Day3\_Assignment

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

UPDATE CATEGORIES

SET

CATEGORY\_NAME = 'Drinks'

WHERE

CATEGORY\_NAME = 'Beverages';

SELECT

CATEGORY\_NAME

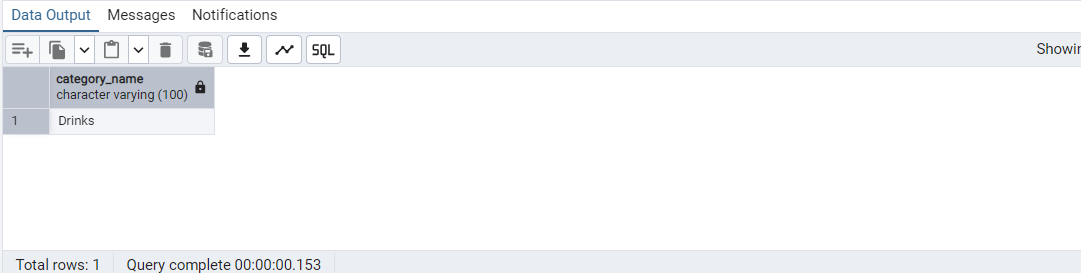
FROM

CATEGORIES

WHERE

CATEGORY\_NAME = 'Drinks'

### OUTPUT



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

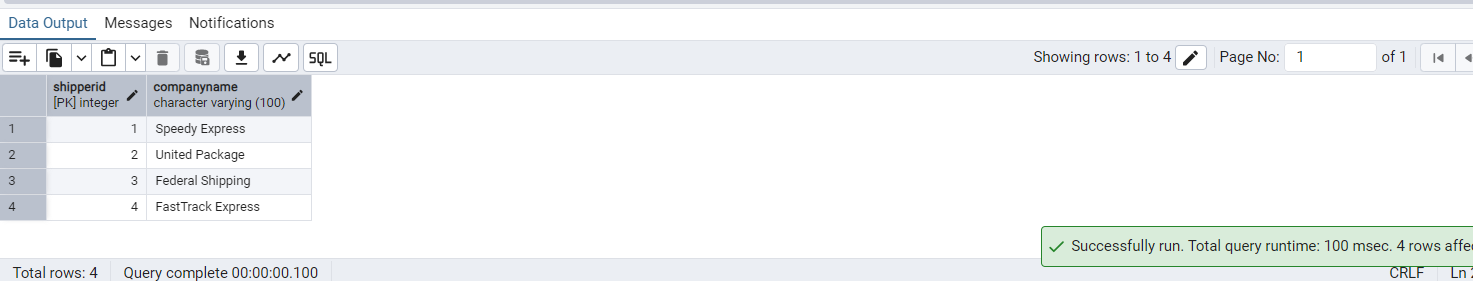
INSERT INTO

SHIPPERS (SHIPPERID, COMPANYNAME)

VALUES

(4, 'FastTrack Express');

### OUTPUT



SELECT

\*

FROM

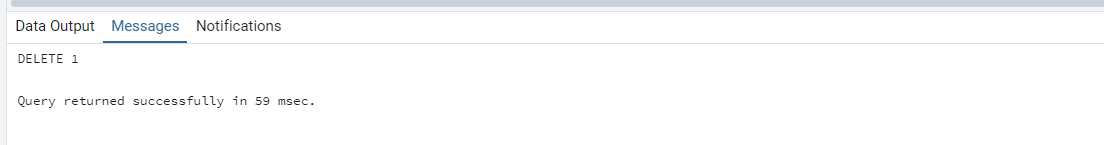
SHIPPERS

DELETE FROM SHIPPERS

WHERE

SHIPPERID = 4;

### OUTPUT



-- 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.

