



## Question Set 1 - Easy

**Q1. Who is the senior most employee based on job title?**

**Sol.**

```
SELECT title, last_name, first_name
FROM employee
ORDER BY levels DESC
LIMIT 1
```

**Q2. Which countries have the most Invoices?**

**Sol. (used group by to merge country as one in output)**

```
select count(billing_country) as invoice_count,
billing_country from invoice
group by billing_country order by invoice_count desc
```

**Q3. What are the top 3 values of total invoice?**

**Sol.**

```
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.**

**Sol.**

```
select billing_city, sum(total) as invoice_total 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.**

**Sol.**

```
select customer.customer_id, customer.first_name, customer.last_name,
sum(total) as most_money from customer
join invoice on customer.customer_id=invoice.customer_id
group by customer.customer_id
order by most_money desc limit 1
```

## Question Set 2 - Medium

**Q1. 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.**

**Sol. Without genre column**

```
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.in
voice_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;
```

**With genre Column**

```
SELECT DISTINCT
    c.email, c.first_name,
    c.last_name,
    g.name AS genre
FROM customer c
JOIN
    invoice i ON c.customer_id =
i.customer_id
JOIN
    invoice_line il ON
i.invoice_id = il.invoice_id
JOIN
```

```
    track t ON il.track_id =
t.track_id
JOIN
    genre g ON t.genre_id =
g.genre_id
WHERE g.name LIKE 'Rock'
ORDER BY c.email;
```

**(Since above query uses multiple join it can be optimize with common table expression (CTE) is used to simplify the join logic related to tracks and genres. SQL 'with' Clause)**

```
WITH TrackGenre AS (
    SELECT
        il.track_id,
        g.name AS genre
    FROM
        invoice_line il
    JOIN
        track t ON il.track_id =
t.track_id
    JOIN
        genre g ON t.genre_id =
g.genre_id
    WHERE g.name LIKE 'Rock'
)
SELECT DISTINCT
    c.email,
    c.first_name,
    c.last_name,
    tg.genre
FROM
    customer c
JOIN
    invoice i ON c.customer_id =
i.customer_id
JOIN
```

```

JOIN
    invoice_line il ON
i.invoice_id = il.invoice_id
JOIN
    TrackGenre tg ON il.track_id =
tg.track_id
ORDER BY c.email;

```

**Q2. 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.**

**Sol.**

```

SELECT
    artist.name,
    COUNT(track.track_id) AS total_tracks
FROM
    artist
JOIN album ON artist.artist_id = album.artist_id
JOIN track ON album.album_id = track.album_id
JOIN genre ON track.genre_id = genre.genre_id
WHERE
    genre.name = 'Rock'
GROUP BY
    artist.name
ORDER BY
    total_tracks DESC
LIMIT 10;

```

**Q3. 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.**

**Sol.**

```

select name, milliseconds from track
where milliseconds > (select avg(milliseconds) from track)
order by milliseconds desc

```

### Question Set 3 - Advance

**Q1.** Find how much amount is spent by each customer on artists? Write a query to return customer name, artist name and total spent

**Sol.**

Steps to Solve:

First, find which artist has earned the most according to the InvoiceLines.

Now use this artist to find which customer spent the most on this artist.

For this query, you will need to use the Invoice, InvoiceLine, Track, Customer, Album, and Artist tables. Note, this one is tricky because the Total spent in the Invoice table might not be on a single product, so you need to use the InvoiceLine table to find out how many of each product was purchased, and then multiply this by the price for each artist.

```
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;
```

**Q2.** 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.

**Sol.**

Steps to Solve:

There are two parts in question:

- first most popular music genre.
- second need data at the country level.

Method 1: Using CTE

```
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: : Using Recursive

```
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;

```

**Q3.** 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.

**Sol.**

Steps to Solve:

Similar to the above question. There are two parts in question-

-first find the most spent on music for each country.

-second filter the data for respective customers.

Method 1: using CTE

```

WITH Customer_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 Customer_with_country WHERE RowNo <= 1

```

Method 2: Using Recursive

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;
```