EY INTERVIEW LOGIC: GAP & ISLAND

Gap & Island SQL Problem

INPUT TABLE

EVENT	DATE
Success	01-01-2020
Success	02-01-2020
Fail	03-01-2020
Fail	04-01-2020
Success	05-01-2020

→

EXPECTED OUTPUT

EVENT	START	END
Success	01-01	02-01
Fail	03-01	04-01
Success	05-01	05-01

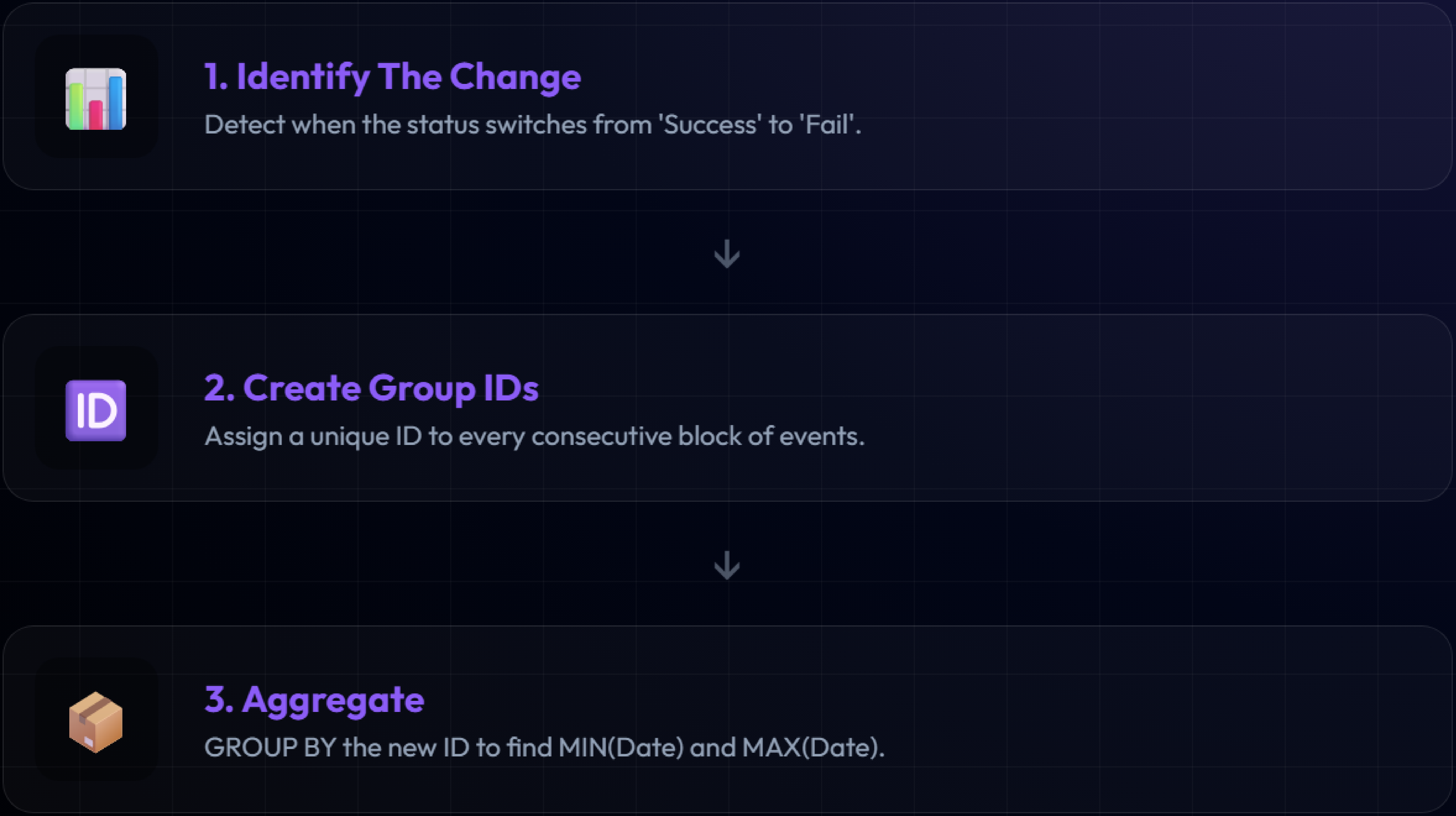
✓ CONSOLIDATE CONSECUTIVE ROWS

FOLLOW FOR MORE SQL PATTERNS

Anandnarayanan S

The Logic Map

How to solve Gap & Island problems



1. Identify The Change

Detect when the status switches from 'Success' to 'Fail'.

↓

2. Create Group IDs

Assign a unique ID to every consecutive block of events.

