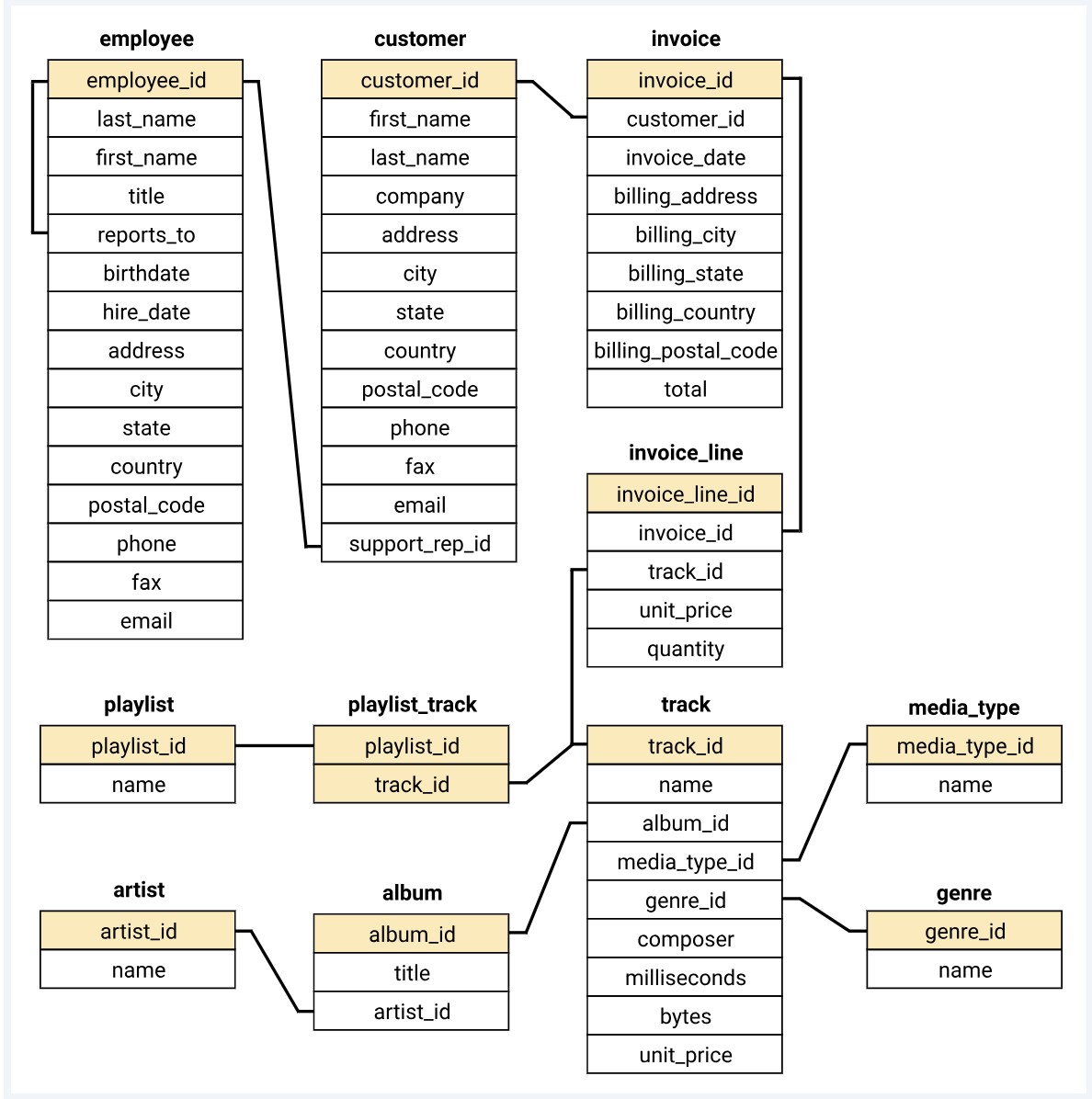
**Music Store Data Analysis project using SQL**

**Objective:** This project is for beginners and will teach you how to analyze the music playlist database. You can examine the dataset with SQL and help the store understand its business growth by answering simple questions. Aim is to business growth of the Music Store.



We will now solve questions based on this Music Store data that we have.

**Q1. Who is the senior most employee?**

SELECT \* FROM EMPLOYEE

ORDER BY LEVELS DESC

LIMIT 1;

**Q2. Which countries have the most invoices?**

SELECT COUNT(\*) AS C, BILLING\_COUNTRY //ANY COLUMN COUNT() WILL WORK HERE

FROM INVOICE

GROUP BY BILLING\_COUNTRY

ORDER BY C DESC

LIMIT 1;

**Q3. What are top 3 values of total invoices?**

SELECT TOTAL FROM INVOICES

ORDER BY TOTAL DESC

LIMIT 3;

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

SELECT BILLING\_CITY, SUM(TOTAL) AS INVOICE\_TOTAL

FROM INVOICE

GROUP BY BILLING CITY

ORDER BY INVOICE\_TOTAL DESC

LIMIT 1;

**Q5. Who is the best customer? Write a query that returns the person who has spent the most money.**

SELECT CUSTOMER.CUSTOMERID, CUSTOMER.FIRSTNAME, CUSTOMER.LASTNAME, SUM(INVOICE.TOTAL) AS TOTAL

FROM CUSTOMER

JOIN INVOICE

ON CUSTOMER.CUSTOMERID = INVOICE.CUSTOMERID

GROUP BY CUSTOMER.CUSTOMERID

ORDER BY TOTAL DESC

LIMIT 1;

**Q6. Write a query to return the email, first name, last name and genre of all rock music listners. 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;

**Q7. Write a query that returns the artist’s 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;

**Q8. Return all track names that have a song length greater than the average song length. Return name and milliseconds. Order the song names 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;

**Q9. 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;

**Q10. We want to find out the most popular music genre for each country. We determine the most popular genre as the genre having the highest number of purchases.**

Method 1: Using CTE

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

Method 2: : Using Recursive

WITH RECURSIVE sales\_per\_country AS(

SELECT COUNT(\*) AS purchases\_per\_genre, customer.country, genre.name, genre.genre\_id

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

),

max\_genre\_per\_country AS (SELECT MAX(purchases\_per\_genre) AS max\_genre\_number, country

FROM sales\_per\_country

GROUP BY 2

ORDER BY 2)

SELECT sales\_per\_country.\*

FROM sales\_per\_country

JOIN max\_genre\_per\_country ON sales\_per\_country.country = max\_genre\_per\_country.country

WHERE sales\_per\_country.purchases\_per\_genre = max\_genre\_per\_country.max\_genre\_number;

**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.**

Method 1: using CTE

WITH Customter\_with\_country AS (

SELECT customer.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

JOIN customer ON customer.customer\_id = invoice.customer\_id

GROUP BY 1,2,3,4

ORDER BY 4 ASC,5 DESC

)

SELECT \* FROM Customter\_with\_country WHERE RowNo <= 1

Method 2: Using Recursive

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;