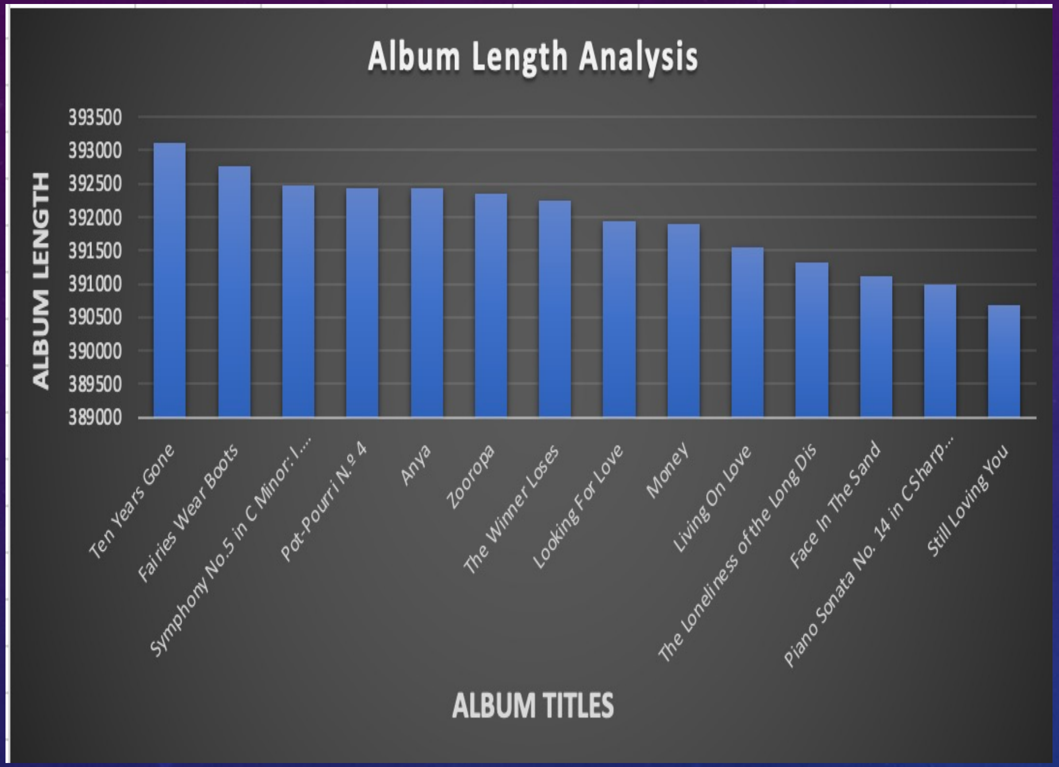
QUERY A DIGITAL MUSIC STORE DATABASE

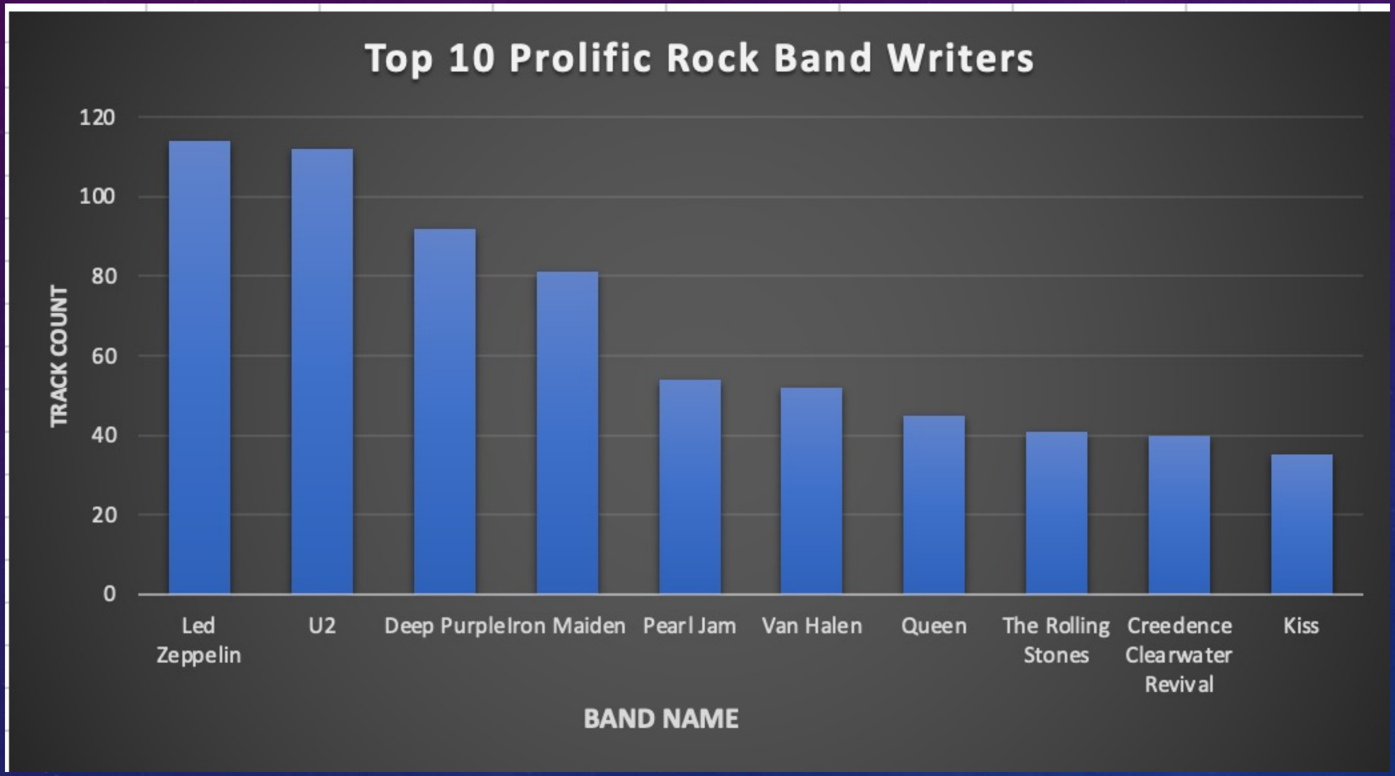
PROJECT 3

Q 1: ALBUM TRACK LENGTH ANALYSIS



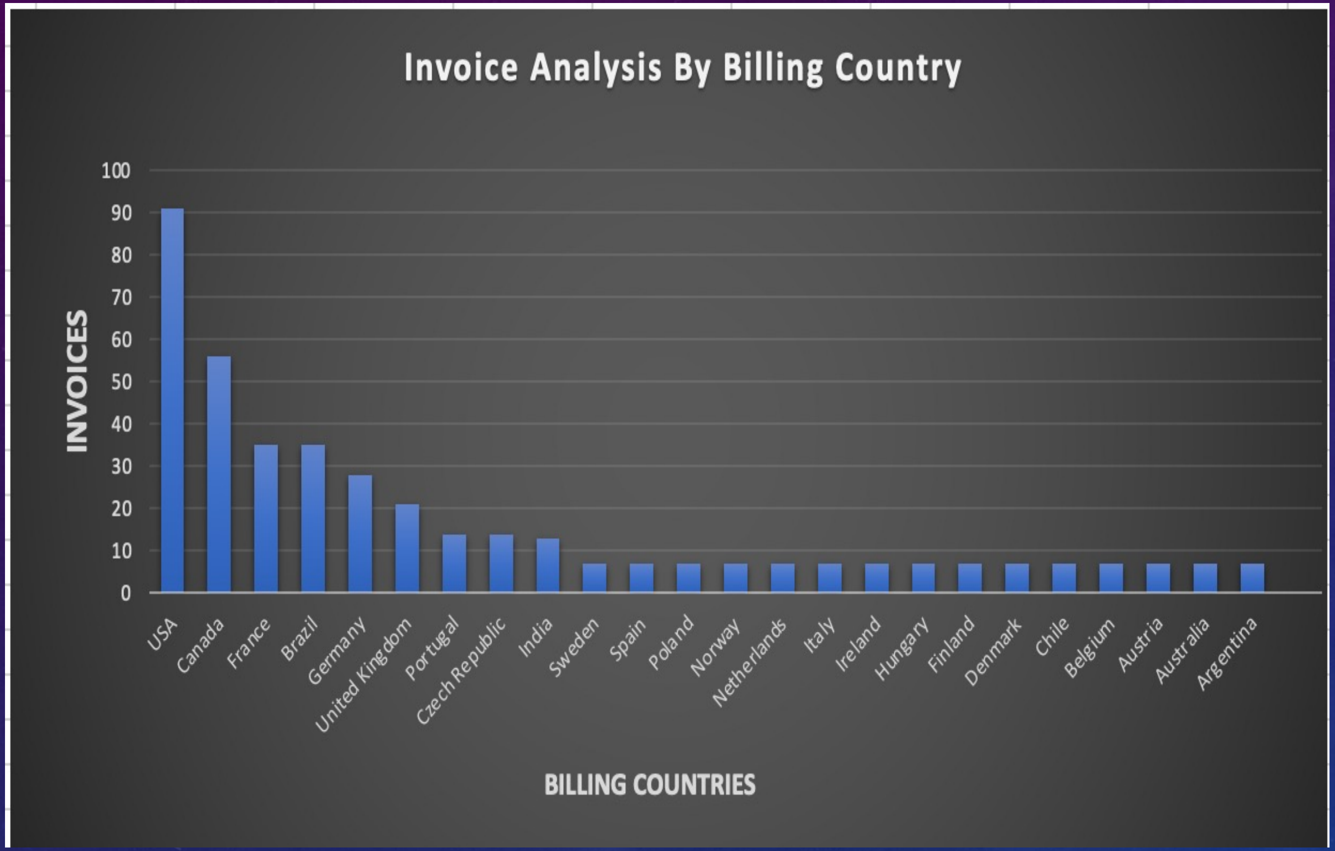
•This SQL query retrieves information about albums and their aggregated track lengths. By joining the "Album" and "Track" tables on the common "AlbumId" column, the query combines data from both tables. It calculates the total duration of tracks in each album using the SUM function on the "Milliseconds" columnand assigns an alias "SongLength_milliseconds" to this aggregated value. The query groups the results by album title and album ID using the GROUP BY clause. The final output is ordered in descending order based on the total track length, which is the "SongLength_milliseconds" column. This query essentially provides a list of albums along with their combined track lengths, arranged from albums with the longest total track lengths to the shortest.

