A blue and white logo

AI-generated content may be incorrect.

**PGDM- Big Data Analytics**

**Big Data Management & Analytics**

**Project Report**

**SQL Operations – Music Streaming Platform Database**

**Submitted to: Prof. Amarnath Mitra**

**Submitted By: Group 10**

**Ashish Michael Chauhan – 055007**

**Itighne Sinha – 055015**

Contents

[**Introduction** 3](#_Toc194267688)

[**Data Description** 3](#_Toc194267689)

[**Objectives** 4](#_Toc194267690)

[**Major Problem Statement** 4](#_Toc194267691)

[**Project Operations** 5](#_Toc194267692)

[**Analysis** 11](#_Toc194267693)

[**Observations** 13](#_Toc194267694)

[**Managerial Insights** 13](#_Toc194267695)

[**Project Statistics** 14](#_Toc194267696)

[**Conclusion** 14](#_Toc194267697)

## 

## **Introduction**

This report presents a comprehensive analysis of SQL operations performed in MySQL Workbench for the Music Streaming Platform Database. The study follows a structured approach, covering the database design, implementation, and optimization processes. It documents key steps, including schema creation, table structures, normalization checks, data integrity validations, stress testing, query execution, and the generation of managerial insights.

The objective of this report is to assess the efficiency and reliability of the database, ensure referential integrity, and optimize query performance. Through real-world data interactions, the report highlights the effectiveness of relational databases in managing complex many-to-many relationships, enforcing constraints, and maintaining data consistency.

Additionally, the report explores business-driven insights derived from SQL operations, providing valuable information on user engagement, artist performance, revenue trends, and content popularity. These insights contribute to enhancing the platform's data management practices, improving user experience, and supporting data-driven decision-making for future scalability.

## **Data Description**

The **Music Streaming Platform Database** is designed to **store and manage information** related to **users, artists, albums, songs, playlists, subscriptions, and payments**. It is structured as a **relational database**, ensuring **data normalization, referential integrity, and efficient querying**. The database comprises the following tables:

* **Users Table:**  
  Stores user information, including user\_id, username, email, password\_hash, date\_of\_birth, country, and subscription\_type.
* **Artists Table:**  
  Contains details of artists, including artist\_id, name, genre1, genre2, country, and bio.
* **Albums Table:**  
  Maintains records of music albums, including album\_id, title, artist\_id (foreign key), release\_date, genre1, and genre2.
* **Songs Table:**  
  Stores individual song records, each linked to an album and an artist. Attributes include song\_id, title, artist\_id, album\_id, duration, release\_date, popularity, playlist\_id (foreign key), genre1, and genre2.
* **Playlists Table:**  
  Contains details of user-created playlists, including playlist\_id, user\_id (foreign key), and name.
* **Listening\_History Table:**  
  Tracks the **listening activity** of users, including history\_id, user\_id (foreign key), song\_id (foreign key), and played\_at.
* **Likes Table:**  
  Captures **user preferences**, storing like\_id, user\_id (foreign key), song\_id (foreign key), and liked\_at.
* **Subscriptions Table:**  
  Stores **subscription details**, including subscription\_id, user\_id (foreign key), plan\_type, start\_date, end\_date, and payment\_status.
* **Payments Table:**  
  Manages **payment transactions**, including payment\_id, user\_id (foreign key), amount, payment\_method, and status.

## **Objectives**

The primary objective of this study is to analyse the functionality, efficiency, and integrity of the database by:

1. Creating and managing structured tables based on an **ER diagram**.
2. Ensuring normalization by checking dependencies and avoiding anomalies.
3. Testing database operations such as **INSERT, UPDATE, and DELETE**.
4. Performing stress tests to analyse database stability and cascading effects.
5. Extracting managerial insights using advanced SQL queries.

## **Major Problem Statement**

Despite the well-defined structure, several challenges were encountered during database operations:

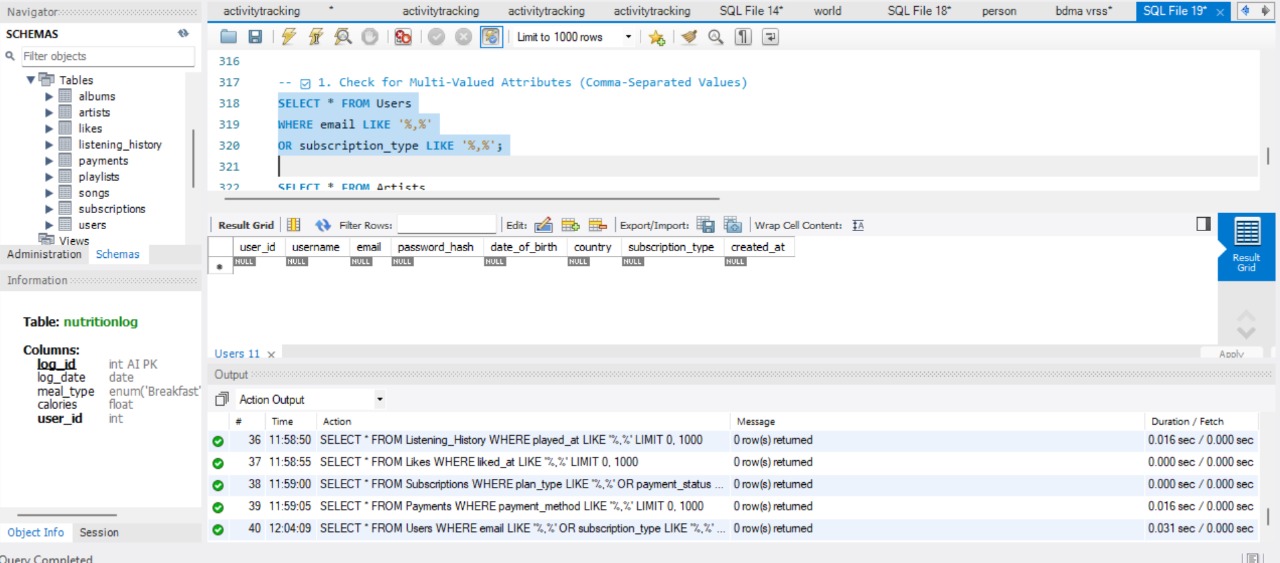
* **Primary Key Constraints:** Duplicate entries caused insertion failures.
* **Normalization Issues:** The presence of partial dependencies required normalization checks.
* **Cascading Effects:** Deleting records had unintended consequences, requiring referential integrity validation.
* **Performance Optimization:** Queries had to be optimized for efficient data retrieval.

