### Subquery mysql

### Definition:

A subquery, also known as an inner query or nested query, is a query nested within another SQL query. It allows you to use the result of one query as a part of another query.

### Types of Subqueries:

#### Scalar Subquery:

A scalar subquery returns a single value (scalar) and can be used wherever an expression is allowed.

SELECT column1, (SELECT MAX(column2) FROM table2) AS max\_value

FROM table1;

#### Single-Row Subquery:

A single-row subquery returns one row of data to the outer query. It's used with comparison operators that expect single values.

SELECT column1

FROM table1

WHERE column2 = (SELECT column2 FROM table2 WHERE condition);

#### Multi-Row Subquery:

A multi-row subquery returns multiple rows of data to the outer query. It's used with operators that expect multiple values.

SELECT column1

FROM table1

WHERE column2 IN (SELECT column2 FROM table2 WHERE condition);

#### Correlated Subquery:

A correlated subquery depends on the outer query and is evaluated once for each row processed by the outer query.

SELECT column1

FROM table1 t1

WHERE column2 = (SELECT MAX(column2) FROM table2 WHERE t1.column3 = table2.column3);

### Usage of Subqueries:

#### In SELECT Statement:

Subqueries can be used in the SELECT statement to retrieve additional information.

SELECT column1, (SELECT MAX(column2) FROM table2) AS max\_value

FROM table1;

#### In WHERE Clause:

Subqueries can be used in the WHERE clause to filter rows based on certain conditions.

SELECT column1

FROM table1

WHERE column2 = (SELECT column2 FROM table2 WHERE condition);

#### In FROM Clause:

Subqueries can be used in the FROM clause to treat the result of the subquery as a temporary table.

SELECT \*

FROM (SELECT column1 FROM table1 WHERE condition) AS subquery\_table;

#### In INSERT, UPDATE, DELETE Statements:

Subqueries can be used in INSERT, UPDATE, and DELETE statements to perform operations based on the result of another query.

DELETE FROM table1

WHERE column1 IN (SELECT column1 FROM table2 WHERE condition);

### Subquery Best Practices:

* Use subqueries when necessary, but avoid nesting too deeply as it can impact readability and performance.
* Test subqueries separately to ensure they return the expected results before incorporating them into larger queries.
* Optimize subqueries by ensuring proper indexing on columns used in subqueries' conditions.

Subqueries are powerful tools in SQL that allow you to perform complex data manipulations and retrieve specific information from your database. Understanding how to use them effectively can greatly enhance your SQL querying capabilities. Adjust the table names, column names, and conditions according to your database schema.