

# DBMS Lab 9 & 10 - Triggers and Functions

KARTIKA NAIR

PES1UG19CS213

09TH NOVEMBER 2021

## 1. Triggers for employee and department tables

Creation of database

```
pes1ug19cs213@pes1ug19cs213: ~  
postgres=# create database company;  
CREATE DATABASE  
postgres=# \c company  
You are now connected to database "company" as user "postgres".
```

Creation of employee table

```
company=# create table employee (fname VARCHAR(15) NOT NULL, lname VARCHAR(15) NOT NULL, essn CHAR(10) NOT NULL, dname VARCHAR(15), PRIMARY KEY (essn), FOREIGN KEY (dname) REFERENCES department(dname));  
CREATE TABLE  
company=# \d employee  
Table "public.employee"  
Column | Type | Collation | Nullable | Default  
-----+-----+-----+-----+-----  
fname | character varying(15) | | not null |  
lname | character varying(15) | | not null |  
essn | character(10) | | not null |  
dname | character varying(15) | | |  
Indexes:  
"employee_pkey" PRIMARY KEY, btree (essn)  
Foreign-key constraints:  
"employee_dname_fkey" FOREIGN KEY (dname) REFERENCES department(dname)
```

## Creation of department table

```
company=# create table department (dname VARCHAR(15), emp_count INT, PRIMARY KEY
(dname));
CREATE TABLE
company=# \d department
          Table "public.department"
  Column      |      Type       | Collation | Nullable | Default
-----+-----+-----+-----+-----
 dname       | character varying(15) |           | not null |
 emp_count   | integer          |           |          |
Indexes:
    "department_pkey" PRIMARY KEY, btree (dname)
```

## Creating increment function and trigger

```
pes1ug19cs213@pes1ug19cs213: ~
company=# create function inc_213() returns trigger as $$
begin
update department set emp_count = emp_count + 1
where department.dname = NEW.dname;
return NEW;
end;
$$ LANGUAGE plpgsql;
CREATE FUNCTION
company=# create trigger increment_213
after insert on employee
for each row
execute procedure inc_213();
CREATE TRIGGER
```

## Creating decrement function and trigger

```
company=# create function dec_213() returns trigger as $$
begin
update department set emp_count = emp_count - 1
where department.dname = OLD.dname;
return OLD;
end;
$$ LANGUAGE plpgsql;
CREATE FUNCTION
company=# create trigger decrement_213
after delete on employee
for each row
execute procedure dec_213();
CREATE TRIGGER
```

## Inserting values into employee and department tables

```
pes1ug19cs213@pes1ug19cs213: ~  
company=# INSERT into department values ('Poison', 0);  
INSERT into department values ('Serum', 0);  
INSERT 0 1  
INSERT 0 1  
company=# INSERT into employee values ('Henry', 'Jekyll', '1234567890', 'Poison'  
);  
INSERT into employee values ('Edward', 'Hyde', '0123456789', 'Poison');  
INSERT into employee values ('Gabriel', 'Utterson', '8264910382', 'Serum');  
INSERT 0 1  
INSERT 0 1  
INSERT 0 1  
company=#
```

## Incremented values displayed in department table

```
pes1ug19cs213@pes1ug19cs213: ~  
company=# select * from department;  
  dname | emp_count  
-----+-----  
 Poison |          2  
  Serum |          1  
(2 rows)
```

## Decrementing values displayed in department table

```
pes1ug19cs213@pes1ug19cs213: ~  
company=# delete from employee where dname = 'Serum';  
DELETE 1  
company=# select * from department;  
  dname | emp_count  
-----+-----  
 Poison |          2  
  Serum |          0  
(2 rows)
```

## 2. Triggers for order item and order summary tables

Creation of database, order\_summary table, and order\_item table

```
postgres=# create database ord;
CREATE DATABASE
postgres=# \c ord
You are now connected to database "ord" as user "postgres".
ord=# create table order_summary (numItems INT, totalPrice INT, ID VARCHAR(5), PRIMARY KEY (ID));
CREATE TABLE
ord=# \d order_summary
Table "public.order_summary"
  Column      |      Type       | Collation | Nullable | Default |
-----+-----+-----+-----+-----+
numitems      | integer         |           |          |         |
totalprice    | integer         |           |          |         |
id            | character varying(5) |           | not null |         |
Indexes:
    "order_summary_pkey" PRIMARY KEY, btree (id)
ord=# create table order_item (name VARCHAR(15), qty INT, unitPrice INT, ID VARCHAR(5), PRIMARY KEY (name), FOREIGN KEY (ID) REFERENCES order_summary(ID));
CREATE TABLE
ord=# \d order_item
Table "public.order_item"
  Column      |      Type       | Collation | Nullable | Default |
-----+-----+-----+-----+-----+
name          | character varying(15) |           | not null |         |
qty           | integer         |           |          |         |
unitprice     | integer         |           |          |         |
id            | character varying(5) |           |          |         |
Indexes:
    "order_item_pkey" PRIMARY KEY, btree (name)
Foreign-key constraints:
    "order_item_id_fkey" FOREIGN KEY (id) REFERENCES order_summary(id)
```

Creation of increment and decrement functions and triggers

```
ord=# create function inc_213() returns trigger as $$
begin
update order_summary set totalPrice = totalPrice + (NEW.qty * NEW.unitPrice)
where order_summary.ID = NEW.ID;
update order_summary set numItems = numItems + 1
where order_summary.ID = NEW.ID;
return NEW;
end;
$$ LANGUAGE plpgsql;
CREATE FUNCTION
ord=# create trigger increment_213
after insert on order_item
for each row
execute procedure inc_213();
CREATE TRIGGER
ord=# create function dec_213() returns trigger as $$
begin
update order_summary set totalPrice = totalPrice - (OLD.qty * OLD.unitPrice)
where order_summary.ID = OLD.ID;
update order_summary set numItems = numItems + 1
where order_summary.ID = OLD.ID;
return OLD;
end;
$$ LANGUAGE plpgsql;
CREATE FUNCTION
ord=# create trigger decrement_213
after delete on order_item
for each row
execute procedure dec_213();
CREATE TRIGGER
ord=#
```

## Insertion of values into tables, displaying incremented values

```
pes1ug19cs213@pes1ug19cs213: ~  
ord=# INSERT into order_summary values (0, 0, '12345');  
INSERT into order_summary values (0, 0, '21345');  
INSERT into order_summary values (0, 0, '31245');  
INSERT 0 1  
INSERT 0 1  
INSERT 0 1  
ord=# INSERT into order_item values ('Hyde', 1, 2, '12345');  
INSERT into order_item values ('Jekyll', 2, 1, '12345');  
INSERT into order_item values ('Enfield', 3, 4, '21345');  
INSERT into order_item values ('Utterson', 5, 3, '31245');  
INSERT 0 1  
INSERT 0 1  
INSERT 0 1  
INSERT 0 1  
ord=# select * from order_summary;  
 numitems | totalprice | id  
-----+-----+-----  
         2 |          4 | 12345  
         1 |         12 | 21345  
         1 |         15 | 31245  
(3 rows)
```

## Displaying decremented values

```
pes1ug19cs213@pes1ug19cs213: ~  
ord=# delete from order_item where name = 'Utterson';  
DELETE 1  
ord=# select * from order_summary;  
 numitems | totalprice | id  
-----+-----+-----  
         2 |          4 | 12345  
         1 |         12 | 21345  
         2 |          0 | 31245  
(3 rows)
```