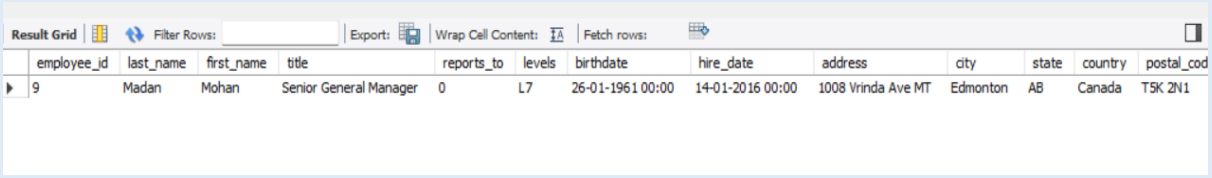
MUSIC DATA ANALYSIS

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

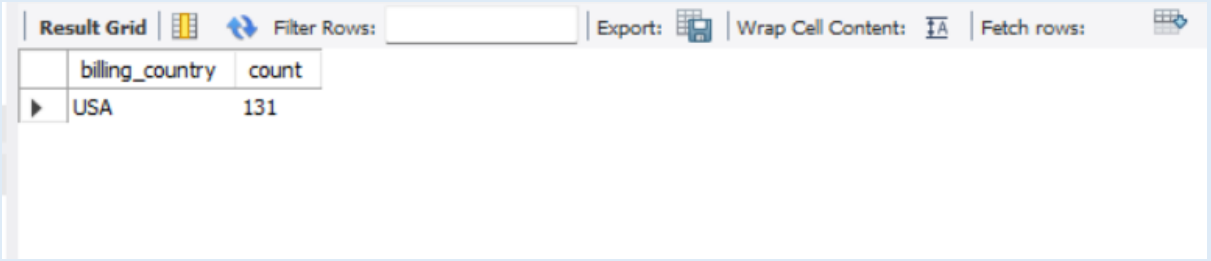
```
SELECT *
FROM employee
ORDER BY levels DESC
LIMIT 1;
```



employee_id	last_name	first_name	title	reports_to	levels	birthdate	hire_date	address	city	state	country	postal_cod
9	Madan	Mohan	Senior General Manager	0	L7	26-01-1961 00:00	14-01-2016 00:00	1008 Vrinda Ave MT	Edmonton	AB	Canada	T5K 2N1

Q2: Which countries have most invoices?

```
SELECT billing_country, COUNT(1) as count
FROM invoice
GROUP BY billing_country
ORDER BY count desc
LIMIT 1;
```



billing_country	count
USA	131

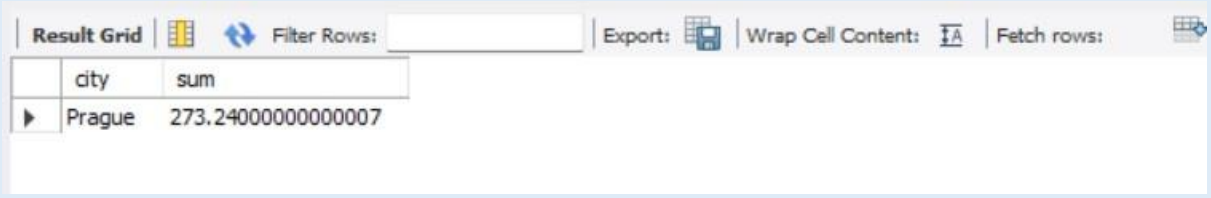
Q3: What are top 3 invoice values?

```
SELECT total FROM invoice
ORDER BY total desc LIMIT 3;
```

invoice_id	customer_id	invoice_date	billing_address	billing_city	billing_state	billing_country	billing_postal_code	total
183	42	2018-02-09 00:00:00	9, Place Louis Barthou	Bordeaux	None	France	33000	23.759999999999998
92	32	2017-07-02 00:00:00	696 Osborne Street	Winnipeg	MB	Canada	R3L 2B9	19.8
526	5	2020-06-08 00:00:00	Klanova 9/506	Prague	None	Czech Republic	14700	19.8

Q4: Which city has the best customers? Write a query that returns one city that has the highest sum of invoice totals. Return both the city name and sum of all invoice totals.

