Subqueries & Query Expressions

Agenda

- Introduction to Subquery
- Properties and benefits
- Subquery using
  - where clause
  - From Clause
- Nested Subqueries

Subquery

- When a select query is nested inside another main query, it is called a subquery.
- Subqueries require common key columns for joining with main queries just like joins.
- Subqueries are also known as virtual tables with independent business logic.
- Subqueries operate independently and share results with the main query, so the complexity of writing queries decreases.

Subquery

- Subqueries are of different types and can be used in different ways according to business logic.

Type	Properties
Single Row Subquery	Feeds a single value to the main query.
Multiple Row Subquery	Returns more number of rows.
Multiple Column Subquery	Gives one or more columns results which matches with outer query columns.
Correlated Subquery	the subquery is dependent on outer query for retrieving each record.
Nested Subquery	Subqueries are placed within another subquery. It is called as nesting.

Subquery - Benefits

- Subquery separates the complex business logic from the main query.
- It is easy to debug an individual subquery instead of large and complex main query where more tables and columns are used.
- Subqueries improve the performance when they are used in a better way.
- Subqueries can be written anywhere in the SELECT clause, FROM clause and WHERE clause of another SQL query, however, the constraints of clauses are applied while using subqueries.

Subqueries in Where Clause

Subquery in Where Clause

- Using multi-row operators such as EXISTS, IN, ANY, and ALL, subqueries can be written in the WHERE clause of another query.
- They can also use single-row comparison operators such as <, >, =

Subqueries in Where Clause - Syntax using IN clause

The IN operator is useful when the main query searches all of the rows returned by a subquery.

Subquery Search Condition

Subquery Comparison Test

- Sub Queries' results are dynamic, not static.
- Developers will not need to manually enter records because dynamic input will be retrieved from a subquery.
- In WHERE clause, these subqueries use dynamic input in a condition to fetch records.

Subquery Existence Test

- Each record of the main query examines the subquery using the common key columns.
- In both the main query and the subquery, the common key column must be the same.
- When the subquery's conditions are satisfied with the input column values in the main query, it returns true (1) or false (0).

Subquery Existence Test using EXISTS

EXISTS operator is used when a record in the main query has one or more matching records in the subquery result set.

Subquery Quantified Test

- Quantify test means “validating many records ”.
- Several times, multiple records are returned from subquery. Usually there is a one-many relationship between main query and subquery.
- In such cases, the main query satisfies the condition if at least “one single record” of main Query matches with matches with “any of the multiple records” of subquery.

Subquery Quantified Test using ANY

- In the next example:
  - The main Query tries to retrieve all the rows if at least one condition in subquery is satisfied.
  - “>” or “<” range operator along with “ANY” walks through of all of the rows.

Subquery Quantified Test using ALL

- Unlike ANY, ALL is used to quantify the infinite results rather than the sample.
- In such cases, each of the main query records satisfies the condition when it matches with every/all of results of subquery.

Nested Subqueries

Nested-Subqueries

- A Subquery can be embedded or ***nested*** in another Subquery.
- Any SELECT query supports multiple subqueries nesting within one another.
- So each nested subquery executes independently and pass its result to next level subquery.
- E.g : A SELECT query consists of Subquery - A ,
  - then Subquery - A consists of Subquery - B, and
  - then the Subquery - B can consists of Subquery - C, and so on…

CRITICAL NOTE

MYSQL executes the low level subquery first and pass its intermediate result to its next immediate subquery.

Here in Eg, Subquery-C is executed first and pass its result to Subquery-B, and so on…

Subqueries with the ‘WITH’ Clause

Subqueries with the ‘WITH’ Clause

- Subqueries written in the WITH Clause plays a factoring role.
- Factoring means , the WITH Clause serves subquery results in the Main-Query wherever it is referenced.
- Once the WITH clause is executed , the same subquery result can be used for multiple times without execution.
- In complex queries, where there is a need to call subqueries repeatedly, the WITH clause plays a major role.

Subqueries and Joins

- Subquery and JOIN perform the execution of queries similarly and retrieves the same output but varies with below features.

Subqueries and Joins

- Unlike subqueries, which separate complex logic from the main query, joins include the whole logic.
- We will see examples on how subqueries can be used along with JOINS.
- Determinations can be performed in the subquery and returned as a single value, while JOIN queries perform those calculations in the main SELECT query.
- Filtering records in a table with subquery is more efficient than join queries.

Summary

This slide explains, often referred to as Inner Queries, Sub Queries, or Nested Queries, nested queries are a query within another SQL query embedded in the WHERE clause. As a result of a subquery, data will be returned that will be used in the main query as a condition for further restricting data to be retrieved.