AIM:

The aim is to understand the concept of triggers in PL/SQL and implement them in simple programs

# BASIC TERMINOLOGIES:

## PL/SQL

**PL/SQL** stands for **Procedural Language** extension of SQL.PL/SQL is a combination of SQL along with the procedural features of programming languages. It was developed by Oracle Corporation in the early 90’s to enhance the capabilities of SQL.

***TRIGGERS***

Triggers are stored programs, which are automatically executed or fired when some events occur. Triggers are, in fact, written to be executed in response to any of the following events:

* A database manipulation (DML) statement (DELETE, INSERT, or UPDATE).
* A database definition (DDL) statement (CREATES, ALTER, or DROP).
* A database operation (SERVERERROR, LOGON, LOGOFF, STARTUP, or SHUTDOWN).

Triggers could be defined on the table, view, schema, or database with which the event is associated.

## PARTS OF A TRIGGER

A database trigger has three parts, namely, a trigger statement, a trigger body and a trigger restriction.

# Trigger Statement: -

A trigger statement specifies the DML statements like update, delete and insert and it fires the trigger body. It also specifies the table to which the trigger is associated.

# Trigger Body: -

Trigger body is a PL/SQL block that is executed when a triggering statement is issued.

# Trigger Restriction: -

Restrictions on a triggers can be achieved using the WHEN clause as shown in the syntax for creating triggers. They can be included in the definition of a row trigger, where in, the condition in the WHEN clause is evaluated for each row that is affected by the trigger.

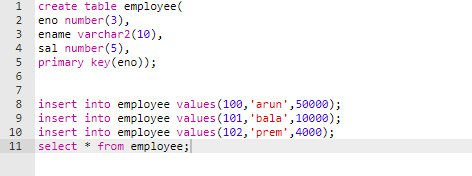
## Syntax for creating a trigger

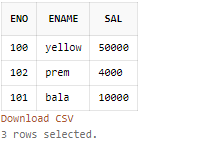
Create or Replace trigger <trigger\_name> [before/after] [insert/update/delete] on <table\_name> [for each statement/for each row]

[when <condition>] PL/SQL block;

# EXERCISES:

# Create an employee table





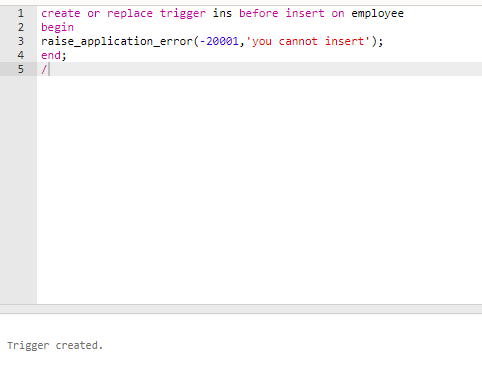
1. **PL/SQL program to create a trigger before the user inserts the data into the table.**

SQL> create or replace trigger ins2 before insert on employee

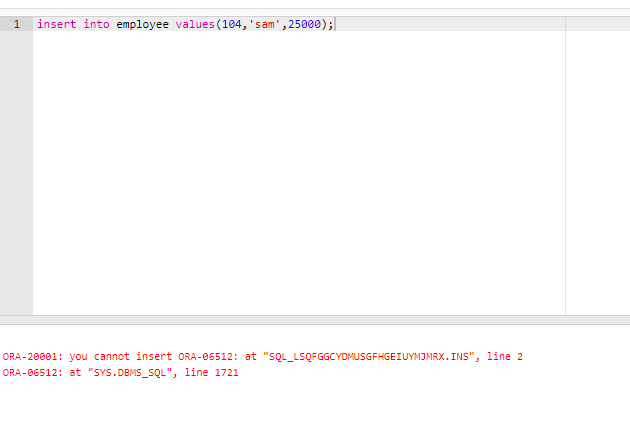
* 1. begin
  2. raise\_application\_error(-20001,'you cant insert a row');
  3. end;

5 /

Trigger created.



**OUTPUT**



# Write a PL/SQL program to create a trigger before the user deletes the data from the table*.*

SQL> create or replace trigger del2 before delete on employee

1. begin
2. raise\_application\_error(-20001,'you cant delete a row');
3. end;

5 /

Trigger created.

# 

# OUTPUT

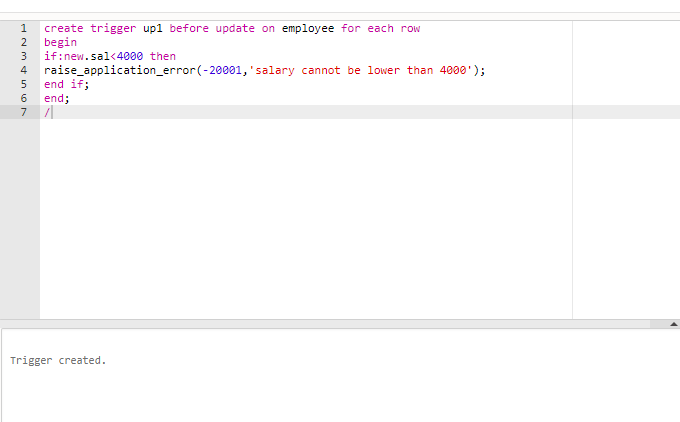
# 

# 3)Write a PL/SQL program to create a trigger before the user changes the value of the salary of an employee

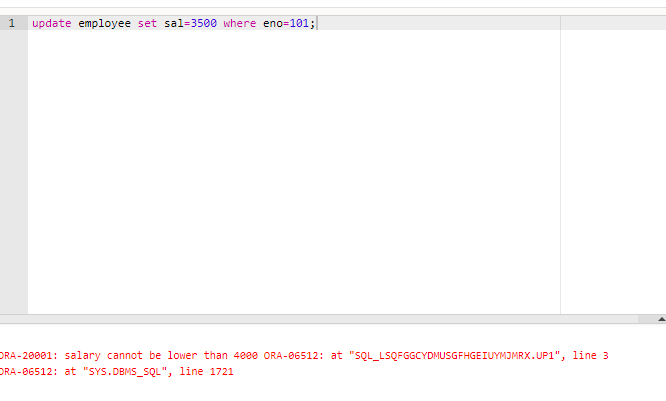
SQL> create trigger up1 before update on employee for each row

1. begin
2. if:new.sal<4000 then
3. raise\_application\_error(-20001,'salary cant be lower than this');
4. end if;
5. end;

7 /

Trigger created. 

# OUTPUT



# RESULT:

Thus the usage of triggers were studied and executed in RDBMS.