**Day 3**

**USE Northwind from Kaggle:**

1. Update the categoryName From “Beverages” to "Drinks" in the categories table.

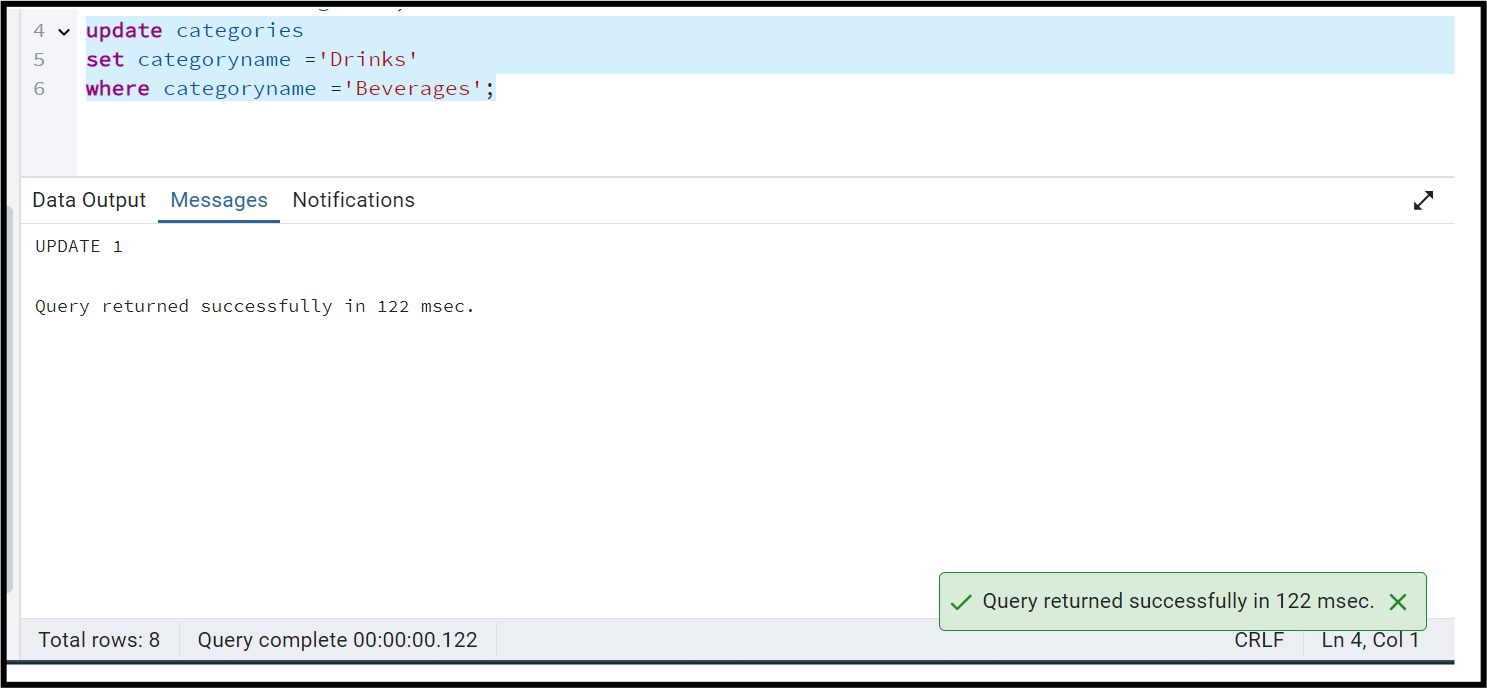
**Query**:

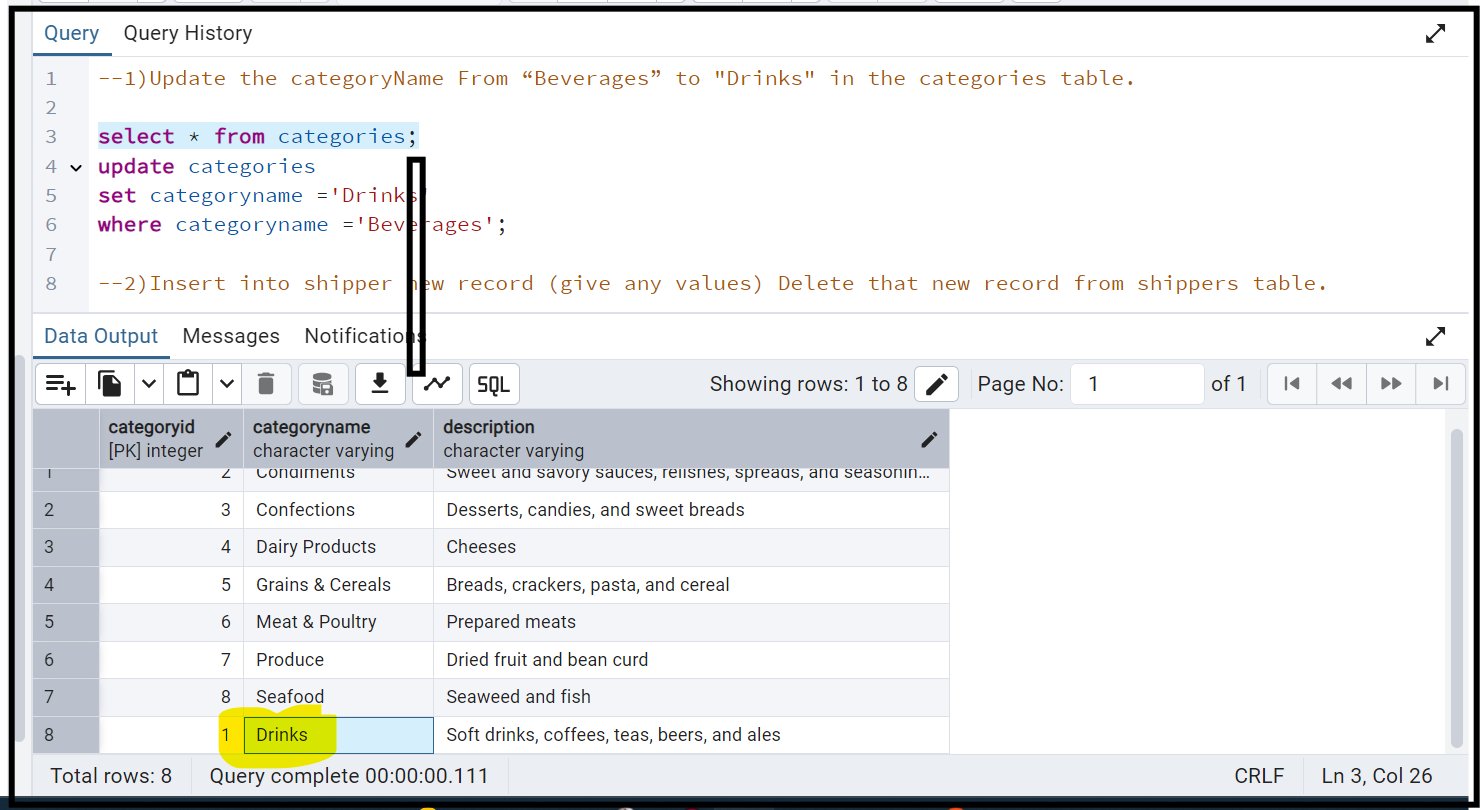
**update categories**

**set categoryname ='Drinks'**

**where categoryname ='Beverages';**

**Screenshot**:





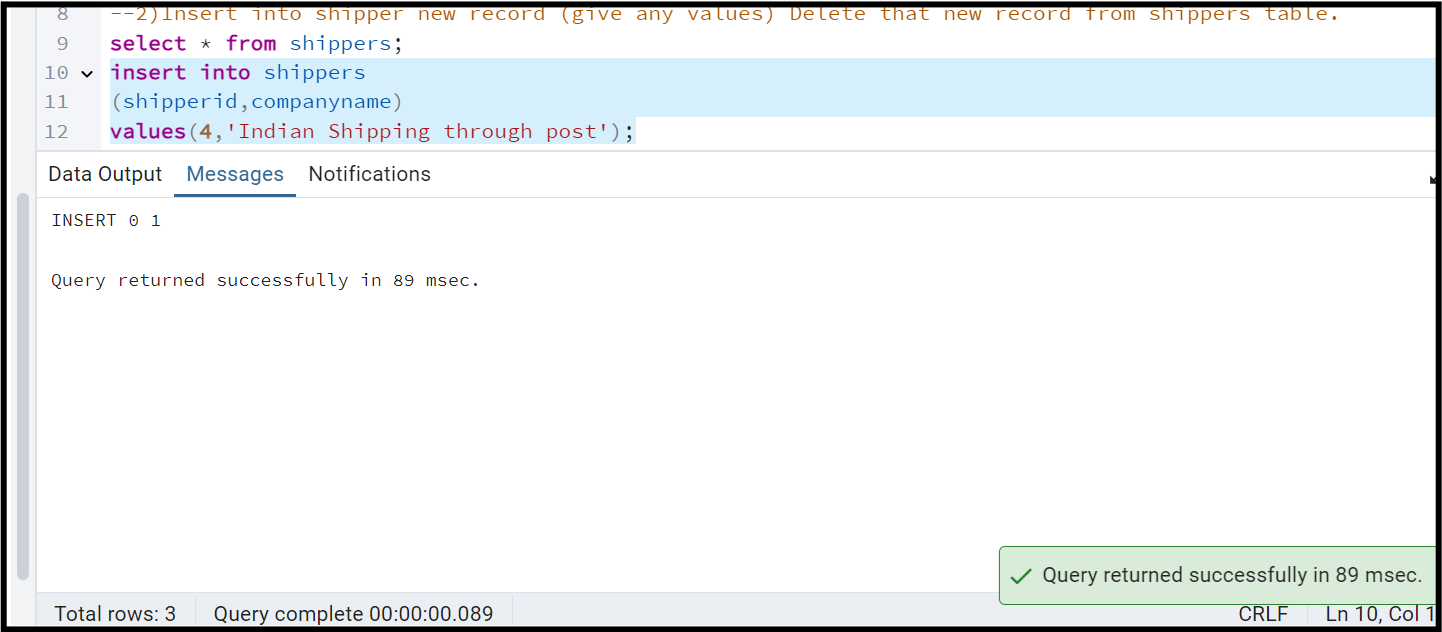
1. Insert into shipper new record (give any values) Delete that new record from shippers table.

