SQL Day 2 Assignment

1.a) Alter Table- Add a new column linkedin_profile to employees table to store LinkedIn URLs as varchar

ALTER TABLE employees

ADD COLUMN linkedin_profile VARCHAR(255);



1.b) Change the linkedin_profile column data type from VARCHAR to TEXT

ALTER TABLE employees

ALTER COLUMN linkedin_profile TYPE TEXT;



1.c) Add unique, not null constraint to linkedin_profile

ALTER TABLE employees

ADD CONSTRAINT unique_linkedin_profile UNIQUE (linkedin_profile);

```
Data Output Messages Notifications

ALTER TABLE

Query returned successfully in 106 msec.
```

ALTER TABLE employees

ALTER COLUMN linkedin_profile SET NOT NULL;

```
ERROR: column "linkedin_profile" of relation "employees" contains null values

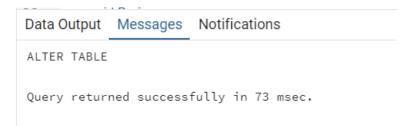
SQL state: 23502
```

The given order Id 11078 already exists in the dataset. Order id should be unique so its shows an error.

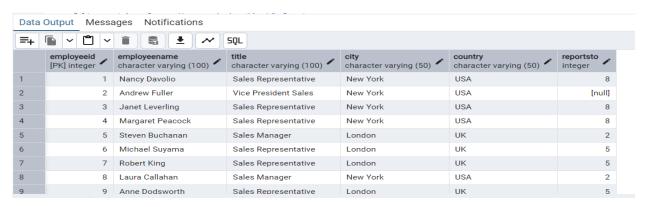
1.d) Drop column linkedin_profile

ALTER TABLE employees

DROP COLUMN linkedin_profile;



After dropping column



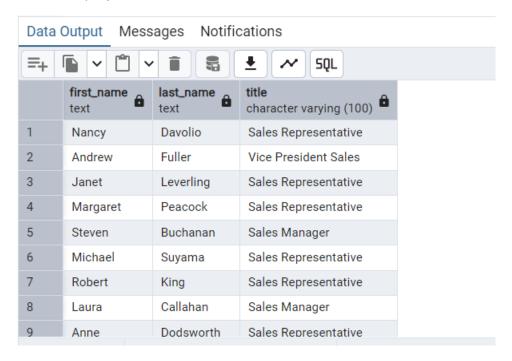
2.a) Querying (Select)

Retrieve the first name, last name, and title of all employees

SELECT

split_part(employeeName, ' ', 1) AS first_name,
split_part(employeeName, ' ', 2) AS last_name,
title

FROM employees;



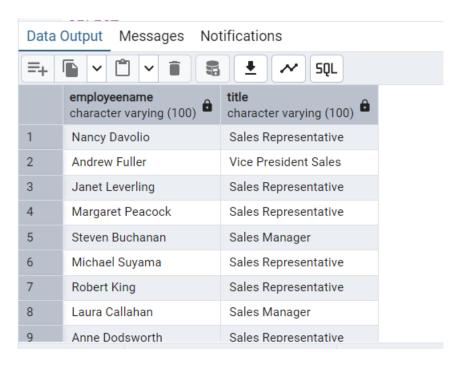
2.b) Retrieve the employee's name, and title of all employees

SELECT

employeeName,

title

FROM employees;

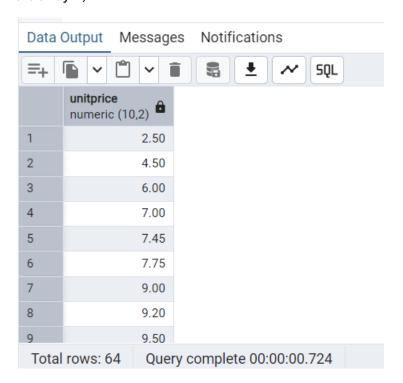


2.b) Find all unique unit prices of products

select distinct unitPrice

from products

order by 1;

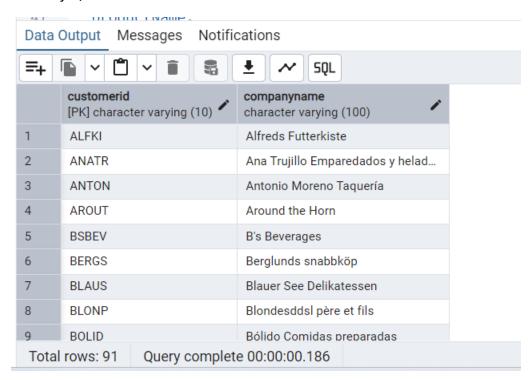


2.c) List all customers sorted by company name in ascending order

select customerid, companyname

from customers

order by 2;



