**Elamathi\_Assignment 2/Day 2**

**-- 1. Alter Table**

**-- a. Add a new column 'linkedin\_profile' to 'employees' table**

ALTER TABLE employees

ADD COLUMN linkedin\_profile VARCHAR(100);

**OUTPUT**:



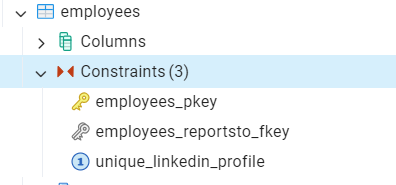
------------------------------------------------------------------------------------------------------------------------------------------**-- b. Change the column type from VARCHAR to TEXT and Add UNIQUE constraints**

ALTER TABLE employees

ALTER COLUMN linkedin\_profile TYPE TEXT,

ADD CONSTRAINT unique\_linkedin\_profile UNIQUE (linkedin\_profile);

**OUTPUT**:



------------------------------------------------------------------------------------------------------------------------------------------

**-- c. Add NOT NULL constraints**

**--First update with values**

UPDATE employees

SET linkedin\_profile = CONCAT ('N/A\_', employeeid)

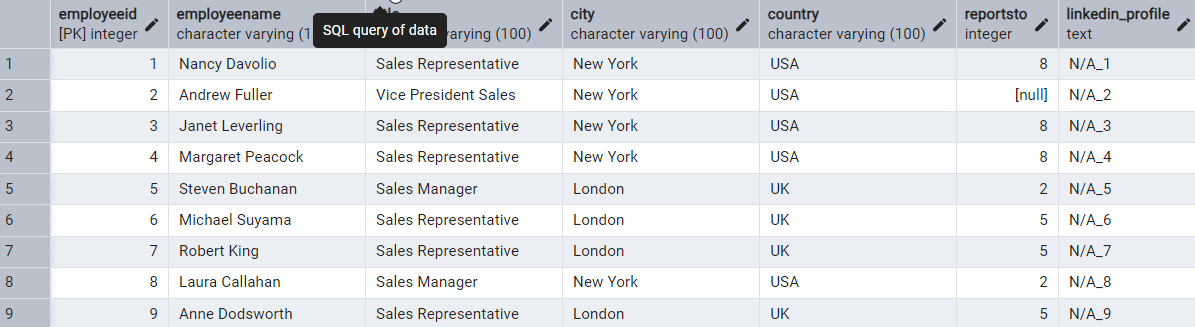
WHERE linkedin\_profile IS NULL;

**-- Then Alter the column with NOT NULL constraint**

ALTER TABLE employees

ALTER COLUMN linkedin\_profile SET NOT NULL;

**OUTPUT**:



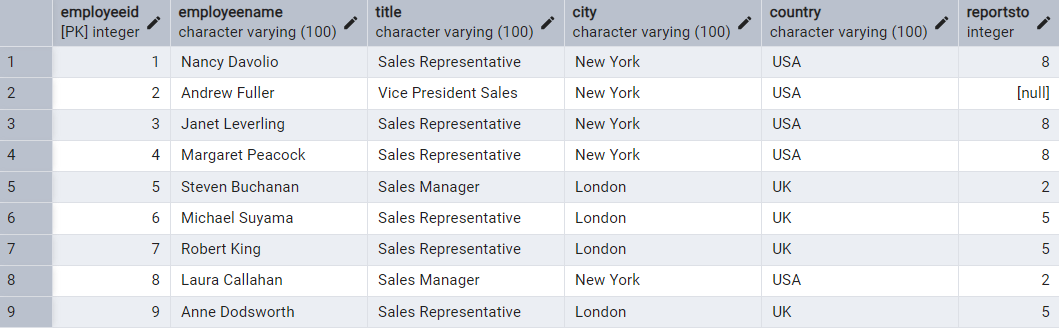
------------------------------------------------------------------------------------------------------------------------------------------

**-- d. Drop the 'linkedin\_profile' column**

ALTER TABLE employees

DROP COLUMN linkedin\_profile;

