

MUSIC STORE

Prepared by : Kumar Balaram Boste

A N A L Y S I S

**Index**

1. **Introduction**………………………………………………………… **2**
2. **Entity Relationship Diagram** (ERD)……………………….......... **3**
3. **Analysis 1**

* 1.1 List all Customers …………………………………..
* 1.2 Get Tracks with Price Greater than 1.20 …………
* 1.3 Find all Albums by Ed Sheeran ……………………..
* 1.4 Count Total Customers from India ……………..
* 1.5 Display all Employees who are Sales Rep………….

1. **Analysis 2**

* 2.1 Find the Total number of Tracks Per Genre …..
* 2.2 Show Invoice Details along with Customer Names.
* 2.3 Find top 5 Most Expensive Tracks …….
* 2.4 Show all Playlists with Track Count …….
* 2.5 Find Customers Who Spent more than 15 …..
* 2.6 Find the Best-selling Track ……………………..
* 2.7 Find Top 3 Artists by Sales Revenue ……………….
* 2.8 Find the Most Popular Genre ………….

1. **Analysis 3**

* 3.1 Get the Top Customer by Spending ……….
* 3.2 Find Employee with Maximum Customers Handled
* 3.3 Find top 3 Customers by Invoices and Spending …
* 3.4 Average Track Length per Genre ………
* 3.5 Rank Artists by Total Revenue …………

**INTRODUCATION**

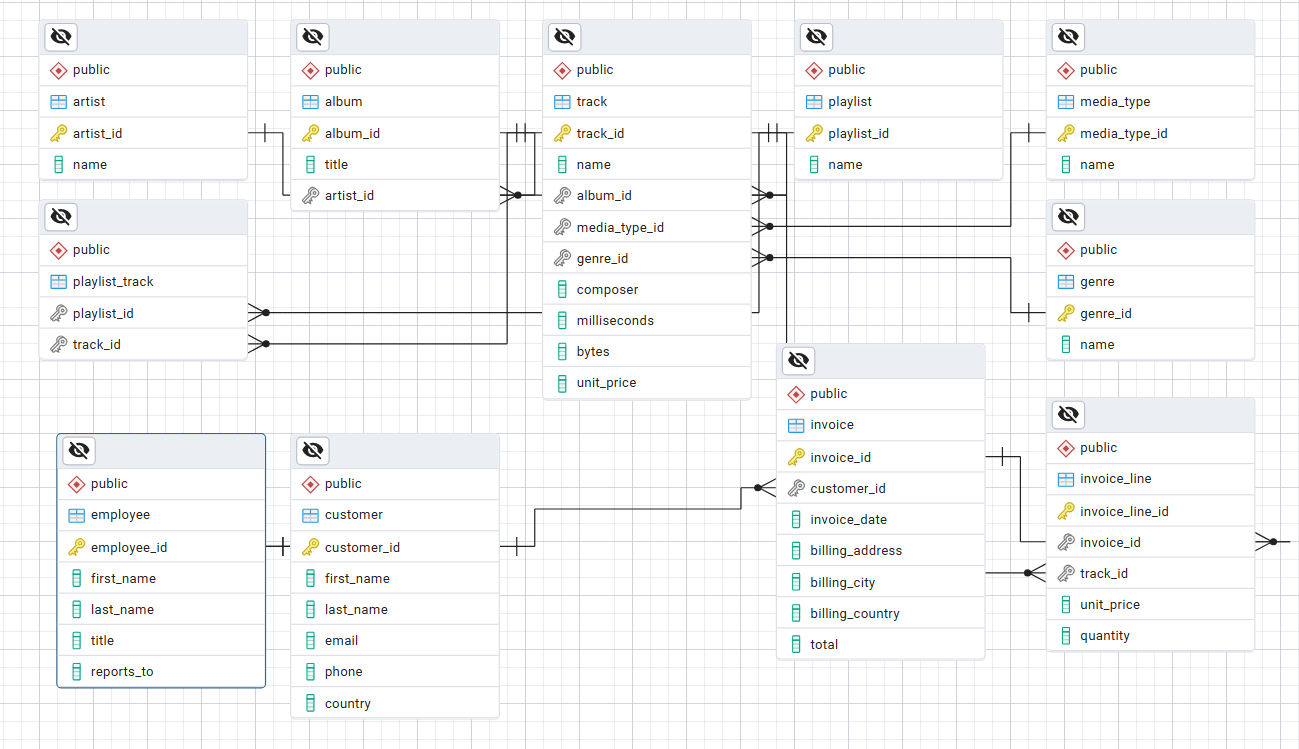
**Music Store Analysis** project simulates a real-world database for an online/offline music store. It contains information about artists, albums, tracks, playlists, customers, employees, invoices, and sales. This project helps in analyzing customer preferences, sales performance, popular tracks, and employee management.

