SQL PROJECT ON

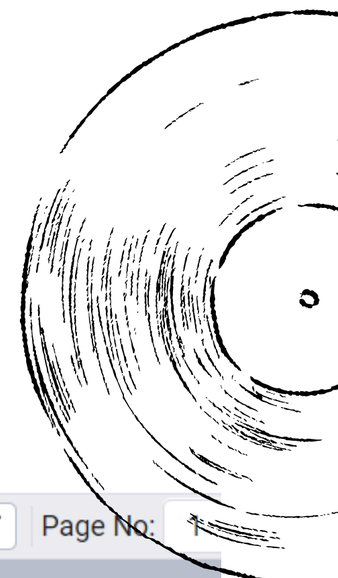
MUSIC STORE

HELLO !

My name is **Hritik Choudhary**,
and this is my project on a **Music Store database**.
I used **SQL** to solve business-related queries,
analyze customer behavior, and explore sales and
genre trends.
This project helped me strengthen my data analysis
skills using real-world scenarios.

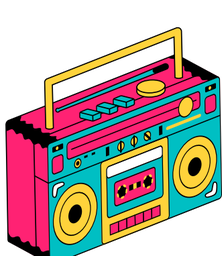
Who is the senior most employee based on job title?


```sql
select * from employee
order by levels desc
limit 1 ;
```



Showing rows: 1 to 1 | Page No: 1

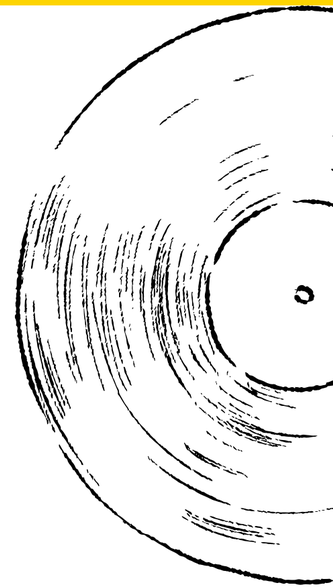
employee_id [PK] character varying (50)	last_name character (50)	first_name character (50)	title character varying (50)	reports_to character varying (30)	levels character varying (10)	birthdate timestamp without time zone	hire_date timestamp without time zone
9	Madan ...	Mohan ...	Senior General Manager	[null]	L7	1961-01-26 00:00:00	2016-01-14 00:00:00



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

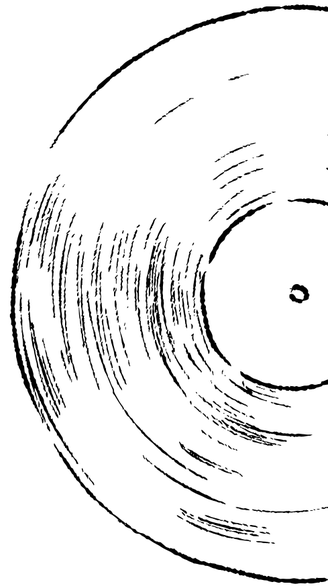
```sql
with popular_genre as
(
    select count(invoice_line.quantity) as purchases , customer.country,genre.name,genre.genre_id,
    row_number() over(partition by customer.country order by count(invoice_line.quantity)desc) as Rowno
    from invoice_line
    join invoice on invoice.invoice_id = invoice_line.invoice_id
    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 bigint	country character varying (50)	name character varying (120)	genre_id character varying (50)	rowno bigint
1	17	Argentina	Alternative & 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 Republic	Rock	1	1
9	24	Denmark	Rock	1	1

Which countries have the most Invoices?


```sql
select count(*) as number_of_invoice , billing_country
from invoice
group by billing_country
order by number_of_invoice desc ;
```

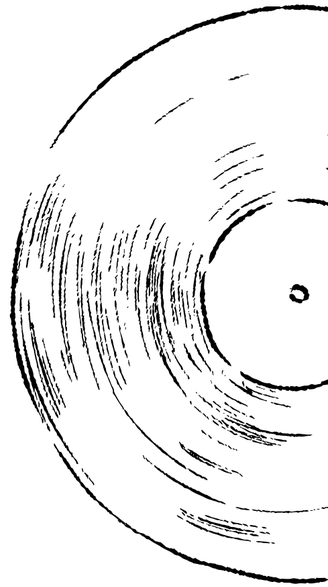


Data Output | Messages | Notifications

	number_of_invoice bigint	billing_country character varying (30)
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

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


