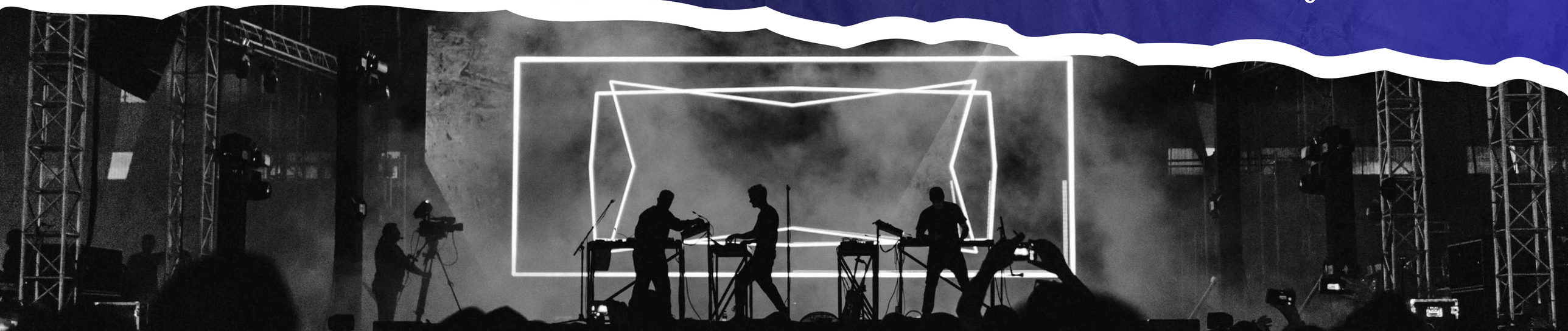
SQL PROJECT

MUSIC STORE ANALYSIS

By Neha Soni

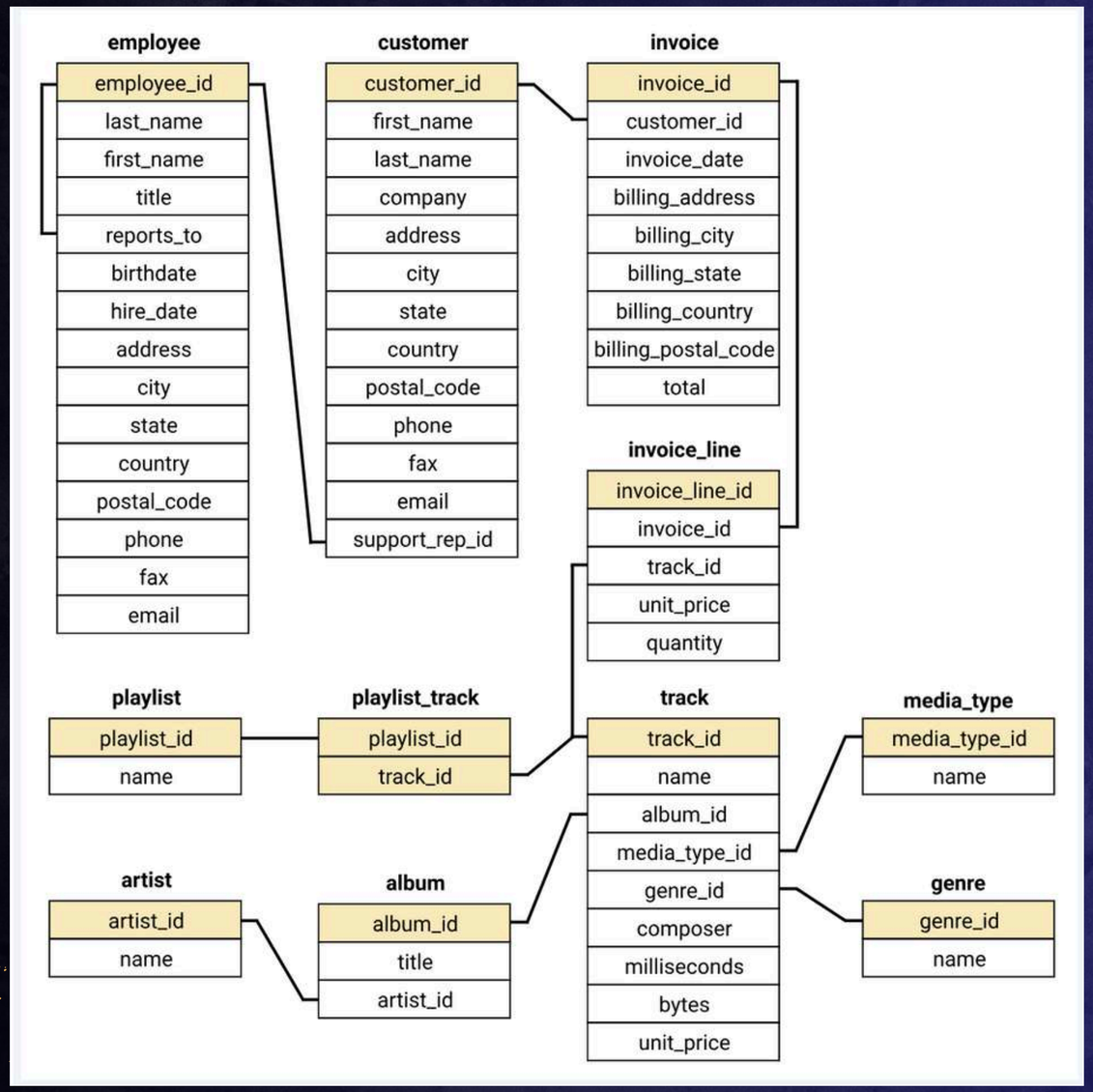


OBJECTIVE

- This project aims to analyze a digital music store database using SQL, providing stakeholders with valuable insights for decision making.

- Through SQL queries, it addresses questions genre performance, revenue, etc. which can help the music store understand its growth by answering simple questions.

DATABASE SCHEMA



LEVEL OF QUESTIONS

Easy

Queries Includes

Select, Group By, Order By, Limit

Moderate

Queries Includes

Joins, Sub-queries

Advance

Queries Includes

CTE (Common Table Expression)

EASY

QUESTION 1

Who is the senior most employee based on job title?

Query

```
SELECT first_name, last_name, title
FROM employee
ORDER BY levels DESC
LIMIT 1;
```

Output

	first_name	last_name	title
▶	Andrew	Adams	General Manager

EASY

QUESTION 2

Which countries have the most Invoices?

Query

```
SELECT billing_country AS Country, COUNT(invoice_id) AS Invoices
FROM invoice
GROUP BY billing_country
ORDER BY count(invoice_id) DESC;
```

Output

Country	Invoices
USA	131
Canada	76
Brazil	61
France	50
Germany	41
Czech Republic	30
Portugal	29
United Kingdom	28
India	21
Ireland	13
Chile	13
Finland	11
Spain	11
Poland	10

EASY

QUESTION 3

What are top 3 values of total invoice and from which country?

Query

```
SELECT ROUND(total,2) AS Total_Invoice, billing_country AS Country
FROM invoice
