SQL OF THE DAY

💻 FINDING USER PURCHASES

NIVAN R. SUGIANTORO

PROBLEMS

🔗 https://platform.stratascratch.com/coding/10322-finding-user-purchases?code_type=1

Problem:
Identify returning active users by finding users who made a second purchase within 1 to 7 days after their first purchase. Ignore same-day purchases. Output a list of these user_ids.

SOLUTION

🧩 Logic breakdown

- Rank purchases per user

Use row_number() to order purchases chronologically for each user.

- Extract first and second purchase

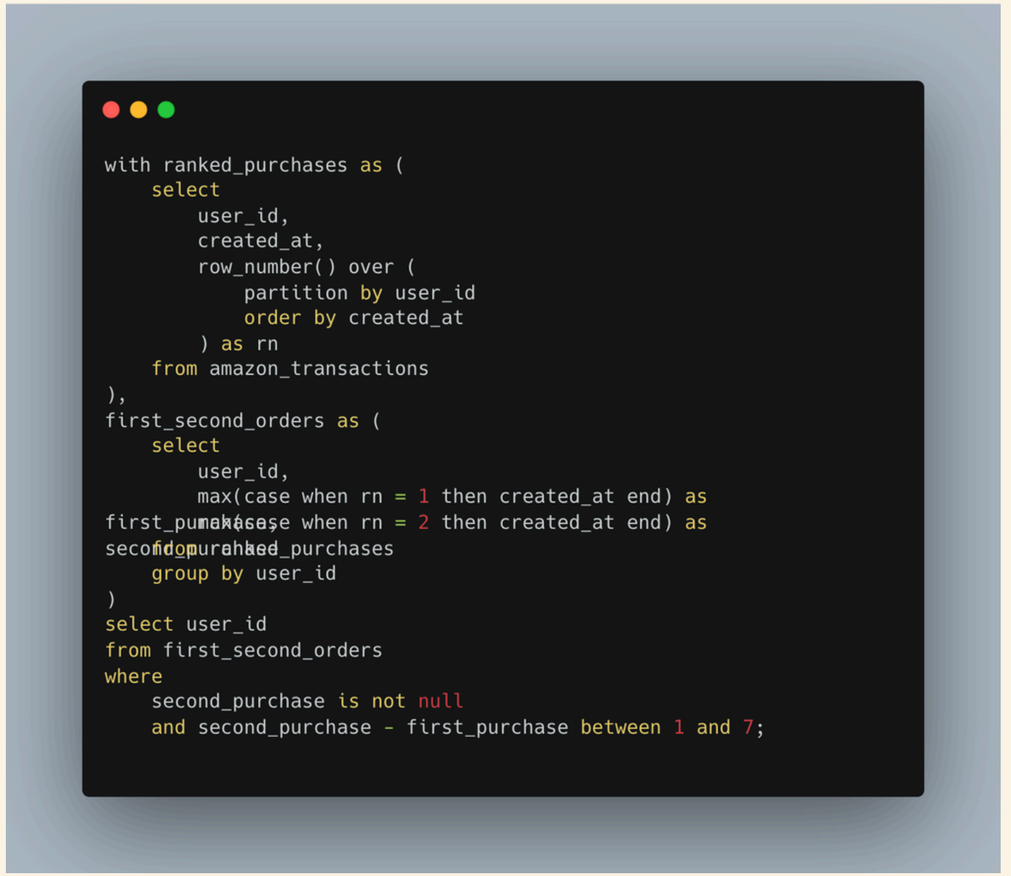
Use case when to mark the first and second purchase. max() is used to safely return the non-null timestamp per user.

- Filter returning users

Keep users where:

a. a second purchase exists

b. the second purchase happens 1–7 days after the first



```sql
with ranked_purchases as (
    select
        user_id,
        created_at,
        row_number() over (
            partition by user_id
            order by created_at
        ) as rn
    from amazon_transactions
),
first_second_orders as (
    select
        user_id,
        max(case when rn = 1 then created_at end) as
first_purchase,
        max(case when rn = 2 then created_at end) as
second_purchase
    from ranked_purchases
    group by user_id
)
select user_id
from first_second_orders
where
    second_purchase is not null
    and second_purchase - first_purchase between 1 and 7;
```

KEY TAKEAWAYS

- Rank purchases per user

Use row_number() to order purchases chronologically for each user.

- Extract first and second purchase

Use case when to mark the first and second purchase.

max() is used to safely return the non-null timestamp per user.

- Filter returning users

Keep users where:

a. a second purchase exists

b. the second purchase happens 1–7 days after the first

Let's Connect

nivanrs@gmail.com