PRSQL-01: IMDB Movies – SQL Project Report

This project focuses on analyzing the IMDB Movies dataset using SQL. The objective is to practice and demonstrate SQL querying skills including filtering, aggregation, joins, and sorting using a real-world dataset.

1. Dataset Overview

The dataset consists of two tables: movies and directors. The movies table contains movie details such as title, popularity, revenue, vote average, release date, and director_id. The directors table contains director information. Both tables are related through the director_id field.

2. Objective of the Project

The objective of this project is to analyze the IMDB movie database using SQL queries to extract meaningful insights about movies and directors.
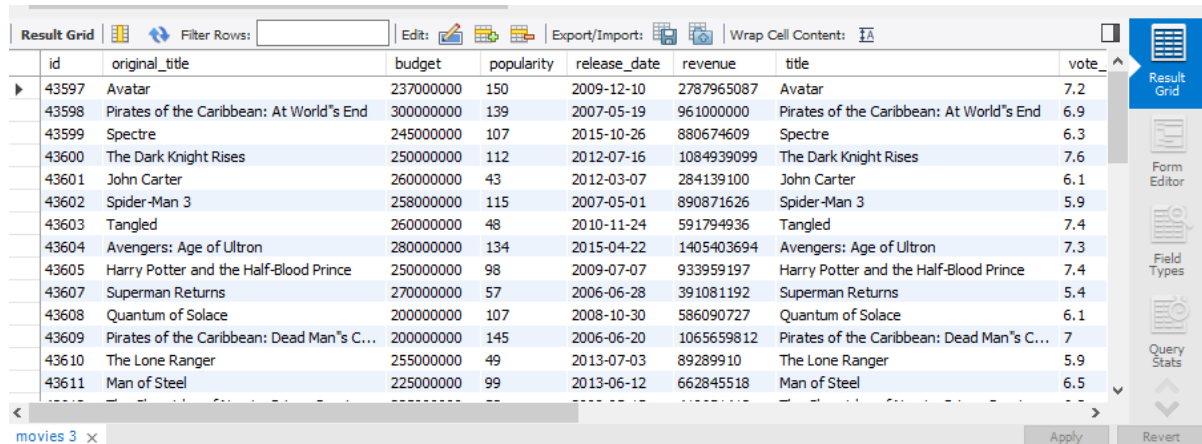
3. SQL Environment

All queries were executed using the DataMites SQL portal where the IMDB database was preloaded.

4. Problem Queries

a) Can you get all data about movies?

SQL Query:
SELECT * FROM movies;

id	original_title	budget	popularity	release_date	revenue	title	vote_
43597	Avatar	237000000	150	2009-12-10	2787965087	Avatar	7.2
43598	Pirates of the Caribbean: At World"s End	300000000	139	2007-05-19	961000000	Pirates of the Caribbean: At World"s End	6.9
43599	Spectre	245000000	107	2015-10-26	880674609	Spectre	6.3
43600	The Dark Knight Rises	250000000	112	2012-07-16	1084939099	The Dark Knight Rises	7.6
43601	John Carter	260000000	43	2012-03-07	284139100	John Carter	6.1
43602	Spider-Man 3	258000000	115	2007-05-01	890871626	Spider-Man 3	5.9
43603	Tangled	260000000	48	2010-11-24	591794936	Tangled	7.4
43604	Avengers: Age of Ultron	280000000	134	2015-04-22	1405403694	Avengers: Age of Ultron	7.3
43605	Harry Potter and the Half-Blood Prince	250000000	98	2009-07-07	933959197	Harry Potter and the Half-Blood Prince	7.4
43607	Superman Returns	270000000	57	2006-06-28	391081192	Superman Returns	5.4
43608	Quantum of Solace	200000000	107	2008-10-30	586090727	Quantum of Solace	6.1
43609	Pirates of the Caribbean: Dead Man"s C...	200000000	145	2006-06-20	1065659812	Pirates of the Caribbean: Dead Man"s C...	7
43610	The Lone Ranger	255000000	49	2013-07-03	89289910	The Lone Ranger	5.9
43611	Man of Steel	225000000	99	2013-06-12	662845518	Man of Steel	6.5

Explanation:
This query retrieves all movie records.

b) How do you get all data about directors?

SQL Query:
SELECT * FROM directors;

name	id	gender	uid	department
James Cameron	4762	2	2710	Directing
Gore Verbinski	4763	2	1704	Directing
Sam Mendes	4764	2	39	Directing
Christopher Nolan	4765	2	525	Directing
Andrew Stanton	4766	2	7	Directing
Sam Raimi	4767	2	7623	Directing
Byron Howard	4768	2	76595	Directing
Joss Whedon	4769	2	12891	Directing
David Yates	4770	2	11343	Directing
Zack Snyder	4771	2	15217	Directing
Bryan Singer	4772	2	9032	Directing
Marc Forster	4773	2	12995	Directing
Andrew Adamson	4774	2	5524	Directing
Rob Marshall	4775	2	17633	Directing
Barry Sonnenfeld	4776	2	5174	Directing

directors 2 ×

Explanation:
This query retrieves all director records.

