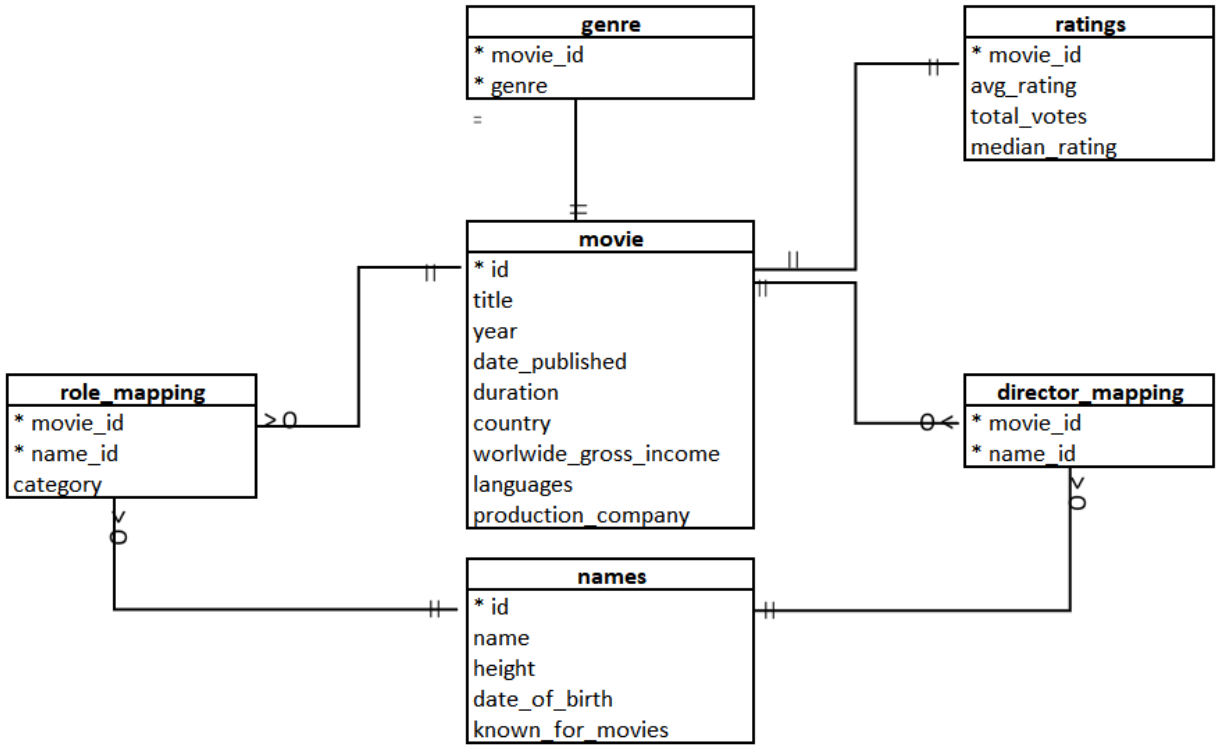
Data Analysis of RSVP Movies

Project Overview:

In this SQL-based project, I explored and analyzed a comprehensive movie dataset to draw insights for RSVP Movies, a production company. By leveraging SQL queries, I uncovered patterns related to movie production, genre trends, ratings, and more, providing data-driven insights to guide future strategic decisions in production, casting, and partnerships.

Entity-Relationship Diagram (ERD): This ERD represents the structure and relationships between key entities in the movie database.



Key Objectives:

1. Understand the distribution of movies by year, month, country, and genre.
2. Identify top-performing genres, directors, actors, and actresses based on ratings and votes.
3. Analyze production houses and recommend partnerships based on historical success.
4. Explore trends in multilingual movies and classify movies by performance (e.g., super hit, hit, flop).
5. Provide detailed insights for key strategic decisions related to production planning, talent hiring, and genre selection.

Key Queries and Insights

1. Schema Exploration & Row Count

```
select 'director_mapping' as table_name , count(*) as row_count from director_mapping
Union all
select 'genre' as table_name , count(*) as row_count from genre
union all
select 'movie' as table_name , count(*) as row_count from movie
Union all
select 'names' as table_name , count(*) as row_count from names
union all
select 'ratings' as table_name , count(*) as row_count from ratings
Union all
select 'role_mapping' as table_name , count(*) as row_count from role_mapping
order by row_count desc;
```

Insight: Understanding the dataset's size gives a broad sense of the data to be analyzed. The largest table, names, contained over 25,000 records, essential for exploring relationships between actors, directors, and movies.

2. Handling Missing Data in the movie Table

```
select count(*) as total_rows,
  sum(case when id is null then 1 else 0 end) as id_nulls,
  sum(case when title is null then 1 else 0 end) as title_nulls,
  sum(case when year is null then 1 else 0 end) as year_nulls,
  sum(case when date_published is null then 1 else 0 end) as date_published_nulls,
  sum(case when duration is null then 1 else 0 end) as duration_nulls,
  sum(case when country is null then 1 else 0 end) as country_nulls,
  sum(case when worlwide_gross_income is null then 1 else 0 end) as worldwide_gross_inc
ome_nulls,
  sum(case when languages is null then 1 else 0 end) as languages_nulls,
  sum(case when production_company is null then 1 else 0 end) as production_company_n
ulls
from movie;
```

Insight: Several columns, including worldwide_gross_income and production_company, had significant null values. Addressing these is crucial for accurate analysis and reporting.

3. Trends in Movie Releases

```
select year, count(id)
from movie
group by year;
```

```
select month(date_published) as month, count(id)
from movie
group by month
order by month;
```

Insight: The analysis highlighted production peaks by month, with March being the most active month for releases, providing a valuable guide for planning release schedules.

