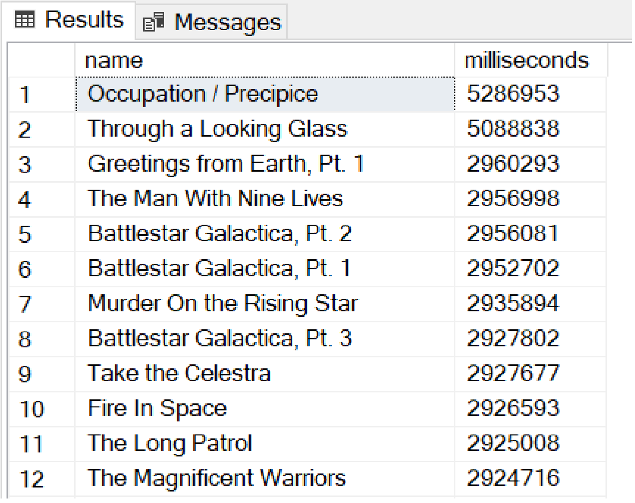
```
WHERE g.name LIKE 'Rock'

GROUP BY s.artist_id,s.name

ORDER BY number_of_songs DESC;
```



	artist_id	name	number_of_songs
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

Q3: 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;
```

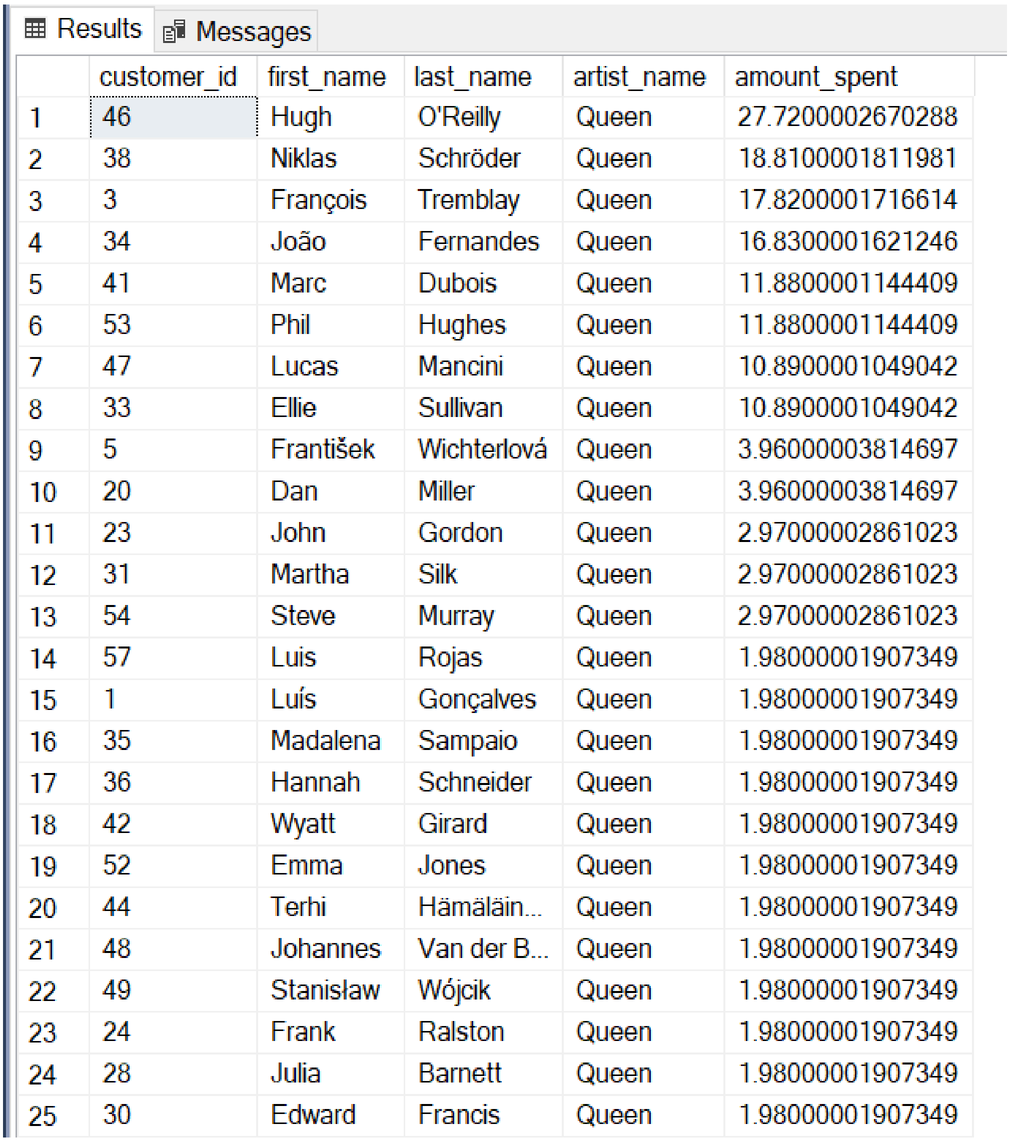


	name	milliseconds
1	Occupation / Precipice	5286953
2	Through a Looking Glass	5088838
3	Greetings from Earth, Pt. 1	2960293
4	The Man With Nine Lives	2956998
5	Battlestar Galactica, Pt. 2	2956081
6	Battlestar Galactica, Pt. 1	2952702
7	Murder On the Rising Star	2935894
8	Battlestar Galactica, Pt. 3	2927802
9	Take the Celestra	2927677
10	Fire In Space	2926593
11	The Long Patrol	2925008
12	The Magnificent Warriors	2924716

SET-3

Q1: 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 TOP 1 s.artist_id AS artist_id, s.name AS artist_name, SUM(l.unit_price*l.quantity) AS total_sales
        FROM invoice_line l
        JOIN track t ON t.track_id = l.track_id
        JOIN album a ON a.album_id = t.album_id
        JOIN artist s ON s.artist_id = a.artist_id
        GROUP BY s.artist_id,s.name
        ORDER BY total_sales DESC
        )
SELECT c.customer_id, c.first_name, c.last_name, bsa.artist_name, SUM(l.unit_price*l.quantity) AS amount_spent
FROM invoice i
JOIN customer c ON c.customer_id = i.customer_id
JOIN invoice_line l ON l.invoice_id = i.invoice_id
JOIN track t ON t.track_id = l.track_id
JOIN album a ON a.album_id = t.album_id
JOIN best_selling_artist bsa ON bsa.artist_id = a.artist_id
GROUP BY c.customer_id, c.first_name, c.last_name, bsa.artist_name
ORDER BY amount_spent DESC;
```

