Module 4 Assignment

A.Optimizing Music Database Queries for Enhanced Data Analysis

Dataset: 🔃 **Track**
　　　　 🔃 **Genre**

Business Problem:
Our music database, comprising the `track` and `genre` tables, requires efficient and insightful queries to extract meaningful data for business analysis. By leveraging different types of SQL joins and set operations, we aim to enhance our understanding of track distribution across genres, identify trends, and improve data-driven decision-making processes.

1. How can we get a list of all tracks along with their corresponding genre names?
2. How can we list all tracks and their genre names, including tracks that do not have a genre assigned?
3. How can we list all genres and their tracks, including genres that do not have any tracks?
4. How can we create a combination of each track with each genre to explore all possible pairings?
5. How can we get a combined list of track names from two different criteria, such as tracks longer than 300,000 milliseconds and tracks priced above 0.99?
6. How can we get a combined list of track names from two different criteria, including duplicates?
7. How can we get all tracks that belong to the genre 'Rock'?
8. How can we count the number of tracks in each genre, including genres with no tracks?
9. How can we find genres that do not have any tracks?
10. How can we list all possible track-genre pairings for genres where no tracks are currently assigned?

B.Utilizing Advanced Data Analysis on Titanic Data

Business Problem:

The Titanic dataset provides a wealth of information about the passengers on the ill-fated voyage. To derive deeper insights and make data-driven decisions, we need to utilize subqueries effectively. This will allow us to perform complex queries such as finding passengers based on nested conditions, calculating aggregated values, and filtering results dynamically.

Dataset: 🗂️ Titanic

1. How can we list passengers who paid a fare higher than the average fare?
2. How can we find all passengers who travelled in the same class as 'Julia Patel'?
3. How can we find passengers who embarked from the most common embarkation town?
4. How can we list all survivors whose age is below the average age of all passengers?
5. How can we find the top 10 passengers who paid the highest fare?
6. How can we list passengers who were in classes with above-average survival rates?
7. How can we find passengers who were on decks with the lowest average fare?
8. How can we list passengers whose fare was higher than the average fare of their travel class?
9. How can we find passengers whose age is exactly the average age of passengers from their embarked town?
10. How can we list passengers who travelled in the deck number that was most commonly used?

C.Advanced Data Analysis on Titanic Dataset Using Window Functions

Business Problem:

The Titanic dataset contains detailed information about passengers, including survival status, class, age, fare, and more. To derive deeper insights and answer complex analytical questions, we will employ advanced SQL window functions such as `LEAD`, `LAG`, `RANK`, `DENSE_RANK`, and `ROW_NUMBER`. These functions will help us understand passenger trends, survival patterns, and other important aspects of the dataset.

Dataset: 🔢 Titanic

1. How can we identify the next passenger (in terms of passenger ID) for each passenger within the same class?
2. How can we compare the fare paid by each passenger with the fare paid by the previous passenger within the same class?
3. How can we rank passengers based on the fare they paid within each class?
4. How can we densely rank passengers based on their age within each class?
5. How can we assign row numbers to passengers based on the town they embarked from?
6. How can we identify the next female passenger (in terms of passenger ID) for each female passenger within the same class?
7. How can we compare the age of each male passenger with the age of the previous male passenger within the same class?
8. How can we rank female passengers based on the fare they paid within each class?
9. How can we densely rank passengers who survived based on their age within each class?
10. How can we assign row numbers to passengers who embarked from

'Southampton'?