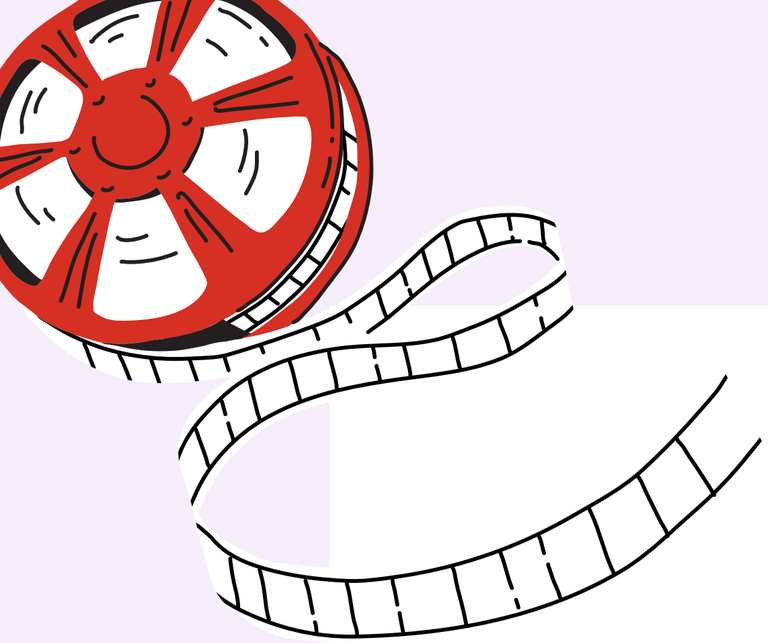


analysing

OTT PLATFORM

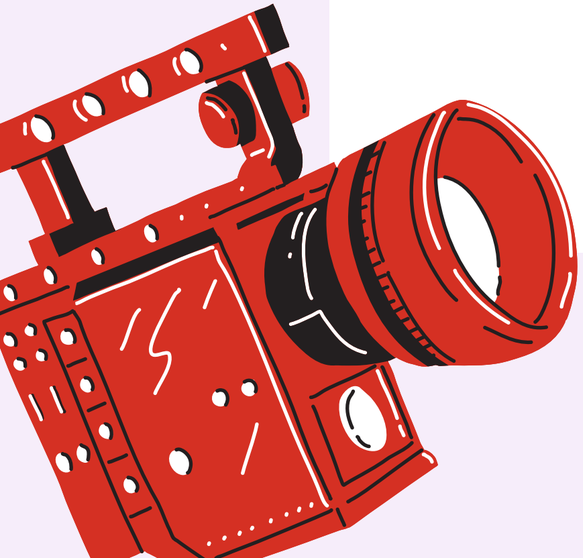


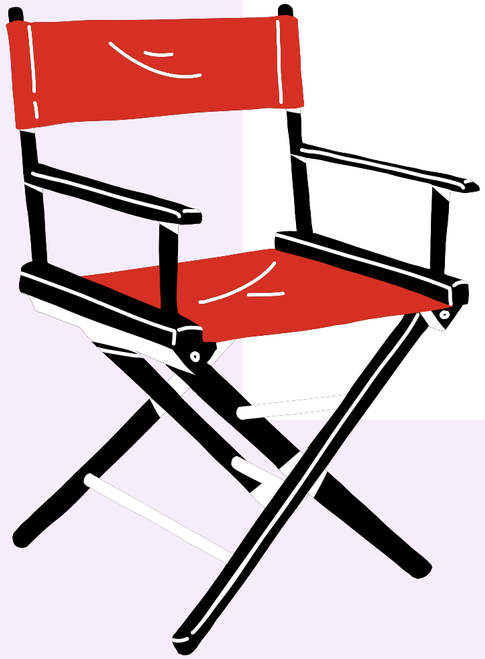
TABLE OF CONTENTS

OBJECTIVE

The project aims to combine advanced relational database design and SQL analytics to support data-driven decision-making

DATASET DESCRIPTION

The dataset is a synthetically generated yet structurally realistic representation of an OTT platform's backend. It mimics platforms like Netflix, Amazon Prime, or Disney+, comprising interlinked tables designed for end-to-end data analysis.



