**NUMPY NINJA SQL BOOTCAMP APRIL 2025**

**DAY 2 SQL ASSIGNMENT**

**Greeshma Ravula DA 164**

**QUESTION 1)**

**Alter Table:**

* Add a new column linkedin\_profile to employees table to store LinkedIn URLs as varchar.
* Change the linkedin\_profile column data type from VARCHAR to TEXT.
* Add unique, not null constraint to linkedin\_profile
* Drop column linkedin\_profile

**QUERY DETAILS**

* Add a new column

ALTER TABLE employees

ADD COLUMN linkedin\_profile VARCHAR(500);

select \* from employees;

* Change datatype

ALTER COLUMN linkedin\_profile

SET DATA TYPE text;

* Add unique, not null constraint to linkedin\_profile

ALTER TABLE employees

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

UPDATE employees

SET linkedin\_profile = 'https://example.com/placeholder\_' || employeeid

WHERE linkedin\_profile IS NULL;

ALTER TABLE employees

ALTER COLUMN linkedin\_profile SET NOT NULL;

* Drop column

ALTER TABLE employees

DROP COLUMN linkedin\_profile;

select \* from employees;

**SCREENSHOTS:**





**QUESTION 2)**

**Querying (Select)**

* Retrieve the employee name and title of all employees
* Find all unique unit prices of products
* List all customers sorted by company name in ascending order
* Display product name and unit price, but rename the unit\_price column as price\_in\_usd

**QUERY DETAILS**

* Retrieve the employee name and title of all employees

select employeename, title

from employees

* Unique unit prices of products

SELECT DISTINCT unitprice FROM products

* Customers sorted by company name in ascending orderSELECT \* FROM customers ORDER BY companyname ASC;
* 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;

**SCREENSHOTS:**





**QUESTION 3)**

**Filtering**

* Get all customers from Germany.
* Find all customers from France or Spain
* 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))

**QUERY DETAILS**

* Get all customers from Germany.

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

* Find all customers from France or Spain

SELECT \* from customers WHERE country IN ( 'France','Spain');

* 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);

**SCREENSHOTS:**







**QUESTION 4)**

**Filtering**

* Retrieve the product\_id, product\_name, and unit\_price of products where the unit\_price is greater than 15.
* List all employees who are located in the USA and have the title "Sales Representative".
* Retrieve all products that are not discontinued and priced greater than 30.

**QUERY DETAILS**

* 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;

* 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';

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

SELECT \* FROM products WHERE discontinued =0 AND unitPrice >30;

**SCREENSHOTS:**







**QUESTION 5)**

**LIMIT/FETCH**

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

**QUERY DETAILS**

* Retrieve the first 10 orders from the orders table.

SELECT \* from orders limit 10;

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

SELECT \* from orders offset 10 limit 10;

**SCREENSHOTS:**





**QUESTION 6)**

**Filtering (IN, BETWEEN)**

* List all customers who are either Sales Representative, Owner
* Retrieve orders placed between January 1, 2013, and December 31, 2013.

