```
-- 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 top 3 values of total invoice?

select * from invoice order by total desc limit 3;


-- 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 SUM (TOTAL) AS INVOICE_TOTAL , BILLING_CITY
FROM INVOICE
GROUP BY BILLING_CITY
ORDER BY INVOICE_TOTAL
DESC ;


-- 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
invoice_total
from customer
join invoice on customer.customer_id = invoice.customer_id
GROUP BY customer.customer_id
ORDER BY invoice_total desc limit 3 ;
```

```
-- 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 * from customer
-- select * from genre

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
-- 2. 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 * from artist

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 = 'Rock'
GROUP BY artist.artist_id
ORDER BY number_of_songs DESC
LIMIT 10
```

```
-- 3. 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



SELECT  name, track.milliseconds FROM track

Where track.milliseconds >

(SELECT AVG(track.milliseconds) as average_length  FROM track)

ORDER BY track.milliseconds desc



-- 1. 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;


-- 2. 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


-- method 2
WITH RECURSIVE
        sales_per_country AS(
                SELECT COUNT(*) AS purchases_per_genre, customer.country, genre.name,
genre.genre_id
                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
        ),
        max_genre_per_country AS (SELECT MAX(purchases_per_genre) AS max_genre_number,
country
                FROM sales_per_country
                GROUP BY 2
                ORDER BY 2)

SELECT sales_per_country.*
FROM sales_per_country
JOIN max_genre_per_country ON sales_per_country.country = max_genre_per_country.country
WHERE sales_per_country.purchases_per_genre = max_genre_per_country.max_genre_number;


-- 3. 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



WITH RECURSIVE
	customter_with_country AS (
		SELECT customer.customer_id,first_name,last_name,billing_country,SUM(total) AS
total_spending
		FROM invoice
		JOIN customer ON customer.customer_id = invoice.customer_id
		GROUP BY 1,2,3,4
		ORDER BY 2,3 DESC),


	country_max_spending AS(
		SELECT billing_country,MAX(total_spending) AS max_spending
		FROM customter_with_country
		GROUP BY billing_country)


SELECT cc.billing_country, cc.total_spending, cc.first_name, cc.last_name, cc.customer_id
FROM customter_with_country cc
JOIN country_max_spending ms
ON cc.billing_country = ms.billing_country
WHERE cc.total_spending = ms.max_spending
ORDER BY 1;
```