Movie Review & Recommendation Engine using SQL

Introduction

In the age of digital entertainment, providing relevant movie recommendations is vital to enhance user experience. This project demonstrates how SQL can be used to extract meaningful insights and suggest top-rated movies based on genres using structured data.

Abstract

This project involves querying a movie database to build a simple recommendation engine. The objective is to retrieve the top 3 highest-rated movies per genre using SQL. It uses aggregate functions and subqueries to process user ratings, identify genre-wise popularity, and present results sorted by average ratings.

Tools Used

- **MySQL** (Database Engine)
- **DBeaver** (SQL IDE)
- **Git & GitHub** (Version Control)
- **DB Fiddle** (for online testing and sharing)

Steps Involved in Building the Project

1. Database Setup

   Created tables for **Movies, Genres, Ratings**, and **Users**.

2. Data Insertion

   Populated tables with sample data covering various movie genres and ratings by multiple users.

3. Query Design
   - Joined Movies, Genres, and Ratings.
   - Grouped data by genre and movie.
   - Calculated average rating per movie.
   - Used window functions or subqueries to fetch the top 3 per genre.

4. Testing & Output

   Tested queries using DB Fiddle and exported the results.

5. Documentation
   Results, screenshots, and code were uploaded to GitHub with a proper README.md.

Conclusion

This project showcases the practical use of SQL in analyzing data to solve real-world problems. It strengthens understanding of joins, aggregation, and subqueries, and demonstrates how SQL can be applied to build foundational recommendation systems in data-driven applications.