Kaggle Disney+ Dataset

Dataset Selection and Structure

The dataset chosen for analysis is the "Disney Movies and TV Shows" collection from Kaggle. It comprises detailed information on titles available on the Disney+ streaming platform, including attributes like type, director, cast, country of origin, date added, release year, rating, and genre.

To optimize the dataset for SQL queries and gain deeper insights, we decomposed the data into several relational tables:

1. Titles: Contains core information about each show, such as the ID, title, type, release year, and date added to Disney+.

2. Directors: Lists directors with a unique ID assigned to each.

3. Casts: Lists cast members with a unique ID for each person.

4. Countries: Enumerates the countries where the titles were produced.

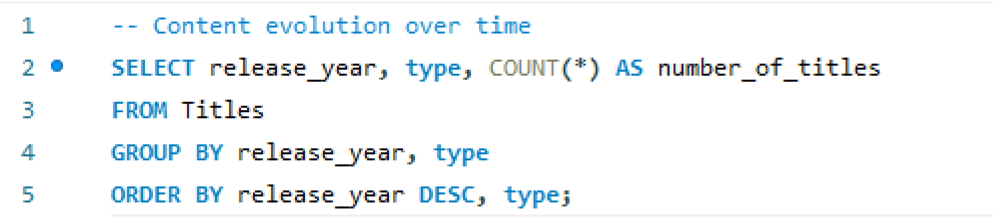
5. Genres: Catalogs the genres of the titles.

6. TitleDirector, TitleCast, TitleCountry, TitleGenre: These junction tables link titles to their respective directors, cast members, countries, and genres, facilitating many-to-many relationships.

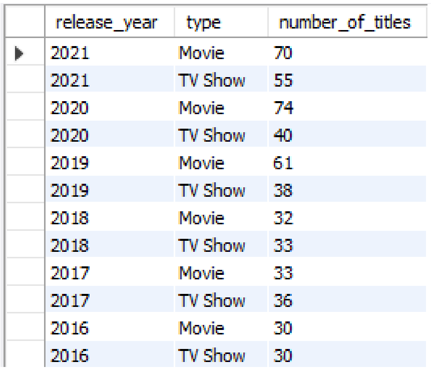
SQL Queries and Findings

Several SQL queries were crafted to extract meaningful insights:

- Content evolution over time, revealing trends in the platform's offerings.

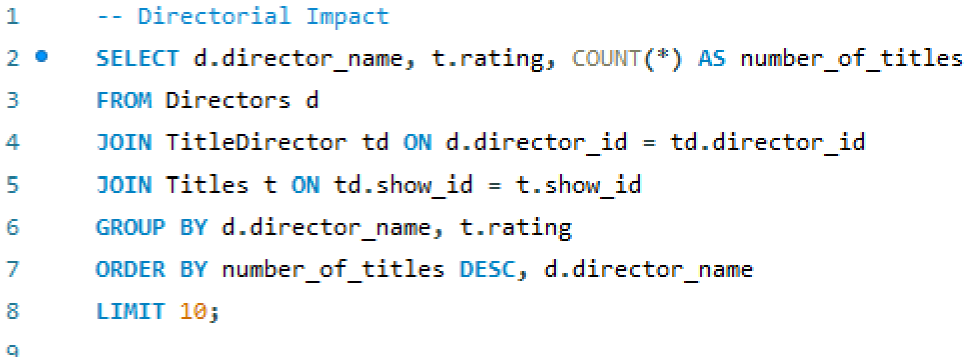


```sql
-- Content evolution over time
SELECT release_year, type, COUNT(*) AS number_of_titles
FROM Titles
GROUP BY release_year, type
ORDER BY release_year DESC, type;
```

