Music Store Analysis

Introduction: The Music Store Analysis project is a comprehensive SQL project aimed at analyzing the data from an online music store. The primary objective of this project is to assist the music store in understanding its business growth by answering various questions and providing insights into its operations. With the availability of relevant datasets and SQL queries, this project enables us to gain valuable insights into the store's performance, customer behavior, and sales trends.

Project Details:

SQL Version: MySQL version 5.7.43-log

Datasets: The project leverages various datasets related to the online music store,

Below is an overview of the core tables within the database:

1. Employee Table: The "employee" table stores information about the store's staff, including employee ID, names, titles, contact details, birthdate, hire date, and address information. Additionally, the table includes a "reports_to" field to establish a hierarchical structure within the organization and a "levels" field for categorizing employee levels.

2. Customer Table: The "customer" table contains data about the store's customers, including customer ID, names, company (if applicable), contact information, and geographical details such as address, city, state, country, and postal code. It also records customer support representative assignments through the "support_rep_id" field.

3. Invoice Table: The "invoice" table stores information about sales transactions, including invoice ID, customer ID, invoice date, billing address, billing city, state, country, postal code, and total amount. This table is crucial for tracking sales and revenue.

4. Invoice Line Table: The "invoice_line" table is used to record individual line items within an invoice. It includes details such as invoice line ID, invoice ID, track ID, unit price, and quantity, enabling a granular view of each transaction.

5. Track Table: The "track" table contains data on individual music tracks available in the store. It includes track ID, track name, associated album ID, media type ID, genre ID, composer information, duration in milliseconds, file size in bytes, and unit price.

6. Playlist Table: The "playlist" table maintains a list of playlists available in the music store. It includes a playlist ID and the name of the playlist.

7. Playlist Track Table: The "playlist_track" table establishes a many-to-many relationship between playlists and tracks. It links playlist IDs with track IDs, allowing multiple tracks to belong to different playlists and vice versa.

8. Artist Table: The "artist" table stores information about music artists. It includes an artist ID and the artist's name. This table is linked to the "album" table to associate artists with their albums.

9. Album Table: The "album" table contains details about music albums, including album ID, title, and the associated artist ID. This table establishes the connection between artists and their albums.

10. Media Type Table: The "media_type" table holds information about different media types, such as CD, MP3, etc. It uses a media type ID and name to categorize media types.

11. Genre Table: The "genre" table categorizes music tracks into various genres, using genre IDs and genre names.

Questions and SQL queries

The Music Store Management System's database design efficiently organizes employee, customer, sales, and music track data. Questions and SQL queries play a crucial role in extracting insights. They help identify senior employees, customer distribution, top invoices, the best customer, and more. Additionally, they target Rock music listeners, invite top bands, and enhance the music experience with longer tracks. These queries enable data-driven decisions, personalized marketing, and improved customer service in the music store.

Conclusion: This database design provides a solid foundation for managing and analyzing data in an online music store. It allows for efficient tracking of employees, customers, sales transactions, music tracks, playlists, artists, albums, media types, and genres. By using these interconnected tables, the music store can optimize its operations, make data-driven decisions, and enhance the customer experience. Additionally, this database design can be extended and customized to meet specific business requirements and analytical needs of the music store.