↓

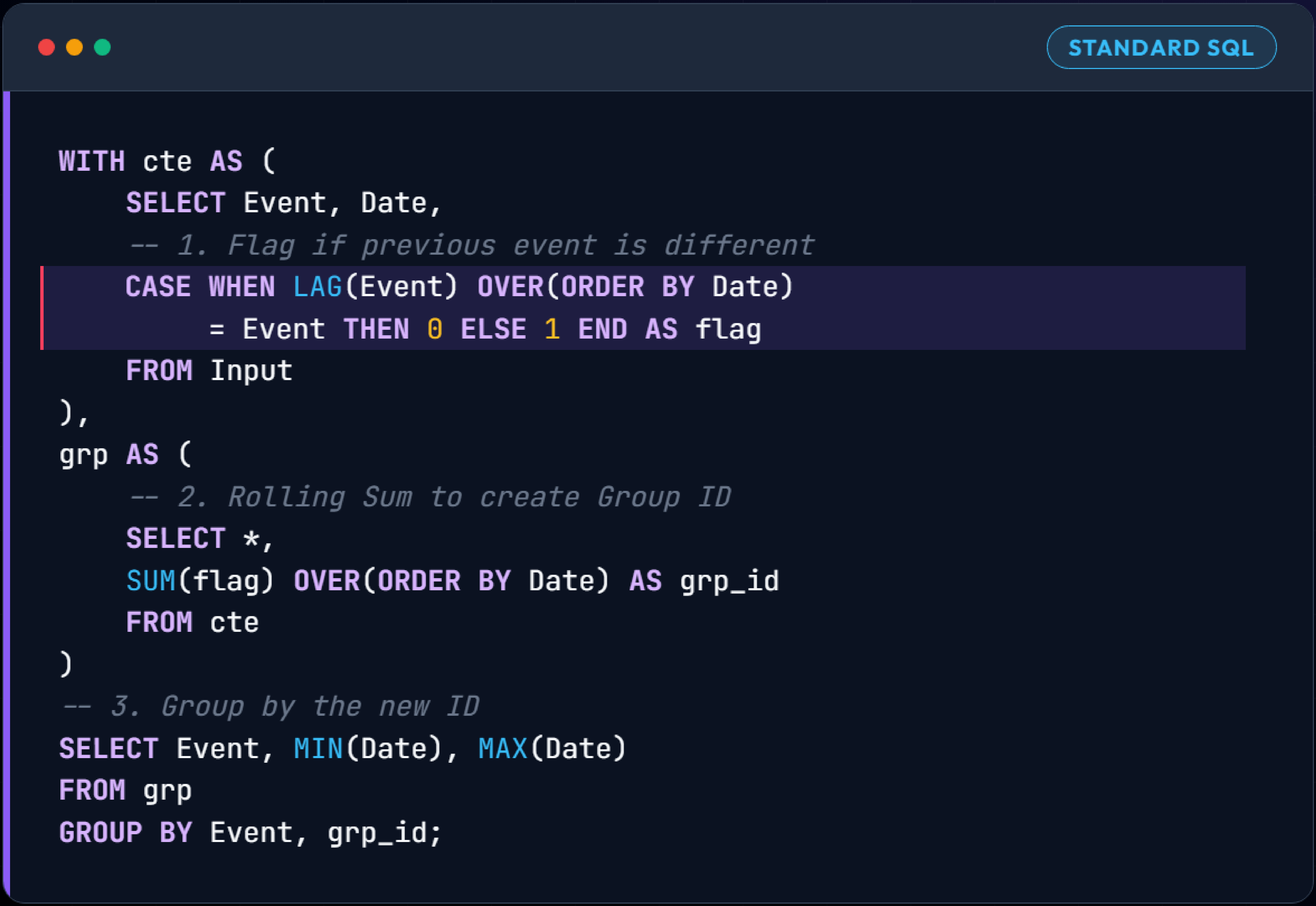
3. Aggregate

GROUP BY the new ID to find MIN(Date) and MAX(Date).

METHOD 1

The LAG Approach

Intuitive and easy to debug



```sql
WITH cte AS (
    SELECT Event, Date,
    -- 1. Flag if previous event is different
    CASE WHEN LAG(Event) OVER(ORDER BY Date)
         = Event THEN 0 ELSE 1 END AS flag
    FROM Input
),
grp AS (
    -- 2. Rolling Sum to create Group ID
    SELECT *,
    SUM(flag) OVER(ORDER BY Date) AS grp_id
    FROM cte
)
-- 3. Group by the new ID
SELECT Event, MIN(Date), MAX(Date)
FROM grp
GROUP BY Event, grp_id;
```

