University Of St. Thomas

Course 732 - Data Stores and Feature Design

Project – Amazon Product Database

Project by – Mugdha Dixit

1. Using the marketplace locate one or more free data sets that interest you. (5 pts)
    - **TRAJECT_DATA_AMAZON_PRODUCT_RESULTS_DATA**

2. Worksheet #1. Build a curation layer. (20 pts)

Worksheet 1

Created new schema named Amazon Schema

```
USE ROLE TRAINING_ROLE;
CREATE WAREHOUSE IF NOT EXISTS BULLFROG_WH;
USE WAREHOUSE BULLFROG_WH;
CREATE DATABASE IF NOT EXISTS BULLFROG_DB;
USE BULLFROG_DB.PUBLIC;

CREATE OR REPLACE SCHEMA Amazon_Schema;

USE SCHEMA Amazon_Schema;
```

Creating tables and adding data into it.

Created table named ‘CUR_Search_Product’ and added data from

```
CREATE OR REPLACE TABLE CUR_Search_Product (
    P_id VARCHAR(5000),
    P_title VARCHAR(5000),
    P_rating FLOAT,
    P_link VARCHAR(5000),
    P_keyword VARCHAR(5000),
    P_coupon VARCHAR(5000),
    P_img VARCHAR(5000),
    Is_bundle BOOLEAN
);
```

Pulling data from ‘Amazon_Product’ database to the ‘CUR_Search_Product’ table.

```
INSERT INTO CUR_Search_Product (P_id, P_title, P_rating, P_link,
P_keyword,P_coupon, P_img, Is_bundle)
SELECT SEARCH_RESULTS_PRODUCT_ASIN, SEARCH_RESULTS_PRODUCT_TITLE,
SEARCH_RESULTS_PRODUCT_RATING, SEARCH_RESULTS_PRODUCT_LINK,
SEARCH_RESULTS_PRODUCT_KEYWORDS,
SEARCH_RESULTS_PRODUCT_COUPON_TEXT,
SEARCH_RESULTS_PRODUCT_MAIN_IMAGE_LINK, SEARCH_RESULTS_PRODUCT_IS_BUNDLE
FROM TRAJECT_DATA_AMAZON_PRODUCT_RESULTS_DATA.INTERNAL.AMAZON_PRODUCT;
```

Creating table ‘Category’ and adding data to the table.

```
CREATE OR Replace TABLE CUR_Category (
    C_id VARCHAR(5000),
    P_id VARCHAR(5000),
    C_name VARCHAR(5000),
    C_link VARCHAR(5000),
    C_fLat VARCHAR(5000)
);
```

```
INSERT INTO CUR_Category (C_id, P_id , C_name, C_link, C_fLat)
SELECT SEARCH_RESULTS_PRODUCT_CATEGORIES_1_CATEGORY_ID,
SEARCH_RESULTS_PRODUCT_ASIN, SEARCH_RESULTS_PRODUCT_CATEGORIES_0_NAME,
SEARCH_RESULTS_PRODUCT_CATEGORIES_1_LINK, SEARCH_RESULTS_PRODUCT_CATEGORIES_FLAT
FROM TRAJECT_DATA_AMAZON_PRODUCT_RESULTS_DATA.INTERNAL.AMAZON_PRODUCT;
```

Creating table ‘Buyer’ and adding data to the table.

```
CREATE OR Replace TABLE CUR_Buyer (
    B_id VARCHAR(5000),
    C_id VARCHAR(5000),
    B_ratings FLOAT,
    B_price VARCHAR(5000),
    B_currency VARCHAR(5000)
);
```

```
INSERT INTO CUR_Buyer (B_id, C_id, B_ratings, B_price, B_currency)
SELECT SEARCH_RESULTS_ALSO_BOUGHT_0_ASIN,
SEARCH_RESULTS_PRODUCT_CATEGORIES_1_CATEGORY_ID, SEARCH_RESULTS_ALSO_BOUGHT_0_RATING,
SEARCH_RESULTS_ALSO_BOUGHT_0_PRICE_VALUE, SEARCH_RESULTS_ALSO_BOUGHT_0_PRICE_CURRENCY
FROM TRAJECT_DATA_AMAZON_PRODUCT_RESULTS_DATA.INTERNAL.AMAZON_PRODUCT;
```

Enhancing the data with additional fields.