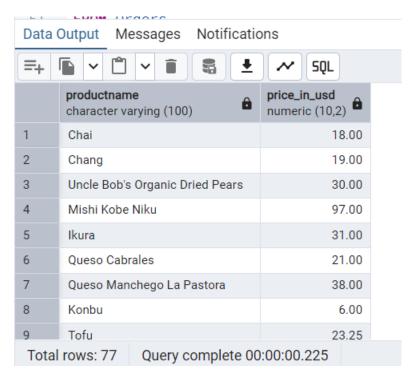
2.d) Display product name and unit price, but rename the unit_price column as price_in_usd

SELECT

productName,

unitPrice AS price_in_usd

FROM products;



3.a) Filtering

Get all customers from Germany

SELECT *

FROM Customers

WHERE Country = 'Germany';



3.b) Find all customers from France or Spain

SELECT*

FROM Customers

WHERE Country IN ('France', 'Spain');



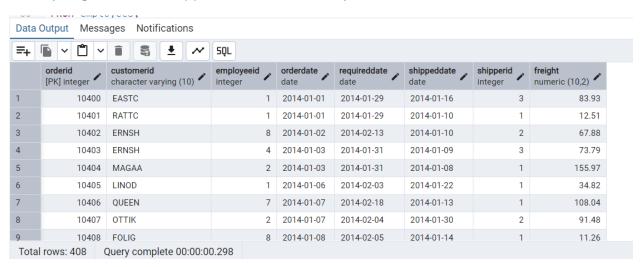
3.c) Retrieve all orders placed in 2014 (based on order_date), and either have freight greater than 50 or the shipped date available (i.e., non-NULL) (Hint: EXTRACT(YEAR FROM order_date))

SELECT*

FROM Orders

WHERE EXTRACT(YEAR FROM OrderDate) = 2014

AND (Freight > 50 OR ShippedDate IS NOT NULL);

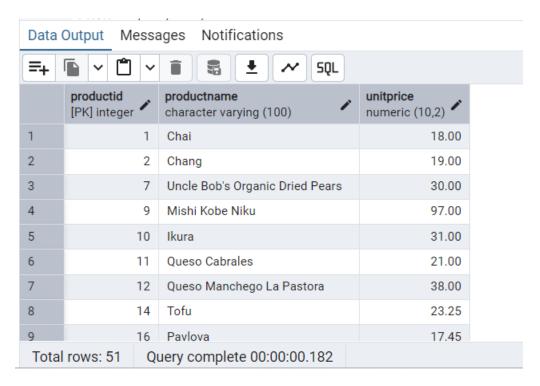


4) Filtering-Retrieve the product_id, product_name, and unit_price of products where the unit_price is greater than 15

SELECT ProductID, ProductName, UnitPrice

FROM Products

WHERE UnitPrice > 15;

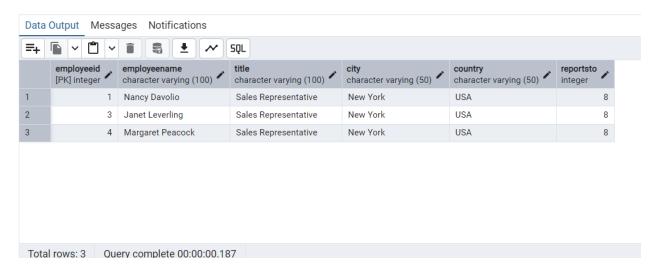


4.b) List all employees who are in the USA and have the title "Sales Representative".

SELECT*

FROM Employees

WHERE Country = 'USA' AND Title = 'Sales Representative';

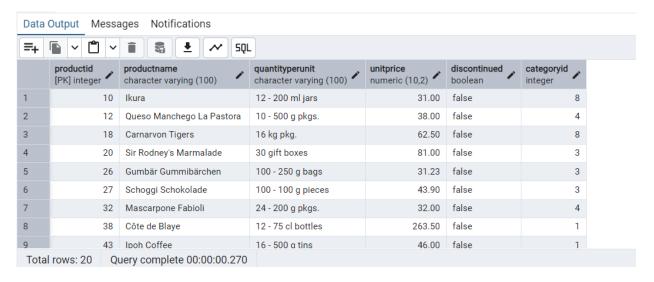


4.c) Retrieve all products that are not discontinued and priced greater than 30.

