Music Data Insights

Analyzing Business, Customer, and Genre Trends

Nandakishore Sasidharan

Project Overview

- **Objective:** Analyze a music database to uncover business insights.
- **Tech Used:** PostgreSQL, SQL (Joins, CTEs, Window Functions).

Database Schema

- **Tables Used:** employee, customer, invoice, invoiceline, track, album, artist, genre, musictype, playlist, playlisttrack.

Key Business Questions

- Who are the top customers?
- What genres are most popular in each country?
- Which city generates the most revenue?
- Who are the top rock artists by track count?

Most Senior Employee

- **Query used:** select lastname, firstname, title, levels from employee order by levels desc;
- **Result:** Mohan Madan – Senior General Manager – Level 7

Countries with Most Invoices

- **Query used:** select billingcountry, count(*) as invoices from invoice group by billingcountry order by invoices desc;
- **Top Country:** USA – 131 invoices

Top 3 Invoice Totals

- **Query used:** select invoiceid, customerid, total from invoice order by total desc limit 3;

1. **Invoice ID:** 183 **Total:** ₹23.76
2. **Invoice ID:** 92 **Total:** ₹19.80
3. **Invoice ID:** 31 **Total:** ₹19.80

Best City for Promotions

- City with highest total invoice value.
- **Query used:** select billingcity, sum(total) as highest_revenue from invoice group by billingcity order by highest_revenue desc limit 1;
- **Recommendation:** Promote in Prague.

Top Spending Customer

- **Query used:** select c.customerid, c.firstname, c.lastname, c.phone, sum(i.total) as total_spent from customer c join invoice i on c.customerid = i.customerid group by c.customerid, c.firstname, c.lastname, c.phone order by total_spent desc limit 1;
- **Customer:** Frantisek Wichterlov
- **Total Spend:** ₹144.54

Rock Genre Listeners

- **Query used:** select distinct c.firstname, c.lastname, c.email from customer c join invoice i on c.customerid = i.customerid join invoiceline il on i.invoiceid = il.invoiceid join track t on t.trackid = il.trackid join genre g on t.genreid = g.genreid where g.name = 'Rock';
- Filtered customers who purchased Rock tracks.

Top 10 Rock Artists

- **Query used:** select a.name as artist, count(t.trackid) as tracks from artist a join album al on a.artistid = al.artistid join track t on t.albumid = al.albumid join genre g on t.genreid = g.genreid where g.name = 'Rock' group by a.name order by tracks desc limit 10;
- Based on track count for genre Rock.
- Top Artist: Led Zeppelin – 114 tracks.

Tracks Above Average Length

- **Query used:** select name from track where milliseconds > (select avg(milliseconds) from track);
- **Average track length:** 3,93,599 Milliseconds.

Customer Spend Per Artist

- **Query used:** with artistrev as(select c.customerid, c.firstname, c.lastname, a.artistid, a.name as artist, sum(il.unitprice * il.quantity) as total from customer c join invoice i on c.customerid = i.customerid join invoiceline il on i.invoiceid = il.invoiceid join track t on il.trackid = t.trackid join album al on t.albumid = al.albumid join artist a on al.artistid = a.artistid group by c.customerid, c.firstname, c.lastname, a.name, a.artistid) select * from artistrev order by customerid, total desc;
- CTE used to calculate per artist earnings.
- Joined with customer and invoice.

Popular Genres by Country

- **Query used:** WITH genre_rank AS (SELECT c.country, g.name AS genre, COUNT(*) AS purchase_count, RANK() OVER (PARTITION BY c.country ORDER BY COUNT(*) DESC) AS rank FROM customer c JOIN invoice i ON c.customerid = i.customerid JOIN invoiceline il ON i.invoiceid = il.invoiceid JOIN track t ON il.trackid = t.trackid JOIN genre g ON t.genreid = g.genreid GROUP BY c.country, g.name) SELECT country, genre, purchase_count FROM genre_rank WHERE rank = 1 ORDER BY country;
- Grouped and ranked by genre count.
- **Argentina:** Alternative and Punk | **Austrailia:** Rock | **Austria:** Rock | etc.

Top Customer by Country

- **Query used:** WITH customer_spending AS (SELECT c.customerid, c.firstname, c.lastname, c.country, SUM(i.total) AS total_spent, RANK() OVER (PARTITION BY c.country ORDER BY SUM(i.total) DESC) AS rank FROM customer c JOIN invoice i ON c.customerid = i.customerid GROUP BY c.customerid, c.firstname, c.lastname, c.country) SELECT customerid, firstname, lastname, country, total_spent FROM customer_spending WHERE rank = 1 ORDER BY country;
- For each country, show highest spender.
- **Argentina:** Diego Gutierrez | **Australia:** Mark Taylor | etc.

Summary and Recommendations

- Promote in high revenue cities.
- Focus on Rock and Latin genres.
- Reward top spending customers.

SQL Techniques Used

- Joins and Aggregations
- CTEs and Window Functions
- Filtering, Sorting, and Grouping

Thank You!