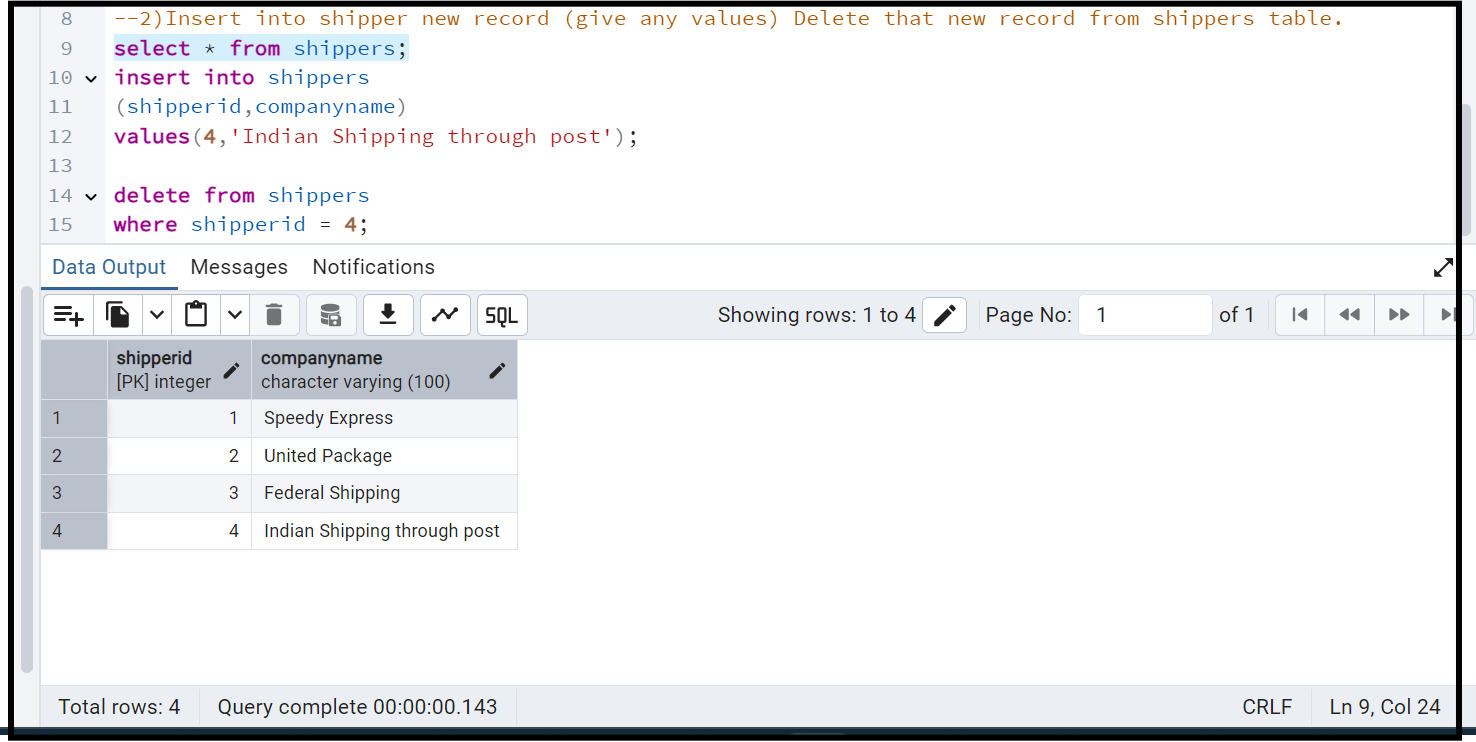
**Query**:

**insert into shippers**

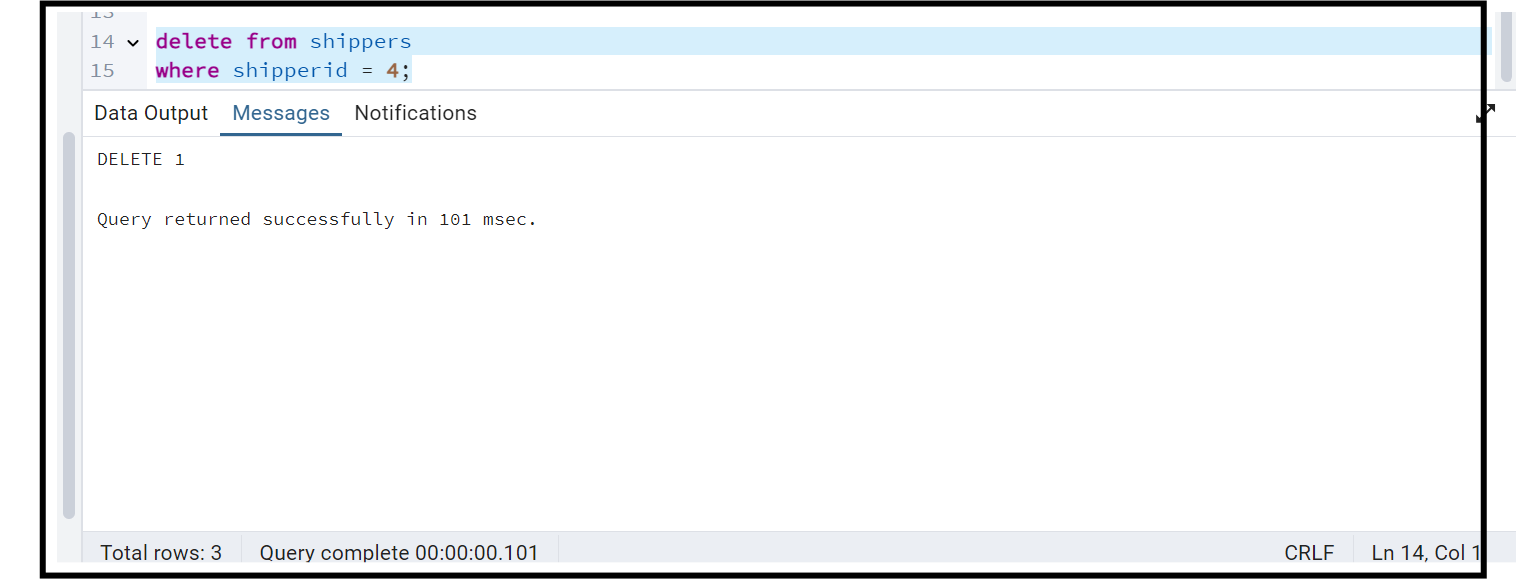
**(shipperid, companyname)**

**values(4,'Indian Shipping through post');**

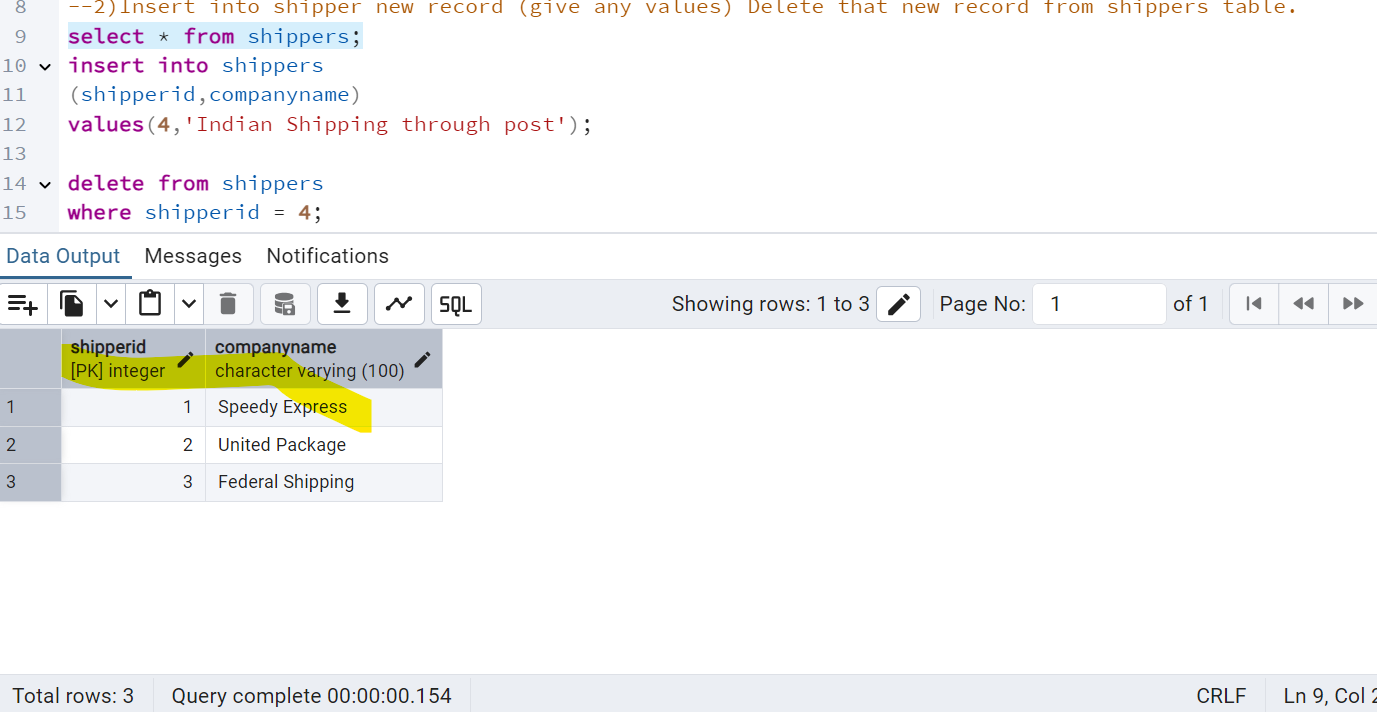




To delete:



