SQL Project

Musical Store Analysis

By: Sanjay Dutt Sharma
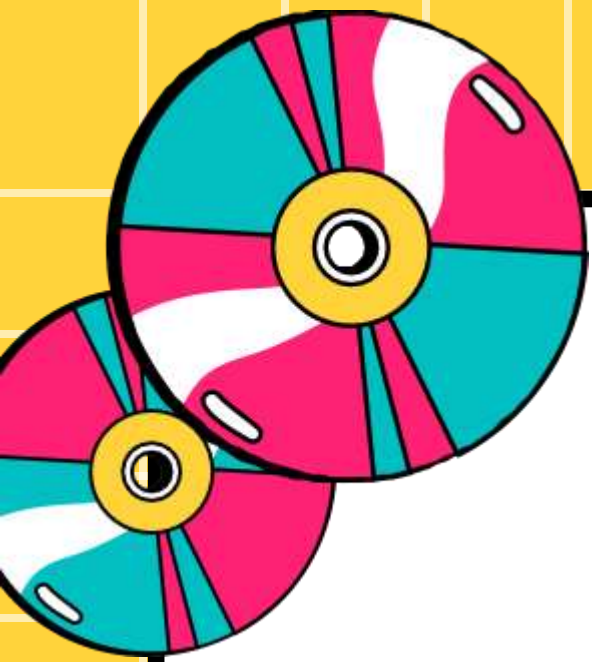
Objective

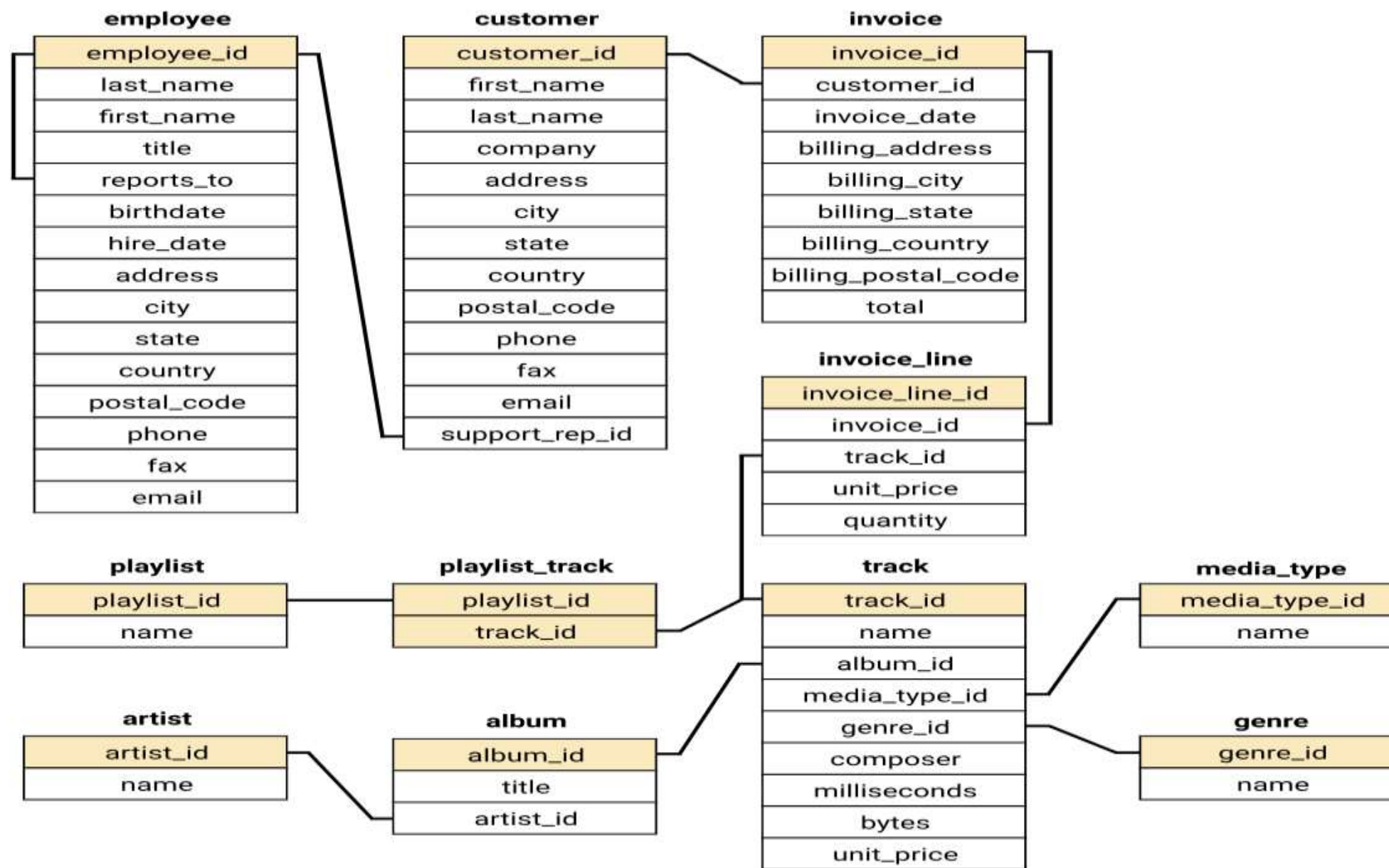|	
01	**Market Penetration**
02	**Customer Behaviour**
03	**Revenue Optimization**
04	**Trend Forecasting**
05	**Customer Segmentation**

