

Lab Exercise 5

Aim:

Write a program to execute row-level trigger in PL/SQL.

Procedure:

Step 1: Create Main Table

Step 2: Create Log Table (for tracking row-level changes)

Step 3: Create a Sequence for log_id

Step 4: Create Row-Level Trigger

Step 5: Test the Trigger and Procedure

Queries:

Create a New Table:

```
CREATE TABLE Employee (  
    emp_id    NUMBER PRIMARY KEY,  
    emp_name  VARCHAR2(50),  
    salary    NUMBER  
);
```

Create the log table:

```
CREATE TABLE Emp_Log (  
    log_id    NUMBER PRIMARY KEY,  
    emp_id    NUMBER,  
    operation  VARCHAR2(20),  
    old_salary NUMBER,  
    new_salary NUMBER,  
    log_date   DATE  
);
```

-- Create a sequence for log_id

```
CREATE SEQUENCE emp_log_seq  
START WITH 1  
INCREMENT BY 1;
```

Trigger to Use the Sequence:

```
CREATE OR REPLACE TRIGGER trg_emp_salary_log
AFTER INSERT OR UPDATE OR DELETE ON Employee
FOR EACH ROW
BEGIN
    IF INSERTING THEN
        INSERT INTO Emp_Log (log_id, emp_id, operation, new_salary, log_date)
        VALUES (emp_log_seq.NEXTVAL, :NEW.emp_id, 'INSERT', :NEW.salary,
        SYSDATE);

    ELSIF UPDATING THEN
        INSERT INTO Emp_Log (log_id, emp_id, operation, old_salary, new_salary,
        log_date)
        VALUES (emp_log_seq.NEXTVAL, :OLD.emp_id,
        'UPDATE', :OLD.salary, :NEW.salary, SYSDATE);

    ELSIF DELETING THEN
        INSERT INTO Emp_Log (log_id, emp_id, operation, old_salary, log_date)
        VALUES (emp_log_seq.NEXTVAL, :OLD.emp_id, 'DELETE', :OLD.salary,
        SYSDATE);
    END IF;
END;
/
```

Test the trigger:

-- Insert a record

```
INSERT INTO Employee VALUES (101, 'Anand', 50000);
```

-- Update a record

```
UPDATE Employee SET salary = 55000 WHERE emp_id = 101;
```

-- Delete a record

DELETE FROM Employee WHERE emp_id = 101;

View log entries:

SELECT * FROM Emp_Log;

Output:

☒ Autocommit Display 10 ▼

```
ELSIF DELETING THEN
    INSERT INTO Emp_Log (log_id, emp_id, operation, old_salary, log_date)
    VALUES (emp_log_seq.NEXTVAL, :OLD.emp_id, 'DELETE', :OLD.salary, SYSDATE);
END IF;
END;
/
INSERT INTO Employee VALUES (101, 'Anand', 50000);
UPDATE Employee SET salary = 55000 WHERE emp_id = 101;
DELETE FROM Employee WHERE emp_id = 101;|
SELECT * FROM Emp_Log;
```

Results Explain Describe Saved SQL History

LOG_ID	EMP_ID	OPERATION	OLD_SALARY	NEW_SALARY	LOG_DATE
1	101	INSERT	-	50000	14-OCT-25
2	101	UPDATE	50000	55000	14-OCT-25
3	101	DELETE	55000	-	14-OCT-25

3 rows returned in 0.15 seconds [CSV Export](#)

Result:

Thus the Row level trigger with PL/SQL queries was executed successfully.