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

-- Find the foreign key constraint name

SELECT

CONNAME

FROM

PG\_CONSTRAINT

WHERE

CONRELID = 'products'::REGCLASS

AND CONTYPE = 'f';

--categoryid\_fk

-- Drop the existing constraint

ALTER TABLE PRODUCTS

DROP CONSTRAINT CATEGORYID\_FK;

-- Recreate with ON UPDATE CASCADE and ON DELETE CASCADE

ALTER TABLE PRODUCTS

ADD CONSTRAINT CATEGORYID\_FK FOREIGN KEY (CATEGORYID) REFERENCES CATEGORIES (CATEGORY\_ID) ON UPDATE CASCADE ON DELETE CASCADE;

UPDATE CATEGORIES

SET

CATEGORY\_ID = 1001

WHERE

CATEGORY\_ID = 1;

SELECT

\*

FROM

CATEGORIES

WHERE

CATEGORY\_ID = 1001;

### OUTPUT



SELECT

\*

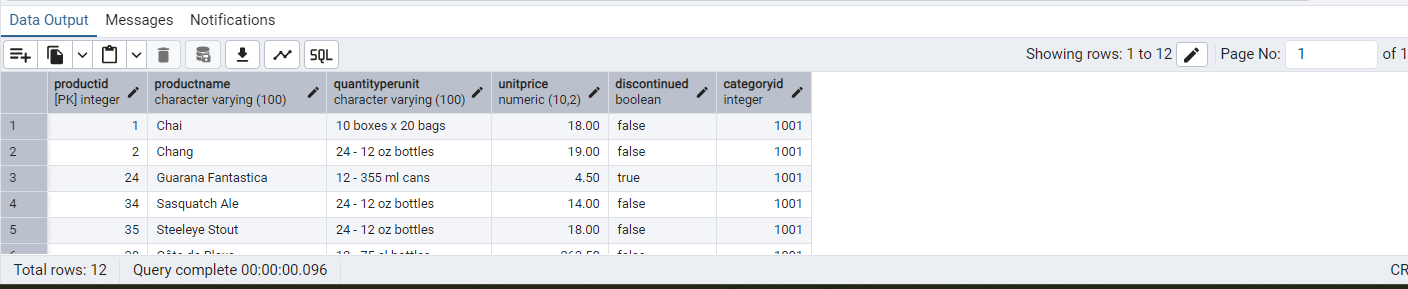
FROM

PRODUCTS

WHERE

CATEGORYID = 1001;

### OUTPUT



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

DELETE FROM CATEGORIES

WHERE

CATEGORY\_ID = 3;

SELECT

CONNAME

FROM

PG\_CONSTRAINT

WHERE

CONRELID = 'order\_details'::REGCLASS

AND CONTYPE = 'f';

"orderId\_fk" "productId\_fk"

ALTER TABLE ORDER\_DETAILS

DROP CONSTRAINT PRODUCTID\_FK;

ALTER TABLE ORDER\_DETAILS

ADD CONSTRAINT PRODUCTID\_FK FOREIGN KEY (PRODUCTID) REFERENCES PRODUCTS (PRODUCTID) ON UPDATE CASCADE ON DELETE CASCADE;

DELETE FROM CATEGORIES

WHERE

CATEGORY\_ID = 3;

SELECT

\*

FROM

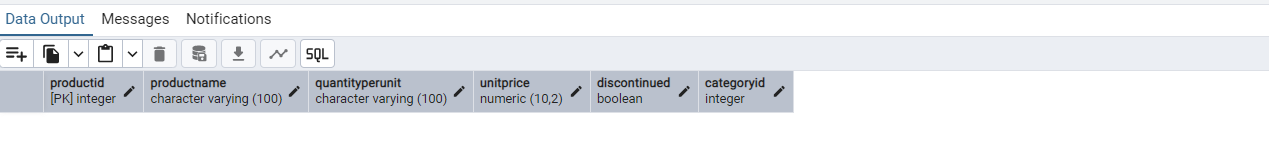
PRODUCTS

WHERE

CATEGORYID = 3;

-- Should return 0 rows

### OUTPUT



-- 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)

SELECT

CONNAME

FROM

PG\_CONSTRAINT

WHERE

CONRELID = 'orders'::REGCLASS

AND CONTYPE = 'f';

"employeeid\_fk" "shippers" "customers"

ALTER TABLE ORDERS

DROP CONSTRAINT CUSTOMERS;