After delete:

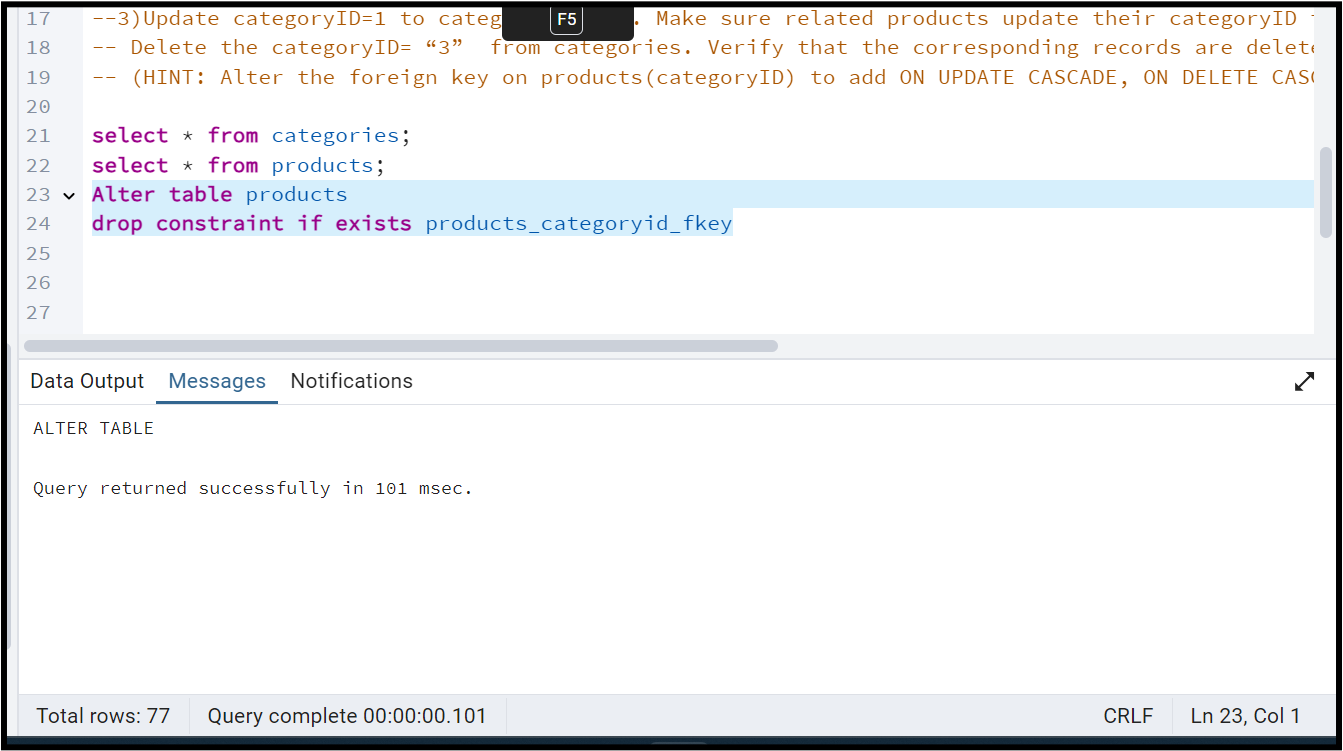


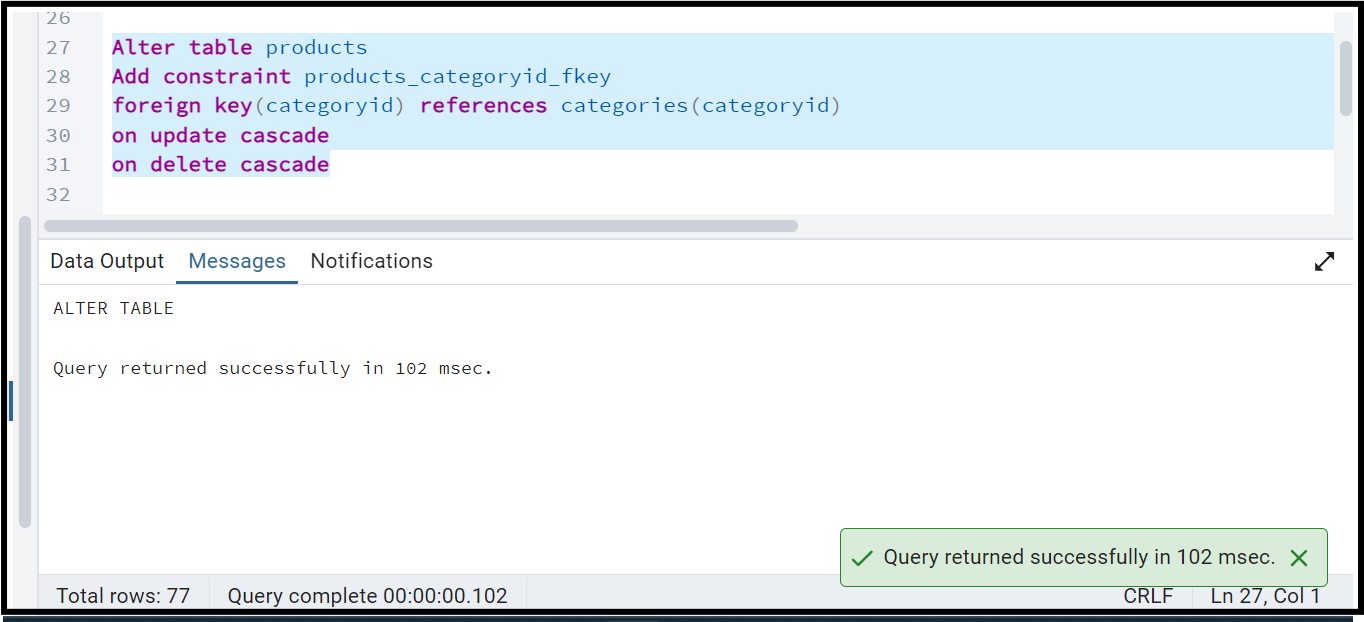
3) Update categoryID=1 to categoryID=1001. Make sure related products update their categoryID too. Display the both category and products table to show the cascade.

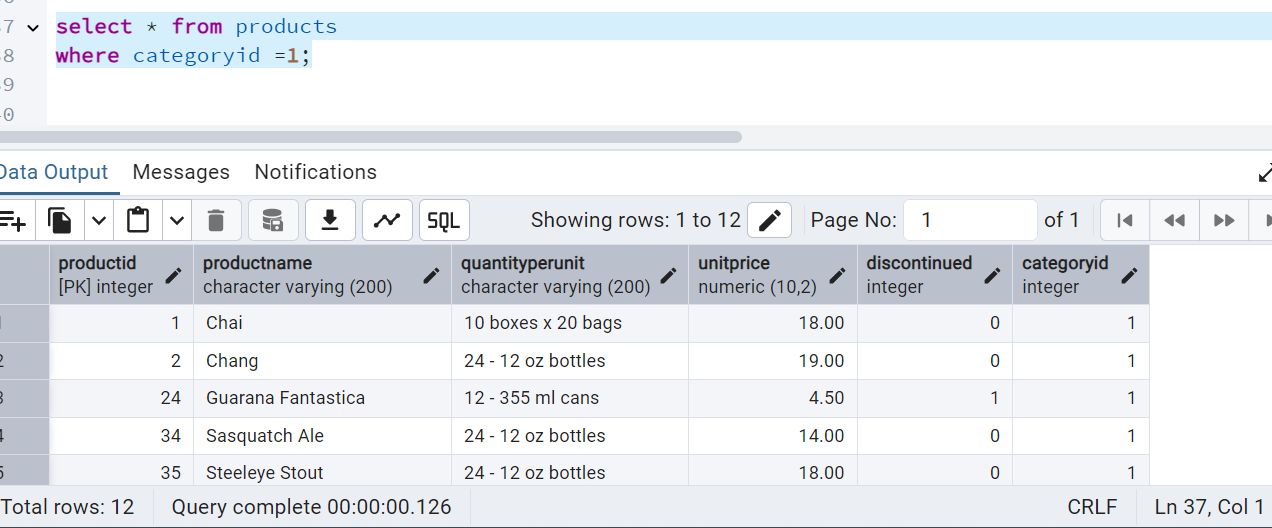
Delete the categoryID= “3” from categories. Verify that the corresponding records are deleted automatically from products.