SELECT*

FROM Products

WHERE Discontinued = FALSE AND UnitPrice > 30;



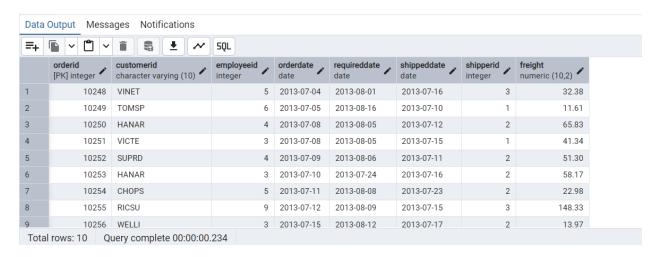
5.a) LIMIT/FETCH

• Retrieve the first 10 orders from the orders table.

SELECT*

FROM Orders

LIMIT 10;

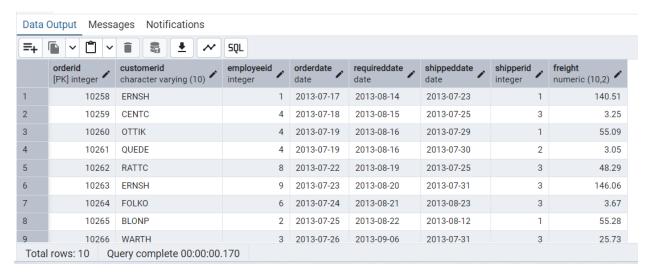


5.b) Retrieve orders starting from the 11th order, fetching 10 rows (i.e., fetch rows 11-20).

SELECT*

FROM Orders

OFFSET 10 LIMIT 10;



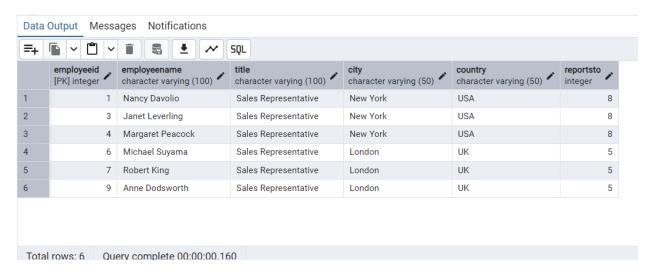
6.a) Filtering (IN, BETWEEN)

● List all customers who are either Sales Representative or Owner

SELECT*

FROM Employees

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

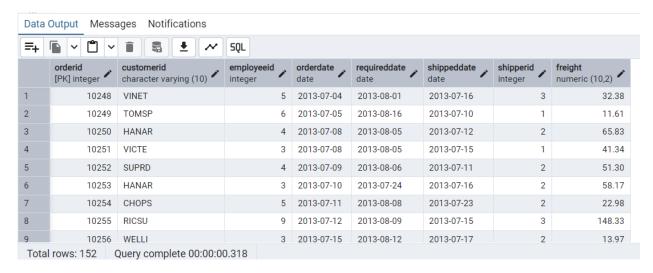


6.b) Retrieve orders placed between January 1, 2013, and December 31, 2013.

SELECT*

FROM Orders

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



7.a) Filtering

● List all products whose category_id is not 1, 2, or 3.

SELECT*

FROM Products

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



7.b) Find customers whose company name starts with "A".

SELECT*

FROM Customers

WHERE CompanyName LIKE 'A%';



8) INSERT into orders table:

Task: Add a new order to the orders table with the following details:

Order ID: 11078

Customer ID: ALFKI

Employee ID: 5

Order Date: 2025-04-23

Required Date: 2025-04-30

Shipped Date: 2025-04-25

shipperID:2

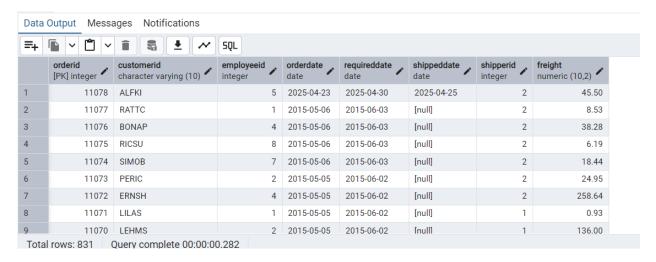
Freight: 45.50

Insert into orders

values(11078, 'ALFKI', 5, '2025-04-23', '2025-04-30', '2025-04-25', 2, 45.50);

select * from orders

order by 1 desc;



9) Increase(Update) the unit price of all products in category_id =2 by 10%.(HINT: unit_price = unit_price * 1.10)

Before update

select * from products where categoryID =2;



After update

update products set "unitPrice" = "unitPrice" * 1.10 where categoryID = 2; select * from products where categoryID =2;



10) Download the northwind.sql file into the 'northwind' database using pgadmin