**QUERY DETAILS**

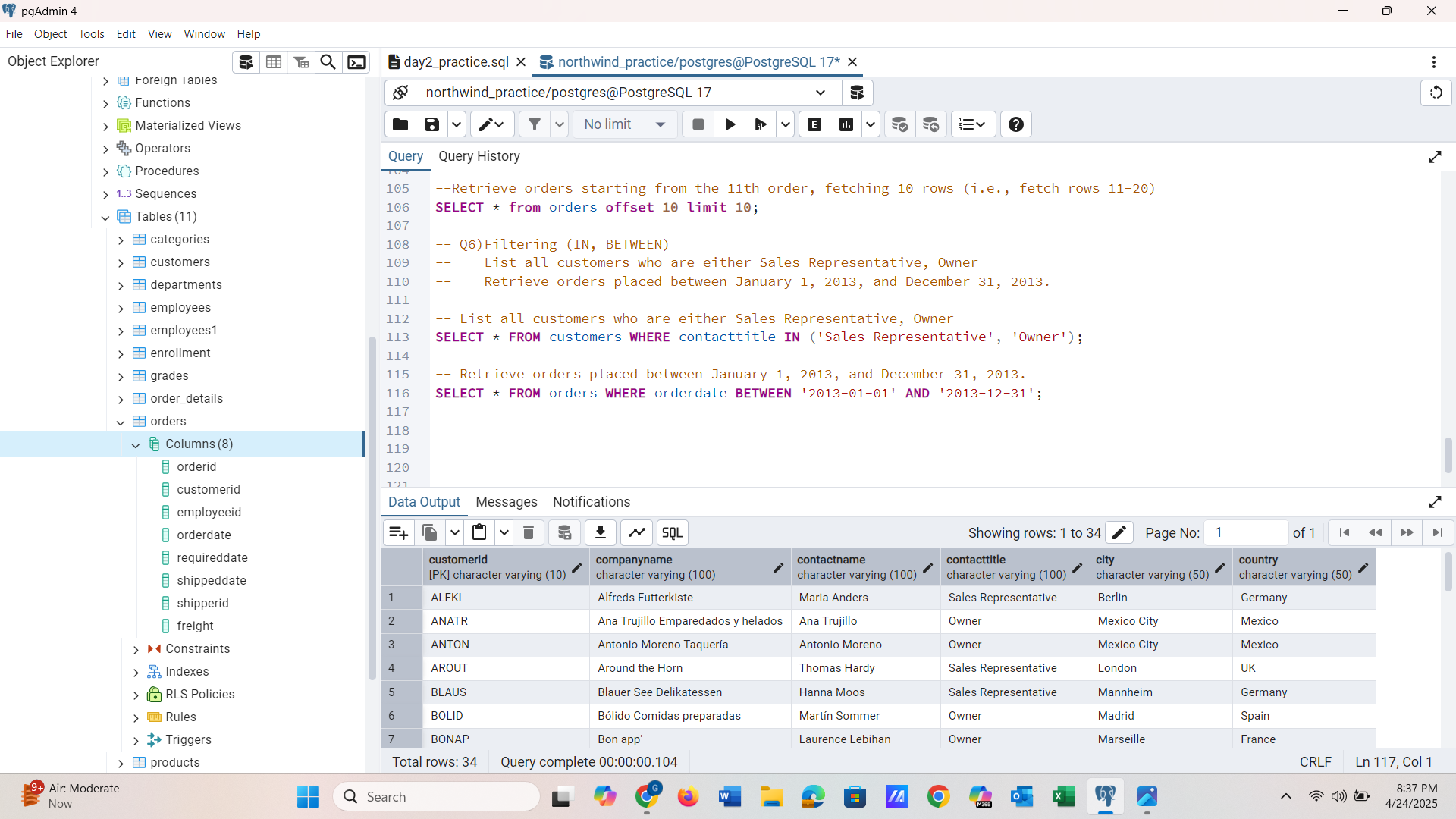
* List all customers who are either Sales Representative, Owner

