



**Uleam**

UNIVERSIDAD LAICA  
ELOY ALFARO DE MANABÍ

**FACULTAD DE CIENCIAS DE LA VIDA Y TECNOLOGÍAS**

**CARRERA INGENIERÍA EN TECNOLOGÍAS DE LA INFORMACIÓN**

**TEMA:**

**GUÍA 2: OPERACIONES Y TRANSACCIONES**

**AUTORES:**

Juan Marco Proaño Chele

Diego Miguel Arteaga Saltos

**DIRECTOR DE TEMA:**

Ing. Edison Almeida Zambrano, Mg.

**MANTA – MANABÍ – ECUADOR**

2025 – 2026

## CONTENIDO

1. Introducción .....	3
1.1 Importancia de las transacciones.....	3
2. Operaciones del Profesor .....	3
2.1 Privilegios del profesor .....	3
2.2 Ejemplos prácticos .....	4
3. Operaciones del Alumno .....	5
3.1 Privilegios del alumno.....	5
3.2 Ejemplos prácticos .....	5
4. Transacciones .....	6
4.1 Transacción válida (uso de COMMIT) .....	6
4.2 Transacción con error y uso de ROLLBACK .....	7
5. Conclusión de la Guía .....	7

## **1. Introducción**

Esta guía tiene como objetivo enseñar el uso de las operaciones CRUD (Crear, Leer, Actualizar y Eliminar) según el rol del usuario y mostrar cómo aplicar transacciones para mantener la integridad de los datos. A través de ejemplos prácticos, se explicará qué acciones puede realizar un profesor y cuáles corresponden a un alumno, así como la forma en que las transacciones permiten confirmar los cambios con **COMMIT** o revertirlos con **ROLLBACK** en caso de error, garantizando que la información en la base de datos se conserve siempre consistente.

### **1.1 Importancia de las transacciones**

Las transacciones permiten agrupar varias operaciones en una sola unidad de trabajo: o se aplican todos los cambios, o no se aplica ninguno. Esto garantiza que los datos permanezcan consistentes incluso si ocurre un error durante el proceso.

## **2. Operaciones del Profesor**

### **2.1 Privilegios del profesor**

Un profesor tiene un rol con privilegios de administrador. Esto significa que puede insertar, modificar y eliminar datos en la base de datos, además de consultarlos.

## 2.2 Ejemplos prácticos

- **Insertar un nuevo profesor**

INSERT INTO profesores (nombre, correo, fecha\_registro, sueldo\_final)

VALUES ('Gabriela López', 'gabriela.lopez@ejemplo.com', '2025-08-01', NULL);

- **Actualizar datos de un profesor existente**

UPDATE profesores

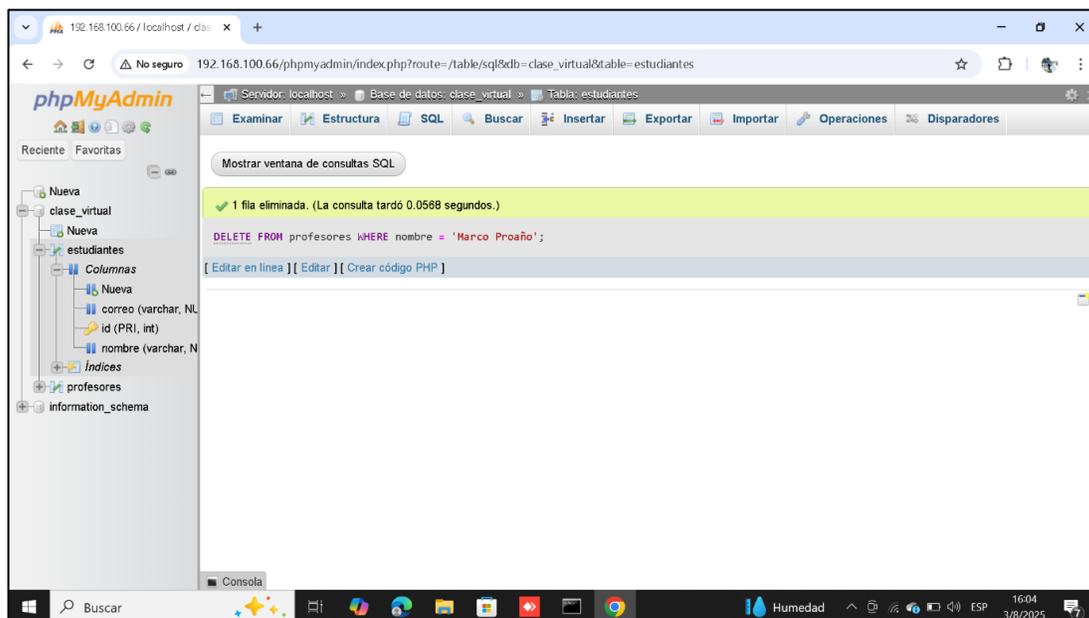
SET correo = 'ana.torres2025@ejemplo.com'

