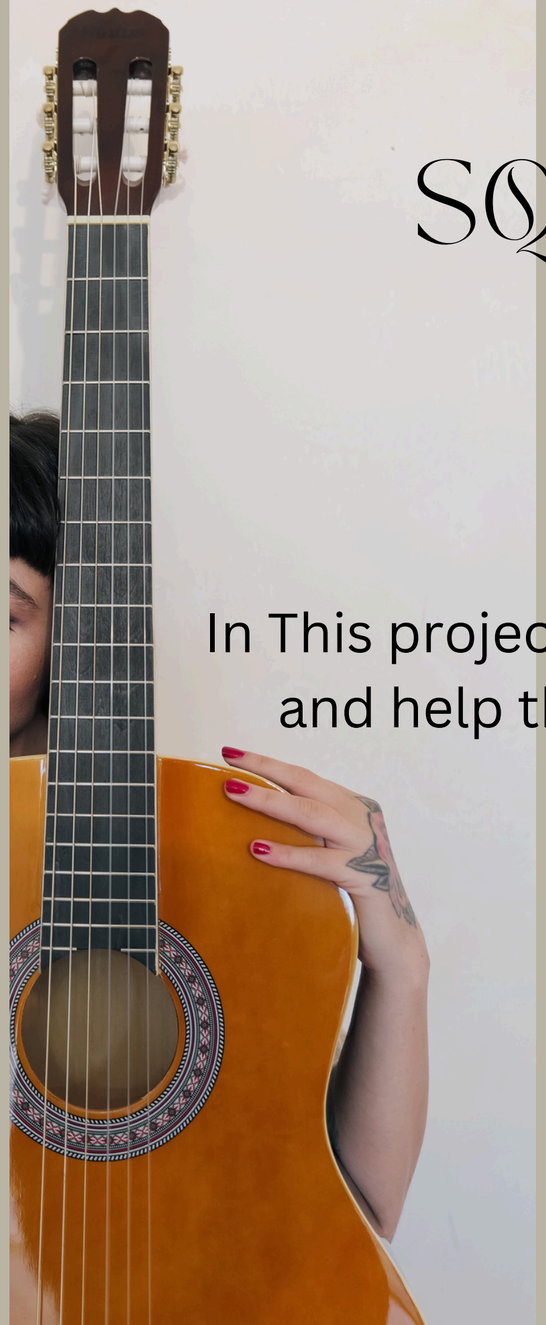
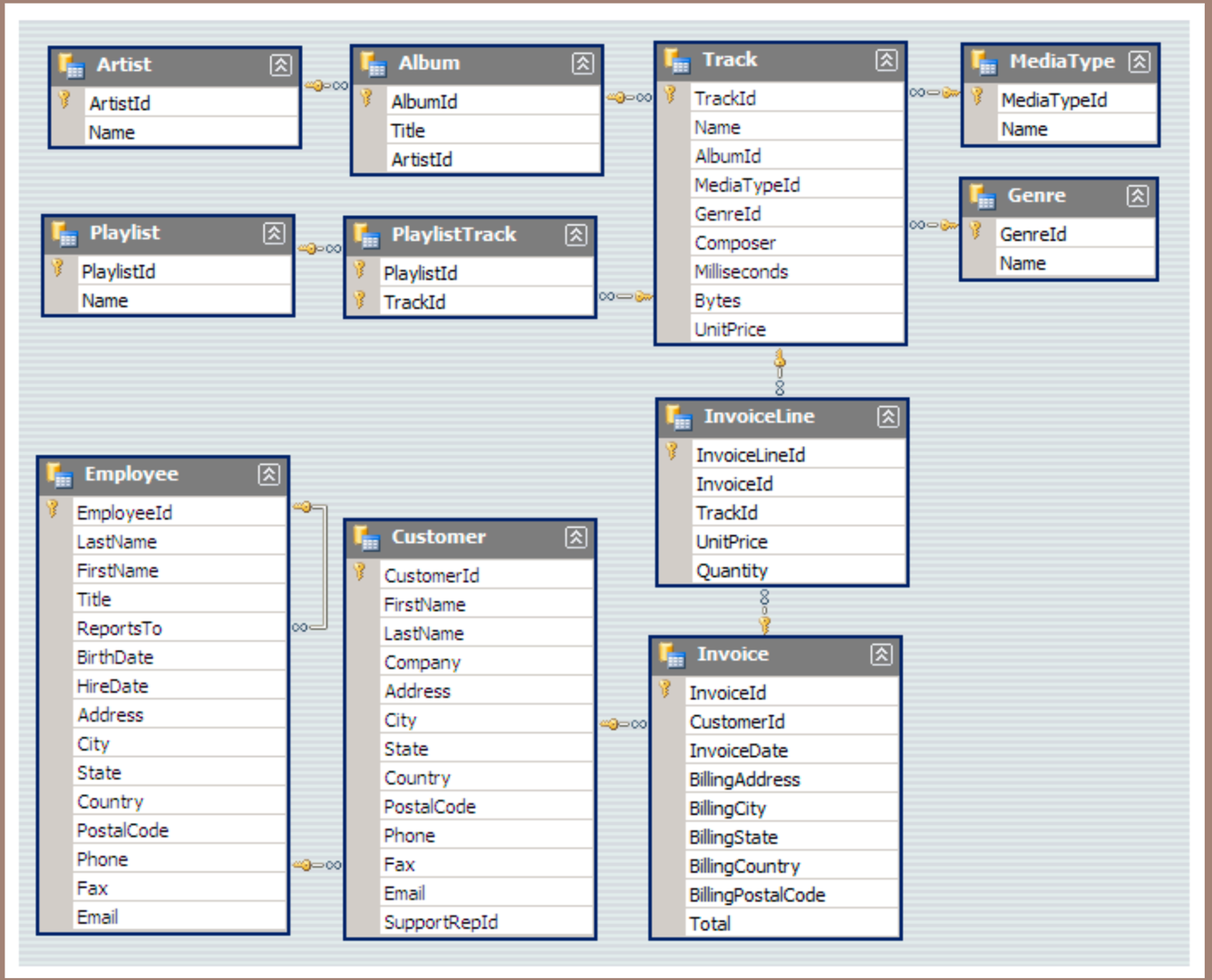
DIGITAL MUSIC STORE ANALYSIS



