



# MUSIC STORE



A N A L Y S I S  
U S I N G S Q L

# QUESTIONS

1. 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.

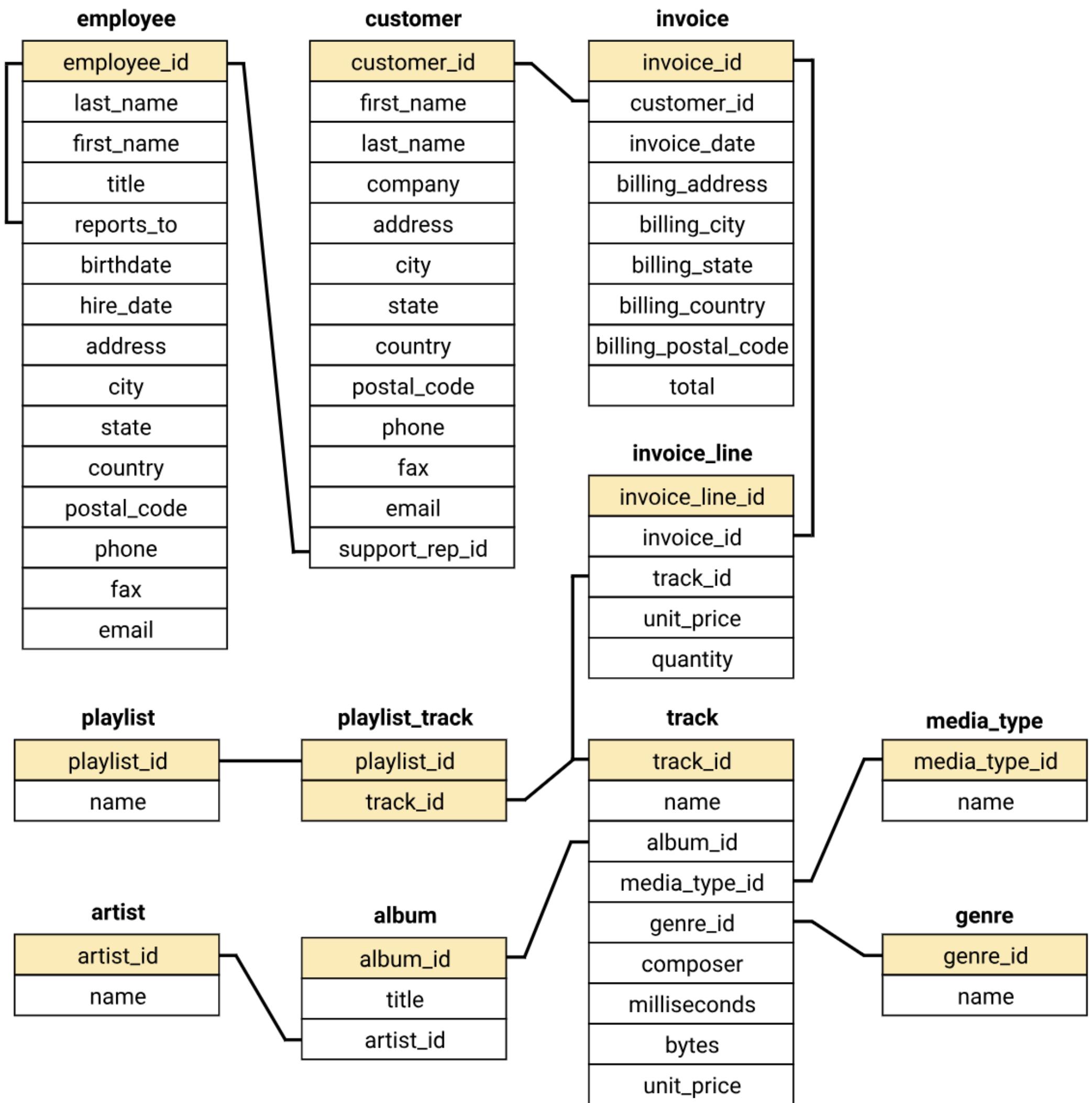
Q2: Let's invite the artists who have written the most rock music in our dataset.  
Write a query that returns the Artist name and total track count of the top 10 rock bands.

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.

**Q4: Find how much amount spent by each customer on artists? Write a query to return customer name, artist name and total spent**

**Q5: 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.**

**Q6: 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**



```
-- .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  
    invoiceline ON invoice.invoice_id = invoiceline.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;
```

```
-- 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
```

```
    artist.artist_id,  
    artist.name,  
    COUNT(artist.artist_id) AS number_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 number_of_songs DESC
```

```
LIMIT 10;
```

```
-- 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;
```

Steps to Solve: First, find which artist has earned the most according to the InvoiceLines. Now use this artist to find which customer spent the most on this artist. For this query, you will need to use the Invoice, InvoiceLine, Track, Customer, Album, and Artist tables. Note, this one is tricky because the Total spent in the Invoice table might not be on a single product, so you need to use the InvoiceLine table to find out how many of each product was purchased, and then multiply this by the price for each artist

```
-- Find how much amount spent by each customer on artists? Write a query to return customer name, artist name and total amount spent by each customer on artists.
```

```
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_amount_spent
    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;
```

- There are two parts in question- first most popular music genre and second need data at country level.

```
-- 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 2,3,4  
    ORDER BY 2 ASC, 1 DESC  
)  
SELECT * FROM popular_genre WHERE RowNo <= 1
```

Steps to Solve: Similar to the above question. There are two parts in question- first find the most spent on music for each country and second filter the data for respective customers.

```
-- 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 RECURSIVE  
    customter_with_country AS (  
        SELECT customer.customer_id,first_name,last_name,billing_country,SUM(total) AS total_spending  
        FROM invoice  
        JOIN customer ON customer.customer_id = invoice.customer_id  
        GROUP BY 1,2,3,4  
        ORDER BY 2,3 DESC),  
  
    country_max_spending AS(  
        SELECT billing_country,MAX(total_spending) AS max_spending  
        FROM customter_with_country  
        GROUP BY billing_country)  
  
SELECT cc.billing_country, cc.total_spending, cc.first_name, cc.last_name, cc.customer_id  
FROM customter_with_country cc  
JOIN country_max_spending ms  
ON cc.billing_country = ms.billing_country  
WHERE cc.total_spending = ms.max_spending  
ORDER BY 1;
```



THANK YOU