

MUSIC STORE SQL QUERIES

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

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

	employee_id	last_name	first_name	title	reports_to	levels	birthdate	hire_date	address	city	state	country	postal_code	phone	fax	email
1	9	Madan	Mohan	Senior General Manager	NULL	L7	1961-01-26	2016-01-14	1008 Vrinda Ave MT	Edmonton	AB	Canada	T5K 2N1	+1 (780) 428-9482	+1 (780) 428-3457	madan.mohan@chinookcorp.com

Q2: Which countries have the most Invoices?

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

	Invoice_Count	Billing_Country
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
9	21	India
10	13	Ireland
11	13	Chile
12	11	Finland
13	11	Spain
14	10	Sweden
15	10	Poland
16	10	Netherlands
17	10	Hungary
18	10	Denmark
19	10	Australia
20	9	Austria
21	9	Italy
22	9	Norway
23	7	Belgium
24	5	Argentina

Q3: What are top 3 values of total invoice?

```
SELECT TOP 3 CEILING(total) AS Total_Invoice_Count
FROM invoice
ORDER BY Total_Invoice_Count DESC;
```

Results		Messages
	Total_Invoice_Count	
1	24	
2	20	
3	20	

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 AS City_Name,
    CEILING(SUM(total)) AS Total_Invoice_Sum
FROM
    invoice
GROUP BY
    billing_city
ORDER BY
    Total_Invoice_Sum DESC;
```

Results			Messages
	City_Name	Total_Invoice_Sum	
1	Prague	274	

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 AS Customer_ID,
    c.first_name AS First_Name,
    c.last_name AS Last_Name,
    CEILING(SUM(i.total)) AS Total_Money_Spent
FROM
    customer c
JOIN
    invoice i ON i.customer_id = c.customer_id
GROUP BY
    c.customer_id, c.first_name, c.last_name
ORDER BY
    Total_Money_Spent DESC;
```

Results		Messages		
	Customer_ID	First_Name	Last_Name	Total_Money_Spent
1	5	František	Wichterlová	145

Q6: Write a 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.first_name AS First_Name,
    c.last_name AS Last_Name,
    c.email AS Email_ID,
    g.name AS Genre
FROM
    customer c
JOIN
    invoice i ON i.customer_id = c.customer_id
JOIN
    invoice_line il ON il.invoice_id = i.invoice_id
JOIN
    track t ON t.track_id = il.track_id
JOIN
    genre g ON g.genre_id = t.genre_id
WHERE
    g.name IN ('Rock')
ORDER BY
    Email_ID ASC;
```

	First_Name	Last_Name	Email_ID	Genre
1	Aaron	Mitchell	aaronmitchell@yahoo.ca	Rock
2	Alexandre	Rocha	alero@uol.com.br	Rock
3	Astrid	Gruber	astrid.gruber@apple.at	Rock
4	Bjørn	Hansen	bjorn.hansen@yahoo.no	Rock
5	Camille	Bernard	camille.bernard@yahoo.fr	Rock
6	Daan	Peeters	daan_peeters@apple.be	Rock
7	Diego	Gutiérrez	diego.gutierrez@yahoo.ar	Rock
8	Dan	Miller	dmiller@comcast.com	Rock
9	Dominique	Lefebvre	dominiquelefebvre@gmail.com	Rock
10	Edward	Francis	edfrancis@yahoo.ca	Rock
11	Eduardo	Martins	eduardo@woodstock.com.br	Rock
12	Ellie	Sullivan	ellie.sullivan@shaw.ca	Rock
13	Emma	Jones	emma_jones@hotmail.com	Rock
14	Enrique	Muñoz	enrique_munoz@yahoo.es	Rock
15	Fernanda	Ramos	fernadaramos4@uol.com.br	Rock
16	Frank	Harris	fharris@google.com	Rock
17	Frank	Ralston	fralston@gmail.com	Rock
18	František	Wichterlová	frantisekw@jetbrains.com	Rock
19	François	Tremblay	ftremblay@gmail.com	Rock
20	Fynn	Zimmerm...	fzimmermann@yahoo.de	Rock
21	Hannah	Schneider	hannah.schneider@yahoo.de	Rock
22	Helena	Holý	hholy@gmail.com	Rock
23	Heather	Leacock	hleacock@gmail.com	Rock
24	Hugh	O'Reilly	hughoreilly@apple.ie	Rock
25	Isabelle	Mercier	isabelle_mercier@apple.fr	Rock
26	Jack	Smith	jacksmith@microsoft.com	Rock
27	Jennifer	Peterson	jenniferp@rogers.ca	Rock
28	João	Fernandes	jfernandes@yahoo.pt	Rock
29	Joakim	Johansson	joakim.johansson@yahoo.se	Rock
30	Johannes	Van der B...	johavanderberg@yahoo.nl	Rock
31	John	Gordon	johngordon22@yahoo.com	Rock
32	Julia	Barnett	jubarnett@gmail.com	Rock
33	Kathy	Chase	kachase@hotmail.com	Rock
34	Kara	Nielsen	kara.nielsen@jubii.dk	Rock
35	Ladislav	Kovács	ladislav_kovacs@apple.hu	Rock
36	Leonie	Köhler	leonekohler@surfeu.de	Rock

37	Lucas	Mancini	lucas.mancini@yahoo.it	Rock
38	Luis	Gonçalves	luisg@embraer.com.br	Rock
39	Luis	Rojas	luisrojas@yahoo.cl	Rock
40	Manoj	Pareek	manoj.pareek@rediff.com	Rock
41	Marc	Dubois	marc.dubois@hotmail.com	Rock
42	Mark	Taylor	mark.taylor@yahoo.au	Rock
43	Martha	Silk	marthasilk@gmail.com	Rock
44	Madalena	Sampaio	masampaio@sapo.pt	Rock
45	Michelle	Brooks	michelleb@aol.com	Rock
46	Mark	Philips	mphilips12@shaw.ca	Rock
47	Niklas	Schröder	nschroder@surfeu.de	Rock
48	Patrick	Gray	patrick.gray@aol.com	Rock
49	Phil	Hughes	phil.hughes@gmail.com	Rock
50	Richard	Cunningh...	ricunningham@hotmail.com	Rock
51	Rishabh	Mishra	rishabh_mishra@yahoo.in	Rock
52	Robert	Brown	robbrown@shaw.ca	Rock
53	Roberto	Almeida	roberto.almeida@riotur.gov.br	Rock
54	Stanislaw	Wójcik	stanislaw.wojcik@wp.pl	Rock
55	Steve	Murray	steve.murray@yahoo.uk	Rock
56	Terhi	Hämäläin...	terhi.hamalainen@apple.fi	Rock
57	Tim	Goyer	tgoyer@apple.com	Rock
58	Victor	Stevens	vstevens@yahoo.com	Rock
59	Wyatt	Girard	wyatt.girard@yahoo.fr	Rock

Q7: 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 a.artist_id AS Artist_ID,
    a.name AS Artist_Name,
    g.name AS Genre,
    COUNT(a.artist_id) AS Number_of_Songs
FROM
    artist a
JOIN
    album al ON al.artist_id = a.artist_id
JOIN
    track t ON t.album_id = al.album_id
JOIN
    genre g ON g.genre_id = t.genre_id
WHERE
    g.name IN ('Rock')
GROUP BY
    a.artist_id, a.name, g.name
ORDER BY
    Number_of_Songs DESC;
```