SQL project to analyze online music store data

In This project we analyze the music playlist database. we examine the dataset with SQL and help the store understand its business growth by answering simple questions.

Music Database Schema



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

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

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

employee_id	last_name	first_name	title	reports_to	levels	birthdate	hire_date	address	city	state	country	postal_code	phone	fax	email
9	madan	mohan	senior general menager	NULL	L7	-1936	-2003	1008 Vrinda Ave MT	Edmonton	AB	Canada	T5K 2N1	+1 (780) 428-9482	+1 (780) 428-3457	madan.mohan@chinookc

Q2: Which 5 countries have the most invoice?

```
SELECT COUNT(*) as c, billing_country
FROM invoice
GROUP BY billing_country
ORDER BY c desc;
```

Result Grid | Filter

c	billing_country
131	USA
76	Canada
61	Brazil
50	France
41	Germany

Q3: What are top 3 values of total invoice?

```
SELECT total FROM invoice
ORDER BY TOTAL DESC
limit 3
```

Result Grid

total
23.76
19.8
19.8

Q4: Which city has the best customer ? we would like to throw a promotional music festival in the city we made the most money. write a query that return 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;
```

Result Grid  Filter Rows:

invoice_total	billing_city
273.24000000000007	Prague

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 the spent the most money.

```sql
SELECT customer.customer_id, customer.first_name, customer.last_name, SUM(invoice.total) as total
FROM customer
JOIN invoice on customer.customer_id = invoice.customer_id
GROUP BY customer.customer_id
ORDER BY total desc
LIMIT 1;
```

```sql
set SQL_MODE = 'TRADITIONAL';
```

Result Grid | Filter Rows: | Export:

customer_id	first_name	last_name	total
5	FrantiÅiek	WichterlovÃ¡	144.54000000000002

Q6: Write query to return the email, first name, last name, & genre of all rock music listeners. return your list ordered alphabatically by email starting with A.

```sql
SELECT DISTINCT email, first_name, last_name
FROM customer
JOIN invoice ON customer.customer_id = invoice.customer_id
JOIN invoiceline ON invoice.invoice_id = invoiceline.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;
```

Result Grid | Filter Rows: | Expor

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

Result Grid | Filter Rows: | Expo

email	first_name	last_name
ellie.sullivan@shaw.ca	Ellie	Sullivan
emma_jones@hotmail.com	Emma	Jones
enrique_munoz@yahoo.es	Enrique	MuÃ±oz
fernadaramos4@uol.com.br	Fernanda	Ramos
fharris@google.com	Frank	Harris
fralston@gmail.com	Frank	Ralston
frantisekw@jetbrains.com	FrantiÅ¡ek	WichterlovÃ¡
ftremblay@gmail.com	FranÃ§ois	Tremblay
fzimmermann@yahoo.de	Fynn	Zimmermann
hannah.schneider@yahoo.de	Hannah	Schneider
hholy@gmail.com	Helena	HolÃ½

Q7: Lets 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 thr top 8 rock bands.

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

Result Grid | Filter Rows: | Export:

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

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.

