

Practical – 2

TRIGGER

1	Create a trigger which displays a message “Employee added” after inserting a new record into the employee table.
2	Create a trigger that will cause an error message and display old and new salary when an update occurs that would result in a salary increase greater than 2000 of the current salary.
3	Create a trigger before employee detail is deleted, it also modifies the data into the department table. EMP(empno,name,salary,deptno,designation) DEPT(deptno,name,noofemployee>Total_salary)
4	Create a login table Login(ID,username,login_date) Create a trigger which will display current user and current date before insert the record and update the user and login_date with current user and current date *take the current username in username field of Login table. *select user from dual;
5	Create table Temp(no integer,name char(10)) . and insert the 5 records; Create a trigger that 1) display all record of table and 2) display message (‘table is about deleted’) 3) delete all records before drop the table Temp.

Trigger Syntax:

```
CREATE [OR REPLACE] TRIGGER <trigger_name>
{ BEFORE|AFTER } { INSERT|DELETE|UPDATE } ON <table_name> | <view_name>
[REFERENCING [NEW AS <new_row_name>] [OLD AS <old_row_name>] ]
[FOR EACH ROW [WHEN <trigger_condition>]]]
<trigger_body>
```

Example 1

```
CREATE OR REPLACE TRIGGER emp_alert_trig
BEFORE INSERT ON emp
BEGIN
    DBMS_OUTPUT.PUT_LINE ('New employees are about to be added');
END;
```

Example 2

```
CREATE OR REPLACE TRIGGER trig1
AFTER INSERT ON T4
    REFERENCING NEW AS newRow
FOR EACH ROW    WHEN (newRow.a <= 10)
BEGIN
    INSERT INTO T5
        VALUES (:newRow.b, :newRow.a);
END trig1;
```