**Overview of the Schema**

The schema consists of 11 interconnected tables that provide insights into the operations of Music Store.

The Project involves multiple tables :

* Artist
* Album
* Track
* Genre
* Media Type
* Playlist
* Playlist Track
* Customer
* Employee
* Invoice
* Invoice Line

**BACKGROUND OF PROJECT**

The Music Industry has evolved significantly with the rise of digital platforms, streaming services, and global customer reach. Music stores—whether physical or online—must efficiently manage large volumes of data related to **artists, albums, tracks, playlists, customers, employees, and sales transactions**.

This project, **Music Store Analysis**, is designed to simulate a real-world database system for such a store. By building a relational database with **11 interconnected tables**, the project provides a structured way to store and analyze information. It allows tracking of customer purchases, employee contributions, artist performance, and genre preferences.

The analysis through SQL queries generates insights into **sales trends, customer spending behavior, popular tracks, and revenue contributions by artists and genres**. This background provides the foundation for better decision-making, targeted marketing strategies, and enhanced customer satisfaction.

**Entity Relationship Diagram (ERD)**

**ANALYSIS - 1**

* 1. **List all Customers**

**Problem Statement :**

The store needs to maintain a complete list of all customers for sales tracking, communication, and marketing purposes. Having a centralized customer database helps in managing customer relationships and segmenting users for different campaigns.

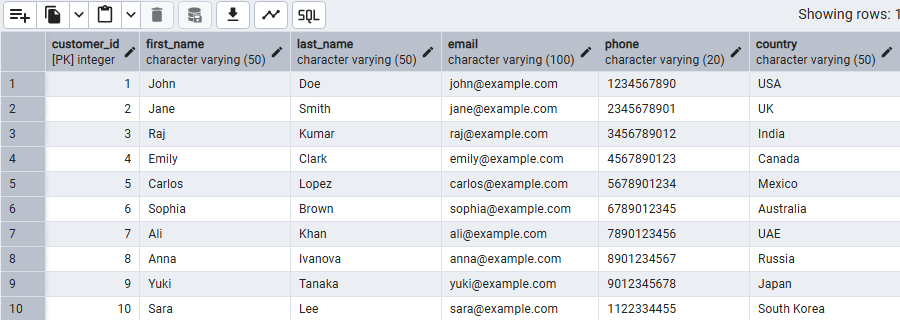
**Approach :**

A simple SELECT \* FROM customer retrieves all available customer records, including their names, addresses, contact information, and associated details.

**SQL Query :**

**1.png**

**Output :**



**Analysis :**

This result serves as the master customer list. It helps identify the geographical spread of customers, track how many are active, and provides the foundation for advanced segmentation such as by country or spending level.

**Recommendations:**

* Keep this table updated regularly with new and modified customer details.
* Use customer information to design personalized offers and communication strategies.
* Integrate this with a CRM (Customer Relationship Management) system for better targeting.

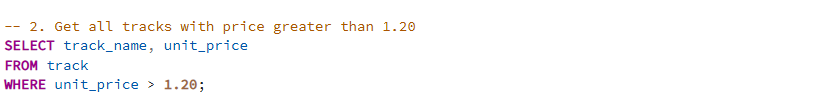
**1.2 Get Tracks with Price Greater than 1.20**

**Problem Statement :**

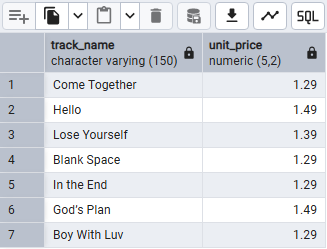
The business needs to identify premium-priced tracks to understand which songs contribute more to revenue and to evaluate their role in pricing strategy.

**Approach :**

Querying the track table and filtering rows where unit price > 1.20 helps isolate tracks that are above the standard baseline price.

**SQL Query :**

**Output :**



**Analysis :**

Premium-priced tracks often represent popular, exclusive, or high-demand songs. Customers willing to pay for these tracks demonstrate stronger brand loyalty or preference.

