Name: Muhamad Fadhli Akbar

Team: Cortana

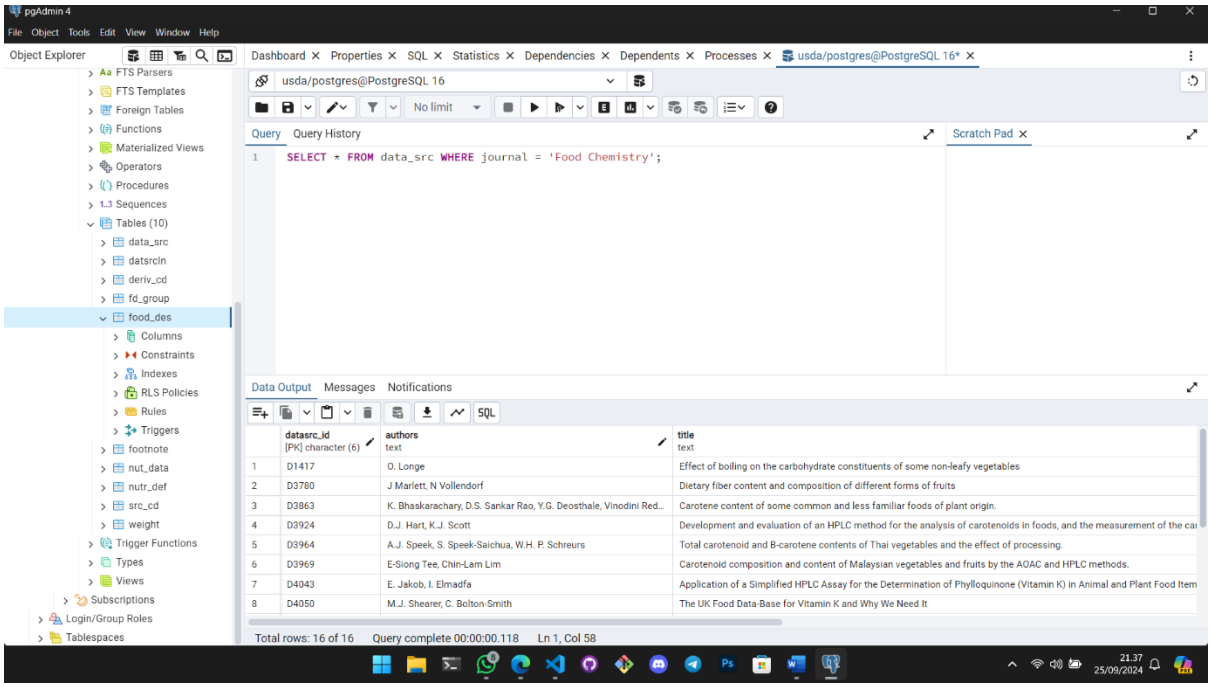
Instructor: Diva Kartika

Assigment SQL 2.1

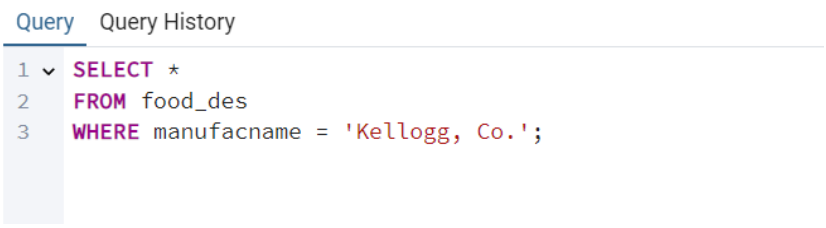
Open the usda database and get a query window for that database. Make sure you are in the right database by looking at the text at the top of the query window. It should reference usda and not northwind where we have been working. Run the following queries.

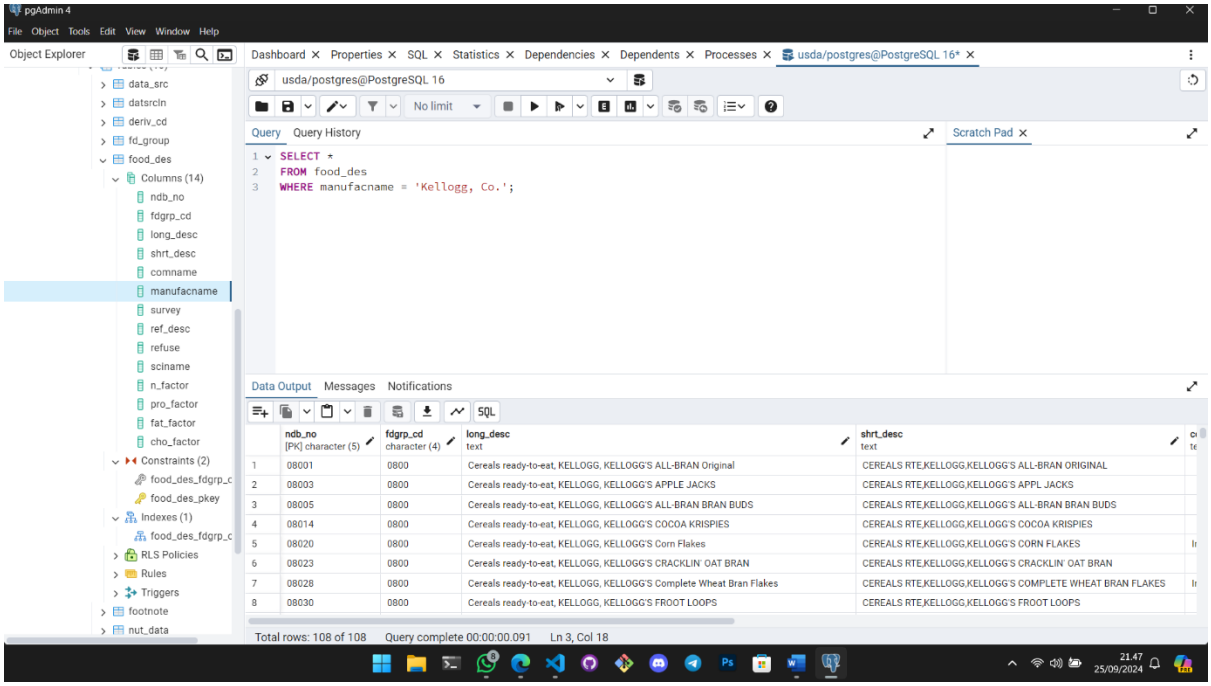
1. Select all records from data_src which came from the journal named 'Food Chemistry'

```
1    SELECT * FROM data_src WHERE journal = 'Food Chemistry';
```

