PL/SQL - TRIGGERS

**Types of PL/SQL Triggers**

There are two types of triggers based on the which level it is triggered.

1) Row level trigger - An event is triggered for each row upated, inserted or deleted.

2) Statement level trigger - An event is triggered for each sql statement executed.

Create the 'product' table and 'product\_price\_history' table

*CREATE TABLE product\_price\_history*

*(product\_id number(5),*

*product\_name varchar2(32),*

*supplier\_name varchar2(32),*

*unit\_price number(7,2) );*

*CREATE TABLE product*

*(product\_id number(5),*

*product\_name varchar2(32),*

*supplier\_name varchar2(32),*

*unit\_price number(7,2) );*

*CREATE TABLE product\_check*

*(Message varchar2(50),*

*Current\_Date date*

*);*

**Insert few tuples into the product table.**

**1.** The price of a product changes constantly. It is important to maintain the history of the prices of the products. Create a trigger to update the 'product\_price\_history' table when the price of the product is updated in the 'product' table.

*CREATE or REPLACE TRIGGER price\_history\_trigger*

*BEFORE UPDATE OF unit\_price*

*ON product*

*FOR EACH ROW*

*BEGIN*

*INSERT INTO product\_price\_history*

*VALUES*

*(:old.product\_id,*

*:old.product\_name,*

*:old.supplier\_name,*

*:old.unit\_price);*

*END;*

*/*

2. Create a trigger which will insert a record into the table **'product\_check'** before each row is updated in the product table.

*CREATE or REPLACE TRIGGER Before\_Upddate\_Row\_product*

*BEFORE*

*UPDATE ON product*

*FOR EACH ROW*

*BEGIN*

*INSERT INTO product\_check*

*Values('Before update row level',sysdate);*

*END;*

*/*

3. Create a trigger which will insert a record into the table **'product\_check'** after each row is updated in the product table.

*CREATE or REPLACE TRIGGER After\_Update\_Row\_product*

*AFTER*

*insert On product*

*FOR EACH ROW*

*BEGIN*

*INSERT INTO product\_check*

*Values('After update, Row level',sysdate);*

*END;*

*/*

Now lets execute a update statement on table product.

*UPDATE PRODUCT SET unit\_price = 800*

*WHERE product\_id in (100,101);*

4. **BEFORE UPDATE, Statement Level:** This trigger will insert a record into the table 'product\_check' before a sql update statement is executed, at the **statement level**.

*CREATE or REPLACE TRIGGER Before\_Update\_Stat\_product*

*BEFORE*

*UPDATE ON product*

*Begin*

*INSERT INTO product\_check*

*Values('Before update, statement level',sysdate);*

*END;*

*/*