ALTER TABLE ORDERS

ADD CONSTRAINT CUSTOMERS FOREIGN KEY (CUSTOMERID) REFERENCES CUSTOMERS (CUSTOMER\_ID) ON DELETE SET NULL;

DELETE FROM CUSTOMERS

WHERE

CUSTOMER\_ID = 'VINET';

SELECT

\*

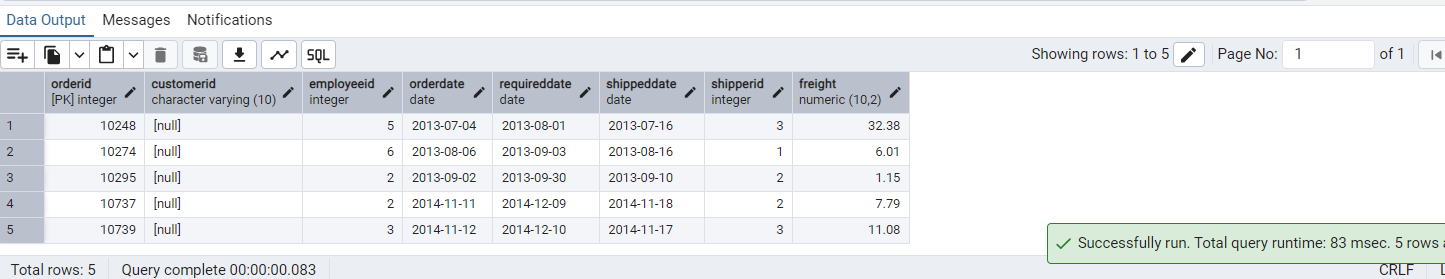
FROM

ORDERS

WHERE

CUSTOMERID IS NULL;

### OUTPUT



-- 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)

INSERT INTO

PRODUCTS (

PRODUCTID,

PRODUCTNAME,

QUANTITYPERUNIT,

UNITPRICE,

DISCONTINUED,

CATEGORYID

)

VALUES

(100, 'Wheat bread', '1', 13, FALSE, 3),

(101, 'White bread', '5 boxes', 13, FALSE, 3)

ON CONFLICT (PRODUCTID) DO

UPDATE

SET

QUANTITYPERUNIT = EXCLUDED.QUANTITYPERUNIT;

–To check

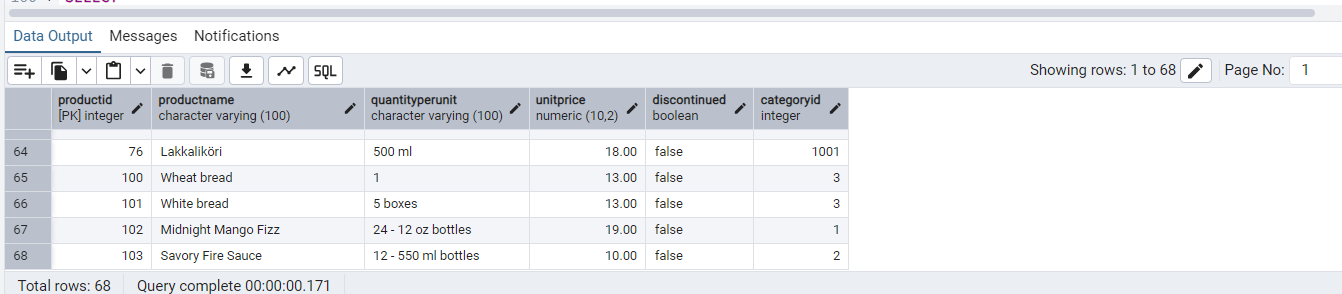
SELECT

\*

FROM

PRODUCTS

### OUTPUT



INSERT INTO

CATEGORIES (CATEGORY\_ID, CATEGORY\_NAME, DESCRIPTION)

VALUES

(

3,

'Bakery',

'Baked goods like bread and pastries'

);

