

Liliana Valdovinos  
A01712128

# **Laboratorio 23**

## **MANIPULACIÓN DE**

## **DATOS USANDO**

## **STORED PROCEDURES**

Pair Programming  
Juan Manuel Murillo López

# REGISTRAR ENTREGA

DELIMITER \$\$

```
CREATE PROCEDURE RegistrarEntrega (
    IN p_clave BIGINT,
    IN p_rfc VARCHAR(15),
    IN p_numeroProyecto BIGINT,
    IN p_fecha DATE,
    IN p_cantidad BIGINT
)
BEGIN
```

DECLARE v\_existencia INT;

-- Verificar si el material existe

```
SELECT COUNT(*) INTO v_existencia
FROM Materiales
WHERE Clave = p_clave;
```

```
IF v_existencia = 0 THEN
```

SIGNAL SQLSTATE '45000'

SET MESSAGE\_TEXT = 'El material no existe.';

ELSE

-- Insertar en la tabla Entregan

```
INSERT INTO Entregan (Clave, RFC, Número, Fecha, Cantidad)
VALUES (p_clave, p_rfc, p_numeroProyecto, p_fecha, p_cantidad);
```

END IF;

END\$\$

DELIMITER ;

The screenshot shows the phpMyAdmin interface for a database named 'actividad'. In the left sidebar, under 'Structure', there is a 'Procedures' section which is currently expanded, showing the 'RegistrarEntrega' procedure. The main query editor window displays the SQL code for creating the procedure. A green message at the top indicates that MySQL returned an empty result set. Below the message, the CREATE PROCEDURE statement is shown, including the parameters (p\_clave, p\_rfc, p\_numeroProyecto, p\_fecha, p\_cantidad), the BEGIN block, the DECLARE statement for v\_existencia, the IF condition to check if the material exists, and the INSERT INTO statement for the Entregan table.

The screenshot shows the phpMyAdmin interface for the same database 'actividad'. The left sidebar now shows the 'Tables' section, which is expanded, displaying the 'Entregan' table. The main query editor window shows the CALL statement for the 'RegistrarEntrega' procedure, with parameters 1000, 'AAAA800101', 5019, '2025-04-12', and 30. A green message at the top indicates an empty result set. Below the message, the CALL statement is displayed.

# CALCULAR COSTOS

DELIMITER \$\$

CREATE PROCEDURE CalcularCostos()

BEGIN

SELECT

Clave,

Descripcion,

Precio,

PorcentajeImpuesto,

(Precio + (Precio \* PorcentajeImpuesto / 100)) AS TotalConImpuesto

FROM Materiales;

END\$\$

DELIMITER ;

The screenshot shows the phpMyAdmin interface for the 'actividad' database. In the SQL tab, the following code is entered:

```
CREATE PROCEDURE CalcularCostos() BEGIN SELECT Clave, Descripcion, Precio, PorcentajeImpuesto, (Precio + (Precio * PorcentajeImpuesto / 100)) AS TotalConImpuesto FROM Materiales; END;
```

The status bar at the bottom indicates: MySQL returned an empty result set (i.e. zero rows). (Query took 0.0058 seconds.)

The screenshot shows the phpMyAdmin interface for the 'actividad' database, specifically for the 'materiales' table. In the SQL tab, the command CALL CalcularCostos(); is run.

The status bar at the top indicates: Showing rows 0 - 24 (44 total, Query took 0.0011 seconds.)

The results are displayed in a table:

Clave	Descripcion	Precio	PorcentajeImpuesto	TotalConImpuesto
1000	Vanilla 3/16	100	16.00	116
1010	Vanilla 4/32	115	16.00	133.4
1020	Vanilla 3/17	130	16.00	150.8
1030	Vanilla 4/33	145	16.00	168.2
1040	Vanilla 3/18	160	16.00	185.6
1050	Vanilla 4/34	175	16.00	203
1060	Vanilla 3/19	190	16.00	220.4
1070	Vanilla 4/35	205	16.00	237.8
1080	Ladrillos rojos	50	16.00	58
1090	Ladrillos grises	35	16.00	40.6
1100	Block	30	16.00	34.8
1110	Megablock	40	16.00	46.4
1120	Sillar rosa	100	16.00	116
1130	Sillar gris	110	16.00	127.6
1140	Cantera blanca	200	16.00	232
1150	Cantera gris	1210	16.00	1403.6
1160	Cantera rosa	1420	16.00	1647.2
1170	Cantera amarilla	230	16.00	266.8
1180	Recubrimiento P1001	200	16.00	232
1190	Recubrimiento P1010	220	16.00	255.2
1200	Recubrimiento P1019	240	16.00	278.4
1200	Console cubrimiento P1028	250	16.00	290

# ENTREGA POR PROVEDOR Y FECHA

DELIMITER \$\$

```
CREATE PROCEDURE EntregasPorProveedorYFecha (
    IN p_rfc VARCHAR(13),
    IN p_fecha DATE
)
BEGIN
    SELECT
        E.Clave,
        M.Descripcion,
        E.`Número`,
        P.Denominacion,
        E.Cantidad
    FROM Entregan E
    JOIN Materiales M ON E.Clave = M.Clave
    JOIN Proyectos P ON E.`Número` = P.Numero
    WHERE E.RFC = p_rfc AND E.Fecha = p_fecha;
END$$
```

DELIMITER ;

The screenshot shows the phpMyAdmin interface for a database named 'actividad'. The left sidebar lists databases (actividad, information\_schema, mysql, performance\_schema, phpmyadmin, test) and tables (entregan, materiales, proveedores, proyectos). The main window displays the SQL tab with the following code:

```
CREATE PROCEDURE EntregasPorProveedorYFecha ( IN p_rfc VARCHAR(13), IN p_fecha DATE ) BEGIN SELECT E.Clave, M.Descripcion, E.`Número`, P.Denominacion, E.Cantidad FROM Entregan E JOIN Materiales M ON E.Clave = M.Clave JOIN Proyectos P ON E.`Número` = P.Numero WHERE E.RFC = p_rfc AND E.Fecha = p_fecha; END;
```

Below the code, a message indicates an empty result set. The SQL tab also contains a call to the procedure:

```
CALL EntregasPorProveedorYFecha('AAAA800101', '2025-04-12');
```

The results tab shows the output of the procedure call:

Clave	Descripcion	Número	Denominacion	Cantidad
1000	Varilla 3/16	5019	Queretaro limpio	30