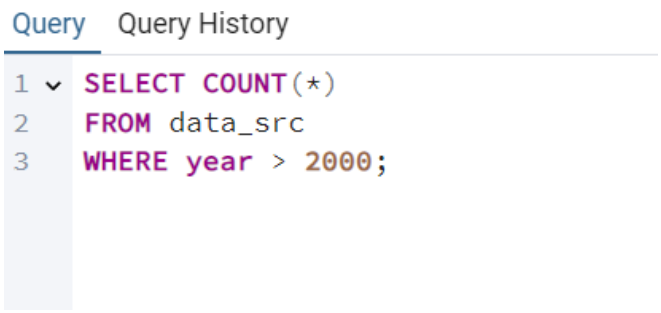


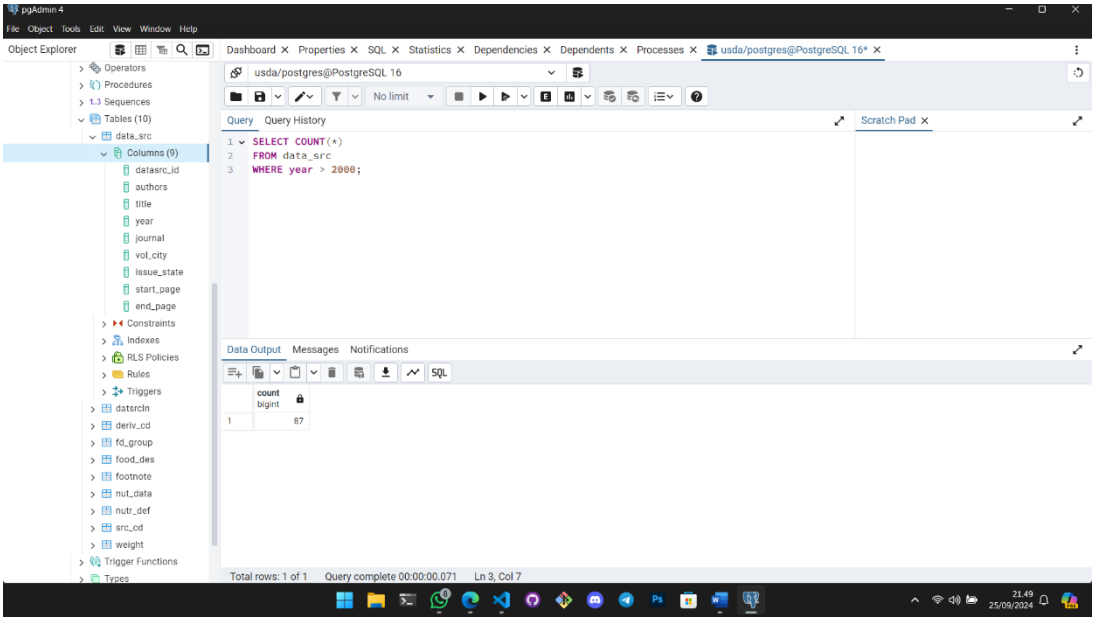
2. Find all the food descriptions (food_des) records for manufacturer Kellogg, Co. (must include punctuation for exact match)