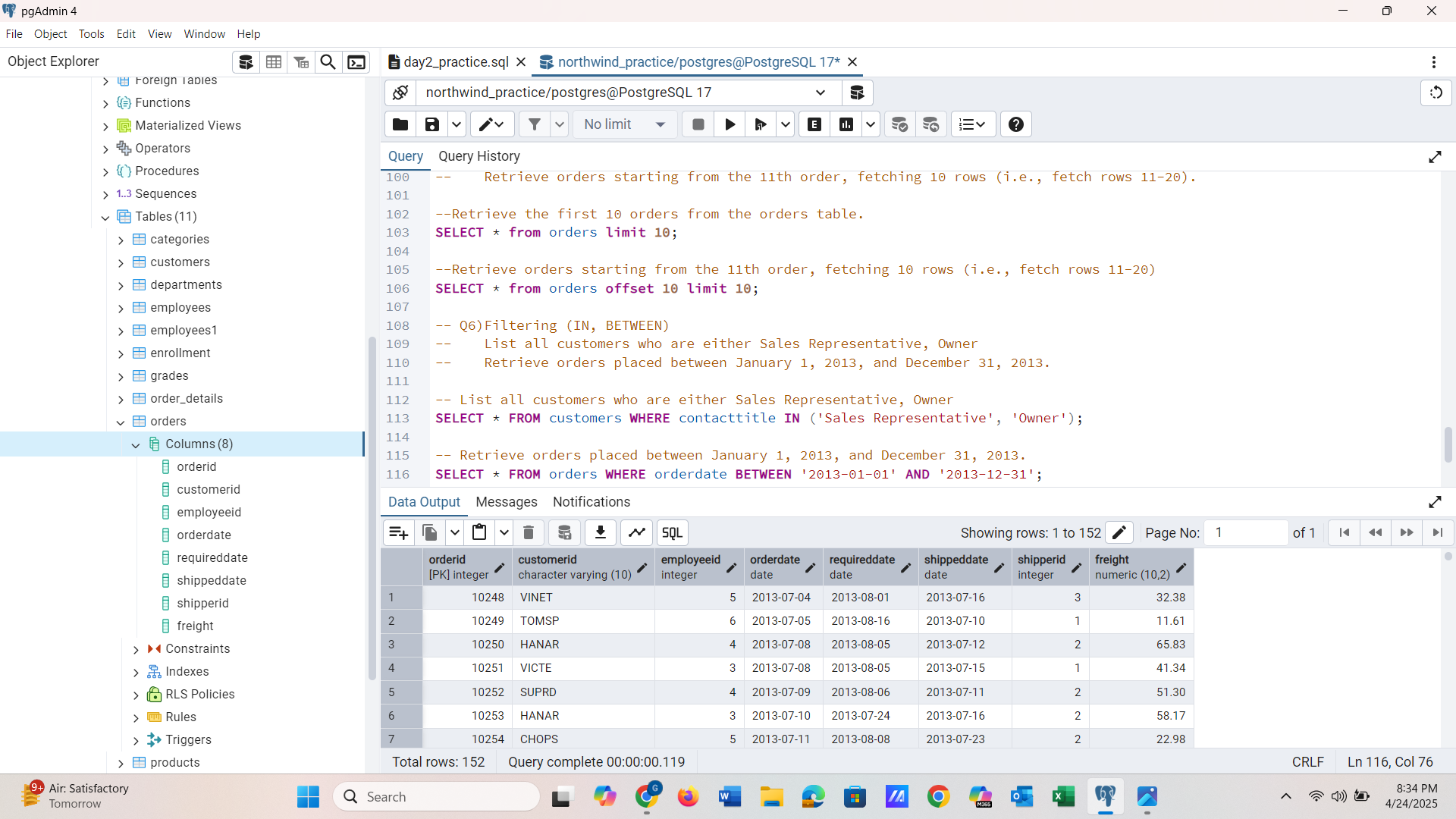
SELECT \* FROM customers WHERE contacttitle IN ('Sales Representative', 'Owner');

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

SELECT \* FROM orders WHERE orderdate BETWEEN '2013-01-01' AND '2013-12-31';

**SCREENSHOTS:**





**QUESTION 7)**

**Filtering**

* List all products whose category\_id is not 1, 2, or 3.
* Find customers whose company name starts with "A".

