**Day 8**

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;

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;

SELECT \* FROM vw\_updatable\_products;

UPDATE vw\_updatable\_products

SET unit\_price = unit\_price \* 1.1

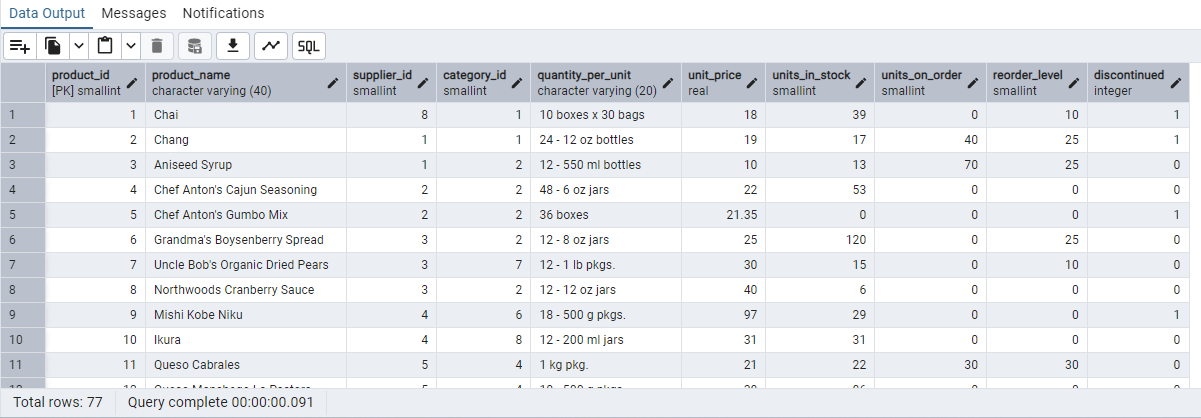
WHERE units\_in\_stock < 10

2. Transaction:

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

Try COMMIT and ROLLBACK and observe what happens.

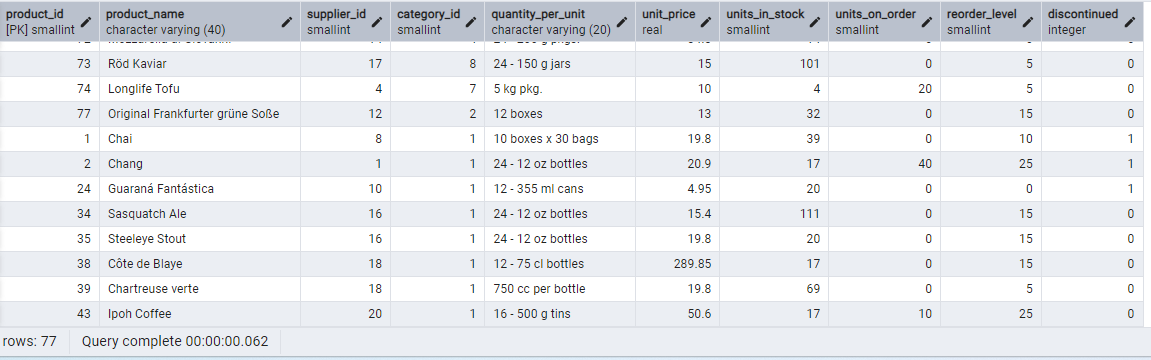
select \* from products;



UPDATE products

SET unit\_price = unit\_price \* 1.10

WHERE category\_id = 1;



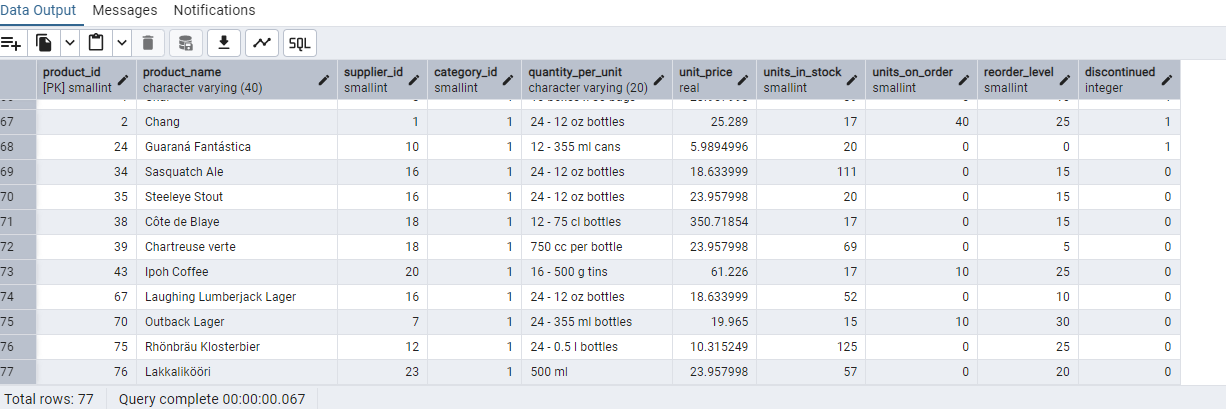
With commit-

UPDATE products

SET unit\_price = unit\_price \* 1.10

WHERE category\_id = 1;

Commit;



With rollback;

UPDATE products

SET unit\_price = unit\_price \* 1.10

WHERE category\_id = 1;

Rollback;

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

CREATE VIEW vw\_employee AS

select e.employee\_id,CONCAT(e.first\_name, ' ', e.last\_name) AS employee\_full\_name,e.title,t.territory\_id,

t.territory\_description,r.region\_description

from employees e

join employee\_territories et ON e.employee\_id = et.employee\_id

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

join region r ON t.region\_id = r.region\_id;

SELECT \* FROM vw\_employee;

4. Create a recursive CTE based on Employee Hierarchy

WITH RECURSIVE cte\_employeehierarchy AS (

SELECT

employee\_id,

first\_name,

last\_name,

reports\_to,

0 AS level

FROM employees

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\_employeehierarchy eh

ON e.reports\_to = eh.employee\_id

)

SELECT \* FROM cte\_employeehierarchy;

