



# MUSIC STORE ANALYSIS

Using SQL

BY-Prasenjit Barman

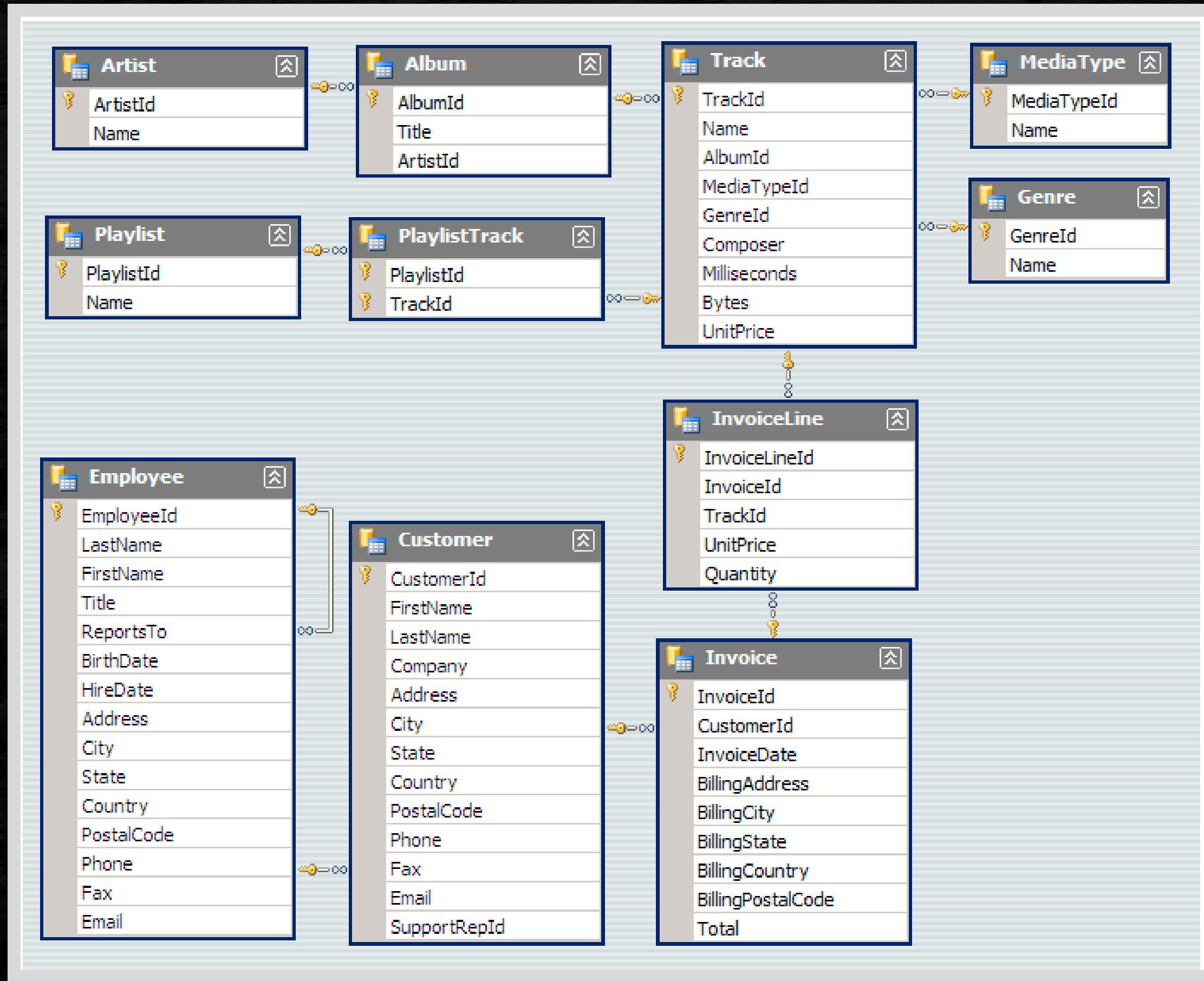


# OBJECTIVE

To Leverage SQL Data Analysis Techniques To Examine The Music Sales Dataset, Providing Actionable Insights That Facilitate A Comprehensive Understanding of Business Growth And Performance For The Store. Through Answering Key Questions, This Presentation Aims To Uncover Trends, Patterns, And Opportunities Within The Data To Inform Strategic Decision-Making And Optimize Sales Performance.