```sql
SELECT name, milliseconds
FROM track
WHERE milliseconds>(
    SELECT AVG (milliseconds) AS avg_track_length
    FROM track)
ORDER BY milliseconds desc;
```

Result Grid | Filter Rows:

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
You Oughta Know (Alternate)	491885

Result Grid | Filter Rows:

name	milliseconds
Terra	482429
Snoopy's search-Red baron	456071
Sozinho (Hitmakers Classic Mix)	436636
Master Of Puppets	436453
Stone Crazy	433397
Snowblind	420022
Computadores Fazem Arte	404323
Jerusalem	402390
Dazed and Confused	401920
The Winner Loses	392254
Love, Hate, Love	387134

Q9: Find how much amount spent by each customer an artist? write a query to return customer name, artist name and total spent

```sql
WITH best_selling_artist AS (
    SELECT artist.artist_id AS artist_id, artist.name AS artist_name,
    sum(invoiceline.unit_price*invoiceline.quantity) AS total_sales
    FROM invoiceline
    JOIN track ON track.track_id = invoiceline.track_id
    JOIN album ON album.album_id = track.album_id
    JOIN artist ON artist.artist_id = album.artist_id
    GROUP BY 1
    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 invoiceline il ON il.invoice_id = i.invoice_id
    JOIN track t ON t.track_id = il.track_id
    JOIN album 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;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

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	StanisÅ‚aw	WÃ³jcik	AC/DC	9.9
1	LuÃs	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

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

customer_id	first_name	last_name	artist_name	amount_spent
35	Madalena	Sampaio	AC/DC	2.9699999999999998
44	Terhi	HÃ¤mÃ¤lÃ...	AC/DC	2.9699999999999998
9	Kara	Nielsen	AC/DC	1.98
34	JoÃ£o	Fernandes	AC/DC	1.98
57	Luis	Rojas	AC/DC	1.98
27	Patrick	Gray	AC/DC	1.98
20	Dan	Miller	AC/DC	1.98
30	Edward	Francis	AC/DC	1.98
5	FrantiÅ¡ek	WichterlovÃ¡	AC/DC	1.98
47	Lucas	Mancini	AC/DC	0.99
43	Isabelle	Mercier	AC/DC	0.99

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 purchase. write query that returns each country along with the top genre. for countries where the maximum number of purchases is shared return all genres.

```sql
WITH popular_genre AS
(
    SELECT COUNT(invoiceline.quantity) AS purchases, customer.country, genre.name, genre.genre_id,
    ROW_NUMBER() OVER(PARTITION BY customer.country ORDER BY COUNT(invoiceline.quantity) DESC) AS RowNo
    FROM invoiceline
    JOIN invoice ON invoice.invoice_id = invoiceline.invoice_id
    JOIN customer ON customer.customer_id = invoice.customer_id
    JOIN track ON track.track_id = invoiceline.track_id
    JOIN genre ON genre.genre_id = track.genre_id
    GROUP BY customer.country, genre.name, genre.genre_id
    ORDER BY 2 ASC, 1 DESC
)
SELECT * FROM popular_genre WHERE RowNo = 1;
```

Result Grid | Filter Rows: | Export:

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

Result Grid | Filter Rows: | Export:

purchases	country	name	genre_id	RowNo
28	Germany	Rock	1	1
4	Hungary	Rock	1	1
13	India	Rock	1	1
2	Ireland	Rock	1	1
3	Italy	Rock	1	1
6	Netherlands	Rock	1	1
2	Norway	Metal	3	1
14	Poland	Rock	1	1
23	Portugal	Rock	1	1
4	Spain	Metal	3	1
5	Sweden	Rock	1	1

