## PRACTICAL NO. 9

To create triggers for various events such as insertion, updation, etc.,

**PROCEDURE**

**a) PL/SQL Syntax:**

**TRIGGER**

A Trigger is a stored procedure that defines an action that the database automatically take when some database-related event such as Insert, Update or Delete occur.

**TRIGGER VS. PROCEDURE VS CURSOR**

| TRIGGER | PROCEDURES | CURSORS |
|---|---|---|
| These are named PL/SQL blocks. | These are named PL/SQL blocks. | These are named PL/SQL blocks. |
| These are invoked automatically. | User as per need invokes these. | These can be created both explicitly and implicitly. |
| These can‟t take parameters. | These can take parameters. | These can take parameters. |
| These are stored in database. | These are stored in database. | These are not stored in database. |

**TYPES OF TRIGGERS**

The various types of triggers are as follows,

- **Before**: It fires the trigger before executing the trigger statement.
- **After**: It fires the trigger after executing the trigger statement.
- **For each row**: It specifies that the trigger fires once per row.
- **For each statement**: This is the default trigger that is invoked. It specifies that the trigger fires once per statement.

**VARIABLES USED IN TRIGGERS**

- :new
- :old

These two variables retain the new and old values of the column updated in the database. The values in these variables can be used in the database triggers for data manipulation

**Row Level Trigger vs. Statement Level Trigger:**

| Row Level Trigger | Statement Level Trigger |
|---|---|
| These are fired for each row affected by the DML statement. | These are fired once for the statement instead of the no of rows modified by it. |
| These are used for generating/checking the values begin inserted or updated. | These are used for generated the summary information. |

**Before trigger vs. after trigger**

| Before Triggers | After Triggers |
|---|---|
| Before triggers are fired before the DML statement is actually executed. | After triggers are fired after the DML statement has finished execution. |

**Sytax:**

Create or replace trigger <trg_name> Before /After Insert/Update/Delete
[of column_name, column_name….]
on
<table_name>
[for each row]
[when
condition] begin
---statement
end;

**Q1: Create a trigger that insert current user into a username column of an existing table**

**b) Procedure for doing the experiment:**

| Step no. | Details of the step |
|---|---|
| 1 | Create a table itstudent4 with name and username as arguments |
| 2 | Create a trigger for each row that insert the current user as user name into a table |
| 3 | Execute the trigger by inserting value into the table |

**d) Program:**

```
SQL> create table itstudent4(name varchar2(15),username varchar2(15));
Table created.
SQL> create or replace trigger itstudent4 before insert on itstudent4 for each
 row 2  declare
 3  name varchar2(20);
 4  begin
 5  select user into name from
 dual; 6  :new.username:=name;
 7 end;
 8   /
```

Trigger created.

**e) Output**:

```
SQL> insert into itstudent4 values('&name','&username');
Enter value for name: akbar
Enter value for username: ranjani
old 1: insert into itstudent4 values('&name','&username')
new 1: insert into itstudent4 values('akbar','ranjani')
1 row
created.
SQL> /
Enter value for name: suji
Enter value for username:
priya
old 1: insert into itstudent4 values('&name','&username')
 new 1: insert into itstudent4 values('suji','priya')
 1 row created.
 SQL> select * from itstudent4;

 NAME          USERNAME
 --------------------
```

akbar SCOTT
suji SCOTT

**Q2: Create a Simple Trigger that does not allow Insert Update and Delete Operations on the Table**

**d) Program:**

**Table used:**

```
SQL> select * from itempls;
ENAME          EID
                SALARY
-------------------------------------------
xxx            11    10000
yyy            12    10500
zzz            13    15500
```

**Trigger:**

```
SQL> create trigger ittrigg before insert or update or delete on itempls for each row
  2 begin
  3 raise_application_error(-20010,'You cannot do manipulation');
  4 end;
  5
  6 /
Trigger created.
```

**e) Output:**

```
SQL> insert into itempls values('aaa',14,34000);
insert into itempls values('aaa',14,34000)
            *
ERROR at line
1:
ORA-20010: You cannot do manipulation
ORA-06512: at "STUDENT.ITTRIGG", line
2
ORA-04088: error during execution of trigger 'STUDENT.ITTRIGG'

SQL> delete from itempls where ename='xxx';
delete from itempls where ename='xxx'
            *
ERROR at line
1:
ORA-20010: You cannot do manipulation
ORA-06512: at "STUDENT.ITTRIGG", line
2
ORA-04088: error during execution of trigger 'STUDENT.ITTRIGG'
SQL> update itempls set eid=15 where ename='yyy';
update itempls set eid=15 where ename='yyy'
       *
ERROR at line 1:
ORA-20010: You cannot do manipulation
ORA-06512: at "STUDENT.ITTRIGG", line
2
ORA-04088: error during execution of trigger 'STUDENT.ITTRIGG'
```

