Task 9 - 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. tecaher\_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.

USE D33;

CREATE TABLE teachers

(id INT PRIMARY KEY,

name VARCHAR(50),

subject VARCHAR(50),

experience INT,

salary DECIMAL(10, 2));

SELECT \*FROM TEACHERS;

INSERT INTO teachers (id, name, subject, experience, salary) VALUES

(1, 'Alice Smith', 'Math', 5, 50000.00),

(2, 'Bob Johnson', 'Science', 8, 55000.00),

(3, 'Carol Williams', 'English', 3, 48000.00),

(4, 'David Brown', 'History', 10, 60000.00),

(5, 'Eva Davis', 'Geography', 7, 53000.00),

(6, 'Frank Miller', 'Physics', 9, 57000.00),

(7, 'Grace Wilson', 'Chemistry', 6, 52000.00),

(8, 'Hank Moore', 'Biology', 4, 49000.00);

SELECT \*FROM TEACHERS;

delimiter #

create trigger before\_insert\_teachers

before insert on teachers

for each row

begin

IF NEW.salary < 0 THEN

SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'Salary cannot be negative';

END IF;

end#

delimiter ;

insert into teachers values(9,'roy toms','Physical Triner',7,0);

select \*from teachers;

CREATE TABLE teacher\_log (

teacher\_id INT,

action VARCHAR(50),

timestamp DATETIME);

select \*from teacher\_log;

DELIMITER #

CREATE TRIGGER after\_insert\_teacher

AFTER INSERT ON teachers

FOR EACH ROW

BEGIN

INSERT INTO teacher\_log (teacher\_id, action, timestamp)

VALUES (NEW.id,'INSERT', NOW());

END;

DELIMITER ;

INSERT INTO teacherS VALUES (10, 'Alice George', 'IT' , 5,50000);

select \*from teacher\_log;

DELIMITER #

CREATE TRIGGER before\_delete\_teacher

BEFORE DELETE ON teachers

FOR EACH ROW

BEGIN

IF OLD.experience > 10 THEN

SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'Cannot delete a teacher with more than 10 years of experience';

END IF;

END;

DELIMITER ;

delimiter #

CREATE TRIGGER after\_delete\_teacher

AFTER DELETE ON teachers

FOR EACH ROW

BEGIN

INSERT INTO teacher\_log (teacher\_id, action, timestamp)

VALUES (OLD.id, 'DELETE', NOW());

END;

DELIMITER ;







