

10-Triggers

1. Create a table named teachers with fields id, name, subject, experience and salary and insert 8 rows.
2. Create a before insert trigger named before_insert_teacher that will raise an error “salary cannot be negative” if the salary inserted to the table is less than zero.
3. Create an after insert trigger named after_insert_teacher that inserts a row with teacher_id, action, timestamp to a table called teacher_log when a new entry gets inserted to the teacher table. teacher_id -> column of teacher table, action -> the trigger action, timestamp -> time at which the new row has got inserted.
4. Create a before delete trigger that will raise an error when you try to delete a row that has experience greater than 10 years.
5. Create an after delete trigger that will insert a row to teacher_log table when that row is deleted from teacher table.

1.

```
1  ##Create the teachers table and insert 8 rows
2  CREATE TABLE teachers (
3      id INT PRIMARY KEY AUTO_INCREMENT,
4      name VARCHAR(100) NOT NULL,
5      subject VARCHAR(50) NOT NULL,
6      experience INT NOT NULL,
7      salary DECIMAL(10, 2) NOT NULL
8  );
9
10 INSERT INTO teachers (name, subject, experience, salary) VALUES
11 ('ARCHANA JAY', 'Math', 5, 50000.00),
12 ('DIVYA KRISHNA', 'Science', 8, 55000.00),
13 ('RITHIKA SOORAJ', 'English', 12, 60000.00),
14 ('VANIKA S', 'History', 3, 45000.00),
15 ('DEVAGNI JAY', 'Math', 2, 40000.00),
16 ('KASHI NATH', 'Science', 7, 48000.00),
17 ('URMI MUKUNDAN', 'English', 10, 62000.00),
18 ('DEV DARSHITH', 'History', 15, 70000.00);
19
20 ##Create the before_insert_teacher trigger
21
```

Output

#	Time	Action	Message	Duration / Fetch
1	22:44:14	CREATE TABLE teachers (id INT PRIMARY KEY AUTO_INCREMENT, name VARCHAR(100) NOT NULL, ...	0 row(s) affected	0.031 sec
2	22:48:29	INSERT INTO teachers (name, subject, experience, salary) VALUES (ARCHANA JAY, 'Math', 5, 50000.00), (DIVY...	8 row(s) affected Records: 8 Duplicates: 0 Warnings: 0	0.000 sec
3	22:51:46	CREATE TRIGGER before_insert_teacher BEFORE INSERT ON teachers FOR EACH ROW BEGIN IF NEW.sa...	0 row(s) affected	0.000 sec
4	22:52:40	CREATE TABLE teacher_log (teacher_id INT, action VARCHAR(50), timestamp DATETIME DEFAULT CU...	0 row(s) affected	0.015 sec
5	22:55:30	CREATE TRIGGER after_insert_teacher AFTER INSERT ON teachers FOR EACH ROW BEGIN INSERT INTO...	0 row(s) affected	0.000 sec

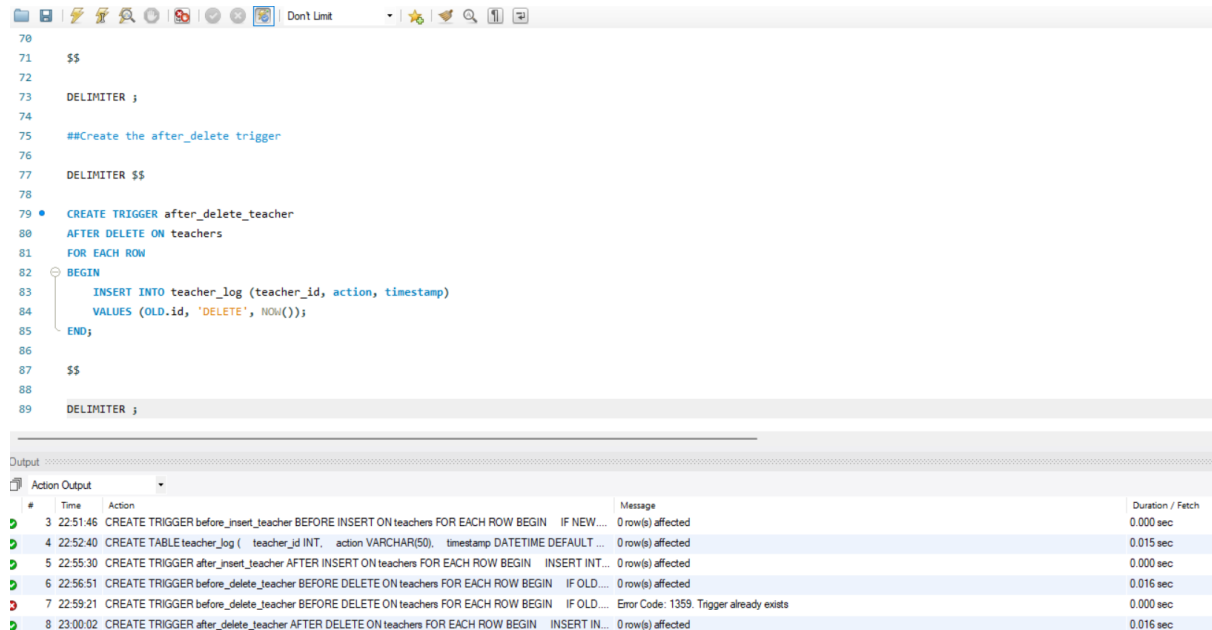
2.

```
37
38 CREATE TABLE teacher_log (
39     teacher_id INT,
40     action VARCHAR(50),
41     timestamp DATETIME DEFAULT CURRENT_TIMESTAMP
42 );
43
44 DELIMITER $$
45
46 CREATE TRIGGER after_insert_teacher
47 AFTER INSERT ON teachers
48 FOR EACH ROW
49 BEGIN
50     INSERT INTO teacher_log (teacher_id, action, timestamp)
51     VALUES (NEW.id, 'INSERT', NOW());
52 END;
53
54 $$
55
56 DELIMITER ;
```