**Q3: Create a Trigger that raises an User Defined Error Message and does not allow**

**updating and Insertion**

**Program:**

**Table used:**

```
SQL> select * from itempls;
ENAME          EID
                SALARY
---------------------------------
xxx            11    10000
yyy            12    10500
zzz            13    15500
```

**Trigger:**

```
SQL> create trigger ittriggs before insert or update of salary on itempls for each row
  2 declare
  3  triggsal itempls.salary%type;
  4  begin
  5  select salary into triggsal from itempls where eid=12;
  6 if(:new.salary>triggsal or :new.salary<triggsal) then
  7 raise_application_error(-20100,'Salary has not been changed');
  8 end if;
  9 end;
 10 /
Trigger created.
```

**Output:**

```
SQL> insert into itempls values ('bbb',16,45000);
insert into itempls values ('bbb',16,45000)
       *
ERROR at line 1:
ORA-04098: trigger 'STUDENT.ITTRIGGS' is invalid and failed re-validation
SQL> update itempls set eid=18 where ename='zzz';
update itempls set eid=18 where ename='zzz'
       *
ERROR at line 1:
ORA-04298: trigger 'STUDENT.ITTRIGGS' is invalid and failed re-validation
```

**Q4: develop a query to Drop the Created Trigger**

**Ans:**

```
SQL> drop trigger ittrigg;
Trigger dropped.
```

**e) Result**:

Thus the creation of triggers for various events such as insertion, updating, etc., was performed and executed successfully.

Consider the following Tables:

EMPLOYEE(Emp_id, EMP_name,Job_name,Manager_id,Hire_date,Salary,Deptno)

DEPARTMENT(Deptno, Dname, MGRSSN)

PROJECT(Pname,Pno,Plocation,Deptno)

| emp_id | emp_name | job_name | manager_id | hire_date | salary | E_Bonus | dep_no |
|---|---|---|---|---|---|---|---|
| 68319 | KAYLING | PRESIDENT | | 1991-11-18 | 6000.00 | 300.00 | 1001 |
| 66928 | BLAZE | MANAGER | 68319 | 1991-05-01 | 2750.00 | 200.00 | 3001 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 67832 | CLARE | MANAGER | 68319 | 1991-06-09 | 2550.00 | 200.00 | 1001 |
| 65646 | JONAS | MANAGER | 68319 | 1991-04-02 | 2957.00 | 200.00 | 2001 |
| 67858 | SCARLET | ANALYST | 65646 | 1997-04-19 | 3100.00 | 250.00 | 2001 |
| 69062 | FRANK | ANALYST | 65646 | 1991-12-03 | 3100.00 | 250.00 | 2001 |
| 63679 | SANDRINE | CLERK | 69062 | 1990-12-18 | 900.00 | 150.00 | 2001 |
| 64989 | ADELYN | SALESMAN | 66928 | 1991-02-20 | 1700.00 | 180.00 | 3001 |
| 65271 | WADE | SALESMAN | 66928 | 1991-02-22 | 1350.00 | 180.00 | 3001 |
| 66564 | MADDEN | SALESMAN | 66928 | 1991-09-28 | 1350.00 | 180.00 | 3001 |
| 68454 | TUCKER | SALESMAN | 66928 | 1991-09-08 | 1600.00 | 180.00 | 3001 |
| 68736 | ADNRES | CLERK | 67858 | 1997-05-23 | 1200.00 | 150.00 | 2001 |
| 69000 | JULIUS | CLERK | 66928 | 1991-12-03 | 1050.00 | 150.00 | 3001 |
| 69324 | MARKER | CLERK | 67832 | 1992-01-23 | 1400.00 | 150.00 | 1001 |

**Department Table**

| deptno | dname | Citylocation | dCountry |
|---|---|---|---|
| 1001 | Accounting | New York | United States of America, |
| 2001 | Research | Dallas | United States |
| 3001 | Sales | Chicago | United States of America |
| 4001 | Marketing | Los Angeles | United States |

**Project Table**

| Pno | Pname | PCitylocation | PCountry |
|---|---|---|---|
| 111 | P_1 | New York | United States of America, |
| 112 | P_2 | Dallas | United States |
| 113 | P_3 | Chicago | United States of America |
| 114 | P_4 | Denmark | northern Europe |
| 115 | P_5 | Paris | France |
| 116 | P_6 | Chicago | United States of America |

**Write a query for the following:-**