**QUERY DETAILS**

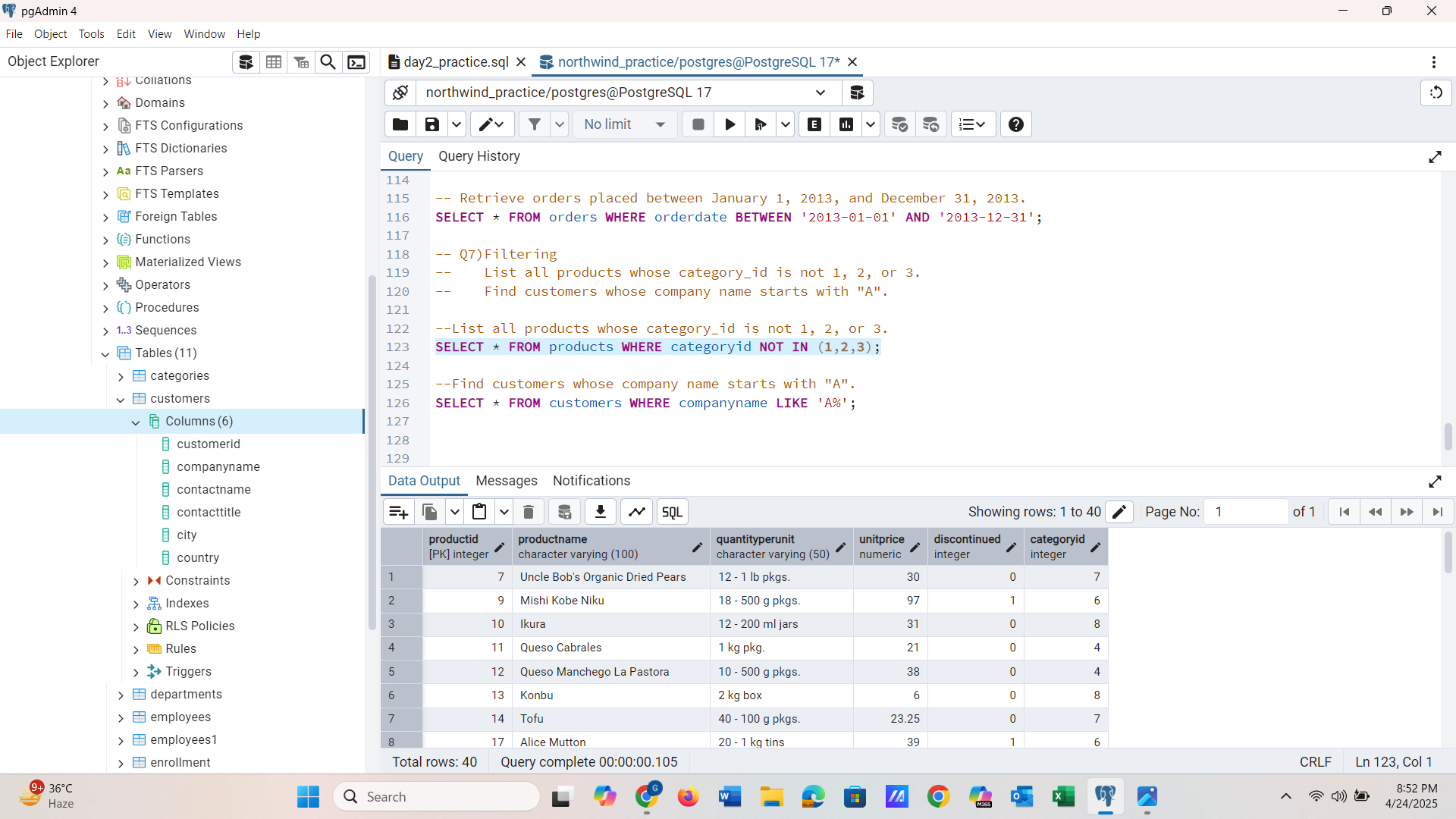
* List all products whose category\_id is not 1, 2, or 3.

