Madison Gish
C000745675
D191 Advanced Data Management
Performance Assessment

Business Report

Summarize one real-world business report that can be created from the attached Data Sets and Associated Dictionaries.

With the provided data sets and associated dictionaries, there is a multitude of different business reports that could be created, one of those revolving around returning customers. It is well known that keeping a returning customer is always easier and more cost-effective than gaining a new customer. Knowing which customers return and how often they do could lead to the implementation of a returning customer program that will reward the customer for continuing to rent movies from our stores and incentivize other customers to rent more often than they currently do. This will help the business's rental sale growth and could potentially lead to a rise in new customers wanting to get in on our program.

Describe the data used for the report.

The data used for this report would be the customer information as well as the rental information. The customer information would be needed to provide the stakeholder with information about the returning customer. They would be able to decide what to do with the customers that return the most.

The rental information would be used in the report as that would provide the stakeholders with the information as to when the customers rent out DVDs. Decisions could then be made for the customer who returns often and recently and looks at the customers that have not returned in some time.

Identify *two* or more specific tables from the given dataset that will provide the data necessary for the detailed and summary sections of the report.

The tables from the provided dataset that will be needed for both the detailed and the summary sections of the report would be that of the rental table and the customer table. Both of those combined will give us all the information that we need to see which customers return to our stores for DVD rentals in order to implement a returning customer program.

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

For the detailed section of the report the following specific fields would need to be included:
- Rental ID
- Rental date
- Return date
- Staff ID
- Customer ID
- Customer-first Name
- Customer last name
- Email

For the summary section of the report the following specific fields would need to be included:
- Customer Full Name
- Email
- Customer count

Identify one field that will require a custom transformation and explain why it should be transformed. For example, you might translate a field with a value of 'N' to 'No' and 'Y' to 'Yes'.

One field found within the detailed section that will need to be transformed are the 'first_name' and "last_name' fields. When initially extracted from the customer table both names, first and last will be split into a first name column and a last name column. A transformation would be required to give this field a little more readability. Concatenating the two names together to a format 'last, first' would more easily let the stakeholders know the full name of the customer.

Explain the different business uses of the detailed and the summary sections of the report.

The Detailed section of the report serves this purpose storing every transaction within the company regarding rental sales. This report will be good for the stakeholders to dive deeper into who is the top customers and when they did support the business.

The summary is a good quick representation of what is going on regarding who the customer is and how often they have supported the business. During a quick update meeting, the summary section would be the best place to find the most important information needed to make returning customer decisions.

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

This report should be refreshed at least every month to review any new sales made within the month and see if we are still allocating the proper rewards to the proper customers. If the business were to award coupons to the top 50 returning customers, the report should be refreshed prior to awarding the coupons.

Explain how the stored procedure can be run on a schedule to ensure data freshness.

The stored procedure for this data can be run whenever it is needed, which should be monthly at a minimum. Without re-running the report, the old information will still be present which will not provide the business with an accurate reading on what is going on in today's rentals. Before the business meeting, the procedure to update the information for both the detailed and the summary tables shall be executed to ensure data freshness.

Report Queries

Creating Tables - The SQL code that creates the tables to hold the report sections is correct and complete.

```
--CREATE DETAILED TABLE

DROP TABLE IF EXISTS detailed;

CREATE TABLE detailed (

        customer_id integer,

        first_name varchar (45),

        last_name varchar (45),

        email varchar(90),

        rental_id integer,

        rental_date timestamp,

        return_date timestamp,

        staff_id integer

);

-- This creates an empty table titled detailed that will hold the information for the

-- detailed section of the business report


--CREATE SUMMARY TABLE


CREATE TABLE summary(

        customer_name varchar(95),

        email varchar(90),

        customer_count integer


);

-- This creates an empty table titled summary that will hold the information for the

-- summary section of the business report
```

SQL Query – The SQL query provided correctly queries the raw data needed for the Detailed section of the report from the source database and verifies the data's accuracy.

```
--INSERT DATA INTO DETAILED TABLE
INSERT INTO detailed(
	customer_id, -- customer
	first_name, -- customer
	last_name, -- customer
	email, -- customer
	rental_id, -- rental
	rental_date, -- rental
	return_date, -- rental
	staff_id  --rental
)
SELECT
	c.customer_id, c.first_name, c.last_name, c.email,
	r.rental_id, r.rental_date, r.return_date, r.staff_id
FROM rental AS r
INNER JOIN customer AS c ON c.customer_id = r.customer_id;

--- This will load the information from the customer and rental tables into the detailed table,
```

Functions – The code for the function(s) that perform the transformation(s) identified in part A4 produced the expected transformations and *all* previously identified functions are included in the submission

```
--CREATE FUNCTION
CREATE FUNCTION summary_refresh_fun()
RETURNS TRIGGER
LANGUAGE plpgsql
AS $$
BEGIN
```

```
DELETE FROM summary;
-- this will empty the summary table.

INSERT INTO summary (
	SELECT
		concat_ws (', ', last_name, first_name) AS customer_name,
		email,
		COUNT(customer_id)
	FROM detailed
	GROUP BY customer_id, customer_name, email
	-- HAVING count(customer_id) > 30
	ORDER BY count(customer_id)DESC
	LIMIT 100
);

RETURN NEW;
END; $$--This function will refresh the summary table with data taken from the detailed table
```

Triggers – The SQL code successfully 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
CREATE TRIGGER summary_refresh
AFTER INSERT ON detailed
FOR EACH STATEMENT
EXECUTE PROCEDURE summary_refresh_fun();
--This trigger will execute the procedure once there is a insert of data into the detailed table.
```

Stored Procedure - The stored procedure provided refreshes the data in *both* the detailed and summary tables, and it can clear the contents of the detailed and summary tables and perform the ETL load process from part C. The submission includes comments that identify how often the stored procedure should be executed.

```
CREATE PROCEDURE refresh_tables()

LANGUAGE plpgsql

AS $$

BEGIN


DELETE FROM detailed; -- this will empty the detailed table on any existing info.

INSERT INTO detailed(

        customer_id, -- customer

        first_name, -- customer

        last_name, -- customer

        email, -- customer

        rental_id, -- rental

        rental_date, -- rental

        return_date, -- rental

        staff_id  --rental)

SELECT

        c.customer_id, c.first_name, c.last_name, c.email,

        r.rental_id, r.rental_date, r.return_date, r.staff_id

FROM rental AS r

INNER JOIN customer AS c ON c.customer_id = r.customer_id;

--reinserting the new data into the detailed table

END;$$

-- To call stored procedure

CALL refresh_tables();

-- To view results

SELECT *

        FROM detailed;

SELECT *

        FROM summary;
```

Web Sources -The record the web sources used to acquire data or segments of third-party code to support the application is both complete and accurate, and the web sources cited are reliable. Or the candidate did not use *any* web sources to acquire data or segments of third-party code and stated this in their submission.

I did not use any external web sources to acquire data or segments of third-party code.