**DVD Rental Database Management/Analysis**

**Dylan S. Franklin (Student ID: 001295234)**

**3/2/2023**

**Western Governors University**

**D191 Advanced Data Management**

**A. Summarize one real-world written business report that can be created from the DVD Dataset from the “Labs on Demand Assessment Environment and DVD Database” attachment.**

In this project, we will be using PostgreSQL to extract data from the DVDRENTAL database to answer the business question, **“what film genre is sold the most?”**. We’re looking for a solution to this question to gauge where our resources should be spent. Let's say action films are sold four times as much as horror films, it would then make sense to spend more resources on action films over horror films to maximize profit. This is an important business question to answer to maximize the resources spent to profit ratio.

**A1 + A2. Identify the specific fields that will be included in the detailed table and the summary table of the report. Describe the types of data fields used for the report.**

The detailed table contains the following fields. The detailed table will handle all the data joined by three tables. The category table, the film\_category table, and the film table.

**detailedtable**

| category \_id (INTEGER) | category\_name (varchar(20)) | title (varchar(200)) | release\_year(integer) | rental\_rate  (numeric(4,2)) | film\_id(integer) |
| --- | --- | --- | --- | --- | --- |
| 1 | Action | Side Ark | 2006 | 2.99 | 24 |
| 1 | Action | Logan | 2006 | 3.99 | 18 |
| … | … | … | … | … | … |

**category\_id** - An integer that holds the film genre ID (Action = 1, Animation = 2, Children = 3, etc.). Acts as the PK for the category table. This information is pulled from the category table. Will be used in the summary table.

**category\_name** - A varchar of set length 20 that holds the film genre name (Action, Animation, Children, etc.) This information is pulled from the category table. Will be used in the summary table.

**title** - A varchar of set length 200 that holds the film title. This information is pulled from the film table.

**release\_year** - An integer that holds the film's year of release. Will be conducted with the film title to improve readability. This information is pulled from the film table.

**rental\_rate -** A numeric value that represents a particular film's rental\_rate, or how often it’s rented. This information is pulled from the film table. Will be used in calculations to determine the total\_rental\_rate for each category in the summary table.

**film\_id** - An integer that associates itself with a particular film. Acts as the primary key (PK) in the film and film\_category tables. Information is pulled from the film\_category table.

The summary table contains the following fields. The summary table will hold the information that answers the business question—done by getting the sum of all rental\_rate numbers for each category of movie.

**summarytable**

| category\_id(integer) | category\_name(varchar(20)) | total\_rental\_rate (numeric) |
| --- | --- | --- |
| 1 | Action | 348.69 |
| 2 | Animation | 380.39 |
| … | … | … |

**category\_id** - An integer that holds the film genre ID (Action = 1, Animation = 2, Children = 3, etc.).

**category\_name** - A varchar of set length 20 that holds the film genre name (Action, Animation, Children, etc.)

**total\_rental\_rate** - A numeric value that shows the rental rate for an entire genre, not just an individual movie like how it’s shown in the detailed table.

**A3. Identify at least two specific tables from the given dataset that will provide the data necessary for the detailed table section and the summary table section of the report.**

Data is pulled from many tables. The category table, the film table, and the film\_category table.

For the detailed table, we need…

* category\_name from the category table
* category\_id from the category table
* title from the film table
* rental\_rate from the film table
* film\_id from the film table OR the film\_category table.

For the summary table, we need…

* category\_id from the detailed table
* category\_name from the detailed table
* total\_rental\_rate , SUM(rental\_rate) from the detailed table for each category\_id

**A4. Identify at least one field in the detailed table section that will require a custom transformation with a user-defined function and explain why it should be transformed (e.g., you might translate a field with a value of N to No and Y to Yes).**

One field in the detailed table that will require a transformation is the title\_and\_year field. It takes two fields, film.title, and film.release\_year, and concates the two together to create the title\_and year\_ fields in the detailed table. This is done to provide extra information to the detailed table by applying the release year of the film to the name in parentheses. This allows for better reliability too.

**A5. Explain the different business uses of the detailed table section and the summary table section of the report.**

The business use for the detailed table includes looking for a film's basic information, such as the name of the film, the genre it’s identified as, and its rental rate. This is useful to stakeholders if they only want a single film's information.

The business use for the summary table includes information on an entire genre’s rental rate. This is useful to stakeholders if they want to see what genre is the most popular/sells the most among all available genres. This information can be used to make business decisions, such as prioritizing sports films over classic films because the rental rate for sports films is greater than classic films.

**A6. Explain how frequently your report should be refreshed to remain relevant to stakeholders.**

The report should be refreshed at least every six months to detect any notable changes in rental rates. Things that could affect the rental rates include trends, current actors, etc. Refreshing the report every season could prove to be useful too, as many people rent films based on the time of the season, such as Christmas films being more prevalent during the winter season.

**B-F. Table creation, Functions, Triggers, Procedures, etc.**

All SQL queries can be viewed by opening the attached SQL file (SQLQueries)

**1. Identify a relevant job scheduling tool that can be used to automate the stored procedure.**

pgAgent is one of if not the best job scheduling tools to use with PostgreSQL and pgAdmin4 to schedule tasks, so we would be using that. Data should be refreshed at least every six months to detect any notable changes in rental rates. However, as stated before, a refresh rate of every 3 months (seasonal) could prove to be useful too, as many people rent films based on the time of the season

**G. Provide a Panopto video recording that includes the presenter and a vocalized demonstration of the functionality of the code used for the analysis.**

PANOPTO VIDEO LINK: <https://wgu.hosted.panopto.com/Panopto/Pages/Viewer.aspx?id=2a2230c0-a356-481d-9a97-afbc0183145a>

**H. Acknowledge all utilized sources, including any sources of third-party code, using in-text citations and references. If no sources are used, clearly declare that no sources were used to support your submission.**

No noteworthy sources were used in this submission.