3. Find the number of records in data sources (data_src) that were published after year 2000 (it is numeric field)

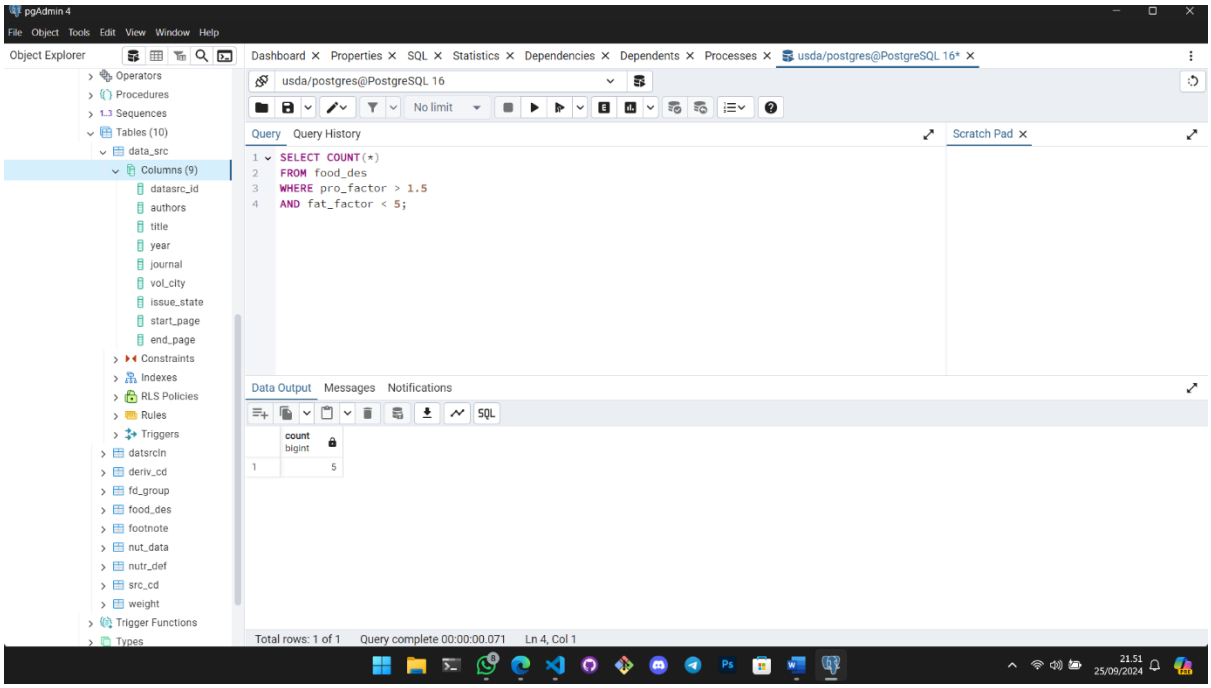




4. Find the number of records in food description table that have pro_factor greater than 1.5 and fat_factor less than 5 Practice What You've Learned

Query Query History

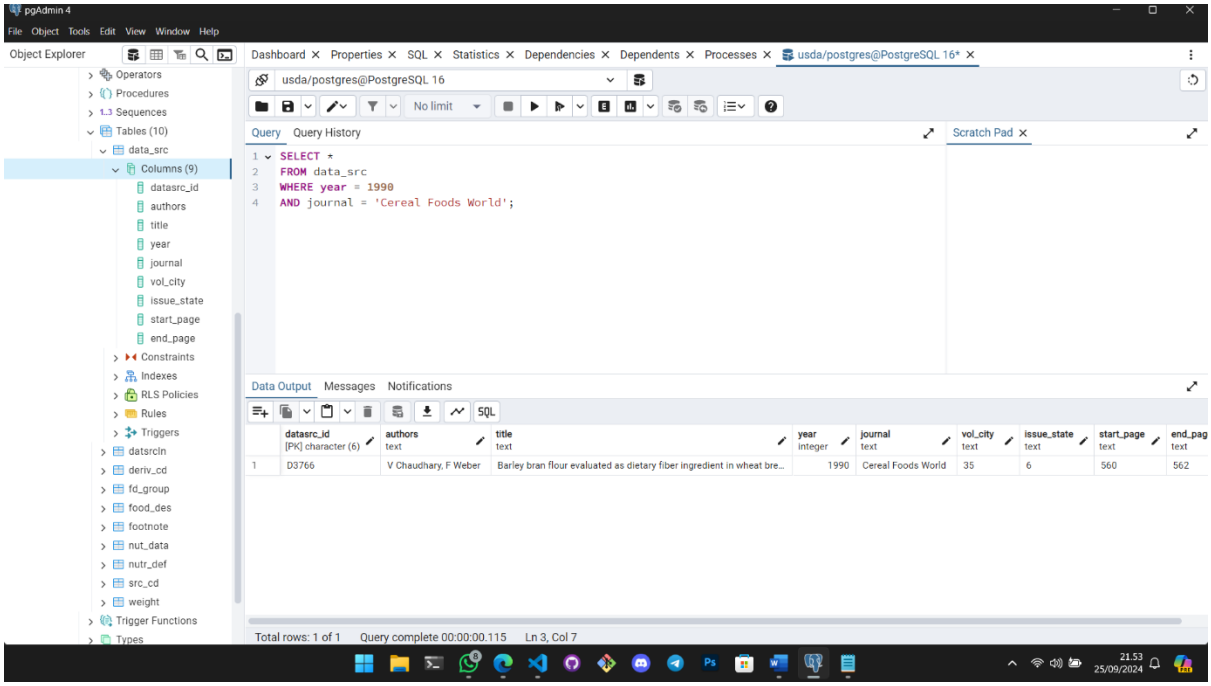
```
1  SELECT COUNT(*)
2  FROM food_des
3  WHERE pro_factor > 1.5
4  AND fat_factor < 5;
```