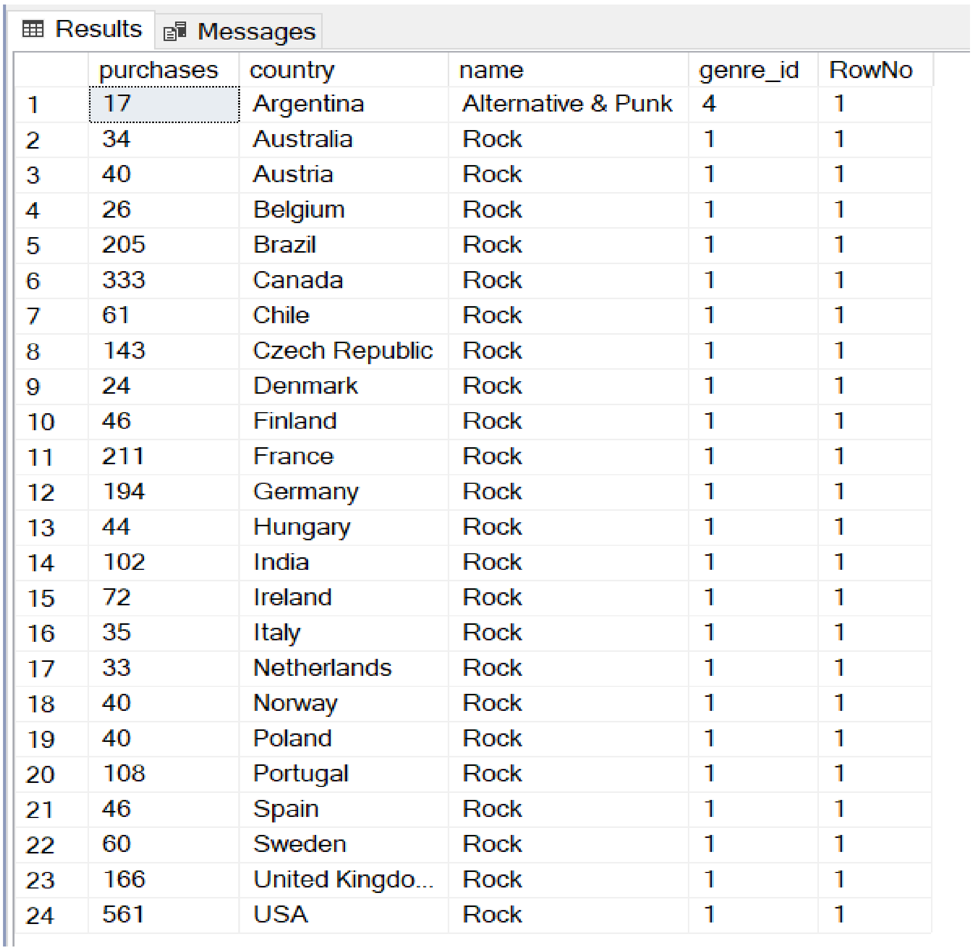


	customer_id	first_name	last_name	artist_name	amount_spent
1	46	Hugh	O'Reilly	Queen	27.7200002670288
2	38	Niklas	Schröder	Queen	18.8100001811981
3	3	François	Tremblay	Queen	17.8200001716614
4	34	João	Fernandes	Queen	16.8300001621246
5	41	Marc	Dubois	Queen	11.8800001144409
6	53	Phil	Hughes	Queen	11.8800001144409
7	47	Lucas	Mancini	Queen	10.8900001049042
8	33	Ellie	Sullivan	Queen	10.8900001049042
9	5	František	Wichterlová	Queen	3.96000003814697
10	20	Dan	Miller	Queen	3.96000003814697
11	23	John	Gordon	Queen	2.97000002861023
12	31	Martha	Silk	Queen	2.97000002861023
13	54	Steve	Murray	Queen	2.97000002861023
14	57	Luis	Rojas	Queen	1.98000001907349
15	1	Luís	Gonçalves	Queen	1.98000001907349
16	35	Madalena	Sampaio	Queen	1.98000001907349