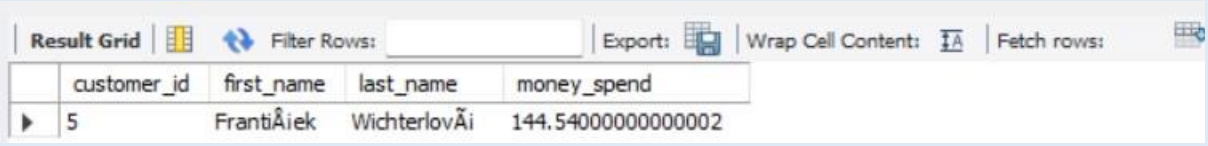
```
SELECT billing_city as city , SUM(total) as sum
FROM invoice
GROUP BY billing_city
ORDER BY sum  DESC
LIMIT 1;
```



city	sum
Prague	273.24000000000007

Q5: Who is the best customer ? The customer who has spent the most money will be declared as the best customer. Write a query that returns the person who has spent the most money.

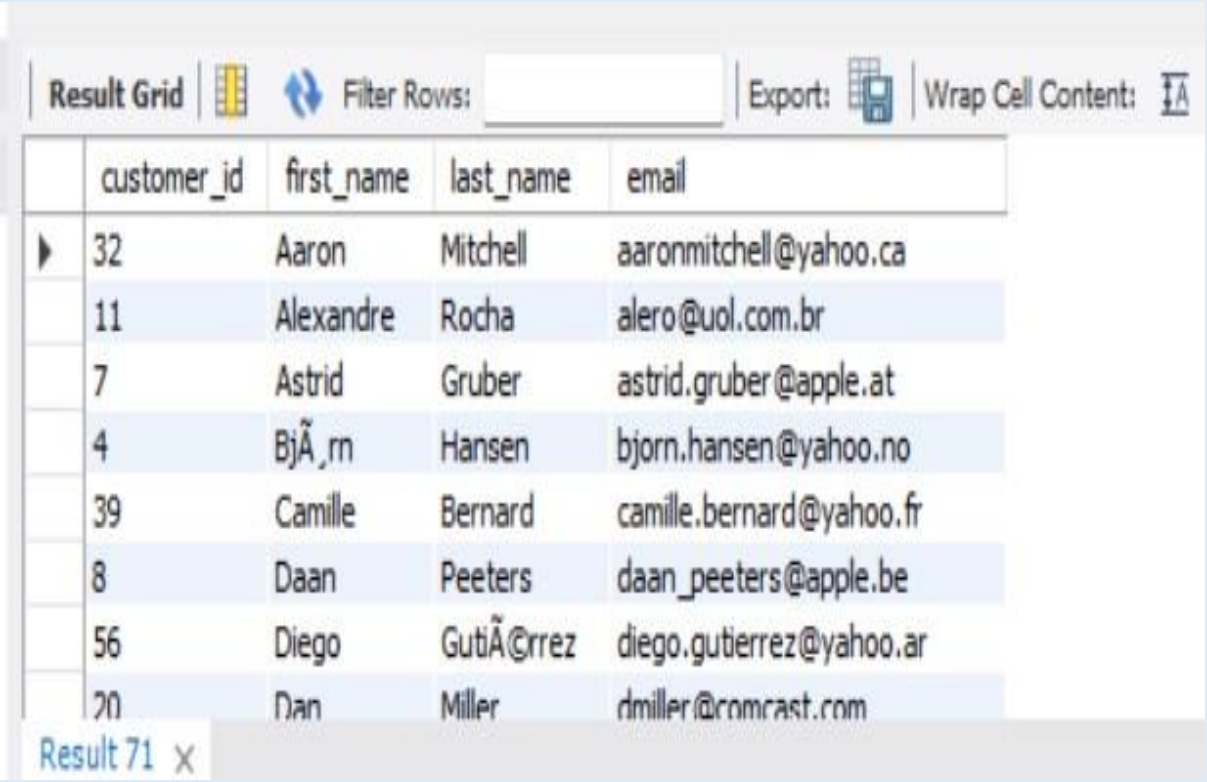
```
SELECT c.customer_id, c.first_name, c.last_name, SUM(i.total) as money_spend
FROM customer c
JOIN invoice i ON c.customer_id = i.customer_id
GROUP BY c.customer_id,c.first_name, c.last_name
ORDER BY money_spend DESC
LIMIT 1;
```



customer_id	first_name	last_name	money_spend
5	FrantiÅiek	WichterlovÃi	144.54000000000002

Q6: Write a query to return the email, first name and last name and Genre of all Rock Music listeners. Return your list alphabetically by email starting with A.