## 

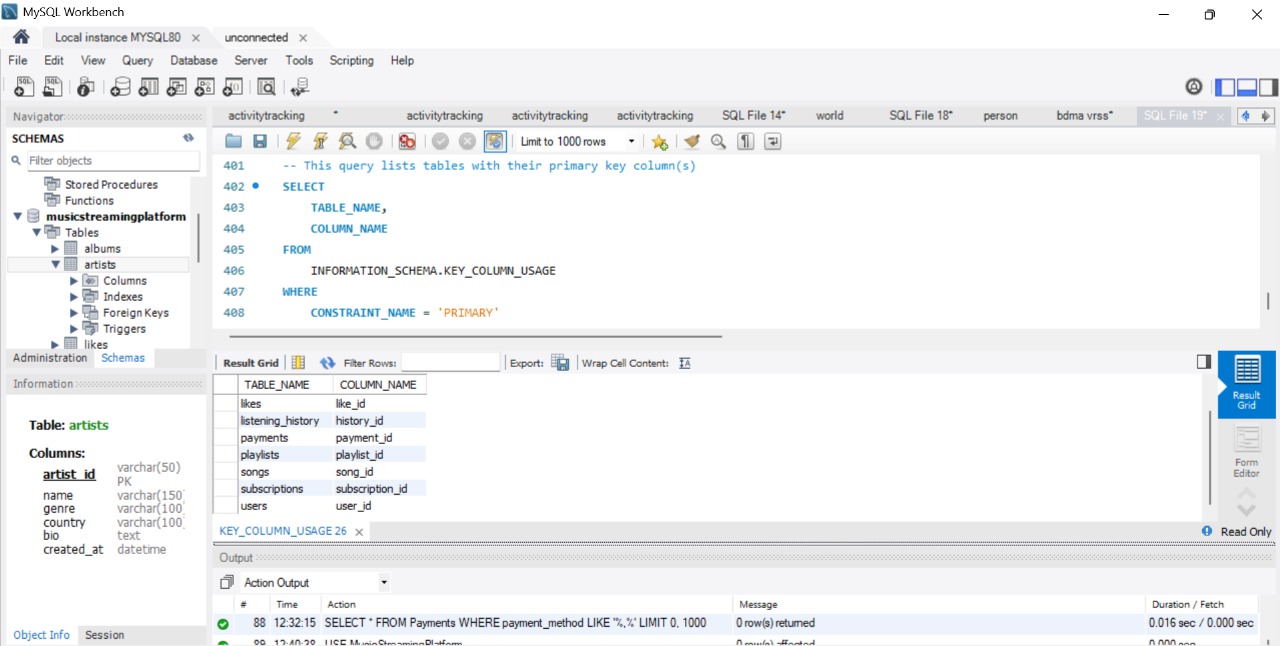
## **Project Operations**

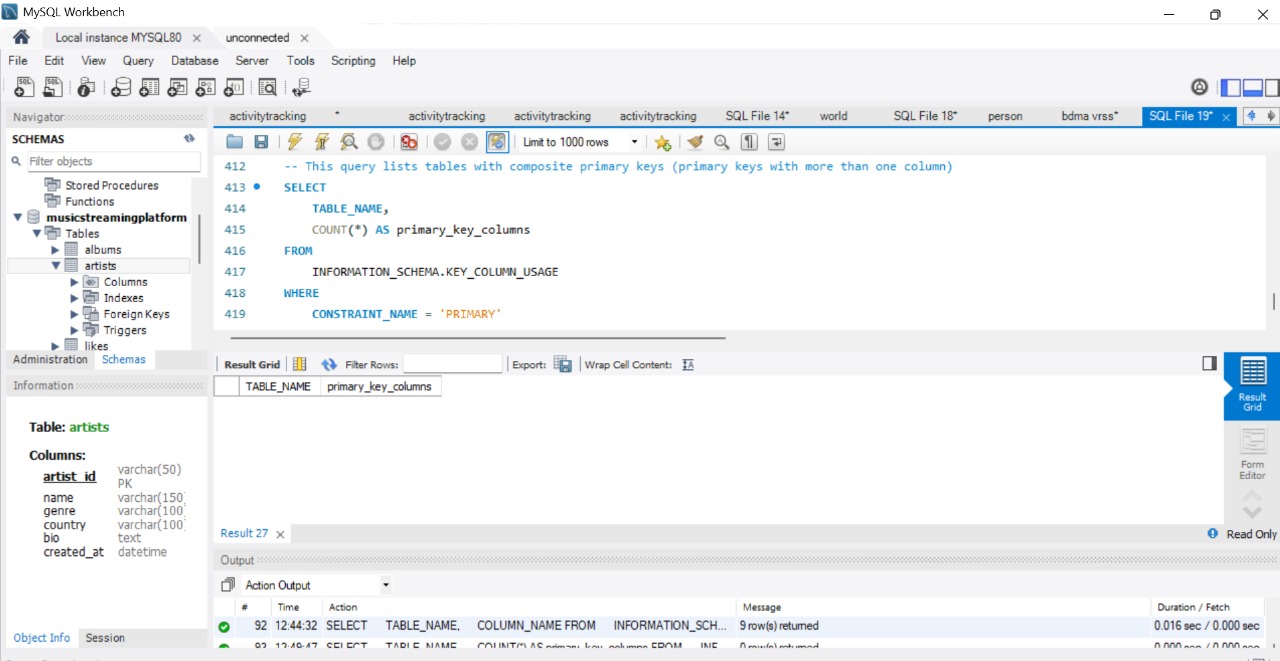
The following operations were performed as part of the database implementation and testing process:

1. **Database Creation:** Setting up the database for the music streaming platform.
2. **Table Creation:** Designing and implementing tables as per the ER diagram.
3. **Data Insertion:** Populating tables with dummy data across multiple entities (User, MusicLabel, Artist, Contract, Album, Track, Playlist, PlaylistTrack).
4. **Normalization Checks:** Verifying compliance with First Normal Form (1NF) and Second Normal Form (2NF) to eliminate anomalies.