Adding new column to the ‘Product’ table and inserting values, 'High', 'Medium' and 'Low' based on ‘P_rating’.

```
ALTER TABLE CUR_Search_Product
ADD COLUMN rating_category VARCHAR(20);
```

```
UPDATE CUR_Search_Product
SET rating_category = CASE
                        WHEN P_rating >= 4.8 THEN 'High'
                        WHEN P_rating >= 4.6 THEN 'Medium'
                        ELSE 'Low'
                      END;
```

Identifying missing values in Product table

```
SELECT * FROM CUR_Search_Product WHERE P_rating IS NULL;
```

Adding new column 'has_coupon' to ‘Product’ table and adding Yes/No indicators.

```
UPDATE CUR_Search_Product
SET has_coupon = CASE
                    WHEN P_coupon IS NOT NULL THEN TRUE
                    ELSE FALSE
                 END;
```

Calculating the average rating from non-null values in the ‘B_ratings’ column.

```
SELECT AVG(B_ratings) FROM CUR_Buyer WHERE B_ratings IS NOT NULL;
```

Identifying the duplicate values.

```
SELECT C_id, COUNT(*) AS count_duplicates
FROM CUR_Category
GROUP BY C_id
HAVING COUNT(*) > 1;
```

Finding number of null values in ‘CUR_Buyer’ table.

```
SELECT B_ratings FROM CUR_Buyer WHERE B_ratings IS NULL;
```

Updating null values in ‘B_ratings’ with the average rating

```
UPDATE CUR_Buyer
SET B_ratings = (SELECT AVG(B_ratings)
FROM CUR_Buyer WHERE B_ratings IS NOT NULL)
WHERE B_ratings IS NULL;
```

Joining CUR_Search_Product with CUR_Category to get details from both tables.

```
SELECT SP.P_id,SP.P_rating,
       CC.C_name, CC.C_link
FROM CUR_Search_Product SP
JOIN CUR_Category CC ON SP.P_id = CC.P_id;
```

Worksheet 2

Build an aggregation layer.

Creating new schema named 'AGG_Schema'.

```
CREATE OR REPLACE SCHEMA AGG_Schema;

USE SCHEMA AGG_Schema;
USE SCHEMA Amazon_Schema;
```

Creating different views using different types of aggregation.

Counting number of products with Coupons from CUR_Search_Product table

```
CREATE OR REPLACE VIEW NoOfProd_WithCoupons AS
SELECT COUNT(*) AS total_products_with_coupons
FROM Amazon_Schema.CUR_Search_Product
WHERE P_coupon IS NOT NULL;
```

Calculating the total purchase amount for each customer from ‘CUR_Buyer’ table.

```
CREATE OR REPLACE VIEW Total_PurchaseAmount_PerCustomer AS
SELECT B_id, SUM(B_price) AS total_purchase_amount
FROM Amazon_Schema.CUR_Buyer
GROUP BY B_id;
```

Counting products in each category from CUR_Category table

```
CREATE OR REPLACE VIEW Total_Products AS
SELECT C_name, COUNT(*) AS num_products
FROM Amazon_Schema.CUR_Category
GROUP BY C_name;
```

Finding the maximum rating among products and the corresponding product title using CUR_Search_Product table.

```
CREATE OR REPLACE VIEW Max_Ratings AS
SELECT P_title, P_rating
FROM Amazon_Schema.CUR_Search_Product
WHERE P_rating = (SELECT MAX(P_rating)
FROM Amazon_Schema.CUR_Search_Product);
```

Worksheet 3

Creating a table function from aggregated view.

```
CREATE OR REPLACE FUNCTION products_in_category ( Category_Name VARCHAR )
  RETURNS TABLE ( Category_Name VARCHAR, Num_Products INTEGER )
  AS 'SELECT C_name AS Category_Name, num_products AS Num_Products
        FROM Total_Products '


SELECT * FROM TABLE ( products_in_category( ))
```

Worksheet 4

Creating a stored procedure.

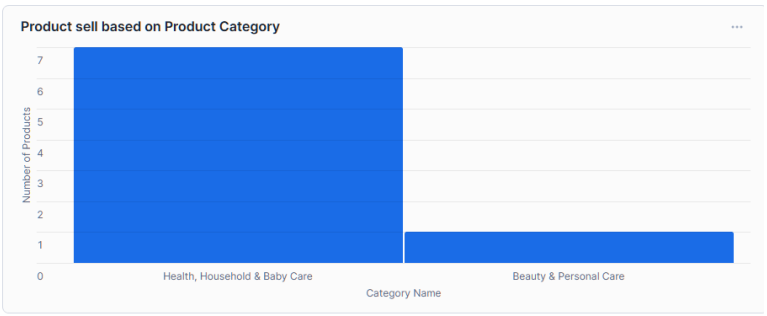
```
CREATE OR REPLACE PROCEDURE GetTop10Products()
  RETURNS TABLE (P_id VARCHAR, P_title VARCHAR, P_rating FLOAT,
  P_link VARCHAR, P_keyword VARCHAR, P_coupon VARCHAR, P_img VARCHAR)
  AS
  DECLARE
  res Resultset DEFAULT(SELECT P_id, P_title, P_rating, P_link, P_keyword, P_coupon, P_img
    FROM CUR_Search_Product
    LIMIT 10 );
  BEGIN
    RETURN TABLE(res);
  END

CALL GetTop10Products();
```