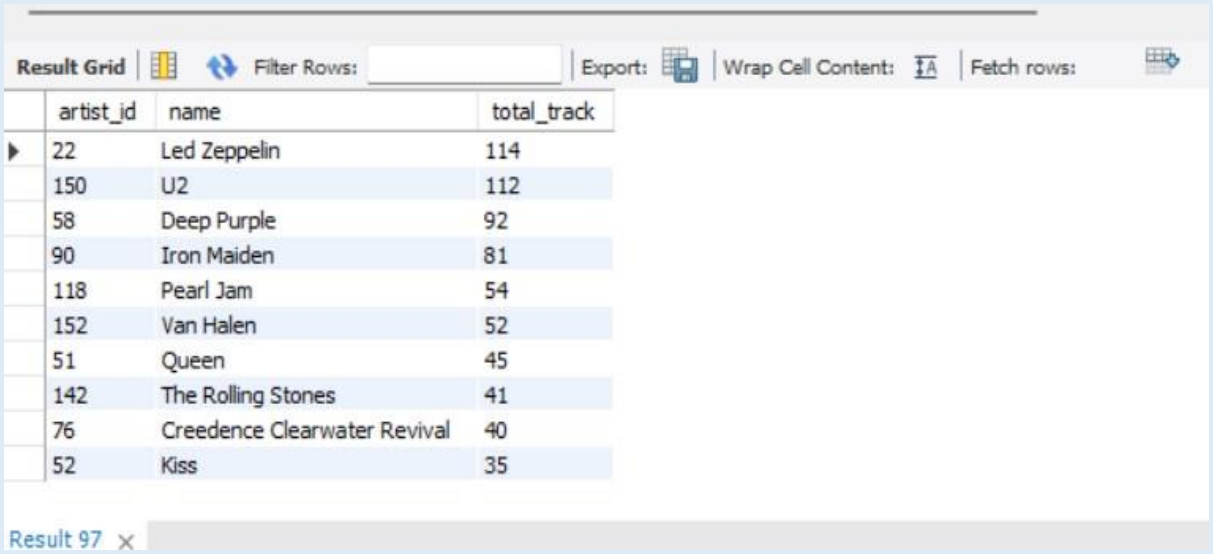
```
SELECT DISTINCT customer.customer_id,customer.first_name,customer.last_name,customer.email
FROM track
JOIN genre ON track.genre_id = genre.genre_id
jOIN invoice_line ON track.track_id = invoice_line.track_id
jOIN invoice ON invoice_line.invoice_id = invoice.invoice_id
JOIN customer ON invoice.customer_id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



customer_id	first_name	last_name	email
32	Aaron	Mitchell	aaronmitchell@yahoo.ca
11	Alexandre	Rocha	alero@uol.com.br
7	Astrid	Gruber	astrid.gruber@apple.at
4	BjÃ¸rn	Hansen	bjorn.hansen@yahoo.no
39	Camille	Bernard	camille.bernard@yahoo.fr
8	Daan	Peeters	daan_peeters@apple.be
56	Diego	GutiÃ©rrez	diego.gutierrez@yahoo.ar
20	Dan	Miller	dmiller@comcast.com

Q7: Let's invite the artists that have written most rock music in our dataset. Write a query that returns the artist's name and total track of the top 10 rock bands