Expected Outcome: Deliver actionable insights to stakeholders, enabling data-driven decisions for sustainable business growth and improved customer satisfaction in the competitive music retail landscape.

Database Schema

employee
- employee_id
- last_name
- first_name
- title
- reports_to
- birthdate
- hire_date
- address
- city
- state
- country
- postal_code
- phone
- fax
- email

customer
- customer_id
- first_name
- last_name
- company
- address
- city
- state
- country
- postal_code
- phone
- fax
- email
- support_rep_id

invoice
- invoice_id
- customer_id
- invoice_date
- billing_address
- billing_city
- billing_state
- billing_country
- billing_postal_code
- total

invoice_line
- invoice_line_id
- invoice_id
- track_id
- unit_price
- quantity

playlist
- playlist_id
- name

playlist_track
- playlist_id
- track_id

track
- track_id
- name
- album_id
- media_type_id
- genre_id
- composer
- milliseconds
- bytes
- unit_price

media_type
- media_type_id
- name

genre
- genre_id
- name

artist
- artist_id
- name

album
- album_id
- title
- artist_id

Level of Questions

EASY (Select, Group By, Order By, Limit)

MODERATE (Select, Joins, Order By, Limit)

ADVANCE (CTE'S)
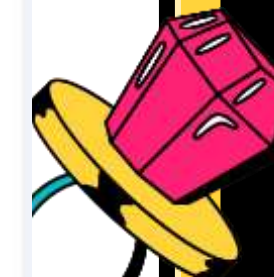
Q1: Who is the senior most employee based on job title?

QUERY

```sql
SELECT * FROM employee
ORDER BY levels desc
LIMIT 1;
```

OUTPUT

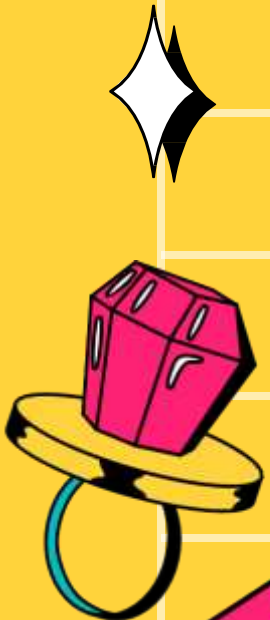| employee_id	last_name	first_name
[PK] character varying (50)	character	character
9	Madan	Mohan

Easy

Q2: Which countries have the most Invoices?

