Histogram of Users and Purchases (Walmart)

QUESTION

Assume you are given the table on Walmart user transactions. Based on a user's most recent transaction date, write a query to obtain the users and the number of products bought.

Output the user's most recent transaction date, user ID and the number of products sorted by the transaction date in chronological order.

P.S. As of 10 Nov 2022, the official solution was changed from output of the transaction date, number of users and number of products to the current output.

`user_transactions` Table:

Column Name	Type
product_id	integer
user_id	integer
spend	decimal
transaction_date	timestamp

`user_transactions` Example Input:

product_id	user_id	spend	transaction_date
3673	123	68.90	07/08/2022 12:00:00
9623	123	274.10	07/08/2022 12:00:00
1467	115	19.90	07/08/2022 12:00:00
2513	159	25.00	07/08/2022 12:00:00
1452	159	74.50	07/10/2022 12:00:00

Step 1: Identify the problem of the case

Step 2 : Analyze and solve problems

First get the most recent date of the customer who purchased the product with the `MAX()` function of each `user_id` :

```
SELECT user_id, MAX(transaction_date) recent_date
FROM user_transactions
GROUP BY user_id
```

Create the obtained table and save it as a CTE.

```
WITH rc_tbl AS (SELECT user_id, MAX(transaction_date) recent_date
                  FROM user_transactions
                  GROUP BY user_id)
```

From the newly created table, we merge with the original data table to push the necessary information to fulfill the request of the case. Then use the `COUNT()` function to count the number of products purchased by each customer.

```
WITH rc_tbl AS (SELECT user_id, MAX(transaction_date) recent_date
                  FROM user_transactions
                  GROUP BY user_id)

SELECT u.transaction_date
      , u.user_id
      , COUNT(u.product_id) purchase_count
FROM user_transactions u
JOIN rc_tbl r
  ON u.user_id = r.user_id
    AND u.transaction_date = r.recent_date
GROUP BY u.transaction_date, u.user_id
```

OUTPUT:

transaction_date	user_id	count
07/11/2022 10:00:00	123	1
07/12/2022 10:00:00	115	1
07/12/2022 10:00:00	159	2

Another Solution:

```
WITH latest_transaction AS (
  SELECT
    transaction_date,
    user_id,
    product_id,
    RANK() OVER (PARTITION BY user_id
      ORDER BY transaction_date DESC) AS days_rank
  FROM user_transactions)

SELECT
  transaction_date,
  user_id,
  COUNT(product_id) AS purchase_count
FROM latest_transaction
WHERE days_rank = 1
GROUP BY transaction_date, user_id
ORDER BY transaction_date;
```

or

```
SELECT
  transaction_date,
  user_id,
  COUNT(product_id) AS purchase_count
FROM (
  SELECT
    transaction_date,
    user_id,
    product_id,
    RANK() OVER (
      PARTITION BY user_id
```

```
      ORDER BY transaction_date DESC) AS days_rank
  FROM user_transactions) AS latest_transaction
WHERE days_rank = 1
GROUP BY transaction_date, user_id
ORDER BY transaction_date;
```