

SQL PROJECT

MUSIC STORE ANALYSIS

- P R I Y A N S H U Y A D A V

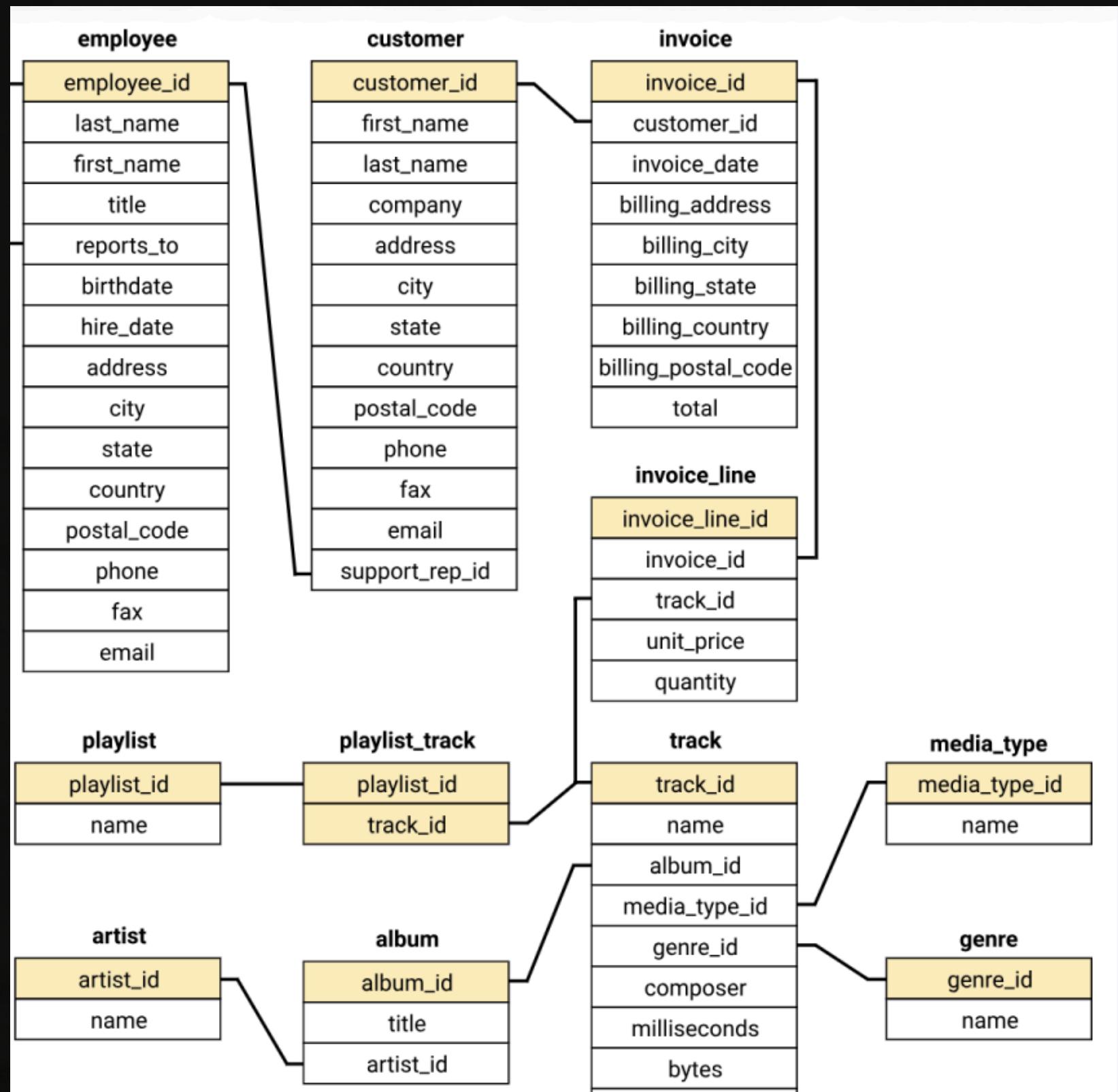


OBJECTIVE

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 understand its business growth by answering simple questions

MUSIC PLAYLIST DATABASE SCHEMA



QUESTION -1

Who is the senior most employee based on job title ?

```
select title,first_name, last_name, levels  
from employee  
order by levels desc  
limit 1
```

	title character varying (50) 	first_name character 	last_name character 	levels character varying (10) 		
1	Senior General Manager	Mohan	...	Madan	...	L7

QUESTION - 2

Which countries have the most Invoices ?

```
select billing_country as country, count(*) as Invoices  
from invoice  
group by country  
order by Invoices desc
```

	country character varying (30) 	invoices bigint 
1	USA	131
2	Canada	76
3	Brazil	61
4	France	50
5	Germany	41
6	Czech Republic	30
7	Portugal	29
8	United Kingdom	28
9	India	21
10	Chile	13
11	U.S.A.	10

QUESTION - 3

What are top 3 values of total invoices ?

```
select total from invoice  
order by total desc  
limit 3
```

	total	double precision
1	23.759999999999998	
2		19.8
3		19.8

QUESTION - 4

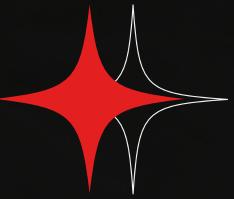
Which city has the best customers? We would like to throw a promotional Music Festival in the city we made the most money. Write a query that return one city that has the highest sum of invoice totals.

Return both the city name & sum of all invoices total

```
select sum(total) as invoice_total,billing_city  
from invoice  
group by billing_city  
order by invoice_total desc  
limit 1
```

	invoice_total double precision	billing_city character varying (30)
1	273.24000000000007	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 return 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 customer.customer_id  
order by total desc  
limit 1
```

	customer_id [PK] integer	first_name character	last_name character	total double precision
1	5	R	Madhav	144.54000000000002



QUESTION - 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.invoice_id = invoice_line.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 character varying (50)	first_name character	last_name character
1	aaronmitchell@yahoo.ca	Aaron	Mitchell
2	alero@uol.com.br	Alexandre	Rocha
3	astrid.gruber@apple.at	Astrid	Gruber

QUESTION - 7

Let's invite the artists who have written the most rock music in our dataset. Write a query that return the artist name and total track count of the top 10 rock bands

```
select artist.name as artist_name, count(track.track_id) as total_track  
from artist  
join album on album.artist_id = artist.artist_id  
join track on track.album_id = album.album_id  
join genre on genre.genre_id = track.genre_id  
where genre.name like 'Rock'  
group by artist_name  
order by total_track desc  
limit 10
```

	artist_name character varying (120)	total_track bigint
1	Led Zeppelin	114
2	U2	112
3	Deep Purple	92
4	Iron Maiden	81
5	Pearl Jam	54
6	Van Halen	52
7	Queen	45
8	The Rolling Stones	41
9	Creedence Clearwater Revival	40
10	Kiss	35



QUESTION - 8

Return all the track names that have a song lenght 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 character varying (150)	milliseconds integer
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	Take the Celestra	2927677
10	Fire In Space	2926593
11	The Long Patrol	2925008





THANK YOU