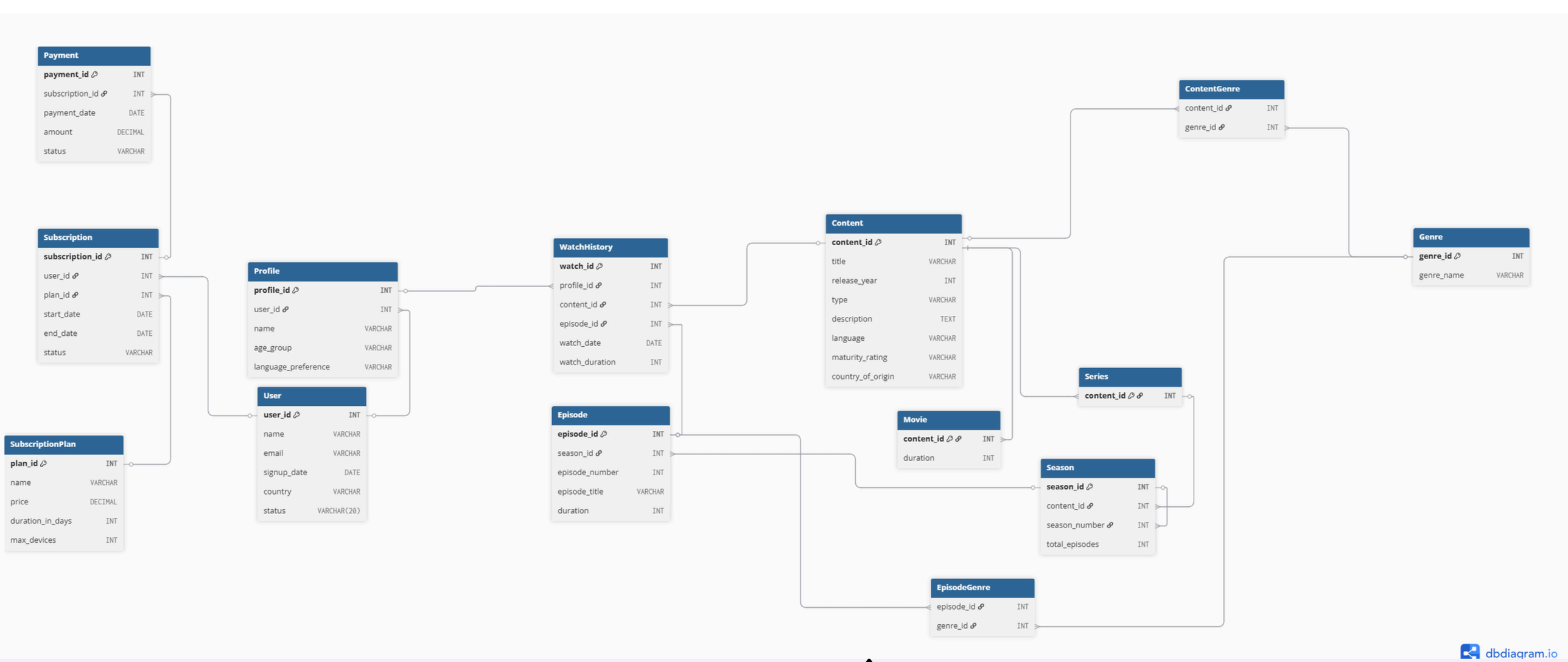
key features

of DATASET

OTT Platform Analytics

Number of Tables: 14
Total Rows: Approx 50,000+

Table Name	Description
AppUser	Contains user account information such as ID, name, email, country, and status.
Profiles	Stores profile-level data (sub-users) linked to AppUser with age group and preferences.
SubscriptionPlans	Lists available plans with price, video quality, device limit, and other features.
Subscriptions	Tracks user subscriptions with start and end dates, linked plan, and status (active/cancelled).
Payments	Stores payment transactions with date, amount, and success/failure status.
Content	Master list of content including movies and series with attributes like maturity rating, language, etc.
Movies	Contains movie-specific data such as duration and content type. Linked to Content.
Series	Contains series-specific data. Linked to Content.
Seasons	Tracks seasons of a series with season number and number of episodes.
Episodes	Contains episodes with duration, title, and link to season and series.
Genres	List of genres available on the platform (e.g., Drama, Action, Comedy).
ContentGenre	Many-to-many mapping table linking content to one or more genres.
EpisodeGenres	Optional table to assign genres at the episode level for fine-grained tagging.
WatchHistory	Logs user watch events by profile, including duration, date, and watched content.


Payment
payment_id INT
subscription_id INT
payment_date DATE
amount DECIMAL
status VARCHAR
Subscription
subscription_id INT
user_id INT
plan_id INT
start_date DATE
end_date DATE
status VARCHAR
SubscriptionPlan
plan_id INT
name VARCHAR
price DECIMAL
duration_in_days INT
max_devices INT
Profile
profile_id INT
user_id INT
name VARCHAR
age_group VARCHAR
