

Stored Procedures and Triggers - II

Syntax:

```
CREATE TRIGGER trigger_name
{BEFORE | AFTER} {INSERT | UPDATE | DELETE}
ON table_name
FOR EACH ROW
BEGIN
    -- SQL statements
END;
```

```
CREATE TABLE student_info (
    stud_id    INT PRIMARY KEY,
    stud_code  INT UNIQUE,
    stud_name  VARCHAR(50),
    subject    VARCHAR(30),
    marks      INT CHECK (marks BETWEEN 0 AND 100),
    phone      VARCHAR(15)
);
```

```
INSERT INTO student_info (stud_id, stud_code, stud_name, subject, marks, phone) VALUES
( 1,101,'Mark','English',68,'3454569357'),
( 2,102,'Joseph','Physics',70,'9876543659'),
( 3,103,'John','Maths',70,'9765326975'),
( 4,104,'Barack','Maths',92,'87069873256'),
( 5,105,'Rinky','Maths',85,'6753159757'),
( 6,106,'Adam','Science',82,'79642256864'),
( 7,107,'Andrew','Science',83,'5674243579'),
( 8,108,'Brayan','Science',83,'7524316576'),
( 9,109,'Alexandar','Biology',67,'2347346438'),
(10,110,'Clara','Biology',74,'2342342345'),
(11,111,'Derek','English',59,'4564564560'),
(12,112,'Ella','English',91,'1231231234'),
(13,113,'Fiona','Physics',65,'9879879876'),
(14,114,'George','Physics',79,'3453453451'),
(15,115,'Helen','Maths',88,'2223334445'),
(16,116,'Ian','Science',95,'6667778880'),
(17,117,'Jane','Biology',57,'9998887776'),
(18,118,'Kyle','English',73,'1112223339'),
(19,119,'Liam','Maths',81,'8887776665'),
(20,120,'Mia','Physics',93,'4445556667');
```

stud_id	stud_code	stud_name	subject	marks	phone
---------	-----------	-----------	---------	-------	-------

stud_id	stud_code	stud_name	subject	marks	phone
1	101	Mark	English	68	3454569357
2	102	Joseph	Physics	70	9876543659
3	103	John	Maths	70	9765326975
4	104	Barack	Maths	92	87069873256
5	105	Rinky	Maths	85	6753159757
6	106	Adam	Science	82	79642256864
7	107	Andrew	Science	83	5674243579
8	108	Brayan	Science	83	7524316576
9	109	Alexandar	Biology	67	2347346438
10	110	Clara	Biology	74	2342342345
11	111	Derek	English	59	4564564560
12	112	Ella	English	91	1231231234
13	113	Fiona	Physics	65	9879879876
14	114	George	Physics	79	3453453451
15	115	Helen	Maths	88	2223334445
16	116	Ian	Science	95	6667778880
17	117	Jane	Biology	57	9998887776
18	118	Kyle	English	73	1112223339
19	119	Liam	Maths	81	8887776665
20	120	Mia	Physics	93	4445556667

Before Insert

```

DELIMITER $$
CREATE TRIGGER trg_before_insert_student
BEFORE INSERT ON student_info
FOR EACH ROW
BEGIN
    -- Reject the impossible marks
    IF New.marks > 100 OR New.marks < 0 THEN
        SIGNAL SQLSTATE '45000'
        SET MESSAGE_TEXT = 'Marks must be between 0 And 100';
    END IF;
END $$
DELIMITER ;

```

043 • SHOW TRIGGERS;

Trigger	Event	Table	Statement	Timing	Created	sql_mode	Definer	character_set_client	collation_connection
trg_before_insert_student	INSERT	student_info	BEGIN -- Reject the impossible marks IF New....	BEFORE	2025-07-15 21:22:37.31	ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLE...	root@localhost	utf8mb4	utf8mb4_0900_ai_ci

```

INSERT INTO student_info
VALUES(21,121,"Victor","Maths",105,"9839871923");

INSERT INTO student_info
VALUES(21,121,"Victor","Maths",-10,"9839871923");

-- Error Code: 1644. Marks must be between 0 And 100

```

INSERT -> NEW ☒ OLD - (X)

