

# **Database and Management System Lab**

## **Lab Experiment – 16**

Name: Kartikeya Singh

Roll No: R2142230051

Sap\_ID: 500123812

**Title:** To understand the concepts of Trigger.

**Objective:** Students will be able to implement the concept of trigger.

1. Create a row level trigger for the customers table

```
1  CREATE OR REPLACE TRIGGER customer_audit_trigger
2  BEFORE INSERT OR UPDATE OR DELETE ON CUSTOMERS
3  FOR EACH ROW
4  BEGIN
5      -- For INSERT operations
6      IF INSERTING THEN
7          DBMS_OUTPUT.PUT_LINE('INSERT Operation');
8          DBMS_OUTPUT.PUT_LINE('New Salary: ' || :NEW.SALARY);
9      -- For UPDATE operations
10     ELSIF UPDATING THEN
11         DBMS_OUTPUT.PUT_LINE('UPDATE Operation');
12         DBMS_OUTPUT.PUT_LINE('Old Salary: ' || :OLD.SALARY);
13         DBMS_OUTPUT.PUT_LINE('New Salary: ' || :NEW.SALARY);
14         DBMS_OUTPUT.PUT_LINE('Salary Difference: ' || (:NEW.SALARY - :OLD.SALARY));
15     -- For DELETE operations
16     ELSIF DELETING THEN
17         DBMS_OUTPUT.PUT_LINE('DELETE Operation');
18         DBMS_OUTPUT.PUT_LINE('Deleted Salary: ' || :OLD.SALARY);
19     END IF;
20 END;
21 /
```

```
1  -- Test 1: INSERT
2  INSERT INTO CUSTOMERS (ID, NAME, AGE, ADDRESS, SALARY)
3  VALUES (7, 'Test User', 30, 'Test City', 3000.00);
4
5
```

Results	Explain	Describe	Saved SQL	History
INSERT Operation New Salary: 3000  1 row(s) inserted.				

```
1  -- Test 2: UPDATE
2  UPDATE CUSTOMERS
3  SET SALARY = 2500.00
4  WHERE ID = 1;
5
```

Results	Explain	Describe	Saved SQL	History
---------	---------	----------	-----------	---------

UPDATE Operation  
Old Salary: 2000  
New Salary: 2500  
Salary Difference: 500

1 row(s) updated.

```
1  -- Test 3: DELETE
2  DELETE FROM CUSTOMERS
3  WHERE ID = 7;
4
5
```

Results	Explain	Describe	Saved SQL	History
---------	---------	----------	-----------	---------

DELETE Operation  
Deleted Salary: 3000

1 row(s) deleted.

# **Database and Management System Lab**

## **Lab Experiment – 17**

Name: Kartikeya Singh

Roll No: R2142230051

Sap\_ID: 500123812

B.Tech CSE, SEM-III, B-2

**Title: To understand the concepts of Trigger.**

**Objective:** Students will be able to implement the concept of trigger.

```
1 CREATE OR REPLACE TRIGGER SALARY_VIOLATION
2 BEFORE INSERT OR UPDATE OF SALARY, SUPERSSN ON EMPLOYEE
3 FOR EACH ROW
4 DECLARE
5     v_super_salary NUMBER;
6     v_dept_mgr_ssn VARCHAR2(9);
7 BEGIN
8     -- Only proceed with checks if supervisor exists
9     IF :NEW.SUPERSSN IS NOT NULL THEN
10         -- Get supervisor's salary
11         SELECT SALARY INTO v_super_salary
12         FROM EMPLOYEE
13         WHERE SSN = :NEW.SUPERSSN;
14
15         -- Check if employee's salary is greater than supervisor's salary
16         IF :NEW.SALARY > v_super_salary THEN
17             RAISE_APPLICATION_ERROR(-20001,
18                 'Employee salary ('|| :NEW.SALARY ||') cannot be greater than supervisor salary ('|| v_super_salary ||')');
19         END IF;
20     END IF;
```

```
21
22     -- Get department manager's SSN for the employee's department
23     IF :NEW.DNO IS NOT NULL THEN
24         SELECT MGRSSN INTO v_dept_mgr_ssn
25         FROM DEPARTMENT
26         WHERE DNUMBER = :NEW.DNO;
27
28         -- If the employee is the department manager, salary must be at least 50000
29         IF :NEW.SSN = v_dept_mgr_ssn AND :NEW.SALARY < 50000 THEN
30             RAISE_APPLICATION_ERROR(-20002,
31                 'Department manager salary must be at least 50000');
32         END IF;
33     END IF;
34
35     -- Validate basic salary rules
36     IF :NEW.SALARY < 20000 THEN
37         RAISE_APPLICATION_ERROR(-20003,
38             'Salary cannot be less than 20000');
39     END IF;
40
```

```
41     -- Validate supervisor changes
42     IF UPDATING('SUPERSSN') AND :NEW.DNO IS NOT NULL THEN
43         -- Prevent department managers from having supervisors
44         IF :NEW.SSN = v_dept_mgr_ssn AND :NEW.SUPERSSN IS NOT NULL THEN
45             RAISE_APPLICATION_ERROR(-20004,
46                 'Department managers cannot have supervisors');
47         END IF;
48     END IF;
49 END;
50 /
51
52
```

Results	Explain	Describe	Saved SQL	History
---------	---------	----------	-----------	---------

Trigger created.