

MySQL

Database Triggers

SQL statements that are
AUTOMATICALLY RUN
when a specific table is
changed

The Syntax

```
CREATE TRIGGER trigger_name  
    trigger_time trigger_event ON table_name FOR EACH ROW  
BEGIN  
    ...  
END;
```

The Syntax

trigger_time	trigger_event	ON	table_name
BEFORE	INSERT		photos
AFTER	UPDATE		users
	DELETE		

Example 1

A Simple Validation

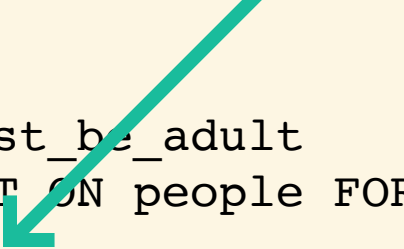
```
DELIMITER $$

CREATE TRIGGER must_be_adult
  BEFORE INSERT ON users FOR EACH ROW
  BEGIN
    IF NEW.age < 18
    THEN
      SIGNAL SQLSTATE '45000'
        SET MESSAGE_TEXT = 'Must be an adult!';
    END IF;
  END;
$$

DELIMITER ;
```

Run Code First.
Ask Questions Later.

Refers to data that is about to be inserted




```
DELIMITER $$

CREATE TRIGGER must_be_adult
  BEFORE INSERT ON people FOR EACH ROW
  BEGIN
    IF NEW.age < 18
    THEN
      SIGNAL SQLSTATE '45000'
        SET MESSAGE_TEXT = 'Must be an adult!';
    END IF;
  END;
$$

DELIMITER ;
```

It's a long story....



```
DELIMITER $$

CREATE TRIGGER must_be_adult
  BEFORE INSERT ON people FOR EACH ROW
  BEGIN
    IF NEW.age < 18
    THEN
      SIGNAL SQLSTATE '45000'
      SET MESSAGE_TEXT = 'Must be an adult!';
    END IF;
  END;
$$

DELIMITER ;
```


MySQL Errors

A numeric error code (1146). This number is MySQL-specific

A five-character SQLSTATE value ('42S02').

The values are taken from ANSI SQL and ODBC and are more standardized.

A message string - textual description of the error

TRY THIS!

Do something to cause an error, like...

TRY THIS!

Do something to cause an error, like...

```
SELECT * FROM kjhsadkjhas;
```

TRY THIS!

Do something to cause an error, like...

```
SELECT * FROM kjhsadjhas;
```



TRY THIS!

Do something to cause an error, like...


```
SELECT * FROM kjhsadkjhas;
```



```
ERROR 1146 (42S02): Table 'ig_clone.kjhsadkjhas' doesn't exist
```

45000

A generic state representing
"unhandled user-defined exception"



```
DELIMITER $$

CREATE TRIGGER must_be_adult
  BEFORE INSERT ON people FOR EACH ROW
  BEGIN
    IF NEW.age < 18
    THEN
      SIGNAL SQLSTATE '45000'
        SET MESSAGE_TEXT = 'Must be an adult!';
    END IF;
  END;
$$

DELIMITER ;
```

WHAT IS THIS?!

DELIMITER \$\$

```
CREATE TRIGGER must_be_adult
  BEFORE INSERT ON people FOR EACH ROW
  BEGIN
    IF NEW.age < 18
    THEN
      SIGNAL SQLSTATE '45000'
      SET MESSAGE_TEXT = 'Must be an adult!';
    END IF;
  END;
```

\$\$

DELIMITER ;

Example 2

Preventing Self-Follows

```
DELIMITER $$

CREATE TRIGGER trigger_name
    trigger_time trigger_event ON table_name FOR EACH ROW
    BEGIN
    END;

$$

DELIMITER ;
```



Starter Code

Example 2

Preventing Self-Follows

```
DELIMITER $$

CREATE TRIGGER example_cannot_follow_self
  BEFORE INSERT ON follows FOR EACH ROW
  BEGIN
    IF NEW.follower_id = NEW.following_id
    THEN
      SIGNAL SQLSTATE '45000'
        SET MESSAGE_TEXT = 'Cannot follow yourself';
    END IF;
  END;
$$

DELIMITER ;
```

Example 3

Logging Unfollows

```
DELIMITER $$

CREATE TRIGGER trigger_name
    trigger_time trigger_event ON table_name FOR EACH ROW
BEGIN
    END;
$$

DELIMITER ;
```



Starter Code

Example 3

Logging Unfollows

```
DELIMITER $$

CREATE TRIGGER create_unfollow
    AFTER DELETE ON follows FOR EACH ROW
BEGIN
    INSERT INTO unfollows
    SET follower_id = OLD.follower_id,
        followee_id = OLD.followee_id;
END$$

DELIMITER ;
```

Managing Triggers

Listing Triggers

```
SHOW TRIGGERS;
```

Removing Triggers

```
DROP TRIGGER trigger_name;
```

A WORD OF WARNING

Triggers can make debugging hard!

YOUR
TURN