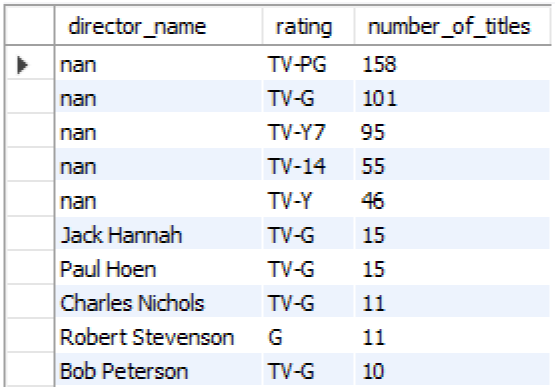


release_year	type	number_of_titles
2021	Movie	70
2021	TV Show	55
2020	Movie	74
2020	TV Show	40
2019	Movie	61
2019	TV Show	38
2018	Movie	32
2018	TV Show	33
2017	Movie	33
2017	TV Show	36
2016	Movie	30
2016	TV Show	30

- Directorial impact, showing which directors' works are most prominent.

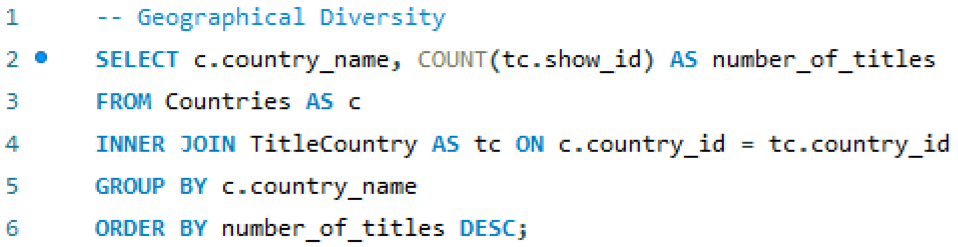


```sql
-- Directorial Impact
SELECT d.director_name, t.rating, COUNT(*) AS number_of_titles
FROM Directors d
JOIN TitleDirector td ON d.director_id = td.director_id
JOIN Titles t ON td.show_id = t.show_id
GROUP BY d.director_name, t.rating
ORDER BY number_of_titles DESC, d.director_name
LIMIT 10;
```

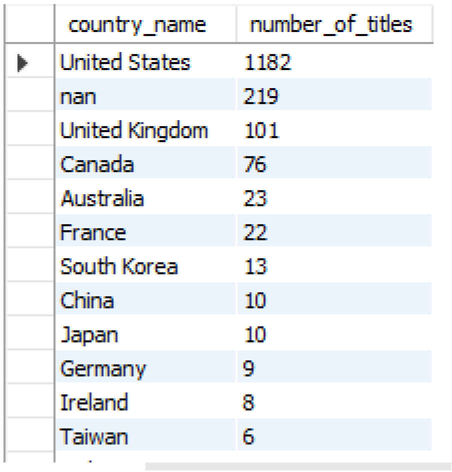


director_name	rating	number_of_titles
nan	TV-PG	158
nan	TV-G	101
nan	TV-Y7	95
nan	TV-14	55
nan	TV-Y	46
Jack Hannah	TV-G	15
Paul Hoen	TV-G	15
Charles Nichols	TV-G	11
Robert Stevenson	G	11
Bob Peterson	TV-G	10

- Geographical diversity, indicating the global footprint of the content.

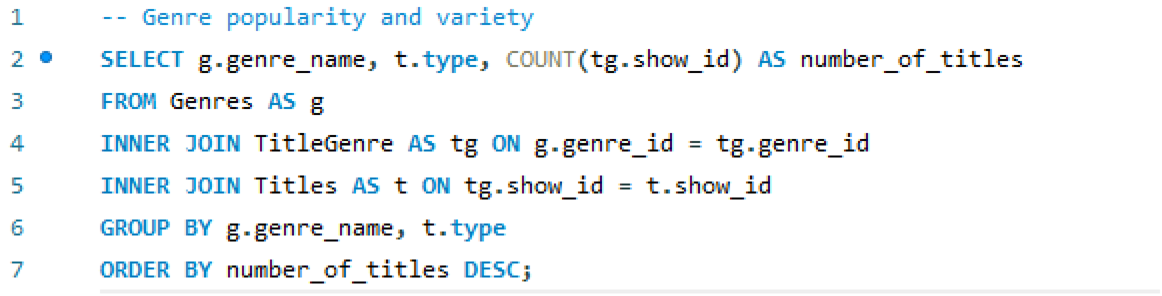


```sql
-- Geographical Diversity
SELECT c.country_name, COUNT(tc.show_id) AS number_of_titles
FROM Countries AS c
INNER JOIN TitleCountry AS tc ON c.country_id = tc.country_id
GROUP BY c.country_name
ORDER BY number_of_titles DESC;
```