QUERY

```sql
SELECT billing_country, COUNT(invoice_id)
FROM invoice
GROUP BY billing_country
ORDER BY COUNT(invoice_id) DESC
```

OUTPUT

Data Output   Messages   Notifications

	billing_country character varying (30)	count bigint
1	USA	131
2	Canada	76
3	Brazil	61
4	France	50
5	Germany	41
6	Czech Republic	30
7	Portugal	29
8	United Kingdom	28
9	India	21
10	Chile	13
11	Ireland	13

Easy

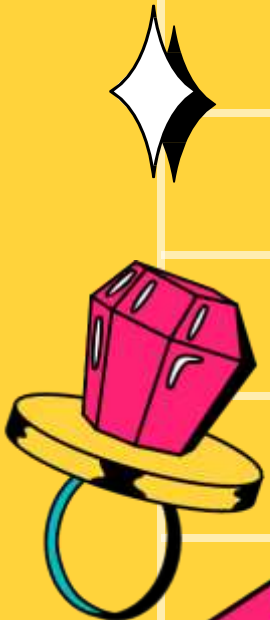## Q3: What are top 3 values of total invoice?

QUERY

```
SELECT total FROM invoice
ORDER BY total DESC
LIMIT 3
```

OUTPUT

Data Output   Messages   Notifications

	total double precision
1	23.759999999999998
2	19.8
3	19.8

Easy

Q 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

```sql
SELECT billing_city, SUM(total) AS TOTAL_INVOICE
FROM invoice
GROUP BY billing_city
ORDER BY TOTAL_INVOICE DESC
```

OUTPUT

Data Output   Messages   Notifications

	billing_city character varying (30)	total_invoice double precision
1	Prague	273.24000000000007
2	Mountain View	169.29
3	London	166.32
4	Berlin	158.4
5	Paris	151.47
6	São Paulo	129.69
7	Dublin	114.83999999999997
8	Delhi	111.86999999999999
9	São José dos Campos	108.89999999999998
10	Brasília	106.91999999999999
11	Lisbon	102.96000000000001

Easy

Q 5. 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
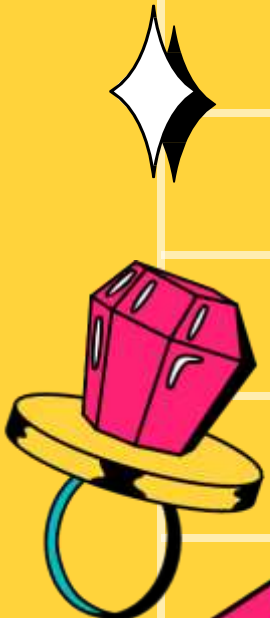
QUERY

```sql
SELECT C.customer_id , C.first_name,C.last_name, SUM(I.total) AS TOTAL_AMOUNT
FROM customer AS C
INNER JOIN invoice AS I
ON C.customer_id = I.customer_id
GROUP BY  C.customer_id , C.first_name,C.last_name
ORDER BY TOTAL_AMOUNT DESC
LIMIT 1
```

OUTPUT

Data Output   Messages   Notifications

	customer_id [PK] integer	first_name character	last_name character	total_amount double precision
1	5	R	Madhav	144.54000000000002

Easy

Q1. 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

```sql
SELECT Distinct c.email,c.first_name, c.last_name,g.name AS genre
FROM customer AS c
JOIN invoice AS i ON c.customer_id = i.customer_id
JOIN invoice_line AS il ON I.invoice_id = il.invoice_id
JOIN track AS t ON il.track_id = t.track_id
JOIN genre AS g ON t.genre_id = g.genre_id
WHERE g.name = 'Rock'
Order BY c.email ASC
```