Q:10 SECOND METHOD --

```sql
WITH RECURSIVE
  sales_per_country AS(
      SELECT COUNT(*) AS purchases_per_genre, customer.country, genre.name, genre.genre_id
      FROM invoiceline
      JOIN invoice ON invoice.invoice_id = invoiceline.invoice_id
    JOIN customer ON customer.customer_id = invoice.customer_id
    JOIN track ON track.track_id = invoiceline.track_id
    JOIN genre ON genre.genre_id = track.genre_id
    GROUP BY customer.country, genre.name, genre.genre_id
    ORDER BY 2
),
max_genre_per_country AS (SELECT MAX(purchases_per_genre) AS max_genre_number, country
 FROM sales_per_country
 GROUP BY 2
 ORDER BY 2)

  SELECT sales_per_country.*
  FROM sales_per_country
  JOIN max_genre_per_country on sales_per_country.country = max_genre_per_country.country
  WHERE sales_per_country.purchases_per_genre = max_genre_per_country.max_genre_number;
```

Result Grid | Filter Rows: | Export:

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

Result Grid | Filter Rows: | Export:

purchases	country	name	genre_id	RowNo
28	Germany	Rock	1	1
4	Hungary	Rock	1	1
13	India	Rock	1	1
2	Ireland	Rock	1	1
3	Italy	Rock	1	1
6	Netherlands	Rock	1	1
2	Norway	Metal	3	1
14	Poland	Rock	1	1
23	Portugal	Rock	1	1
4	Spain	Metal	3	1
5	Sweden	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 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
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

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	108.89999999999999	1
3	FranÃ§ois	Tremblay	Canada	99.99	1
57	Luis	Rojas	Chile	97.02000000000001	1
5	FrantiÅiek	WichterlovÃi	Czech Republic	144.54000000000002	1
9	Kara	Nielsen	Denmark	37.61999999999999	1
44	Terhi	HÃ¤mÃ¤lÃ¤inen	Finland	79.2	1
42	Wyatt	Girard	France	99.99	1

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

customer_id	first_name	last_name	billing_country	total_spending	RowNo
37	Fynn	Zimmermann	Germany	94.05000000000001	1
45	Ladislav	KovÃics	Hungary	78.21	1
58	Manoj	Pareek	India	111.86999999999999	1
46	Hugh	O'Reilly	Ireland	114.83999999999997	1
47	Lucas	Mancini	Italy	50.49	1
48	Johannes	Van der Berg	Netherlands	65.34	1
4	BjÃ¸rn	Hansen	Norway	72.27000000000001	1
49	StanisÅ‚aw	WÃ³jcik	Poland	76.22999999999999	1
34	JoÃ£o	Fernandes	Portugal	102.96000000000001	1
50	Enrique	MuÃ±oz	Spain	98.01	1
51	Joakim	Johansson	Sweden	75.24	1

Q11: SECOND METHOD-

```
WITH RECURSIVE
    customter_with_country AS (
        SELECT customer.customer_id,first_name,last_name,billing_country,SUM(total) AS total_spending
        FROM invoice
        JOIN customer ON customer.customer_id = invoice.customer_id
        GROUP BY 1,2,3,4
     ORDER BY 2,3 DESC),

     country_max_spending AS (
       SELECT billing_country,MAX(total_spending) AS max_spending
       FROM customter_with_country
       GROUP BY billing_country)

SELECT cc.billing_country, cc.total_spending, cc.first_name, cc.last_name, cc.customer_id
FROM customter_with_country cc
JOIN country_max_spending ms ON cc.billing_country = ms.billing_country
WHERE cc.total_spending = ms.max_spending
ORDER BY 1;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

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	108.89999999999999	1
3	FranÃ§ois	Tremblay	Canada	99.99	1
57	Luis	Rojas	Chile	97.02000000000001	1
5	FrantiÅ¡ek	WichterlovÃ¡	Czech Republic	144.54000000000002	1
9	Kara	Nielsen	Denmark	37.61999999999999	1
44	Terhi	HÃ¤mÃ¤lÃ¤inen	Finland	79.2	1
42	Wyatt	Girard	France	99.99	1

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

customer_id	first_name	last_name	billing_country	total_spending	RowNo
37	Fynn	Zimmermann	Germany	94.05000000000001	1
45	Ladislav	KovÃ¡cs	Hungary	78.21	1
58	Manoj	Pareek	India	111.86999999999999	1
46	Hugh	O'Reilly	Ireland	114.83999999999997	1
47	Lucas	Mancini	Italy	50.49	1
48	Johannes	Van der Berg	Netherlands	65.34	1
4	BjÃ¸rn	Hansen	Norway	72.27000000000001	1
49	StanisÅ‚aw	WÃ³jcik	Poland	76.22999999999999	1
34	JoÃ£o	Fernandes	Portugal	102.96000000000001	1
50	Enrique	MuÃ±oz	Spain	98.01	1
51	Joakim	Johansson	Sweden	75.24	1


THANK
YOU