WHERE nombre = 'Ana Torres';

- **Eliminar un registro**

DELETE FROM profesores WHERE nombre = 'Marco Proaño';

*Ilustración 1. Resultado del ejercicio de registro eliminado.*



*Fuente. Elaborado por los autores de la guía.*

- **Consultar el total de profesores**

```
SELECT COUNT(*) AS total_profesores FROM profesores;
```

### **3. Operaciones del Alumno**

El alumno, en cambio, **solo puede leer información.**

#### **3.1 Privilegios del alumno**

El alumno, a diferencia del profesor, solo tiene permiso de **lectura**. No puede modificar ni eliminar datos, lo que protege la integridad de la información.

#### **3.2 Ejemplos prácticos**

- **Consultar todos los estudiantes registrados**

```
SELECT * FROM estudiantes;
```

- **Consultar la cantidad de profesores existentes**

```
SELECT COUNT(*) AS total_profesores FROM profesores;
```

## 4. Transacciones

Una transacción es un conjunto de operaciones SQL que se ejecutan como una sola unidad lógica. Esto asegura que los cambios se realicen en su totalidad o no se apliquen en absoluto.

### 4.1 Transacción válida (uso de COMMIT)

Si todas las operaciones se ejecutan correctamente, se confirma el resultado con COMMIT.

```
START TRANSACTION;
```

```
UPDATE profesores SET sueldo_final = 9.0 WHERE nombre = 'Luis Mendoza';
```

```
UPDATE profesores SET sueldo_final = 9.5 WHERE nombre = 'Ana Torres';
```

```
COMMIT;
```

*Ilustración 2. Resultado de la transacción válida.*



*Fuente. Elaborado por los autores de la guía.*

## 4.2 Transacción con error y uso de ROLLBACK

Cuando una operación falla, la transacción se cancela y se deshacen los cambios con ROLLBACK.

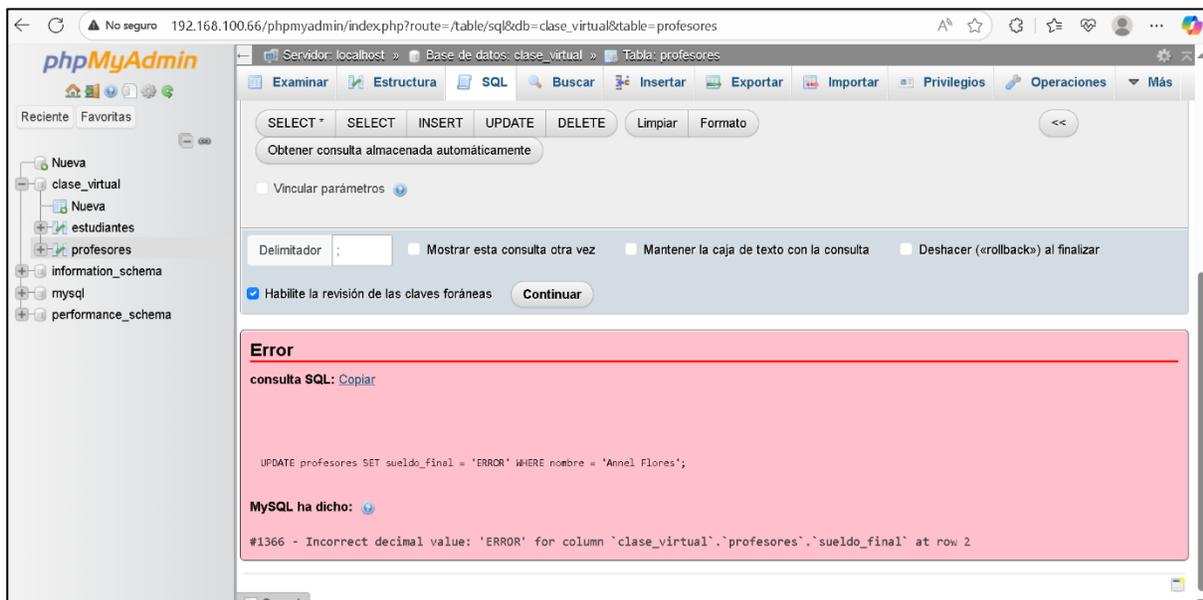
```
START TRANSACTION;
```

```
UPDATE profesores SET sueldo_final = 7.8 WHERE nombre = 'Luis Mendoza';
```

```
UPDATE profesores SET sueldo_final = 'ERROR' WHERE nombre = 'Ana Torres'; -- error
```

```
ROLLBACK;
```

*Ilustración 3. Ejemplo de transacción errónea.*



*Fuente. Elaborado por los autores de la guía.*

## 5. Conclusión de la Guía

Aprendiste a diferenciar operaciones según el rol y a usar **transacciones** para garantizar que los datos se mantengan consistentes, incluso ante errores.