ORDER BY total DESC
LIMIT 3;
```

Output

Total_Invoice	Country
23.76	France
19.8	Canada
19.8	Czech Republic

EASY

QUESTION 4

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.

Query

```
SELECT billing_city as City, ROUND(SUM(total),2) as Invoice_total
FROM invoice
GROUP BY billing_city
ORDER BY count(total) DESC
LIMIT 1;
```

Output

	City	Invoice_total
▶	Prague	273.24

MODERATE

QUESTION 1

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.

Query

```
SELECT
  customer.first_name AS first_name,
  customer.last_name AS last_name,
  ROUND(SUM(invoice.total),2) AS Amount_spent
FROM customer
JOIN invoice ON customer.customer_id = invoice.customer_id
GROUP BY 1,2
ORDER BY 3 DESC
LIMIT 1;
```

Output

	first_name	last_name	Amount_spent
▶	FrantiÅ¡ek	WichterlovÃ¡	144.54

MODERATE

QUESTION 2

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.

Query

```
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;
```

Output

email	first_name	last_name
aaronmitchell@yahoo.ca	Aaron	Mitchell
alero@uol.com.br	Alexandre	Rocha
astrid.gruber@apple.at	Astrid	Gruber
bjorn.hansen@yahoo.no	BjÃ¸rn	Hansen
camille.bernard@yahoo.fr	Camille	Bernard
daan_peeters@apple.be	Daan	Peeters
diego.gutierrez@yahoo.ar	Diego	GutiÃ©rrez
dmiller@comcast.com	Dan	Miller
dominiquelefebvre@gmail.com	Dominique	Lefebvre
edfrancis@yachoo.ca	Edward	Francis
eduardo@woodstock.com.br	Eduardo	Martins

MODERATE

QUESTION 3

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.

