Silver Screen Project

By: Bianca Niemann

Role: BI Analyst, Entertainment Company

Tooling: dbt + Snowflake

A Case Study on Silver Screen Theaters – New Jersey

Company Context:

- Entertainment company recently acquired Silver Screen
- Operates 3 movie theaters in New Jersey

Goal:

- Management wants to understand the relationship between movie rental costs and revenue generated

Data Sources

Five data sources in csv format (loaded into Snowflake):

movie_catalogue	Movie detailed info (title, genre, studio, director, budget etc) for 2024
invoices	Monthly rental cost per movie/location
nj_001	Transactions from location #1
nj_002	Daily transactions from location #2
nj_003	Individual purchases from location #3 (incl snacks, drinks and tickets)

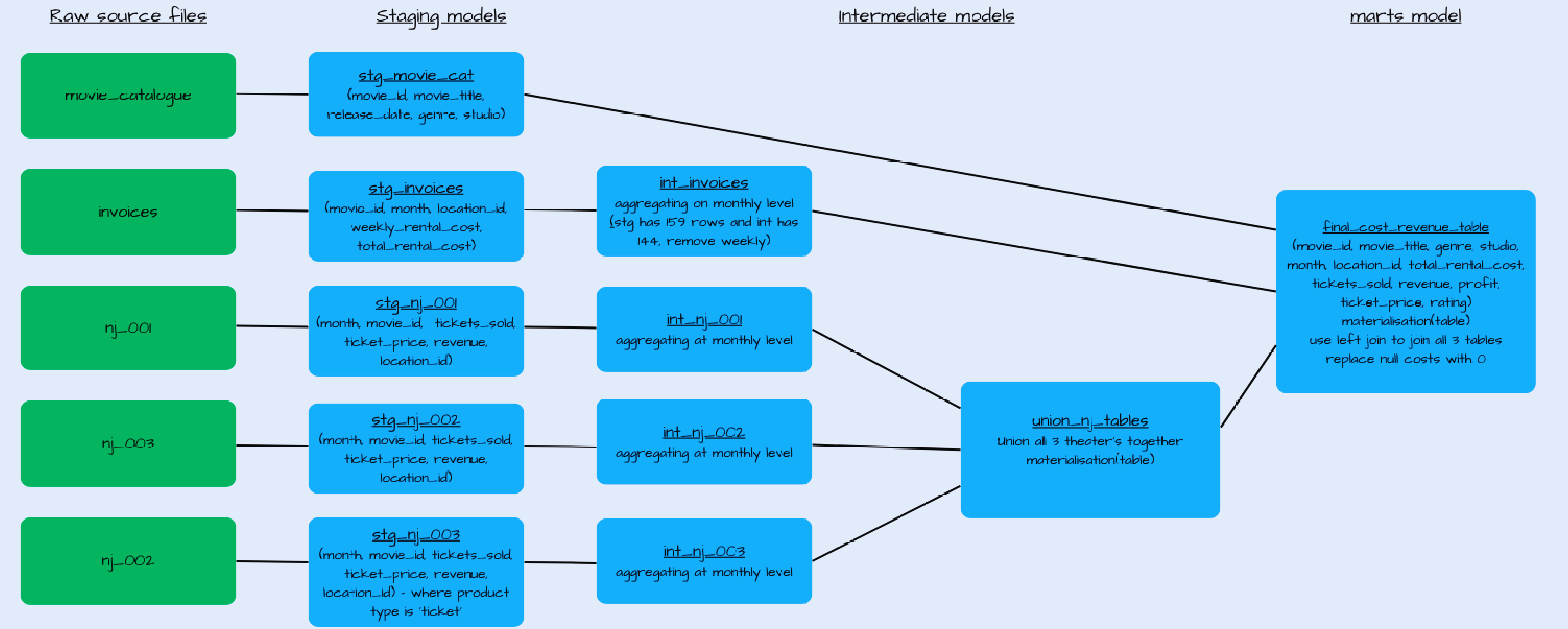
Data Challenges & Cleaning

- Different column names/structures
- Extract the month from timestamp (nj_001)
- Aggregate the data from daily to monthly (nj_002)
- Mixed product types in location (e.g. tickets vs snacks) - need to filter just tickets (nj_003)
- Replace null Genres with 'Unknown' (movie_catalogue)
- Check for duplicates