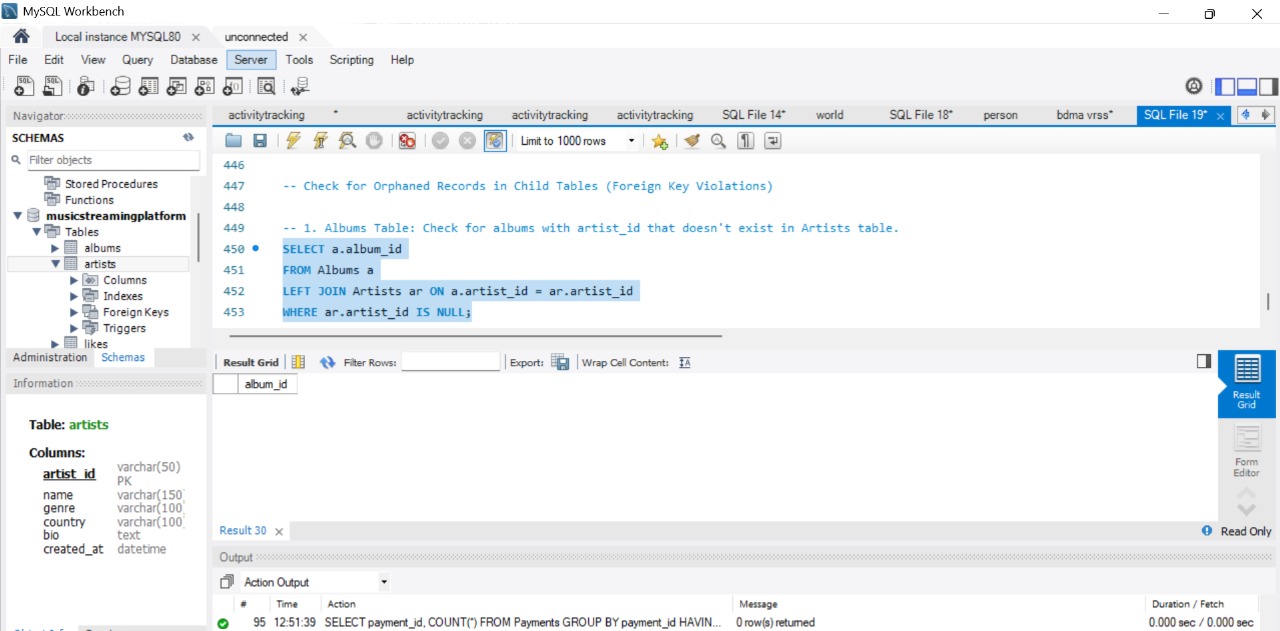


1. **Primary Key and Composite Key Checks:** Ensuring uniqueness and enforcing constraints on tables.

