

# Triggers in MySQL

---

A **trigger** is a special kind of stored program that is **automatically executed (triggered)** when a specific event occurs in a table — such as `INSERT` , `UPDATE` , or `DELETE` .

Triggers are commonly used for:

- Logging changes
  - Enforcing additional business rules
  - Automatically updating related data
- 

## Basic Trigger Structure

---

```
CREATE TRIGGER trigger_name
AFTER INSERT ON table_name
FOR EACH ROW
BEGIN
    -- statements to execute
END;
```

Triggers can be fired:

- `BEFORE` or `AFTER` an event
  - On `INSERT` , `UPDATE` , or `DELETE`
- 

## Scenario: Log Every New User Insertion

---

Suppose we want to log every time a new user is inserted into the `users` table. We'll create a separate table called `user_log` to store log entries.

---

## Step 1: Create the Log Table

---

```
CREATE TABLE user_log (  
  id INT AUTO_INCREMENT PRIMARY KEY,  
  user_id INT,  
  name VARCHAR(100),  
  created_on TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
);
```

---

## Step 2: Create the Trigger

---

We now define a trigger that runs **after a new user is added**.

```
DELIMITER $$  
  
CREATE TRIGGER after_user_insert  
AFTER INSERT ON users  
FOR EACH ROW  
BEGIN  
  INSERT INTO user_log (user_id, name)  
  VALUES (NEW.id, NEW.name);  
END$$  
  
DELIMITER ;
```

### Explanation:

- `AFTER INSERT` means the trigger fires after the user is inserted.
  - `NEW` refers to the new row being added to the `users` table.
  - We insert the new user's ID and name into the `user_log` table.
-

## Step 3: Test the Trigger

```
CALL AddUser('Ritika Jain', 'ritika@example.com', 'Female', '1996-03-12', 74000);
```

Now check the `user_log` table:

```
SELECT * FROM user_log;
```

You should see Ritika's info automatically logged.

## Dropping a Trigger

If you need to remove a trigger:

```
DROP TRIGGER IF EXISTS after_user_insert;
```

## Summary

| Trigger Component   | Description  |
|---|--|
| <code>BEFORE</code> / <code>AFTER</code>                        | When the trigger runs  |
| <code>INSERT</code> / <code>UPDATE</code> / <code>DELETE</code> | What kind of action triggers it  |
| <code>NEW.column</code>   | Refers to the new row (for <code>INSERT</code> , <code>UPDATE</code> ) |
| <code>OLD.column</code>   | Refers to the old row (for <code>UPDATE</code> , <code>DELETE</code> ) |
| <code>FOR EACH ROW</code>                                       | Executes for each affected row   |