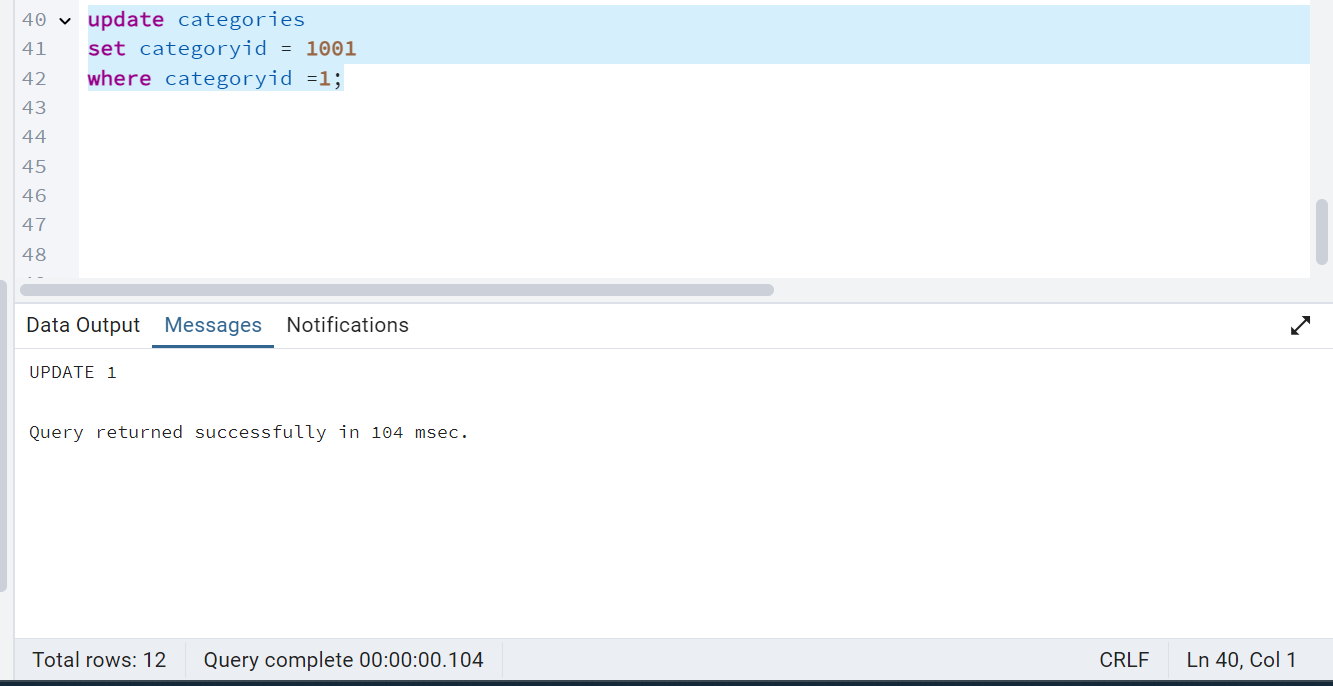
(HINT: Alter the foreign key on products(categoryID) to add ON UPDATE CASCADE, ON DELETE CASCADE)

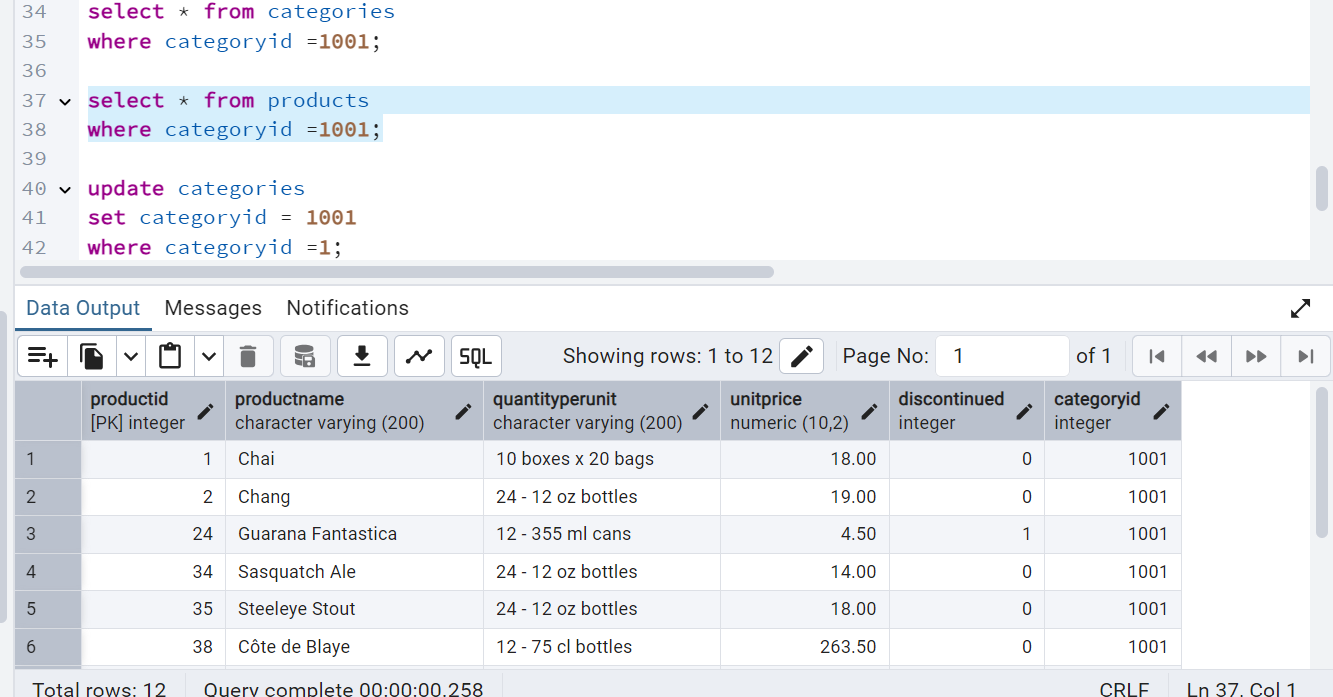
**Ans**:



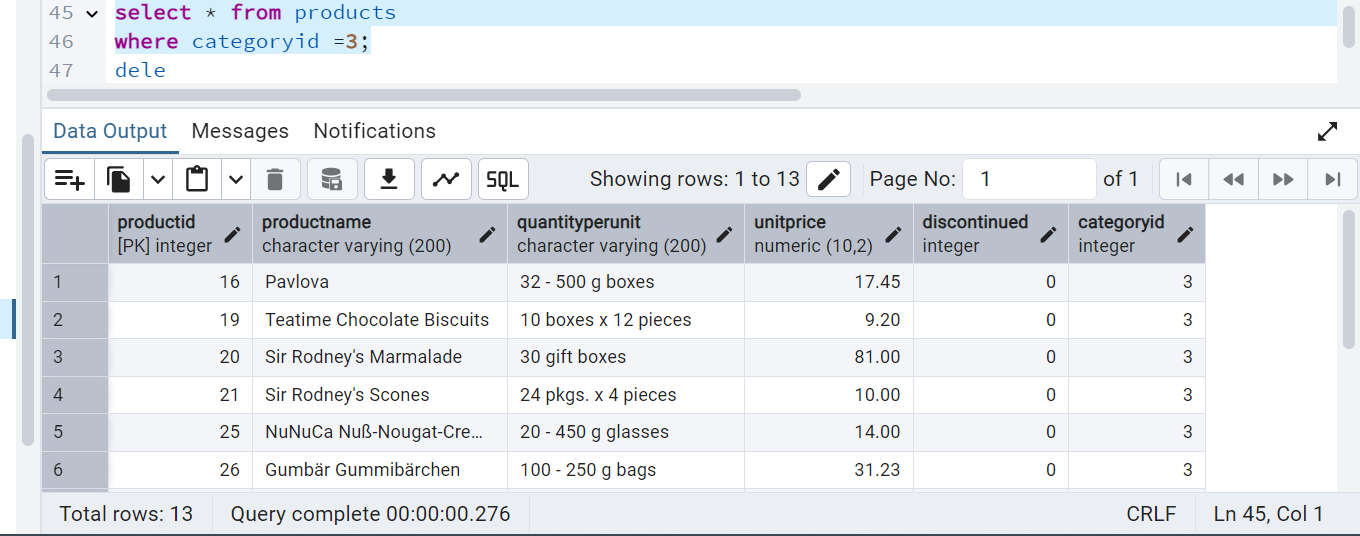


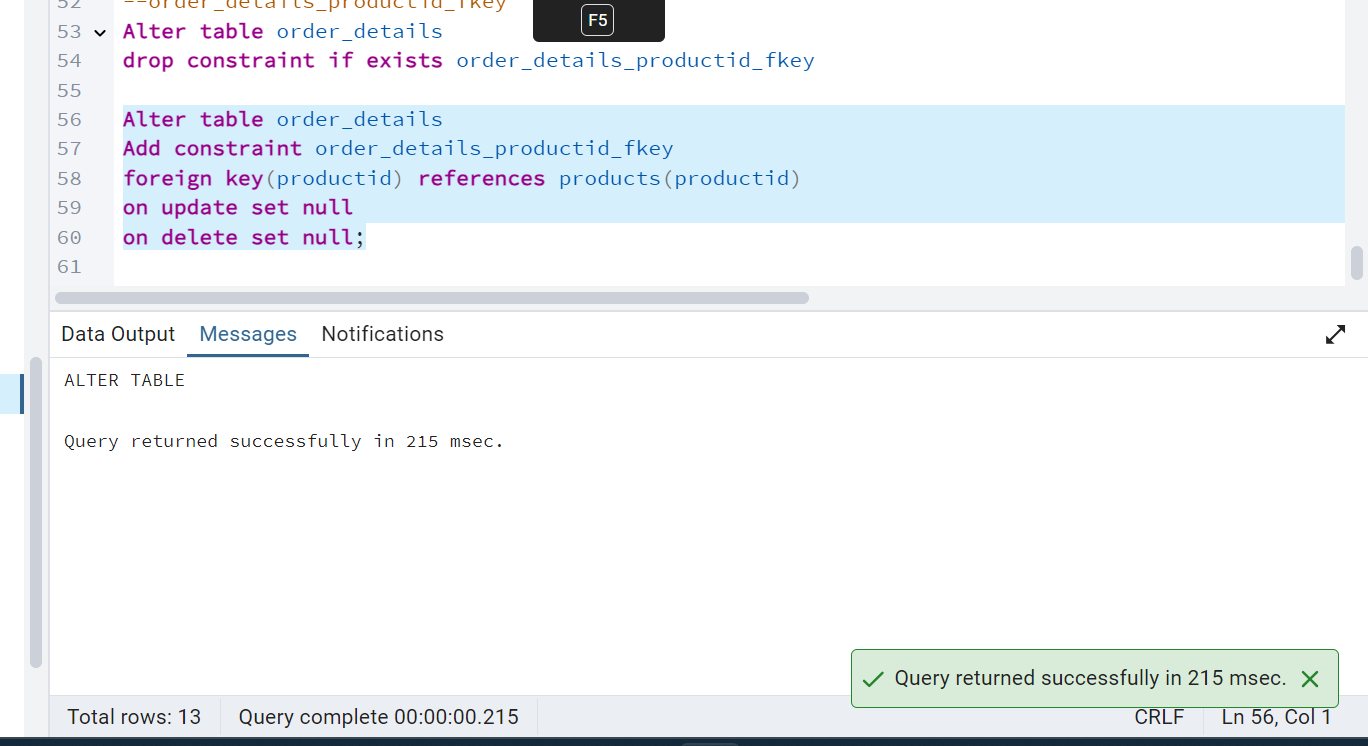






Before delete :