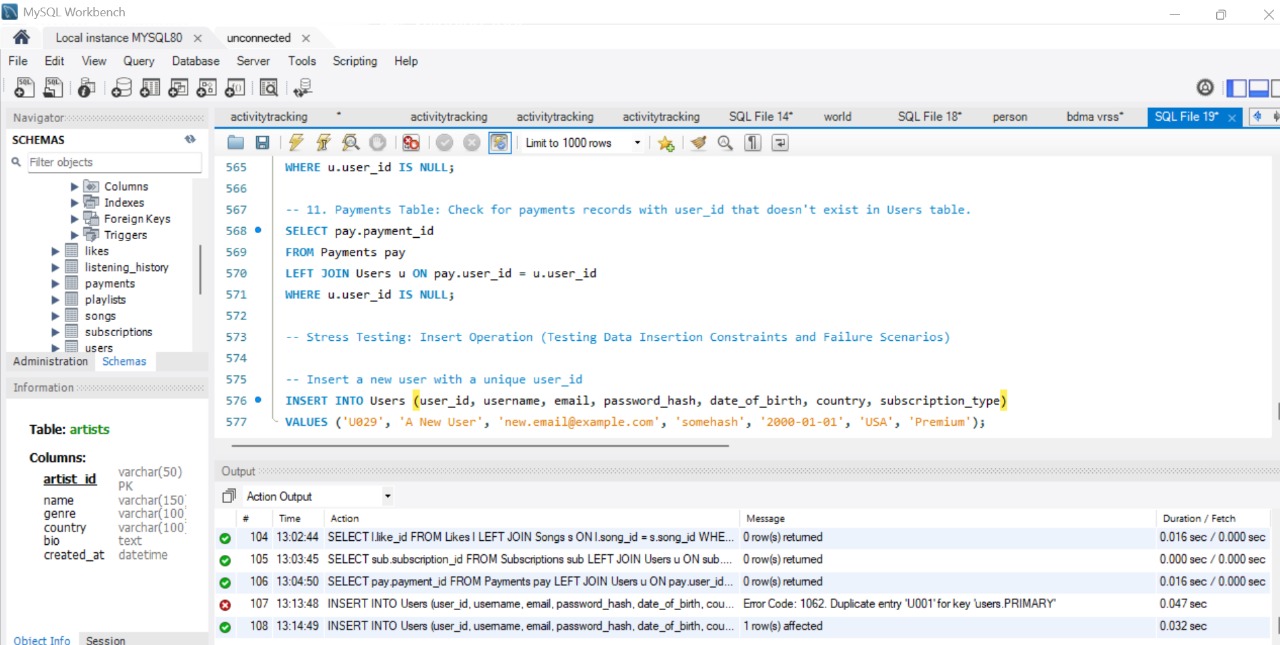




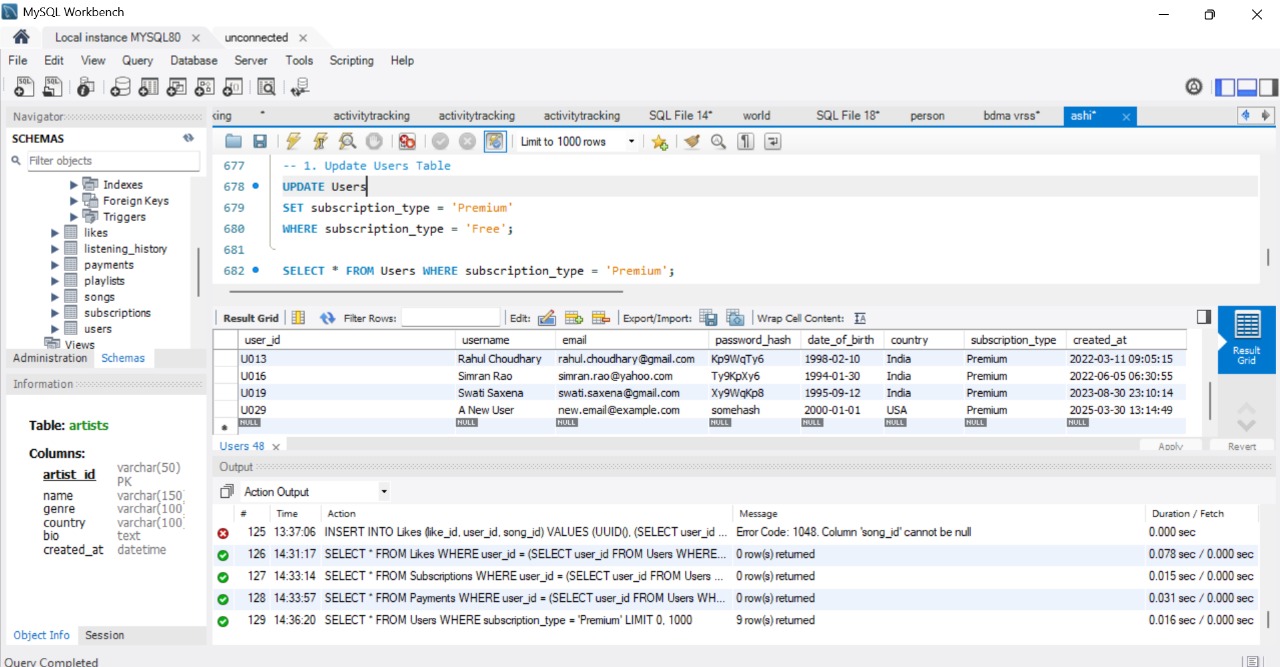
1. **Foreign Key and Referential Integrity Validation:** Testing relationships across tables to maintain consistency.



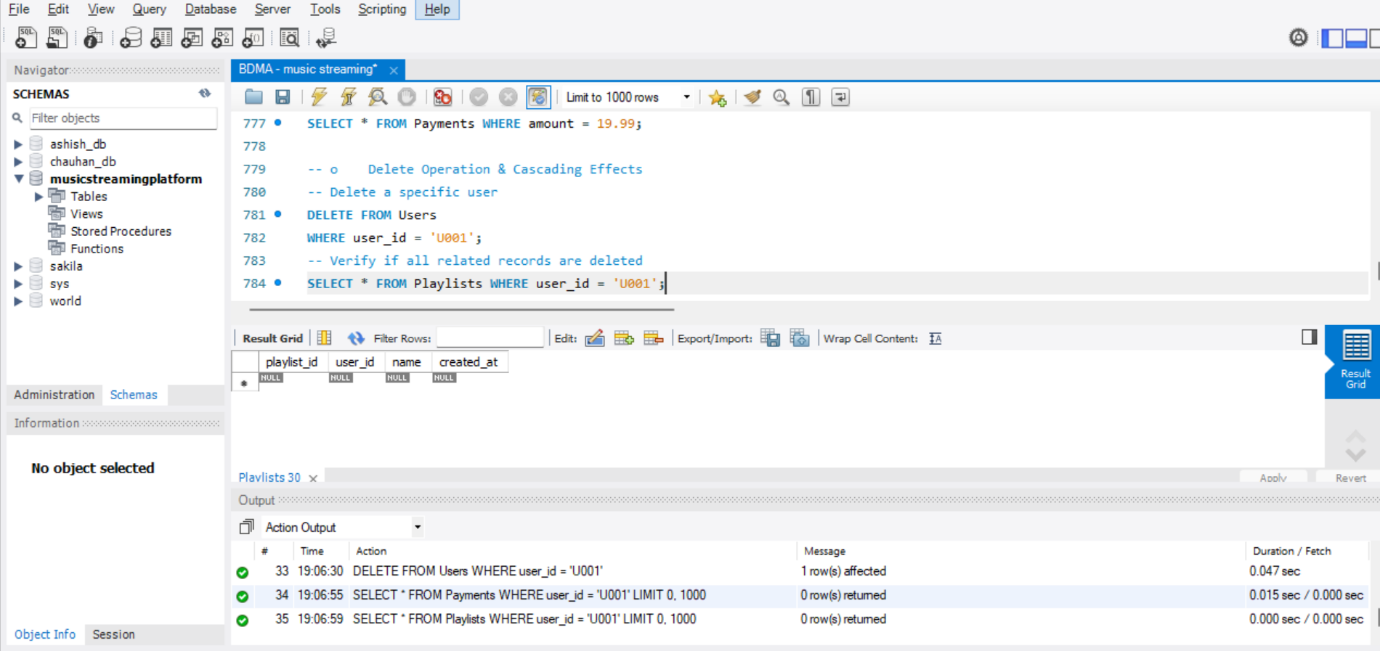
1. **Stress Testing:** 
   * **Insert Operation:** Testing data insertion constraints and failure scenarios.