17	36	Hannah	Schneider	Queen	1.98000001907349
18	42	Wyatt	Girard	Queen	1.98000001907349
19	52	Emma	Jones	Queen	1.98000001907349
20	44	Terhi	Hämäläin...	Queen	1.98000001907349
21	48	Johannes	Van der B...	Queen	1.98000001907349
22	49	Stanisław	Wójcik	Queen	1.98000001907349
23	24	Frank	Ralston	Queen	1.98000001907349
24	28	Julia	Barnett	Queen	1.98000001907349
25	30	Edward	Francis	Queen	1.98000001907349

Q2: 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(l.quantity) AS purchases, c.country, g.name, g.genre_id,

        ROW_NUMBER() OVER(PARTITION BY c.country ORDER BY COUNT(l.quantity) DESC)
AS RowNo

        FROM invoice_line l
```

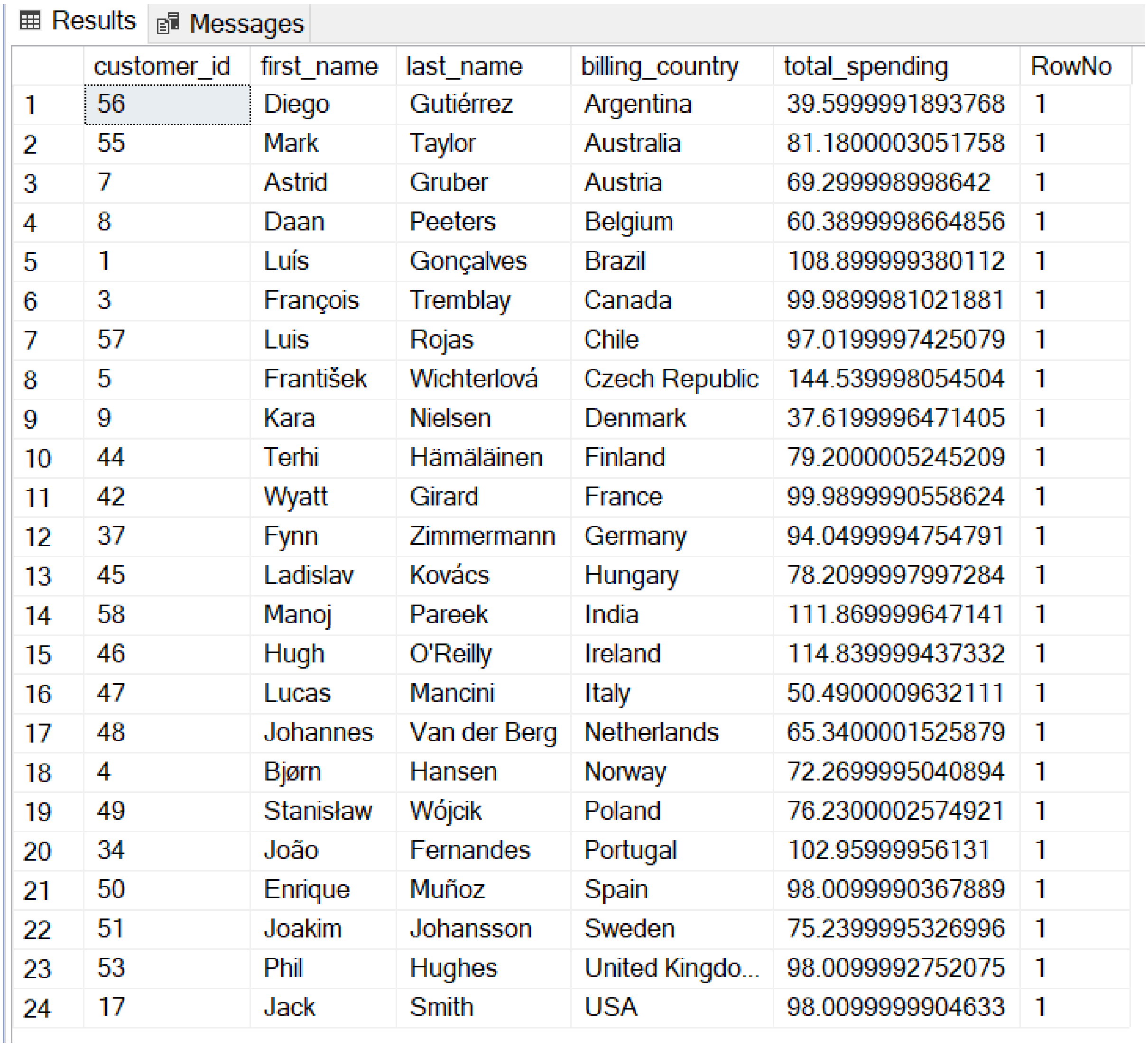
```
            JOIN invoice i ON i.invoice_id = l.invoice_id