Tools Used:

- SQL in Snowflake for exploration
- dbt for cleaning, transformation, and modeling

DAG



Raw source files

movie_catalogue

invoices

nj_001

nj_003

nj_002

Staging models

stg_movie_cat
(movie_id, movie_title, release_date, genre, studio)

stg_invoices
(movie_id, month, location_id, weekly_rental_cost, total_rental_cost)

stg_nj_001
(month, movie_id, tickets_sold, ticket_price, revenue, location_id)

stg_nj_002
(month, movie_id, tickets_sold, ticket_price, revenue, location_id)

stg_nj_003
(month, movie_id, tickets_sold, ticket_price, revenue, location_id) - where product type is 'ticket'

Intermediate models

int_invoices
aggregating on monthly level (stg has 159 rows and int has 144, remove weekly)

int_nj_001
aggregating at monthly level

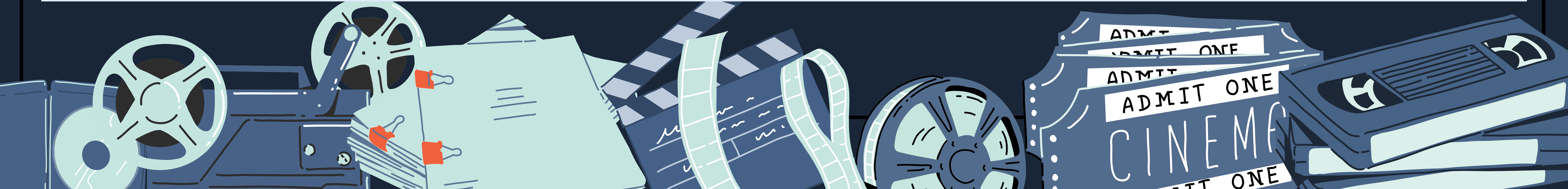
int_nj_002
aggregating at monthly level

int_nj_003
aggregating at monthly level

union_nj_tables
Union all 3 theater's together
materialisation(table)

marts model

final_cost_revenue_table
(movie_id, movie_title, genre, studio, month, location_id, total_rental_cost, tickets_sold, revenue, profit, ticket_price, rating)
materialisation(table)
use left join to join all 3 tables
replace null costs with 0



dbt Workflow

Step-by-step process using dbt:

1. Staging models for each source:
   - Cleaned column names
   - Parsed dates and standardized formats
2. Intermediate models:
   - Unified ticket sales across locations at monthly level
   - Aggregated invoices to be at monthly level
3. Final model:
   - Rental cost (invoices) and Movie details joined to Unioned sales table
   - Aggregated to month x movie x location
   - Created a unique_row_id using dbt_utils and generate_surrogate_key

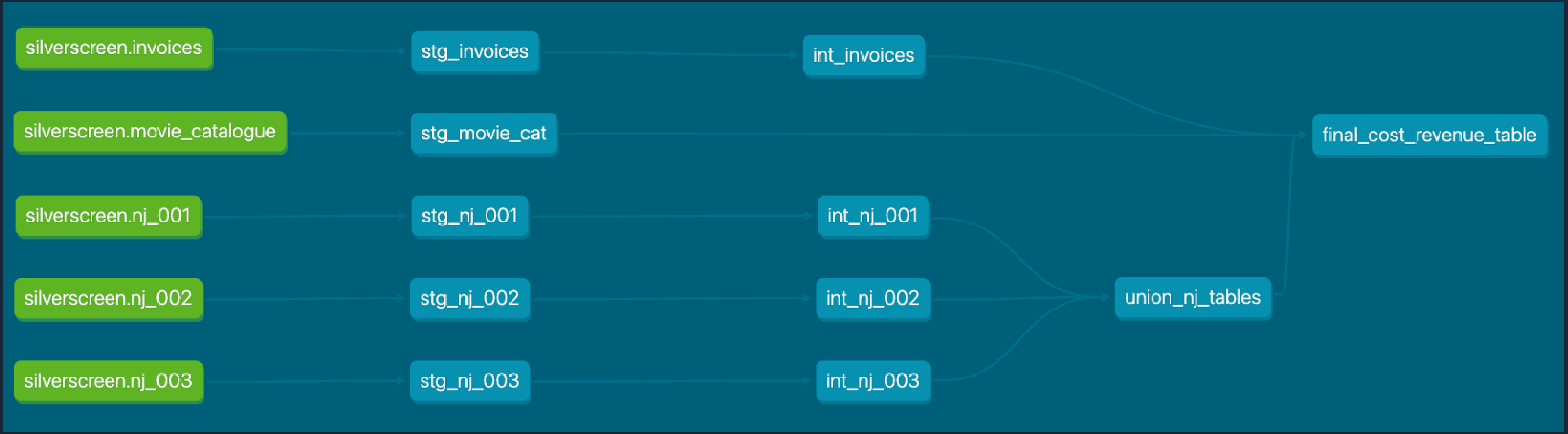
📁 Final table: fct_movie_monthly_performance