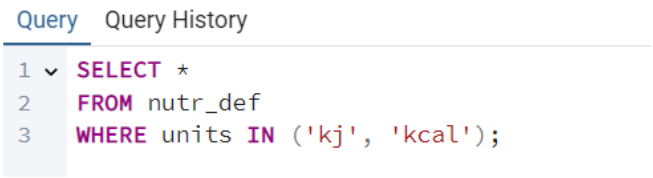
5. Find the record in data source table that is from year 1990 and the journal Cereal Foods World

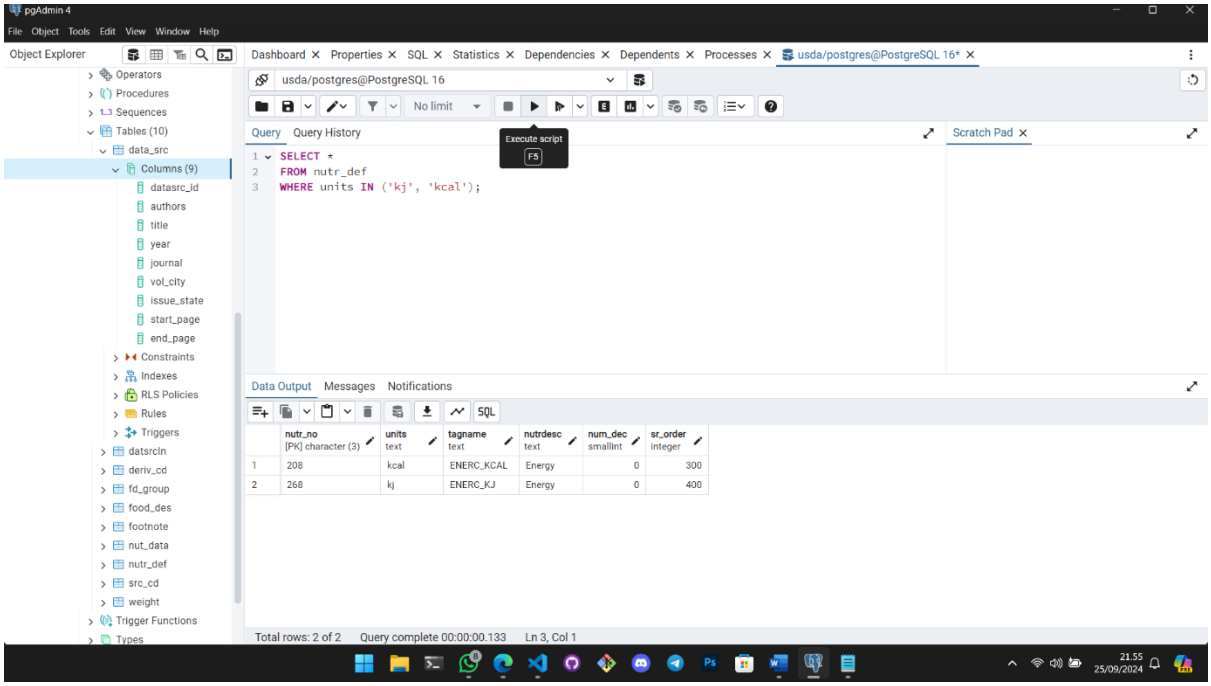
Query Query History

```
1  SELECT *
2  FROM data_src
3  WHERE year = 1990
4  AND journal = 'Cereal Foods World';
```



