

LESSON 6

SUBQUERIES IN WHERE CLAUSE. SUBQUERIES IN FROM CLAUSE

Ihor Liutak

1. Teach how to use subqueries in the WHERE and FROM clauses
2. Explain scenarios where these subqueries are beneficial

Short description

Familiarize students with the MySQL console and basic commands. Teach how to create databases and tables with appropriate data types. Cover essential database management commands (import/export, user management).

Kurzbeschreibung

Machen Sie die Studierenden mit der MySQL-Konsole und grundlegenden Befehlen vertraut. Bringen Sie ihnen bei, wie man Datenbanken und Tabellen mit entsprechenden Datentypen erstellt. Behandeln Sie wichtige Befehle zur Datenbankverwaltung (Import/Export, Benutzerverwaltung).

Subqueries in WHERE Clause

Definition:

Subqueries in WHERE are used to filter results based on values retrieved by another query

Syntax:

```
SELECT columns  
FROM table  
WHERE column operator  
      (SELECT subquery);
```

Basic Filtering with a Single Value:

Task:

Find employees earning more than the average salary

```
SELECT employee_id, name, salary  
FROM employees  
WHERE salary > (SELECT  
AVG(salary) FROM employees);
```

Subqueries in WHERE Clause. Examples

Using IN with Subqueries:

Task: Retrieve products sold in specific orders

```
SELECT product_id, name
FROM products
WHERE product_id IN
    (SELECT product_id
     FROM order_details
     WHERE order_id = 101);
```

Using EXISTS with Subqueries:

Task: Find customers who have placed at least one order

```
SELECT customer_id, name
FROM customers
WHERE EXISTS
    (SELECT 1 FROM orders
     WHERE customers.customer_id =
       orders.customer_id);
```

Subqueries in WHERE Clause

Best Practices for WHERE Subqueries

Use appropriate operators (=, >, IN, EXISTS).

Avoid returning multiple rows where a single value is expected.

Subqueries in FROM Clause

Definition:

Subqueries in FROM are used to create temporary tables (derived tables) for the main query to process.

Syntax:

```
SELECT columns  
FROM (SELECT subquery) AS alias  
WHERE condition;
```

Aggregating Data in a Subquery:

Task: Find departments with their average salaries

```
SELECT department_id, AVG_SALARY  
FROM (SELECT department_id,  
        AVG(salary) AS AVG_SALARY  
      FROM employees  
      GROUP BY department_id) AS dept_avg  
WHERE AVG_SALARY > 50000;
```

Subqueries in FROM Clause. Examples

Combining Data with a Derived Table:

Task: Show each product and its total sales amount

```
SELECT p.name, sales.total_sales
FROM products AS p
JOIN (SELECT product_id, SUM(quantity
* price) AS total_sales
FROM sales
GROUP BY product_id) AS sales
ON p.product_id = sales.product_id;
```

Using Subqueries for Complex Filters:

Task: Identify students whose average grade is above 85

```
SELECT s.name, avg_grades.avg_grade
FROM students AS s
JOIN (SELECT student_id, AVG(grade) AS
avg_grade
FROM grades
GROUP BY student_id) AS avg_grades
ON s.student_id = avg_grades.student_id
WHERE avg_grades.avg_grade > 85;
```

Subqueries in FROM Clause

Best Practices for FROM Subqueries

Always use an alias for subqueries in FROM.

Optimize subqueries to avoid performance bottlenecks.