            JOIN customer c ON c.customer_id = i.customer_id

            JOIN track t ON t.track_id = l.track_id

            JOIN genre g ON g.genre_id = t.genre_id

            GROUP BY c.country, g.name, g.genre_id

    )

    SELECT * FROM popular_genre WHERE RowNo <= 1
```



	purchases	country	name	genre_id	RowNo
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
9	24	Denmark	Rock	1	1
10	46	Finland	Rock	1	1
11	211	France	Rock	1	1
12	194	Germany	Rock	1	1
13	44	Hungary	Rock	1	1
14	102	India	Rock	1	1
15	72	Ireland	Rock	1	1
16	35	Italy	Rock	1	1
17	33	Netherlands	Rock	1	1
18	40	Norway	Rock	1	1
19	40	Poland	Rock	1	1
20	108	Portugal	Rock	1	1
21	46	Spain	Rock	1	1
22	60	Sweden	Rock	1	1
23	166	United Kingdo...	Rock	1	1
24	561	USA	Rock	1	1

Q3: 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
c.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 i

                JOIN customer c ON c.customer_id = i.customer_id

                GROUP BY c.customer_id,first_name,last_name,billing_country

                )

SELECT * FROM Custometer_with_country WHERE RowNo <= 1
```



	customer_id	first_name	last_name	billing_country	total_spending	RowNo
1	56	Diego	Gutiérrez	Argentina	39.5999991893768	1
2	55	Mark	Taylor	Australia	81.1800003051758	1
3	7	Astrid	Gruber	Austria	69.299998998642	1
4	8	Daan	Peeters	Belgium	60.3899998664856	1
5	1	Luís	Gonçalves	Brazil	108.899999380112	1
6	3	François	Tremblay	Canada	99.9899981021881	1
7	57	Luis	Rojas	Chile	97.0199997425079	1
8	5	František	Wichterlová	Czech Republic	144.539998054504	1
9	9	Kara	Nielsen	Denmark	37.6199996471405	1
10	44	Terhi	Hämäläinen	Finland	79.2000005245209	1
11	42	Wyatt	Girard	France	99.9899990558624	1
12	37	Fynn	Zimmermann	Germany	94.0499994754791	1
13	45	Ladislav	Kovács	Hungary	78.2099997997284	1
14	58	Manoj	Pareek	India	111.869999647141	1
15	46	Hugh	O'Reilly	Ireland	114.839999437332	1
16	47	Lucas	Mancini	Italy	50.4900009632111	1
17	48	Johannes	Van der Berg	Netherlands	65.3400001525879	1
18	4	Bjørn	Hansen	Norway	72.2699995040894	1
19	49	Stanisław	Wójcik	Poland	76.2300002574921	1
20	34	João	Fernandes	Portugal	102.95999956131	1
21	50	Enrique	Muñoz	Spain	98.0099990367889	1
22	51	Joakim	Johansson	Sweden	75.2399995326996	1
23	53	Phil	Hughes	United Kingdo...	98.0099992752075	1
24	17	Jack	Smith	USA	98.0099999904633	1