6. Select the records in nutr_def table (nutrition definitions) that have units of kj or kcal

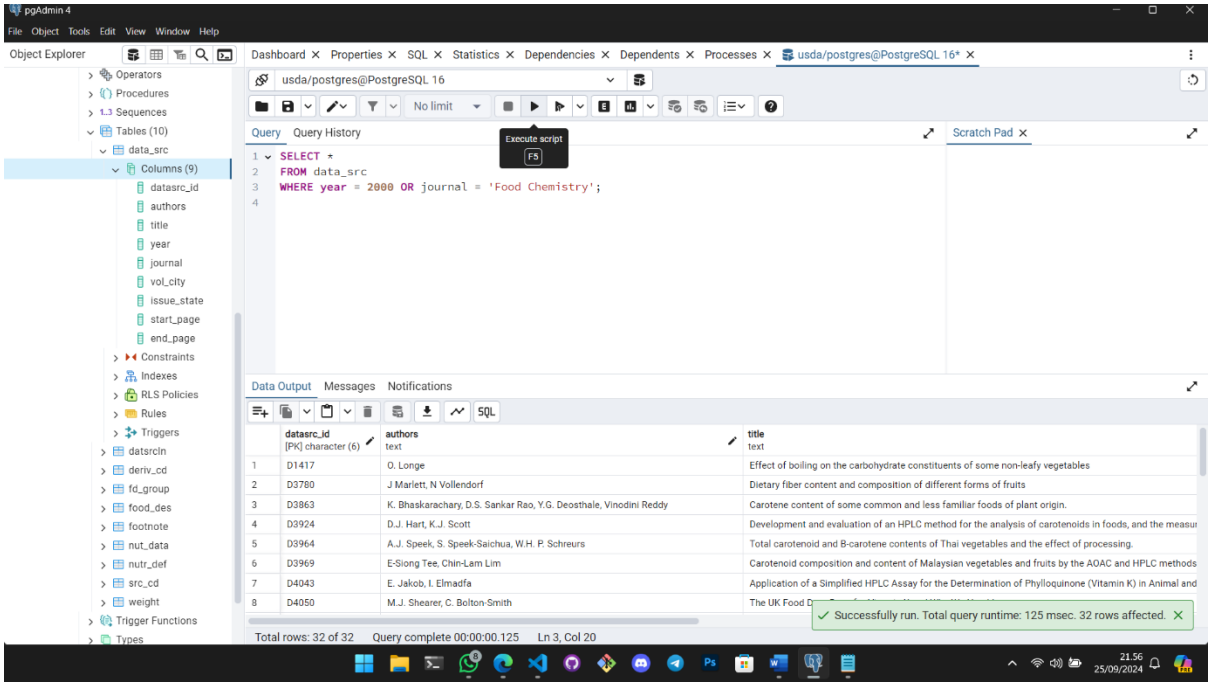




7. Select all records from data source table (data_src) that where from the year 2000 or the journal Food Chemistry

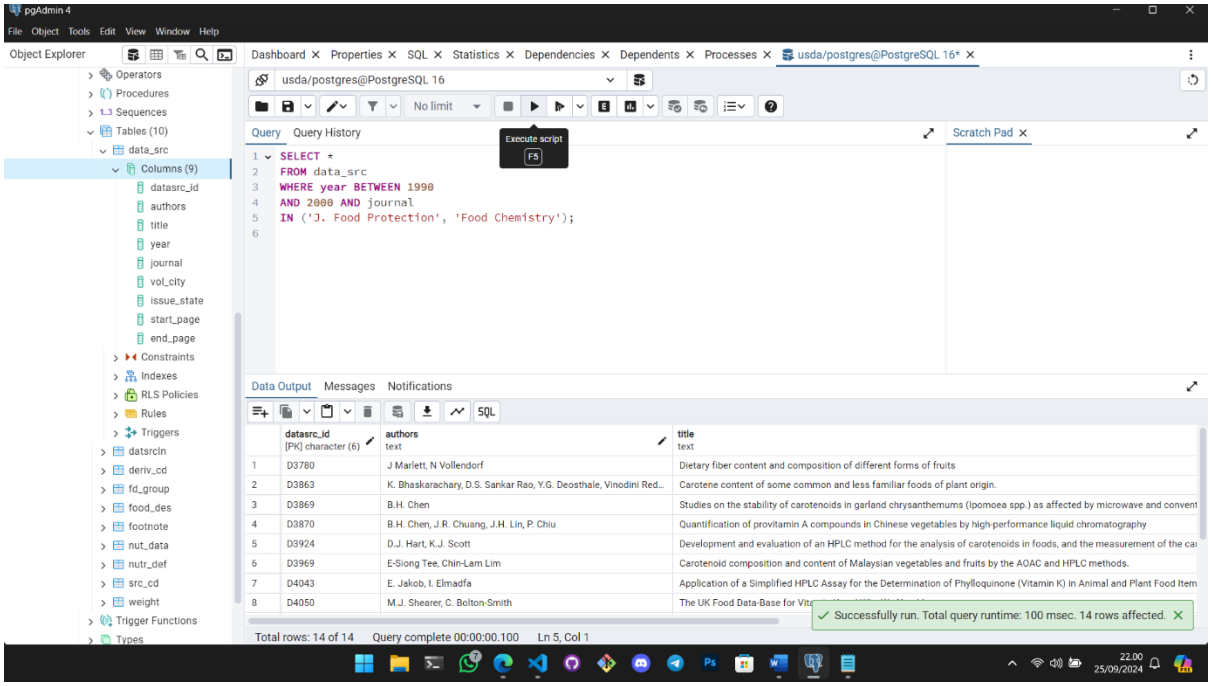
Query Query History

```
1 SELECT *
2 FROM data_src
3 WHERE year = 2000 OR journal = 'Food Chemistry';
4
```

