WEEK 9-10

SQL – Creating Triggers and Functions

Problem Statement: **15*2 = 30 marks**

Write the SQL Triggers and functions for the following using Postgres sql.

1. **Create an employee table which contains employee details and the department he works for. Create another table department consisting of dname and number of employees. Write triggers to increment or decrement the number of employees in a department table when the record in the employee table is inserted or deleted respectively.**

```
triggers=# CREATE TABLE EMPLOYEE(
triggers(# empid int not null primary key,
triggers(# name varchar(255),
triggers(# phone_no varchar(20),
triggers(# email varchar(20),
triggers(# D_no int);
CREATE TABLE
triggers=# select * from employee;
 empid | name | phone_no | email | d_no
-------+------+----------+-------+------
(0 rows)


triggers=# create table department(
triggers(# dno int not null primary key,
triggers(# dnumber varchar(255),
triggers(# no_of_emp int,
triggers(# loc varchar(255));
CREATE TABLE
triggers=# alter table department
triggers-# rename column dnumber to dname;
ALTER TABLE
triggers=# select * from department;
 dno | dname | no_of_emp | loc
-----+-------+-----------+-----
(0 rows)


triggers=# alter table employee
triggers-# add foreign key(d_no) references department(d_no);
ERROR:  column "d_no" referenced in foreign key constraint does not exist
triggers=# alter table employee
triggers-# add foreign key(d_no) references department(dno);
ALTER TABLE
triggers=#
```

```
triggers=# create or replace function add_count() returns trigger as
triggers-# $BODY$
triggers$# BEGIN
triggers$# Update Department set no_of_emp = no_of_emp+1 where dno=new.dno;
triggers$# return new;
triggers$# END;
triggers$# $BODY$
triggers-# language plpgsql;
CREATE FUNCTION
```

```
triggers=# create trigger add_num
triggers-#     after insert on employee
triggers-#     for each row
triggers=# insert into employee(empid,name,phone_no,email,dno)
triggers-# values(3,'kamra','1234432112','kam@kmao.com',1);
INSERT 0 1
triggers=# select * from department;
 dno |    dname    | no_of_emp |    loc
-----+-------------+-----------+-----------
   2 | marketing   |         1 | chennai
   1 | engineering |         2 | bangalore
(2 rows)
```

```
triggers=# create or replace function del_count() returns trigger as
triggers-# $BODY$
triggers$# BEGIN
triggers$# UPDATE DEPARTMENT set no_of_emp = no_of_emp-1 where dno=old.dno;
triggers$# return new;
triggers$# END;
triggers$# $BODY$
triggers-# language plpgsql;
CREATE FUNCTION
triggers=# delete from employee where empid=2;
```

```
triggers=# create trigger delete_num
triggers-#     after delete on employee
triggers-#     FOR EACH ROW
triggers-#     EXECUTE PROCEDURE del_count();
CREATE TRIGGER
triggers=# delete from employee where empid=2;
DELETE 1
triggers=# SELECT * FROM DEPARTMENT;
 dno |    dname    | no_of_emp |    loc
-----+-------------+-----------+-----------
   1 | engineering |         2 | bangalore
   2 | marketing   |         3 | chennai
(2 rows)
```

2. **Create an order_item table which contains details like name, quantity and unit price of every item purchased. Create an order summary table that contains number of items and total price. Create triggers to update entry in order summary whenever an item is inserted or deleted in the order item table.**

```
triggers=# create table order_summary(
triggers(# order_id int,
triggers(# no_of_items int,
triggers(# unit_price int);
CREATE TABLE
triggers=#
```

```
triggers=# create table order_item(
triggers(# item_id int not null primary key,
triggers(# name varchar(255),
triggers(# qty int,
triggers(# unit_price int);
CREATE TABLE
triggers=# create table order_summary(
```

```
triggers=# create or replace function add_ctr() returns trigger as
triggers-# $$
triggers$# begin
triggers$# update order_summary set no_of_items = no_of_items+1 where order_id=new.order_id;
triggers$# return new;
triggers$# end;
triggers$# $$
triggers-# language plpgsql;
CREATE FUNCTION
```

```
triggers=# create trigger add_ctr
triggers-#     after insert on order_item
triggers-#     for each row
triggers-#     execute procedure add_ctr();
CREATE TRIGGER
triggers=# insert into order_item(item_id,name,qty,unit_price,order_id)
triggers-# values(7,'pencil_case',2,50,1);
INSERT 0 1
triggers=# select * from order_summary;
 order_id | no_of_items | unit_price
----------+-------------+------------
        2 |           3 |        540
        1 |           4 |        180
(2 rows)
```

```
triggers=# create or replace function del_ctr() returns trigger as
triggers-# $$
triggers$# begin
triggers$# update order_summary set no_of_items = no_of_items-1 where order_id=old.order_id;
triggers$# return new;
triggers$# end
triggers$# ;
triggers$# $$
triggers-# language plpgsql;
CREATE FUNCTION
triggers=# create trigger del_ctr
triggers-#     after delete on order_item
triggers-#     for each row
triggers-#     execute procedure del_ctr();
CREATE TRIGGER
triggers=# delete from order_item where item_id=7;
DELETE 1
triggers=# select * from order_summary;
 order_id | no_of_items | unit_price
----------+-------------+------------
        2 |           3 |        540
        1 |           3 |        180
(2 rows)
```