Mastering SQL with the Chinook Database: A Journey Through Music Data Analytics

Chiamaka Okakpu

Photo by Sincerely Media on Unsplash

Introduction

Data analysis is like solving a puzzle, and SQL is one of the most powerful tools in a data analyst's toolkit. Recently, I embarked on an exciting journey analyzing the Chinook music database using PostgreSQL, and let me tell you it was both challenging and rewarding!

The Chinook database represents a digital media store, complete with tables for artists, albums, tracks, employees, invoices, and customers. It's a perfect playground for practicing SQL skills and understanding relational database concepts. This project taught me valuable lessons about database management, query optimization, and the importance of understanding your data structure.

The Learning Curve: Challenges and Breakthroughs

The Table Name Mystery

My journey started with what seemed like a simple task querying the customer table. However, I kept encountering this frustrating error message: *"relation 'customer' doesn't exist."* Truly, I couldn't figure where

i was wrong.

After multiple attempts and feeling like I was going in circles, I turned to Stack Overflow and Reddit for answers. *I turn to Reddit whenever i feel stuck because trust me* ***whatever you're going through, others have gone through it too!*** The solution was simpler than I thought , I needed to reference the tables exactly as they were created, using the schema prefix: `public."tablename"`.

Get Chiamaka Okakpu's stories in your inbox

Join Medium for free to get updates from this writer.

This experience taught me an important lesson: understanding your database structure and naming conventions is crucial before diving into analysis.

Import Struggles and Persistence

The dataset import process was another hurdle. I had to delete and restart the process twice before getting it right . Sometimes, the setup phase takes longer than the actual analysis, but persistence pays off!

The Analysis: 10 Key Questions About Music Sales Data

Working with the Chinook database, I tackled 10 comprehensive analytical questions that covered various aspects of SQL querying:

1. Geographic Sales Analysis

Question: Which countries have the most invoices?

```
Select country, count (invoice_id) as Totalinvoice
From public."Customer" As C
Join public."Invoice" As I
On C.customer_id = I.customer_id
Group by C.country
Order by Totalinvoice DESC;
```

Result: USA leads with 91 invoices, showing the strong North American market presence.

Country with the most invoices

2. City-Level Customer Performance

Question: What city has the best customer?

```
Select city,  sum (total) as Totalspent
From public."Customer" as C
Join public."Invoice" as I
On C.customer_id = I.customer_id
```

```
Group by C.city
Order by Totalspent Desc;
```

Result: Prague emerges as the top city with a customer spending amount of 90.24.

3. Top Customer Identification

Question: Who is the best customer?

```
Select C.firstname, C.lastname, sum (total) as Totalspent
From public."Customer" as C
Join public."Invoice" as I
On C.customer_id = I.customer_id
Group by C.firstname, C.lastname
Order by Totalspent DESC;
```

Result: Helena Holy takes the crown with a total spending of 49.62.

4. Genre-Specific Customer Segmentation

Question: Return email, first name, last name, and genre of all Rock Music listeners.

```
Select Distinct C.firstname, C.lastname,C.email, g.name As Genr
From public."Customer" as C
Join public."Invoice" as i
On c.customer_id = i.customer_id
Join public."Invoiceline" As il
On i.invoice_id = il.invoice_id
Join public."Track" As t
On il.track_id = t.track_id
Join public."Genre" As g
On t.genre_id = g.genre_id
Where g.name = 'Rock'
Order by C.email ASC;
```

Result: 59 rows and 4 columns of rock music enthusiasts, ordered alphabetically by email.

5. Artist Productivity in Rock Genre

Question: Which artist has written the most ROCK songs?

```
Select a.name As ArtistName, Count (t.name) as TotalRock
From public."Artist" As a
Join public."Album" As al
On a.artist_id = al.artist_id
Join public."Track" As t
On al.album_id = t.album_id
Join public."Genre" As g
On t.genre_id = g.genre_id
Where g.name = 'Rock'
```

```
Group by a.artist_id
Order by TotalRock DESC;
```

Result: Led Zeppelin dominates the rock scene with 114 rock songs.

