- **Name :-** Rohan Bharat Kalkumbe
- **E-mail :-** rohankalkumbe31@gmail.com

Music Store DataBase Analysis

---------- **Easy Query** ----------

Q1) Who is the employee with the highest Levels?

Q2) What are the top 3 countries have the most Invoices?

Q3) Who is the senior most employee based on job title?

Q4) What are the top 3 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.

Q5) Which city has the best customers? Write a query that returns one city with the highest sum of invoice totals.

Q6) Which city has the lowest invoice total?

Q7) Who are the top 3 employees hired in the early stages of the company, and what impact have they had on its growth and success?

---------- **Moderate Query** ----------

Q1) 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.

Q2) Identify the top 2 artists based on the total number of albums.

Q3) List Top 3 tracks with a length greater than the average length of all tracks.

Q4) 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.

---------- **Advanced Query** ----------

Q1) Amount Spent by Each Customer on Each Artist.

Q2) For countries where the top amount spent is shared, provide all customers who spent this amount.

Q3) 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.

Q4) 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.

---------- **Easy Query** ----------

-- Q1) Who is the employee with the highest Levels?

```
select * from employee;

SELECT first_name, Levels
FROM employee
ORDER BY Levels DESC
LIMIT 1;
```

-- Q2. What are the top 3 countries have the most Invoices?

```
select * from invoice;

SELECT billing_country, COUNT(invoice_id) AS total_invoices
FROM invoice
GROUP BY billing_country
ORDER BY total_invoices DESC
LIMIT 3;
```

-- Q3. Who is the senior most employee based on job title?

```
select * from Employee;

SELECT title, first_name, country
FROM employee
ORDER BY levels DESC
```

```
LIMIT 1;
```

-- Q4) What are the top 3 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 C.customer_id, first_name, country,
ROUND(CAST(SUM(total) AS NUMERIC),2) AS total_spending
FROM customer C
JOIN invoice I ON C.customer_id = I.customer_id
GROUP BY C.customer_id
ORDER BY total_spending DESC
LIMIT 3;
```

-- Q5) Which city has the best customers? Write a query that returns one city with the highest sum of invoice totals.

```
select * from invoice;


SELECT billing_city, billing_country,
ROUND(CAST(SUM(total) AS NUMERIC), 2) AS total_sales
FROM invoice
GROUP BY billing_city, billing_country
ORDER BY total_sales DESC
LIMIT 1;
```

--Q6) Which city has the lowest invoice total?

```
Select * From Invoice;


SELECT billing_city, billing_country,
ROUND(CAST(SUM(total) AS Numeric), 2) AS total_invoices
FROM invoice
GROUP BY billing_city, billing_country
ORDER BY total_invoices ASC
LIMIT 1;
```

-- Q7) Who are the top 3 employees hired in the early stages of the company, and what impact have they had on its growth and success?

```
select * from employee;


SELECT employee_id, first_name, hire_date
FROM employee
ORDER BY hire_date ASC
LIMIT 3;
```

---------- **Moderate** ----------

--Q1) 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 genre;



SELECT DISTINCT email,first_name, last_name
FROM customer C
JOIN invoice I ON C.customer_id = I.customer_id
JOIN invoice_line IL ON I.invoice_id = IL.invoice_id
WHERE track_id IN(
	SELECT track_id FROM track T
	JOIN genre G ON T.genre_id = G.genre_id
	WHERE G.name LIKE 'Rock'
)
ORDER BY email;
```

-- Q2) Identify the top 2 artists based on the total number of albums.

```
SELECT * From Album;
SELECT * from artist;

SELECT name, COUNT(A.artist_id) AS total_albums
FROM album AL
JOIN artist A ON AL.album_id = A.artist_id
GROUP BY name
ORDER BY total_albums DESC
LIMIT 2;
```

-- Q3) List Top 3 tracks with a length greater than the average length of all tracks.

```
SELECT * From track;

SELECT name, milliseconds
FROM track
WHERE milliseconds > (SELECT AVG(milliseconds) FROM track)
order  by milliseconds DESC
LIMIT 3;
```

-- Q4) 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;
```

---------- **Advanced** ----------

-- Q1) Amount Spent by Each Customer on Each Artist.

```
SELECT * From customer;
SELECT * From album;
SELECT * From invoice;
SELECT * From invoice_line;
SELECT * FROM track;
SELECT * FROM artist;
SELECT * FROM playlist;


SELECT customer.first_name AS customer_first_name,
customer.last_name AS customer_last_name,
artist.name AS artist_name,
ROUND(CAST(SUM(invoice_line.unit_price * invoice_line.quantity) as numeric), 2)
AS total_spent
FROM customer JOIN invoice ON customer.customer_id = invoice.customer_id
JOIN invoice_line ON invoice.invoice_id = invoice_line.invoice_id
JOIN track ON invoice_line.track_id = track.track_id
JOIN album ON track.album_id = album.album_id
JOIN artist ON album.artist_id = artist.artist_id
GROUP BY customer.customer_id, artist.artist_id
ORDER BY total_spent DESC;
```

-- Q2) For countries where the top amount spent is shared, provide all customers who spent this amount.

```
WITH RECURSIVE
    customter_with_country AS (
        SELECT customer.customer_id,first_name,last_name,billing_country,
        ROUND(CAST(SUM(total) AS Numeric), 2) 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;
```

-- Q3. 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 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;
```

--Q4: 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.

```
WITH Customter_with_country AS (
		SELECT customer.customer_id,first_name,last_name,billing_country,
		ROUND(CAST(SUM(total)AS Numeric), 2) 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 Customter_with_country WHERE RowNo <= 1
```