**1./\*Create view vw\_updatable\_products** (use same query whatever

I used in the training)

Try updating view with below query and see if the product

table also gets updated.

Update query:

UPDATE updatable\_products SET unit\_price = unit\_price \* 1.1 WHERE

units\_in\_stock < 10;\*/

select \* from products;

**QUERY:**

CREATE view vw\_updatable\_products AS

select

product\_id,

product\_name,

unit\_price,

units\_in\_stock,

discontinued

from products

where discontinued=0

with CHECK option;

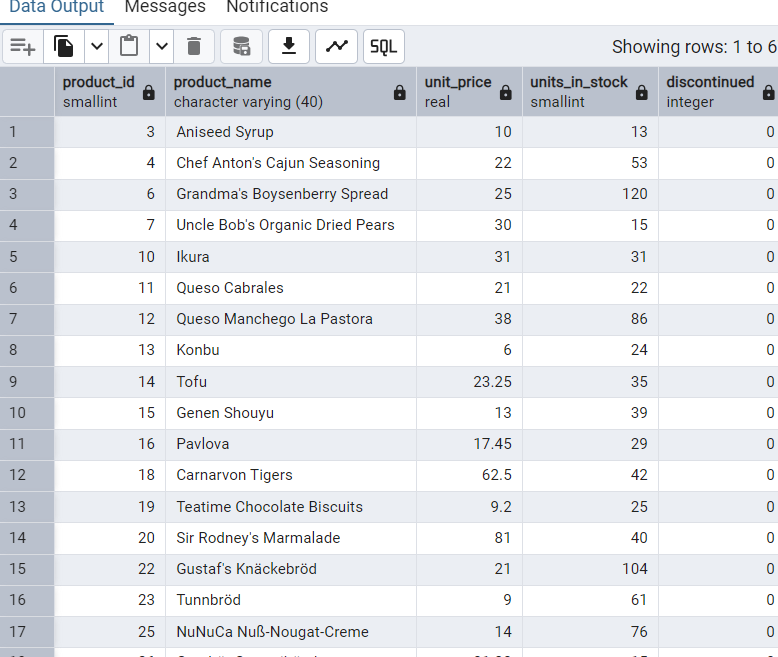
UPDATE vw\_updatable\_products

set unit\_price=unit\_price\*1.1

where units\_in\_stock<10;

select \* from vw\_updatable\_products;

**OUTPUT:**



**2. /\* Transaction:**

Update the product price for products by 10% in category id=1

Try COMMIT and ROLLBACK and observe what happens.\*/

**QUERY:**

select \* from products;

BEGIN;

update products

set unit\_price=unit\_price\*1.10

where category\_id=1;

commit;

rollback;

**OUTPUT:**

****

**3./\* Create a regular view which will have below details**

**(Need to do joins):**

**Employee\_id,**

**Employee\_full\_name,**

**Title,**

**Territory\_id,**

**territory\_description,**

**region\_description\*/**

**QUERY:**

**select \* from employees;**

**select \* from employee\_territories;**

**select \* from region;**

**create VIEW vw\_employee\_territory AS**

**select**

**e.employee\_id,**

**e.first\_name||''||e.last\_name as fullname,**

**e.title,**

**t.territory\_id,**

**t.territory\_description,**

**r.region\_description**

**from employees e**

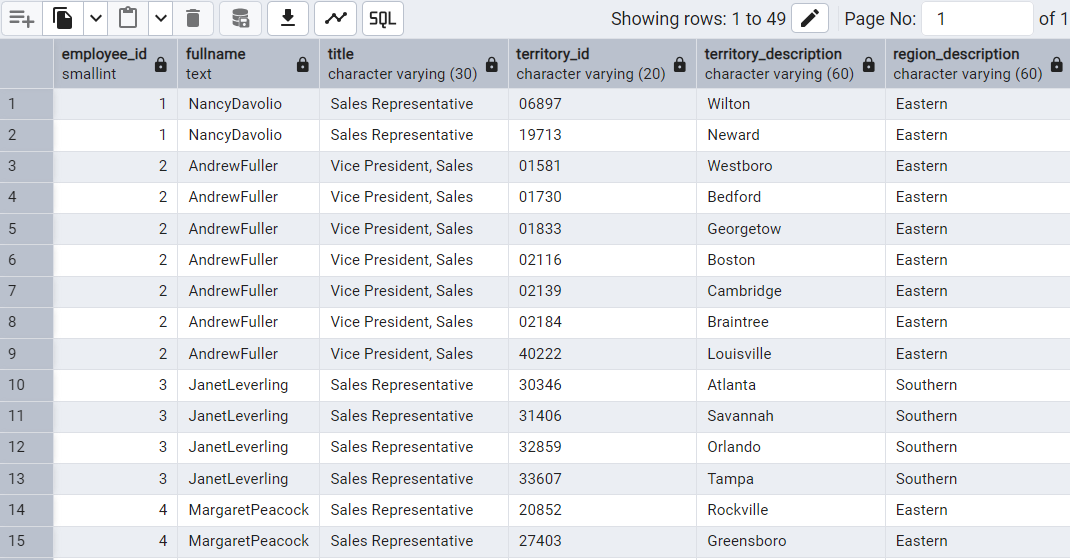
**JOIN employee\_territories et ON et.employee\_id=e.employee\_id**

**join territories t ON t.territory\_id = et.territory\_id**

**JOIN region r ON r.region\_id = t.region\_id;**

**select \* from vw\_employee\_territory;**

**OUTPUT:**

****

**4./\*Create a recursive CTE based on Employee Hierarchy\*/**

**QUERY:**

**with recursive cte\_employee\_hierarchy AS(**

**select**

**employee\_id,**

**first\_name,**

**last\_name,**

**reports\_to,0 as level**

**from employees e**

**where reports\_to is null**

**union all**

**select**

**e.employee\_id,**

**e.first\_name,**

**e.last\_name,**

**e.reports\_to,eh.level+1**

**from employees e**

**join cte\_employee\_hierarchy eh ON eh.employee\_id=e.reports\_to)**

**select**

**level,**

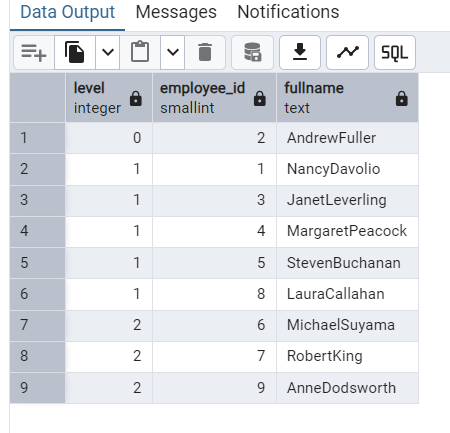
**employee\_id,**

**first\_name||''||last\_name as fullname**

**from cte\_employee\_hierarchy**

**order by level,employee\_id;**

**OUTPUT:**

****