

SQL Subquery Review

Database Development 2021
Cody Henrichsen

Statement Order

➤ Written

- ▷ SELECT
- ▷ FROM/JOIN ... ON
- ▷ WHERE
- ▷ GROUP BY
- ▷ HAVING
- ▷ ORDER BY

➤ Executed

- ▷ FROM/JOIN ... ON
- ▷ WHERE
- ▷ GROUP BY
- ▷ HAVING
- ▷ SELECT
- ▷ ORDER BY

Subquery

- A complete query surrounded by ()
- Should be aliased
- Can go basically anywhere
 - Most common in
 - FROM/JOIN
 - SELECT
 - WHERE

Subquery Types

- ▶ Scalar
 - ▶ Returns a table with a single value
- ▶ Multirow
 - ▶ Returns a table with multiple rows
- ▶ Correlated
 - ▶ References outer table in the subquery
 - ▶ Executed for each row of the table
- ▶ Inline View
 - ▶ In the FROM/JOIN clause

Scalar

- Used to answer a single question
- Find all the results WHERE some field is ($>$, $<$, $=$) to the (AVG/SUM/MIN/MAX) of some column
- Calculated value in the SELECT clause
- Correlated queries CAN be scalar

Multi-row

- Used to match many values
- Match results with IN, ANY, SOME, ALL
- No symbol operators allowed
 - ▷ ...
 - ▷ WHERE
 - field_name IN
 - (
 - Subquery
 -)

Multi-row and Multi-column

- ▶ If the subquery also has multiple columns group fields in parens and separated by commas

- ▶ ...

- ▶ WHERE

- (field_one, field_n) IN

- (

- Subquery

-)

ANY/SOME

- ▶ Used in conjunction with the relational operators
 - ▷ `<,>,<=,>=`
- ▶ Same operation different name
 - ▷ Pick one and stick with it
- ▶ Checks if the field and operator matches against ANY of the results from the subquery
 - ▷ `field_name operator ANY (subquery)`
 - `val.score < ANY (subquery)`
 - `demo.value >= ANY (other subquery)`

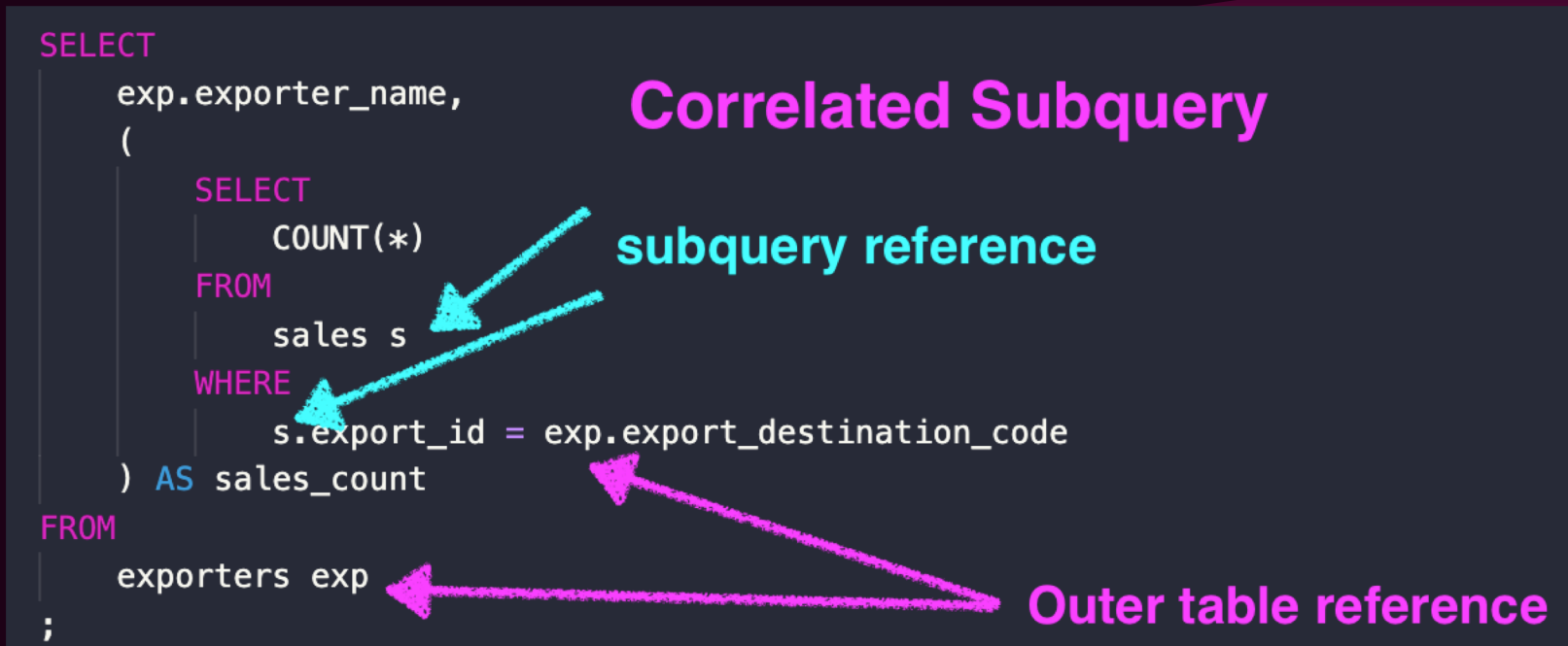
ALL

- ▶ Used in conjunction with the relational operators
 - ▶ $<, >, \leq, \geq$
- ▶ Checks if the field and operator matches against ALL of the results from the subquery
 - ▶ field_name operator ALL (subquery)
 - val.score < ALL (subquery)

Correlated

- Unlike others, this subquery executes for EVERY ROW of the outer query
 - Kinda like a nested for loop
- Easy to visually identify by the reference of the outer table in the subquery
- Subquery CANNOT run on its own because of outer/inner comparison

Correlated Annotation



Inline View

- The MOST POWERFUL subquery of them all!
- Any subquery that is in the FROM/JOIN clauses
- These should always be aliased for ease of use
- Use AS for column names to ease disambiguation
- Extremely helpful for complex questions
 - ▷ Build custom structures for filtering results