

The streak question is one subtraction.

D8A.ACADEMY/RESOURCES/GAPS-AND-ISLANDS

THE PROBLEM

islands	runs of consecutive days
gaps	the holes between them
the ask	← longest streak per user.

DEDUPE FIRST

one row per day	per user, no dupes
SELECT DISTINCT	user_id, login_at::date
two logins a day	split the streak

THE TRICK

day - ROW_NUMBER()	fixed per island
per user	PARTITION BY user_id
one subtraction	← that is the whole trick.

GROUP IT

GROUP BY the key	one row per island
MIN / MAX(day)	start and end
COUNT(*)	the streak length

THE LAG VARIANT

day - LAG(day) > 1	flag: new island
running SUM(flag)	is the island id
use it when	the gap rule is custom

AT WORK

streaks	logins, habits, orders
sessions	30 min of silence = new
uptime, subs	periods with no hole

WORTH MEMORIZING

01 Every streak, in one query

Dedupe to one row per user per day, subtract the row number, group by the result.

```
-- every login streak, per user
WITH d AS (
  SELECT DISTINCT user_id, login_at::date AS day
  FROM logins
), k AS (
  SELECT *, day - ROW_NUMBER() OVER (
    PARTITION BY user_id ORDER BY day)::int AS grp
  FROM d
)
SELECT user_id, MIN(day), MAX(day), COUNT(*) AS streak
FROM k GROUP BY user_id, grp;
```

→ Oct 1, 2, 3 minus row numbers 1, 2, 3 all give Sep 30: one island. After a gap the date jumps and the row number does not, so the key changes. Longest streak: MAX(streak) per user on top.

02 Trap: three ways the streak breaks

Each one still runs and still returns rows. The streaks are just wrong.

YOU WROTE	WHAT GOES WRONG	THE FIX
no dedupe	two logins a day split the streak	DISTINCT user, day
no PARTITION BY	users share one row number	PARTITION BY user_id
key = day - LAG(day)	every streak gets the key 1	flag gap > 1, running SUM

→ Interview phrasing: consecutive dates minus a row number give a constant, so I group by that difference to get one row per streak.