

Experiment No 26: Implement triggers for given database

I. **Practical Significance:** PL/SQL triggers are named database objects stored in a database that can be called automatically on the occurrence of a particular event. Triggers can be used to automate database processes, enforce data integrity, and improve database performance. This practical allows students to implement PL/SQL triggers for given database.

ii. **INDUSTRY / EMPLOYER EXPECTED OUTCOME:**

To implement PL/SQL triggers for applying integrity, consistency and error handling in a database

III. **COURSE LEVEL LEARNING OUTCOMES (COS):**

CO4 - Implement PL/SQL codes for given application.

IV. **LABORATORY LEARNING OUTCOME:**

Implement triggers for given database.

V. **Relevant Affective Domain related outcome(s)**

- a. Follow precautionary measures.
- b. Follow installation steps.
- c. Follow ethical practices.

VI. **Relevant Theoretical Background**

A PL/SQL trigger is a database object that *automatically executes* a specific action such as inserting, updating, or deleting data in response to events like a row being modified or a table being dropped. Triggers are often used to enforce business rules, maintain data integrity, and automate tasks within the database.

Syntax:

```
CREATE [OR REPLACE ] TRIGGER trigger_name
{BEFORE | AFTER | INSTEAD OF }
{INSERT [OR] | UPDATE [OR] | DELETE}
```


DATABASE MANAGEMENT SYSTEM (313302)

[OF col_name]

ON table_name

[REFERENCING OLD AS o NEW AS n]

[FOR EACH ROW]

WHEN (condition)

DECLARE

Declaration-statements

BEGIN

Executable-statements

EXCEPTION

Exception-handling-statements

END;

VII. Required Resources/apparatus/equipment with specifications

Sr.No	Equipment Name with Broad Specifications	Relevant LLO Number
1	Computer system with all necessary components like; motherboard, random access memory (RAM), read-only memory (ROM), internal hard disk drives, Mouse, Keyboard, and RDBMS applications such as Oracle Express Edition, MySQL, SQLite, Oracle Apex etc.	All

VIII. Procedure

1. Define the PL/SQL block structure
2. Implement the logic for the given problem

IX. Result(s)

*In this practical we studied to create
implement triggers for given database*

X. Practical related questions (Provide space for answers)

Note: Below are a few sample questions for reference. Teacher must design more such

Maharashtra State Board of Technical Education ('K Scheme')

* Practical related questions.

1) →

Row level Trigger	Statement level Trigger
(i) Row level trigger execute once for each & every Row in the transaction	(i) Statement level triggers executes only once for each single transaction.
(ii) Specifically used for data auditing purpose	(ii) used for enforcing all additional security on the transaction performed on the table
(iii) "FOR EACH ROW" clause is present in CREATE TRIGGER Command	(iii) "FOR EACH ROW" clause is omitted in CREATE TRIGGER Command.
(iv) Ex: IF 1500 rows are to be inserted into a table, the row level trigger would execute 1500 times.	(iv) Ex: IF 1500 rows are to be inserted into a table the Statement level trigger would execute only one time.

→

The WHEN clause in a trigger is used to specify a condition that must be true for the trigger to fire. It allows you to fine-tune the activation of the trigger.

Purpose:

Efficiency: Reduces unnecessary trigger execution by filtering out cases that don't meet.

Specificity: Allows you to handle very specific business logic within triggers

Ex:

```
CREATE OR REPLACE TRIGGER Salary_Update_Trigger
BEFORE UPDATE OF salary ON employees
FOR EACH ROW
```

```
WHEN (NEW.salary > 10000)
```

```
BEGIN
```

```
DBMS_OUTPUT.PUT_LINE ('Salary updated to' ||
: New.Salary);
```

```
END;
```


* Exercise

1) →

CREATE OR REPLACE TRIGGER Salary check trigger
BEFORE UPDATE of Salary on emp
FOR EACH ROW

WHEN (NEW.Salary < 5000)

BEGIN

DBMS_OUTPUT.PUT_LINE ('Alert: Employee ' || NEW.
emp_name || ' has a Salary below 5000');

END;

/

2) →

CREATE OR REPLACE TRIGGER dept update trigger
AFTER UPDATE ON department

FOR EACH ROW

BEGIN

DBMS_OUTPUT.PUT_LINE ('Department record
updated Dept ID: ' || NEW.dept_id);

END;

/