

MUSIC STORE ANALYSIS

SQL PROJECT

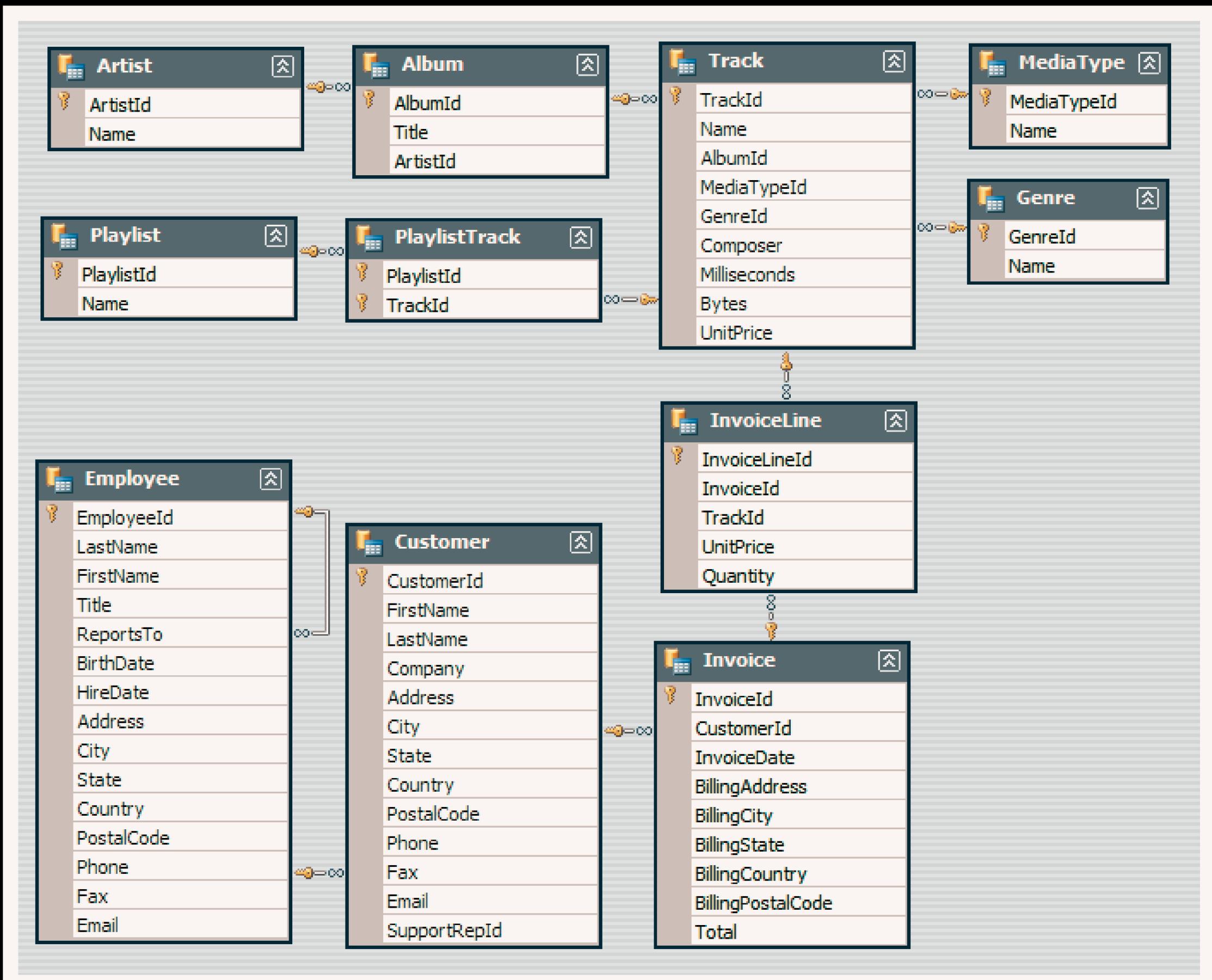


BY HARDIK THAKOR

OBJECTIVE

- The main goal of the music store is to attain sustainable business development while confronting current obstacles.
- We must analyze the dataset using SQL to assist the music store in comprehending its business expansion by responding to straightforward inquiries.

