

**Data Analysis**

**Samuel Diaz, #001361588**

**Western Governors University**

**03/22/2023**

**Plan for and compose the sections of a real-world business report that can be created from the "Labs on Demand Assessment Environment and DVD Database" web link, and demonstrate the functionality of the supporting SQL code by doing the following:**

**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.**

After analyzing the tables and fields included in the DVD Dataset database, I concluded that a good business solution that can be solved with this data is finding which customers produce the most revenue. This is of major concern to a business as it identifies prominent customers they should focus on and target as they are a major source of revenue for the company. This also ties back to the idea that it is easier to retain a current customer than attract a new one. Therefore, the business should focus on sustaining current customers that generate the most revenue and try to find ways to keep them fulfilled with their purchases with positive reinforcement tactics such as incentivizing each of their purchases. For example, the marketing team can use this information to reward those customers that have spent the greatest or least with different types of coupons to promote further spending.

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

The specific fields that will be included in the detailed table:

- Customer ID
- First Name
- Last Name
- Email Address
- Payment ID
- Amount
- Payment Date

The specific fields that will be included in the summary table:

- Customer Full Name
- Email Address
- Amount

**2. Describe the types of data fields used for the report.**

For this report, the fields in the two tables mentioned above will be using data types of integer, varchar, numeric, and timestamp. The integer data type species whole numbers and will be used for those fields holding only discrete numbers such as Customer ID and Payment ID. The varchar data type will be used for those fields holding multiple characters such as First Name, Last Name, and Email Address. The data type numeric specifies decimal values and will only be used for the field containing Amount which holds the payments made by customers. Lastly, time stamp will be the data field utilized for the column Payment Date which will be storing calendar date values.

**3. 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.**

Two prominent tables that will allow me to obtain the information required for both my detailed and summary tables are the 'customer' table and the 'payment table' located in the current DVD rental database.

**4. 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*).**

Two fields, first name and last name, from the detailed table will need be combined to provide the full name of a customer which will go into the summary section. This field requires transformation to

make it easier for the stakeholder accessing the data to understand which customer the given row belongs to. Concatenating first and last name further improves readability especially since the summary table is meant to be skimmed or quickly looked at.

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

The detailed table shows data for all of the customers' payments and transactions. Every single payment a customer has made will be displayed along with other relevant information such as when the payment was made and their email address. Given the extensive amount of data, this table will have, the detailed section will be used for marketing officials to distinguish the profitable vs the not-so-profitable customers. With this information, they can devise a way to target different segments of customers that are distinguished based on how much they are spending. For example, offering varying coupons to each distinct customer segment or creating a reward system targeted towards a specific segment of customers based on how much they are spending.

The summary table shows only three attributes: name, email, and amount (total payments made in a lifetime). This table is short, brief, and concise. Therefore, this table is designed to be skimmed over and not given much attention to detail. In a business environment, this would happen in events such as a meeting or when holding it to potential stakeholders who want to know important information about the company's revenues when it ties to customers.

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

Given that the business and the database itself aren't extraordinarily massive, my report should refresh at the beginning of each month which is when most accounting and finance reports are conducted to obtain data. Also, anyone attempting to retrieve data relevant to their situation should refresh the database as they see convenient. For example, if it's the middle of the month but a businessperson wants knowledge on the current customer with the most payments, they should refresh the dataset to obtain accurate information.

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

```
CREATE FUNCTION trigger_function()
  RETURNS TRIGGER
  LANGUAGE PLPGSQL
AS $$
BEGIN

  TRUNCATE TABLE summary;

INSERT INTO summary(
  SELECT
    CONCAT(first_name, ' ', last_name) AS full_name,
    email,
    SUM(amount)
  FROM detailed
  GROUP BY
    customer_id,
    full_name,
    email
  ORDER BY SUM(AMOUNT) DESC
);
```

```
END;  
$$
```

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

```
CREATE TABLE detailed (  
    customer_id    integer,  
    first_name     varchar(50),  
    last_name      varchar(50),  
    email          varchar(100),  
    payment_id     integer,  
    amount         numeric,  
    payment_date   timestamp  
);
```

```
CREATE table summary (  
    full_name      varchar(101),  
    email          varchar(100),  
    amount         numeric  
);
```

**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.**

```
INSERT INTO detailed (  
    customer_id,  
    first_name,  
    last_name,  
    email,  
    payment_id,  
    amount,  
    payment_date  
  
)
```

```
SELECT  
    c.customer_id,  
    c.first_name,  
    c.last_name,  
    c.email,  
    p.payment_id,  
    p.amount,  
    p.payment_date
```

```
FROM payment AS p  
INNER JOIN customer AS c  
ON c.customer_id = p.customer_id;
```

**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.**

```
CREATE TRIGGER trigger_function  
    AFTER INSERT ON detailed  
    FOR EACH STATEMENT
```

```
EXECUTE PROCEDURE trigger_function();
```

**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.**

```
CREATE PROCEDURE refresh_tables()  
LANGUAGE PLPGSQL  
AS $$  
BEGIN
```

```
TRUNCATE TABLE detailed;
```

```
INSERT INTO detailed
```

```
SELECT  
    c.customer_id,  
    c.first_name,  
    c.last_name,  
    c.email,  
    p.payment_id,  
    p.amount,  
    p.payment_date
```

```
FROM payment AS p  
INNER JOIN customer AS c  
ON c.customer_id = p.customer_id;
```

```
END;  
$$
```

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

A relevant job scheduling tool that can be used to automate the store PostgreSQL procedure is Windows Task Scheduler.

**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.**

- Website: <https://stackoverflow.com/questions/22290256/how-to-execute-a-postgresql-procedure-through-windows-task-scheduler>
- Website: <https://www.geeksforgeeks.org/difference-between-function-and-procedure/#>