In this guide we are going to see how to create and configure **MySQL stored functions** . These functions can be used for a large number of tasks and can improve **data integrity** and security, as well as **improve performance** compared to using other methods. They also improve the readability of the code.

Contents

* [**1 What is a stored function**](https://www.neoguias.com/funciones-almacenadas-mysql/#Que_es_una_funcion_almacenada)
* [**2 Database configuration**](https://www.neoguias.com/funciones-almacenadas-mysql/#Configuracion_de_la_base_de_datos)
* [**3 How to create a stored function**](https://www.neoguias.com/funciones-almacenadas-mysql/#Como_crear_una_funcion_almacenada)
* [**4 How to use a stored function**](https://www.neoguias.com/funciones-almacenadas-mysql/#Como_usar_una_funcion_almacenada)
* [**5 How to delete a stored function**](https://www.neoguias.com/funciones-almacenadas-mysql/#Como_borrar_una_funcion_almacenada)

What is a stored function

MySQL stored functions allow us to process and manipulate data procedurally in a very efficient way. You can use them in SQL statements regardless of the programming language of the server on which the queries are executed.

The **syntax of a stored function** is as follows:

CREATE FUNCTION function\_name ( param1 , param2,…)

RETURNS datatype

[NOT] DETERMINISTIC

sql\_statements

To create a stored function, you only need to have **INSERT** and **DELETE permissions** on the database.

Database configuration

As an example, we are going to configure a database on which we will then create a stored function. We'll use the command line, but you can also create a [**MySQL database**](https://www.neoguias.com/como-crear-una-base-de-datos-mysql/) in many other ways.

To create the database, open the command terminal or connect to your server using SSH. Then run MySQL with this command, replacing *user* with your username.

mysql -u user -p

Let's create a database named ***example\_base*** . To do this, follow these steps:

1. Use the create statement to create the database:

CREATE DATABASE example\_base ;

1. Now use the following command to select the database:

USE example\_base ;

If you use phpMyAdmin , you simply click on the name of the database, which in our case is *example\_base* .

1. Now create a table in the database that we will call ***products*** . To do this, run the following command:
2. CREATE TABLE products (
3. id INT NOT NULL AUTO\_INCREMENT ,
4. name VARCHAR ( 20 ) NOT NULL ,
5. cost FLOAT NOT NULL DEFAULT 0.0 ,
6. FLOAT NOT NULL price DEFAULT 0.0 ,
7. PRIMARY KEY (id)

);

1. Now let's insert some test data into the ***products table*** :

INSERT INTO products (name, cost, price) VALUES ( 'Product A' , 4 , 8 ), ( 'Product B' , 1 , 1.5 ), ( 'Product C' , 50 , 80 );

We already have the database created. Now let's create a stored procedure.

How to create a stored function

Now that we have set up a database and a table, let's create a **stored function** . We will make one that calculates the profit obtained for each product, which will be called ***calculateProfit*** . This function will accept two parameters, which, as it could not be otherwise, are the **purchase price (cost)** and the **sale price (price)** of a product. The result of the function will simply be the subtraction of the sale price and the purchase price, resulting in the profit obtained from its sale.

To create the function, run these SQL statements:

DELIMITER $$

CREATE FUNCTION calculateProfit ( FLOAT cost , FLOAT price ) RETURNS DECIMAL ( 9 , 2 )

BEGIN

DECLARE DECIMAL profit ( 9 , 2 );

SET profit = price - cost;

RETURN profit;

END $$

DELIMITER

You may be wondering what MySQL does with this function or why it doesn't execute it directly. The answer is in the **DELIMITER $$ statement** , which stops **MySQL execution** in order to define the function. When MySQL encounters the **DELIMITER statement again, execution** will continue normally.

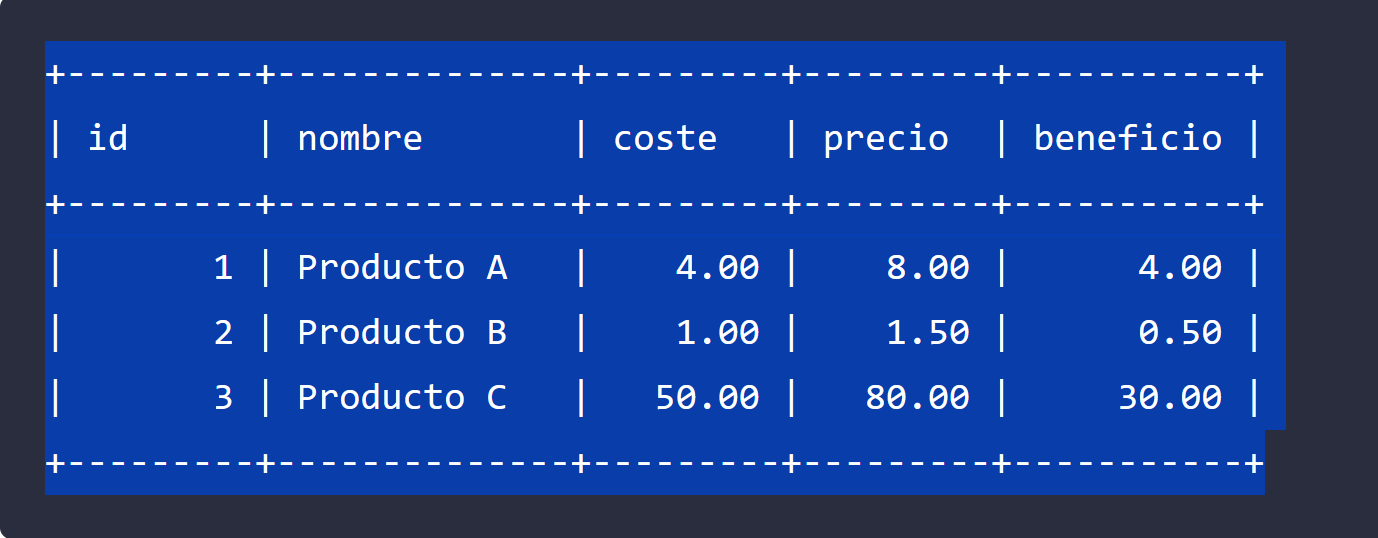
The name of the functions and their type are stored in the system table **func , in the** mysql database itself . You will not be able to define a function twice. If you want to redefine it, you will have to delete it with the **DROP statement** as we will see later, and then define it again.

How to use a stored function

Once you have created the function, you can use it directly in any query. As an example, we are going to execute this query on the **products table of the example\_base** database :

SELECT \*, calculateProfit ( cost, price) AS profit FROM products;

This SQL query will return the following results: +---------+



As you can see in the results, the stored function automatically calculates the profit for each product, being the subtraction of the price minus the cost.

How to delete a stored function

You can delete a stored function using the **DROP FUNCTION statement** . For example, if you want to eliminate the ***calculateProfit function*** from the previous example, you will have to execute this statement:

DROP FUNCTION calculateProfit ;

And that's all. If you want, you can check more details about the stored functions in the official [**MySQL documentation**](https://dev.mysql.com/doc/refman/8.0/en/create-function.html) .