SQL PROJET

Music Store Data Analysis

```sql
-- Who is the senior most employee based on the Job title?
select *
from employee
order by levels desc
LIMIT 1
```

```sql
-- Which countries have the most invoices?
select billing_country, count(*) as most_invoices
from invoice
group by billing_country
order by most_invoices desc
limit 1
```

```sql
-- What are top three values of invoices?
select total
from invoice
order by total desc
limit 3
```

```sql
-- Which city has the best customers? We would like to throw a promotional
-- MusicFestival 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
-- and sum of all invoice totals.

select billing_city, sum(total) as total_invoice
from invoice
group by billing_city
order by total_invoice desc
limit 1
```

```sql
-- 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_id, sum(total) as best_customer
from invoice
group by customer_id
order by best_customer desc
limit 1
```

```sql
    -- Write a query to return the email, first name, last name and 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;
```

```sql
-- Lets 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 no_of_songs
from track
join album2 on album2.album_id= track.album_id
join artist on artist.artist_id= album2.artist_id
join genre on genre.genre_id= track.genre_id
where genre.name like 'Rock'
group by artist.artist_id, artist.name
order by no_of_songs desc
limit 10;
```

```sql
-- 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_length
    from track)
order by milliseconds desc
```

```sql
-- We want to find out the most popular music genre for each country.
-- We determine the most popular genre as the genre with 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
```

```sql
-- 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 spentis shared, provide all customers who spent this amount

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
```