**OUTPUT: COLUMN linkedin\_profile is deleted**



------------------------------------------------------------------------------------------------------------------------------------------

**-- 2. Querying (Select)**

**--a. Retrieve first name, last name, title of all employees**

SELECT

employeename,

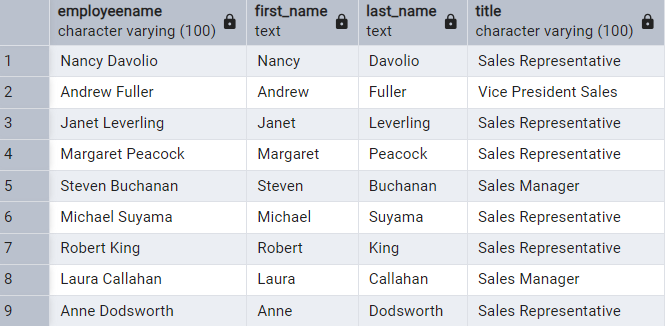
SPLIT\_PART(employeename, ' ', 1) AS first\_name,

SPLIT\_PART(employeename, ' ', 2) AS last\_name,

title

FROM employees;

**OUTPUT**:

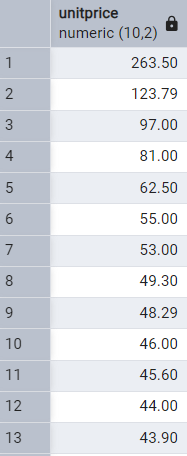


------------------------------------------------------------------------------------------------------------------------------------------

**--b. Find all unique unit prices of products**

SELECT DISTINCT unitprice FROM products ORDER BY unitprice DESC;

**OUTPUT**:



------------------------------------------------------------------------------------------------------------------------------------------

**-- c. List all customers sorted by company name in ascending order**

SELECT \* FROM customers ORDER BY companyname ASC;

**OUTPUT**:



------------------------------------------------------------------------------------------------------------------------------------------

**--d. Display product name and unit price, but rename the unit\_price column as price\_in\_usd**

SELECT productname, unitprice AS price\_in\_usd FROM products;

**OUTPUT**:



------------------------------------------------------------------------------------------------------------------------------------------

**-- 3. Filtering**

**--a. Get all customers from Germany.**

SELECT \* FROM customers WHERE country = 'Germany';

**OUTPUT**:



------------------------------------------------------------------------------------------------------------------------------------------

**--b. Find all customers from France or Spain**

SELECT \* FROM customers

WHERE country = 'France' OR country = 'Spain';

**OUTPUT**:



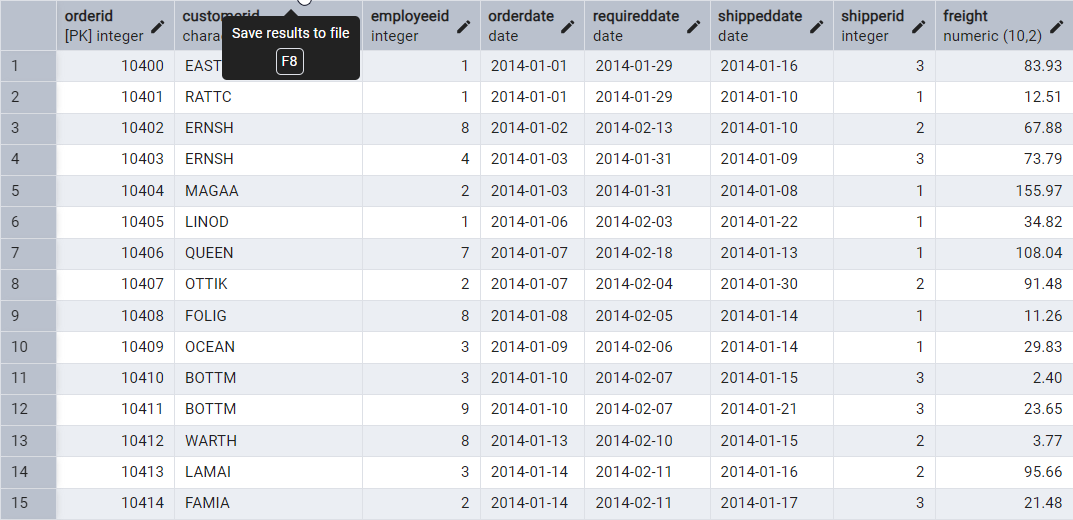
------------------------------------------------------------------------------------------------------------------------------------------