	Artist_ID	Artist_Name	Genre	Number_of_Songs
1	22	Led Zeppelin	Rock	114
2	150	U2	Rock	112
3	58	Deep Purple	Rock	92
4	90	Iron Maiden	Rock	81
5	118	Pearl Jam	Rock	54
6	152	Van Halen	Rock	52
7	51	Queen	Rock	45
8	142	The Rolling Stones	Rock	41
9	76	Creedence Clearwater Revival	Rock	40
10	52	Kiss	Rock	35

Q8: 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
    t.track_id AS Track_ID,
    t.name AS Track_Name,
    t.milliseconds AS Track_Length
FROM
    track t
WHERE
    t.milliseconds > (SELECT AVG(t.milliseconds) AS Avg_Track_Length
                     FROM track t)
GROUP BY
    t.track_id,
    t.name,
    t.milliseconds
ORDER BY
    Track_Length DESC;
```

Q9: 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 a.artist_id AS Artist_ID,  
        a.name AS Artist_Name,  
        SUM(il.unit_price * il.quantity) AS Total_Sales  
    FROM  
        artist a  
    JOIN  
        album al ON al.artist_id = a.artist_id  
    JOIN  
        track t ON t.album_id = al.album_id  
    JOIN  
        invoice_line il ON il.track_id = t.track_id  
    GROUP BY  
        a.artist_id,  
        a.name  
    ORDER BY  
        Total_Sales DESC  
)  
SELECT  
    c.customer_id AS Customer_ID,  
    c.first_name AS First_Name,  
    c.last_name AS Last_Name,  
    bsa.Artist_Name AS Artist_Name,  
    SUM(il.unit_price * il.quantity) AS Total_Spent  
FROM  
    customer c  
JOIN  
    invoice i ON i.customer_id = c.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 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  
    Total_Spent DESC;
```