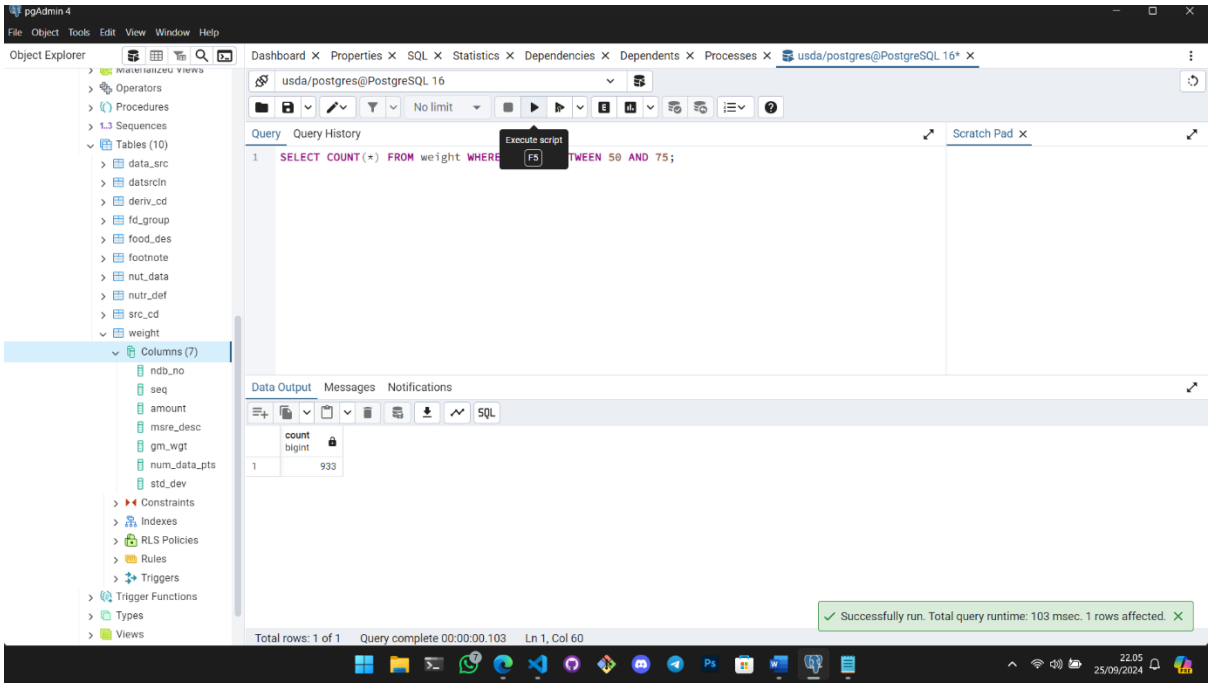


8. Find all the records in data sources that where between 1990 and 2000 and either ‘J. Food Protection’ or ‘Food Chemistry’

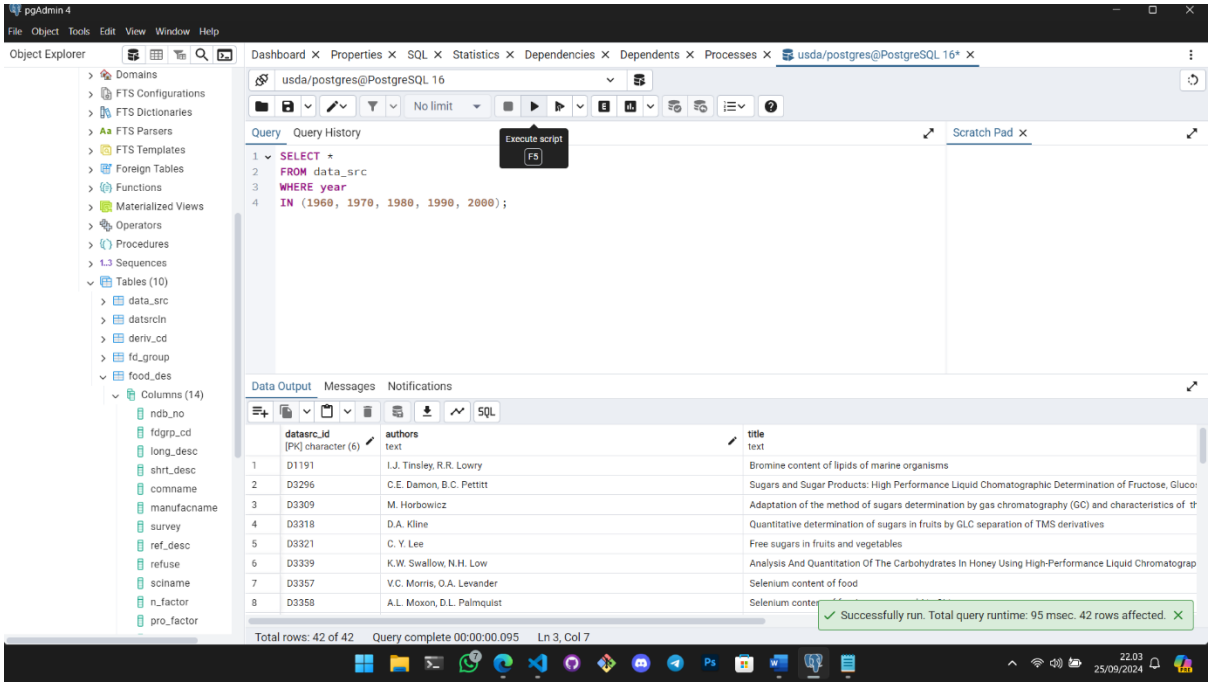
```
SELECT *
FROM data_src
WHERE year BETWEEN 1990
AND 2000 AND journal
IN ('J. Food Protection', 'Food Chemistry');
```



