NETFLIX MOVIES AND TV SHOWS DATA ANALYSIS USING SQL

ABOUT

PROJECT OVERVIEW

This project provides a comprehensive analysis of Netflix's movie and TV show data using SQL. The primary goal is to extract valuable insights regarding content types, ratings, geographical distribution, and various other aspects of the Netflix content library. The analysis is structured to address key business questions and deliver meaningful insights based on the dataset.

OBJECTIVES

1. Analyze the distribution of content types (movies vs TV shows).
2. Identify the most common ratings for movies and TV shows.
3. Analyze content based on release years, countries, and durations.
4. Explore and categorize content using specific criteria and keywords.

DATASET

```sql
-- Netflix Project

CREATE TABLE netflix
(
    show_id      VARCHAR(5),
    type         VARCHAR(10),
    title        VARCHAR(250),
    director     VARCHAR(550),
    casts        VARCHAR(1050),
    country      VARCHAR(550),
    date_added   VARCHAR(55),
    release_year INT,
    rating       VARCHAR(15),
    duration     VARCHAR(15),
    listed_in    VARCHAR(250),
    description  VARCHAR(550)
);
SELECT * FROM netflix;
SELECT COUNT(*) FROM netflix;

```

BUSINESS PROBLEMS AND SQL QUERIES

01. COUNT THE NUMBER OF MOVIES VS TV SHOWS

```sql
-- 01.Count the Number of Movies vs TV Shows
SELECT type, COUNT(*) as total_content FROM netflix GROUP BY 1;
```

PURPOSE:

- THIS QUERY COUNTS THE TOTAL NUMBER OF CONTENT ITEMS FOR EACH TYPE (MOVIES VS TV SHOWS) ON NETFLIX.
- TO DETERMINE THE DISTRIBUTION OF CONTENT TYPES (MOVIES VS. TV SHOWS) IN THE NETFLIX DATASET.

REASONING:

THIS HELPS IN UNDERSTANDING WHICH TYPE OF CONTENT DOMINATES NETFLIX'S LIBRARY AND AIDS IN STRATEGIC DECISIONS ON CONTENT INVESTMENTS.

02. FIND THE MOST COMMON RATING FOR MOVIES AND TV SHOWS

```sql
-- 02.Find the Most Common Rating for Movies and TV Shows
WITH RatingCounts AS (
    SELECT
        type,
        rating,
        COUNT(*) AS rating_count
    FROM netflix
    GROUP BY type, rating
),
RankedRatings AS (
    SELECT
        type,
        rating,
        rating_count,
        RANK() OVER (PARTITION BY type ORDER BY rating_count DESC) AS rank
    FROM RatingCounts
)
SELECT
    type,
    rating AS most_frequent_rating
FROM RankedRatings
WHERE rank = 1;
```

PURPOSE:

- THIS QUERY IDENTIFIES THE MOST FREQUENTLY OCCURRING RATING FOR EACH TYPE OF CONTENT (MOVIES AND TV SHOWS).
- TO IDENTIFY THE MOST FREQUENTLY OCCURRING RATING FOR MOVIES AND TV SHOWS SEPARATELY.

REASONING:

UNDERSTANDING COMMON RATINGS CAN HELP GAUGE THE CONTENT'S TARGET AUDIENCE AND INFORM DECISIONS ABOUT CONTENT AGE RESTRICTIONS AND CLASSIFICATIONS.

03. FIND THE MOST COMMON RATING FOR MOVIES AND TV SHOWS

```sql
-- 03.List All Movies Released in a Specific Year (e.g., 2020)
SELECT *
FROM netflix
WHERE release_year = 2020;
```

PURPOSE:

- TO LIST ALL MOVIES RELEASED IN 2020.
- RETRIEVE ALL MOVIES THAT WERE RELEASED SPECIFICALLY IN THE YEAR 2020.

REASONING:

USEFUL FOR TREND ANALYSIS, HELPING UNDERSTAND WHAT KIND OF CONTENT WAS RELEASED IN A SPECIFIC YEAR.