**Recommendations :**

* Promote these premium tracks more aggressively through featured playlists.
* Bundle them with standard-priced tracks for up-selling.
* Analyze whether higher prices negatively affect sales volume.
  1. **Find all Albums by Ed Sheeran**

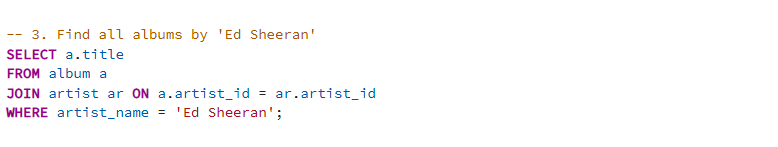
**Problem Statement :**

The store wants to find all albums released by a specific artist, in this case Ed Sheeran. This is useful for artist-focused campaigns and promotions.

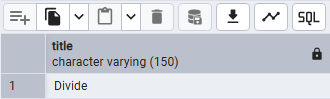
**Approach :**

Join the album and artist tables on their artist\_id and filter where artist\_name = 'Ed Sheeran'.

**SQL Query :**



**Output :**



**Analysis :**

This identifies all albums in the catalog belonging to Ed Sheeran. Such queries help in understanding catalog coverage for specific artists.

**Recommendations :**

* Promote artist collections (e.g., "All Ed Sheeran Albums").
* Identify gaps in catalog completeness for top-selling artists.
* Use this data for artist-specific recommendations to customers.
  1. **Count Total Customers from India**

**Problem Statement :**

To measure regional customer distribution, the store wants to find how many customers are from India. This supports regional marketing and expansion strategies.

**Approach :**

Use COUNT(\*) on the customer table filtered by country = 'India'.

**SQL Query :**



**Output :**



**Analysis :**

The result gives a numerical measure of the Indian customer base. This metric is useful for understanding international presence.

**Recommendations :**

* Launch localized promotions in India.
* Introduce region-specific playlists (e.g., Bollywood, Indian classical).
* Monitor customer growth in this region over time.
  1. **Display all Employees Who are Sales Reps**

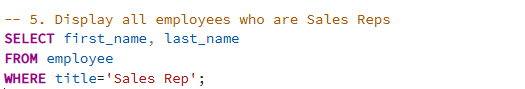
**Problem Statement :**

Identify employees working in direct customer-facing sales roles. This is important for performance management and workforce planning.

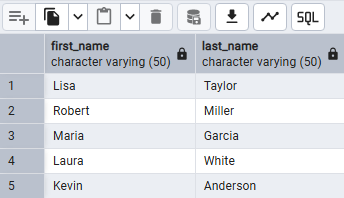
**Approach :**

Query the employee table where title = 'Sales Rep'.

**SQL Query :**



**Output :**



**Analysis:**

This returns the names of employees who directly handle customer accounts. These individuals are critical for driving revenue.

**Recommendations:**

* Provide training programs to Sales Reps to improve customer experience.
* Track their sales performance via linked invoices.
* Incentivize top-performing Sales Reps with bonuses.

**ANALYSIS - 2**

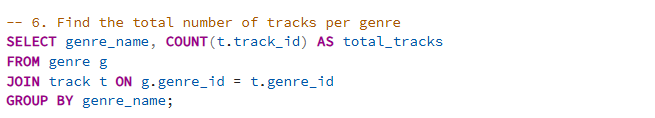
**2.1 Find the Total Number of Tracks Per Genre**

**Problem Statement:**

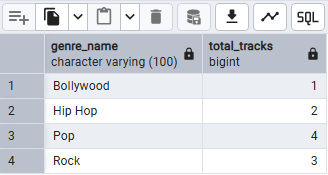
The business wants to analyze content diversity and determine how tracks are distributed across genres.

**Approach:**

Join genre and track tables, group by genre, and count the number of tracks in each.

**SQL Query :**

**Output :**



**Analysis:**

This query shows whether the catalog is well-balanced or skewed toward certain genres.

**Recommendations:**

* Expand in genres with fewer tracks to attract a wider audience.
* Focus marketing efforts on the most populated genres.
* Balance catalog diversity to cater to niche markets.

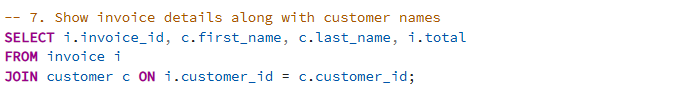
**2.2 Show Invoice Details along with Customer Names**

**Problem Statement:**

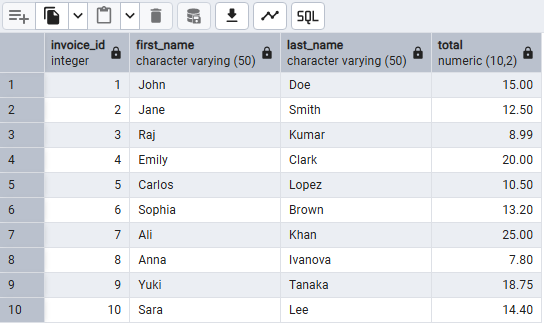
Management wants a detailed view of invoices along with the customers who generated them for better financial tracking.