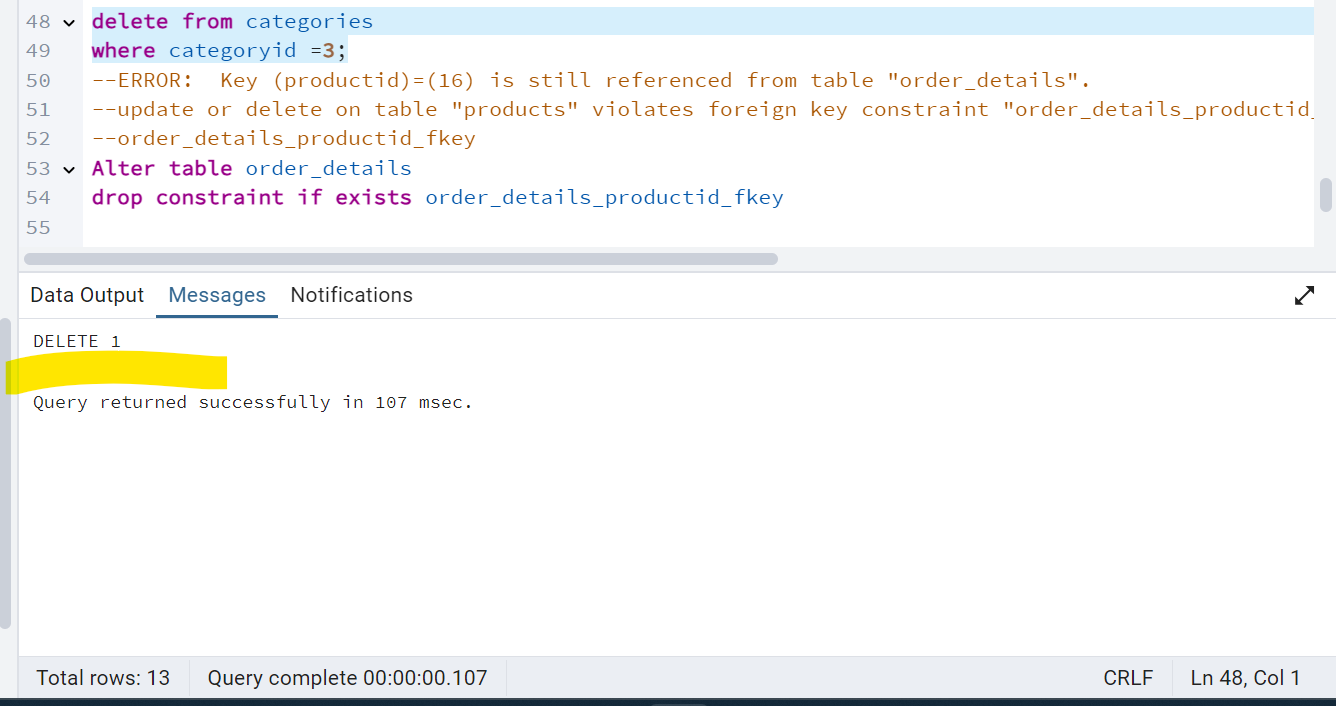




Now able to delete category table with category id =3

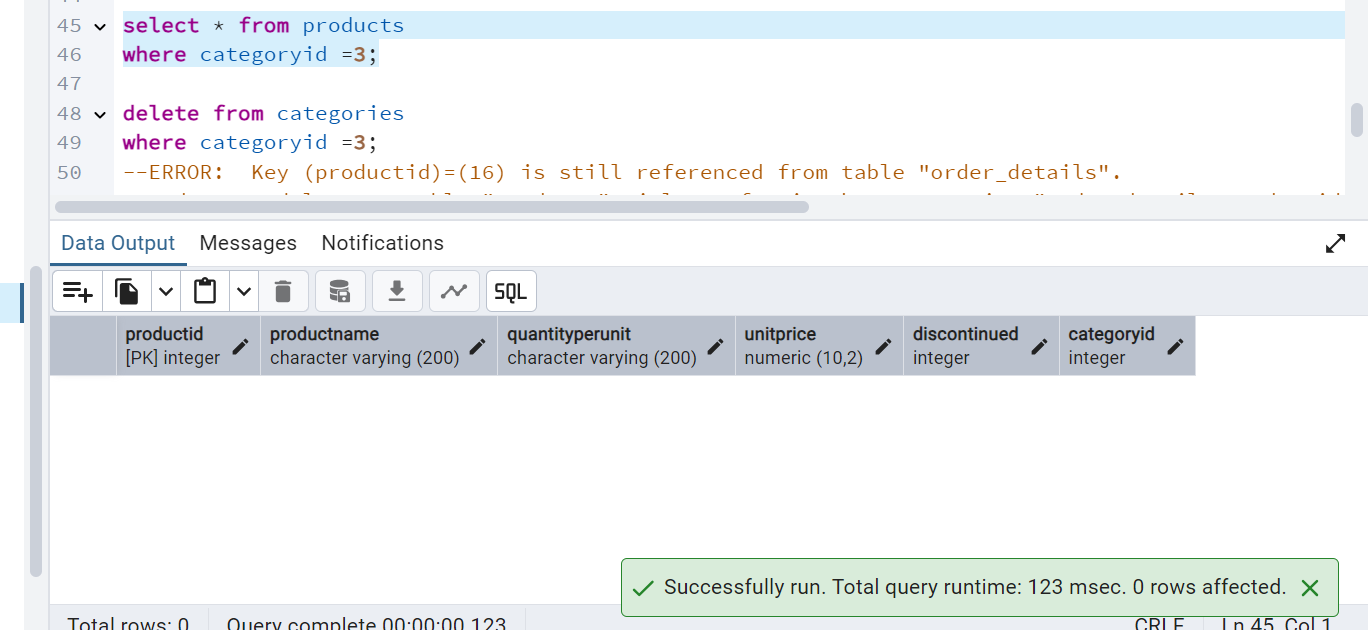
delete from categories

where categoryid =3;