04.FIND THE TOP 5 COUNTRIES WITH THE MOST CONTENT ON NETFLIX

```sql
-- 04.Find the Top 5 Countries with the Most Content on Netflix
SELECT *
FROM
(
    SELECT
        UNNEST(STRING_TO_ARRAY(country, ',')) AS country,
        COUNT(*) AS total_content
    FROM netflix
    GROUP BY 1
) AS t1
WHERE country IS NOT NULL
ORDER BY total_content DESC
LIMIT 5;
```

PURPOSE:

TO IDENTIFY THE TOP 5 COUNTRIES THAT HAVE CONTRIBUTED THE MOST CONTENT TO NETFLIX.

REASONING:

UNDERSTANDING GEOGRAPHICAL CONTENT CONTRIBUTIONS HELPS IN ANALYZING REGIONAL CONTENT STRATEGIES AND THE DIVERSITY OF THE NETFLIX LIBRARY.

05. IDENTIFY THE LONGEST MOVIE

```
66
67  -- 05.Identify the Longest Movie
68  SELECT
69      *
70  FROM netflix
71  WHERE type = 'Movie'
72  ORDER BY SPLIT_PART(duration, ' ', 1)::INT DESC;
73
```

PURPOSE:
FIND THE LONGEST MOVIE BASED ON ITS DURATION.

REASONING:
IDENTIFYING THE LONGEST MOVIE PROVIDES INSIGHTS INTO NETFLIX'S INVESTMENT IN LENGTHY CONTENT, WHICH MIGHT CATER TO SPECIFIC AUDIENCE PREFERENCES.

06. FIND CONTENT ADDED IN THE LAST 5 YEARS

```
74
75  -- 06.Find Content Added in the Last 5 Years
76  SELECT *
77  FROM netflix
78  WHERE TO_DATE(date_added, 'Month DD, YYYY') >= CURRENT_DATE - INTERVAL '5 years';
79
80
```

PURPOSE:

TO RETRIEVE CONTENT THAT HAS BEEN ADDED TO NETFLIX IN THE LAST FIVE YEARS..

REASONING:

IANALYZING RECENTLY ADDED CONTENT HELPS IN UNDERSTANDING NETFLIX'S RECENT CONTENT ACQUISITION TRENDS AND STRATEGIES.

07. FIND ALL MOVIES/TV SHOWS BY DIRECTOR 'RAJIV CHILAKA'

```
80
81  -- 07.Find All Movies/TV Shows by Director 'Rajiv Chilaka'
82  SELECT *
83  FROM (
84      SELECT
85          *,
86          UNNEST(STRING_TO_ARRAY(director, ',')) AS director_name
87      FROM netflix
88  ) AS t
89  WHERE director_name = 'Rajiv Chilaka';
90
```

PURPOSE:

TO LIST ALL MOVIES AND TV SHOWS DIRECTED BY RAJIV CHILAKA.

REASONING:

THIS QUERY SPLITS THE DIRECTOR FIELD INTO INDIVIDUAL NAMES IN CASE THERE ARE MULTIPLE DIRECTORS, AND THEN IT FILTERS TO FIND CONTENT DIRECTED BY 'RAJIV CHILAKA'. USING UNNEST() ENSURES THAT CONTENT WITH MULTIPLE DIRECTORS IS ALSO CORRECTLY IDENTIFIED.

08.LIST ALL TV SHOWS WITH MORE THAN 5 SEASONS

```
91
92  -- 08.List All TV Shows with More Than 5 Seasons
93  SELECT *
94  FROM netflix
95  WHERE type = 'TV Show'
96    AND SPLIT_PART(duration, ' ', 1)::INT > 5;
97
```

PURPOSE:

TO RETRIEVE ALL TV SHOWS WITH MORE THAN FIVE SEASONS.

