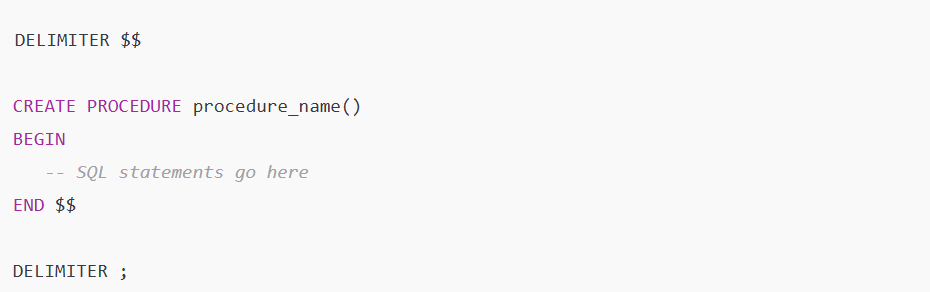
**What Is a Stored Procedure?**

A Stored Procedure is like a function in programming   
it’s a set of SQL statements saved in the database that you can run (call) whenever needed.

Instead of writing the same SQL code again and again,  
you can store it once as a procedure and reuse it.

**Basic Syntax:**

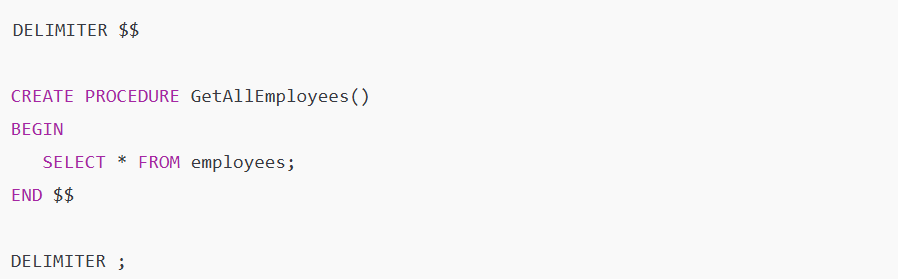
****

**Why use DELIMITER $$ ?**

Because inside a procedure we use ; many times.  
So we temporarily change the delimiter from ; to $$  
so MySQL knows where the procedure actually ends.

WE USE ; IN SQL QUERY INSIDE THE PROCEDURE SO MYSQL THIKS IT’S THE END OF QUERY BUT ACTUALLY AFTER THE END KEYWORD THE QUERY ENDS SO WE USE DELIMITER SYMBOL TO TELL THAT THIS IS THE ACTUAL END

**Example 1: Simple Procedure (No Parameters)**

****

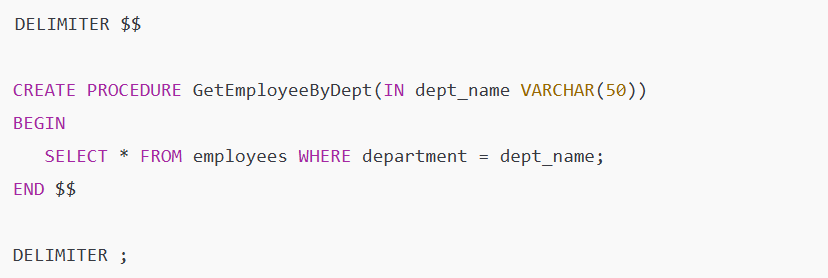
You can use any symbol (like //, #, &&, @@, etc.) as a delimiter —  
it’s just a temporary marker to tell MySQL:

“Hey, don’t stop the procedure when you see a ;, stop when you see this symbol!”

**Run it:**

****

**Example 2: Procedure With Input Parameter**

****

**Run it:**

****

It shows all employees from the Sales department.

IN means INPUT WHERE IT is used to **pass a value *into*** the procedure.  
You **give** the procedure some data to work with.

