SQL Project

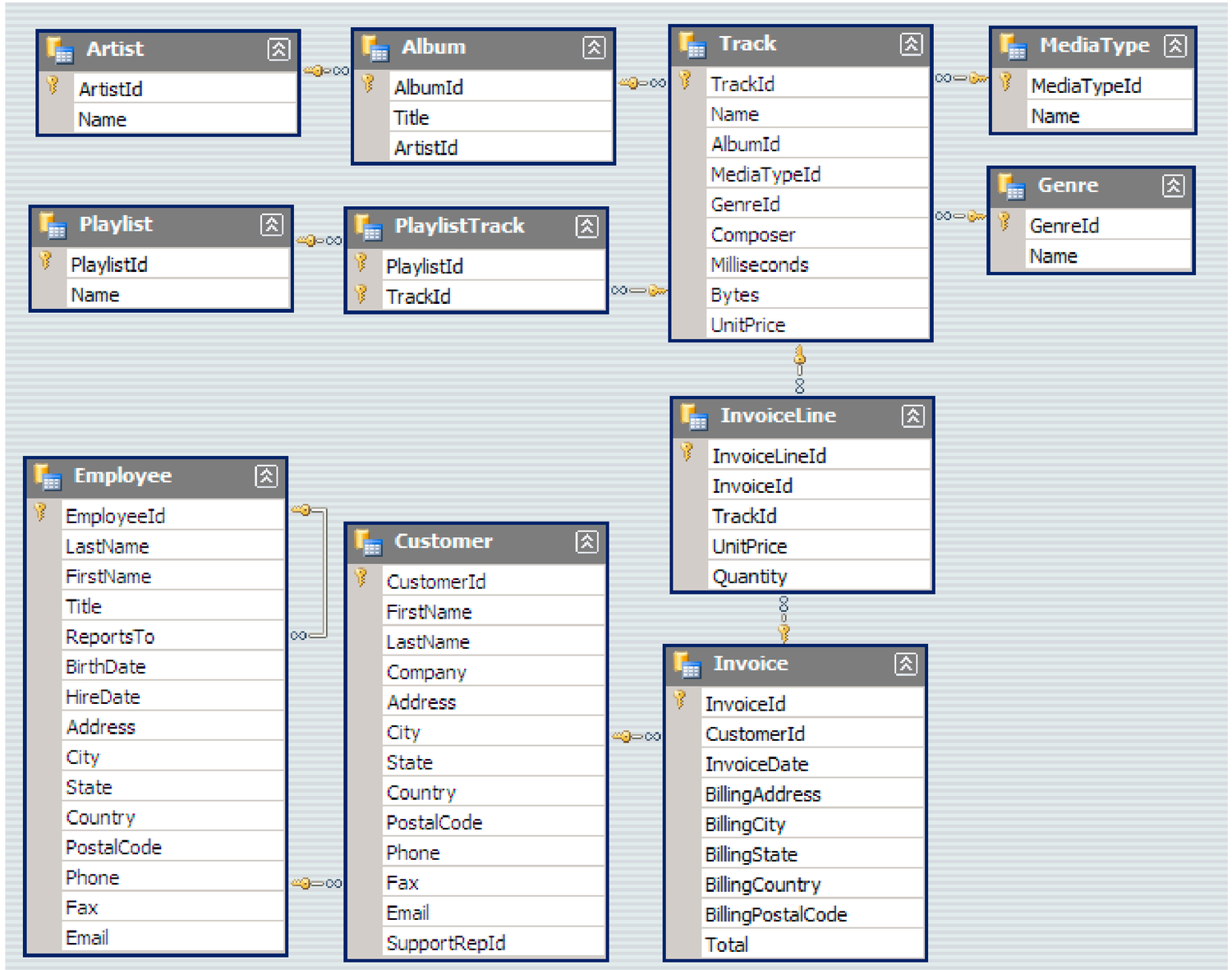
Music Store Analysis

- Poonam Bhosle

Objective

- The Primary objective of the music store is to achieve sustainable business growth while addressing existing challenges.
- We need to examine the dataset with SQL and help the music store to understand its business growth by answering simple questions.

Music Playlist Database Schema



Division Of Questions

Easy

Queries Include:

SELECT, GROUP BY, ORDER BY,LIMIT, DESC

Moderate

Queries Include:

JOINS, GROUP BY, ORDER BY,LIMIT

Advance

Queries Include:

CTE (COMMON TABLE EXPRESSIONS)

Easy

Question 1

Who is the senior most employee based on job title?

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

	employee_id	last_name	first_name	title	reports_to	levels
▸	1	Adams	Andrew	General Manager	9	L6

Question 2

Which countries have the most Invoices?

```
SELECT COUNT(*) AS c , billing_country FROM invoice
GROUP BY billing_country
ORDER BY c DESC;
```

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

Question 3

What are top 3 values of total invoice?