Dashboard

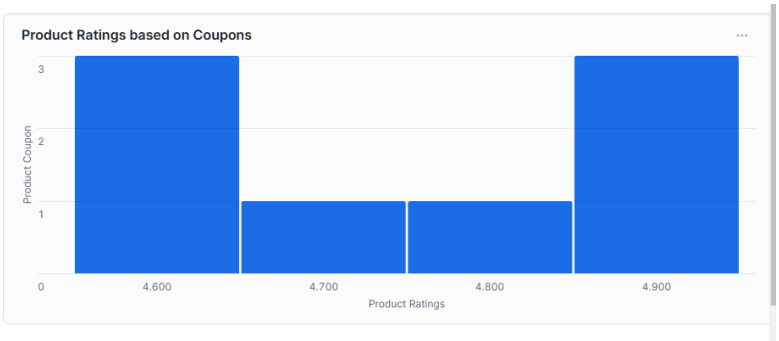
Result generated by function using aggregated view. 2 rows

CATEGORY_NAME	NUM_PRODUCTS
Health, Household & Baby Care	7
Beauty & Personal Care	1



Result generated from stored procedure 8 rows

P_ID	P_TITLE	P_RATING	P_LINK
B09KXTPQ94	Dreft Stage 1: Newborn Baby Liquid Laundry Detergent, 89 loads 128 fl oz, 1 Choice	4.9	https://ww
B09KXTY82H	Dreft Stage 2: Active Baby Liquid Laundry Detergent, 89 Loads, 128 fl oz, Helps rem	4.9	https://ww
B01MTVHCTV	Mrs. Meyer's Baby Laundry Detergent Liquid, Infused with Essential Oils, Baby Blossc	4.7	https://ww
B09PPQQMZM	ARM & HAMMER Baby, 75 Loads Liquid Laundry Detergent, 118.1 Fl oz	4.6	https://ww
B07FVT99YM	Seventh Generation Concentrated Baby Laundry Detergent, Stain Fighting Formula, F	4.6	https://ww
B074QZRCTZ	Noodle & Boo Baby Laundry Essentials Ultra-Safe Laundry Detergent Creme Douce 3	4.6	https://ww
B00T2CFQUQ	Babyganics 3X Baby Laundry Detergent, Fragrance Free, 60oz, Packaging May Vary	4.8	https://ww
B09KXTPQ94	Dreft Stage 1: Newborn Baby Liquid Laundry Detergent, 89 loads 128 fl oz, 1 Choice	4.9	https://ww



Worksheet 5

A. Provide the name and description of the data set you chose.

Name of the Dataset: Traject Data: Amazon Product Results Data
Description:
The Traject Data: Amazon Product Results dataset provides real-time information on Amazon product listings, offering detailed insights such as brand details, specifications, seller information, imagery, videos, deals, and more. This dataset enables accurate monitoring of product listings, price trends, market research, and competitor intelligence, supporting various business functions across industries.

B. Explain in 1-2 sentences what your naming convention is and what your schemas are so the instructor can locate them quickly.

I created three different tables in the curation layer as Product, Category and Buyer.

All the tables names are named starting with 'CUR'.

C. Briefly explain the logic/formulas used for any custom fields you create in your curation.
layer (This is a mini data catalog)

- I used the Amazon product dataset to establish three distinct tables aimed at analyzing product sales based on various factors.
- Firstly, I structured a schema and populated data from the Amazon database to newly created tables.
- Initially the database I selected had a single table with multiple columns. I derived three separate tables.
- While addressing null values within the dataset, I removed irrelevant entries and incorporated average values into certain columns using the 'AVG' aggregation function.
- In next worksheet, using SUM, COUNT, AVG, and MAX, I crafted views to showcase key insights:

    - Total products popularized through coupons.
    - Total purchase amounts per customer.
    - Product categorization.
    - Maximum product rating with its title.

- From these views, I developed a function in the third worksheet to retrieve category names and product counts based on search criteria.
- In worksheet four, I crafted a stored procedure and created dashboard. The dashboard featured four tiles,
- Following two visualizations were shown in the dashboard:
    - Product sales based on Product Category
    - Product Ratings based on product category.