

Music Store Analysis –SQL Queries

Senior most employee based on job title:

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

Countries that have the most invoices :

```
SELECT billing_country, COUNT(*) AS no_of_invoices
FROM invoice
GROUP BY billing_country
ORDER BY no_of_invoices DESC;
```

Top 3 values of total invoices :

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

City that has the best customers and the highest sum of invoice totals :

```
SELECT billing_city, SUM(total) AS invoiceTOTAL
FROM invoice
GROUP BY billing_city
ORDER BY invoiceTOTAL DESC;
```

Customer who spent the most money :

```
SELECT c.customer_id, c.first_name, c.last_name, SUM(i.total) AS total
FROM customer c
JOIN invoice i ON i.customer_id = c.customer_id
GROUP BY c.customer_id
ORDER BY total DESC
LIMIT 1;
```

Return the email, first & last name and genre of all rock music listeners (alphabetically):

```
SELECT DISTINCT c.email, c.first_name, c.last_name
FROM customer c
JOIN invoice i ON i.customer_id = c.customer_id
JOIN invoice_line il ON il.invoice_id = i.invoice_id
WHERE track_id IN(
    SELECT track_id FROM track t
    JOIN genre g ON g.genre_id = t.genre_id
    WHERE g.name LIKE 'Rock'
)
ORDER BY email;
```

Return the artist name and total track count of the top 10 rock bands :

```
SELECT ar.artist_id, ar.name, COUNT(ar.artist_id) AS no_of_songs
FROM track t
JOIN album a ON a.album_id = t.album_id
JOIN artist ar ON ar.artist_id = a.artist_id
JOIN genre g ON g.genre_id = t.genre_id
WHERE g.name LIKE 'Rock'
GROUP BY ar.artist_id
ORDER BY no_of_songs DESC
LIMIT 10;
```

Return all track names that are longer than avg length. Return name & milliseconds for each track :

```
SELECT name, milliseconds
FROM track
WHERE milliseconds > (
    SELECT AVG(milliseconds) AS avg_track_length
    FROM track)
ORDER BY milliseconds DESC;
```

Amount spent by each customer on top artists, return customer & artist name, and total spent :

```
WITH best_selling_artist AS (
    SELECT ar.artist_id AS artist_id, ar.name AS artist_name,
    SUM(il.unit_price * il.quantity) AS total_sales
    FROM invoice_line il
    JOIN track t ON t.track_id = il.track_id
    JOIN album a ON a.album_id = t.album_id
    JOIN artist ar ON ar.artist_id = a.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 a ON a.album_id = t.album_id
JOIN best_selling_artist bsa ON bsa.artist_id = a.artist_id
GROUP BY 1,2,3,4
ORDER BY 5 DESC;
```

Most popular music genre for each country i.e genre with the highest amount of purchase :

```
WITH popular_genre AS
(
    SELECT COUNT(il.quantity) AS purchases, c.country, g.name, g.genre_id,
    ROW_NUMBER() OVER(PARTITION BY c.country ORDER BY COUNT(il.quantity) DESC)
AS RowNo
    FROM invoice_line il
    JOIN invoice i ON il.invoice_id = i.invoice_id
    JOIN customer c ON c.customer_id = i.customer_id
    JOIN track t ON t.track_id = il.track_id
    JOIN genre g ON g.genre_id = t.genre_id
    GROUP BY 2,3,4
    ORDER BY 2 ASC, 1 DESC
)
SELECT * FROM popular_genre WHERE RowNo <= 1
```

Return the country along with top customer and how much they spent:

```
WITH RECURSIVE
    customer_with_country AS (
        SELECT c.customer_id, first_name, last_name, billing_country,
        SUM(total) AS total_spending
        FROM invoice i
        JOIN customer c ON c.customer_id = i.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 customer_with_country  
    GROUP BY billing_country)  
  
SELECT cc.billing_country, cc.total_spending, cc.first_name, cc.last_name,  
cc.customer_id  
FROM customer_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;
```