Group A

Assignment No 8

Title of the Assignment: Database Trigger (All Types: Row level and Statement level triggers, Before and After Triggers).

Write a database trigger on Library table. The System should keep track of the records that are being updated or deleted. The old value of updated or deleted records should be added in Library_Audit table.

Objective of the Assignment: To understand the concept of Triggers and its types.

Outcome: Students will be able to learn and understand All types: All Types: Row level and Statement level triggers, Before and After Triggers.

Theory:

A PL/SQL trigger is a named database object that encapsulates and defines a set of actions that are to be performed in response to an insert, update, or delete operation against a table. Triggers are created using the PL/SQL CREATE TRIGGER statement.

Types of triggers (PL/SQL)

The data server supports row-level and statement-level triggers within a PL/SQL context.

- A row-level trigger fires once for each row that is affected by a triggering event. For
 example, if deletion is defined as a triggering event for a particular table, and a single
 DELETE statement deletes five rows from that table, the trigger fires five times,
 once for each row.
- A statement-level trigger fires only once for each statement. Using the previous
 example, if deletion is defined as a triggering event for a particular table, and a single
 DELETE statement deletes five rows from that table, the trigger fires once.
 Statement-level trigger granularity cannot be specified for BEFORE triggers or
 INSTEAD OF triggers.

Trigger variables (PL/SQL)

NEW and OLD are special variables that you can use with PL/SQL triggers without explicitly defining them.

- NEW is a pseudo-record name that refers to the new table row for insert and update operations in row-level triggers. Its usage is : NEW.column, where column is the name of a column in the table on which the trigger is defined.
- OLD is a pseudo-record name that refers to the old table row for update and delete operations in row-level triggers. Its usage is :OLD.column, where column is the name of a column in the table on which the trigger is defined.

Trigger event predicates (PL/SQL)

The trigger event predicates, UPDATING, DELETING, and INSERTING can only be used in a trigger to identify the event that activated the trigger.

OR REPLACE

Specifies to replace the definition for the trigger if one exists at the current server. The existing definition is effectively dropped before the new definition is replaced in the catalog. This option is ignored if a definition for the trigger does not exist at the current server.

trigger-name

Names the trigger. The name, including the implicit or explicit schema name, must not identify a trigger already described in the catalog (SQLSTATE 42710). If a two-part name is specified, the schema name cannot begin with 'SYS' (SQLSTATE 42939).

BEFORE

Specifies that the associated triggered action is to be applied before any changes caused by the actual update of the subject table are applied to the database.

AFTER

Specifies that the associated triggered action is to be applied after the changes caused by the actual update of the subject table are applied to the database.

INSTEAD OF

Specifies that the associated triggered action replaces the action against the subject view.

trigger-event

Specifies that the triggered action associated with the trigger is to be executed whenever one of the events is applied to the subject table. Any combination of the events can be specified, but each event (INSERT, DELETE, and UPDATE) can only be specified once (SQLSTATE 42613).

INSERT

Specifies that the triggered action associated with the trigger is to be executed whenever an INSERT operation is applied to the subject table.

DELETE

Specifies that the triggered action associated with the trigger is to be executed whenever a DELETE operation is applied to the subject table.

UPDATE

Specifies that the triggered action associated with the trigger is to be executed whenever an UPDATE operation is applied to the subject table, subject to the columns specified or implied. If the optional column-name list is not specified, every column of the table is implied. Therefore, omission of the column-name list implies that the trigger will be activated by the update of any column of the table.

OF column-name,...

Each column-name specified must be a column of the base table (SQLSTATE 42703). If the trigger is a BEFORE trigger, the column-name specified cannot be a generated column other than the identity column (SQLSTATE 42989). No column-name can appear more than once in the column-name list (SQLSTATE 42711). The trigger will only be activated by the update of a column that is identified in the column-name list. This clause cannot be specified for an INSTEAD OF trigger (SQLSTATE 42613).

ON table-name

Designates the subject table of the BEFORE trigger or AFTER trigger definition. The name must specify a base table or an alias that resolves to a base table (SQLSTATE 42704 or 42809). The name must not specify a catalog table (SQLSTATE 42832), a materialized query table (SQLSTATE 42997), a created temporary table, a declared temporary table (SQLSTATE 42995), or a nickname (SQLSTATE 42809).

REFERENCING

Specifies the correlation names for the *transition variables*. Correlation names identify a specific row in the set of rows affected by the triggering SQL operation. Each row affected by the triggering SQL operation is available to the triggered action by qualifying columns with correlation-names specified as follows.

OLD AS correlation-name

Specifies a correlation name that identifies the row state prior to the triggering SQL operation. If the trigger event is INSERT, the values in the row are null values.

NEW AS correlation-name

Specifies a correlation name that identifies the row state as modified by the triggering SQL operation and by any SET statement in a BEFORE trigger that has already executed. If the trigger event is DELETE, the values in the row are null values.

If the REFERENCING clause is not invoked, then trigger variables NEW and OLD can optionally be used without explicitly defining them.

FOR EACH ROW

Specifies that the triggered action is to be applied once for each row of the subject table that is affected by the triggering SQL operation.

FOR EACH STATEMENT

Specifies that the triggered action is to be applied only once for the whole statement.

WHEN

(search-condition)

Specifies a condition that is true, false, or unknown. The search-condition provides a capability to determine whether or not a certain triggered action should be executed. The associated action is performed only if the specified search condition evaluates as true.

declaration

Specifies a variable declaration.

statement or handler-statement

Specifies a PL/SQL program statement. The trigger body can contain nested blocks.

condition

Specifies an exception condition name, such as NO_DATA_FOUND.

4Program

```
delimiter //
create trigger tr_ins77_Borrower
before insert
on table1
for each row
begin
insert into table2 values(new.rollno,new.name,new.nameofbook);
end
//
```

```
mysql> insert into table1 values(1,'gg','ggh');
 ->//
Query OK, 1 row affected (0.06 sec)
mysql> select * from table2;
 ->//
| rollno | name | nameofbook |
+----+
1 row in set (0.00 sec)
delimiter //
create trigger tr_del_Borrower
before delete
on table1
for each row
begin
delete from table2 where rollno=old.rollno;
end
//
mysql> delete from table1 where rollno=1;
 -> //
Query OK, 1 row affected (0.07 sec)
mysql> select * from table2;
 -> //
+----+
| rollno | name | nameofbook |
+----+
   2 | komal | oop
+----+
1 row in set (0.00 sec)
delimiter //
create trigger tr_up1_Borrower
before update
on table1
for each row
begin
update table2 set rollno=new.rollno where rollno=old.rollno;
end
//
```

Conclusion: Performed implementation of Triggers in PL/SQL successfully.

Viva Question:

- What is trigger explain with examples?
- Write is advantages of trigger?
- Write is application of trigger?
- What is before and after the event in Triggers?
- What are the types for trigger?

Date:	
Marks obtained:	
Sign of course coordinator:	
Name of course Coordinator:	