UPDATE -> NEW ☒ OLD - ☒

DELETE -> NEW (X) OLD - ☒

AFTER - Audit Log

After Insert

```
-- AFTER INSERT [Audit Log]
CREATE TABLE audit_log(
    log_id INT AUTO_INCREMENT PRIMARY KEY,
    action ENUM('INSERT', 'UPDATE', 'DELETE'),
    stud_id INT,
    log_time TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
DELIMITER $$
CREATE TRIGGER trg_after_insert_student
AFTER INSERT ON student_info
FOR EACH ROW
BEGIN
    INSERT INTO audit_log(action , stud_id)
    VALUES('INSERT', NEW.stud_id);
END $$
DELIMITER ;

INSERT INTO student_info
VALUES(21,121,"Victor","Maths",92,"9839871923");

SELECT * FROM student_info;
SELECT * FROM audit_log;
```

074 • **SELECT * FROM student_info;**

Result Grid						
Filter Rows:						
	stud_id	stud_code	stud_name	subject	marks	phone
	12	112	Ella	English	91	1231231234
	13	113	Fiona	Physics	65	9879879876
	14	114	George	Physics	79	3453453451
	15	115	Helen	Maths	88	2223334445
	16	116	Ian	Science	95	6667778880
	17	117	Jane	Biology	57	9998887776
	18	118	Kyle	English	73	1112223339
	19	119	Liam	Maths	81	8887776665
	20	120	Mia	Physics	93	4445556667
▶	21	121	Victor	Maths	92	9839871923
•	HULL	HULL	HULL	HULL	HULL	HULL

075 • **SELECT * FROM audit_log;**

Result Grid				
Filter Rows:				
	log_id	action	stud_id	log_time
▶	1	INSERT	21	2025-07-15 21:44:55
•	HULL	HULL	HULL	HULL

```

INSERT INTO student_info VALUES (22, 122, "Aarav", "Science", 88, "9876543210");
INSERT INTO student_info VALUES (23, 123, "Meera", "English", 75, "9812345678");
INSERT INTO student_info VALUES (24, 124, "Rohan", "History", 81, "9823456789");
INSERT INTO student_info VALUES (25, 125, "Sneha", "Biology", 90, "9834567890");
INSERT INTO student_info VALUES (26, 126, "Kabir", "Computer", 95, "9845678901");

```

log_id	action	stud_id	log_time
1	INSERT	21	2025-07-15 21:44:55
2	INSERT	26	2025-07-15 21:47:31
3	INSERT	22	2025-07-15 21:47:53
4	INSERT	23	2025-07-15 21:47:55
5	INSERT	24	2025-07-15 21:47:58
6	INSERT	25	2025-07-15 21:48:01

After Update

```

-- AFTER UPDATE
DELIMITER $$
CREATE TRIGGER trg_after_update_student
AFTER UPDATE ON student_info
FOR EACH ROW
BEGIN
INSERT INTO audit_log(action , stud_id)
VALUES('UPDATE', NEW.stud_id);
END $$
DELIMITER ;

```

095 • SHOW TRIGGERS;

096

Trigger	Event	Table	Statement	Timing	Created	sql_mode	Definer	character_set_client	collation_connection
trg_before_insert_student	INSERT	student_info	BEGIN -- Reject the impossible marks IF New....	BEFORE	2025-07-15 21:22:37.31	ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLE...	root@localhost	utf8mb4	utf8mb4_0900_ai_ci
trg_after_insert_student	INSERT	student_info	BEGIN INSERT INTO audit_log(action , stud_id) ...	AFTER	2025-07-15 21:44:07.41	ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLE...	root@localhost	utf8mb4	utf8mb4_0900_ai_ci
trg_after_update_student	UPDATE	student_info	BEGIN INSERT INTO audit_log(action , stud_id) ...	AFTER	2025-07-15 21:53:43.10	ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLE...	root@localhost	utf8mb4	utf8mb4_0900_ai_ci

097 • UPDATE student_info

098 SET marks = 95

099 WHERE stud_id = 4;

100

stud_id	stud_code	stud_name	subject	marks	phone
1	101	Mark	English	68	3454569357
2	102	Joseph	Physics	70	9876543659
3	103	John	Maths	70	9765326975
4	104	Barack	Maths	95	87069873256
5	105	Rinky	Maths	85	6753159757