**--c. Retrieve all orders placed in 1997 and either have freight greater than 50 or the shipped date available (i.e., non-NULL)**

SELECT \* FROM orders

WHERE EXTRACT (YEAR FROM orderdate) = 2014 AND (freight > 50 OR shippeddate IS NOT NULL);

**OUTPUT**:



------------------------------------------------------------------------------------------------------------------------------------------

**-- 4. Filtering**

**-- a. Retrieve the product\_id, product\_name, and unit\_price of products where the unit\_price is**

**greater than 15.**

SELECT productid, productname, unitprice FROM products WHERE unitprice > 15;

**OUTPUT**:



------------------------------------------------------------------------------------------------------------------------------------------

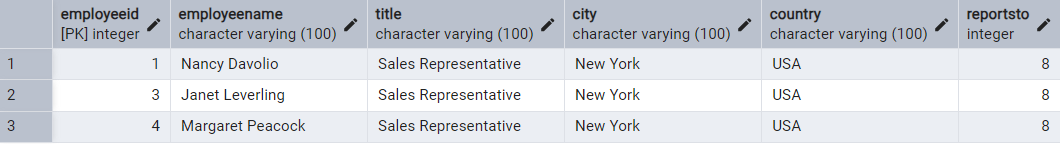
**--b. List all employees who are located in the USA and have the title "Sales Representative".**

SELECT \* FROM employees WHERE country = 'USA' AND title = 'Sales Representative';

SELECT employeename FROM employees WHERE country = 'USA' AND title = 'Sales

Representative';

**OUTPUT**



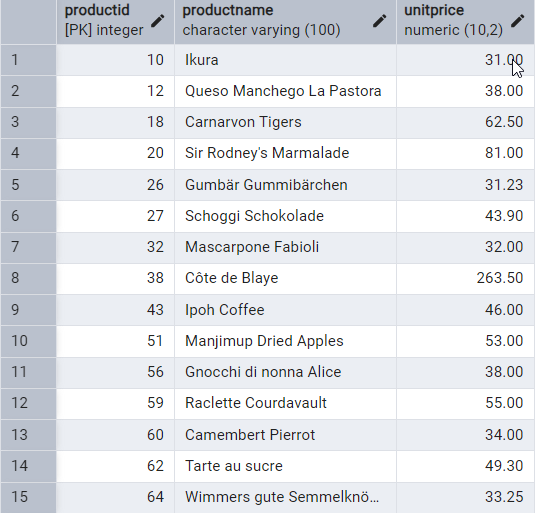
------------------------------------------------------------------------------------------------------------------------------------------

**--c. Retrieve all products that are not discontinued and priced greater than 30.**

SELECT productid, productname, unitprice FROM products

WHERE unitprice > 30 AND discontinued = FALSE;