REASONING:

THE QUERY EXTRACTS THE NUMERIC PART FROM THE DURATION FIELD (E.G., '6 SEASONS'), CONVERTS IT TO AN INTEGER, AND THEN CHECKS WHETHER IT'S GREATER THAN 5, ENSURING THE RETRIEVAL OF TV SHOWS WITH MORE THAN 5 SEASONS.

09.COUNT THE NUMBER OF CONTENT ITEMS IN EACH GENRE

```sql
-- 09.Count the Number of Content Items in Each Genre
SELECT
    UNNEST(STRING_TO_ARRAY(listed_in, ',')) AS genre,
    COUNT(*) AS total_content
FROM netflix
GROUP BY 1;
```

PURPOSE:

TO COUNT HOW MANY CONTENT ITEMS BELONG TO EACH GENRE.

REASONING:

THE LISTED_IN FIELD MAY CONTAIN MULTIPLE GENRES (E.G., 'ACTION, DRAMA'). THE QUERY SPLITS THESE VALUES INTO INDIVIDUAL GENRES USING UNNEST() AND COUNTS THE TOTAL NUMBER OF CONTENT ITEMS FOR EACH GENRE.

10.FIND EACH YEAR AND THE AVERAGE NUMBERS OF CONTENT RELEASE IN INDIA ON NETFLIX

```sql

-- 10.Find each year and the average numbers of content release in India on netflix
SELECT
    country,
    release_year,
    COUNT(show_id) AS total_release,
    ROUND(
        COUNT(show_id)::numeric /
        (SELECT COUNT(show_id) FROM netflix WHERE country = 'India')::numeric * 100, 2
    ) AS avg_release
FROM netflix
WHERE country = 'India'
GROUP BY country, release_year
ORDER BY avg_release DESC
LIMIT 5;
```

PURPOSE:

TO IDENTIFY THE TOP FIVE YEARS WITH THE MOST CONTENT RELEASES IN INDIA.

REASONING:

THE QUERY FILTERS CONTENT PRODUCED IN INDIA, GROUPS IT BY THE RELEASE YEAR, AND COUNTS THE NUMBER OF RELEASES PER YEAR. THE TOP 5 YEARS WITH THE HIGHEST RELEASE COUNT ARE RETURNED.

11.LIST ALL MOVIES THAT ARE DOCUMENTARIES

```
122
123  -- 11.List All Movies that are Documentaries
124  SELECT *
125  FROM netflix
126  WHERE listed_in LIKE '%Documentaries';
127
```

PURPOSE:
TO LIST ALL MOVIES CLASSIFIED AS DOCUMENTARIES.

REASONING:
THIS QUERY CHECKS WHETHER THE LISTED_IN FIELD (WHICH CONTAINS GENRE INFORMATION) INCLUDES THE WORD 'DOCUMENTARIES'. IT RETRIEVES ALL CONTENT THAT FALLS UNDER THIS GENRE.

12.FIND ALL CONTENT WITHOUT A DIRECTOR

```sql
-- 12.Find All Content Without a Director
SELECT *
FROM netflix
WHERE director IS NULL;
```

PURPOSE:
TO LIST ALL CONTENT THAT DOES NOT HAVE A DIRECTOR ASSOCIATED WITH IT.

REASONING:
THIS QUERY SIMPLY FILTERS OUT ALL CONTENT WHERE THE DIRECTOR FIELD IS NULL, HELPING TO IDENTIFY CONTENT WITH MISSING DIRECTOR INFORMATION.

13.FIND HOW MANY MOVIES ACTOR 'SALMAN KHAN' APPEARED IN THE LAST 10 YEARS

```
135  -- 13.Find How Many Movies Actor 'Salman Khan' Appeared in the Last 10 Years
136  SELECT *
137  FROM netflix
138  WHERE casts LIKE '%Salman Khan%'
139    AND release_year > EXTRACT(YEAR FROM CURRENT_DATE) - 10;
140
```

PURPOSE:

TO COUNT THE NUMBER OF MOVIES FEATURING 'SALMAN KHAN' RELEASED IN THE PAST 10 YEARS.

