Music Store Data Analysis

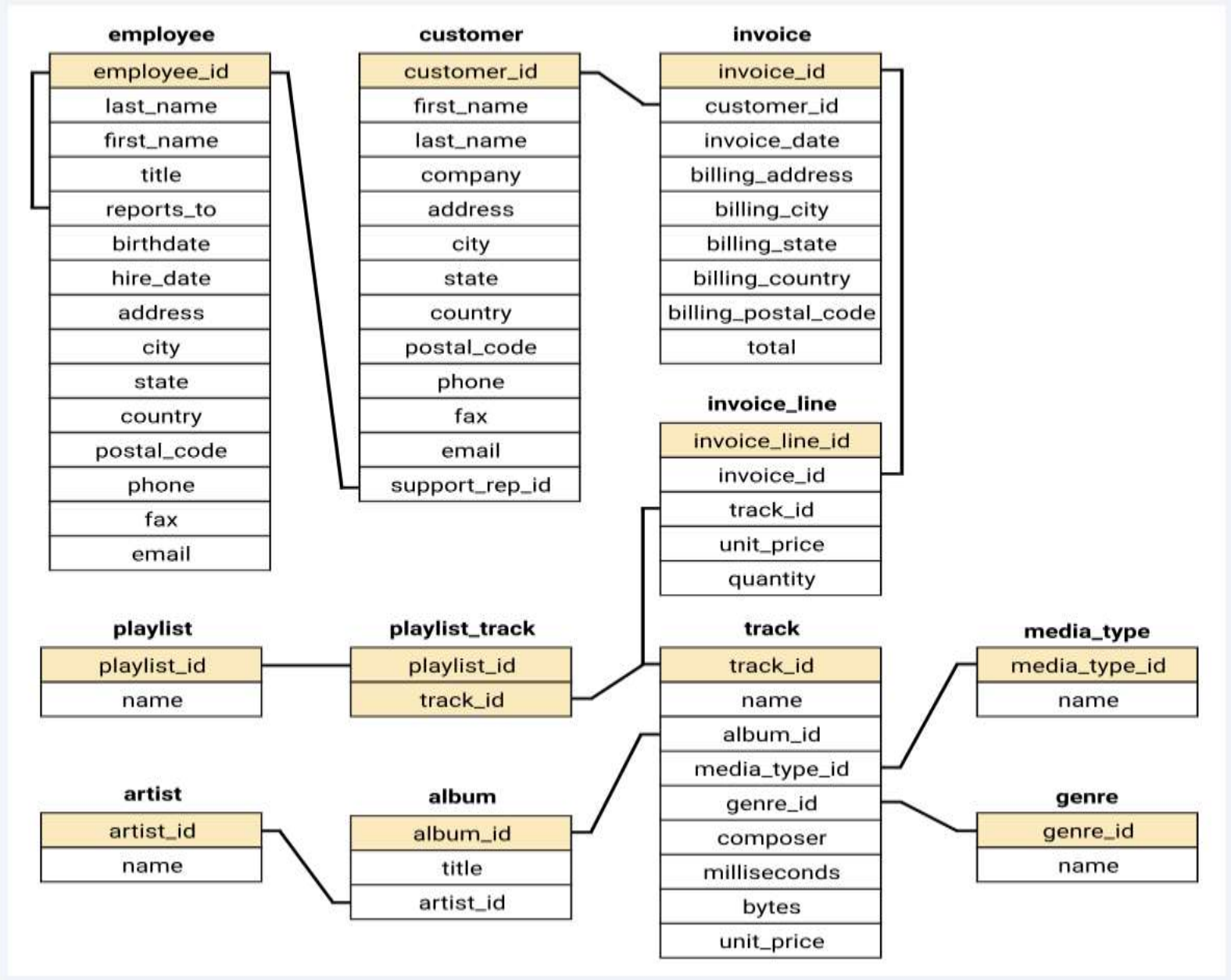
SQL PROJECT

- NIKITA SUHAS JANGAM

OBJECTIVE

- Analyze sales data from a music store to identify popular genres, customer purchasing patterns, and revenue-driving artists.
- This analysis will provide insights to optimize inventory and improve targeted marketing strategies.

Database Schema



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	reports_to	levels	birthdate	hire_date
▶	1	Adams	Andrew	General Manager	9	L6	18-02-1962 00:00	14-08-2016 00:00

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
	10	Denmark
	10	Australia
	10	Hungary
	10	Sweden

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 total,billing_city from invoice
group by billing_city
order by total desc ;
```

total	billing_city
273.24000000000007	Prague
169.29	Mountain View
166.32	London
158.4	Berlin
151.47	Paris
129.69	SÃ£o Paulo
114.83999999999997	Dublin
111.86999999999999	Delhi
108.89999999999998	SÃ£o JosÃ© dos Campos
106.91999999999999	BrasÃlia
102.96000000000001	Lisbon
99.99	Bordeaux
99.99	MontrÃ©al
98.01	Madrid
98.01	Redmond
97.02000000000001	Santiago
94.05000000000001	Frankfurt
92.07000000000001	Orlando

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 total
FROM customer
JOIN invoice ON customer.customer_id = invoice.customer_id
GROUP BY invoice.customer_id, customer.first_name, customer.last_name
ORDER BY total DESC
LIMIT 1;
```

customer_id	first_name	last_name	total
5	FrantiÅ¡ek	WichterlovÃ¡	144.54000000000002

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 join invoice
on customer.customer_id=invoice.customer_id
join invoice_line
on invoice_line.invoice_id=invoice.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;
```

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
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¡

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

```sql
SELECT artist.name, COUNT(artist.artist_id) AS songs
FROM track
JOIN album ON track.album_id = album.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.name
ORDER BY songs DESC
LIMIT 10;
```

	name	songs
▶	AC/DC	18
	Aerosmith	15
	Audioslave	14
	Led Zeppelin	14
	Alanis Morissette	13
	Alice In Chains	12
	Frank Zappa & Captain Beefheart	9
	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
	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
	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

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 album ON album.album_id = track.album_id
    JOIN artist ON artist.artist_id = album.artist_id
    GROUP BY artist.artist_id, artist.name
    ORDER BY total_sales 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 album alb ON alb.album_id = t.album_id
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;
```

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	LuÃ-s	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
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

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 maximum number of purchases is shared return all Genres.

```sql
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
    ORDER BY customer.country ASC, purchases DESC
)
SELECT
    country,
    genre_name,
    purchases
FROM popular_genre
WHERE RowNo = 1;
```

country	genre_name	purchases
Argentina	Rock	1
Australia	Rock	18
Austria	Rock	6
Belgium	Rock	5
Brazil	Rock	26
Canada	Rock	57
Chile	Rock	7
Czech Republic	Rock	14
Denmark	Rock	6
Finland	Rock	6
France	Rock	26
Germany	Rock	28
Hungary	Rock	4
India	Rock	13
Ireland	Rock	2
Italy	Rock	3
Netherlands	Rock	6
Norway	Metal	2

Thank you