**Project 1: IMDb Movies Analysis using SQL**

Bolly Movies, an Indian film production company, has a successful track record of producing numerous blockbuster films. While primarily catering to the Indian audience, they have decided to venture into the global market with their upcoming project scheduled for release in 2022.

Objective:

Recognizing the value of data-driven decision-making, Bolly Movies has enlisted your expertise as a data analyst and SQL specialist. The objective of this case study is to analyse the movie dataset using SQL queries and extract valuable insights to guide Bolly Movies in planning their new project. The analysis will cover various aspects such as table exploration, movie release trends, production statistics, genre popularity, ratings analysis, crew members, and more.

Segment 1: Database - Tables, Columns, Relationships

* What are the different tables in the database and how are they connected to each other in the database?
* Find the total number of rows in each table of the schema.
* Identify which columns in the movie table have null values.

Segment 2: Movie Release Trends

* Determine the total number of movies released each year and analyse the month-wise trend.
* Calculate the number of movies produced in the USA or India in the year 2019.

Segment 3: Production Statistics and Genre Analysis

* Retrieve the unique list of genres present in the dataset.
* Identify the genre with the highest number of movies produced overall.
* Determine the count of movies that belong to only one genre.
* Calculate the average duration of movies in each genre.
* Find the rank of the 'thriller' genre among all genres in terms of the number of movies produced.

Segment 4: Ratings Analysis and Crew Members

* Retrieve the minimum and maximum values in each column of the ratings table (except movie\_id).
* Identify the top 10 movies based on average rating.
* Summarise the ratings table based on movie counts by median ratings.
* Identify the production house that has produced the most number of hit movies (average rating > 8).
* Determine the number of movies released in each genre during March 2017 in the USA with more than 1,000 votes.
* Retrieve movies of each genre starting with the word 'The' and having an average rating > 8.

Segment 5: Crew Analysis

* Identify the columns in the names table that have null values.
* Determine the top three directors in the top three genres with movies having an average rating > 8.
* Find the top two actors whose movies have a median rating >= 8.
* Identify the top three production houses based on the number of votes received by their movies.
* Rank actors based on their average ratings in Indian movies released in India.
* Identify the top five actresses in Hindi movies released in India based on their average ratings.

Segment 6: Broader Understanding of Data

* Classify thriller movies based on average ratings into different categories.
* analyse the genre-wise running total and moving average of the average movie duration.
* Identify the five highest-grossing movies of each year that belong to the top three genres.
* Determine the top two production houses that have produced the highest number of hits among multilingual movies.
* Identify the top three actresses based on the number of Super Hit movies (average rating > 8) in the drama genre.
* Retrieve details for the top nine directors based on the number of movies, including average inter-movie duration, ratings, and more.

Segment 7: Recommendations

* Based on the analysis, provide recommendations for the types of content Bolly movies should focus on producing.

-- Count rows in movie table

SELECT COUNT(\*) FROM movie;

-- Count rows in genre table

SELECT COUNT(\*) FROM genre;

-- Count rows in director\_mapping table

SELECT COUNT(\*) FROM director\_mapping;

-- Count rows in role\_mapping table

SELECT COUNT(\*) FROM role\_mapping;

-- Count rows in names table

SELECT COUNT(\*) FROM names;

-- Count rows in ratings table

SELECT COUNT(\*) FROM ratings;

-- Count null values in each column of the movie table

SELECT COUNT(\*) FROM movie WHERE title IS NULL;

SELECT COUNT(\*) FROM movie WHERE year IS NULL;

SELECT COUNT(\*) FROM movie WHERE date\_published IS NULL;

SELECT COUNT(\*) FROM movie WHERE duration IS NULL;

SELECT COUNT(\*) FROM movie WHERE country IS NULL;

SELECT COUNT(\*) FROM movie WHERE worldwide\_gross\_income IS NULL;

SELECT COUNT(\*) FROM movie WHERE languages IS NULL;

-- Segment 2: Movie Release Trends

SELECT YEAR(date\_published) AS release\_year, COUNT(\*) AS total\_movies

FROM movie

GROUP BY date\_published

ORDER BY date\_published;

select \* from movie;

-- Month-wise Trend of Movie Releases

SELECT MONTH(date\_published) AS release\_month, COUNT(\*) AS total\_movies

FROM movie

GROUP BY date\_published

ORDER BY date\_published;

-- Calculate the number of movies produced in the USA or India in the year 2019.