**OUTPUT**:



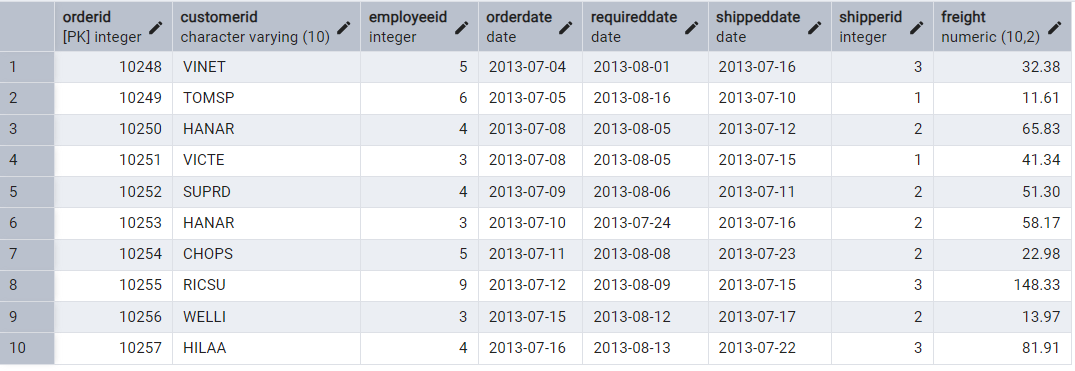
------------------------------------------------------------------------------------------------------------------------------------------

**--5. LIMIT/FETCH**

**--a. Retrieve the first 10 orders from the orders table.**

SELECT \* FROM orders LIMIT 10;

**OUTPUT**:

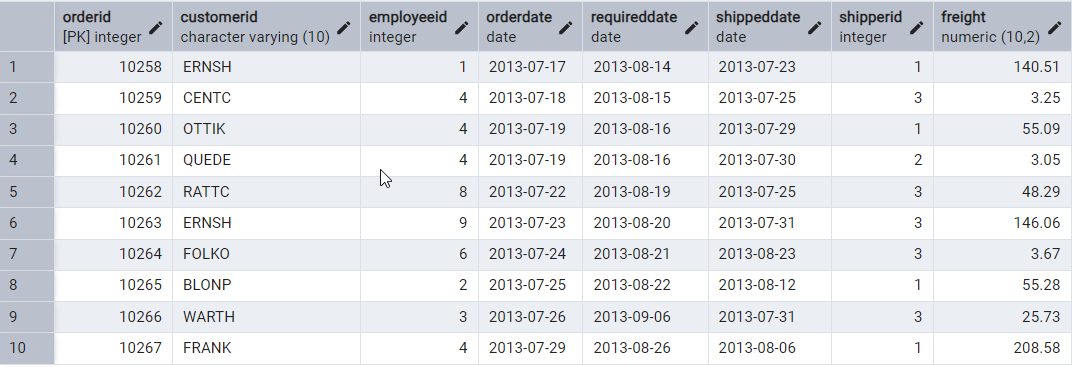


------------------------------------------------------------------------------------------------------------------------------------------

**--b. Retrieve orders starting from the 11th order, fetching 10 rows (i.e., fetch rows 11-20).**

SELECT \* FROM orders LIMIT 10 OFFSET 10;

**OUTPUT**:



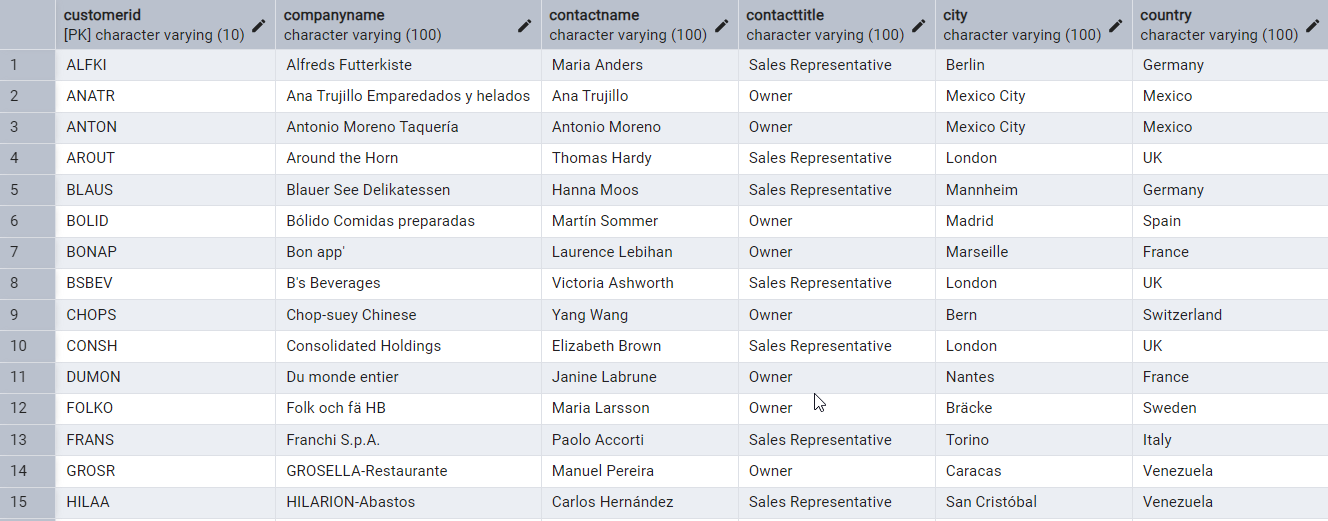
------------------------------------------------------------------------------------------------------------------------------------------**-- 6. Filtering (IN, BETWEEN)**

