

Trigger

```
CREATE TRIGGER nombre_trigger
{BEFORE | AFTER} {INSERT | UPDATE | DELETE} ON nombre_tabla
FOR EACH ROW
BEGIN
    -- Lógica del trigger aquí
END;
```

Sintaxis:

- `CREATE TRIGGER`: Esta es la declaración para crear un trigger en MySQL.
- `nombre_trigger`: Es el nombre que le asignas al trigger. Puedes elegir cualquier nombre válido.
- `{BEFORE | AFTER}`: Indica cuándo se ejecutará el trigger, ya sea antes o después de la acción que dispara el trigger.
- `{INSERT | UPDATE | DELETE}`: Especifica qué tipo de acción disparará el trigger. Puede ser INSERT (cuando se inserta una fila), UPDATE (cuando se actualiza una fila) o DELETE (cuando se elimina una fila).
- `ON nombre_tabla`: Especifica la tabla en la que se creará el trigger.
- `FOR EACH ROW`: Indica que el trigger se ejecutará para cada fila afectada por la acción que lo dispara.
- `BEGIN` y `END`: Estas palabras clave delimitan el bloque de código del trigger donde se coloca la lógica que deseas ejecutar.

Dentro del bloque de código del trigger, puedes escribir cualquier instrucción SQL o utilizar variables especiales como `NEW` y `OLD` para acceder a los valores de la fila actualizada o eliminada.

Ejemplo más completo de un trigger que actualiza una columna `ultima_actualizacion` con la fecha y hora actual cada vez que se inserta o actualiza una fila en una tabla llamada `mi_tabla`:

```
CREATE TRIGGER actualizar_fecha
BEFORE INSERT ON mi_tabla
FOR EACH ROW
BEGIN
    SET NEW.ultima_actualizacion = NOW();
END;
```

Este trigger se ejecutará antes de cada inserción en la tabla `mi_tabla` y actualizará la columna `ultima_actualizacion` con la fecha y hora actual.

Ejemplo

```
-- Trigger para insertar en `alimentacion_log` cada vez que se inserte, actualice o
elimine en `alimentacion`
DELIMITER //
CREATE TRIGGER tr_alimentacion_insert
AFTER INSERT ON alimentacion
FOR EACH ROW
BEGIN
    INSERT INTO alimentacion_log (
```

```
    operacion,
    id_alimentacion,
    tipo_alimento,
    fecha)
VALUES (
    'INSERT',
    NEW.id_alimentacion,
    NEW.tipo_alimento,
    NOW());
END;

DELIMITER //
CREATE TRIGGER tr_alimentacion_update
AFTER UPDATE ON alimentacion
FOR EACH ROW
BEGIN
    INSERT INTO alimentacion_log (
        operacion,
        id_alimentacion,
        tipo_alimento,
        fecha)
VALUES (
    'UPDATE',
    OLD.id_alimentacion,
    OLD.tipo_alimento,
    NOW());
END;

DELIMITER //
CREATE TRIGGER tr_alimentacion_delete
BEFORE DELETE ON alimentacion
FOR EACH ROW
BEGIN
    INSERT INTO alimentacion_log (
        operacion,
        id_alimentacion,
        tipo_alimento,
        fecha)
VALUES (
    'DELETE',
    OLD.id_alimentacion,
    OLD.tipo_alimento,
    NOW());
END;
```