Rebekah Romang

010241827

D326

Advanced Data Management

3/25/2025

PART 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.

The business report I'm creating focuses on the store revenue. This report consists of two tables. The first being a detailed table which will contain all data from the store id, rental id, rental date, month, and payment amount for both stores. The second being a summary table which will contain transformed data from the detailed table showing a count of the number of rentals and a sum of the total sales for each store sorted by month.

PART A1

Identify the specific fields that will be included in the detailed table and the summary table of the report.

Detailed table:

- store_id
- payment_id
- month
- payment_date
- amount

Summary table:

- store_id
- month
- total_rentals
- total_sales

PART A2

Describe the types of data fields used for the report.

Detailed table:

- store_id SMALLINT – this column contains a digit representing the specific store. There are two stores so this field will contain a value of either 1 or 2.
- payment_id INT – this column contains an integer representing a customer payment.
- month VARCHAR – this column will contain a month displayed as 'Month' which has been transformed from the payment_date timestamp.
- payment_date TIMESTAMP – this column contains a date and time from when a payment occurred.
- amount MONEY – this column contains a money value from a rental transaction.

Summary table:

- store_id SMALLINT – same as on the detailed table, this column will contain a single digit representing the specific store.
- month VARCHAR – same as on the detailed table, this column will contain a month displayed as 'Month' which has been transformed from the rental_date timestamp.
- total_rentals INT – this column contains a count of payment_id from the detailed table.
- total_sales MONEY - this column contains a sum of the amount values from the detailed table.
-

PART 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.

The detailed table will contain data from the rental, inventory, and payment tables, while the summary table will be using data from the detailed table.

PART 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).

I will be transforming the payment_date from a timestamp that returns 2007-02-15 22:25:46.996577 to the more readable month value of 'February'. This transformation is required for the summary table to function as a monthly report showing total rentals and sales.

PART A5

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

The detailed table contains all rental revenue data and could be used for record-keeping and tax preparations. The summary table is great for an 'at a glance' type of report for comparing store performance. The summary report could provide information on which store is performing better and provide insight for business questions. Perhaps stakeholders would like to increase the price of new arrivals but are worried about customer receptiveness, one store could do a test run with the price increase, then at the end of the month the summary report will show how the price increase effected the sales and rental totals.

PART A6

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

The reports need refreshing every month as new data becomes available. The best-case scenario would be to run on the first day of the month to remain relevant with the most up to date data.

PART B

Provide original code for function(s) in text format that perform the transformation(s) you identified in part A4.

```
-- Function to transform timestamp to month

CREATE OR REPLACE FUNCTION get_month(payment_date TIMESTAMP)
RETURNS VARCHAR(9)
LANGUAGE plpgsql
AS $$
BEGIN
```

```
RETURN to_char(payment_date, 'Month');
END; $$;

-- Test function
SELECT get_month('2025-08-25');
```

PART C

Provide original SQL code in a text format that creates the detailed and summary tables to hold your report table sections.

```
-- Create detailed table
CREATE TABLE detailed_table(
      store_id SMALLINT,
      payment_id INT,
      month VARCHAR(9),
      payment_date TIMESTAMP,
      amount MONEY );

-- Create summary table
CREATE TABLE summary_table(
      store_id SMALLINT,
      month VARCHAR(9),
      total_rentals INT,
      total_sales MONEY );
```

PART D

Provide an original SQL query in a text format that will extract the raw data needed for the detailed section of your report from the source database.

```
-- Populate detailed table
TRUNCATE detailed_table;
INSERT INTO detailed_table
      SELECT
            i.store_id,
            p.payment_id,
            get_month(p.payment_date),
            p.payment_date::date,
            p.amount::money
      FROM payment AS p
      INNER JOIN rental r ON p.rental_id = r.rental_id
      INNER JOIN inventory i ON r.inventory_id = i.inventory_id
      ORDER BY p.payment_id;
```

PART E

Provide original SQL code in a text format that creates a trigger on the detailed table of the report that will continually update the summary table as data is added to the detailed table.

```
-- Trigger to update summary table when changes are made to detailed table
CREATE OR REPLACE FUNCTION summary_trigger()
RETURNS TRIGGER
LANGUAGE plpgsql
AS $$
BEGIN
	TRUNCATE summary_table;
	INSERT INTO summary_table
		SELECT store_id, month,
			COUNT(payment_id),
			SUM(amount)
	FROM detailed_table
	GROUP BY store_id, month
	ORDER BY store_id;
RETURN NULL;
END; $$;
```

```
CREATE TRIGGER update_summary
AFTER INSERT OR DELETE ON detailed_table
FOR EACH STATEMENT EXECUTE FUNCTION summary_trigger();
```

PART F

Provide an original stored procedure in a text format that can be used to refresh the data in *both* the detailed table and summary table. The procedure should clear the contents of the detailed table and summary table and perform the raw data extraction from part D.

```
-- Procedure to refresh data in both tables
CREATE OR REPLACE PROCEDURE tables_refresh()
LANGUAGE plpgsql
AS $$
BEGIN
	TRUNCATE detailed_table;
	INSERT INTO detailed_table
		SELECT
			i.store_id,
			p.payment_id,
			get_month(p.payment_date) AS month,
			p.payment_date::date AS payment_date,
			p.amount::money AS amount
	FROM payment AS p
```

```
      INNER JOIN rental AS r ON p.rental_id = r.rental_id
      INNER JOIN inventory AS i ON r.inventory_id = i.inventory_id
      ORDER BY p.payment_id DESC;
END; $$;


-- Call the procedure
CALL tables_refresh();
```

PART F1

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

A great tool for scheduling automation tasks with PostgreSQL is pgAgent. The ideal settings would have the reports update on the first day of the month, early in the morning so they are finished and available for management as soon as they arrive.

PART G

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

https://wgu.hosted.panopto.com/Panopto/Pages/Viewer.aspx?id=d9d4f295-4996-4858-b560-b2ae0187d5c6

PART 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 sources were used to support this submission.