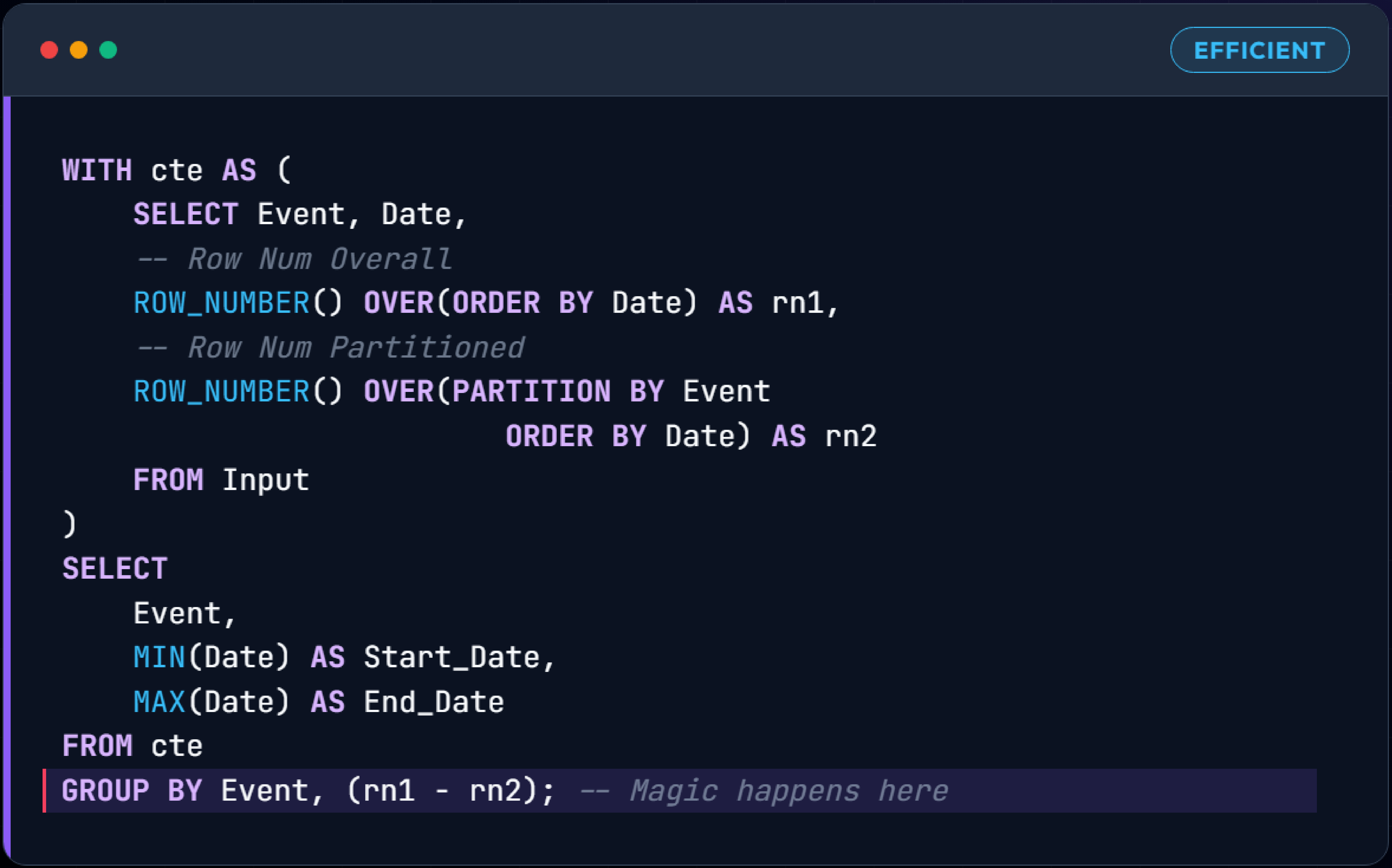
FOLLOW FOR MORE SQL PATTERNS

Anandnarayanan S

PRO TIP

The Math Trick

Using Row_Number differences



```sql
WITH cte AS (
    SELECT Event, Date,
    -- Row Num Overall
    ROW_NUMBER() OVER(ORDER BY Date) AS rn1,
    -- Row Num Partitioned
    ROW_NUMBER() OVER(PARTITION BY Event
                         ORDER BY Date) AS rn2
    FROM Input
)
SELECT
    Event,
    MIN(Date) AS Start_Date,
    MAX(Date) AS End_Date
FROM cte
GROUP BY Event, (rn1 - rn2); -- Magic happens here
```

💡 The difference (rn1 - rn2) remains constant for consecutive rows of the same event!

FOLLOW FOR MORE SQL PATTERNS

Anandnarayanan S

VS

Which is Better?

Comparing the two approaches

🛠️ LAG Method

- ✅ **Intuitive:** Reads like logical steps.
- ✅ **Safe:** Handles NULLs well.
- ❌ **Complex:** Requires 2 CTEs (Flag + Sum).

🧮 Row_Number

- ✅ **Concise:** Less code to write.
- ✅ **Performance:** Often faster (Single Pass).
- ❌ **Tricky:** "Magic Math" is harder to explain.

🏆 The Verdict

Use **LAG** for readability in interviews.
Use **Row_Number** for production scripts.

FOLLOW FOR MORE SQL PATTERNS

Anandnarayanan S

Real World Use

Where companies use this logic

🌐 User Sessions

Grouping consecutive page views into a single "Session" based on time gaps.

📈 Stock Market

Identifying "Bull Run" streaks where price increased for consecutive days.

🏭 Quality Control

Finding batches of consecutive defective products in manufacturing.

📞 Customer Support

Detecting consecutive failed login attempts (Security Alerts).

MASTERY

Quick Summary:

- ✓ Gap-and-Island is a Top Tier Question
- ✓ LAG approach is safe & readable
- ✓ Row_Number difference is a pro trick
- ✓ Key for Analytics & Data Engineering

Repost if you learned something new!