

SQL Challenge: Identifying Customer Retention Trends

Question:

A retail company wants to analyze customer retention by identifying customers who made purchases in **both 2022 and 2023**. Additionally, they want to find out **which product category contributed the most revenue for each retained customer in 2023**.

Write an SQL query to:

1. Identify customers who made purchases in **both 2022 and 2023**.
2. For each retained customer, determine **the product category that contributed the most revenue in 2023**.
3. If there is a tie in revenue contribution, select the category with the **higher total quantity sold**.
4. Display the output in the following format:
 - customer_name
 - city
 - top_category_2023 (category that contributed the most revenue in 2023 for that customer)
 - total_revenue_2023 (revenue from that category)
 - total_quantity_2023 (total quantity sold for that category)

Table Definitions:

ORDERS
order_id
customer_id
order_date

CUSTOMERS
customer_id
customer_name
city

CATEGORIES
category_id
category_name

ORDER_ITEMS
order_id
product_id
quantity
price_per_unit

PRODUCTS
product_id
category_id
product_name

Notes:

- A **retained customer** is someone who made at least **one** purchase in both 2022 and 2023.
- Revenue is calculated as **price_per_unit × quantity**.
- If a customer did not make any purchase in 2023, they should **not** be included in the final result.
- If a customer only made purchases in 2022 but not in 2023, they should **not** be included.
- The results should be **ordered alphabetically** by customer_name.

**** Query:**

// Creating a Base table with all the required fields

```
WITH BASE AS(  
  SELECT O.*,  
         OI.PRODUCT_ID, OI.QUANTITY, OI.PRICE_PER_UNIT,  
         C.CUSTOMER_NAME, C.CITY,  
         P.CATEGORY_ID, P.PRODUCT_NAME,  
         CG.CATEGORY_NAME  
  FROM ORDERS AS O  
  LEFT JOIN ORDER_ITEMS AS OI  
  ON O.ORDER_ID = OI.ORDER_ID  
  LEFT JOIN CUSTOMERS AS C  
  ON O.CUSTOMER_ID = C.CUSTOMER_ID  
  LEFT JOIN PRODUCTS AS P  
  ON OI.PRODUCT_ID = P.PRODUCT_ID  
  LEFT JOIN CATEGORIES AS CG  
  ON P.CATEGORY_ID = CG.CATEGORY_ID  
)
```

// Creating a table to get all the Retained Customers - PART-1

```
,RETAINED_CUST_BASE AS(  
  SELECT CUSTOMER_ID, YEAR, COUNT_ORDERS,  
         LEAD(COUNT_ORDERS) OVER (PARTITION BY CUSTOMER_ID ORDER BY YEAR) AS  
  NXT_YR_COUNT_ORDERS  
  FROM  
  (  
    SELECT CUSTOMER_ID, YEAR(ORDER_DATE) AS YEAR, COUNT(ORDER_ID) AS COUNT_ORDERS  
    FROM BASE  
    WHERE YEAR(ORDER_DATE) IN (2022,2023)  
    GROUP BY CUSTOMER_ID  
  ) AS SUB_TABLE_1  
)
```

// Creating a table to get all the Retained Customers - PART-2

```
,RETAINED_CUST_ACTUAL AS(
SELECT CUSTOMER_ID,
CASE WHEN COUNT_ORDERS >= 1 AND NXT_YR_COUNT_ORDERS >= 1 THEN "RETAINED"
ELSE "NOT RETAINED"
END AS RETAINED_STATUS
FROM RETAINED_CUST_BASE
)
```

// Calculating highest revenue generating Category for the "Retained" customers in "2023"

```
,CATEGORY_REVENUE AS(
SELECT *,
SUM(QUANTITY * PRICE_PER_UNIT) OVER (PARTITION BY CUSTOMER_ID, CATEGORY_ID ORDER BY
YEAR(ORDER_DATE)) AS CUST_CG_REVENUE,
SUM(QUANTITY) OVER (PARTITION BY CUSTOMER_ID, CATEGORY_ID ORDER BY YEAR(ORDER_DATE)) AS
CUST_CG_ORDERS
FROM BASE
WHERE CUSTOMER_ID IN ( SELECT CUSTOMER_ID
FROM RETAINED_CUST_ACTUAL
WHERE RETAINED_STATUS = "RETAINED")
AND YEAR(ORDER_DATE) = 2023
)
```

```
SELECT CUSTOMER_NAME, CITY, CATEGORY_NAME AS TOP_CATEGORY_2023, CUST_CG_REVENUE AS
TOTAL_REVENUE_2023, CUST_CG_ORDERS AS TOTAL_QUANTITY_2023
FROM (
SELECT *,
ROW_NUMBER() OVER (PARTITION BY CUSTOMER_ID ORDER BY CUST_CG_REVENUE DESC,
CUST_CG_ORDERS DESC) AS RN
FROM CATEGORY_REVENUE
) AS SUB_TABLE_2
WHERE RN = 1
;
```