language_preference VARCHAR
User
user_id INT
name VARCHAR
email VARCHAR
signup_date DATE
country VARCHAR
status VARCHAR(20)
WatchHistory
watch_id INT
profile_id INT
content_id INT
episode_id INT
watch_date DATE
watch_duration INT
Episode
episode_id INT
season_id INT
episode_number INT
episode_title VARCHAR
duration INT
Content
content_id INT
title VARCHAR
release_year INT
type VARCHAR
description TEXT
language VARCHAR
maturity_rating VARCHAR
country_of_origin VARCHAR
Movie
content_id INT
duration INT
Series
content_id INT
Season
season_id INT
content_id INT
season_number INT
total_episodes INT
EpisodeGenre
episode_id INT
genre_id INT
ContentGenre
content_id INT
genre_id INT
Genre
genre_id INT
genre_name VARCHAR
dbdiagram.io

E-R Diagram

QUERY 1

Which users are the most active on the platform based on watch duration?

```sql
select
appuser.user_id,appuser.name,appuser.email,
sum(watch_duration) as totaltime
from appuser join profiles on appuser.user_id = profiles.user_id
join watchhistory on profiles.profile_id = watchhistory.profile_id
group by appuser.user_id
order by totaltime desc limit 5;
```

	user_id [PK] integer	name character varying (20)	email character varying (100)	totaltime bigint
1	53	Evan	barnold@example.net	4401
2	54	Daniel	amydavenport@example.net	4278
3	93	Kimberly	websterstefanie@example.org	4226
4	98	Brittany	curtisbarton@example.net	4178
5	86	Melissa	vmedina@example.net	4166

QUERY 2

Which 5 content titles have been watched the most in terms of time spent watching across all users?

```
select
content.content_id,content.title ,content.content_type,
sum(watch_duration) as totaltime
from content join watchhistory on content.content_id = watchhistory.content_id
group by content.content_id
order by totaltime desc limit 5;
```

