```
SELECT * FROM album
```

Q1: Who is the Senior most employee based on the job title?

```
SELECT * FROM employee
ORDER BY levels desc
LIMIT 1
```

Q2: Which countries have the most Invoices?

```
SELECT COUNT(*) AS C, billing_country
        FROM invoice
group by billing_country
ORDER BY C DESC
```

Q3: What are top 3 of the total invoice?

```
SELECT total FROM Invoice
        order by total desc
        LIMIT 3
```

Q4: 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 invoice_total, billing_city
        FROM invoice
GROUP BY billing_city
ORDER BY invoice_total desc
```

```
LIMIT 1
```

Q5: 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
```

Q6: 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;
```

Q7: 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.

```
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;
```

Q8: Return all the track names with a song length longer than the average.

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;
```

Q9: 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 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;
```

Q10: We want to find out each country's most popular music genres.

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
```

Q11: 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
```