1. Create a table named teachers with fields id, name, subject, experience and salary and insert 8 rows.

```
CREATE TABLE TEACHERS (
TEACHER_ID INT AUTO_INCREMENT PRIMARY KEY,
NAME VARCHAR(50),
SUBJECT VARCHAR(30),
EXPERIENCE INT,
SALARY DECIMAL(10, 2)
);

INSERT INTO TEACHERS (TEACHER_ID, NAME, SUBJECT, EXPERIENCE,SALARY) VALUES
(1, 'FAZIL', 'MATHAMATICS', 12,50000);

INSERT INTO TEACHERS ( NAME, SUBJECT, EXPERIENCE,SALARY) VALUES
( 'SHAHID', 'PHYSICS', 10,45000),( 'KIRAN', 'CHEMISTRY', 15,65000),
( 'JONAH', 'BIOLOGY', 8,47000),( 'TESSA', 'HINDI', 11,48500),
( 'MARIA', 'MALAYALAM', 15,55000),( 'SURESH', 'ENGLISH', 20,75000),
( 'SUPRIYA', 'HISTORY', 25,88500);
```

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.

```
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 ( NAME, SUBJECT, EXPERIENCE,SALARY) VALUES
('MENON', 'HISTORY', 25,-2000);
```

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.

```
DELIMITER $$
CREATE TRIGGER AFTER_TEACHER_INSERT
AFTER INSERT ON TEACHERS
FOR EACH ROW
BEGIN
INSERT INTO TEACHERS_LOG(LOG_ID,TEACHER_ID,ACTION)
VALUES(LOG_ID,NEW.TEACHER_ID,'NEW TEACHER ENTRY');
END $$
DELIMITER ;


CREATE TABLE TEACHERS_LOG (
LOG_ID INT AUTO_INCREMENT PRIMARY KEY,
TEACHER_ID INT,
ACTION VARCHAR(70),
ACTION_TIME TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);


SELECT * FROM TEACHERS_LOG;
```

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.

```
DELIMITER $$
CREATE TRIGGER BEFORE_TEACHER_DELETE
BEFORE DELETE ON TEACHERS
FOR EACH ROW
BEGIN
IF OLD.EXPERIENCE>10
THEN
SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT= 'YOU DONT HAVE THE RIGHTS TO
DELETE THIS TEACHER';
END IF;
END $$
DELIMITER ;
```

```
SELECT * FROM TEACHERS;


DELETE FROM TEACHERS WHERE NAME='MARIA'
```

5. Create an after delete trigger that will insert a row to teacher_log table when that row is deleted from teacher table.

```
DELIMITER $$
CREATE TRIGGER AFTER_TEACHER_DELETE
AFTER DELETE ON TEACHERS
FOR EACH ROW
BEGIN
INSERT INTO TEACHERS_LOG(LOG_ID,ACTION) VALUES(OLD.TEACHER_ID,'RETIRED
TEACHER');
END $$
DELIMITER ;

DROP TRIGGER IF EXISTS BEFORE_TEACHER_DELETE;
SET SQL_SAFE_UPDATES = 0 ;

DELETE  FROM TEACHERS WHERE NAME ='MARIA';

SELECT * FROM TEACHERS_LOG;
```