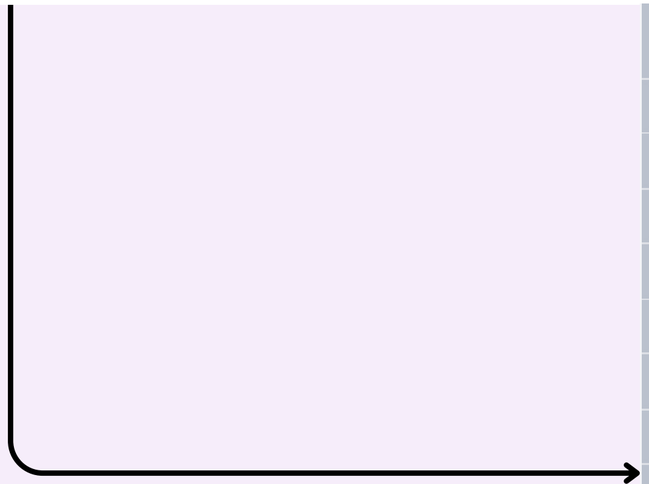
	content_id [PK] integer	title character varying (200)	content_type character varying (20)	totaltime bigint
1	1547	Black Butler	series	283
2	2053	Hannah Gadsby: Douglas	movies	274
3	7370	The Adderall Diaries	movies	260
4	3221	The Edge of Democracy	movies	258
5	1404	Kaali Khuhi	movies	241

Total rows: 5 | Query complete 00:00:00.119

QUERY 3

What is the distribution of content types watched across different age groups?

```
select profiles.age_group , count(watchhistory.content_id)
as content_count, content.content_type
from profiles join watchhistory
on profiles.profile_id = watchhistory.profile_id join content
on watchhistory.content_id = content.content_id
group by
profiles.age_group,content.content_type
order by content.content_type asc, content_count desc;
```



	age_group character varying (20)	content_count bigint	content_type character varying (20)
1	Adult	1046	movies
2	Kids	938	movies
3	Senior	808	movies
4	Teen	740	movies
5	Adult	410	series
6	Senior	379	series
7	Kids	368	series
8	Teen	309	series

Total rows: 8 | Query complete 00:00:00.058

QUERY 4

Classify users as Binge Watcher, Casual Viewer, or Inactive based on total hours watched, and get a count of each category.

```sql
WITH user_watch_time AS (
SELECT
appuser.user_id,
SUM(watchhistory.watch_duration) AS total_watch_time
FROM appuser
JOIN profiles ON appuser.user_id = profiles.user_id
JOIN watchhistory ON profiles.profile_id = watchhistory.profile_id
GROUP BY appuser.user_id
)

SELECT
CASE
WHEN total_watch_time > 1000 THEN 'Binge Watcher'
WHEN total_watch_time BETWEEN 300 AND 1000 THEN 'Casual Viewer'
ELSE 'Inactive'
END AS user_type,
COUNT(*) AS user_count
FROM user_watch_time
GROUP BY user_type
ORDER BY user_count DESC;
```

	user_type text	user_count bigint
1	Binge Watch...	85
2	Casual Viewer	15

Total rows: 2 | Query complete 00:00:00.071

QUERY 5

What is the most watched content (by total watch duration) for each language available on the platform?

```sql
WITH content_watch AS (
SELECT c.content_id,c.title,c.language_type,
SUM(w.watch_duration) AS total_watch_time
FROM content c JOIN watchhistory w ON c.content_id = w.content_id
GROUP BY c.content_id, c.title, c.language_type),

ranked_content AS (
SELECT *,RANK() OVER (PARTITION BY language_type ORDER BY total_watch_time DESC) AS lang_rank
FROM content_watch)

SELECT language_type,title AS top_content,total_watch_time
FROM ranked_content
WHERE lang_rank = 1;
```

language_type character varying (20)	top_content character varying (200)	total_watch_time_mins bigint
English	Hannah Gadsby: Douglas	274
French	The Bonfire of Destiny	178
German	Barbarians	215
Hindi	Kaali Khuhi	241
Japanese	Black Butler	283
Korean	Miss Panda & Mr. Hedgehog	205
Portuguese	The Edge of Democracy	258
Spanish	Bomb Scared	225