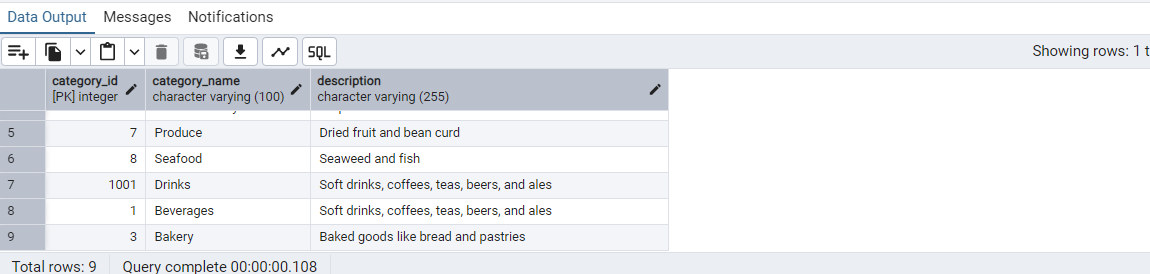
SELECT

\*

FROM

CATEGORIES;

### OUTPUT



-- Insert or update product 100 and 101

INSERT INTO

PRODUCTS (

PRODUCTID,

PRODUCTNAME,

QUANTITYPERUNIT,

UNITPRICE,

DISCONTINUED,

CATEGORYID

)

VALUES

(100, 'Wheat bread', '10 boxes', 13, FALSE, 3)

ON CONFLICT (PRODUCTID) DO

UPDATE

SET

QUANTITYPERUNIT = EXCLUDED.QUANTITYPERUNIT;

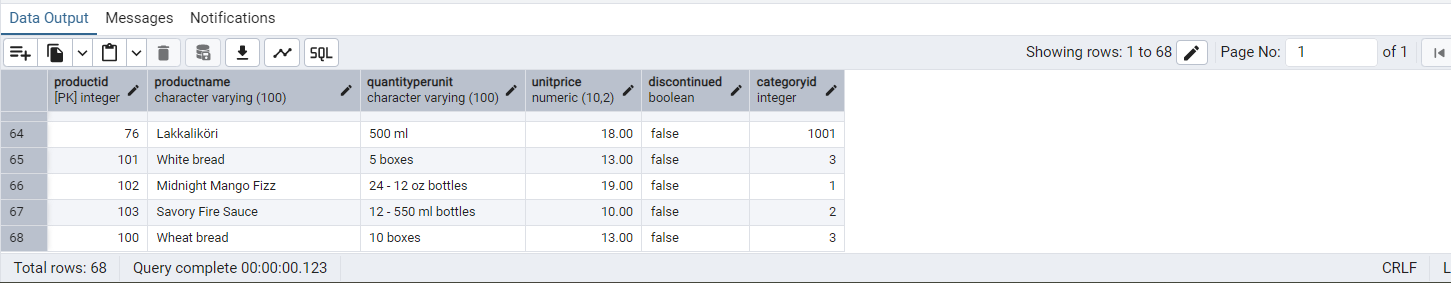
SELECT

\*

FROM

PRODUCTS;

### OUTPUT



-- 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.

CREATE TEMP TABLE UPDATED\_PRODUCTS (

PRODUCT\_ID INT,

PRODUCT\_NAME TEXT,

QUANTITYPERUNIT TEXT,

UNITPRICE NUMERIC,

DISCONTINUED BOOLEAN,

CATEGORY\_ID INT

);