SELECT COUNT(\*) AS total\_movies

FROM movie

WHERE (country = 'USA' OR country = 'India') AND YEAR(date\_published) = 2019;

-- Segment 3: Production Statistics and Genre Analysis

-- Retrieve the unique list of genres present in the dataset.

SELECT \* from genre;

SELECT DISTINCT genre FROM genre;

-- Identify the genre with the highest number of movies produced overall.

SELECT genre, COUNT(\*) AS genre\_count

FROM genre

GROUP BY genre

ORDER BY genre\_count DESC

LIMIT 1;

-- Determine the count of movies that belong to only one genre.

-- Calculate the average duration of movies in each genre.

select \* from movie;

select \* FROM genre;

select genre,avg(duration) as Avg\_duration

from movie as A

join genre as B

on A.id = B.movie\_id

group by genre

order by genre asc;

-- Find the rank of the 'thriller' genre among all genres in terms of the number of movies produced.

SELECT genre, COUNT(\*) AS genre\_count, RANK() OVER (ORDER BY genre DESC) AS genre\_rank

FROM genre

GROUP BY genre

HAVING genre = 'Thriller';

-- Segment 4: Ratings Analysis and Crew Members

-- Retrieve the minimum and maximum values in each column of the ratings table (except movie\_id).

select \* from ratings;

SELECT

MIN(AVG\_RATING) AS Min\_value,

MIN(total\_votes) AS Min\_votes,

MIN(median\_rating) AS Min\_median,

MAX(AVG\_RATING) AS Max\_value,

MAX(total\_votes) As Max\_votes,

MAX(median\_rating)As Max\_median

FROM

ratings;

-- Identify the top 10 movies based on average rating.

select \* from movie;

select \* from ratings;

-- Query to identify the top 10 movies based on average rating

SELECT m.id, m.title, r.avg\_rating

FROM movie m

JOIN ratings r ON m.id = r.movie\_id

ORDER BY r.avg\_rating DESC

LIMIT 10;

-- Summarise the ratings table based on movie counts by median ratings.

SELECT median\_rating, COUNT(\*) as movie\_count

FROM ratings

GROUP BY median\_rating

ORDER BY median\_rating;

-- Identify the production house that has produced the most number of hit movies (average rating > 8).

select \* from ratings;

select \* from movie;

select production\_company,count(avg\_rating) from movie as A

join ratings as B

on A.id = B.movie\_id

where B.avg\_rating > 8

group by production\_company

order by production\_company asc;

-- Determine the number of movies released in each genre during March 2017 in the USA with more than 1,000 votes.

select \* from movie;

select \* from genre;

select \* from ratings;

SELECT g.genre, COUNT(m.id) AS num\_movies

FROM genre g

JOIN movie m ON g.movie\_id = m.id

JOIN ratings r ON m.id = r.movie\_id

WHERE m.date\_published >= '2017-03-01' AND m.date\_published <= '2017-03-31'

AND m.country = 'USA'

AND r.total\_votes > 1000

GROUP BY g.genre

ORDER BY num\_movies DESC;

-- Retrieve movies of each genre starting with the word 'The' and having an average rating > 8.

select \* from movie;

select \* from genre;

select \* from ratings;

select A.title, B.genre from movie as A

join genre as B

on A.id = B.movie\_id

join ratings as C

on A.id = C.movie\_id

where A.title like "The%"

and C.avg\_rating >8

order by genre asc;

-- Segment 5: Crew Analysis

-- Identify the columns in the names table that have null values.

select \* from names;

select count(id) from names

where id = null;

select count(name) from names

where name = null;

select count(height) from names

where height = null;

select count(date\_of\_birth) from names

where date\_of\_birth = null;

select count(known\_for\_movies) from names

where known\_for\_movies = null;

-- Determine the top three directors in the top three genres with movies having an average rating > 8.

select \* from director\_mapping;

select \* from genre;

select \* from names;

select \* from ratings;

select \* from movie;

select D.name,B.genre, A.title from movie as A

join genre as B

on A.id = B.movie\_id

join director\_mapping as C

on c.movie\_id = A.id

join names as D

on D.id = c.name\_id

join ratings as E

on A.id = E.movie\_id

where E.avg\_rating >8

limit 3;

-- Determine the top three directors in the top three genres with movies having an average rating > 8.