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

Total_Invoice
23.76
19.8
19.8

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.

```sql
SELECT ROUND(SUM(total),2) AS invoice_total,
billing_city FROM invoice
GROUP BY billing_city
ORDER BY invoice_total DESC
LIMIT 1;
```

	invoice_total	billing_city
	273.24	Prague

Question 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.

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

customer_id	first_name	last_name	invoice_total
5	FrantiÅiek	WichterlovÃ¡	144.54000000000002

Moderate

Question 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

```sql
SELECT DISTINCT customer.email, customer.first_name, customer.last_name
FROM customer
JOIN invoice ON customer.customer_id = invoice.customer_id
JOIN invoice_line ON invoice.invoice_id = invoice_line.invoice_id
JOIN track ON invoice_line.track_id = track.track_id
JOIN genre ON track.genre_id = genre.genre_id
WHERE genre.name LIKE 'Rock'
ORDER BY customer.email;
```

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

Question 2

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

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

artist_id	name	number_of_songs
1	AC/DC	18
3	Aerosmith	15
8	Audioslave	14
22	Led Zeppelin	14
4	Alanis Morissette	13
5	Alice In Chains	12
23	Frank Zappa & Captain Beefheart	9
2	Accept	4

Question 3

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;
```

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

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.

```sql
WITH artist_sales 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 album2 ON album2.album_id = track.album_id
        JOIN artist ON artist.artist_id = album2.artist_id
    GROUP BY artist.artist_id, artist.name
)
SELECT
    c.customer_id, c.first_name, c.last_name, a.artist_name,
    ROUND(SUM(il.unit_price * il.quantity),2) 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 album2 alb ON alb.album_id = t.album_id
    JOIN artist_sales a ON a.artist_id = alb.artist_id
GROUP BY c.customer_id, c.first_name, c.last_name, a.artist_name
ORDER BY amount_spent DESC
```

Output

customer_id	first_name	last_name	artist_name	amount_spent
54	Steve	Murray	AC/DC	17.82
15	Jennifer	Peterson	Aerosmith	14.85
55	Mark	Taylor	Aerosmith	14.85
13	Fernanda	Ramos	AntÃ´nio Carlos Jobim	13.86
2	Leonie	KÃ¶hler	Audioslave	13.86
30	Edward	Francis	Alanis Morissette	12.87
52	Emma	Jones	Alanis Morissette	12.87
34	JoÃ£o	Fernandes	Alanis Morissette	12.87
25	Victor	Stevens	Alice In Chains	11.88
53	Phil	Hughes	AC/DC	10.89
21	Kathy	Chase	AC/DC	10.89
49	StanisÅ‚aw	WÃ³jcik	Buddy Guy	10.89
49	StanisÅ‚aw	WÃ³jcik	AC/DC	9.9

Question 2

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 Genre.

Query

```sql
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
    ORDER BY customer.country ASC , COUNT(invoice_line.quantity) DESC
)
SELECT * FROM popular_genre
WHERE RowNo <= 1;
```

Output

purchases	country	name	genre_id	RowNo
1	Argentina	Rock	1	1
18	Australia	Rock	1	1
6	Austria	Rock	1	1
5	Belgium	Rock	1	1
26	Brazil	Rock	1	1
57	Canada	Rock	1	1
7	Chile	Rock	1	1
14	Czech Republic	Rock	1	1
6	Denmark	Rock	1	1
6	Finland	Rock	1	1
26	France	Rock	1	1
28	Germany	Rock	1	1
4	Hungary	Rock	1	1
13	India	Rock	1	1

Question 3

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.customer_id , first_name , last_name , billing_country , ROUND(SUM(total),2) 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 customer.customer_id , first_name , last_name , billing_country
    ORDER BY billing_country ASC , SUM(total) DESC
)
SELECT * FROM customer_with_country
WHERE RowNo <= 1;
```

Output

customer_id	first_name	last_name	billing_country	total_spending	RowNo
56	Diego	GutiÃ©rrez	Argentina	39.6	1
55	Mark	Taylor	Australia	81.18	1
7	Astrid	Gruber	Austria	69.3	1
8	Daan	Peeters	Belgium	60.39	1
1	LuÃ-s	GonÃ§alves	Brazil	108.9	1
3	FranÃ§ois	Tremblay	Canada	99.99	1
57	Luis	Rojas	Chile	97.02	1
5	FrantiÅiek	WichterlovÃi	Czech Republic	144.54	1
9	Kara	Nielsen	Denmark	37.62	1
44	Terhi	HÃ¤mÃ¤lÃ¤inen	Finland	79.2	1
42	Wyatt	Girard	France	99.99	1
37	Fynn	Zimmermann	Germany	94.05	1
45	Ladislav	KovÃics	Hungary	78.21	1
58	Manoj	Pareek	India	111.87	1

Thank You!