**Approach:**

Join invoice with customer to display invoice ID, customer name, and invoice total.

**SQL Query :**

**Output :**



**Analysis:**

The output helps connect each transaction to a customer, making it easier to analyze spending behavior and purchase history.

**Recommendations:**

* Use this for auditing and reconciliation.
* Implement customer segmentation based on spending patterns.
* Provide customers with invoice history for transparency.

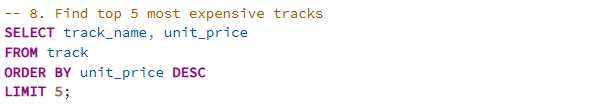
**2.3 Find Top 5 Most Expensive Tracks**

**Problem Statement:**

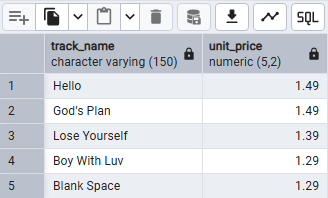
Identify tracks with the highest unit prices to assess catalog pricing.

**Approach:**

Order track records by unit\_price descending and select the top 5.

**SQL Query :**

**Output :**



**Analysis:**

These are the most premium-priced tracks, potentially influencing overall revenue.

**Recommendations:**

* Highlight these tracks in "Premium Collections."
* Study whether higher prices correlate with reduced sales volume.
* Consider limited discounts to increase demand.

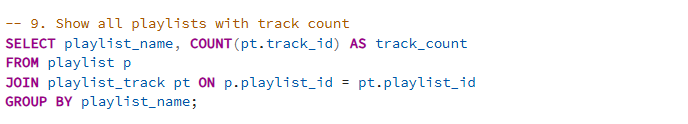
**2.4 Show all Playlists with Track Count**

**Problem Statement:**

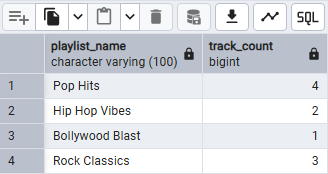
Determine the size of playlists to understand catalog curation.

**Approach:**

Join playlist with playlist\_track, group by playlist, count tracks per playlist.

**SQL Query :**

**Output :**



**Analysis:**

Reveals which playlists are content-rich and which are sparse. This is important for customer satisfaction.

**Recommendations:**

* Standardize playlist sizes to ensure consistency.
* Promote playlists with balanced track counts (not too short, not too long).
* Curate niche playlists for specialized audiences.

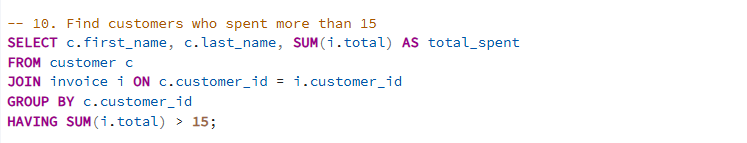
**2.5 Find Customers Who Spent more than 15**

**Problem Statement:**

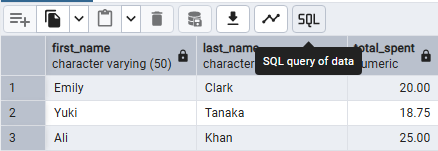
Identify high-value customers contributing more than $15 in purchases.

**Approach:**

Join customer and invoice, group by customer, filter with SUM(total) > 15.

**SQL Query :**

**Output :**



**Analysis:**

This highlights big spenders who are valuable to retain.

**Recommendations:**

* Enroll these customers in loyalty programs.
* Offer personalized recommendations and discounts.
* Send them early access to new releases.

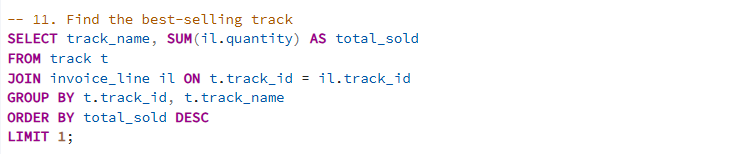
**2.6 Find the Best-Selling Track**

**Problem Statement:**

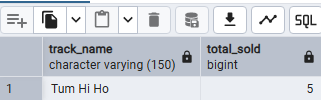
Find the track that has sold the most copies.

