**A: BUSINESS REPORT: Revenue by Movie**

This report uses the DVD dataset to determine the total revenue generated by each movie to see which movies are the most profitable. Knowing the most profitable movies will help the DVD rental business make decisions about many things including which movies to promote and which movies to stock more of or remove from inventory. Knowing the most and least profitable movies is important for the business to maximize revenue.

**A1: SPECIFIC FIELDS**

Fields for detailed table, “Movie Revenue Details”

* Film ID (film\_id)
* Film Title (film\_title)
* Total Rentals (total\_rentals)
* Total Revenue (total\_revenue)
* Average Revenue per Rental (average\_revenue\_per\_rental)
  + Average amount of money generated per rental for each film, calculated by averaging rental amounts

Fields for the summary table, “Top Profitable Movies”

* Film ID (film\_id)
* Film Title (film\_title)
* Total Revenue (total\_revenue)

**A2: TYPES OF DATA**

* Integers (INT)
  + Film ID - unique identifier for each film used to link data across tables
  + Total Rentals - count of how many times a movie has been rented
* Decimals (DECIMAL)
  + Total Revenue - sum of money generated from renting each movie
  + Average Revenue per rental - average amount of money generated per rental for each film
* Strings (VARCHAR)
  + Film Title - name of the film

**A3: SPECIFIC TABLES**

Specific tables used to generate the detailed and summary tables include the film table, payment table, inventory table, and rental table.

**A4: FIELD TRANSFORMATION**

The Total Revenue field in the Top Profitable Movies summary report could require a custom transformation with a user-defined function for the total revenue amounts. The transformation will take the plain numbers and turn them into currency format for better readability. It will allow all stakeholders to easily notice the monetary amounts to make business decisions.

**A5: BUSINESS USES**

The Movie Revenue Details table will help the business analyze each film’s performance on a granular level including rental frequency and revenue generation. It will also shed light on the least profitable movies which is important to know for inventory purposes.

The Top Profitable Movies table will provide a quick overview of the top-performing films in terms of profit. It will be useful for operational decisions for marketing strategies and setting rental pricing.

**A6: REPORT FRESHNESS**

This report should be refreshed weekly since movie rentals and revenue figures can change semi frequently due to new releases and marketing promotions. A weekly refresh will make sure stakeholders have the most recent data without having to maintain a daily update routine.