**--a. List all customers who are either Sales Representative, Owner**

SELECT \* FROM customers

WHERE contacttitle IN ('Sales Representative', 'Owner');

**OUTPUT**:



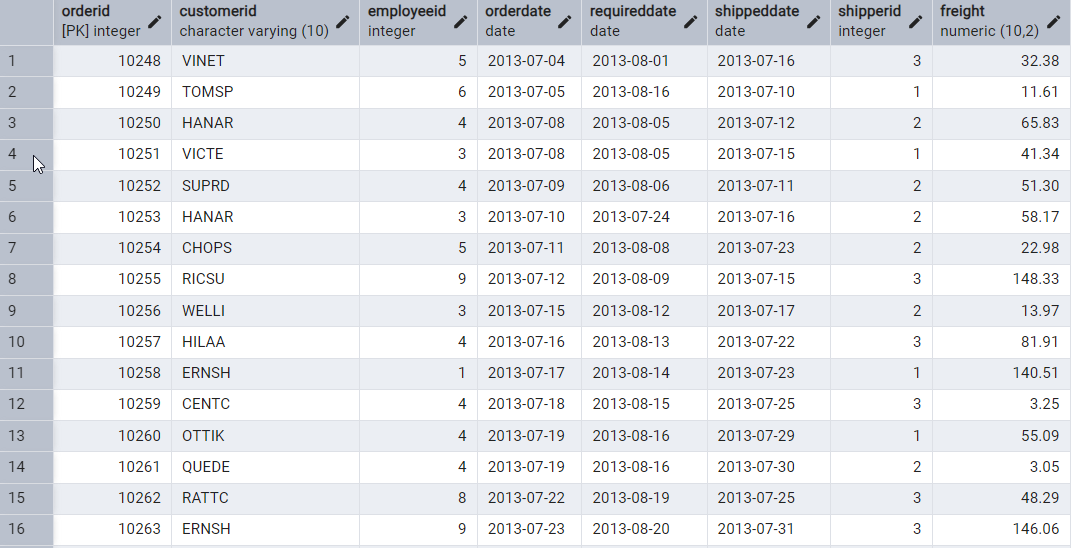
------------------------------------------------------------------------------------------------------------------------------------------

**-- b. Retrieve orders placed between January 1, 2013, and December 31, 2013.**

SELECT \* FROM orders

WHERE orderdate BETWEEN '2013-01-01' AND '2013-12-31';