Query

```
SELECT artist.name AS Artist_name, COUNT(artist.artist_id) AS Total_tracks
FROM track
JOIN album2 ON track.album_id = album2.album_id
JOIN artist ON album2.artist_id = artist.artist_id
JOIN genre ON track.genre_id = genre.genre_id
WHERE genre.name LIKE 'ROCK'
GROUP BY artist.artist_id, artist.name
ORDER BY Total_tracks DESC
LIMIT 10;
```

Output

Artist_name	Total_tracks
AC/DC	18
Aerosmith	15
Audioslave	14
Led Zeppelin	14
Alanis Morissette	13
Alice In Chains	12
Frank Zappa & Captain Beefheart	9
Accept	4

MODERATE

QUESTION 4

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.

Query

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

Output

Track_name	milliseconds
How Many More Times	711836
Advance Romance	677694
Sleeping Village	644571
You Shook Me(2)	619467
Talkin' 'Bout Women Obviously	589531
Stratus	582086
No More Tears	555075
The Alchemist	509413
Wheels Of Confusion / The Straightener	494524

ADVANCE

QUESTION 1

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

Query

```
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 invoice_line.track_id = track.track_id
    JOIN album2 ON track.album_id = album2.album_id
    JOIN artist ON album2.artist_id = artist.artist_id
    GROUP BY 1,2
    ORDER BY 3 DESC
    LIMIT 1
)
SELECT
     customer.customer_id,
     customer.first_name,
     customer.last_name,
     best_selling_artist.artist_name,
     ROUND(SUM(invoice_line.quantity*invoice_line.unit_price),2) AS Total_spent
FROM invoice
JOIN customer ON invoice.customer_id= customer.customer_id
JOIN invoice_line ON invoice.invoice_id = invoice_line.invoice_id
JOIN track ON invoice_line.track_id = track.track_id
JOIN album2 ON track.album_id = album2.album_id
JOIN best_selling_artist ON best_selling_artist.artist_id = album2.artist_id
GROUP BY customer.customer_id, customer.first_name, customer.last_name, best_selling_artist.artist_name
ORDER BY Total_spent DESC;
```

ADVANCE

Output

customer_id	first_name	last_name	artist_name	Total_spent
54	Steve	Murray	AC/DC	17.82
53	Phil	Hughes	AC/DC	10.89
21	Kathy	Chase	AC/DC	10.89
49	StanisÅ‚aw	WÃ³jcik	AC/DC	9.9
1	LuÃ-s	GonÃ§alves	AC/DC	7.92
24	Frank	Ralston	AC/DC	7.92
31	Martha	Silk	AC/DC	3.96
16	Frank	Harris	AC/DC	2.97
42	Wyatt	Girard	AC/DC	2.97
6	Helena	HolÃ½	AC/DC	2.97
38	Niklas	SchrÃ¶der	AC/DC	2.97
35	Madalena	Sampaio	AC/DC	2.97
44	Terhi	HÃ¤mÃ¤lÃ…	AC/DC	2.97

ADVANCE

QUESTION 2

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.

Query

```sql
WITH customer_with_country AS (
    SELECT customer.country, customer.customer_id, customer.first_name, customer.last_name, SUM(invoice.total) AS amount_spent,
    ROW_NUMBER() OVER(PARTITION BY customer.country ORDER BY SUM(invoice.total) DESC) AS Row_no
    FROM customer
    JOIN invoice ON customer.customer_id = invoice.customer_id
    GROUP BY 1, 2, 3, 4
    ORDER BY 1 ASC, amount_spent DESC
)
SELECT *
FROM customer_with_country
WHERE Row_no <= 1;
```

ADVANCE

Output

country	customer_id	first_name	last_name	amount_spent	Row_no
Argentina	56	Diego	GutiÃ©rrez	39.6	1
Australia	55	Mark	Taylor	81.18	1
Austria	7	Astrid	Gruber	69.3	1
Belgium	8	Daan	Peeters	60.38999999999999	1
Brazil	1	LuÃ-s	GonÃ§alves	108.89999999999998	1
Canada	3	FranÃ§ois	Tremblay	99.99	1
Chile	57	Luis	Rojas	97.02000000000001	1
Czech Republic	5	FrantiÅ¡ek	WichterlovÃ¡	144.54000000000002	1
Denmark	9	Kara	Nielsen	37.61999999999999	1
Finland	44	Terhi	HÃ¤mÃ¤lÃ¤inen	79.2	1
France	42	Wyatt	Girard	99.99	1
Germany	37	Fynn	Zimmermann	94.05000000000001	1

THANK YOU