**Approach:**

Join track with invoice\_line, group by track, sum quantities, order descending.

**SQL Query :**

**Output :**



**Analysis:**

The result shows the most popular track by sales volume.

**Recommendations:**

* Promote this track on the homepage or featured section.
* Create playlists around similar tracks.
* Use it as a model for predicting future hits

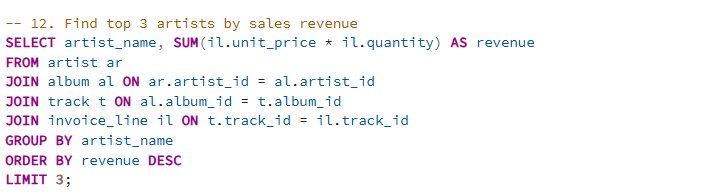
**2.7 Find top 3 Artists by Sales Revenue**

**Problem Statement:**

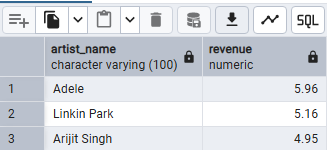
Identify the artists generating the highest revenue for the store.

**Approach:**

Join artist, album, track, invoice\_line; group by artist, sum revenue, and select top 3.

**SQL Query :**

**Output :**

****

**Analysis:**

This shows which artists are driving the most income.

**Recommendations:**

* Prioritize top artists in promotions.
* Consider exclusive partnerships with these artists.
* Allocate more marketing funds to their albums.

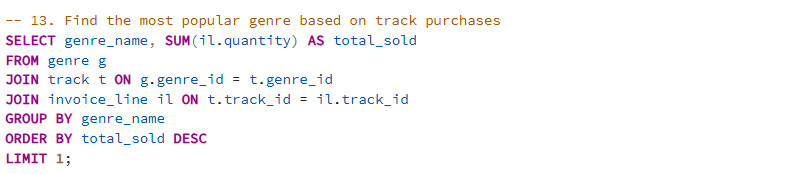
**2.8 Find the Most Popular Genre**

**Problem Statement:**

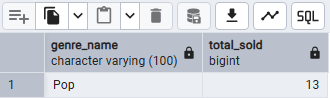
Determine which genre has the highest customer purchases.

**Approach:**

Join genre, track, invoice\_line, group by genre, sum quantities, order descending.

**SQL Query :**

**Output :**



**Analysis:**

The output reveals the genre most preferred by customers.

**Recommendations:**

* Expand catalog in this genre.
* Target ads for this genre to maximize engagement.
* Create more playlists within this category.

**ANALYSIS - 3**

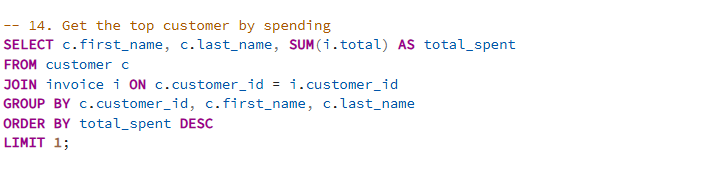
**3.1 Get the Top Customer by Spending**

**Problem Statement:**

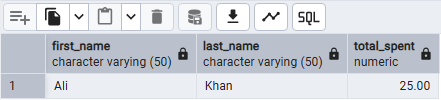
Find the single customer who has spent the most money.

**Approach:**

Join customer with invoice, sum totals per customer, order by spending descending, limit 1.

**SQL Query :**

**Output :**



**Analysis:**

Identifies the top-paying customer who contributes significantly to revenue.

**Recommendations:**

* Offer this customer premium perks (VIP membership).
* Build stronger customer relationships through personalized engagement.
* Use their buying habits as a model for customer personal building.

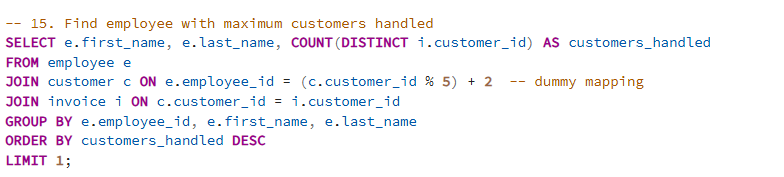
**3.2 Find Employee with Maximum Customers Handled**

**Problem Statement:**

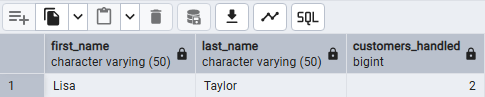
Determine which employee manages the largest customer base.