```sql
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 asc ;
```

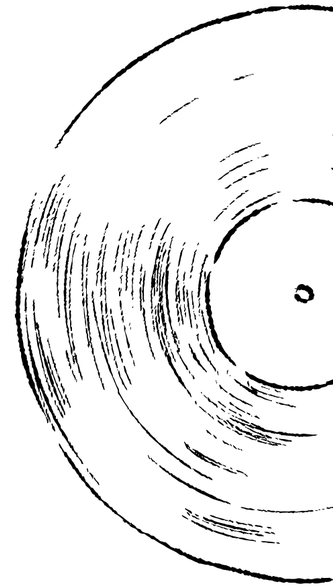


Data Output | Messages | Notifications

	email character varying (50)	first_name character (50)	last_name character (50)
1	aaronmitchell@yahoo.ca	Aaron	Mitchell
2	alero@uol.com.br	Alexandre	Rocha
3	astrid.gruber@apple.at	Astrid	Gruber
4	bjorn.hansen@yahoo.no	Bjørn	Hansen
5	camille.bernard@yahoo.fr	Camille	Bernard
6	daan_peeters@apple.be	Daan	Peeters
7	diego.gutierrez@yahoo.ar	Diego	Gutiérrez
8	dmiller@comcast.com	Dan	Miller

What are top 3 values of total invoice?


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



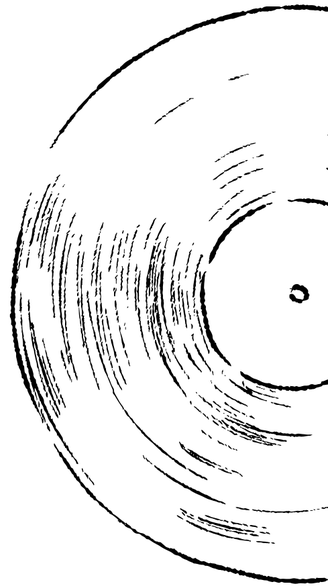
Data Output | Messages

	total double precision
1	23.759999999999998
2	19.8
3	19.8

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

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

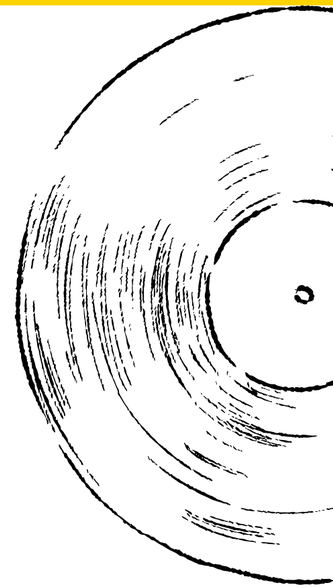


Data Output | Messages | Notifications

	invoice_total double precision	billing_city character varying (30)
1	273.24000000000007	Prague
2	169.29	Mountain View
3	166.32	London
4	158.4	Berlin
5	151.47	Paris
6	129.69	São Paulo
7	114.83999999999997	Dublin
8	111.86999999999999	Delhi

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


```sql
select customer.customer_id,customer.first_name,customer.last_name,sum(invoice.total) as total_spent
from customer
join invoice
on customer.customer_id = invoice.customer_id
group by customer.customer_id
order by total_spent desc
limit 1;
```



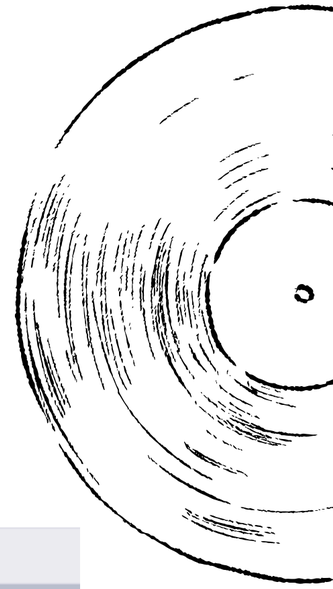
Data Output | Messages | Notifications

	customer_id [PK] integer	first_name character (50)	last_name character (50)	total_spent double precision
1	5	R ...	Madhav ...	144.54000000000002

Find how much amount spent by each customer on artists? Write a query to return customer name, artist name and total spent

QUERY

```sql
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 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 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 1,2,3,4
order by 5 desc;
```



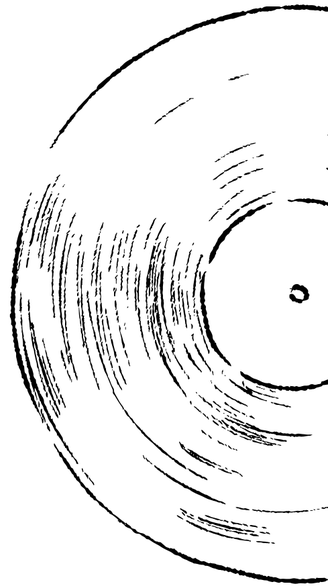
Data Output | Messages | Notifications

	customer_id integer	first_name character (50)	last_name character (50)	artist_name character varying (120)	amount_spent double precision
1	46	Hugh ...	O'Reilly ...	Queen	27.719999999999985
2	38	Niklas ...	Schröder ...	Queen	18.81
3	3	François ...	Tremblay ...	Queen	17.82
4	34	João ...	Fernandes ...	Queen	16.830000000000002
5	53	Phil ...	Hughes ...	Queen	11.88
6	41	Marc ...	Dubois ...	Queen	11.88
7	47	Lucas ...	Mancini ...	Queen	10.89
8	33	Ellie ...	Sullivan ...	Queen	10.89

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

```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 ;
```



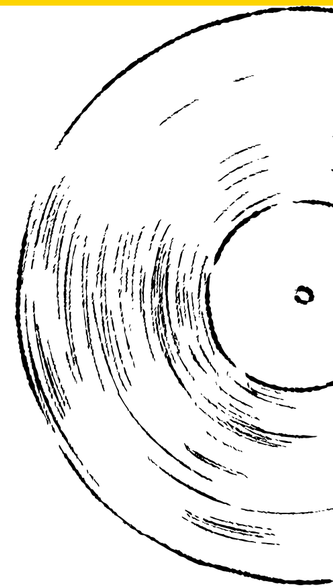
Data Output | Messages | Notifications

	artist_id [PK] character varying (50)	name character varying (120)	number_of_songs bigint
1	22	Led Zeppelin	114
2	150	U2	112
3	58	Deep Purple	92
4	90	Iron Maiden	81
5	118	Pearl Jam	54
6	152	Van Halen	52
7	51	Queen	45
8	142	The Rolling Stones	41

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

```sql
select name , milliseconds
from track
where milliseconds > (
    select avg(milliseconds) as length_of_song
    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