TABLE SCHEMA



LEVEL OF QUERIES



EASY

Includes : Select, Group By,
Order By, Limit, Desc



MODERATE

Includes : Joins, Group By,
Order By, Limit



ADVANCE

Includes : CTE (Common Table
Expression),
Window Functions

EASY

1. Who is the senior most employee based on job title?

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

	employee_id	last_name	first_name	title
▶	1	Adams	Andrew	General Manager

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

3. What are top 3 values of total invoice?

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

	total
▶	23.759999999999998
	19.8
	19.8

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.

```
SELECT SUM(total) AS Invoice_Total, billing_city FROM invoice  
GROUP BY billing_city  
ORDER BY Invoice_Total DESC  
LIMIT 1;
```

	Invoice_Total	billing_city
▶	273.24000000000007	Prague

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

	customer_id	first_name	last_name	SUM(i.total)
▶	5	František	Wichterlová	144.5400000000002

MODERATE

6. 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 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
WHERE track_id IN (
    SELECT track_id FROM track AS t
    JOIN genre AS g ON t.genre_id = g.genre_id
    WHERE g.name LIKE 'ROCK'
)
ORDER BY 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©rez
	dmiller@comcast.com	Dan	Miller
	dominiquelefrevre@gmail.com	Dominique	Lefebvre
	edfrancis@yahoo.ca	Edward	Francis
	eduardo@woodstock.com.hr	Eduardo	Martins

7. 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 ar.artist_id, ar.name, COUNT(ar.artist_id) AS  
Number_Of_Songs FROM track AS t  
JOIN album2 AS al ON al.album_id = t.album_id  
JOIN artist AS ar ON ar.artist_id = al.artist_id  
JOIN genre AS g ON g.genre_id = t.genre_id  
WHERE g.name LIKE 'Rock'  
GROUP BY ar.artist_id, ar.name  
ORDER BY COUNT(ar.artist_id) 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

8. 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
	Wheels Of Confusion / The Straightener	494524
	Book Of Thel	494393

ADVANCE

9. Find how much amount spent by each customer on artists?

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 album2 ON album2.album_id = track.album_id
    JOIN artist ON artist.artist_id = album2.artist_id
    GROUP BY artist.artist_id, artist.name
    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 album2 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;
```

	customer_id	first_name	last_name	artist_name	amount_spent
▶	54	Steve	Murray	AC/DC	17.82
	53	Phil	Hughes	AC/DC	10.89
	21	Kathy	Chase	AC/DC	10.89
	49	StanisÅaw	WÅjciech	AC/DC	9.9
	1	LuÅs	GonÃ§alves	AC/DC	7.920000000000001
	24	Frank	Ralston	AC/DC	7.920000000000001
	31	Martha	Silk	AC/DC	3.96
	16	Frank	Harris	AC/DC	2.9699999999999998
	42	Wyatt	Girard	AC/DC	2.9699999999999998
	6	Helena	HolÃ½	AC/DC	2.9699999999999998
	38	Niklas	SchrÃ¶der	AC/DC	2.9699999999999998

10. 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;
```

	purchases_per_genre	country	name	genre_id
▶	1	Argentina	Rock	1
	18	Australia	Rock	1
	6	Austria	Rock	1
	5	Belgium	Rock	1
	26	Brazil	Rock	1
	57	Canada	Rock	1
	7	Chile	Rock	1
	14	Czech Republic	Rock	1
	6	Denmark	Rock	1
	6	Finland	Rock	1
	26	France	Rock	1

11. 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 customer_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 customer_with_country WHERE RowNo <=1;
```

	customer_id	first_name	last_name	billing_country	total_spending	RowNo
▶	56	Diego	Gutiérrez	Argentina	39.6	1
	55	Mark	Taylor	Australia	81.18	1
	7	Astrid	Gruber	Austria	69.3	1
	8	Daan	Peeters	Belgium	60.38999999999999	1
	1	Luís	Gonçalves	Brazil	108.8999999999998	1
	3	François	Tremblay	Canada	99.99	1
	57	Luis	Rojas	Chile	97.0200000000001	1
	5	František	Wichterlová	Czech Republic	144.5400000000002	1
	9	Kara	Nielsen	Denmark	37.61999999999999	1
	44	Terhi	Härmänen	Finland	79.2	1
	42	Wvatt	Girard	France	99.99	1



THANK YOU

**Those who were seen dancing were thought to be insane by those who could not
feel the music**