4. Genre Distribution & Top Genres

- **Top Genre Identification**: By exploring the genre distribution, the project found that the **Drama** genre had the highest number of movies produced.

```
select genre,count(movie_id) as genre_count
from genre
group by genre
order by genre_count desc
limit 1;
```

Insight: Drama emerged as the most popular genre, while 3,289 movies belonged to only one genre, giving insight into how focused vs. multi-genre

films perform.

- **Movies with Multiple Genres**: The project also analyzed how many movies belonged to only one genre versus those that spanned multiple genres.

```
select count(*)
from(select movie_id, count(movie_id) as movie_genre_count
 from genre
 group by movie_id
 having movie_genre_count = 1) as mv;
```

Insight: Over 3,000 movies had only one genre, indicating a trend toward more genre-specific storytelling.

5. Top Performers by Ratings

- **Top 10 movies based on average rating**

```
select title, avg_rating, rank() over(order by avg_rating desc) as movie_rank
from movie m
inner join ratings r on r.movie_id = m.id
limit 10;
```

- **Top Three Directors in the Top Three Genres (Average Rating > 8)**

```
WITH top_genres AS (
   SELECT genre, COUNT(gn.movie_id) AS movie_count
   FROM genre gn
   INNER JOIN ratings rt ON rt.movie_id = gn.movie_id
   WHERE rt.avg_rating > 8
   GROUP BY genre
   ORDER BY COUNT(gn.movie_id) DESC
)
SELECT nm.name, COUNT(m.id) AS movie_count
FROM movie m
INNER JOIN ratings r ON r.movie_id = m.id
INNER JOIN genre g ON g.movie_id = m.id
INNER JOIN role_mapping rm ON rm.movie_id = m.id
INNER JOIN names nm ON nm.id = rm.name_id
```

```
WHERE g.genre IN (SELECT genre FROM top_genres)
AND r.avg_rating > 8
GROUP BY nm.name
ORDER BY movie_count DESC
LIMIT 3;
```

Insight: Actors who consistently perform in highly rated movies bring credibility and strong box office appeal to future projects. By focusing on actors whose movies have a median rating of 8 or more, we can shortlist those with proven talent.

6.Analyzing Production House Success

- **Top Three Production Houses Based on Votes**

```
SELECT production_company, SUM(r.total_votes) AS vote_count,
RANK() OVER (ORDER BY SUM(r.total_votes) DESC) AS prod_comp_rank
FROM movie m
INNER JOIN ratings r ON r.movie_id = m.id
GROUP BY production_company
ORDER BY vote_count DESC;
```

- **Top Two Production Houses for Multilingual Movies (Median Rating >= 8)**

```
SELECT production_company, COUNT(m.id) AS movie_count,
RANK() OVER (ORDER BY COUNT(m.id) DESC) AS prod_comp_rank
FROM movie m
INNER JOIN ratings r ON r.movie_id = m.id
WHERE r.median_rating >= 8
AND m.languages LIKE '%,%'
GROUP BY production_company
ORDER BY movie_count DESC
LIMIT 2;
```

Insight: Production houses with a track record of producing movies that receive high engagement (measured by total votes) are important candidates for future collaborations. This analysis reveals the top three production houses based on audience engagement.

7.Movie Performance Analysis

- **Classifying Thriller Movies by Performance**

```
SELECT m.title, r.avg_rating,
CASE
  WHEN r.avg_rating > 8 THEN 'Superhit movies'
  WHEN r.avg_rating BETWEEN 7 AND 8 THEN 'Hit movies'
  WHEN r.avg_rating BETWEEN 5 AND 7 THEN 'One-time-watch movies'
  ELSE 'Flop movies'
END AS movie_type
```

```
FROM movie m
INNER JOIN ratings r ON r.movie_id = m.id
INNER JOIN genre g ON g.movie_id = m.id
WHERE g.genre = 'Thriller'
ORDER BY r.avg_rating DESC;
```

Insight: This analysis helps classify thriller movies based on their average ratings, categorizing them into "Superhit," "Hit," "One-time-watch," or "Flop." Understanding which thrillers performed well provides insights for potential future projects in this genre.

- **Highest-Grossing Movies by Genre**

```
WITH top_genres AS (
  SELECT genre, COUNT(m.id) AS genre_count
  FROM genre g
  INNER JOIN movie m ON m.id = g.movie_id
  GROUP BY genre
  ORDER BY genre_count DESC
  LIMIT 3
),
ranked_movies AS (
  SELECT g.genre, m.year, m.title, m.worlwide_gross_income,
  RANK() OVER (PARTITION BY m.year, g.genre ORDER BY m.worlwide_gross_income DESC
) AS movie_rank
  FROM movie m
```

```
    INNER JOIN genre g ON g.movie_id = m.id
    WHERE g.genre IN (SELECT genre FROM top_genres)
)
SELECT genre, year, title, worlwide_gross_income
FROM ranked_movies
WHERE movie_rank <= 5
ORDER BY year, genre, worlwide_gross_income DESC;
```

Insight: Identifying the top five highest-grossing movies from the top three genres each year allows RSVP Movies to understand which movies and genres are most successful financially. This is crucial for strategic production planning and investment decisions.

Technologies Used

- **SQL**: Extracted, transformed, and analyzed data from a relational database.
- **Entity-Relationship Diagram (ERD)**: Mapped out relationships between entities like movies, genres, actors, and directors for better understanding and analysis.
- **Window Functions**: Used to rank performers, calculate running totals, and compute moving averages for comprehensive analysis.