

You fixed the bug. The past is still wrong.

D8A.ACADEMY/RESOURCES/BACKFILLING

WHAT IT IS

backfill	recompute past periods
trigger	bug fix, new column, late data
not a rerun	← a rerun is one period.

RUN IT RIGHT

per partition	one day: delete + insert
oldest first	late data lands in order
idempotent	← run twice, same rows.

THE COST

6 months x 2 TB	price it first
off-peak window	nights, weekends
sample one day	verify, then the rest

BEFORE YOU START

blast radius	first bad day to today
freeze the code	one version, all days
tell downstream	dashboards will move

AIRFLOW / DBT

airflow backfill	-s start -e end
dbt --full-refresh	incrementals, from 0
max_active_runs	or the warehouse melts

GOTCHAS

append-only sink	doubles every rerun
today() in SQL	past days get today
downstream	rerun those too

WORTH MEMORIZING

01 One partition, safely, forever

The unit of a backfill is a partition. Delete it, rebuild it, move to the next day.

```

-- one day of the blast radius, oldest first
DELETE FROM fct_orders
  WHERE order_date = '2026-03-14';

INSERT INTO fct_orders
SELECT ... FROM stg_orders
  WHERE order_date = '2026-03-14';

-- run it twice: same rows, never double rows
    
```

→ Pass the day in as a parameter and loop from the first bad day to today. Never compute the date with CURRENT_DATE inside the query: March rows would be built as if it were today.

02 Trap: the three ways a backfill lies

Each one ships silently. The dashboard looks fine for a week.

THE BACKFILL	WHAT GOES WRONG	THE FIX
INSERT only	every rerun doubles rows	delete + insert per day
CURRENT_DATE in SQL	March built as today	pass the run date in
layer 1 only	marts still show the bug	rerun the whole lineage

→ Interview phrasing: a backfill is a rerun of every affected partition with frozen code, and it is only safe if the job is idempotent.