

Run it twice. Same table. That is the test.

D8A.ACADEMY/RESOURCES/UPSERT

THE PROBLEM

INSERT	run it twice: duplicates
DELETE + INSERT	← a hole in the table while it runs

THE ANSWER

INSERT ... ON CONFLICT	Postgres: DO UPDATE SET
MERGE INTO	BigQuery, Snowflake, SQL Server
a unique key	← no key, no upsert

REMEMBER

WHEN NOT MATCHED	insert
WHEN MATCHED	update, or delete
updated_at	set it in the update branch

WORTH MEMORIZING

01 Two syntaxes, one idea

Match on the key. Matched rows update, the rest insert.

POSTGRES

```
INSERT INTO daily (day, total)
SELECT day, SUM(amount)
FROM staging GROUP BY day
ON CONFLICT (day) DO UPDATE
SET total = EXCLUDED.total,
    updated_at = now();
```

BIGQUERY / SNOWFLAKE

```
MERGE INTO daily d
USING staging s ON d.day = s.day
WHEN MATCHED THEN
    UPDATE SET total = s.total
WHEN NOT MATCHED THEN
    INSERT (day, total)
VALUES (s.day, s.total);
```

02 Trap: the upsert that duplicates anyway

Three ways an upsert still fails the rerun test.

no unique key	ON CONFLICT needs a unique index on the conflict columns. No index, no conflict, plain insert.
---------------	--

duplicate source	Two staging rows with the same key in one MERGE is an error on most engines. Dedupe staging first.
------------------	--

wrong key	Matching on day when the grain is day and store inserts one row per rerun. The key is the grain.
-----------	--

→ The test is in the name of the sheet: run the job twice on the same input and count the rows. Same count, same values, you are done.