REASONING:

THE QUERY CHECKS IF 'SALMAN KHAN' IS LISTED IN THE CASTS FIELD AND FILTERS THE CONTENT BASED ON THE RELEASE YEAR BEING WITHIN THE LAST 10 YEARS. THIS HELPS TRACK THE ACTOR'S PRESENCE ON THE PLATFORM.

14. FIND THE TOP 10 ACTORS WHO HAVE APPEARED IN THE HIGHEST NUMBER OF MOVIES PRODUCED IN INDIA

```sql
-- 14. Find the Top 10 Actors Who Have Appeared in the Highest Number of Movies Produced in Ind
SELECT
    UNNEST(STRING_TO_ARRAY(casts, ',')) AS actor,
    COUNT(*)
FROM netflix
WHERE country = 'India'
GROUP BY actor
ORDER BY COUNT(*) DESC
LIMIT 10;
```

PURPOSE:

TO IDENTIFY THE TOP 10 ACTORS WITH THE MOST APPEARANCES IN MOVIES PRODUCED IN INDIA.

REASONING:

THE CASTS FIELD MAY CONTAIN MULTIPLE ACTORS, SO THE QUERY SPLITS THESE ENTRIES INTO INDIVIDUAL NAMES USING UNNEST(). IT THEN GROUPS BY ACTOR AND COUNTS THEIR APPEARANCES IN INDIAN MOVIES, RETURNING THE TOP 10 ACTORS WITH THE HIGHEST COUNTS.

15.CATEGORIZE CONTENT BASED ON THE PRESENCE OF 'KILL' AND 'VIOLENCE' KEYWORDS

```sql
-- 15.Categorize Content Based on the Presence of 'Kill' and 'Violence' Keywords
SELECT
    category,
    COUNT(*) AS content_count
FROM (
    SELECT
        CASE
            WHEN description ILIKE '%kill%' OR description ILIKE '%violence%' THEN 'Bad'
            ELSE 'Good'
        END AS category
    FROM netflix
) AS categorized_content
GROUP BY category;
```

PURPOSE:

TO CATEGORIZE CONTENT AS 'GOOD' OR 'BAD' BASED ON THE PRESENCE OF SPECIFIC KEYWORDS LIKE 'KILL' OR 'VIOLENCE' IN THE CONTENT DESCRIPTION.

REASONING:

THIS QUERY CHECKS THE DESCRIPTION FIELD FOR KEYWORDS SUCH AS 'KILL' OR 'VIOLENCE'. IF ANY OF THESE WORDS ARE FOUND, THE CONTENT IS CLASSIFIED AS 'BAD'; OTHERWISE, IT IS CLASSIFIED AS 'GOOD'. THE RESULTS ARE THEN GROUPED AND COUNTED BASED ON THESE CLASSIFICATIONS.

CONCLUSION

THIS SQL ANALYSIS OF NETFLIX'S DATASET REVEALED KEY INSIGHTS:

- Content Type Distribution: Grouping by type showed the balance between movies and TV shows, aiding in content strategy decisions.
- Rating Insights: Using RANK() revealed the most frequent ratings, helping understand Netflix's target audience.
- Temporal Analysis: Filtering by release_year provided insights into content trends over specific years.
- Geographical Distribution: UNNEST() allowed us to identify the top content-producing countries, showing regional diversity.
- Longest Content and Seasons: Parsing duration identified the longest movies and TV shows with over 5 seasons.
- Recent Additions: Date functions (TO_DATE()) highlighted content added in the last 5 years.
- Director/Actor Insights: UNNEST() revealed content associated with specific directors and actors like Rajiv Chilaka and Salman Khan.
- Genre and Keyword Categorization: Genre analysis and keyword filtering categorized content for genre diversity and sensitivity.

GOT ANY QUESTIONS?

These SQL queries provided clear insights into Netflix's content catalog, supporting strategic decision-making on content development and acquisition.