OUTPUT

Data Output   Messages   Notifications

	email character varying (50)	first_name character	last_name character	genre character varying (120)
1	aaronmitchell@yahoo.ca	Aaron	Mitchell	Rock
2	alero@uol.com.br	Alexandre	Rocha	Rock
3	astrid.gruber@apple.at	Astrid	Gruber	Rock
4	bjorn.hansen@yahoo.no	Bjørn	Hansen	Rock
5	camille.bernard@yahoo.fr	Camille	Bernard	Rock
5	daan_peeters@apple.be	Daan	Peeters	Rock
7	diego.gutierrez@yahoo.ar	Diego	Gutiérrez	Rock
3	dmiller@comcast.com	Dan	Miller	Rock
9	dominiquelefebvre@gmail.c...	Dominique	Lefebvre	Rock

Moderate

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
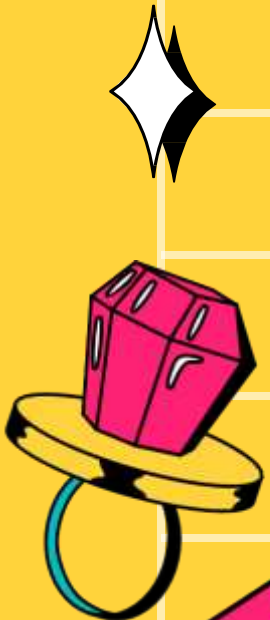
QUERY

```sql
SELECT a.name,COUNT(t.track_id) AS total_track_count
FROM artist AS a
JOIN album AS al ON a.artist_id = al.artist_id
JOIN track as t ON al.album_id = t.album_id
JOIN genre as g ON t.genre_id = g.genre_id
WHERE g.name LIKE 'Rock'
GROUP BY a.name
ORDER BY total_track_count DESC
LIMIT 10
```

OUTPUT

Data Output   Messages   Notifications

	name character varying (120)	total_track_count bigint
1	Led Zeppelin	114
2	U2	112
3	Deep Purple	92
4	Iron Maiden	81
5	Pearl Jam	54
6	Van Halen	52
7	Queen	45
8	The Rolling Stones	41
9	Creedence Clearwater Revival	40
10	Kiss	35

Moderate

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

QUERY

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

OUTPUT

Data Output   Messages   Notifications

	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
11	The Long Patrol	2925008
12	The Magnificent Warriors	2924716
13	The Living Legend, Pt. 1	2924507

Moderate
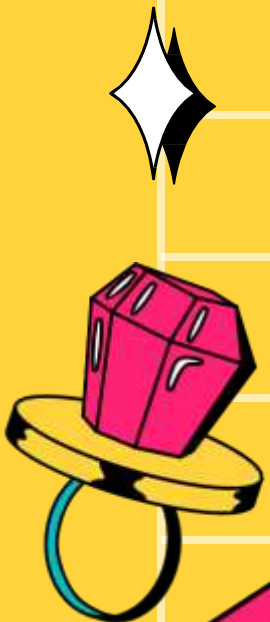
Q1. Find how much amount spent by each customer on artists? Write a query to return customer name, artist name and total spent

QUERY

```sql
SELECT c.first_name , a.name, ROUND(SUM(il.unit_price * il.quantity)::numeric ,2) AS Total_Spent
FROM customer AS c
INNER JOIN invoice AS i ON c.customer_id =i.customer_id
INNER JOIN invoice_line AS il ON i.invoice_id = il.invoice_id
INNER JOIN track AS t ON il.track_id = t.track_id
INNER JOIN album AS al ON t.album_id = al.album_id
INNER JOIN artist AS a ON al.artist_id = a.artist_id
GROUP BY c.first_name,a.name
ORDER BY total_Spent DESC
```

OUTPUT

Data Output   Messages   Notifications

	first_name character	name character varying (120)	total_spent numeric
1	Hugh	Queen	27.72
2	Wyatt	Frank Sinatra	23.76
3	R	Kiss	19.80
4	Helena	Red Hot Chili Peppers	19.80
5	François	The Who	19.80
6	Robert	Creedence Clearwater Revival	19.80
7	Aaron	James Brown	19.80
8	Mark	The Clash	19.80
9	Niklas	Queen	18.81
10	Heather	House Of Pain	18.81
11	Hugh	Nirvana	18.81
12	Richard	Marvin Gaye	17.82
13	Ladislav	The Cult	17.82