# DATABASE SCHEMA



# Q1. Who is the senior most employee based on job title?

## Query

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

## Output

employee_id	last_name	first_name	title	reports_to	levels	birthdate	hire_date
9	Madan	Mohan	Senior General Manager	NULL	7.00	1961-01-26 00:00:00.0000000	2016-01-14 00:00:00.000000



# Q2. Which countries have the most Invoices?

## Query

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

## Output

	Results	Messages
	count	billing_country
1	131	USA
2	76	Canada
3	61	Brazil
4	50	France
5	41	Germany
6	30	Czech Republic
7	29	Portugal
8	28	United Kingdom
9	21	India
10	13	Ireland
11	12	Netherlands



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

## Query

```
SELECT TOP 3 invoice_id, total FROM invoice  
ORDER BY total DESC;
```



## Output

	invoice_id	total
1	183	23.7600002288818
2	31	19.7999992370605
3	92	19.7999992370605



**Q4. 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, SUM(total) AS invoice_total  
FROM invoice  
GROUP BY billing_city  
ORDER BY invoice_total DESC;
```



### Output

	billing_city	invoice_total
1	Prague	273.240000247955

**Q5. 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 TOP 1 customer.first_name, customer.last_name, SUM(invoice.total) AS total  
FROM customer  
INNER JOIN invoice ON customer.customer_id=invoice.customer_id  
GROUP BY customer.first_name, customer.last_name  
ORDER BY total DESC;
```

## Output

	first_name	last_name	total
1	František	Wichterlová	144.539998054504

**Q6. 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 first_name, last_name, email
FROM customer
INNER JOIN invoice ON customer.customer_id=invoice.customer_id
INNER JOIN invoice_line ON invoice.invoice_id=invoice_line.invoice_id
WHERE track_id IN(
    SELECT track_id FROM track
    INNER JOIN genre ON track.genre_id=genre.genre_id
    WHERE genre.name LIKE 'Rock'
)
ORDER BY email;
```



## Output

	first_name	last_name	email
1	Aaron	Mitchell	aaronmitchell@yahoo.ca
2	Alexandre	Rocha	alerio@uol.com.br
3	Astrid	Gruber	astrid.gruber@apple.at
4	Bjørn	Hansen	bjom.hansen@yahoo.no
5	Camille	Bernard	camille.bernard@yahoo.fr
6	Daan	Peeters	daan_peeters@apple.be
7	Diego	Gutiérrez	diego.gutierrez@yahoo.ar
8	Dan	Miller	dmiller@comcast.com
9	Dominique	Lefebvre	dominiquelefrevre@gmail.com
10	Edward	Francis	edfrancis@yahoo.ca

**Q7. 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 TOP 5 artist.artist_id, artist.name, COUNT(artist.artist_id) AS number_of_songs  
FROM track  
INNER JOIN album ON album.album_id=track.album_id  
INNER JOIN artist ON artist.artist_id=album.artist_id  
INNER 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;
```



## Output

	artist_id	name	number_of_songs
1	22	Led Zeppelin	114
2	150	U2	112
3	58	Deep Purple	92
4	90	Iron Maiden	81
5	118	Pearl Jam	54