**B: FUNCTIONS**

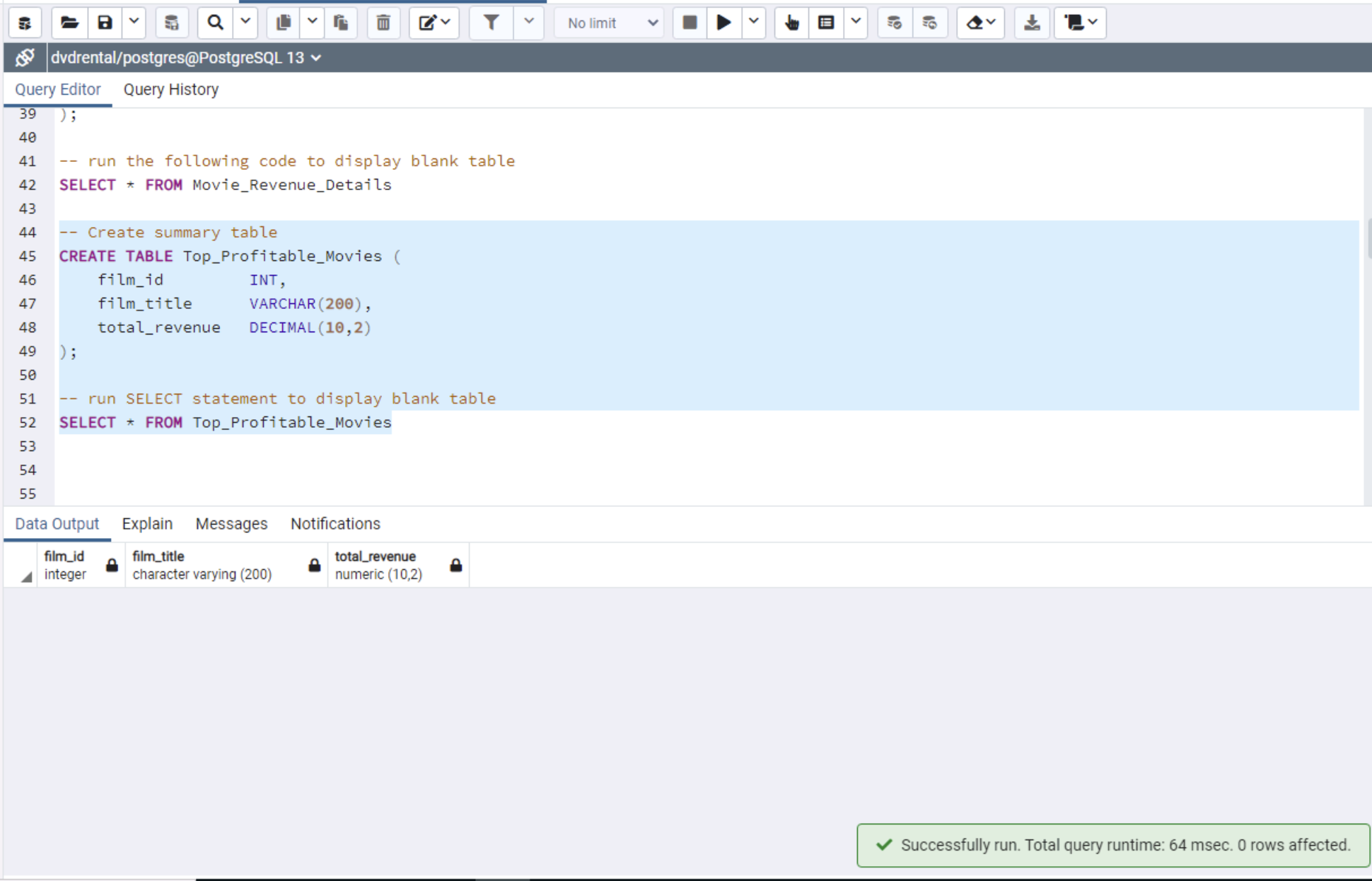
**A screenshot of a computer

Description automatically generated**

**C: CREATING TABLES**

A screenshot of a computer

Description automatically generated



**D: SQL QUERY**

A screenshot of a computer

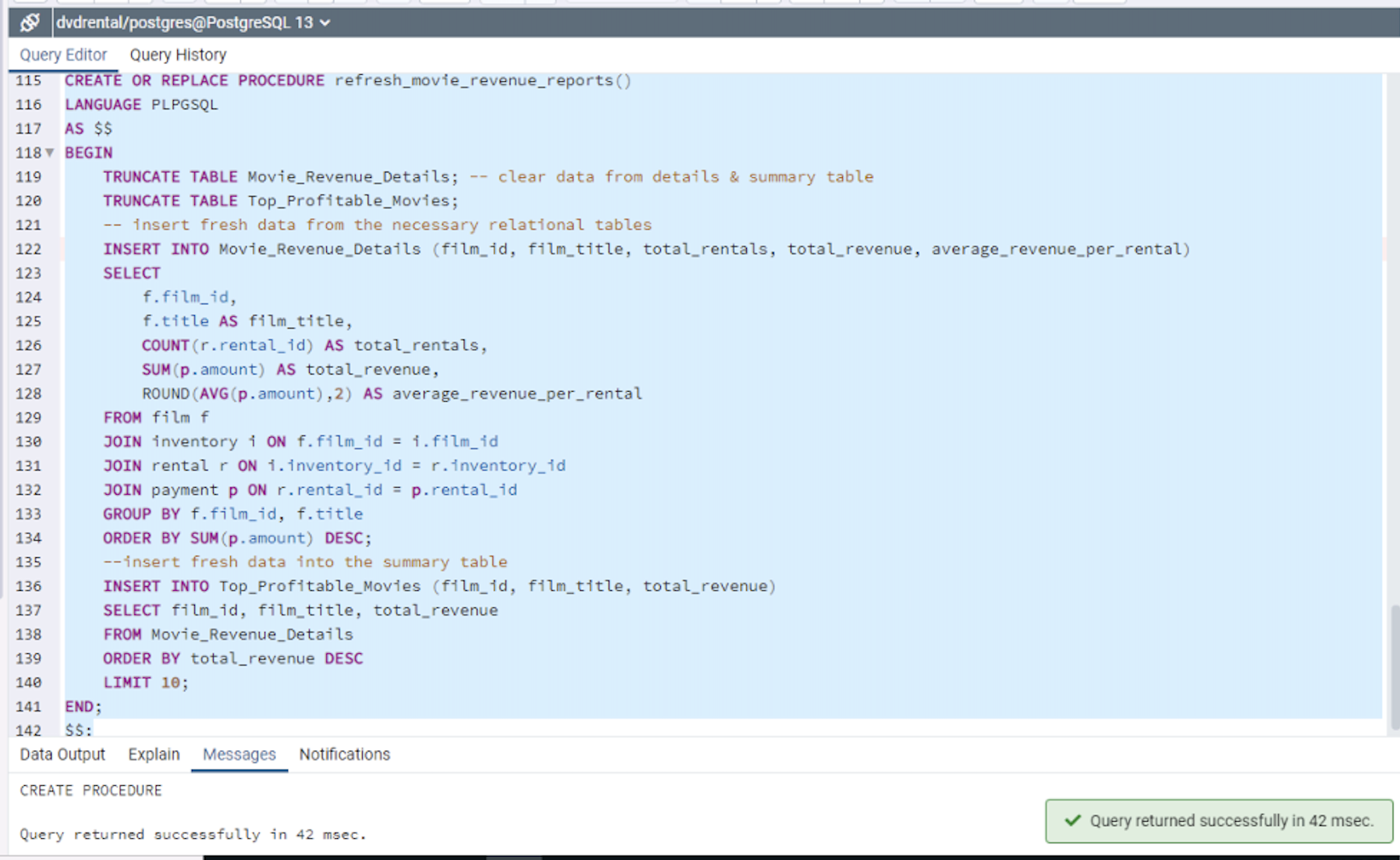
Description automatically generated

**E: TRIGGERS**

A screenshot of a computer program

Description automatically generated

**F: STORED PROCEDURES**



**F1: JOB SCHEDULING TOOL**

According to an article from Digital Ocean, one job scheduling tool that could be used to automate this stored procedure is pgAgent. It integrates with PostgreSQL and is managed with pgAdmin.

**H: SOURCES**

DigitalOcean

* DigitalOcean. (n.d.). How to Schedule Automatic Backups for PostgreSQL with pgAgent in pgAdmin. Retrieved from <https://www.digitalocean.com/community/tutorials/how-to-schedule-automatic-backups-for-postgresql-with-pgagent-in-pgadmin>

PostgreSQLTutorial.com

* PostgreSQLTutorial.com. (n.d.). Getting Started with PostgreSQL Sample Database. Retrieved from <https://www.postgresqltutorial.com/postgresql-getting-started/postgresql-sample-database/>
* PostgreSQLTutorial.com. (n.d.). PostgreSQL CREATE FUNCTION Statement. Retrieved from <https://www.postgresqltutorial.com/postgresql-plpgsql/postgresql-create-function/>
* PostgreSQLTutorial.com. (n.d.). PostgreSQL Triggers. Retrieved from <https://www.postgresqltutorial.com/postgresql-triggers/>

SQLShack

* SQLShack. (n.d.). A Comprehensive Guide to the SQL FORMAT Function. Retrieved from <https://www.sqlshack.com/a-comprehensive-guide-to-the-sql-format-function/>

W3Schools

* W3Schools. (n.d.). SQL Stored Procedures. Retrieved from <https://www.w3schools.com/sql/sql_stored_procedures.asp>