Example 1:

Triggers(Creation of insert,update and delete trigger)

**AIM**

To create and test **Insert**, **Update**, and **Delete** triggers on a database table to track changes in data.

**Design Steps**

1. **Create Tables**
   * bus(busno,source,desitnation,coach\_type)
   * audit\_log(id,souce,datetime,action)
2. **Create Triggers**
   * insert\_trigger → Fires on INSERT
   * update\_trigger → Fires on UPDATE
   * delete\_trigger → Fires on DELETE
3. **Insert Data into bus**
4. **Perform Operations (Insert/Update/Delete)**
5. **Check the audit\_log table for triggered entries**

**Trigger Creation (MySQL):**

**Create bus table**

CREATE TABLE BUS

(BUSNO CHAR(10),

SOURCE CHAR(10),

DESTINATION CHAR(10),

Coach\_type char(10),

PRIMARY KEY(BUSNO));

Insert data into bus table

insert into bus values('A102','HYD','VZG','LUX');

insert into bus values('A103','HYD','BNGL','AC');

insert into bus values('A104','HYD','WNGL','LUX');

INSERT INTO BUS VALUES('TS123','HYD','CHENNAI','40');

Create bus audit table

CREATE TABLE BUS\_AUDIT1

(ID INT AUTO\_INCREMENT, SOURCE

CHAR(10) NOT NULL, CHANGEDON DATETIME DEFAULT NULL, ACTION CHAR(10) DEFAULT NULL,

PRIMARY KEY(ID));

**UPDATE Trigger**

Mysql>DELIMITER $$

Mysql>

CREATE TRIGGER BEFORE\_BUS\_UPDATE1 BEFORE UPDATE ON BUS

FOR EACH ROW

BEGIN

INSERT INTO BUS\_AUDIT1

SET action='update', source=OLD.source, changedon=NOW();

END $$

Mysql>

Mysql>

Mysql>UPDATE BUS SET SOURCE='KERALA' WHERE BUSNO='AP123'$$

Mysql>select \* from bus\_audit1; $$

**INSERT Trigger**

Mysql>CREATE TRIGGER BEFORE\_BUS\_INSERT BEFORE INSERT ON BUS

FOR EACH ROW

BEGIN

INSERT INTO BUS\_AUDIT1

SET action='Insert', source=NEW.source, changedon=NOW();

END $$

Mysql>INSERT INTO BUS VALUES('TS123','HYD','CHENNAI','40');

Mysql select \* from bus\_audit1; $$

**DELETE Trigger**

Mysql>CREATE TRIGGER BEFORE\_BUS\_DELETE1 BEFORE DELETE ON BUS

FOR EACH ROW

BEGIN

INSERT INTO BUS\_AUDIT1 (action,source,changedon)values('Delete', old.source,NoW());

END $$

Mysql>delete from bus where busno='A102'; $$

Mysql>select \* from bus\_audit1 ; $$

mysql> select \* from bus\_audit1;

+----+--------+---------------------+--------+

| ID | SOURCE | CHANGEDON | ACTION |

+----+--------+---------------------+--------+

| 11 | HYD | 2023-07-24 12:54:11 | Delete |

| 12 | HYD | 2023-07-24 12:55:42 | Insert |

| 13 | HYD | 2023-07-24 12:57:35 | update |

| 14 | HYD | 2023-07-24 12:57:35 | update |

| 15 | HYD | 2023-07-24 12:57:35 | update |

| 16 | HYD | 2023-07-24 15:18:17 | update |

| 17 | HYD | 2023-07-24 15:18:17 | update |

| 18 | HYD | 2023-07-24 15:18:17 | update |

+----+--------+---------------------+--------+

8 rows in set (0.01 sec)

**Test Cases:**

| **Test Case ID** | **Description** | **Operation** | **Expected Result in audit\_log** | **Actual Result** | **Status** |
| --- | --- | --- | --- | --- | --- |
| TC01 | Insert student record | INSERT | Row with action='INSERT' | As expected | Pass |
| TC02 | Update student name | UPDATE | Row with action='UPDATE' | As expected | Pass |
| TC03 | Delete student record | DELETE | Row with action='DELETE' | As expected | Pass |

Example 2:

Triggers(Creation of insert,update and delete trigger)

**AIM:**

To create and test **Insert**, **Update**, and **Delete** triggers on a database table to track changes in data.

**Design Steps**

1. **Create Tables:**
   * students(id, name, marks)
   * audit\_log(action, student\_id, name, marks, action\_time)
2. **Create Triggers:**
   * insert\_trigger → Fires on INSERT
   * update\_trigger → Fires on UPDATE
   * delete\_trigger → Fires on DELETE
3. **Insert Sample Data into students**
4. **Perform Operations (Insert/Update/Delete)**
5. **Check the audit\_log table for triggered entries**

**Trigger Creation (MySQL):**

-- Audit table

CREATE TABLE audit\_log (

action VARCHAR(10),

student\_id INT,

name VARCHAR(100),

marks INT,

action\_time TIMESTAMP DEFAULT CURRENT\_TIMESTAMP

);

-- INSERT Trigger

CREATE TRIGGER insert\_trigger

AFTER INSERT ON students

FOR EACH ROW

INSERT INTO audit\_log (action, student\_id, name, marks)

VALUES ('INSERT', NEW.id, NEW.name, NEW.marks);

-- UPDATE Trigger

CREATE TRIGGER update\_trigger

AFTER UPDATE ON students

FOR EACH ROW

INSERT INTO audit\_log (action, student\_id, name, marks)

VALUES ('UPDATE', NEW.id, NEW.name, NEW.marks);

-- DELETE Trigger

CREATE TRIGGER delete\_trigger

AFTER DELETE ON students

FOR EACH ROW

INSERT INTO audit\_log (action, student\_id, name, marks)

VALUES ('DELETE', OLD.id, OLD.name, OLD.marks);

**Test Cases:**

| **Test Case ID** | **Description** | **Operation** | **Expected Result in audit\_log** | **Actual Result** | **Status** |
| --- | --- | --- | --- | --- | --- |
| TC01 | Insert student record | INSERT | Row with action='INSERT' | As expected | Pass |
| TC02 | Update student name | UPDATE | Row with action='UPDATE' | As expected | Pass |
| TC03 | Delete student record | DELETE | Row with action='DELETE' | As expected | Pass |
| TC04 | Insert without trigger | INSERT (off) | No row in audit\_log | No entry | Pass |
| TC05 | Multiple updates | Multiple UPD | Multiple UPDATE entries | As expected | Pass |

**Result**

Triggers executed as expected, and corresponding entries were logged in the audit\_log table for each operation.