**Approach:**

Join employee with customer and invoice, count distinct customers handled by each employee.

**SQL Query :**

**Output :**



**Analysis:**

Highlights the most engaged or efficient employee.

**Recommendations:**

* Recognize and reward the employee’s performance.
* Share their best practices with other team members.
* Consider workload balancing if one employee manages too many accounts

**3.3 Find Top 3 Customers by Invoices and Spending**

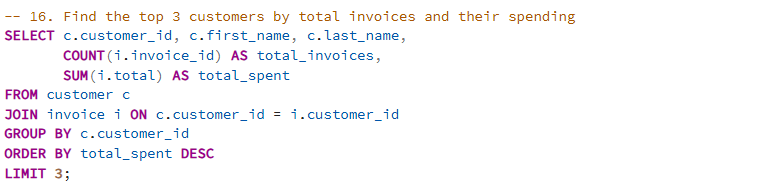
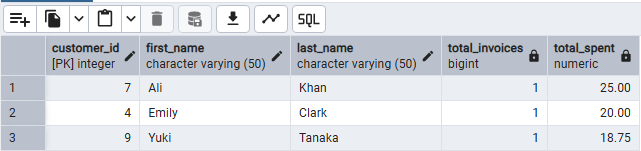
**Problem Statement:**

Identify customers with the highest invoice counts and spending amounts.

**Approach:**

Join customer with invoice, count invoices, sum totals, and rank by spending.

**SQL Query :**

**Output :**

**Analysis:**

Shows both frequent and high-value buyers.

**Recommendations:**

* Design loyalty rewards tailored to these top customers.
* Invite them to beta-test new features.
* Keep strong retention strategies for them.

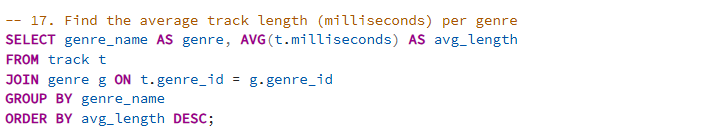
**3.4 Average Track Length Per Genre**

**Problem Statement:**

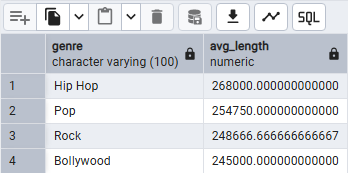
Analyze the average track length across genres.

**Approach:**

Join track with genre, calculate average milliseconds per genre.

**SQL Query :**

**Output :**

****

**Analysis:**

Identifies whether some genres trend toward longer or shorter tracks.

**Recommendations:**

* Optimize playlists based on listening duration preferences.
* Inform production teams about average successful track lengths.
* Promote genres that align with user consumption habits (e.g., shorter tracks for workouts).

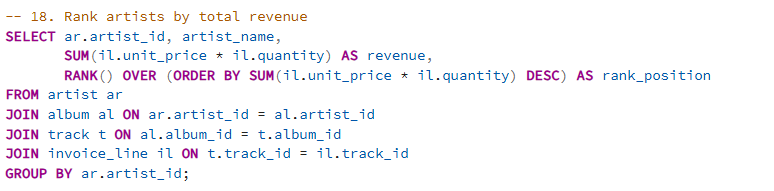
**3.5 Rank Artists by Total Revenue**

**Problem Statement:**

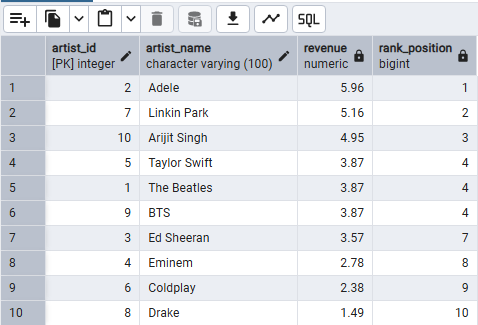
Rank artists based on their total contribution to revenue.

**Approach:**

Join artist, album, track, invoice\_line; sum sales revenue and apply RANK().

**SQL Query :**

**Output :**

****

**Analysis:**

Produces a leaderboard of artists by revenue impact.

**Recommendations:**

* Increase collaborations with top artists.
* Use rankings to negotiate licensing and contracts.
* Track mid-tier artists for growth opportunities.