rows: 8 | Query complete 00:00:00.081

QUERY 6

What are the top 5 most-watched genre combinations on the platform?

```sql
WITH genre_combinations AS (
SELECT cg.content_id,ARRAY_AGG(g.genre_name ORDER BY g.genre_name) AS genres
FROM contentgenres cg
JOIN genres g ON cg.genre_id = g.genre_id
GROUP BY cg.content_id),

combination_views AS (
SELECT gc.genres,COUNT(wh.watch_id) AS total_views
FROM genre_combinations gc
JOIN watchhistory wh ON gc.content_id = wh.content_id
 GROUP BY gc.genres)

SELECT genres AS genre_combination,total_views
FROM combination_views
ORDER BY total_views DESC
LIMIT 5;
```

	genre_combination character varying[]	total_views bigint
1	{Drama,Other}	650
2	{Comedy,Drama,Other}	372
3	{Other}	329
4	{Comedy,Other}	292
5	{Comedy}	286

Total rows: 5 | Query complete 00:00:00.050

QUERY 7

Which series episodes have the highest drop-off in viewers compared to the previous episode?

```sql
WITH episode_views AS (
SELECT e.episode_id,e.season_id,e.episode_number,s.content_id,
COUNT(DISTINCT wh.profile_id) AS viewers
FROM episodes e JOIN seasons s ON e.season_id = s.season_id
JOIN watchhistory wh ON e.episode_id = wh.episode_id
GROUP BY e.episode_id, s.content_id, e.season_id, e.episode_number
),
episode_dropoff AS (
SELECT ev.*,LAG(viewers)
OVER (PARTITION BY content_id, season_id ORDER BY episode_number) AS prev_episode_viewers,
CASE WHEN LAG(viewers)
OVER (PARTITION BY content_id, season_id ORDER BY episode_number) IS NOT NULL THEN
(LAG(viewers) OVER (PARTITION BY content_id, season_id ORDER BY episode_number) - viewers)
ELSE NULL
END AS viewer_dropoff
FROM episode_views ev)

SELECT c.title AS series_title,ep.episode_number,ep.viewers AS current_episode_viewers,
ep.prev_episode_viewers,ep.viewer_dropoff
FROM episode_dropoff ep
JOIN content c ON ep.content_id = c.content_id
WHERE ep.viewer_dropoff IS NOT NULL
ORDER BY ep.viewer_dropoff DESC
LIMIT 10;
```

series_title character varying (200)	episode_number integer	current_episode_viewers bigint	prev_episode_viewers bigint	viewer_dropoff bigint
Chef's Table	5	1	2	1
Skin Wars	4	1	2	1
Extracurricular	5	1	2	1
Oh My Ghost	3	1	2	1
Trailer Park Boys	2	1	2	1
Liv and Maddie	7	1	2	1
Family Reunion	7	1	2	1
A Queen Is Born	4	1	2	1
Arrow	8	1	2	1
The New Legends of Monkey	4	1	1	0

rows: 10 | Query complete 00:00:00.077

QUERY 8

What are the top 2 most-watched content items for each subscription plan?

```
WITH user_plan AS (
SELECT s.user_id,sp.name AS plan_name
FROM subscriptions s JOIN subscriptionplans sp ON s.plan_id = sp.plan_id
WHERE s.status = 'active')
,
watch_data AS (
SELECT up.plan_name, wh.content_id,COUNT(wh.watch_id) AS total_views
FROM user_plan up JOIN profiles p ON up.user_id = p.user_id
JOIN watchhistory wh ON p.profile_id = wh.profile_id
GROUP BY up.plan_name, wh.content_id),

ranked_content AS (
SELECT wd.*,ROW_NUMBER() OVER (PARTITION BY plan_nameORDER BY total_views DESC)
AS content_rank FROM watch_data wd)

SELECT rc.plan_name,c.title AS content_title,rc.total_views
FROM ranked_content rc
JOIN content c ON rc.content_id = c.content_id
WHERE rc.content_rank <= 2
ORDER BY rc.plan_name, rc.content_rank;
```