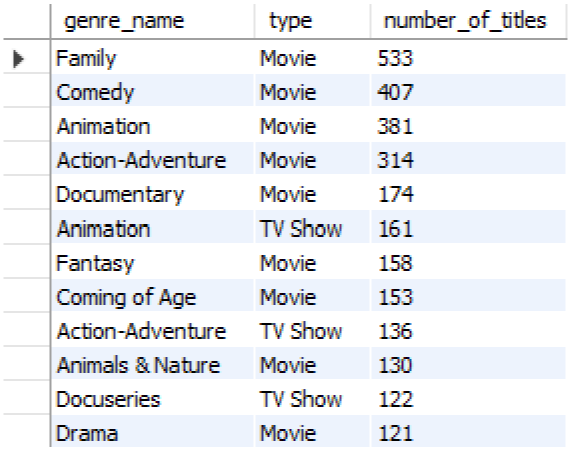


country_name	number_of_titles
United States	1182
nan	219
United Kingdom	101
Canada	76
Australia	23
France	22
South Korea	13
China	10
Japan	10
Germany	9
Ireland	8
Taiwan	6

- Genre popularity and variety, highlighting the most common genres.

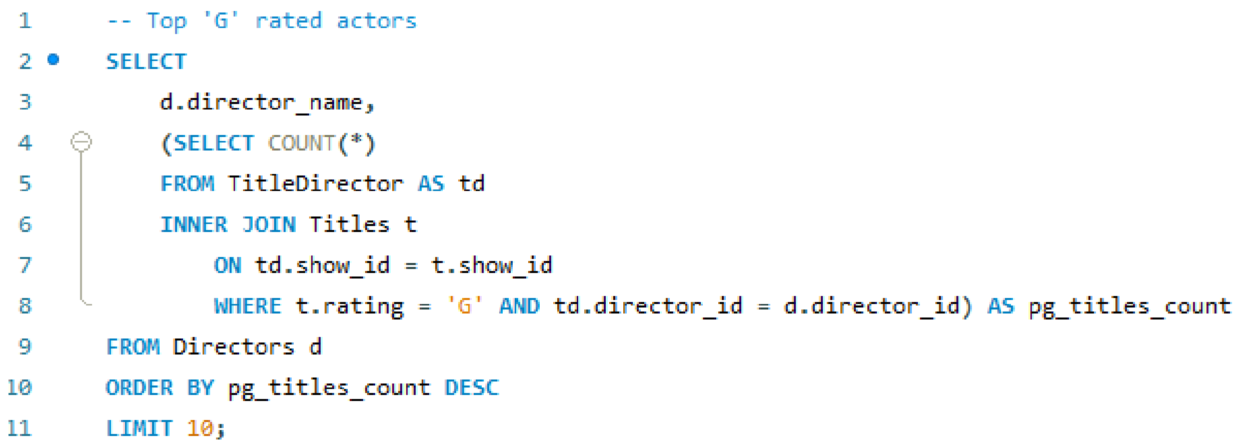


```sql
-- Genre popularity and variety
SELECT g.genre_name, t.type, COUNT(tg.show_id) AS number_of_titles
FROM Genres AS g
INNER JOIN TitleGenre AS tg ON g.genre_id = tg.genre_id
INNER JOIN Titles AS t ON tg.show_id = t.show_id
GROUP BY g.genre_name, t.type
ORDER BY number_of_titles DESC;
```

