Group: - 1 from T4 Batch
0077 – Aakash Joshi
2039 – Akanksha Lokhande

- **Title:**
  Database Trigger (All Types: Row level and Statement level triggers, Before and After Triggers).

- **Date of completion:**

- **Problem statement:**
  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.

- **Objectives:**
  1. To develop Database programming skills.
  2. To develop basic Database administration skills.

- **Software and Hardware Requirements:**
  Software – 64-bit Open-Source Linux/Windows, Oracle.
  Hardware – Computer ,Mouse ,Keyboard , CPU

- **Theory :**

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 (CREATE, ALTER, or DROP).
- A database operation (SERVERERROR, LOGON, LOGOFF, STARTUP, or SHUTDOWN).

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

➢ Benefits of Triggers-

Triggers can be written for the following purposes:

1. Generating some derived column values automatically
2. Enforcing referential integrity
3. Event logging and storing information on table access
4. Auditing
5. Synchronous replication of tables
6. Imposing security authorizations
7. Preventing invalid transactions

➢ 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.

➢ Creating Triggers-

The syntax for creating a trigger is :

```
CREATE [OR REPLACE ] TRIGGER trigger_name
{BEFORE | AFTER | INSTEAD OF }
{INSERT [OR] | UPDATE [OR] | DELETE}
[OF col_name]
ON table_name
[REFERENCING OLD AS o NEW AS n]
[FOR EACH ROW]
WHEN (condition)
DECLARE
 Declaration-statements
BEGIN
 Executable-statements
EXCEPTION
```

Exception-handling-statements

END;

Where,

- CREATE [OR REPLACE] TRIGGER trigger_name − Creates or replaces an existing trigger with the trigger_name.
- {BEFORE | AFTER | INSTEAD OF} − This specifies when the trigger will be executed. The INSTEAD OF clause is used for creating trigger on a view.
- {INSERT [OR] | UPDATE [OR] | DELETE} − This specifies the DML operation.
- [OF col_name] − This specifies the column name that will be updated.
- [ON table_name] − This specifies the name of the table associated with the trigger.
- [REFERENCING OLD AS o NEW AS n] − This allows you to refer new and old values for various DML statements, such as INSERT, UPDATE, and DELETE.
- [FOR EACH ROW] − This specifies a row-level trigger, i.e., the trigger will be executed for each row being affected. Otherwise, the trigger will execute just once when the SQL statement is executed, which is called a table level trigger.
- WHEN (condition) − This provides a condition for rows for which the trigger would fire. This clause is valid only for row-level triggers

CODE:

```
CREATE OR REPLACE TRIGGER library_after_insert
AFTER INSERT ON Library
FOR EACH ROW
BEGIN
  INSERT INTO Library_Audit VALUES (
    library_audit_seq.NEXTVAL,
    :NEW.book_id,
    :NEW.book_name,
    :NEW.author,
    'INSERT',
    SYSDATE
  );
END;
/

-- Before update trigger
CREATE OR REPLACE TRIGGER library_before_update
```

```
BEFORE UPDATE ON Library
FOR EACH ROW
BEGIN
  INSERT INTO Library_Audit VALUES (
    library_audit_seq.NEXTVAL,
    :OLD.book_id,
    :OLD.book_name,
    :OLD.author,
    'UPDATE',
    SYSDATE
  );
END;
/

-- After delete trigger
CREATE OR REPLACE TRIGGER library_after_delete
AFTER DELETE ON Library
FOR EACH ROW
BEGIN
  INSERT INTO Library_Audit VALUES (
    library_audit_seq.NEXTVAL,
    :OLD.book_id,
    :OLD.book_name,
    :OLD.author,
    'DELETE',
    SYSDATE
  );
END;
/
```

OUTPUT:

```
Library table:

+---------+---------------+-----------+
| BOOK_ID | BOOK_NAME     | AUTHOR    |
+---------+---------------+-----------+
```

```
| 1       | The Odyssey   | Homer     |
| 2       | The Great Gatsby | F. Scott Fitzgerald |
+---------+---------------+-----------+


Library_Audit table:

+---------+---------+---------------+-----------+-----------+-----------------
| AUDIT_ID | BOOK_ID | BOOK_NAME     | AUTHOR    | OPERATION | TIMESTAMP
+---------+---------+---------------+-----------+-----------+-----------------
| 1       | 2       | The Great Gatsby | F. Scott Fitzgerald | INSERT | 2022-
03-01 09:15:00 |
| 2       | 1       | The Odyssey   | Homer     | DELETE | 2022-03-01 09:45:00
+---------+---------+---------------+-----------+-----------+-----------------
```