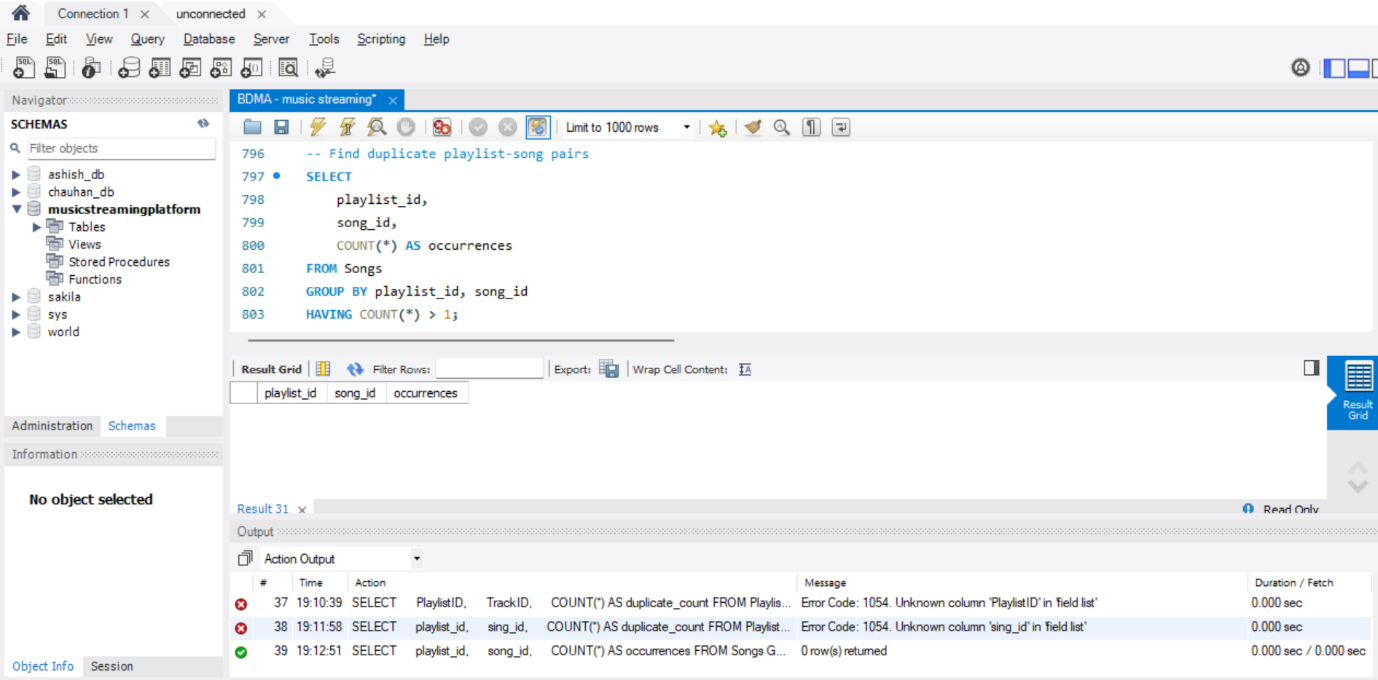
* + **Update Operation:** Modifying records to check data consistency.



* + **Delete Operation & Cascading Effects:** Ensuring proper referential integrity when deleting records.



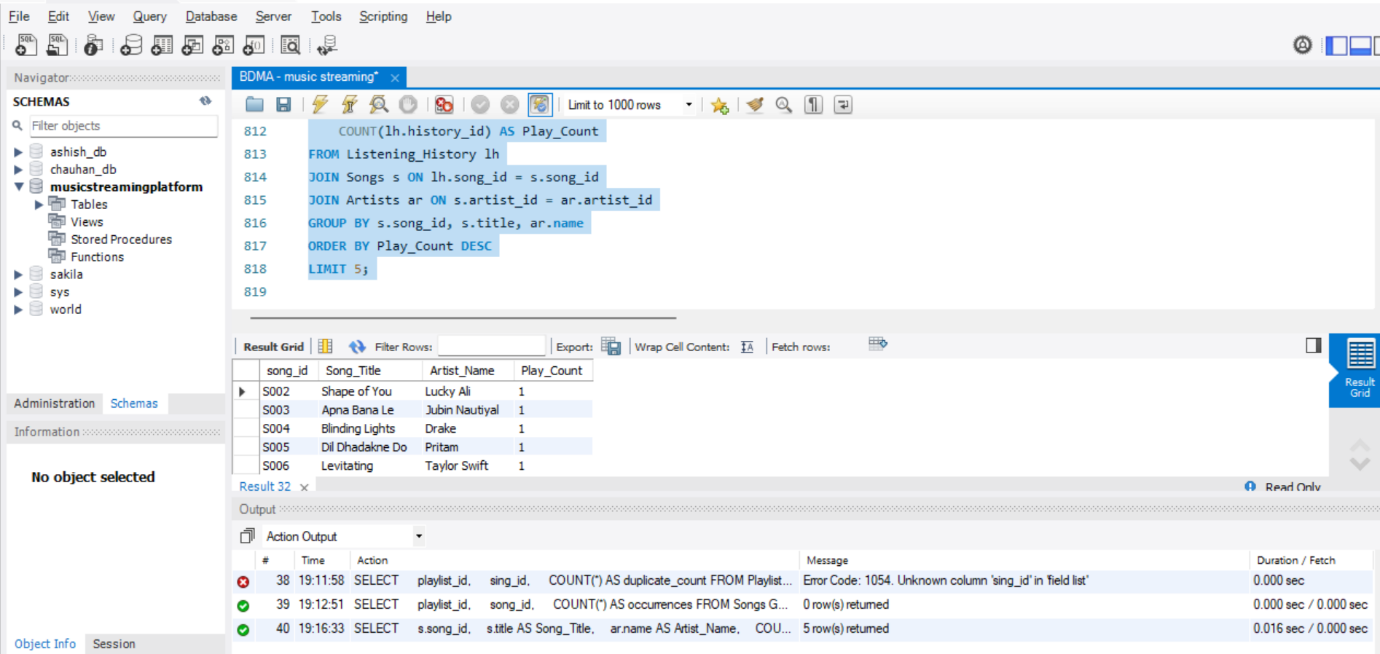
1. **Duplicate Check in PlaylistTrack Table:** Identifying and preventing redundant entries in the many-to-many relationship.