plan_name character varying (50)	content_title character varying (200)	total_views bigint
Basic	What Did I Mess	4
Basic	Odu Raja Odu	4
Premium	Accomplice	3
Premium	Eggnoid: Love & Time Portal	3
Standard	The Fear	4
Standard	Force 2	4

rows: 6 | Query complete 00:00:00.042

QUERY 9

Which countries have the highest number of active users on a paid subscription plan in the last 365 days?

```sql
SELECT a.country,COUNT(DISTINCT a.user_id) AS active_paid_users
FROM appuser a
JOIN subscriptions s ON a.user_id = s.user_id
JOIN payments p ON s.subscription_id = p.subscription_id
WHERE a.status = 'active' AND p.status = 'paid'
AND p.payment_date >= CURRENT_DATE - INTERVAL '365 days'
GROUP BY a.country
ORDER BY active_paid_users DESC
LIMIT 5;
```

	country character varying (50)	active_paid_users bigint
1	France	2
2	Japan	2
3	Australia	1
4	Brazil	1
5	Germany	1

Total rows: 5 | Query complete 00:00:00.053

QUERY 10

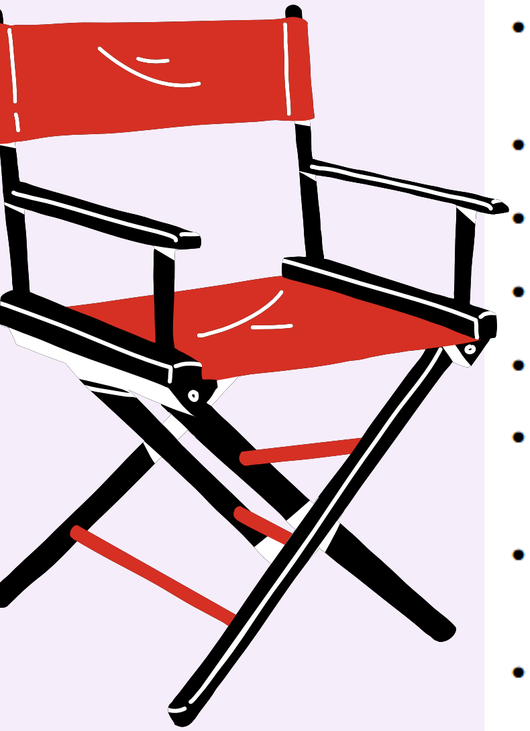
Which genre has the highest number of total watches, and how does the average watch time compare across genres?

```sql
SELECT g.genre_name,COUNT(watchhistory.watch_id) AS total_watches,
ROUND(AVG(watchhistory.watch_duration), 2) AS avg_watch_duration
FROM contentgenres cg
JOIN genres g ON cg.genre_id = g.genre_id
JOIN watchhistory ON cg.content_id = watchhistory.content_id
GROUP BY  g.genre_name
ORDER BY total_watches DESC ;
```

genre_name character varying (50)	total_watches bigint	avg_watch_duration numeric
Other	3155	46.83
Drama	1667	46.49
Comedy	1468	46.59
Action	904	47.01
Documentary	636	46.85
Family/Kids	602	47.63
Reality/Non-Fiction	392	43.45
International	330	46.81
Crime/Mystery	329	46.71
Horror	216	47.98
Sci-Fi/Fantasy	163	51.09

