**/\* Who is the senior most employee based on job title? \*/**

select \* from employee order by levels desc limit 1;

**/\* Which countries have the most Invoices? \*/**

select count(\*) as total\_no\_of\_invoices, billing\_country as country from invoice group by billing\_country order by country desc;

**/\* What are top 3 values of total invoice? \*/**

select total from invoice order by total desc limit 3;

/\* 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 invoice\_total\_amount from invoice group by billing\_city order by invoice\_total\_amount desc ;

**/\* 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, customer.first\_name, customer.last\_name, total order by total desc limit 1;

**/\* 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 AS Email, first\_name AS FirstName, last\_name AS LastName, genre.name AS Name FROM customer

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 LIKE 'Rock'

ORDER BY email

LIMIT 10;

**/\* 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,name

ORDER BY number\_of\_songs DESC

LIMIT 10;

**/\* 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, milliseconds from track

where milliseconds >( select avg(milliseconds) as avg\_track\_lenght from track)

order by milliseconds desc;

**/\*. 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 artist.artist\_id, artist.name

ORDER BY total\_sales 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 c.customer\_id, c.first\_name, c.last\_name, bsa.artist\_name

ORDER BY amount\_spent DESC;

**/\* 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