**OUTPUT**:



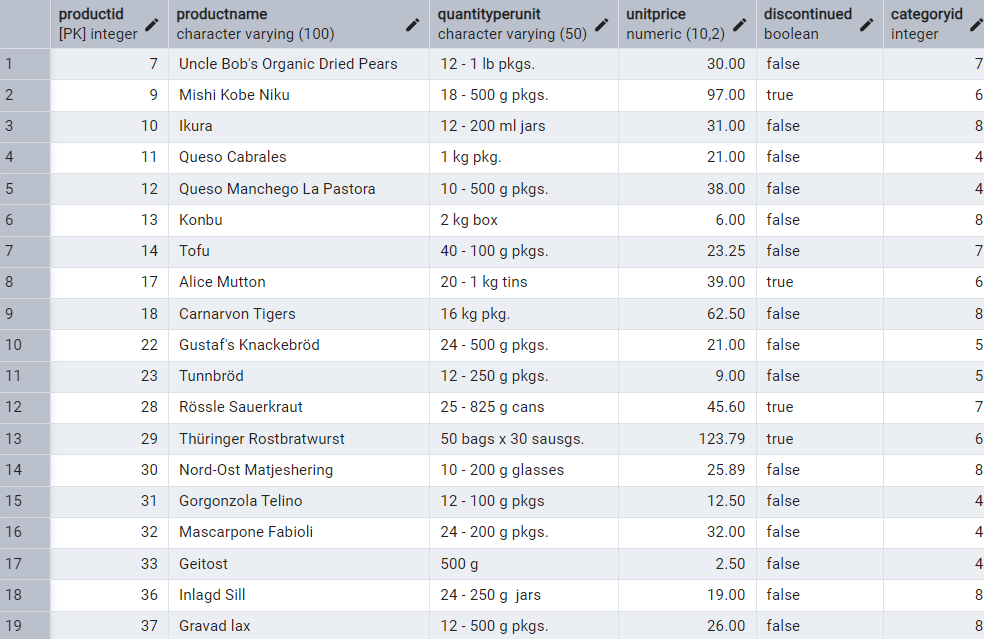
**-- 7. Filtering**

**-- a. List all products whose category\_id is not 1, 2, or 3.**

SELECT \* FROM products

WHERE categoryid NOT IN (1, 2, 3);

**OUTPUT**:



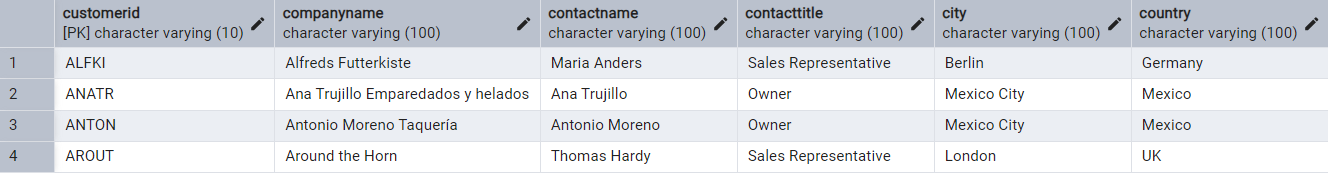
------------------------------------------------------------------------------------------------------------------------------------------

**-- b. Find customers whose company name starts with "A".**

SELECT \* FROM customers

WHERE companyname LIKE 'A%';

**OUTPUT**:



------------------------------------------------------------------------------------------------------------------------------------------**-- 8. INSERT into orders table:**

INSERT INTO orders (orderid, customerid, employeeid, orderdate, requireddate, shippeddate,

shipperid, freight)

VALUES (

11078,

'ALFKI',

5,

'2025-04-23',

'2025-04-30',

'2025-04-25',

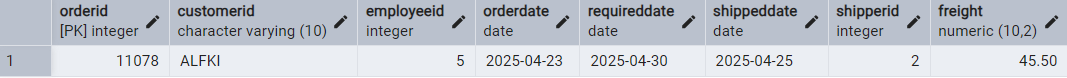
2,

45.50

);

**OUTPUT:**

SELECT \* FROM orders WHERE orderid = 11078;



------------------------------------------------------------------------------------------------------------------------------------------**-- 9. Increase (Update) the unit price of all products in category\_id =2 by 10%.**

UPDATE products

SET unitprice = unitprice \* 1.10

WHERE categoryid = 2;

**OUTPUT:**