Advance

Q2. 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

QUERY _ ✕

```sql
WITH popular_music_genre AS (
    SELECT c.country,g.name, g.genre_id, COUNT(il.quantity) AS purchases,
    ROW_NUMBER() OVER (PARTITION BY c.country ORDER BY COUNT(il.quantity)DESC) AS Row_no
    FROM invoice_line AS il
    JOIN invoice AS i ON il.invoice_id = i.invoice_id
    JOIN customer AS c ON i.customer_id = c.customer_id
    JOIN track AS t ON t.track_id = il.track_id
    JOIN genre AS g ON g.genre_id = t.genre_id
    GROUP BY c.country , g.name,g.genre_id
    ORDER BY purchases DESC,c.country ASC
)

SELECT * FROM popular_music_genre WHERE Row_no <=1
```

OUTPUT _ ✕

Data Output   Messages   Notifications

	country character varying (50)	name character varying (120)	genre_id character varying (50)	purchases bigint	row_no bigint
1	USA	Rock	1	561	1
2	Canada	Rock	1	333	1
3	France	Rock	1	211	1
4	Brazil	Rock	1	205	1
5	Germany	Rock	1	194	1
6	United Kingdom	Rock	1	166	1
7	Czech Republic	Rock	1	143	1
8	Portugal	Rock	1	108	1

Advance

Q3. 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
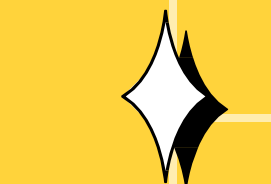
QUERY _ ✕

```
WITH customer_spend AS (
SELECT c.customer_id,c.first_name,c.last_name, i.billing_country , ROUND(sum(total)::NUMERIC ,2)AS total_spend,
    ROW_NUMBER() OVER (PARTITION BY i.billing_country ORDER BY sum(total) DESC) AS Row_no
    FROM invoice AS i
    JOIN customer AS c ON c.customer_id = i.customer_id
    GROUP BY c.customer_id,c.first_name,c.last_name,i.billing_country
    ORDER BY total_spend DESC, 4 ASC)


SELECT * FROM customer_spend WHERE Row_no <=1
```
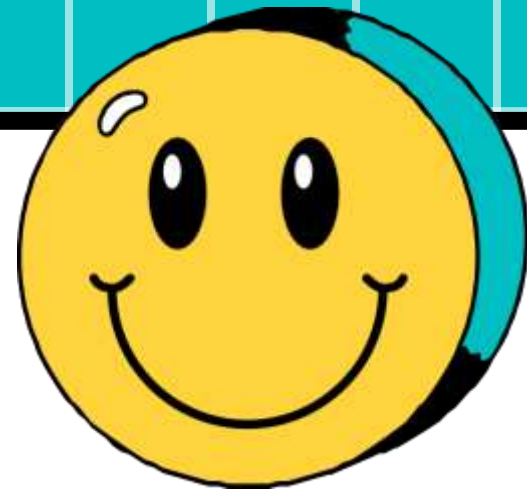
OUTPUT _ ✕

Data Output   Messages   Notifications

	customer_id integer	first_name character		last_name character		billing_country character varying (30)	total_spend numeric	row_no bigint
1	5	R		Madhav		Czech Republic	144.54	1
2	46	Hugh	...	O'Reilly		Ireland	114.84	1
3	58	Manoj		Pareek		India	111.87	1
4	1	Luís		Gonçalves	...	Brazil	108.90	1
5	34	João	...	Fernandes	...	Portugal	102.96	1
6	3	François		Tremblay	...	Canada	99.99	1
7	42	Wyatt	...	Girard		France	99.99	1
8	50	Enrique	...	Muñoz	...	Spain	98.01	1
9	53	Phil		Hughes		United Kingdom	98.01	1
10	17	Jack	...	Smith		USA	98.01	1
11	57	Luis		Rojas		Chile	97.02	1

Advance

Thank YOU !