**EXPERIMENT 12**

**AIM:**

Study and implement the assigned topic from the Embedded SQL, PL SQL Concepts, Cursors, Stored Procedures, Stored Function, Triggers, Tuple relational calculus, domain relational calculus.

**INTRODUCTION:**

A trigger, in database terms, is a set of instructions that are 'fired' by some specific event, normally a command issued through the database's Data Manipulation Language (DML).

**SYNTAX:**

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;

**USE CASES FOR OUR DATABASE:**

1. **A trigger that can be used to update the empty vessel weight with the difference to get the quantity of product filled in.**

CREATE TRIGGER after\_update\_weights

AFTER UPDATE

ON Band\_Tag

FOR EACH ROW

BEGIN

IF

OLD.Vessel\_Weight<>new.Vessel\_Weight

THEN

SET new.Vessel\_Weight = OLD.Vessel\_weight - new.Vessel\_weight

END IF

END;

1. **Triggers can be used in our database to display the customer’s receipt when they are done shopping.**

CREATE TRIGGER before\_update\_item\_quantity

BEFORE UPDATE

ON groceries

FOR EACH ROW

BEGIN

INSERT INTO shopping\_list

VALUES(item\_id, item\_name, item\_quantity,item\_cost);

END;

1. **Triggers are also useful in auditing the database after every transaction.**

CREATE TRIGGER after\_update\_groceries

AFTER UPDATE

ON groceries

FOR EACH ROW

BEGIN

INSERT INTO backup

VALUES(NEW.item\_id, NEW.item\_name,

NEW.item\_category, NEW.item\_quantity,

NEW.item\_cost);

END;