075 • SELECT * FROM audit_log;

log_id	action	stud_id	log_time
2	INSERT	26	2025-07-15 21:47:31
3	INSERT	22	2025-07-15 21:47:53
4	INSERT	23	2025-07-15 21:47:55
5	INSERT	24	2025-07-15 21:47:58
6	INSERT	25	2025-07-15 21:48:01
7	UPDATE	4	2025-07-15 21:59:22
	NULL	NULL	NULL

Let's give 2 marks grace for all students on "Physics"

075 • **SELECT * FROM audit_log;**

log_id	action	stud_id	log_time
8	UPDATE	2	2025-07-15 22:06:13
9	UPDATE	13	2025-07-15 22:06:13
10	UPDATE	14	2025-07-15 22:06:13
11	UPDATE	20	2025-07-15 22:06:13
•	NULL	NULL	NULL

```
SET SQL_SAFE_UPDATES = 0;
UPDATE student_info
SET marks = marks + 2
WHERE subject = "Physics";
```

After DELETE

```
-- AFTER DELETE
DELIMITER $$
CREATE TRIGGER trg_after_delete_student
AFTER DELETE ON student_info
FOR EACH ROW
BEGIN
    INSERT INTO audit_log(action , stud_id)
    VALUES('DELETE', OLD.stud_id);
END $$
DELIMITER ;
```

117 • **DELETE FROM student_info WHERE stud_id = 26;**
 118 • **SELECT * FROM student_info;**
 119 • **SELECT * FROM audit_log;**

log_id	action	stud_id	log_time
9	UPDATE	13	2025-07-15 22:06:13
10	UPDATE	14	2025-07-15 22:06:13
11	UPDATE	20	2025-07-15 22:06:13
12	DELETE	26	2025-07-15 22:11:16
•	NULL	NULL	NULL

BEFORE UPDATE → Prevent Marks Reduction.

INSERT -> NEW ✓ OLD - (X)

UPDATE -> NEW ✓ OLD - ✓

DELETE -> NEW (X) OLD - ✓

135 • **SHOW TRIGGERS;**

Trigger	Event	Table	Statement	Timing	Created
trg_before_update_no_marks_drop	UPDATE	student_info	BEGIN IF New.marks < OLD.marks THEN SIG...	BEFORE	2025-07-15 22:20:15.45
trg_after_update_student	UPDATE	student_info	BEGIN INSERT INTO audit_log(action , stud_id) ...	AFTER	2025-07-15 21:53:43.10

```
-- BEFORE UPDATE

DELIMITER $$
CREATE TRIGGER trg_before_update_no_marks_drop
BEFORE UPDATE ON student_info
FOR EACH ROW
BEGIN
    IF New.marks < OLD.marks THEN
        SIGNAL SQLSTATE '45000'
        SET MESSAGE_TEXT = 'Reducing Marks are not allowed';
    END IF;
END $$
DELIMITER ;

SHOW TRIGGERS;

UPDATE student_info
SET marks = marks - 2
WHERE subject = "Physics";

-- Error Code: 1644. Reducing Marks are not allowed
```

BEFORE DELETE

156 • SHOW TRIGGERS;

Trigger	Event	Table	Statement	Timing	Created	sql_mode	Definer	character_set_client	collation_connection
trg_before_insert_student	INSERT	student_info	BEGIN -- Reject the impossible marks IF New....	BEFORE	2025-07-15 21:22:37.31	ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLE...	root@localhost	utf8mb4	utf8mb4_0900
trg_after_insert_student	INSERT	student_info	BEGIN INSERT INTO audit_log(action , stud_id) ...	AFTER	2025-07-15 21:44:07.41	ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLE...	root@localhost	utf8mb4	utf8mb4_0900
trg_before_update_no_marks_drop	UPDATE	student_info	BEGIN IF New.marks < OLD.marks THEN SIG...	BEFORE	2025-07-15 22:20:15.45	ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLE...	root@localhost	utf8mb4	utf8mb4_0900
trg_after_update_student	UPDATE	student_info	BEGIN INSERT INTO audit_log(action , stud_id) ...	AFTER	2025-07-15 21:53:43.10	ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLE...	root@localhost	utf8mb4	utf8mb4_0900
trg_before_delete_high_performers	DELETE	student_info	BEGIN IF OLD.marks >= 90 THEN SIGNAL SQ...	BEFORE	2025-07-15 22:29:54.15	ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLE...	root@localhost	utf8mb4	utf8mb4_0900
trg_after_delete_student	DELETE	student_info	BEGIN INSERT INTO audit_log(action , stud_id) ...	AFTER	2025-07-15 22:09:28.61	ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLE...	root@localhost	utf8mb4	utf8mb4_0900