I rows: 11 | Query complete 00:00:00.062

Business Insights



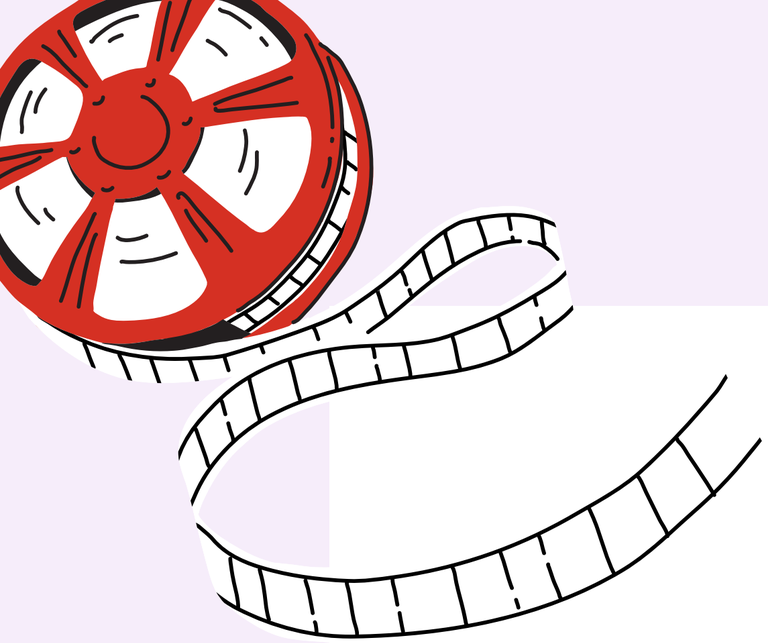
- **Top Users by Watch Time:** Power users like *Evan*, *Daniel*, and *Kimberly* contribute disproportionately to the total watch time.
- **High Engagement Titles:** *Black Butler* and *Hannah Gadsby: Douglas* are among the most-watched content pieces.
- **Age Group Preferences:** *Adults* consume the highest volume of both movies and series.
- **User Segmentation:** About 85% of users are *Binge Watchers*.
- **Language Preferences:** *English*, *Japanese*, and *Hindi* dominate in total watch time.
- **Popular Genre Combinations:** *Drama*, *Comedy*, and *Other* are most frequently viewed.
- **Viewer Drop-Off in Series:** Consistent drop-off (1 viewer) suggests room for more engaging content arcs.
- **Content by Subscription Plan:** Similar watch counts across plans suggest opportunity for tailored content bundles.
- **Genre-Level Patterns:** *Sci-Fi/Fantasy* has the highest average watch duration.
- **Active Paid Users by Country:** *France* and *Japan* are top-performing regions.

CONCLUSION

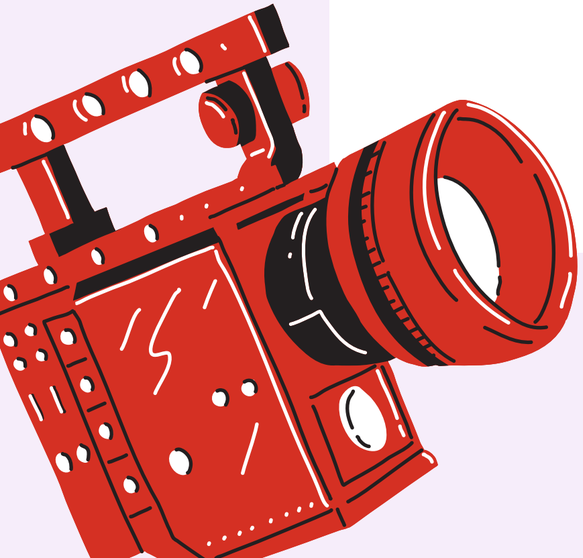
The OTT Platform Analytics successfully delivers a comprehensive and scalable analytics solution that simulates real-world streaming platforms like Netflix or Amazon Prime. Through a well-normalized relational schema and advanced SQL queries, we have extracted key business insights spanning user engagement, content performance, subscription behavior, and revenue trends.Title

we implemented 10 business-relevant queries that power decision-making in areas like:

- User segmentation (binge-watchers, casual users, inactive profiles)
- Content strategy (top genres, most rewatched shows, maturity rating trends)
- Revenue analytics (monthly revenue trends, average revenue per user)
- Subscription insights (churn detection, active/inactive users by country)



the end



What might I analyse further?

- Churn Analysis
- User Segmentation via Clustering
- Drop-off Analysis by Series
- Subscription Plan Optimization