---------------------------------QUESTION SET 1 (EASY)---

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

```
SELECT * FROM employee
ORDER BY LEVELS DESC
LIMIT 1;
```

2) Which countries have the most invoices?

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

3) What are the top 3 values of total invoices?

```
SELECT total FROM invoice
ORDER BY total DESC
LIMIT 3;
```

4) Which city has the best customers? We would like to throw a promotional Musical Festival in the city we made the most money.

Write a query that returns one city that has the highest sum of invoice total. Return both the city name and sum of of all

invoice totals.

```
SELECT SUM(total) as Total_invoice, billing_city
FROM invoice
GROUP BY billing_city
ORDER BY Total_invoice DESC
LIMIT 1;
```

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 customer.customer_id
ORDER BY TOTAL DESC
LIMIT 1;
```

---QUESTION SET 2 (Moderate)--

6) Write a 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;
```

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 the total track count of the top 10 rock bands.

```
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;
```

8) Return all the track names that has a song length longer than the average song length. Return the name

and the 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;
```

--------------------------------------QUESTION SET 3 (Difficult)--

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 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 i.customer_id = C.customer_id

JOIN Invoice_line il ON i.Invoice_id = il.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;
```

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
```

11) 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.

```
WITH Customter_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 Customter_with_country WHERE RowNo <= 1
```

---THE END--