```
SELECT artist.artist_id, artist.name, COUNT(*) as total_track
FROM track
JOIN album2  ON track.album_id = album2.album_id
JOIN artist ON album2.artist_id = artist.artist_id
JOIN genre ON genre.genre_id = track.genre_id
WHERE genre.name LIKE 'Rock'
GROUP BY artist.name,artist.artist_id
ORDER BY total_track DESC
LIMIT 10;
```



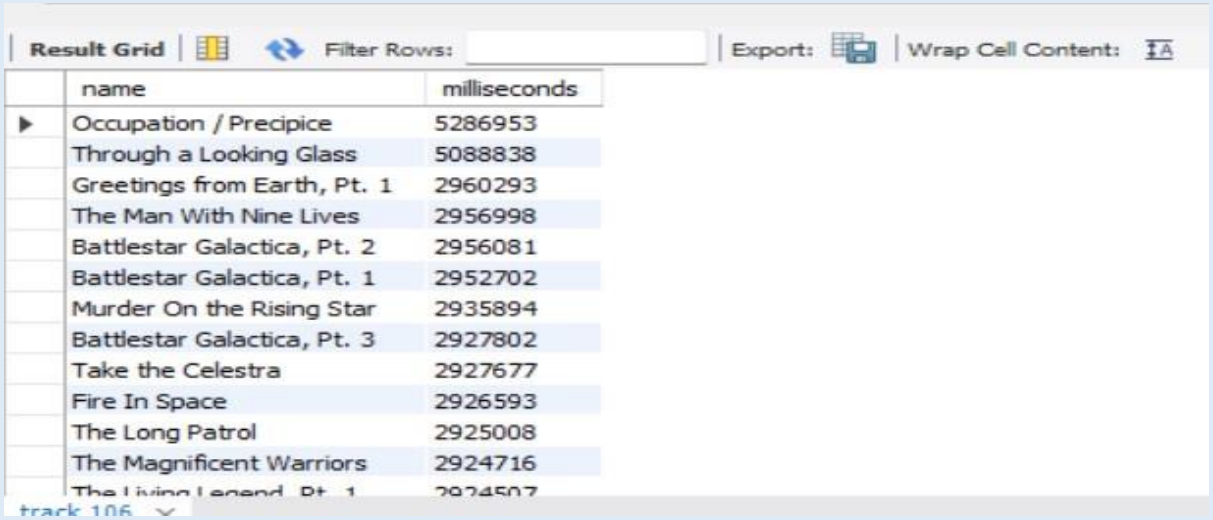
Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

artist_id	name	total_track
22	Led Zeppelin	114
150	U2	112
58	Deep Purple	92
90	Iron Maiden	81
118	Pearl Jam	54
152	Van Halen	52
51	Queen	45
142	The Rolling Stones	41
76	Creedence Clearwater Revival	40
52	Kiss	35

Result 97

Q8: Return all the tracks that have a song length longer than the average song length.
Return the Name and Milliseconds for each track. Order by song length with longest songs listed first.

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



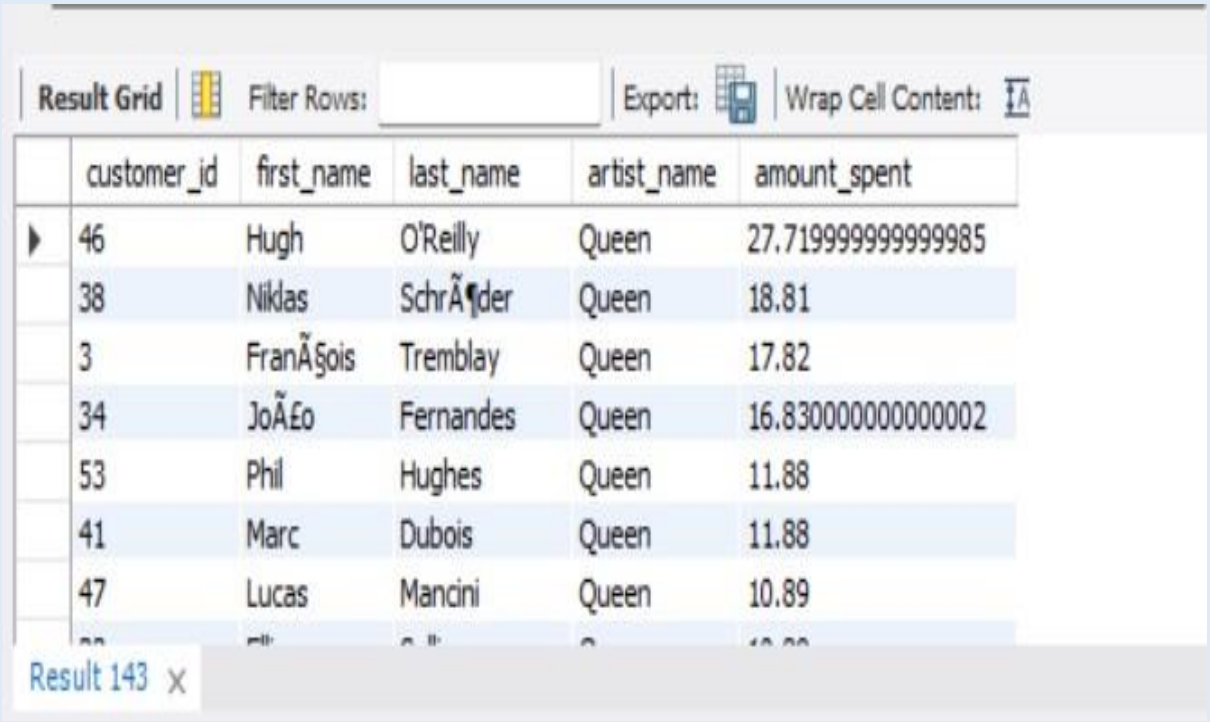
Result Grid | Filter Rows: | Export: | Wrap Cell Content:

name	milliseconds
Occupation / Precipice	5286953
Through a Looking Glass	5088838
Greetings from Earth, Pt. 1	2960293
The Man With Nine Lives	2956998
Battlestar Galactica, Pt. 2	2956081
Battlestar Galactica, Pt. 1	2952702
Murder On the Rising Star	2935894
Battlestar Galactica, Pt. 3	2927802
Take the Celestra	2927677
Fire In Space	2926593
The Long Patrol	2925008
The Magnificent Warriors	2924716
The Living Legend, Pt. 1	2924507

track 106

Q9: Find how much amount spent by each customer on best-selling artist. Write a query to return the artist's name , customer name and total spent.