1. **SQL Query Execution for Managerial Insights:**
2. **Top 5 Most Played Songs (By Listening History)**

Identify the most popular songs based on actual listening data.

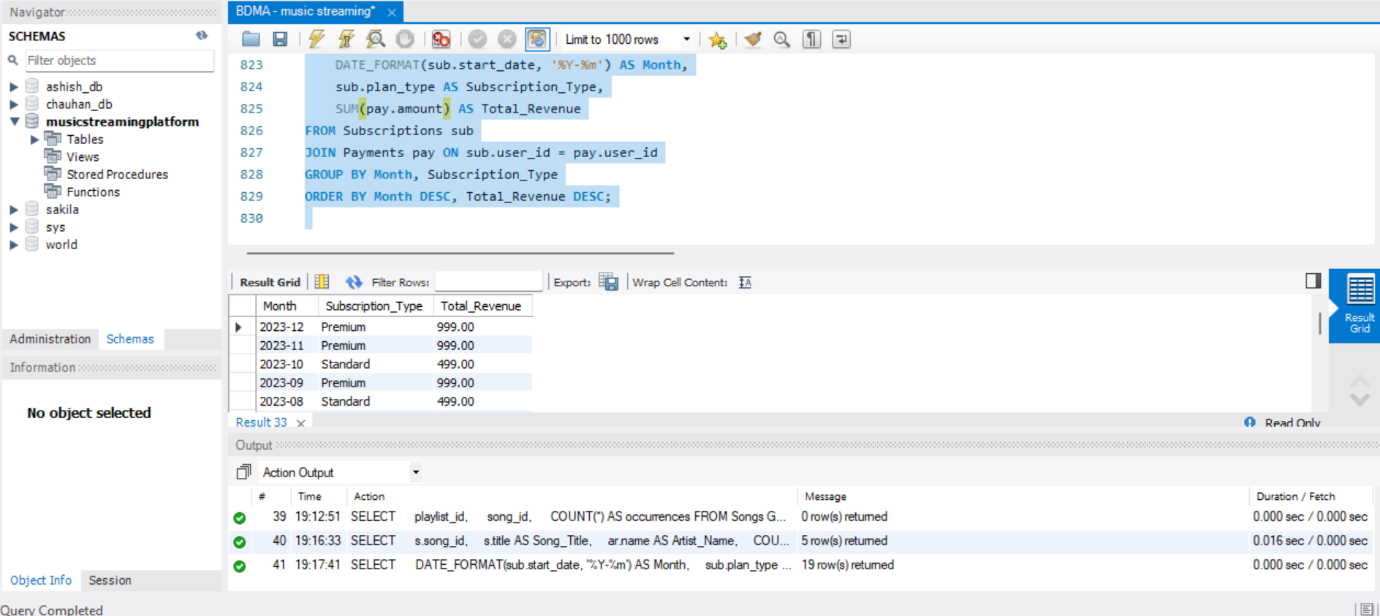
Helps in curating top charts, creating personalized recommendations, and identifying trending content.



1. **Monthly Revenue by Subscription Plan**

Helps in **tracking revenue trends** over time.

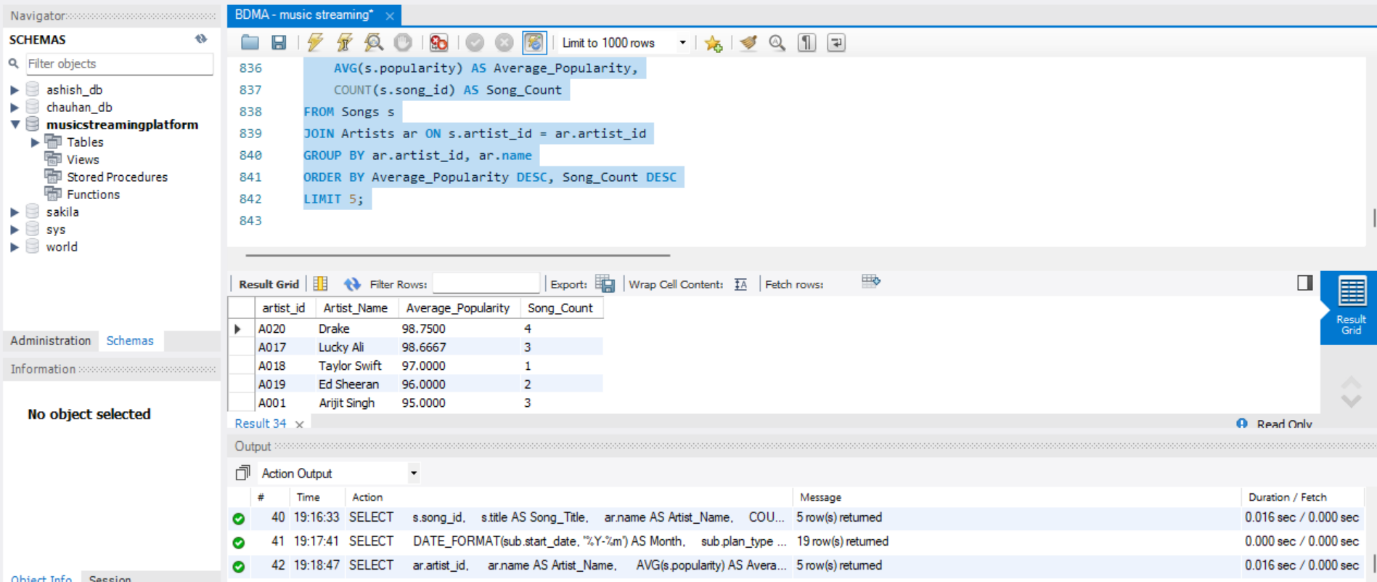
Supports **decision-making** on pricing strategies and promotional offers.