9. Use BETWEEN syntax to find number of weight records that weight between 50 and 75 grams (gm_wgt)



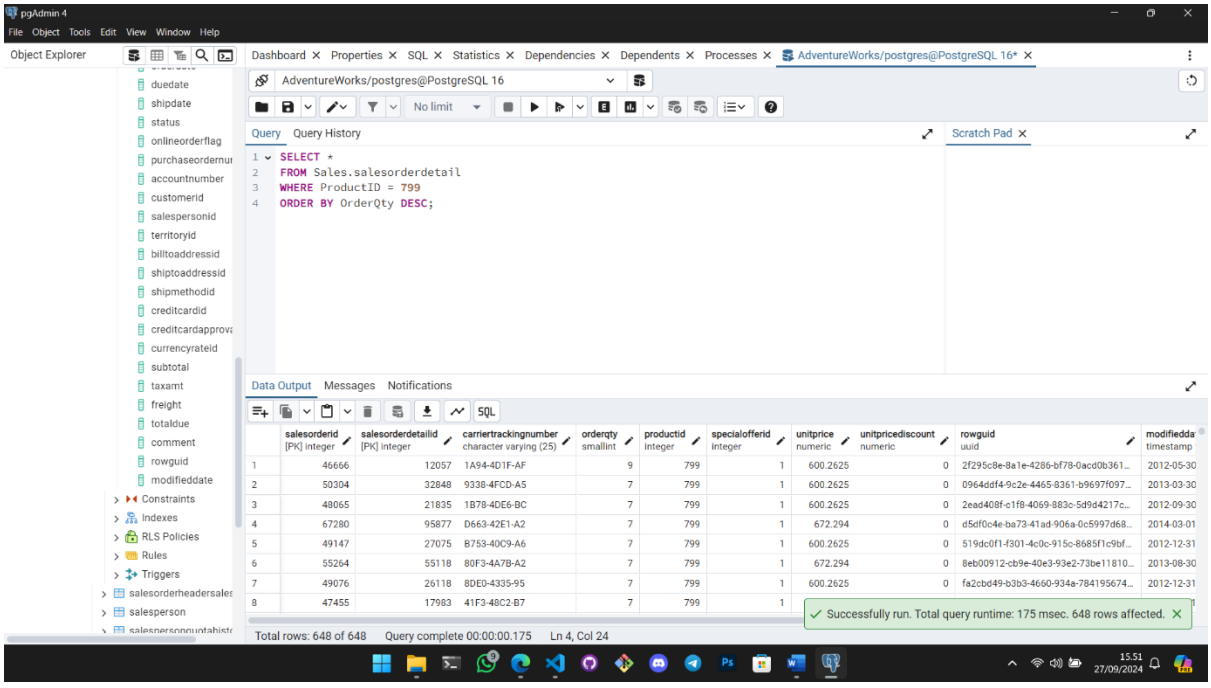
10. Select all records from the data source table that were published in years 1960, 1970, 1980, 1990 and 2000. Use the IN syntax



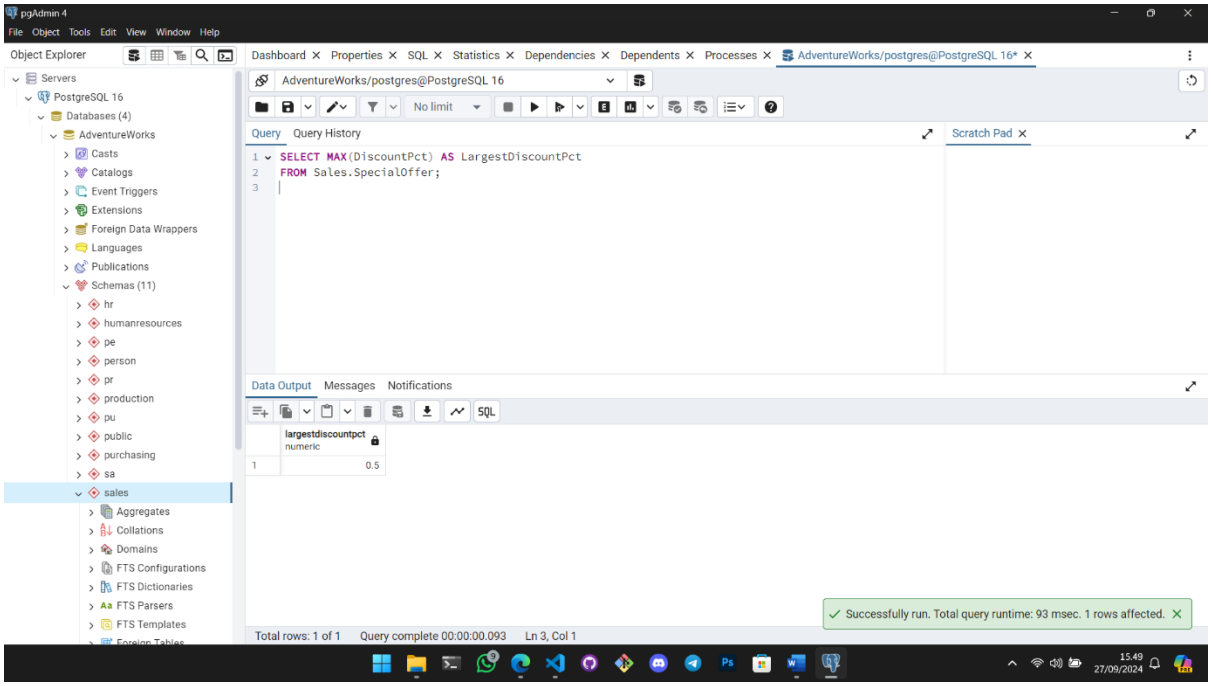
Assigment SQL 2.2

Open the AdventureWorks database and get a query window for that database. Make sure you are in the right database by looking at the text at the top of the query window. Run the following queries.

