

SQL QUERY

\$(MUSIC_DATABASE ANALYSIS)

create database music_database;

use music_database;

select * from album2;

select * from artist;

select * from customer;

select * from employee;

select * from genre;

select * from invoice;

select * from invoice_line;

select * from media_type;

select * from playlist;

select * from playlist_track;

select * from track;

-- Q1 who is the senior most employee based on job title?

select * from employee

order by levels desc

limit 1;

-- Q2 which countries have the most Invoices ?

select count(*) as c,billing_country

from invoice

group by billing_country

order by c desc;

-- Q3 what are top 3 values of total invoice

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.

select sum(total) as invoice_total, billing_city

from invoice

group by billing_city

order by invoice_total desc;

-- 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.

```
select * from customer;
```

```
select * from invoice;
```

```
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
```

```
order by total desc
```

```
limit 1;
```

-- Q6 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 invoice;
```

```
select * from invoice_line;
```

```
select * from track;
```

```
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;
```

-- Q7 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 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 number_of_songs DESC
LIMIT 10;
```

- Q8 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 * from track;
```

```
select name,milliseconds
```

```
from track
```

```
where milliseconds > (
```

```
    select AVG(milliseconds) as avg_track_length
```

```
    from track)
```

```
ORDER BY milliseconds DESC;
```

REPORT

- **THE senior most employee is Andrew adams who's job title is General Manager**
- **From Canada.**

- **USA HAS MOST INVOICE WHICH IS 131.**

- **TOP 3 VALUES OF TOTAL INVOICE =23.75,19.8 & 19.8**

- **PRAGUE CITY HAS THE BEST CUSTOMERS WHICH IS GENERATED TOTAL INVOICE=273.240**

- **THE BEST CUSTOMER WHO IS SPENT MOST MONEY IS FRANTIAIEK WICHTERLOVAi HIS CUSTOMER_ID IS 5 & TOTAL IS 144.54.**

- **AMONG ALL ROCK MUSIC LISTENERS IS TOP POSITION WHO'S EMAIL IS aaronmitchell@yahoo.ca, name is Aaron Mitchell**

- **AN ARTIST HAS THE MOST ROCK MUSIC NAME IS Alice in Chains,total no of song have 15 & artist_id is 5.**
- **Longest song name is How Many More Times which is (711836) milliseconds.**