	Customer_ID	First_Name	Last_Name	Artist_Name	Total_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	Stanislaw	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
26	8	Daan	Peeters	Queen	1.98000001907349
27	11	Alexandre	Rocha	Queen	1.98000001907349
28	15	Jennifer	Peterson	Queen	1.98000001907349
29	16	Frank	Harris	Queen	1.98000001907349
30	17	Jack	Smith	Queen	1.98000001907349
31	19	Tim	Goyer	Queen	0.990000009536...
32	13	Fernanda	Ramos	Queen	0.990000009536...
33	6	Helena	Holý	Queen	0.990000009536...
34	22	Heather	Leacock	Queen	0.990000009536...
35	26	Richard	Cunningh...	Queen	0.990000009536...
36	27	Patrick	Gray	Queen	0.990000009536...
37	50	Enrique	Muñoz	Queen	0.990000009536...
38	45	Ladislav	Kovács	Queen	0.990000009536...
39	43	Isabelle	Mercier	Queen	0.990000009536...
40	39	Camille	Bernard	Queen	0.990000009536...
41	58	Manoj	Pareek	Queen	0.990000009536...
42	59	Rishabh	Mishra	Queen	0.990000009536...
43	55	Mark	Taylor	Queen	0.990000009536...

Q10: 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
        g.genre_id AS Genre_ID,
        g.name AS Genre_Name,
        c.country AS Country,
        COUNT(il.quantity) AS Highest_Purchase,
        ROW_NUMBER() OVER(PARTITION BY c.country ORDER BY
COUNT(il.quantity) DESC) AS Rank_No
    FROM
        genre g
    JOIN
        track t ON t.genre_id = g.genre_id
    JOIN
        invoice_line il ON il.track_id = t.track_id
    JOIN
        invoice i ON i.invoice_id = il.invoice_id
    JOIN
        customer c ON c.customer_id = i.customer_id
    GROUP BY
        g.genre_id,
        g.name,
        c.country
)
SELECT *
FROM Popular_Genre
WHERE Rank_No <= 1
ORDER BY Country ASC;
```

Results		Messages			
	Genre_ID	Genre_Name	Country	Highest_Purchase	Rank_No
1	4	Alternative & Punk	Argentina	17	1
2	1	Rock	Australia	34	1
3	1	Rock	Austria	40	1
4	1	Rock	Belgium	26	1
5	1	Rock	Brazil	205	1
6	1	Rock	Canada	333	1
7	1	Rock	Chile	61	1
8	1	Rock	Czech Republic	143	1
9	1	Rock	Denmark	24	1
10	1	Rock	Finland	46	1
11	1	Rock	France	211	1
12	1	Rock	Germany	194	1
13	1	Rock	Hungary	44	1
14	1	Rock	India	102	1
15	1	Rock	Ireland	72	1
16	1	Rock	Italy	35	1
17	1	Rock	Netherlands	33	1
18	1	Rock	Norway	40	1
19	1	Rock	Poland	40	1
20	1	Rock	Portugal	108	1
21	1	Rock	Spain	46	1
22	1	Rock	Sweden	60	1
23	1	Rock	United Kingdo...	166	1
24	1	Rock	USA	561	1

Q11: 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 Customer_with_Country AS (
    SELECT
        c.customer_id AS Customer_ID,
        c.first_name AS First_Name,
        c.last_name AS Last_Name,
        i.billing_country AS Billing_Country,
        CEILING(SUM(i.total)) AS Total_Spent,
        ROW_NUMBER() OVER(PARTITION BY i.billing_country ORDER BY
SUM(i.total) DESC) AS Rank_No
    FROM
        customer c
    JOIN
        invoice i ON i.customer_id = c.customer_id
    GROUP BY
        c.customer_id,
        c.first_name,
        c.last_name,
        i.billing_country
)
SELECT *
FROM Customer_with_Country
WHERE Rank_No <= 1
ORDER BY Billing_Country ASC;

```

Results		Messages				
	Customer_ID	First_Name	Last_Name	Billing_Country	Total_Spent	Rank_No
1	56	Diego	Gutiérrez	Argentina	40	1
2	55	Mark	Taylor	Australia	82	1
3	7	Astrid	Gruber	Austria	70	1
4	8	Daan	Peeters	Belgium	61	1
5	1	Luís	Gonçalves	Brazil	109	1
6	3	François	Tremblay	Canada	100	1
7	57	Luis	Rojas	Chile	98	1
8	5	František	Wichterlová	Czech Republic	145	1
9	9	Kara	Nielsen	Denmark	38	1
10	44	Terhi	Hämäläinen	Finland	80	1
11	42	Wyatt	Girard	France	100	1
12	37	Fynn	Zimmermann	Germany	95	1
13	45	Ladislav	Kovács	Hungary	79	1
14	58	Manoj	Pareek	India	112	1
15	46	Hugh	O'Reilly	Ireland	115	1
16	47	Lucas	Mancini	Italy	51	1
17	48	Johannes	Van der Berg	Netherlands	66	1
18	4	Bjørn	Hansen	Norway	73	1
19	49	Stanislaw	Wójcik	Poland	77	1
20	34	João	Fernandes	Portugal	103	1
21	50	Enrique	Muñoz	Spain	99	1
22	51	Joakim	Johansson	Sweden	76	1
23	53	Phil	Hughes	United Kingdo...	99	1
24	17	Jack	Smith	USA	99	1