```
with best_selling_artist 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 track
join album2 on track.album_id = album2.album_id
join artist on artist.artist_id = album2.artist_id
join invoice_line on track.track_id = invoice_line.track_id
GROUP BY artist.artist_id,artist.name
ORDER BY total_sales 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 album2 alb ON alb.album_id = t.album_id
JOIN best_selling_artist bsa ON bsa.artist_id = alb.artist_id
GROUP BY 1,2,3,4
ORDER BY 5 DESC;
```



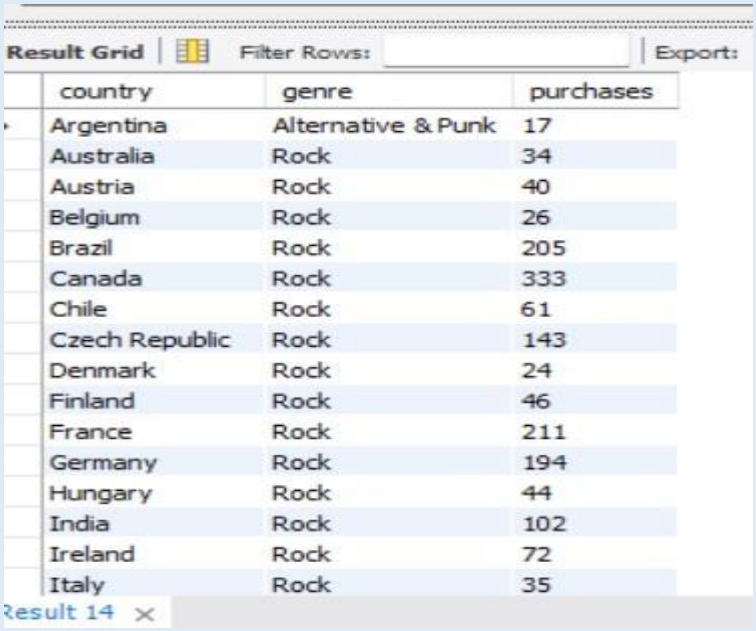
Result Grid | Filter Rows: | Export: | Wrap Cell Content:

customer_id	first_name	last_name	artist_name	amount_spent
46	Hugh	O'Reilly	Queen	27.719999999999985
38	Niklas	SchrÃ¶der	Queen	18.81
3	FranÃ§ois	Tremblay	Queen	17.82
34	JoÃ£o	Fernandes	Queen	16.830000000000002
53	Phil	Hughes	Queen	11.88
41	Marc	Dubois	Queen	11.88
47	Lucas	Mancini	Queen	10.89

Result 143

Q10: We want to find out the most popular music genre in each country. We determine the most popular genre as genre with highest number of purchases. Write a query that returns each country with top genre. Countries for which maximum number of purchases is shared return all genres.

```
WITH popular_genre AS
(
   SELECT  customer.country as country, genre.name as genre, genre.genre_id, COUNT(invoice_line.quantity)
AS purchases,
          RANK() 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 1,2,3
          ORDER BY 1 ASC, 4 DESC
)
SELECT p.country,p.genre,p.purchases FROM popular_genre p WHERE RowNo = 1
```



Result Grid | Filter Rows: | Export:

country	genre	purchases
Argentina	Alternative & Punk	17
Australia	Rock	34
Austria	Rock	40
Belgium	Rock	26
Brazil	Rock	205
Canada	Rock	333
Chile	Rock	61
Czech Republic	Rock	143
Denmark	Rock	24
Finland	Rock	46
France	Rock	211
Germany	Rock	194
Hungary	Rock	44
India	Rock	102
Ireland	Rock	72
Italy	Rock	35

Result 14