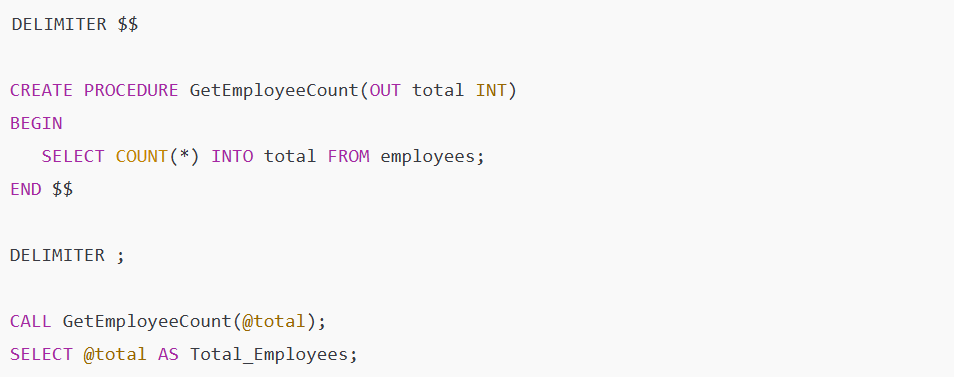
IN ABOVE EXAMPLE WE GAVE ‘SALES’ SO IT WILL USE THAT VALUE TO FILTER ROWS.

There are 3 types of parameters in procedures:

| **Keyword** | **Meaning** | **Example** |
| --- | --- | --- |
| IN | Input only | pass a value inside |
| OUT | Output only | procedure sends value out |
| INOUT | Both input & output | can modify and return value |

**Output Parameters (OUT)**

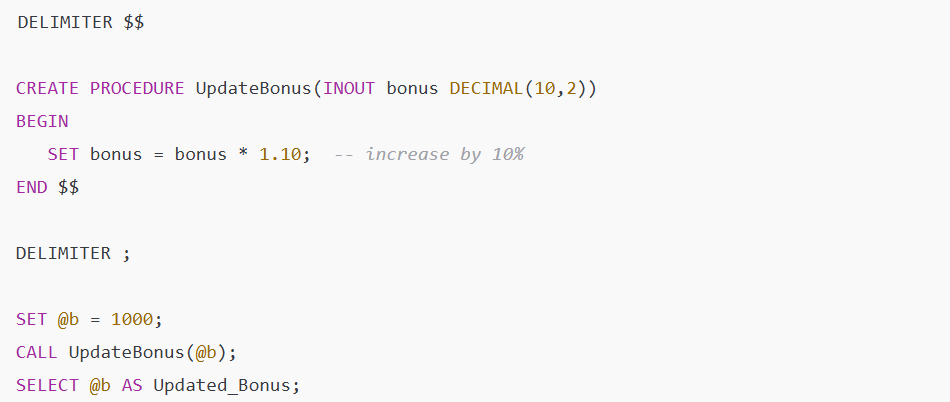
Used when you want the procedure to **return a single value** (not a table).



@total is a **session variable** that stores the result.

**Input + Output (INOUT)**

Used when you want to **pass a value in, modify it, and get it back.**



@b is sent inside, updated, and returned.

**Session Variables (@variable)**

A **session variable** is a temporary variable that exists for your SQL session.



Explanation:

* @name → variable name (starts with @)
* SET → assigns a value
* You can use it in queries like:
* SELECT \* FROM employees WHERE department = @name;

💡 Variables starting with @ can be used anywhere in your session (outside procedures too).

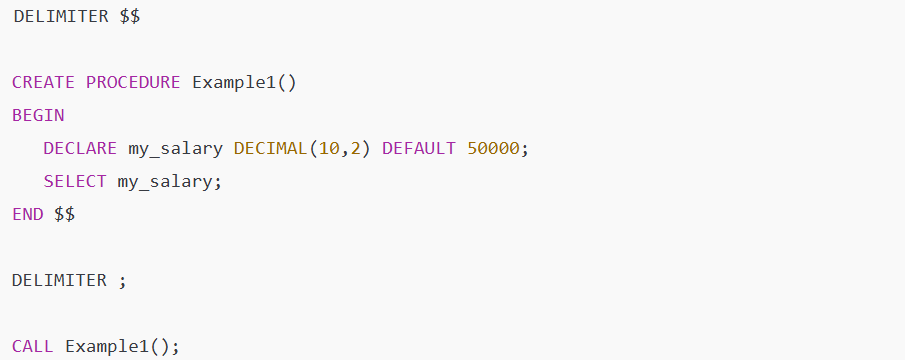
**Local Variables (DECLARE)**

Local variables are **used only inside stored procedures**.  
They are temporary and disappear when the procedure ends.

SYNTAX:



EXAMPLE:



OUTPUT :

50000

**Assigning Value → SET and SELECT INTO**

Inside procedures, we can assign values in 2 ways:

**✅ Using SET**



**✅ Using SELECT INTO**

It’s used when the value comes from a **query**.



Meaning:  
Run the query → store the result into the variable my\_salary.