**Project Title: "StreamFlix Business Analytics Platform"**

**Theme**: You’re a data engineer/analyst for a fictional streaming service, **StreamFlix**, tasked with building a full analytical suite to support product, marketing, and operations teams.

**Goal**: Learn SQL while solving **real-world business problems**, including data quality, aggregations, trends, segmentation, and recommendations.

**Dataset Source**

Use a mix of **IMDb datasets** and **Kaggle streaming datasets** (or create dummy transactional tables). Example:

* [IMDb Dataset](https://datasets.imdbws.com/)
* [Kaggle Netflix Viewing Habits](https://www.kaggle.com/datasets/shivamb/netflix-shows)
* [User engagement simulator CSV](you can manually generate a CSV or use Faker)

**Tools Required**

* **MySQL**, **SQLite**, or **DuckDB**
* **DBeaver** or **MySQL Workbench**
* **CSV files** (loaded via LOAD DATA INFILE or .import)

**Project Phases: Business Use Cases with SQL Concepts**

**Phase 1: Initial Setup + Raw Data Load**

*Business Context*: Load all base data tables: movies, users, views, ratings.

**Tables:**

* movies: Metadata from IMDb
* users: User profile table
* views: Each row = 1 view of a movie (user\_id, movie\_id, watch\_time, timestamp)
* ratings: User-submitted rating (movie\_id, user\_id, rating, created\_at)

**Skills Covered:**

* Installing DB & tools
* Creating tables, understanding DDL/DML
* Loading CSVs into tables
* Basic SELECT/WHERE queries

**Phase 2: Movie Catalog + Popularity Analysis**

*Use Case*: Help marketing find the most-watched and highest-rated movies by genre.

**Tasks:**

* Find top 5 genres by average rating (with MIN 100 votes).
* Count of views per movie per month (GROUP BY with DATE\_TRUNC)
* Create a CASE column for movie length: Short (<90 min), Standard (90–120), Long (>120)

**Skills Covered:**

* GROUP BY, HAVING, JOINs
* Aggregate Functions
* CASE Statements

**Phase 3: Power Users & Behaviour Segmentation**

*Use Case*: Identify binge-watchers, dormant users, and likely churn risks.

**Tasks:**

* Segment users using CASE:
  + "Binger": >10 views/week
  + "Occasional": 1–10 views/week
  + "Inactive": 0–1 view/week
* Use COALESCE to handle null ratings.
* Find top 10% of users by watch time (window RANK() or NTILE())

**Skills Covered:**

* Conditional logic (IF, CASE, COALESCE)
* Subqueries, CTEs
* Window functions

**Phase 4: Recency-Frequency Model for Engagement**

*Use Case*: Help retention team identify loyal users vs drop-offs using RFM.

**Tasks:**

* Create a CTE to get:
  + recency: Days since last view
  + frequency: Total views in last 30 days
  + monetary: Total watch time
* Classify:

sql CASE WHEN recency < 5 AND frequency > 10 THEN 'Loyal' WHEN recency BETWEEN 5 AND 15 THEN 'At Risk' ELSE 'Dormant' END

**Skills Covered:**

* CTEs, date operations, advanced CASE
* Data modeling and user-level aggregations

**Phase 5: Data Quality & Performance Optimization**

*Use Case*: Ops team wants to track data quality issues & optimize slow queries.

**Tasks:**

* Use REGEX to identify movies with malformed titles (REGEXP '^[^A-Za-z]')
* Run EXPLAIN to analyze performance of a JOIN-heavy query
* Add indexes on user\_id, movie\_id for the views table
* Clean duplicates via ROW\_NUMBER() in a window + delete

**Skills Covered:**

* REGEXP
* Query Plans
* Indexing
* Data cleaning via CTEs + Window Functions

**Phase 6: Executive Dashboard (All SQL)**

*Use Case*: Final output: a single dashboard query for daily reporting.

**Fields:**

* Total Views Today
* New Users This Week
* Avg. Watch Time / Movie
* Top 3 Movies by Genre This Week

Build it using:

* Windowed aggregates
* Multiple CTEs
* CASE classification
* Subqueries for metrics

**Final Deliverables**

| **Deliverable** | **Tool** | **Format** |
| --- | --- | --- |
| Cleaned SQL schema | DDL script | .sql |
| Sample data files | CSVs | .csv |
| Dashboard query file | Final SQL | dashboard.sql |
| Optional ERD | DBeaver export | PNG/SQL |

**Summary of Concepts by Phase**

| **Concept** | **Phase** |
| --- | --- |
| DDL, DML, CSV Load | Phase 1 |
| Joins, Aggregates, CASE | Phase 2 |
| CTE, IF, COALESCE, Windows | Phase 3 |
| Nested Subqueries, Date Math | Phase 4 |
| Indexing, EXPLAIN, REGEX | Phase 5 |
| Multi-CTE Dashboard Query | Phase 6 |