C)Check how many movies are present in IMDB.

SQL Query:
SELECT COUNT(*) AS total_movies FROM movies;

total_movies
47

Explanation:
Counts total movies.

d)Find these 3 directors: James Cameron ; Luc Besson ; John Woo

SQL Query:
SELECT * FROM directors WHERE name IN ('James Cameron','Luc Besson','John Woo');

name	id	gender	uid	department
James Cameron	4762	2	2710	Directing
John Woo	4893	2	11401	Directing
Luc Besson	4949	2	59	Directing
NULL	NULL	NULL	NULL	NULL

Explanation:
Filters directors by specific names.

e)Find all directors with name starting with S.

SQL Query:
SELECT * FROM directors WHERE name LIKE 'S%';

name	id	gender	uid	department
Sam Mendes	4764	2	39	Directing
Sam Raimi	4767	2	7623	Directing
Shane Black	4784	2	1108	Directing
Steven Spielberg	4799	2	488	Directing
Stephen Sommers	4815	2	7775	Directing
Shawn Levy	4842	2	17825	Directing
Steve Hickner	4852	2	44113	Directing
Simon Wells	4855	2	21879	Directing
Steven Soderbergh	4909	2	1884	Directing
Simon West	4930	2	12786	Directing
Stefen Fangmeier	4931	0	25453	Directing
Spike Jonze	4932	2	5953	Directing
Steve Martino	4943	2	71729	Directing
Sergei Bodrov	4952	0	130938	Directing
Sydney Pollack	4965	2	2226	Directing
Sylvester Stallone	4992	2	16483	Directing
Seth Gordon	4997	2	71600	Directing
Scott Derrickson	5004	2	55499	Directing
Stephen Hopkins	5008	2	2042	Directing

Explanation:
Finds directors whose names start with S.

f)Count female directors.

SQL Query:
SELECT COUNT(*) FROM directors WHERE gender = 1;

female_directors
150

Explanation:
Counts female directors.

g)Find the name of the 10th first women directors?

SQL Query:
SELECT name FROM directors WHERE gender = 1 ORDER BY id LIMIT 1 OFFSET 9;

name
▶ Angelina Jolie

Explanation:
Fetches the 10th female director.

h)What are the 3 most popular movies?

SQL Query:
SELECT title, popularity FROM movies ORDER BY popularity DESC LIMIT 3;

title	popularity
▶ Jurassic World	418
Captain America: Civil War	198
Avatar	150

Explanation:
Displays top 3 popular movies.

i)What are the 3 most bankable movies?

SQL Query:
SELECT title, revenue FROM movies ORDER BY revenue DESC LIMIT 3;

title	revenue
▶ Avatar	2787965087
Titanic	1845034188
The Avengers	1519557910

Explanation:
Displays top revenue-generating movies.

j)What is the most awarded average vote since the January 1st, 2000?

SQL Query:
SELECT title, vote_average FROM movies WHERE release_date >= '2000-01-01' ORDER BY vote_average DESC LIMIT 1;

title	vote_average
The Dark Knight Rises	7.6

Explanation:
Finds highest rated movie released after 2000.

k)Which movie(s) were directed by Brenda Chapman?

SQL Query:
SELECT m.title FROM movies m JOIN directors d ON m.director_id = d.id WHERE d.name = 'Brenda Chapman';

title

Explanation:
No records returned, indicating no movies directed by Brenda Chapman in dataset.

l)Which director made the most movies?

SQL Query:
SELECT d.name, COUNT(m.id) AS total_movies FROM directors d JOIN movies m ON d.id = m.director_id GROUP BY d.name ORDER BY total_movies DESC LIMIT 1;

name	total_movies
Gore Verbinski	3

Explanation:
Finds director with most movies.

m)Which director is the most bankable?

SQL Query:
SELECT d.name, SUM(m.revenue) AS total_revenue FROM directors d JOIN movies m ON d.id = m.director_id GROUP BY d.name ORDER BY total_revenue DESC LIMIT 1;

name	total_revenue
James Cameron	4632999275

Explanation:
Finds director with highest total revenue.

5. Key Insights

- Certain directors dominate in terms of movie count and revenue
- Popularity does not always correlate directly with revenue
- Female director representation is comparatively lower
- SQL joins are essential for meaningful relational analysis

6. Conclusion

This project successfully demonstrates the application of SQL in analyzing a real-world IMDB dataset.

By using filtering, aggregation, joins, and sorting techniques, actionable insights related to movies and directors were derived.

The project strengthens practical SQL skills and builds confidence in handling relational databases for analytics tasks.