Music Store Data Analysis Using SQL

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

```
select * from employee

order by levels desc

limit 1;
```

Data Output | Messages | Notifications

	employee_id [PK] character varying (50)	last_name character (50)	first_name character (50)	title character varying (50)	reports_to character varying (30)	levels character varying (10)	birthdate timestamp without time zone
1	9	Madan	Mohan	Senior General Manager	[null]	L7	1961-01-26 00:00:00

Q3: Which countries have the most Invoices?

```
select COUNT(*) as C, billing_country from invoice

group by billing_country

order by C desc

limit 10;
```

	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
9	21	India
10	13	Chile

Q3: What are top 3 values of total invoice ?

```
select total from invoice

order by total desc

limit 3;
```

	total double precision
1	23.759999999999998
2	19.8
3	19.8

Music Store Data Analysis Using SQL

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 SUM (total) as invoice_ttl, billing_city

from invoice

group by billing_city

order by invoice_ttl desc

limit 10;
```

	invoice_ttl double precision	billing_city character varying (30)
1	273.24000000000007	Prague
2	169.29	Mountain View
3	166.32	London
4	158.4	Berlin
5	151.47	Paris
6	129.69	São Paulo
7	114.83999999999997	Dublin
8	111.86999999999999	Delhi
9	108.89999999999998	São José dos Campos
10	106.91999999999999	Brasília

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 customer.customer_id, customer.first_name, customer.last_name, sum(invoice.total) as total

from customer

join invoice on customer.customer_id = invoice.customer_id

group by customer.customer_id

order by total desc

limit 1;
```

	customer_id [PK] integer	first_name character (50)	last_name character (50)	total double precision
1	5	R ...	Madhav	144.54000000000002

Music Store Data Analysis Using SQL

Q6: 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 email, first_name, last_name

FROM customer

JOIN invoice ON customer.customer_id = invoice.customer_id

JOIN invoice_line ON invoice.invoice_id =invoice_line.invoice_id

WHERE track_id IN (

SELECT track_id FROM track

JOIN genre ON track.genre_id= genre.genre_id

WHERE genre.name LIKE 'Rock'

)

ORDER BY email;
```

	email character varying (50)	first_name character (50)	last_name character (50)
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
10	edfrancis@yachoo.ca	Edward	Francis

Q7: Lets 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 artist.artist_id, artist.name, count (artist.artist_id) as no_of_songs from track

join album on album.album_id= track.album_id

join artist on artist.artist_id= album.artist_id

join genre on genre.genre_id = track.genre_id

where genre.name like 'Rock'

group by artist.artist_id

order by no_of_songs desc

limit 10;
```

Music Store Data Analysis Using SQL

	artist_id [PK] character varying (50)	name character varying (120)	no_of_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
9	76	Creedence Clearwater Revival	40
10	52	Kiss	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 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 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

Music Store Data Analysis Using SQL

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 artist.artist_id as artist_id, artist.name as artist_name,
        sum(invoice_line.unit_price*invoice_line.quantity) as total_sales
        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 1
        order by 3 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 album 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;
```

	customer_id integer	first_name character (50)	last_name character (50)	artist_name character varying (120)	amount_spent 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
6	41	Marc	Dubois	Queen	11.88
7	47	Lucas	Mancini	Queen	10.89
8	33	Ellie	Sullivan	Queen	10.89
9	20	Dan	Miller	Queen	3.96
10	5	R	Madhav	Queen	3.96

Music Store Data Analysis Using SQL

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
      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
      customer.country, genre.name, genre.genre_id
)
SELECT * FROM popular_genre
WHERE rowno = 1
limit 10;
```

	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
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

Music Store Data Analysis Using SQL

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 customer.customer_id, customer.first_name, customer.last_name, invoice.billing_country,
      SUM(invoice.total) AS total_spending
   FROM invoice
   JOIN customer ON customer.customer_id = invoice.customer_id
   GROUP BY
      customer.customer_id, customer.first_name, customer.last_name, invoice.billing_country
),
country_max_spending AS (
   SELECT  billing_country, MAX(total_spending) AS max_spending
   FROM customer_with_country
   GROUP BY billing_country
)

SELECT cc.billing_country, cc.customer_id, cc.total_spending, cc.first_name, cc.last_name
FROM
   customer_with_country cc
JOIN country_max_spending ms ON cc.billing_country = ms.billing_country
   AND cc.total_spending = ms.max_spending
ORDER BY
   cc.billing_country;
```

	billing_country character varying (30)	customer_id integer	total_spending double precision	first_name character (50)	last_name character (50)
1	Argentina	56	39.6	Diego	Gutiérrez
2	Australia	55	81.18	Mark	Taylor
3	Austria	7	69.3	Astrid	Gruber
4	Belgium	8	60.38999999999999	Daan	Peeters
5	Brazil	1	108.89999999999998	Luís	Gonçalves
6	Canada	3	99.99	François	Tremblay
7	Chile	57	97.02000000000001	Luis	Rojas
8	Czech Republic	5	144.54000000000002	R	Madhav
9	Denmark	9	37.61999999999999	Kara	Nielsen
10	Finland	44	79.2	Terhi	Hämäläinen

Music Store Data Analysis Using SQL

OR

```
WITH customer_spending AS (
   SELECT customer.customer_id, customer.first_name, customer.last_name, invoice.billing_country,
      SUM(invoice.total) AS total_spending,
      ROW_NUMBER() OVER(PARTITION BY invoice.billing_country ORDER BY SUM(invoice.total) DESC) AS
rowno
   FROM invoice
   JOIN customer ON customer.customer_id = invoice.customer_id
   GROUP BY
      customer.customer_id, customer.first_name, customer.last_name, invoice.billing_country
)
SELECT * from customer_spending where rowno <=1
```

	customer_id integer	first_name character (50)	last_name character (50)	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
6	3	François	Tremblay	Canada	99.99	1
7	57	Luis	Rojas	Chile	97.02000000000001	1
8	5	R	Madhav	Czech Republic	144.54000000000002	1
9	9	Kara	Nielsen	Denmark	37.61999999999999	1
10	44	Terhi	Hämäläinen	Finland	79.2	1

I have conducted an extensive analysis of the music store data using SQL, focusing on 11 critical business questions. The insights derived from this analysis can significantly enhance decision-making across marketing, sales, and customer engagement strategies.

Conclusion:

This analysis identifies **top regions**, **best customers**, and **popular music genres**, allowing the company to target high-revenue areas with personalized promotions and events. Insights into **top sales transactions** and **customer preferences** guide decisions on artist partnerships and tailored offerings. These findings help optimize marketing, enhance customer loyalty, and drive growth in key markets.

Music Store Data Analysis Using SQL