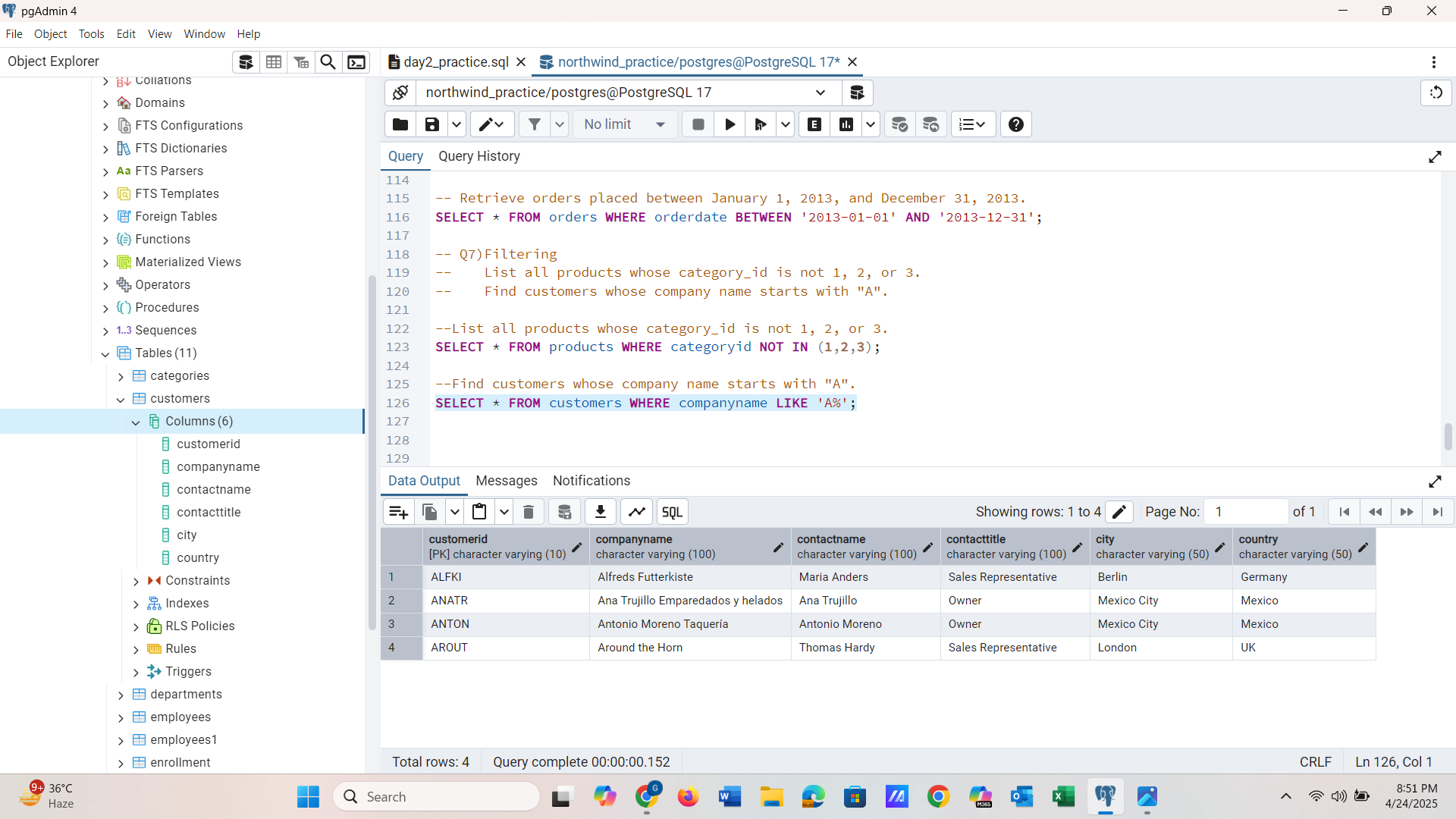
SELECT \* FROM products WHERE categoryid NOT IN (1,2,3);

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

SELECT \* FROM customers WHERE companyname LIKE 'A%';