INSERT INTO

UPDATED\_PRODUCTS

VALUES

(100, 'Wheat bread', '10', 20, TRUE, 3),

(101, 'White bread', '5 boxes', 19.99, FALSE, 3),

(

102,

'Midnight Mango Fizz',

'24 - 12 oz bottles',

19,

FALSE,

1

),

(

103,

'Savory Fire Sauce',

'12 - 550 ml bottles',

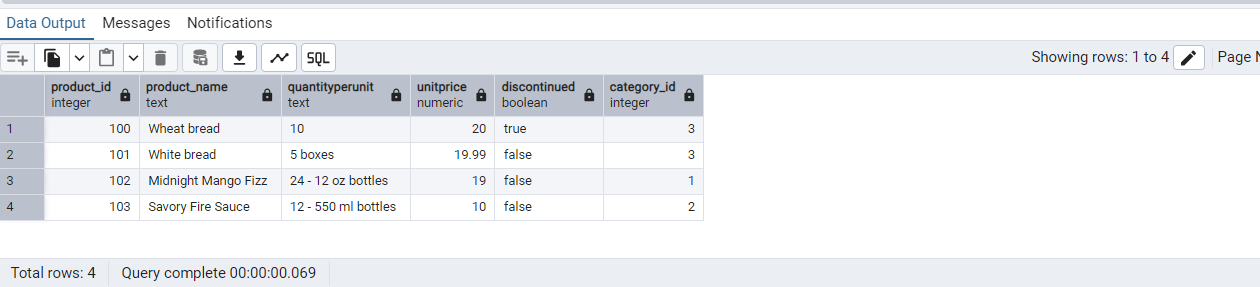
10,

FALSE,

2

);

### OUTPUT



-- 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.

MERGE INTO PRODUCTS P USING UPDATED\_PRODUCTS U ON P.PRODUCTID = U.PRODUCT\_ID WHEN MATCHED

AND U.DISCONTINUED = FALSE THEN

UPDATE

SET

UNITPRICE = U.UNITPRICE,

DISCONTINUED = U.DISCONTINUED

-- DELETE if discontinued = true

WHEN MATCHED

AND U.DISCONTINUED = TRUE THEN DELETE

-- INSERT if new and discontinued = false

WHEN NOT MATCHED

AND U.DISCONTINUED = FALSE THEN INSERT (

PRODUCTID,

PRODUCTNAME,

QUANTITYPERUNIT,

UNITPRICE,

DISCONTINUED,

CATEGORYID

)

VALUES

(

U.PRODUCT\_ID,

U.PRODUCT\_NAME,

U.QUANTITYPERUNIT,

U.UNITPRICE,

U.DISCONTINUED,

U.CATEGORY\_ID

);

–To check

SELECT

\*

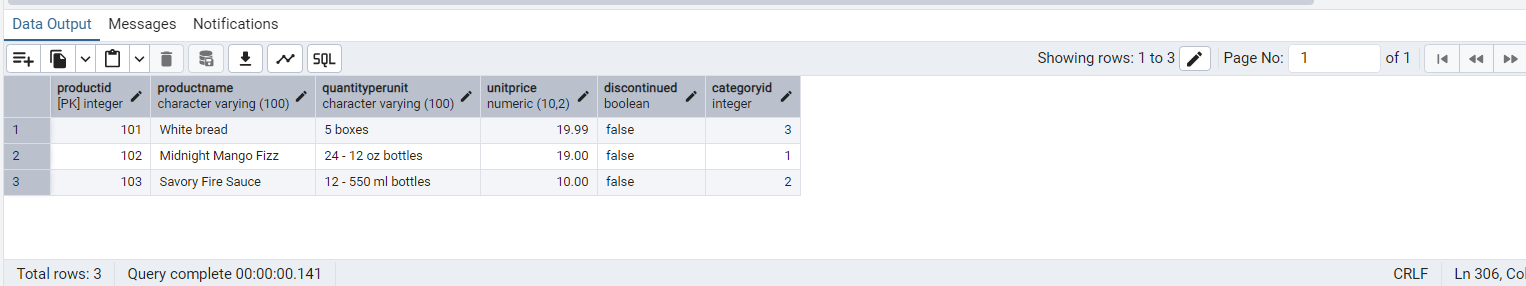
FROM

PRODUCTS

WHERE

PRODUCTID IN (100, 101, 102, 103);

OUTPUT



–From northwind Database

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

SELECT

O.ORDER\_ID,

O.ORDER\_DATE,

O.CUSTOMER\_ID,

E.EMPLOYEE\_ID,

E.FIRST\_NAME || ' ' || E.LAST\_NAME AS EMPLOYEE\_FULL\_NAME

FROM

ORDERS O

INNER JOIN EMPLOYEES E ON O.EMPLOYEE\_ID = E.EMPLOYEE\_ID;

### OUTPUT