6. Revenue Leader Analysis

Question: Which artist has earned the most according to InvoiceLines?

```
Select a.name As ArtistName, Sum (il.Quantity * il.unit_price)
From public."Artist" As a
Join public."Album" As al
On a.artist_id = al.artist_id
Join public."Track" As t
On al.album_id = t.album_id
Join public."Invoiceline" As il
On t.track_id = il.track_id
Group by a.artist_id
Order by TotalRevenue DESC;
```

Result: Iron Maiden tops the earnings chart with 138.6 in total revenue.

7. Fan Loyalty Investigation

Question: Find the customer who spent the most on the top-earning artist.

```
With ArtistEarning As (
Select a.name As ArtistName, Round(Sum(il.Quantity * il.unit_pr
From public."Artist" As a
Join public."Album" As al
On a.artist_id = al.artist_id
Join public."Track" As t
On al.album_id = t.album_id
Join public."Invoiceline" As il
On t.track_id = il.track_id
Join public."Invoice" As i
On il.invoice_id = i.invoice_id
Group by a.name), TopArtist As ( Select ArtistName From ArtistE
Order by TotalRevenue DESC
Limit 1),
CustomerSpending As (
Select c.Customer_id, c.firstname, c.lastname, Sum(il.unit_pric
From public."Customer" As c
Join public."Invoice" As i
On c.customer_id = i.customer_id
Join public."Invoiceline" As il
```

```
On i.invoice_id = il.invoice_id
Join public."Track" As t
On il.track_id = t.track_id
Join public."Album" As al
On t.album_id = al.album_id
Join public."Artist" As a
On al.artist_id = a.artist_id
Where a.name = (
Select ArtistName From TopArtist)
Group by
c.customer_id,
c.firstname,
c.lastname
)
Select customer_id, firstname, lastname, TotalSpent
From CustomerSpending
Order by TotalSpent Desc
Limit 1;
```

Result: Mark Taylor is Iron Maiden's biggest fan in terms of spending.

8. Country-Specific Genre Preferences

Question: What's the most popular music genre for each country?

```
Select i.billing_country As Country, g.name As Genre, Count(*)
From public."Invoice" As i
Join public."Invoiceline" As il
On i.invoice_id = il.invoice_id
```

```
Join public."Track" As t
On il.track_id = t.track_id
Join public."Genre" As g
On t.genre_id = g.genre_id
Group by Country, Genre
Order by TotalPurchased Desc;
```

Result: USA shows a strong preference for Rock music with 157 purchases.

9. Track Length Analysis

Question: Return track names with song length longer than average.

```
Select t.name As TrackName, milliseconds
From public."Track" As t
Where milliseconds >  ( Select Avg(milliseconds) As avg_length
From public."Track")
Order by milliseconds Desc;
```

Result: 494 tracks exceed the average length, with

“Occupation/Precipice” leading at 5,286,953 milliseconds.

10. Top Spender by Country

Question: Determine the customer who spent the most on music for each country.

```
Select C.firstname, C.lastname, C.country, sum (total) as Total
From public."Customer" as C
Join public."Invoice" as I
On C.customer_id = I.customer_id
Group by C.firstname, C.lastname, C.country
Order by Totalspent DESC;
```

Result: Helena Holy from Czech Republic leads with 49.62 in total spending.

Conclusion

Working with the Chinook database has been an incredible learning experience that reinforced several important principles:

SQL is about logic and persistence. Every error message is an opportunity to learn something new about your data and your tools. The frustration of encountering “relation doesn’t exist” errors transformed into

the satisfaction of understanding PostgreSQL's schema requirements.

Community learning accelerates growth. When stuck, don't hesitate to leverage community resources. Chances are, someone else has faced the same challenge and shared their solution.

Data tells stories. Through these 10 analytical questions, i uncovered fascinating insights about music preferences, customer behavior, and sales patterns across different regions and genres.

Practice makes perfect. Each query taught me something new about JOIN operations, aggregate functions, and subqueries.

The journey from struggling with basic table references to successfully executing complex analytical queries demonstrates that persistence and continuous learning are key to mastering SQL. Whether you're analyzing music sales data or any other dataset, remember that every expert was once a beginner who refused to give up.