**SCREENSHOTS:**





**QUESTION 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

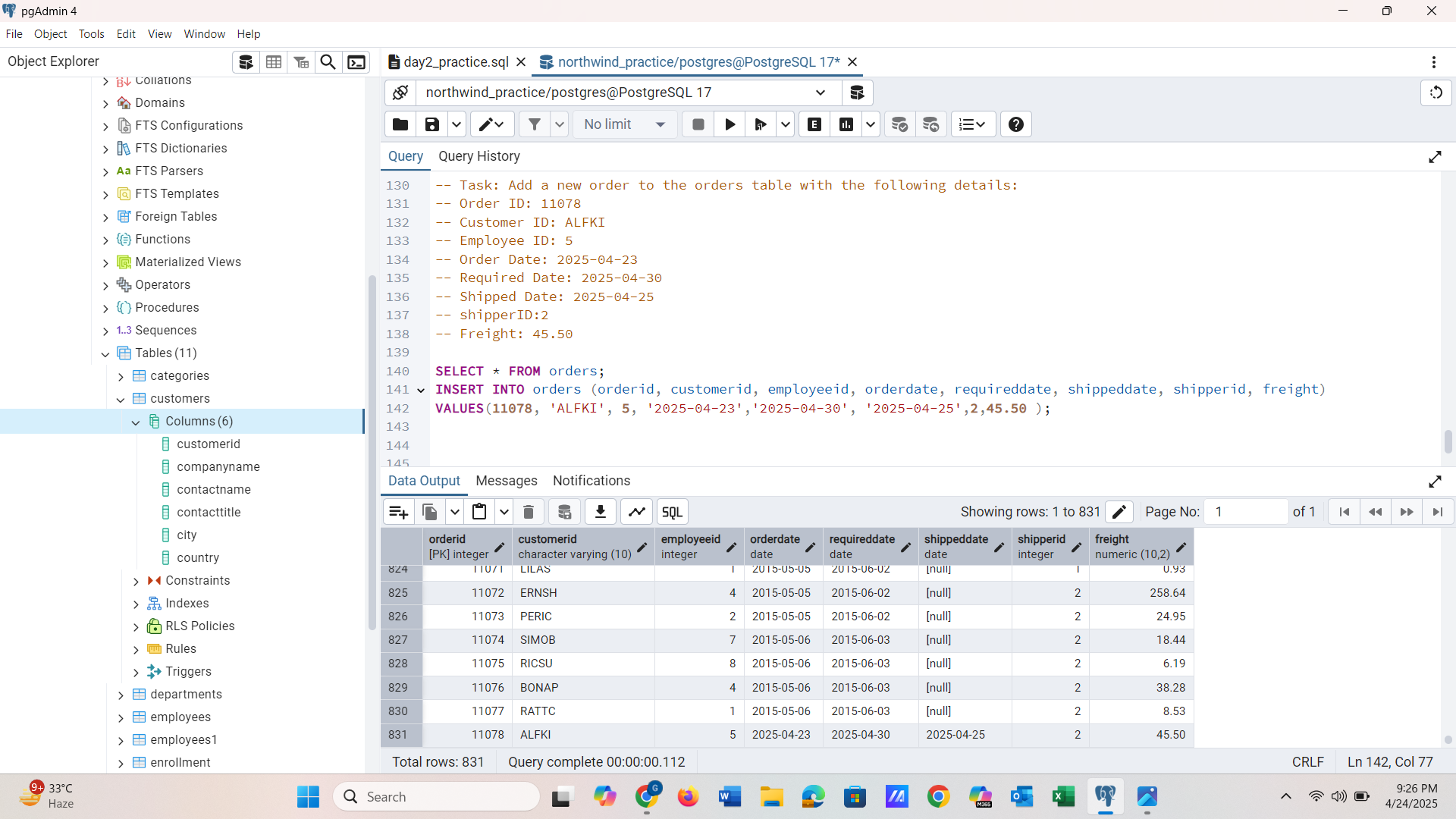
**QUERY DETAILS**

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 )

**SCREENSHOTS:**

****

**QUESTION 9)**

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

(HINT: unit\_price =unit\_price \* 1.10)