**Q8. 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,milliseconds  
FROM track  
WHERE milliseconds > (  
    SELECT AVG(milliseconds) AS avg_track_length  
    FROM track)  
ORDER BY milliseconds DESC;
```



### Output

	name	milliseconds
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	Till the Cows Come Home	2927877

**Q9. 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 TOP 1 artist.artist_id, artist.name AS artist_name,
    SUM(invoice_line.unit_price*invoice_line.quantity) AS total_sales
    FROM invoice_line
    INNER JOIN track ON track.track_id = invoice_line.track_id
    INNER JOIN album ON album.album_id = track.album_id
    INNER JOIN artist ON artist.artist_id = album.artist_id
    GROUP BY artist.artist_id, artist.name
    ORDER BY total_sales DESC
)
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
INNER JOIN customer c ON c.customer_id = i.customer_id
INNER JOIN invoice_line il ON il.invoice_id = i.invoice_id
INNER JOIN track t ON t.track_id = il.track_id
INNER JOIN album alb ON alb.album_id = t.album_id
INNER JOIN best_selling_artist bsa ON bsa.artist_id = alb.artist_id
GROUP BY c.customer_id, c.first_name, c.last_name,
bsa.artist_name
ORDER BY amount_spent DESC;
```

# Output



	customer_id	first_name	last_name	artist_name	amount_spent
1	46	Hugh	O'Reilly	Queen	27.7200002670288
2	38	Niklas	Schröder	Queen	18.8100001811981
3	3	François	Tremblay	Queen	17.8200001716614
4	34	João	Femandes	Queen	16.8300001621246
5	41	Marc	Dubois	Queen	11.8800001144409
6	53	Phil	Hughes	Queen	11.8800001144409
7	47	Lucas	Mancini	Queen	10.8900001049042
8	33	Ellie	Sullivan	Queen	10.8900001049042
9	5	František	Wichterl...	Queen	3.9600003814697
10	20	Dan	Miller	Queen	3.9600003814697
11	23	John	Gordon	Queen	2.9700002861023
12	31	Martha	Silk	Queen	2.9700002861023
13	54	Steve	Murray	Queen	2.9700002861023
14	57	Luis	Rojas	Queen	1.9800001907349
15	1	Luis	Gonçalves	Queen	1.9800001907349
16	35	Madalena	Sampaio	Queen	1.9800001907349
17	36	Hannah	Schneider	Queen	1.9800001907349

**Q10. 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



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

# Output



	purchases	country	genre_name	genre_id	RowNo
1	17	Argentina	Altemative & Punk	4	1
2	34	Australia	Rock	1	1
3	40	Austria	Rock	1	1
4	26	Belgium	Rock	1	1
5	205	Brazil	Rock	1	1
6	333	Canada	Rock	1	1
7	61	Chile	Rock	1	1
8	143	Czech ...	Rock	1	1
9	24	Denmark	Rock	1	1
10	46	Finland	Rock	1	1
11	211	France	Rock	1	1
12	194	Germany	Rock	1	1
13	44	Hungary	Rock	1	1
14	102	India	Rock	1	1
15	72	Ireland	Rock	1	1
16	35	Italy	Rock	1	1
17	33	Nether...	Rock	1	1

**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.**

## Query

```
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
    INNER JOIN customer ON customer.customer_id=invoice.customer_id
    GROUP BY customer.customer_id,first_name,last_name,billing_country)
SELECT * FROM customer_with_country
WHERE RowNo<=1
ORDER BY billing_country ASC,total_spending DESC;
```



# Output



	customer_id	first_name	last_name	billing_country	total_spending	RowNo
1	56	Diego	Gutiérrez	Argentina	39.5999991893768	1
2	55	Mark	Taylor	Australia	81.1800003051758	1
3	7	Astrid	Gruber	Austria	69.299998998642	1
4	8	Daan	Peeters	Belgium	60.3899998664856	1
5	1	Luis	Gonçalves	Brazil	108.899999380112	1
6	3	François	Tremblay	Canada	99.9899981021881	1
7	57	Luis	Rojas	Chile	97.0199997425079	1
8	5	František	Wichterlová	Czech Republic	144.539998054504	1
9	9	Kara	Nielsen	Denmark	37.6199996471405	1
10	44	Terhi	Hämäläinen	Finland	79.2000005245209	1
11	42	Wyatt	Girard	France	99.9899990558624	1
12	37	Fynn	Zimmema...	Germany	94.0499994754791	1
13	45	Ladislav	Kovács	Hungary	78.2099997997284	1



THANK YOU