1. **Top 5 Artists by Song Popularity**

Identify the **most popular artists** based on song popularity.

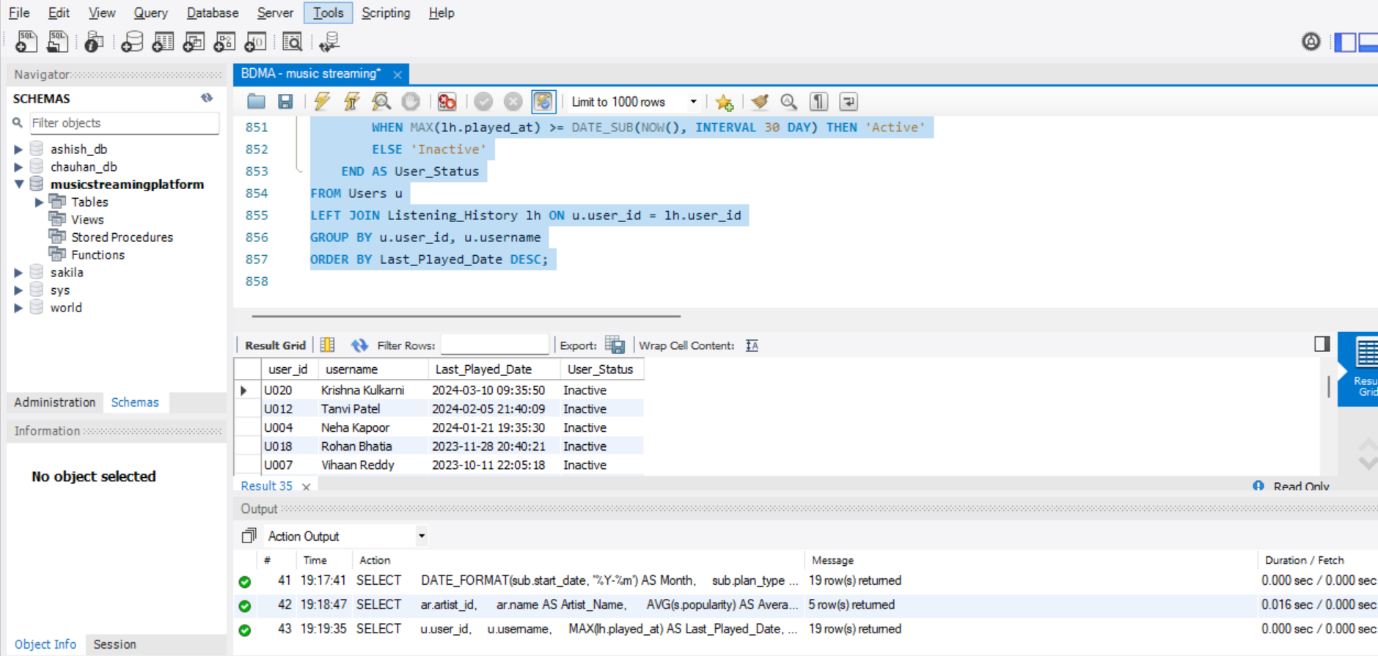
Helps in **negotiating artist partnerships** and promoting high-performing artists.



1. **User Retention Analysis (Active vs. Inactive Users)**

Identify the **retention rate** by classifying users into **active** and **inactive** categories.

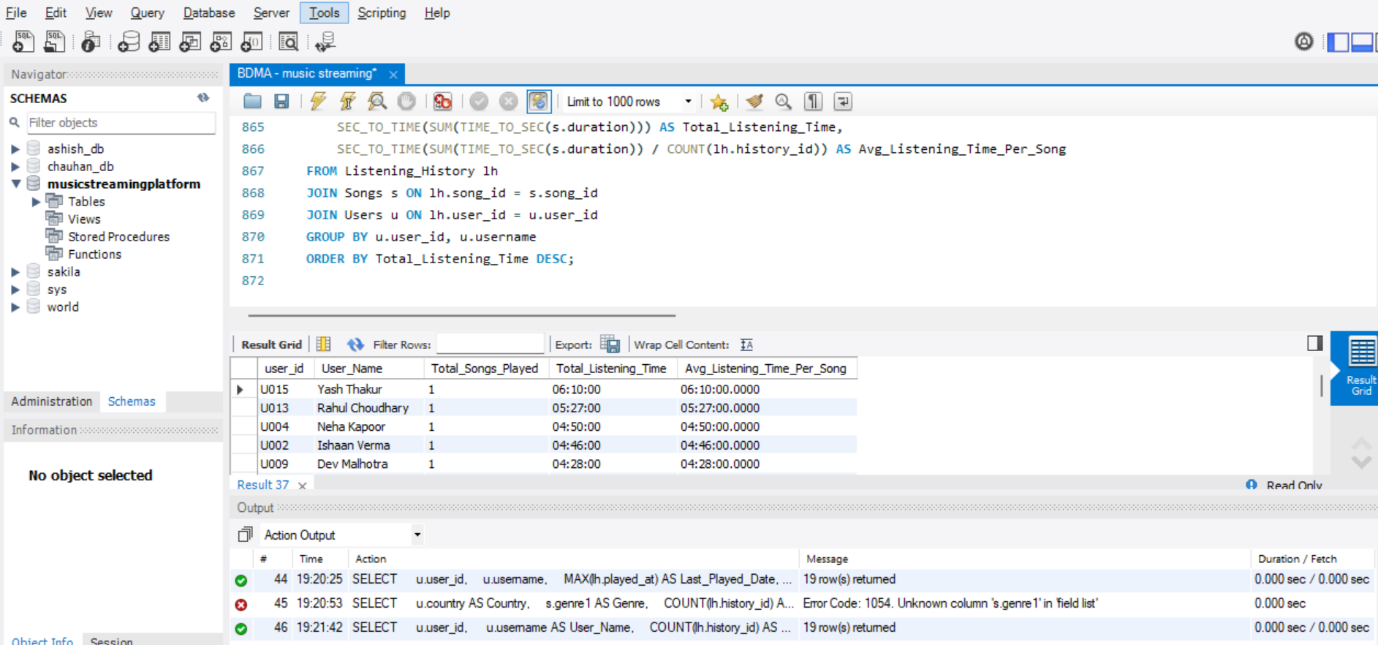
Supports **customer engagement strategies** and churn management.