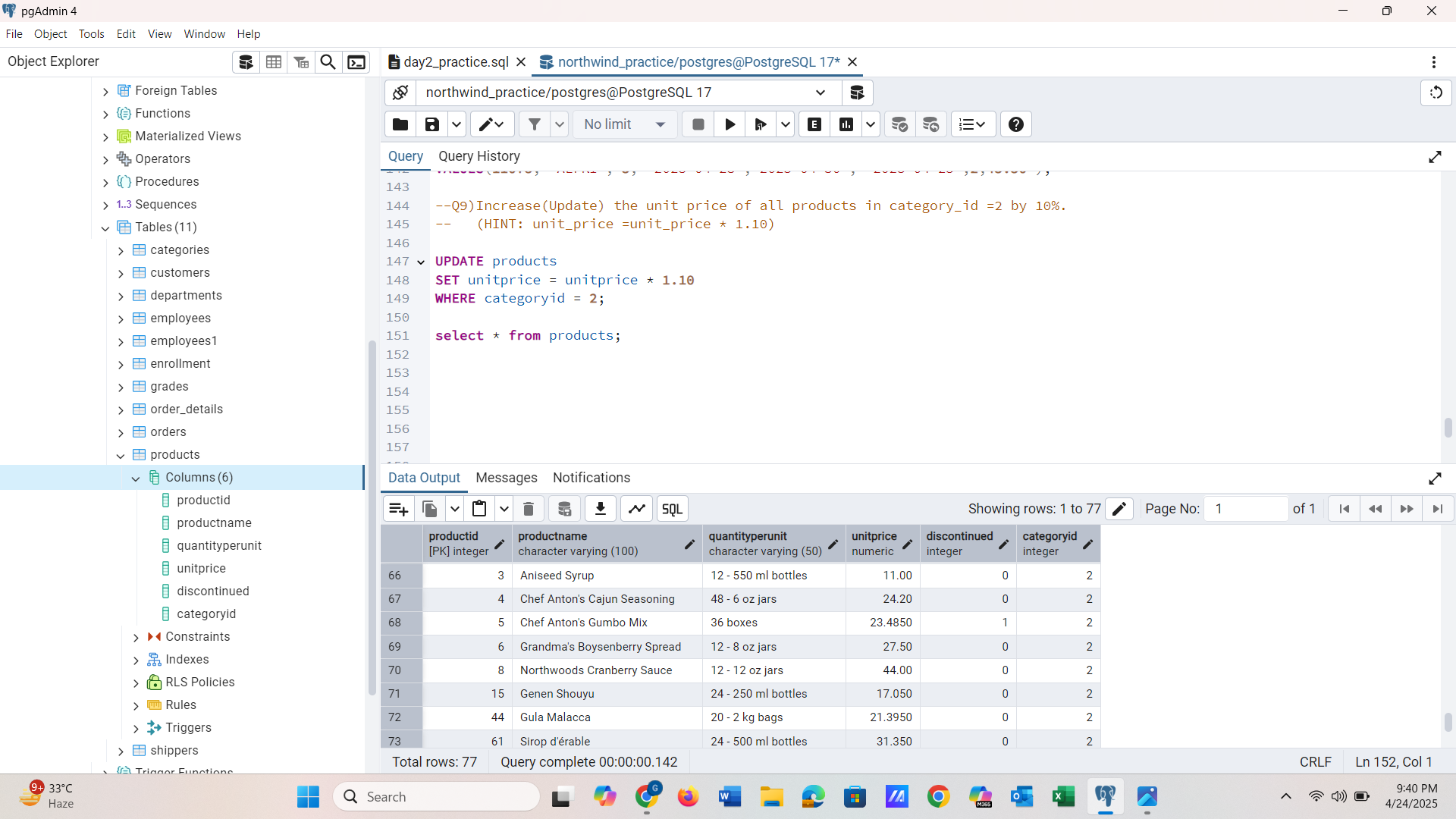
**QUERY DETAILS**

UPDATE products

SET unitprice = unitprice \* 1.10

WHERE categoryid = 2;

**SCREENSHOTS:**



**QUESTION 10)**

**Sample Northwind database:**

Download

1. Download northwind.sql from below link into your local. Sign in to Git first <https://github.com/pthom/northwind_psql>
2. Manually Create the database using pgAdmin:
   1. Right-click on "Databases" → Create → Database
   2. Give name as ‘northwind’ (all small letters)
   3. Click ‘Save’

Import database:

1. Open pgAdmin and connect to your server
2. Select the database ‘northwind’
3. Right Click-> Query tool.
4. Click the folder icon to open your northwind.sql file
5. Press F5 or click the Execute button.
6. You will see total 14 tables loaded
7. Databases → your database → Schemas → public → Tables

**SCREENSHOTS:**