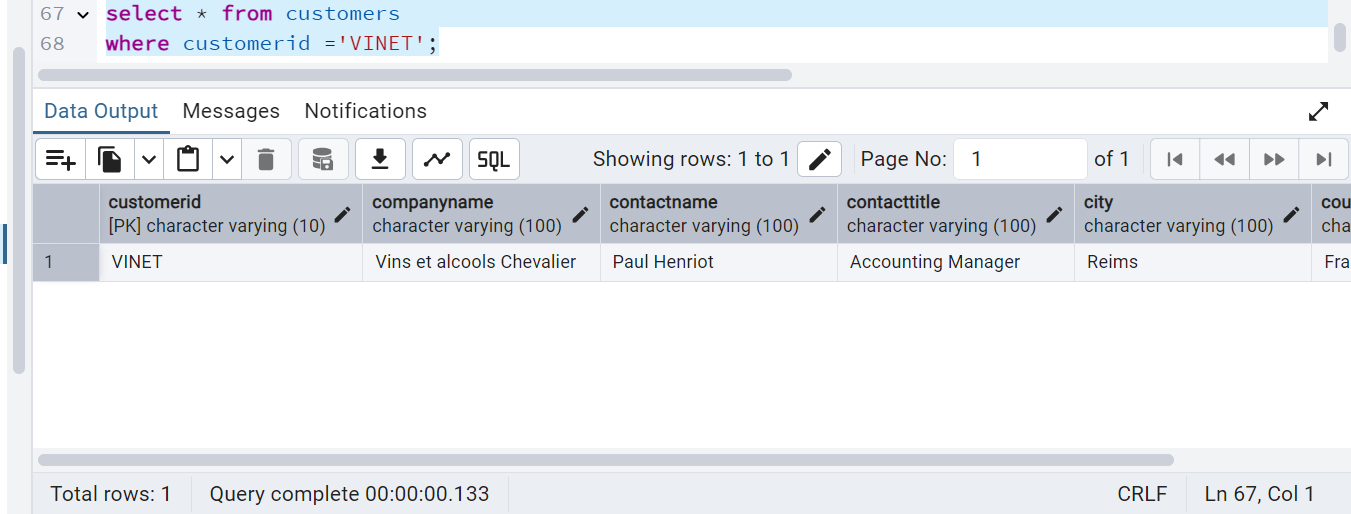
Product table

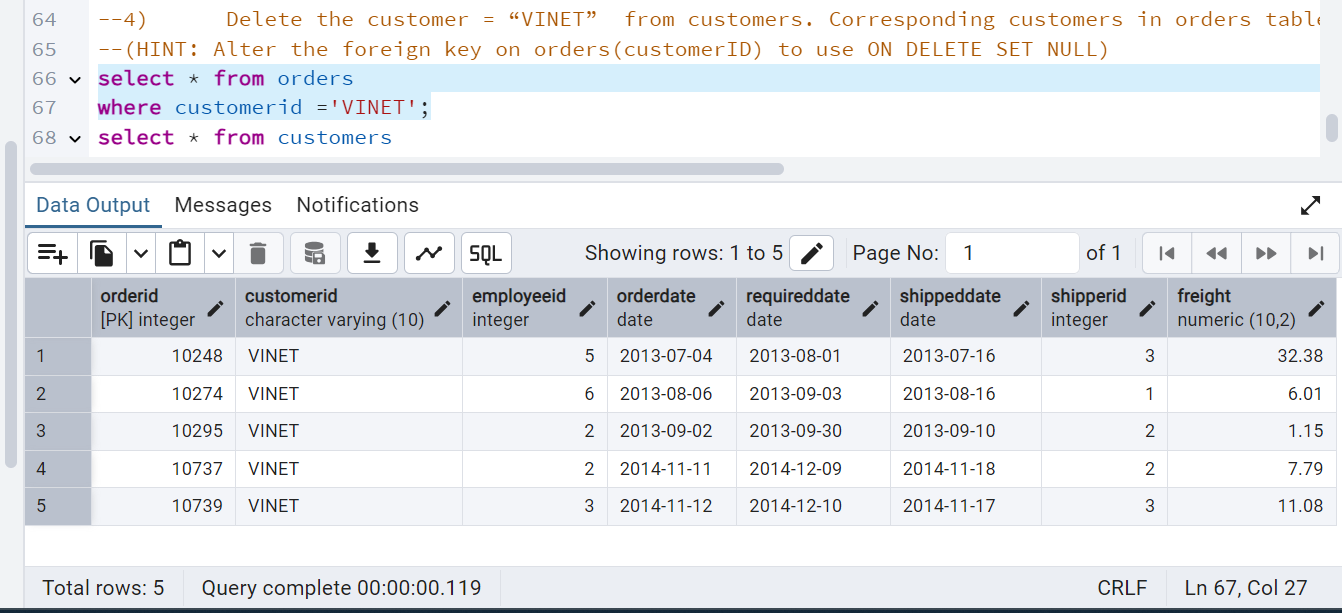
After deletion:

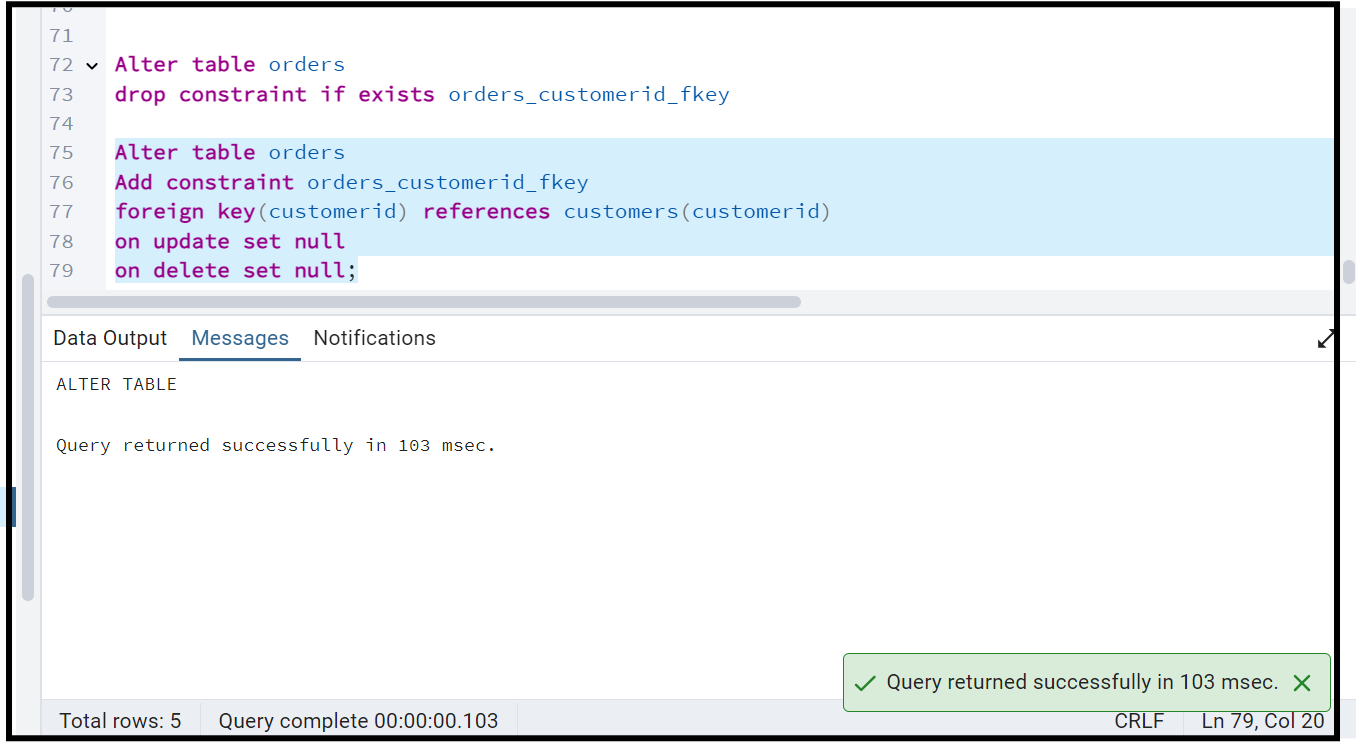


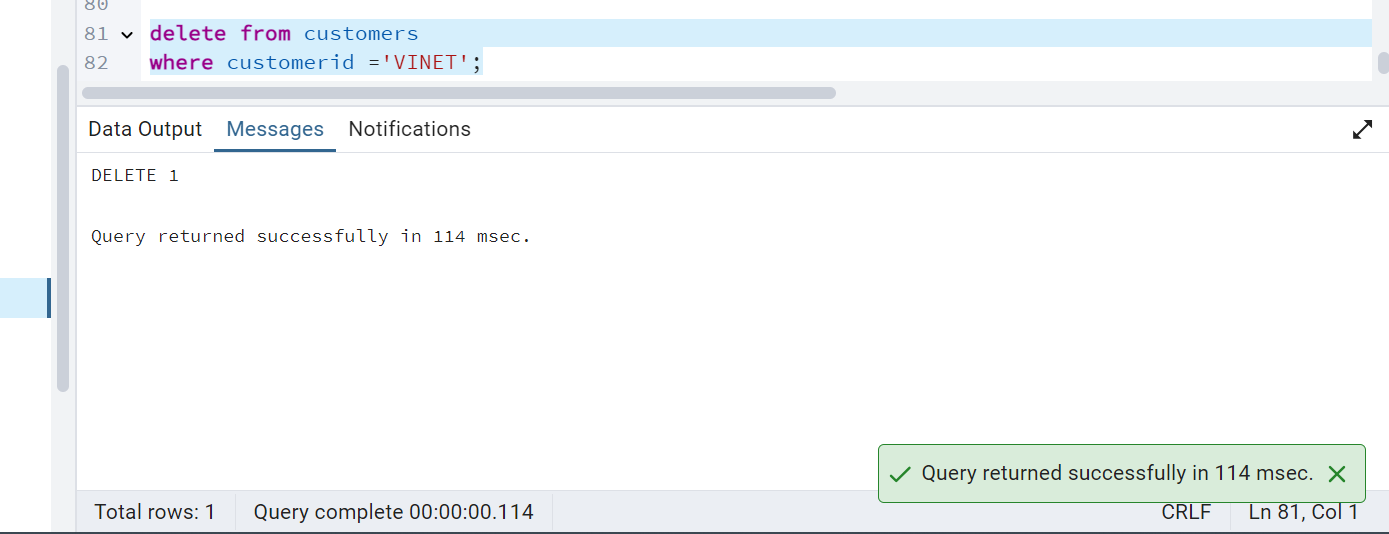
4) Delete the customer = “VINET” from customers. Corresponding customers in orders table should be set to null (HINT: Alter the foreign key on orders(customerID) to use ON DELETE SET NULL)

Customer table:

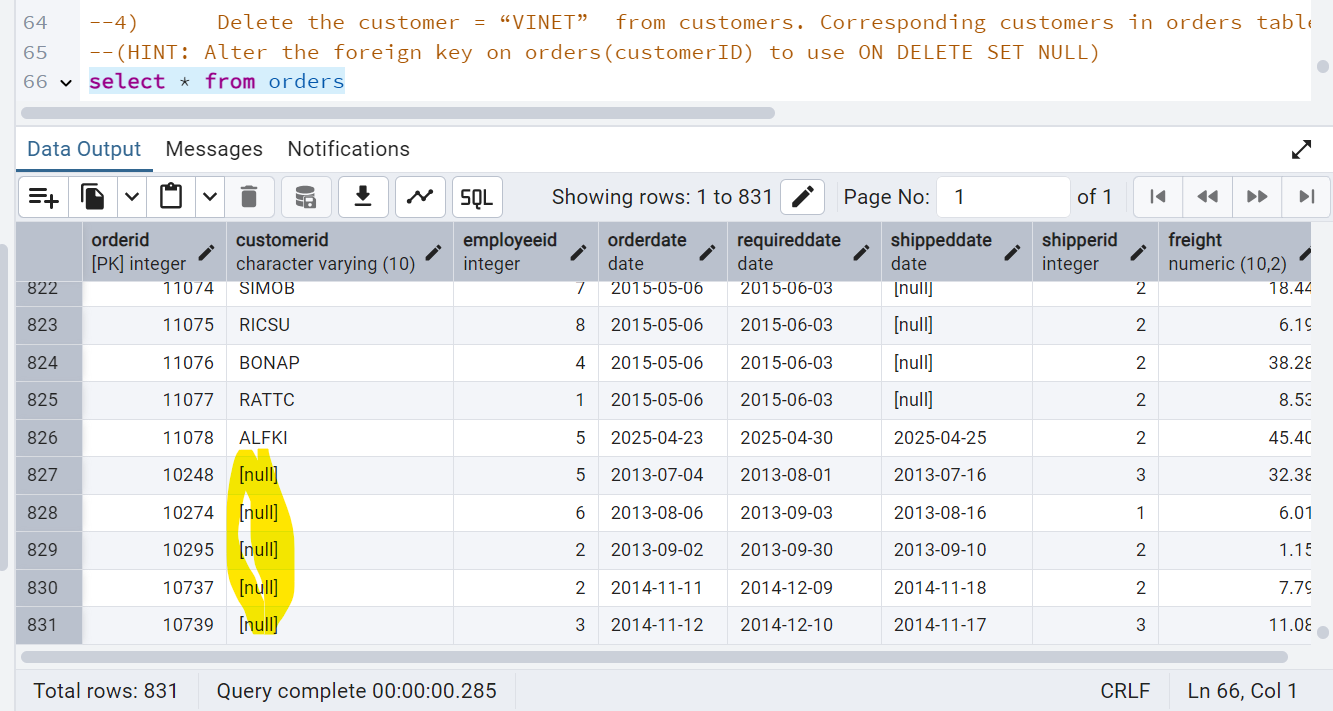








After delete on Customer , Product table customerid is set to null



5) Insert the following data to Products using UPSERT:

product\_id = 100, product\_name = Wheat bread, quantityperunit=1,unitprice = 13, discontinued = 0, categoryID=3

product\_id = 101, product\_name = White bread, quantityperunit=5 boxes,unitprice = 13, discontinued = 0, categoryID=3

product\_id = 100, product\_name = Wheat bread, quantityperunit=10 boxes,unitprice = 13, discontinued = 0, categoryID=3