Q2: TOP ROCK MUSIC ARTISTS: IDENTIFYING THE PROLIFIC TRACK WRITERS



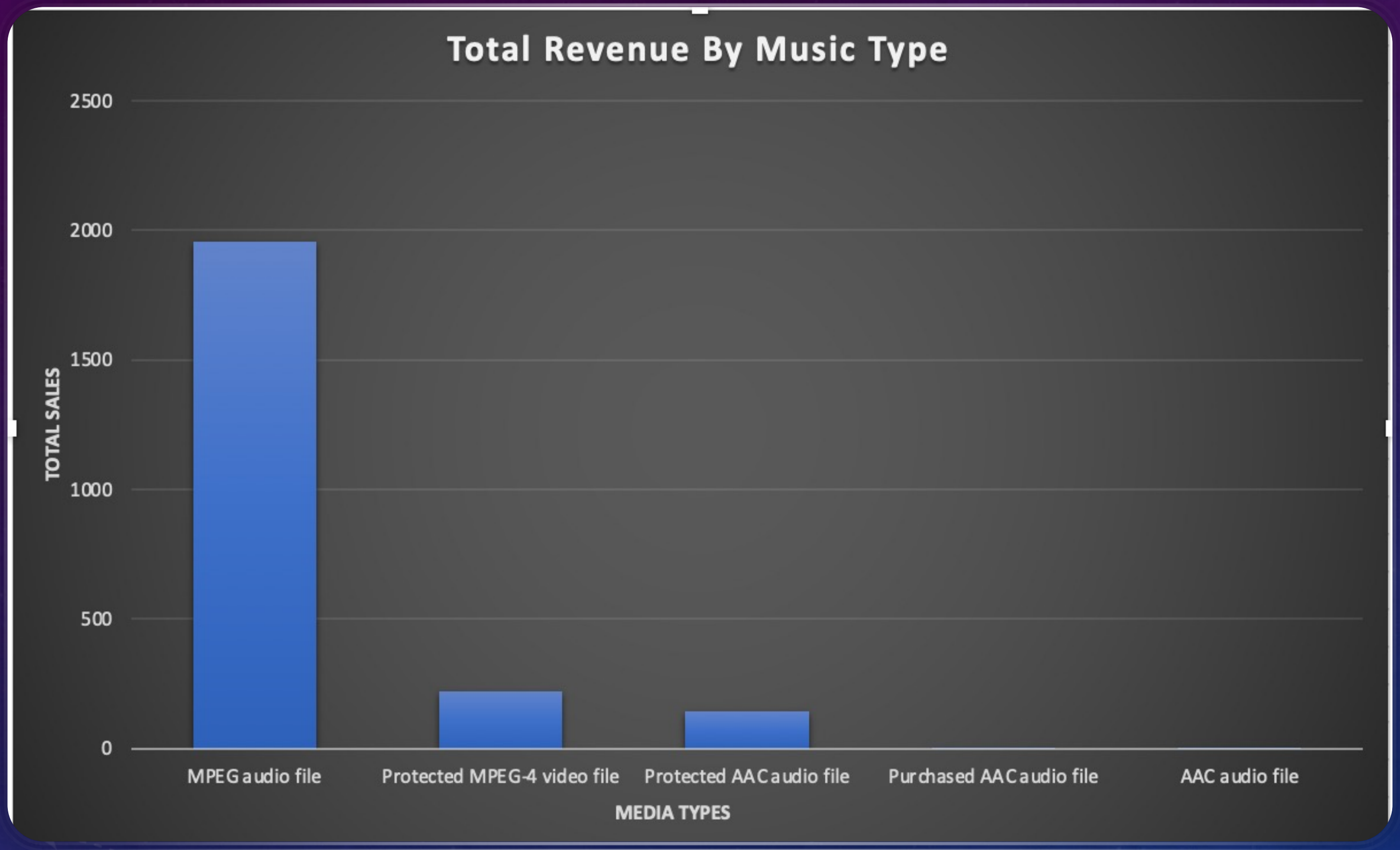
- This SQL query is designed to identify the top 10 artists who have written the most rock music tracks in a given dataset. The query leverages data from the "Artist," "Album," "Track," and "Genre" tables. By joining these tables based on their respective relationships, the query calculates the count of tracks authored by each artist within the rock genre. This count is associated with the artist's name and then grouped and ordered in descending order. The resulting list showcases the artists who have contributed the highest number of rock music tracks, allowing for informed decisions on which musicians to invite for a concert focused on rock music.

Q3:INVOICE DISTRIBUTION ANALYSIS BY BILLING COUNTRY



- This SQL query aims to analyze invoice distribution across different billing countries. By joining the "Invoice" table (aliased as "i") with the "Customer" table (aliased as "c") using the shared "CustomerId" column, the query establishes a connection between invoices and customers. Subsequently, the query calculates two metrics for each unique billing country: the total number of invoices ("TotalInvoices") and the count of distinct customers ("TotalCustomers"). The DISTINCT keyword is used to ensure each customer is counted only once. The results are grouped by billing country using the GROUP BY clause, allowing the aggregation of these metrics on a per-country basis. Lastly, the query orders the results in descending order based on the total number of invoices ("TotalInvoices"), showcasing the countries with the highest invoice counts at the top. This query offers insights into how invoices are distributed among various billing countries and provides an understanding of the customer diversity contributing to these invoices..

COMPARING REVENUE ACROSS MEDIA TYPES



This SQL query aims to analyze revenue generation across various media types. By utilizing data from the `InvoiceLine`, `Track`, and `MediaType` tables, the query computes the total revenue for each media type. This computation involves joining the tables through their related columns—specifically, connecting `InvoiceLine` with `Track` using the shared `TrackId`, and then linking `Track` with `MediaType` using the `MediaTypeId`. After establishing these relationships, the query calculates the total revenue by multiplying the `UnitPrice` and `Quantity` columns from `InvoiceLine`, representing the price and quantity of each track sold, respectively. The results are grouped by media type name (`MediaType`) using the `GROUP BY` clause, allowing the aggregation of revenue figures for each type. Finally, the outcomes are arranged in descending order based on the total revenue (`TotalRevenue`) using the `ORDER BY` clause. This query helps assess if different media types generate revenue at similar levels or if particular types stand out as more lucrative.