Output

#	Time	Action	Message	Duration / Fetch
1	22:44:14	CREATE TABLE teachers (id INT PRIMARY KEY AUTO_INCREMENT, name VARCHAR(100) NOT NULL, ...	0 row(s) affected	0.031 sec
2	22:48:29	INSERT INTO teachers (name, subject, experience, salary) VALUES (ARCHANA JAY, 'Math', 5, 50000.00), (DIVY...	8 row(s) affected Records: 8 Duplicates: 0 Warnings: 0	0.000 sec
3	22:51:46	CREATE TRIGGER before_insert_teacher BEFORE INSERT ON teachers FOR EACH ROW BEGIN IF NEW.sa...	0 row(s) affected	0.000 sec
4	22:52:40	CREATE TABLE teacher_log (teacher_id INT, action VARCHAR(50), timestamp DATETIME DEFAULT CU...	0 row(s) affected	0.015 sec
5	22:55:30	CREATE TRIGGER after_insert_teacher AFTER INSERT ON teachers FOR EACH ROW BEGIN INSERT INTO...	0 row(s) affected	0.000 sec

3.



```
70
71 $$
72
73 DELIMITER ;
74
75 ##Create the after_delete trigger
76
77 DELIMITER $$
78
79 CREATE TRIGGER after_delete_teacher
80 AFTER DELETE ON teachers
81 FOR EACH ROW
82 BEGIN
83     INSERT INTO teacher_log (teacher_id, action, timestamp)
84     VALUES (OLD.id, 'DELETE', NOW());
85 END;
86
87 $$
88
89 DELIMITER ;
```

Output

#	Time	Action	Message	Duration / Fetch
3	22:51:46	CREATE TRIGGER before_insert_teacher BEFORE INSERT ON teachers FOR EACH ROW BEGIN	IF NEW.... 0 row(s) affected	0.000 sec
4	22:52:40	CREATE TABLE teacher_log (teacher_id INT, action VARCHAR(50), timestamp DATETIME DEFAULT ...	0 row(s) affected	0.015 sec
5	22:55:30	CREATE TRIGGER after_insert_teacher AFTER INSERT ON teachers FOR EACH ROW BEGIN	INSERT INT... 0 row(s) affected	0.000 sec
6	22:56:51	CREATE TRIGGER before_delete_teacher BEFORE DELETE ON teachers FOR EACH ROW BEGIN	IF OLD.... 0 row(s) affected	0.016 sec
7	22:59:21	CREATE TRIGGER before_delete_teacher BEFORE DELETE ON teachers FOR EACH ROW BEGIN	IF OLD.... Error Code: 1359. Trigger already exists	0.000 sec
8	23:00:02	CREATE TRIGGER after_delete_teacher AFTER DELETE ON teachers FOR EACH ROW BEGIN	INSERT IN... 0 row(s) affected	0.016 sec

Summary

1. The teachers table has been created with the required fields and populated with 8 rows.
2. A BEFORE INSERT trigger prevents negative salaries.
3. An AFTER INSERT trigger logs actions in the teacher_log table.
4. A BEFORE DELETE trigger restricts deletion for teachers with over 10 years of experience.
5. An AFTER DELETE trigger logs deletions in the teacher_log table.

This setup ensures data integrity and provides a log of teacher actions in the database.