(this should update the quantityperunit for product\_id = 100)

**Query:**

**insert into products(**

**productid,**

**productname,**

**quantityperunit,**

**unitprice,**

**discontinued,**

**categoryid)**

**values**

**(100, 'Wheat bread', '1', 13, 0, 1001),**

**(101, 'White bread', '5 boxes', 13, 0, 1001)**

**on conflict(productid)**

**do update set**

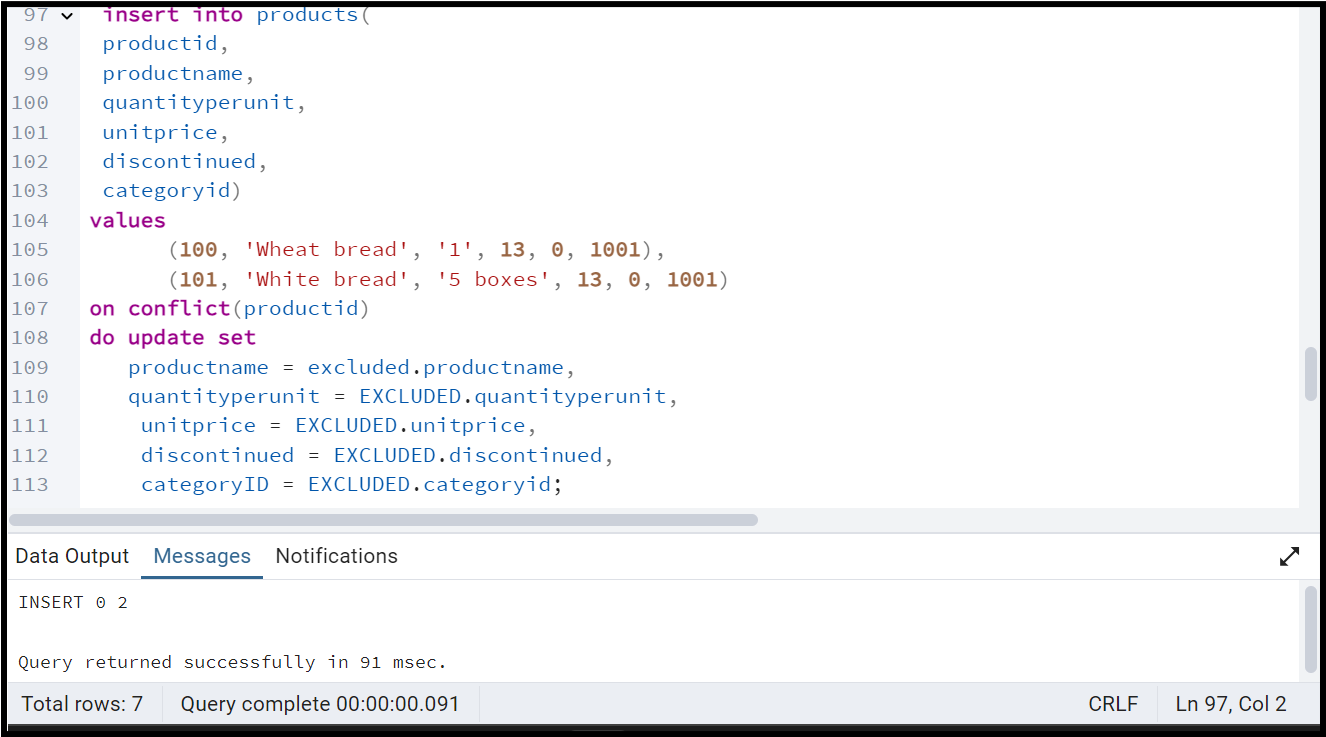
**productname = excluded.productname,**

**quantityperunit = EXCLUDED.quantityperunit,**

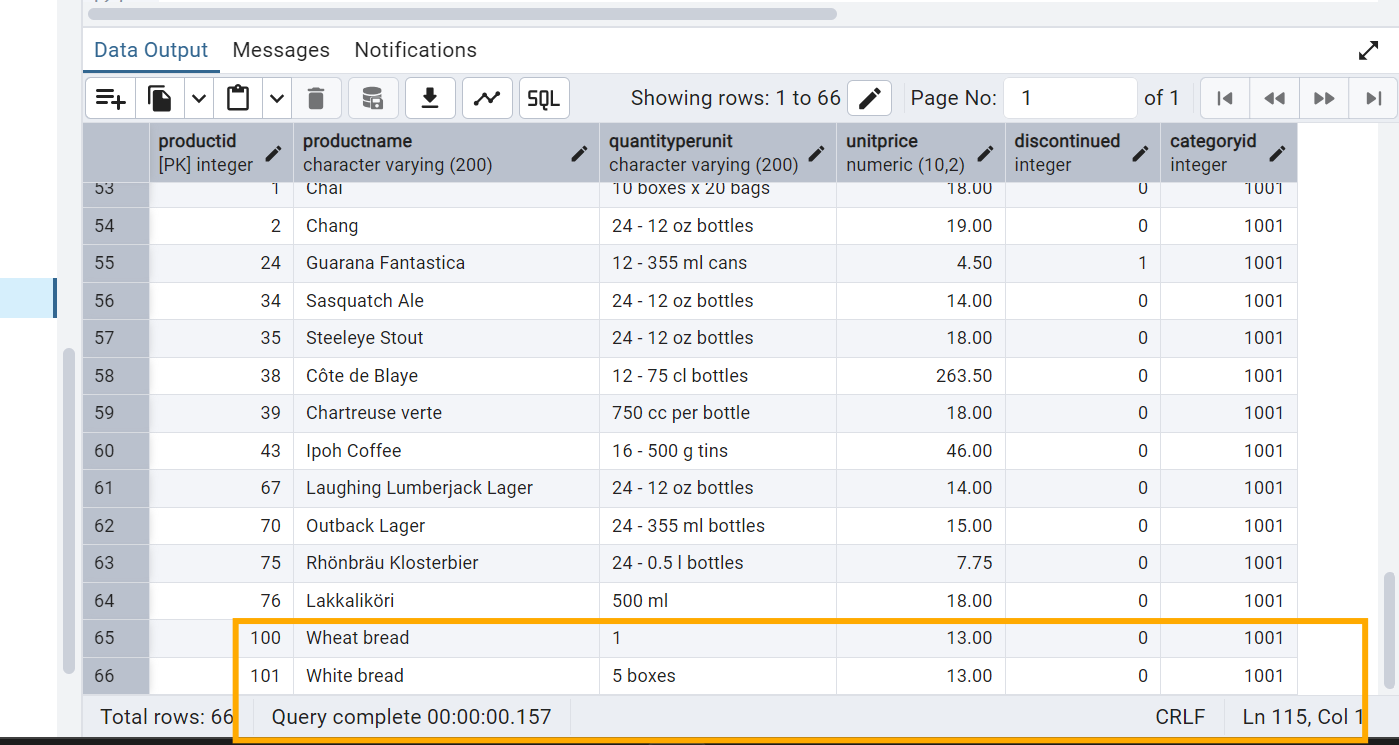
**unitprice = EXCLUDED.unitprice,**

**discontinued = EXCLUDED.discontinued,**

**categoryID = EXCLUDED.categoryid;**

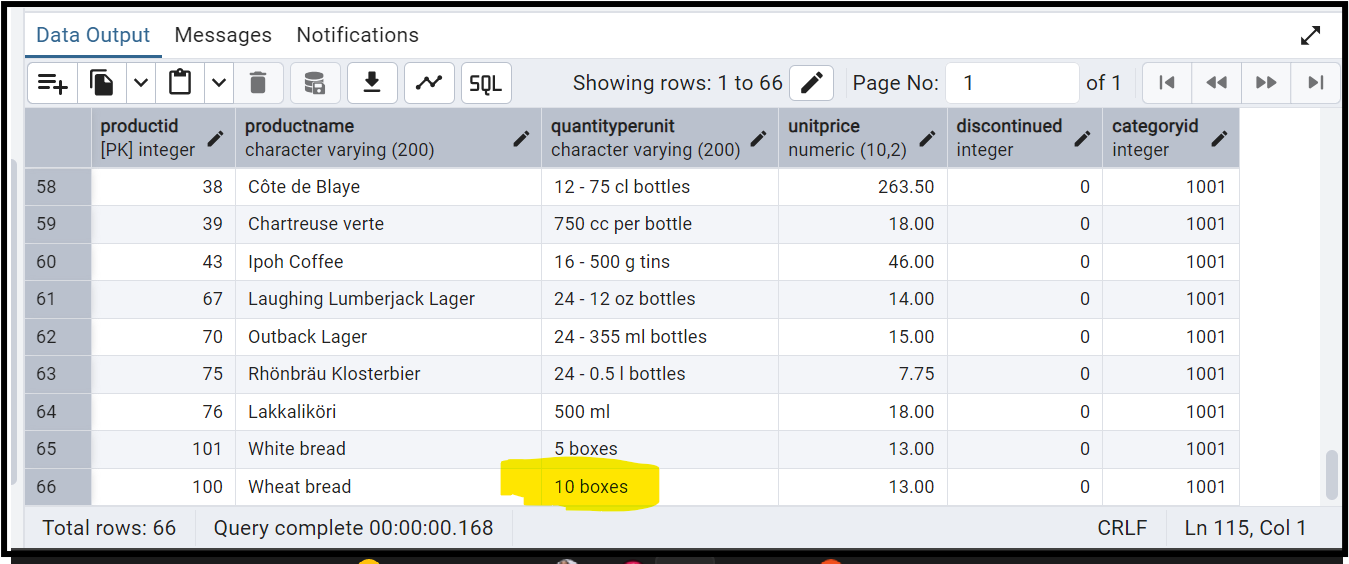


After Insert:





After Update:



6) Write a **MERGE query**:

Create **temp table with name:**  ‘updated\_products’ and insert values as below:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| productID | productName | quantityPerUnit | unitPrice | discontinued | categoryID |
| 100 | Wheat bread | 10 | 20 | 1 | 3 |
| 101 | White bread | 5 boxes | 19.99 | 0 | 3 |
| 102 | Midnight Mango Fizz | 24 - 12 oz bottles | 19 | 0 | 1 |
| 103 | Savory Fire Sauce | 12 - 550 ml bottles | 10 | 0 | 2 |

* Update the price and discontinued status for from below table ‘updated\_products’ only if there are matching products and updated\_products .discontinued =0
* If there are matching products and updated\_products .discontinued =1 then delete

* Insert any new products from updated\_products that don’t exist in products only if updated\_products .discontinued =0.

Ans:

**Query**:

**MERGE into products p**

**using (**

**VALUES**

**(100, 'Wheat bread', '10', 20, 1, 1001),**

**(101, 'White bread', '5 boxes', 19.99, 0, 1001),**

**(102, 'Midnight Mango Fizz', '24-12 oz bottles', 19, 0, 4),**

**(103, 'Savory Fire Sauce', '12-550 ml bottles', 10, 0, 2)**

**) AS updated\_products(productid, productname,quantityperunit,unitprice,discontinued,categoryid)**

**on p.productid =updated\_products.productid**

**when matched and updated\_products .discontinued =0 then**

**update set**

**unitprice = updated\_products.unitprice,**

**discontinued = updated\_products.discontinued**

**when matched and updated\_products .discontinued =1 then**

**delete**

**when not matched and updated\_products .discontinued =0 then**

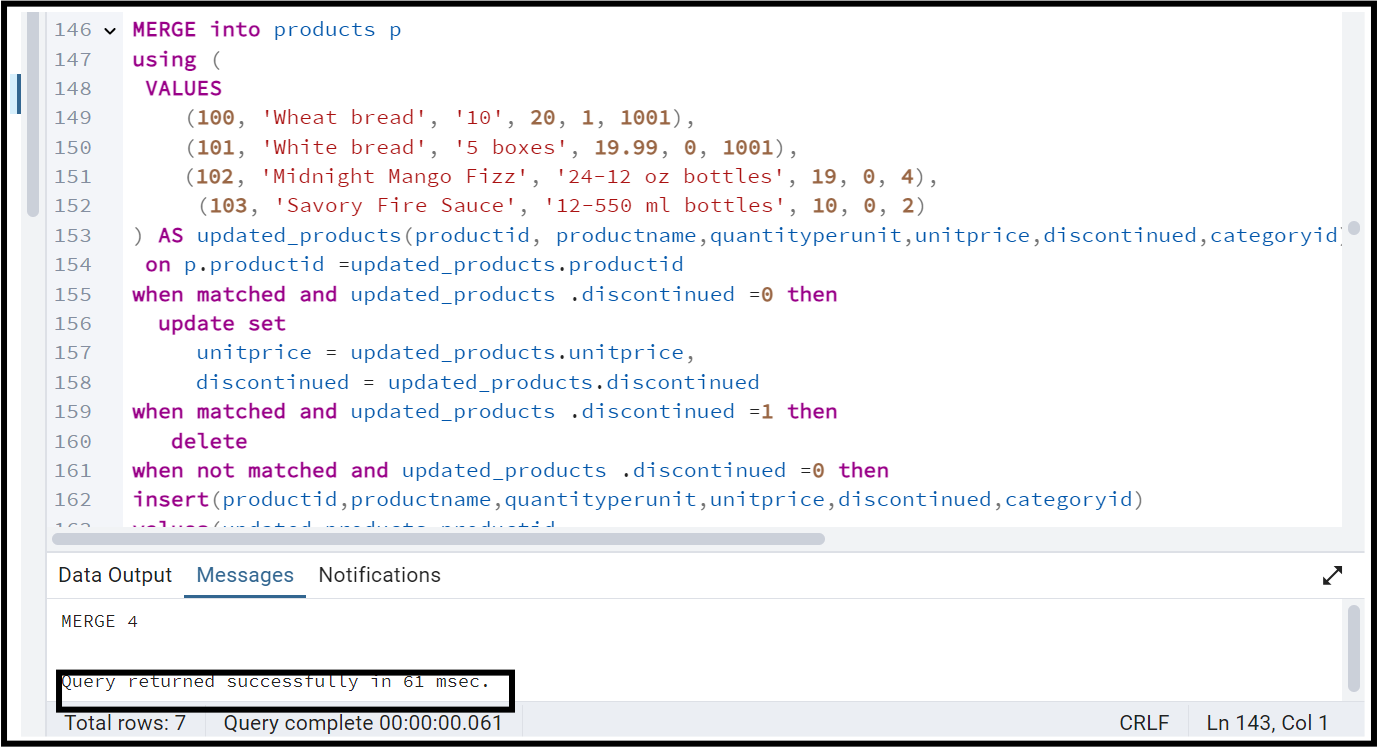
**insert(productid,productname,quantityperunit,unitprice,discontinued,categoryid)**

**values(updated\_products.productid,**

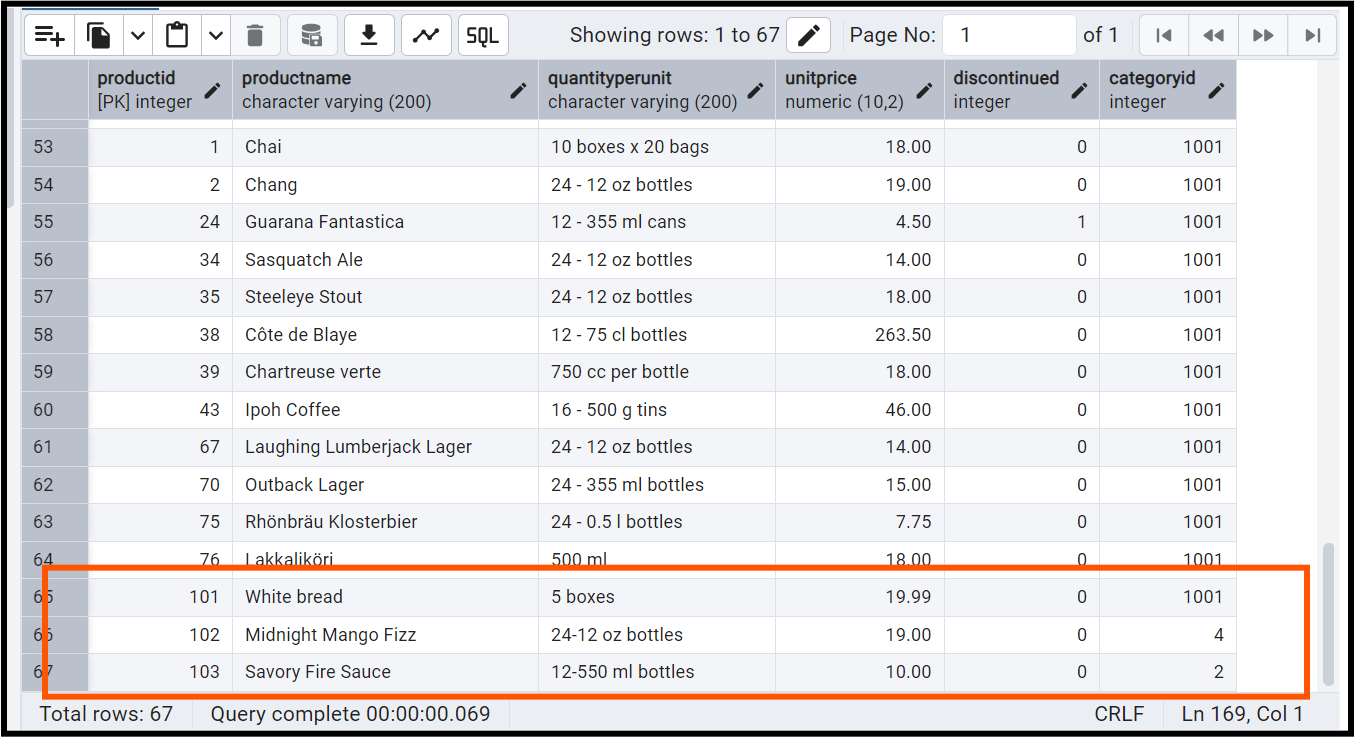
**updated\_products.productname,updated\_products.quantityperunit,**

**updated\_products.unitprice,**

**updated\_products.discontinued,updated\_products.categoryid)**



After above query executed, Product table



**USE NEW Northwind DB:**

7) List all orders with employee full names. (Inner join)

**Query**:

**select e.employee\_id,e.first\_name || ' '||e.last\_name as Full\_Name,**

**e.title, o.order\_id,o.customer\_id,o.order\_date,o.required\_date,o.shipped\_date**

**from employees e**

**inner join orders o**

**on e.employee\_id = o.employee\_id;**