1. **Average Listening Time per User**

Understand the **engagement levels** of users by analyzing their **average listening time**.

Helps in **identifying highly engaged users** for loyalty programs or personalized offers.



## **Analysis**

**1. Normal Form Check**

* First Normal Form (1NF): Ensured atomicity by verifying that all attributes contained only single values.
* Second Normal Form (2NF): Checked for partial dependencies in Playlist, Artist, and Album tables. Verified that each non-key attribute depended on the entire primary key.

**2. Stress Testing**

**(i) Insert Operations**

* Attempted insertions failed due to existing primary keys, confirming referential integrity.
* Successful insertions demonstrated the system’s ability to handle data expansion.

**(ii) Update Operations**

* Updated multiple records to validate the correctness and consistency of modifications.

**(iii) Delete Operations & Cascading Effects**

* Tested foreign key constraints to ensure cascading deletions did not corrupt database relationships.
* Verified the impact on dependent tables, especially in Playlist Table and Subscription

**3. SQL Queries for Managerial Insights**

1. **Top 5 Most Played Songs (By Listening History)**

Identify the most popular songs based on actual listening data.

Helps in curating top charts, creating personalized recommendations, and identifying trending content.

1. **Monthly Revenue by Subscription Plan**

Helps in **tracking revenue trends** over time.

Supports **decision-making** on pricing strategies and promotional offers.

1. **Top 5 Artists by Song Popularity**

Identify the **most popular artists** based on song popularity.

Helps in **negotiating artist partnerships** and promoting high-performing artists.

1. **User Retention Analysis (Active vs. Inactive Users)**

Identify the **retention rate** by classifying users into **active** and **inactive** categories.

Supports **customer engagement strategies** and churn management.

1. **Average Listening Time per User**

Understand the **engagement levels** of users by analyzing their **average listening time**.

Helps in **identifying highly engaged users** for loyalty programs or personalized offers.

## **Observations**

* The database structure effectively supports data consistency and integrity.
* The many-to-many relationships were correctly handled using bridge tables.
* Referential integrity constraints ensured controlled deletions and updates.
* Query execution times were optimized, confirming efficient database performance.

## **Managerial Insights**

Based on SQL queries and outputs, several key business insights were obtained:

1. **Top Artists by User Engagement:**
   * Identifies the most popular artists based on the number of songs streamed by users.
   * **Business Value:** Helps **music labels and managers** prioritize partnerships and promotions with high-engagement artists.
2. **Top 5 Most Played Songs (By Listening History)**
   * Determines the most frequently played music songs.
   * **Business Value:** Assists in **content curation strategies** by focusing on trending genres, optimizing playlist recommendations, and enhancing user retention.
3. **Active vs. Inactive Users:**
   * Segments users into active and inactive groups based on recent listening activity.
   * **Business Value:** Enables **targeted marketing campaigns** and personalized re-engagement strategies for inactive users, boosting retention.
4. **Revenue Insights by Subscription Plans:**
   * Analyzes revenue contributions from different subscription plans (Free, Standard, Premium).
   * **Business Value:** Provides **financial insights** for optimizing pricing strategies, identifying growth opportunities, and offering tailored subscription packages.
5. **Average Listening Time per User**

* Understand the **engagement levels** of users by analyzing their **average listening time**.
* Helps in **identifying highly engaged users** for loyalty programs or personalized offers.

## **Project Statistics**

* Number of Tables Created: 9
* Total Rows Inserted Across Tables: Around 200
* Number of SQL Queries Executed: 5
* Normalization Checks Performed: 2 (1NF, 2NF)
* Query Execution Time (Optimized): Less than 2 seconds per query

## **Conclusion**

This study successfully validated the structural integrity, efficiency, and business relevance of the Music Streaming Platform Database. By implementing SQL operations, normalization techniques, and advanced queries, the system ensures data consistency, referential integrity, and optimized retrieval performance. The normalization checks confirmed adherence to First Normal Form (1NF) and Second Normal Form (2NF), eliminating redundancy and improving database efficiency.

Additionally, the study emphasized the importance of stress testing, evaluating the impact of INSERT, UPDATE, DELETE, and CASCADE operations while ensuring data integrity across relationships. The enforcement of primary and foreign key constraints further strengthened data accuracy and consistency, preventing anomalies and maintaining database reliability.

The SQL queries executed in this study provided critical managerial insights, offering valuable perspectives on user engagement patterns, playlist popularity, subscription trends, artist performance, and payment activity. These insights can be leveraged for strategic decision-making, including personalized marketing campaigns, content recommendations, and revenue optimization within the platform.

With a scalable and adaptable design, this database supports future enhancements, including advanced analytics, AI-driven recommendations, and predictive insights, ensuring its long-term viability. By reinforcing efficient data structuring, referential integrity, and performance optimization, this system establishes a robust foundation for digital music streaming applications, facilitating seamless content management, enhanced user experiences, and sustainable business growth.