SQL PROJECT- MUSIC STORE DATA ANALYSIS

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

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

	first_name character (50) 🔒	last_name character (50) 🔒
1	Mohan ...	Madan ...

2. Which countries have the most Invoices?

```
select count(*) as c, billing_country from invoice
group by billing_country
order by c desc limit 1
```

	c bigint 🔒	billing_country character varying (30) 🔒
1	131	USA

3. What are top 3 values of total invoice?

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

	total double precision 🔒
1	23.759999999999998
2	19.8
3	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 billing_city, sum(total) as totals from invoice
group by billing_city
order by totals desc limit 1
```

	billing_city character varying (30) 🔒	totals double precision 🔒
1	Prague	273.24000000000007

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 sum(invoice.total) as totals, customer.customer_id, customer.first_name, custome
join invoice on customer.customer_id = invoice.customer_id
group by customer.customer_id
order by totals desc limit 1
```

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

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.

```sql
select distinct genre.name, customer.email, customer.first_name, customer.last_name fr
join invoice on invoice.customer_id = customer.customer_id
join invoice_line on invoice_line.invoice_id = invoice.invoice_id
join track on track.track_id = invoice_line.track_id
join genre on genre.genre_id = track.genre_id
where genre.name = 'Rock'
order by customer.email
```

	name character varying (120)	email character varying (50)	first_name character (50)	last_name character (50)
1	Rock	aaronmitchell@yahoo.ca	Aaron ...	Mitchell ...
2	Rock	alero@uol.com.br	Alexandre ...	Rocha ...
3	Rock	astrid.gruber@apple.at	Astrid	Gruber ...
4	Rock	bjorn.hansen@yahoo.no	Bjørn	Hansen ...
5	Rock	camille.bernard@yahoo.fr	Camille ...	Bernard ...
6	Rock	daan_peeters@apple.be	Daan	Peeters ...

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.artist_id, artist.name, count(artist.artist_id) as no_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 = 'Rock'
group by artist.artist_id
order by no_of_songs desc
limit 10
```

	artist_id [PK] character varying (50)	name character varying (120)	no_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

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.

```sql
select track.name, track.milliseconds from track
where track.milliseconds > (select avg(milliseconds) as avg_length from track)
order by track.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

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

```
1   with best_selling_artist as (
2       select artist.artist_id as artist_id, artist.name as artist_name,
3       sum(invoice_line.unit_price*invoice_line.quantity) as total_sales
4       from invoice_line
5       join track on track.track_id = invoice_line.track_id
6       join album on album.album_id = track.album_id
7       join artist on artist.artist_id = album.artist_id
8       group by 1
9       order by 3 desc
10      limit 1
11  )
12  select customer.customer_id, customer.first_name, customer.last_name, best_selling_artist.artist_name,
13  sum(invoice_line.unit_price*invoice_line.quantity) as amount_spend from invoice
14  join customer on customer.customer_id = invoice.customer_id
15  join invoice_line on invoice_line.invoice_id = invoice.invoice_id
16  join track on track.track_id = invoice_line.track_id
17  join album on album.album_id = track.album_id
18  join best_selling_artist on best_selling_artist.artist_id = album.artist_id
19  group by 1,2,3,4
20  order by 5 desc;
```

	customer_id integer	first_name character (50)	last_name character (50)	artist_name character varying (120)	amount_spend 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

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 the maximum number of purchases is shared return all Genres.

```
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

11. 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.

```sql
with customer_with_country as (
    select customer.customer_id, customer.first_name, customer.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
```

	customer_id integer	first_name character (50)	last_name character (50)	billing_country character varying (30)	total_spending double precision	rowno bigint
1	56	Diego	...	Gutiérrez	...	Argentina	39.6	1
2	55	Mark	...	Taylor	Australia	81.18	1
3	7	Astrid	Gruber	...	Austria	69.3	1
4	8	Daan	...	Peeters	...	Belgium	60.38999999999999	1
5	1	Luís	Gonçalves	...	Brazil	108.89999999999998	1
6	3	François	...	Tremblay	...	Canada	99.99	1
7	57	Luis	Rojas	Chile	97.02000000000001	1
8	5	R	Madhav	...	Czech Republic	144.54000000000002	1
9	9	Kara	Nielsen	...	Denmark	37.61999999999999	1