Final Table Structure

unique_row_id	created using dbt_utils and generate_surrogate_key
movie_id	unique movie identifier
movie_title	Title of the movie
genre	Genre of the movie
studio	Producing Studio
month	Calander month of invoice/sale

location_id	Theater location
total_rental_cost	Monthly cost of renting movie
total_tickets_sold	Total tickets sold that month at location
total_revenue	Total revenue from ticket sales
profit	Total revenue less total rental cost (not pure profit)

Lineage



silverscreen.invoices
stg_invoices
int_invoices

silverscreen.movie_catalogue
stg_movie_cat
final_cost_revenue_table

silverscreen.nj_001
stg_nj_001
int_nj_001

silverscreen.nj_002
stg_nj_002
int_nj_002
union_nj_tables

silverscreen.nj_003
stg_nj_003
int_nj_003

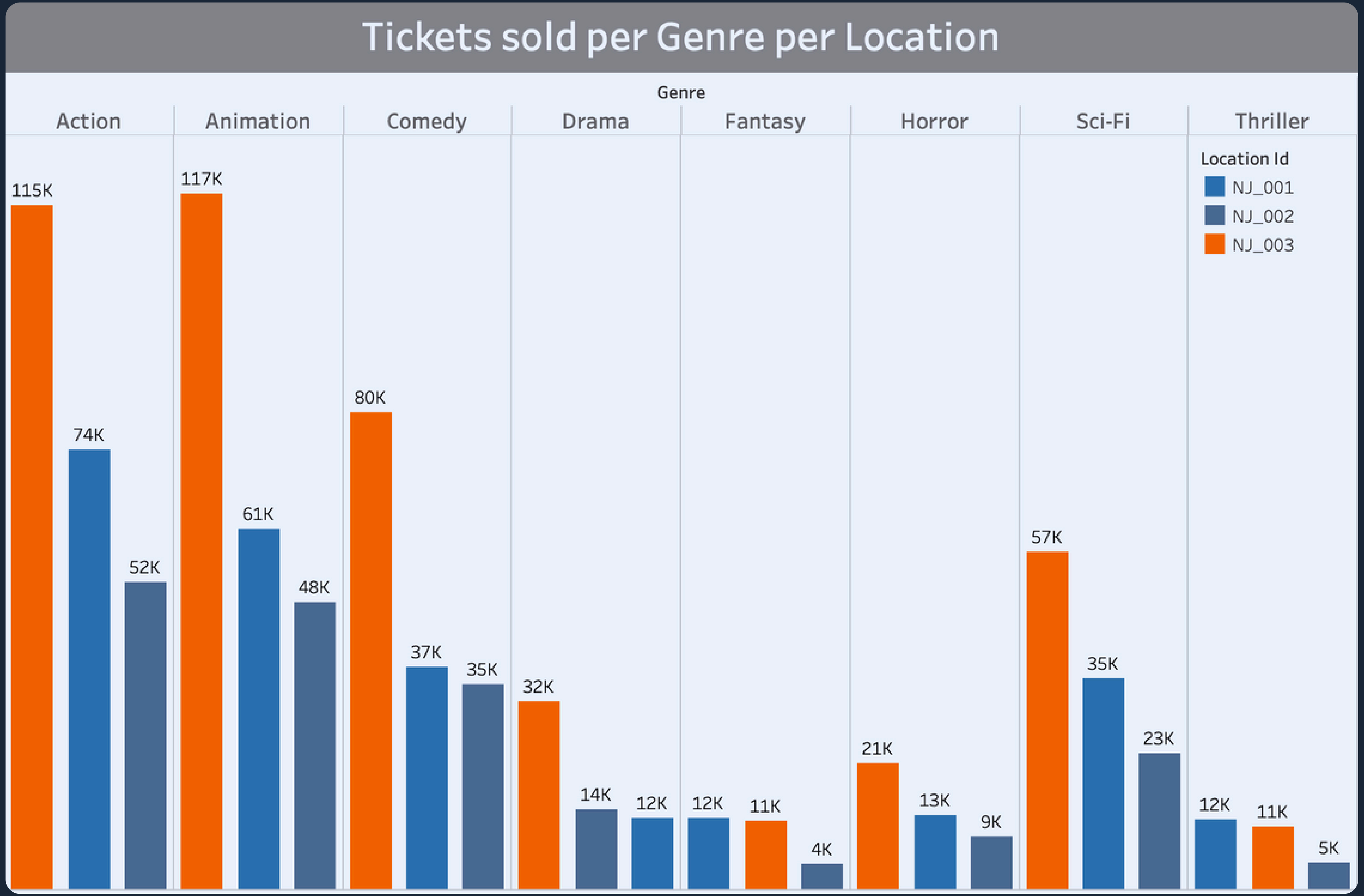
Tests and Macros

- Test - rental_cost_non_negative (tests no invoices reflect a negative cost)
- Macro - revenue_non_negative (tests via scheme that no stg_nj tables have negative revenue

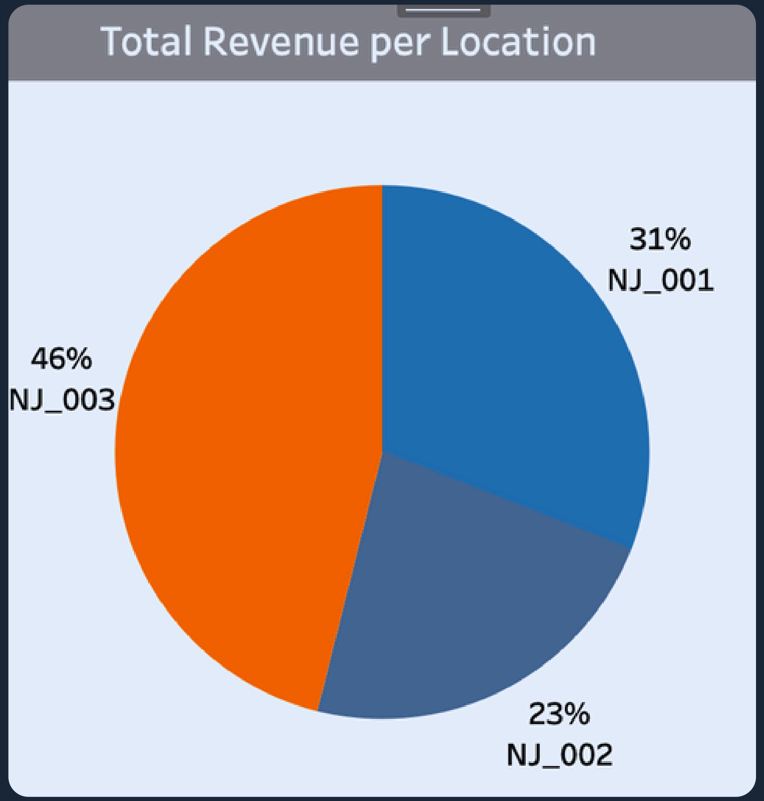
Automation with dbt Cloud

- Job scheduled weekly via dbt Cloud
- Includes:
- dbt build (gold)
    - dbt run to refresh models
    - dbt test to validate data (e.g. revenue ⩾ 0)
- Version controlled with Git, fully reproducible

