TRIGGERS:

* Triggers are a special type of procedure which is invoked/executed/fired automatically by the system when an event is occured.

*A trigger is associated with a table, views and executed when an event is occured.

* They provide High Security on tables. They are stored in "USER_TRIGGERS" system table.

ADVANTAGES/USES

> Used to control DDL,DML Operations.

> Used to perform some business logical conditions.

> Validates Data Before insertion/deletion/updation.

> Prevents invalid transactions.

* There are 2 types of triggers.

1.DML TRIGGERS

These triggers are invoked automatically by a system when an user performs some DML operations on a table.These are hadles by DB developers only.

DDL TRIGGERS(DB TRIGGERS)

These triggers are invoked automatically by a system when we performs DDL operations over database.(SERVERERROR,LOGON,LOGOFF,STARTUP,SHUTDOWN).

Trigger Parts:

1. Trigger Event :

- Indicates when to activate the Trigger

  Before

  Insert, Update, Delete = 6 Events

  After

- In case of after trigger first dml operations will be performed and then trigger will activate but in case of before trigger first trigger will activate then dml operations will be performed but both will give the same result

LEVELS OF TRIGGER/TRIGGER TYPE

1. **Statement Level Trigger/Table Level Trigger:**

- In this case trigger body is executed only once for DML command.
- If we have to insert 10 rows in a table then trigger body will be executes only once. It is bydefault.

2. **Row Level Trigger :**

- In this case trigger body is executed for each row wise in a table for DML command.
- If we have to insert 10 rows in a table then trigger body will be executes 10 times.

3. **Trigger Restriction :**

- It Supports to stop the Trigger Execution based on condition

BIND VARIABLES/ROWTYPE VARIABLES

:old,:new

- These are not allowed in table level(ststement level) Triggers.

Syntax:

```
Create or Replace Trigger <Trigger_name>
Before/After insert/update/delete
[ of <columns> ] on <table name>
[ for each row
  When <condition>  (TRUE -> Executes the trigger,
                     FALSE - Not execute)
  Declare
      <variable declaration>; ]
Begin
  <exec statements>;
  [ Exception
      <exec statements>; ]
End;
```

```
sql> insert into dept values(50,'economics','hyd')
      :dname='ECONOMICS'   :loc='hyd'   :depnto=40   -> sql buffer
```

Ex:

```
Create or Replace Trigger Dept_Trig
Before insert on dept
for each row
begin
 :new.dname:=upper(:new.dname);
 :NEW.LOC:=UPPER(:NEW.LOC);
End;
```

Write a Trigger program to take employee backup.
Create a backup table with same structure of EMP table.
Create table Backup as select * from emp where 1=2;
If where condition is false it copies Employee table structure only.

Sol:

Trigger Name: Backup_Trig

Table Name: EMP

Trigger Event: After Delete

```
Create or Replace Trigger Backup_Trig
After Delete on Emp
for each row
Begin
 insert into backup values(:old.empno,:old.ename,:old.job,:old.mgr,
                  :old.Hiredate,:old.sal,:old.comm,:old.deptno);
End;
```

```
SQL> Delete from Emp where Empno=7934;
sql> select * from Emp;
sql> Select * from Backup;
```

Example:

- **to increment employee salary automatically.**

incr:

empno	amount
7788	1000

Create a Increment Table:

```
Create table Incr(Empno Number(4), Amount Number(8,2));
```

Trigger Name: Incr_Trig
Table Name: Incr
Trigger Event: After Insert

Sol:

```
Create or Replace Trigger Incr_Trig

After Insert on Incr

for each row

Begin

 Update Emp set sal=sal+:new.Amount where Empno=:new.empno;

End;
```

Statement Level Triggers:

Create a Holiday table:

```
Create table Holiday(Hdate Date);
```

Insert into rows into Holiday table:

```
Insert into Holiday values('&hdate');
```

Ex:

Trigger Name: Holi_Trig
Table Name: Emp
Trigger Event: Before insert or update or delete

sol:

```
Create or Replace Trigger Holi_Trig

Before Insert or update or delete
```

```
on Emp
Declare
 cnt Number;
Begin
 if to_char(sysdate,'hh24') not between 10 and 16 then
     raise_application_error(-20001,'Offtimings, Trans. are Not allowed.');
 End if;

 if to_char(sysdate,'dy') in ('sat','sun') then
     raise_application_error(-20002,'Weekends, Trans. are Not Allowed.');
 end if;

 select count(hdate) into cnt from Holiday
               where to_char(sysdate,'dd/mm/yy')=to_char(hdate,'dd/mm/yy');
 if cnt>0 then
     raise_application_error(-20003,'Today Public Holiday, Trans. are Not
Allowed.');
 End if;
End;
```

Dropping Triggers:

sql> Drop trigger <triggername>;

Ex:

Drop trigger Holi_Trig;

- **If we delete a table then its corresponding triggers will be deleted automatically.**