

WHERE vs HAVING, finally clear.

D8A.ACADEMY/RESOURCES/WHERE-VS-HAVING

01 Why WHERE cannot see AVG()

When WHERE runs, the groups do not exist yet.

```
FROM employees      -- 1. raw rows
WHERE ...           -- 2. filter rows
GROUP BY dept       -- 3. build groups
HAVING ...          -- 4. filter groups
SELECT dept, AVG()  -- 5. output
```

→ AVG(salary) is a property of a group. At WHERE time there are only raw rows, so the engine rejects it: there is nothing to average yet.

02 The one-word fix

Same condition, one clause later.

```
SELECT dept, AVG(salary)
FROM employees
GROUP BY dept
HAVING AVG(salary) > 60000;
```

→ HAVING runs after GROUP BY, so aggregates are fair game there. It keeps or drops whole groups, never individual rows.

03 Rows vs groups

Ask what you are filtering: rows go to WHERE, groups go to HAVING.

WHERE FILTERS ROWS

- runs before GROUP BY
- sees raw column values
- aggregates are forbidden
- shrinks the data early

HAVING FILTERS GROUPS

- runs after GROUP BY
- sees aggregated results
- AVG, SUM, COUNT welcome
- keeps or drops whole groups

05 Use both in one query

Row conditions up front, group conditions after.

```
SELECT dept, AVG(salary)
FROM employees
WHERE hired_at >= '2024-01-01'
GROUP BY dept
HAVING AVG(salary) > 60000
ORDER BY 2 DESC;
```

→ Recent hires only (rows), then well-paid departments only (groups). Sheet 005 covers the full execution order this all falls out of.

▲ COMMON TRAPS

04 Trap: row filters in HAVING

It runs fine, and it groups rows you never wanted.

```
-- groups EVERY dept, then drops
HAVING dept = 'sales'

-- filters first, groups one dept
WHERE dept = 'sales'
```

→ Both return the same result, but the WHERE version aggregates a fraction of the data. Save HAVING for conditions that genuinely need an aggregate.