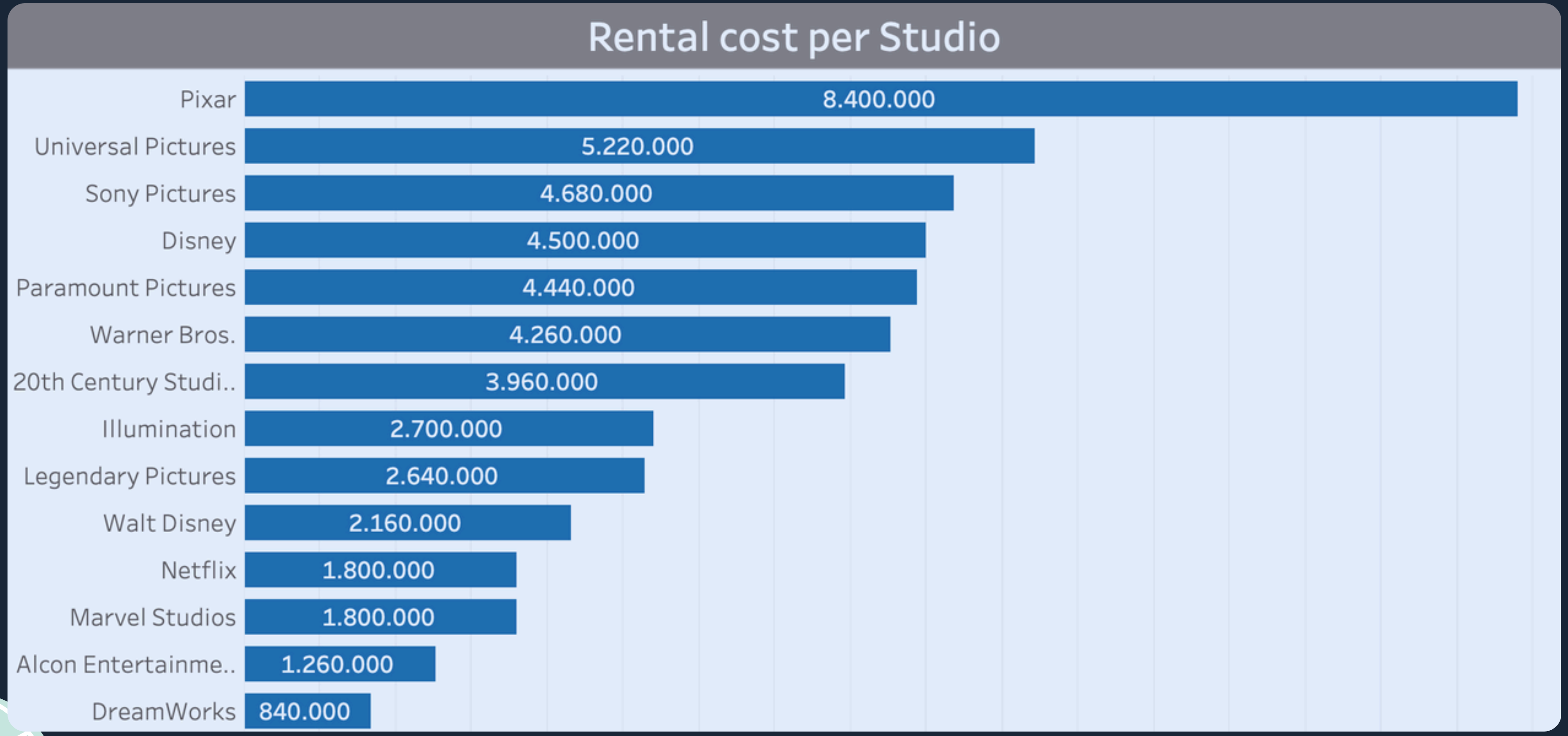
Visualizations



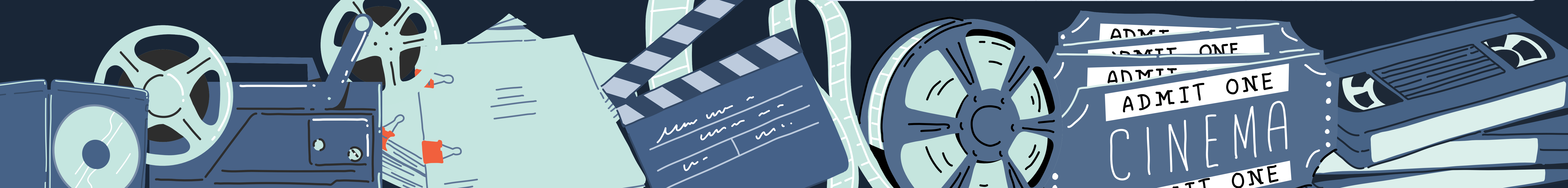
Tickets sold per Genre per Location



Total Revenue per Location



Rental cost per Studio



Thank You

from Bianca Niemann

FINISH