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

```
18
19 •  select*from teachers;
20
```

Result Grid | Filter Rows: | Edit: | Export/Import:

id	name	subject	experience	salary
1	Ashly	Maths	5	45000.00
2	Jibin	Science	12	50000.00
3	Goshal	English	3	40000.00
4	Richu	History	8	48000.00
5	Mebin	Physics	10	53000.00
6	Jonhy	Biology	6	47000.00
7	Gopika	Chemistry	2	38000.00
8	George	Geography	15	60000.00
NULL	NULL	NULL	NULL	NULL

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.

```
21    DELIMITER //
22
23 •  CREATE TRIGGER before_insert_teacher
24    BEFORE INSERT ON teachers
25    FOR EACH ROW
26    BEGIN
27        IF NEW.salary < 0 THEN
28            SIGNAL SQLSTATE '45000'
29            SET MESSAGE_TEXT = 'Salary cannot be negative';
30        END IF;
31    END //
32
33    DELIMITER ;
34
35 •  INSERT INTO teachers (id, name, subject, experience, salary)
```

Output

Action Output

#	Time	Action	Message
4	21:52:10	select*from teachers LIMIT 0, 1000	8 row(s) returned
5	21:54:29	CREATE TRIGGER before_insert_teacher BEFORE INSERT ON teachers FOR EACH ROW BEGIN IF NE...	0 row(s) affected
6	21:55:24	INSERT INTO teachers (id, name, subject, experience, salary) VALUES (9, 'Mariya', 'Economics', 7, -10000)	Error Code: 1644. Salary cannot be negative

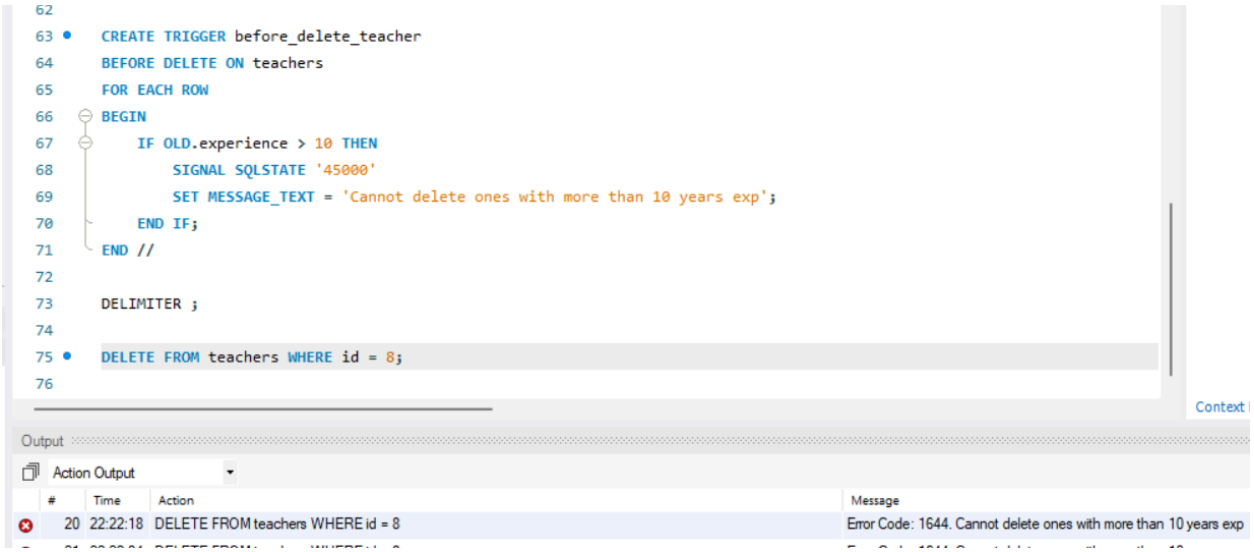
3. Create an after insert trigger named after_insert_teacher that inserts a row with teacher_id,action, timestamp to a table called teacher_log when a new entry gets inserted to the teacher table. tecaher_id -> column of teacher table, action -> the trigger action, timestamp -> time at which the new row has got inserted.

```
45 •  CREATE TRIGGER after_insert_teacher
46    AFTER INSERT ON teachers
47    FOR EACH ROW
48    BEGIN
49        INSERT INTO teacher_log (teacher_id, action, log_timestamp)
50        VALUES (NEW.id, 'INSERT', NOW());
51    END //
52
53    DELIMITER ;
54
55 •  INSERT INTO teachers (id, name, subject, experience, salary)
56    VALUES (9, 'Mariya', 'Economics', 7, 55000);
57
58 •  SELECT * FROM teacher_log;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

teacher_id	action	log_timestamp
9	INSERT	2024-11-23 22:15:59

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.



```
62
63 •  CREATE TRIGGER before_delete_teacher
64    BEFORE DELETE ON teachers
65    FOR EACH ROW
66    BEGIN
67        IF OLD.experience > 10 THEN
68            SIGNAL SQLSTATE '45000'
69            SET MESSAGE_TEXT = 'Cannot delete ones with more than 10 years exp';
70        END IF;
71    END //
72
73    DELIMITER ;
74
75 •  DELETE FROM teachers WHERE id = 8;
76
```

Context

Output

Action Output

#	Time	Action	Message
20	22:22:18	DELETE FROM teachers WHERE id = 8	Error Code: 1644. Cannot delete ones with more than 10 years exp

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

```
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, log_timestamp)
84      VALUES (OLD.id, 'DELETE', NOW());
85  END //
86
87  DELIMITER ;
88
89
90  DELETE FROM teachers WHERE id = 4;
91
92  SELECT * FROM teacher_log;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

teacher_id	action	log_timestamp
9	INSERT	2024-11-23 22:15:59
4	DELETE	2024-11-23 22:23:28