157 • SELECT * FROM student_info;

stud_id	stud_code	stud_name	subject	marks	phone
1	101	Mark	English	68	3454569357
2	102	Joseph	Physics	72	9876543659
3	103	John	Maths	70	9765326975
4	104	Barack	Maths	95	87069873256
5	105	Rinky	Maths	85	6753159757
6	106	Adam	Science	82	79642256864
7	107	Andrew	Science	83	5674243579
8	108	Brayan	Science	83	7524316576
9	109	Alexandar	Biology	67	2347346438
10	110	Clara	Biology	74	2342342345
11	111	Derek	English	59	4564564560

```

DELIMITER $$
CREATE TRIGGER trg_before_delete_high_performers
BEFORE DELETE ON student_info
FOR EACH ROW
BEGIN
    IF OLD.marks >= 90 THEN
        SIGNAL SQLSTATE '45000'
        SET MESSAGE_TEXT = 'Can not Delete high-performing students who scored >=90';
    END IF;
END $$
DELIMITER ;

SHOW TRIGGERS;
SELECT * FROM student_info;
DELETE FROM student_info WHERE stud_id = 4;
-- Error Code: 1644. Can not Delete high-performing students who scored >=90
DELETE FROM student_info WHERE stud_id = 24;

```

Tracker - Marks Change

```

INSERT -> NEW ☒ OLD - (X)
UPDATE -> NEW ☒ OLD - ☒
DELETE -> NEW (X) OLD - ☒

```

-- AFTER UPDATE [track Marks Changes]

```

CREATE TABLE marks_history(
    history_id INT AUTO_INCREMENT PRIMARY KEY,
    stud_id INT,
    old_marks INT,
    new_marks INT,
    changed_on TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);

```

```

DELIMITER $$
CREATE TRIGGER trg_after_update_marks
AFTER UPDATE ON student_info
FOR EACH ROW
BEGIN
    IF OLD.marks <> NEW.marks THEN
        INSERT INTO marks_history(stud_id,old_marks,new_marks)
        VALUES(OLD.stud_id, OLD.marks , NEW.marks);
    END IF;
END $$
DELIMITER ;

SELECT * FROM student_info;
UPDATE student_info
SET marks = 88
WHERE stud_id = 1;

SELECT * FROM marks_history;

```

```

187 • UPDATE student_info
188 SET marks = 88
189 WHERE stud_id = 1;

```

stud_id	stud_code	stud_name	subject	marks	phone
1	101	Mark	English	68	3454569357
2	102	Joseph	Physics	72	9876543659
3	103	John	Maths	70	9765326975
4	104	Barack	Maths	95	87069873256
5	105	Rinky	Maths	85	6753159757
6	106	Adam	Science	82	79642256864
7	107	Andrew	Science	83	5674243579
8	108	Brayan	Science	83	7524316576
9	109	Alexandar	Biology	67	2347346438
10	110	Clara	Biology	74	2342342345
11	111	Derek	English	59	4564564560

```

191 • SELECT * FROM marks_history;

```

history_id	stud_id	old_marks	new_marks	changed_on
1	1	68	88	2025-07-15 22:48:15
NULL	NULL	NULL	NULL	NULL

STORE PROCEDURE - IN , INOUT

stud_id	stud_code	stud_name	subject	marks	phone
4	104	Barack	Maths	92	87069873256
5	105	Rinky	Maths	85	6753159757
6	106	Adam	Science	82	79642256864
7	107	Andrew	Science	83	5674243579
8	108	Brayan	Science	83	7524316576

```

210 • SET @marks = 5;
211 • CALL increase_marks_by_id(7,@marks);
212 • SELECT @marks;

```

@marks
88

```

-- Store Procedure -> CALL -> Update
DELIMITER $$
CREATE PROCEDURE increase_marks_by_id(IN in_stud_id INT , INOUT new_marks INT)
BEGIN
    UPDATE student_info
    SET marks = marks + new_marks
    WHERE stud_id = in_stud_id;
    SELECT marks INTO new_marks
    FROM student_info
    WHERE stud_id = in_stud_id;
END $$
DELIMITER ;

SET @marks = 5;
CALL increase_marks_by_id(7,@marks);
SELECT @marks;

```