
	GUÍA DE TRABAJO N° 9 – GRADO 11		
	Articulación SENA	Programación de Software	
	Ing. Néstor Raúl Suarez Perpiñan Página 1 de 4		

GUIA 9 – PROCEDIMIENTOS ALMACENADOS EN MYSQL

Objetivo:

- ✓ Conocer e implementar procedimientos almacenados usando el motor de bases de datos MySQL.

I. DEFINICIONES BASICAS

Un procedimiento almacenado es un conjunto de comandos SQL que se almacenan en el servidor de bases de datos. Una vez que se crean, los programadores o desarrolladores de software no necesitan escribir los comandos individuales ya que en su lugar pueden referirse o llamar al procedimiento almacenado.

Algunas situaciones en que los procedimientos almacenados pueden ser particularmente útiles:

- ✓ Cuando múltiples aplicaciones cliente se escriben en distintos lenguajes o funcionan en distintas plataformas, pero necesitan realizar la misma operación en la base de datos.
- ✓ Cuando la seguridad es muy importante. Los bancos, por ejemplo, usan procedimientos almacenados para todas las operaciones comunes. Esto proporciona un entorno seguro y consistente, y los procedimientos pueden asegurar que cada operación se ejecuta apropiadamente dentro del servidor. En entorno como este, las aplicaciones y los usuarios no obtendrían ningún acceso directo a las tablas de la base de datos, sólo pueden ejecutar algunos procedimientos almacenados.

Los procedimientos almacenados pueden mejorar el rendimiento ya que se necesita enviar menos información entre el servidor y el cliente. El intercambio de datos e información aumenta la carga del servidor de la base de datos ya que la mayoría del trabajo se realiza en la parte del servidor y no en el cliente. Esta situación se hace notar cuando muchas máquinas cliente y/o servidores Web se conectan con el servidor de bases de datos.

Los procedimientos almacenados permiten tener bibliotecas o funciones en el servidor de base de datos. Esta característica es compartida por los lenguajes de programación modernos que permiten este tipo de diseño, por ejemplo la POO usando clases. Usando estas características del lenguaje de programación cliente es beneficioso para el programador ya que con solo llamar los procedimientos por el nombre asignado pueden acceder al amplio abanico de posibilidades que brinda el Lenguaje SQL ejecutándose directamente sobre el servidor de bases de datos

A continuación se muestran los Comandos de MySQL relacionados con el tema de procedimientos almacenados:

	GUÍA DE TRABAJO N° 9 – GRADO 11		
	Articulación SENA	Programación de Software	
	Ing. Néstor Raúl Suarez Perpiñan Página 2 de 4		

1. CREAR UN PROCEDIMIENTO ALMACENADO

CREATE PROCEDURE *NombreProcedimiento* (parametro1 tipo, parametro2 tipo,...)
Cuerpo_Procedimiento;

Nota Importante: Los parámetros de entrada son opcionales. El Cuerpo del procedimiento consiste en los comandos SQL a ejecutar cuando sea llamando el procedimiento.

2. VER LISTADO DE PROCEDIMIENTOS YA CREADOS EN EL SERVIDOR

SELECT specific_name **FROM** information_schema.routines;

3. VER DETALLES DE UN PROCEDIMIENTO

SHOW CREATE PROCEDURE NombreProcedimiento \G;

4. ELIMINAR UN PROCEDIMIENTO

DROP PROCEDURE IF EXISTS NombreProcedimiento;

EJERCICIO DE EJEMPLO



Abra la consola de MYSQL y realice paso a paso el ejercicio que se muestra a continuación:

1. Creación de la base de datos:

DROP DATABASE IF EXISTS videoteca;
CREATE DATABASE videoteca;

2. Creación de la Tabla Actor:

USE videoteca;
DROP TABLE IF EXISTS actor;
CREATE TABLE actor (
id **INT PRIMARY KEY NOT NULL AUTO_INCREMENT**,
nombre **VARCHAR(64)**,
apellidos **VARCHAR(64)**,
mejorpelicula **VARCHAR(64)**
);

	GUÍA DE TRABAJO N° 9 – GRADO 11		
	Articulación SENA	Programación de Software	
	Ing. Néstor Raúl Suarez Perpiñan Página 3 de 4		

3. Ingresar Registros de prueba:

`INSERT INTO` actor (nombre, apellidos, mejorpelicula)

`VALUES` ('Harrison', 'Ford', 'Star Wars');

`INSERT INTO` actor (nombre, apellidos, mejorpelicula)

`VALUES` ('Christian', 'Bane', 'Batman');

4. Crear procedimiento almacenado para consultar la Lista de actores:

`CREATE PROCEDURE` lista_actores()

`SELECT * FROM` actor;

5. Ejecutar el procedimiento almacenado lista_actores()

`CALL` lista_actores();

6. Crear procedimiento almacenado para consultar el Número de registros en la tabla de actores:

`CREATE PROCEDURE` cantidad_actores()

`SELECT COUNT(*) As Numero_De_Actores FROM` actor;

7. Ejecutar el procedimiento almacenado cantidad_actores()

`CALL` cantidad_actores();

8. Procedimiento con parámetros de entrada. Consultar Lista de actores cuyo nombre comienza con una determinada letra:

`CREATE PROCEDURE` buscar_actores(letra `CHAR(2)`)

`SELECT * FROM` actor `WHERE` nombre `LIKE` letra;

9. Llamar procedimiento buscar_actores. Actores cuyo nombre comienza con la letra h:



`CALL` buscar_actores('h%');

10. Eliminar un procedimiento almacenado y crearlo de nuevo:

`DROP PROCEDURE IF EXISTS` lista_actores;

`CREATE PROCEDURE` lista_actores()

`SELECT * FROM` actor `ORDER BY` apellidos, nombre `ASC`;

	GUÍA DE TRABAJO N° 9 – GRADO 11		
	Articulación SENA	Programación de Software	
	Ing. Néstor Raúl Suarez Perpiñan Página 4 de 4		

11. Ejecutar nuevamente el procedimiento almacenado lista_actores()

`CALL lista_actores();`

12. Código de un procedimiento almacenado existente:

`SHOW CREATE PROCEDURE lista_actores\G;`

13. Detalles de un procedimiento almacenado:

`SHOW PROCEDURE STATUS LIKE 'lista_actores'\G;`

14. Detalles de todos los procedimientos almacenados:

`SHOW PROCEDURE STATUS\G`

15. Procedimiento almacenado para Inserción de un nuevo actor:

```
CREATE PROCEDURE insertar_actor(_nombre VARCHAR(64), _apellidos VARCHAR(64),
_mejorpelicula VARCHAR(32) )
INSERT INTO actor (nombre,apellidos, mejorpelicula)
VALUES (_nombre,_apellidos,_mejorpelicula);
```

16. Llamar Procedimiento insertar_ actor:

`CALL insertar_actor('Jhonny','Depp','Piratas Del Caribe');`

17. Llamar procedimiento actores lista_ para verificar la nueva Lista de actores:

`CALL lista_actores();`

TALLER

Defina, implemente y verifique los procedimientos almacenados para insertar, consultar, modificar y eliminar en cada una de las tablas de la base de datos de su proyecto Web (PHP – MySQL)