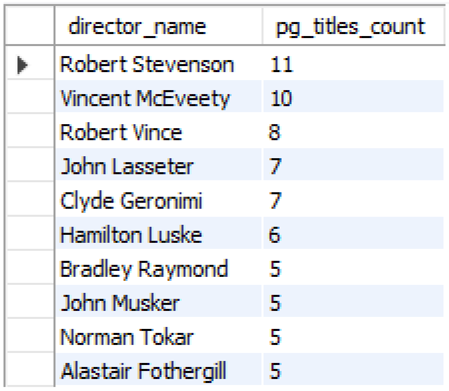


genre_name	type	number_of_titles
Family	Movie	533
Comedy	Movie	407
Animation	Movie	381
Action-Adventure	Movie	314
Documentary	Movie	174
Animation	TV Show	161
Fantasy	Movie	158
Coming of Age	Movie	153
Action-Adventure	TV Show	136
Animals & Nature	Movie	130
Docuseries	TV Show	122
Drama	Movie	121

- Top 10 directors who have the highest number of titles with a specific rating, say 'PG'.



```sql
-- Top 'G' rated actors
SELECT
    d.director_name,
    (SELECT COUNT(*)
    FROM TitleDirector AS td
    INNER JOIN Titles t
        ON td.show_id = t.show_id
        WHERE t.rating = 'G' AND td.director_id = d.director_id) AS pg_titles_count
FROM Directors d
ORDER BY pg_titles_count DESC
LIMIT 10;
```



	director_name	pg_titles_count
▶	Robert Stevenson	11
	Vincent McEveety	10
	Robert Vince	8
	John Lasseter	7
	Clyde Geronimi	7
	Hamilton Luske	6
	Bradley Raymond	5
	John Musker	5
	Norman Tokar	5
	Alastair Fothergill	5

After running the queries, we learned, for instance, that the content on Disney+ includes a diverse range of genres and has expanded over the years to include a significant number of countries, indicating Disney's global content strategy.

Learnings from the Activity

Through this exercise, we were able to practice data normalization, which is crucial for efficient SQL querying. We also observed the importance of understanding the data types within a dataset, such as the categorical nature of content ratings, which required us to adjust our SQL queries accordingly.

Concluding Thoughts

The Disney+ dataset offered a rich opportunity to apply SQL for real-world data analysis, highlighting the platform's content strategy and diversity. The SQL analysis demonstrated the power of relational databases in dissecting and examining large datasets to uncover trends and patterns. This activity has reinforced the practical applications of data analytics in the media and entertainment industry, showcasing how data-driven insights can guide content creation and business strategies. The ability to transform raw data into actionable knowledge is a valuable skill in the era of big data, and the Disney+ dataset served as an excellent canvas for this endeavor.

Week 10 Assignment: SQL Query Assignment #2

Instructions: Perform Data Analysis using MySQL.

1. Retrieve a dataset of your choice from Data.gov or another opensource data repository of your choice. Provide a comment in your assignment submission to document where the dataset came from. You will NOT be submitting the actual dataset when you submit your project source code.
2. Load the dataset into MySQL using the MySQL Workbench.
3. Break out the data and create a new table with a subset of the data from the original table.
4. Use SQL commands to extract some of the columns (at least three) from the table you created. Specifically, you need to write queries that include the following SQL features:
    - One query that performs a select on multiple rows and columns.One query that contains an aggregate function.
    - One query that contains sorting and grouping.
    - One query that contains a subquery.
    - One query to join two tables together.

Create a 1-2-page summary (using MS Word) that includes the following information:

- Which dataset did you select?
- How did you break out the data into multiple tables?
- What queries did you run?
- What information did you find about the data after performing your analysis?
- What did you learn from this activity?
- What are your concluding thoughts about the dataset you selected and the SQL analysis you were able to perform?

This assignment is worth 10 points. Submit your summary document by Sunday at 11:59 PM ET. Do not submit the dataset.