STORED PROCEDURES AND FUNCTIONS IN MYSQL: OPTIMIZE AND SIMPLIFY YOUR SQL QUERIES!

WHAT ARE STORED PROCEDURES?

A Stored Procedure is a set of SQL statements that can be saved and reused multiple times. It allows you to group logic and operations into one callable unit.
Key Benefits:

Code reuse: Define once, use many times.
Performance: Stored procedures are precompiled, speeding up execution.
Security: Limits direct access to the data, improving security.

WHY USE STORED PROCEDURES?

Stored procedures are used to:

- Perform repetitive tasks: Like inserting, updating, or deleting records.
- Automate complex queries: By bundling them into one callable procedure.
- Reduce network traffic: Since the logic is executed on the server side.

Stored procedures enhance your database performance and simplify maintenance.

HOW TO CREATE A STORED PROCEDURE?

```
DELIMITER $$
CREATE PROCEDURE procedure_name (params)
BEGIN
-- SQL Statements
END $$
DELIMITER ;
```

Example: DELIMITER $$ CREATE PROCEDURE get_employee_by_department (IN dept_id INT) BEGIN SELECT * FROM employees WHERE department_id = dept_id; END $$ DELIMITER ;

This stored procedure retrieves all employees from a specific department. You can call it multiple times with different department IDs!

EXECUTING A STORED PROCEDURE?

```
CALL procedure_name (params);
```

Example

```
CALL get_employee_by_department (5);
```

By calling get_employee_by_department(5), the procedure fetches all employees in department 5. This simplifies querying and reuses code.

WHAT ARE FUNCTIONS IN MYSQL?

A Function in MySQL is similar to a stored procedure, but it returns a value and can be used within SQL statements like a regular function.
Key Benefits:

Reusability: Useful for calculations or data transformations.
Modularity: Break down complex tasks into manageable functions.

HOW TO CREATE A FUNCTION

```
DELIMITER $$
CREATE FUNCTION function_name (params)
RETURNS datatype
BEGIN
-- SQL Statements
RETURN value;
END $$
DELIMITER ;
```

Example:

```
DELIMITER $$
CREATE FUNCTION calculate_bonus (salary DECIMAL(10, 2))
RETURNS DECIMAL(10, 2)
BEGIN
RETURN salary * 0.10;
END $$
DELIMITER ;
```

The function calculate_bonus returns 10% of an employee's salary as a bonus. Functions like this are great for reusable logic within queries.

WHEN TO USE STORED PROCEDURES VS. FUNCTIONS

Stored Procedures:

- Execute multiple SQL statements
- Don’t have to return a value
- Used for performing actions (insert, update, delete)

Functions:

- Always return a value
- Used in SELECT statements for calculations or transformations

Choose the right tool based on whether you need an action or a calculation!

BEST PRACTICES

Keep stored procedures modular: Break large procedures into smaller units. Avoid side effects in functions: Functions should focus on returning values. Test and optimize for performance, especially when dealing with large datasets.

“Now that you know how to use stored procedures and functions, it’s time to practice!
Implement them in your projects to simplify your SQL queries and optimize performance!”