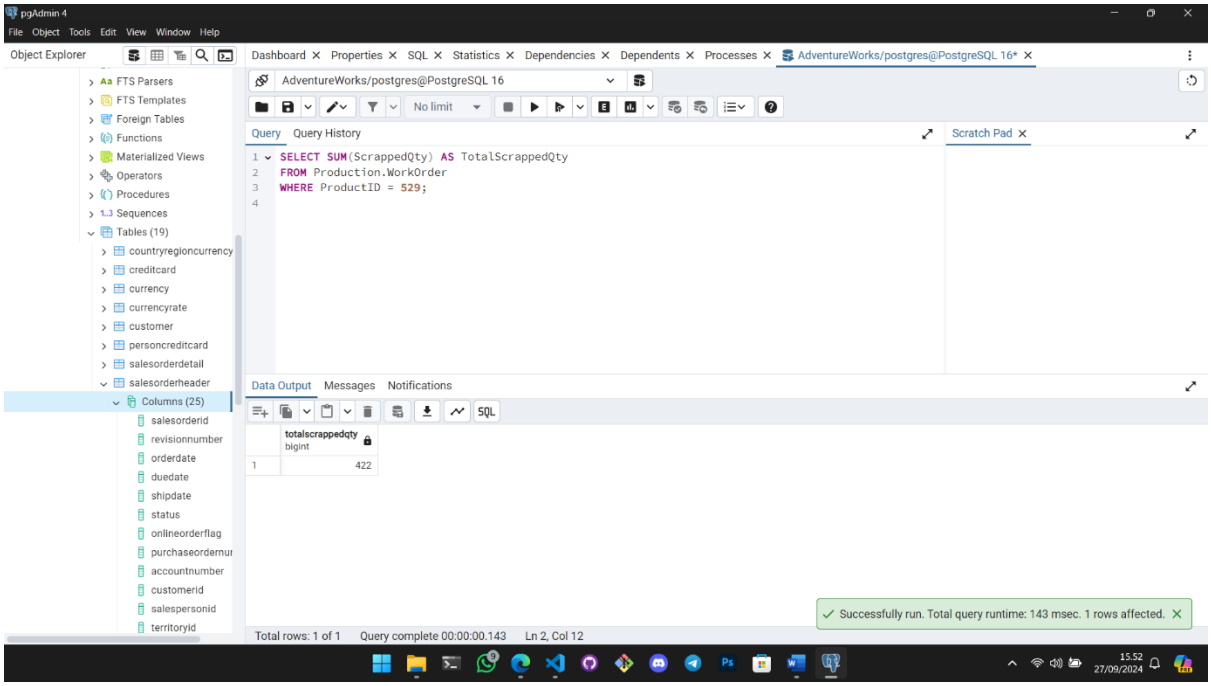
1. Find all the salesorderdetail records to productid 799 and order them by largest orderqty to smallest



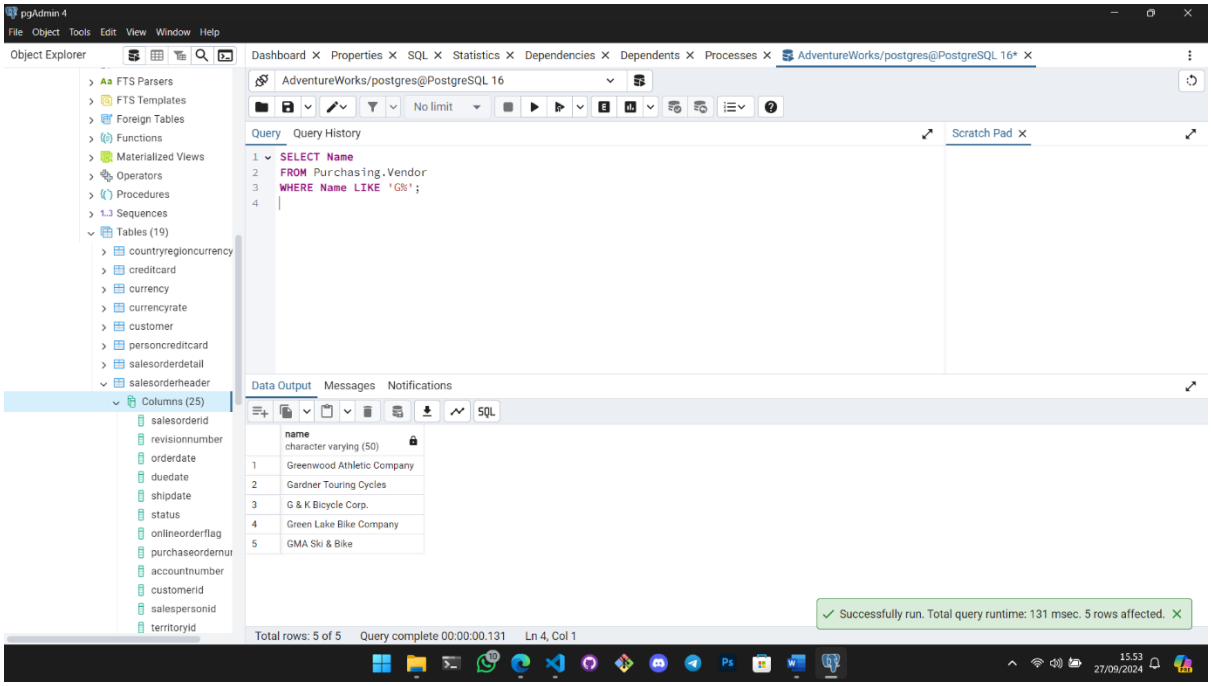
2. What is the largest discount percentage offered in the specialoffer table



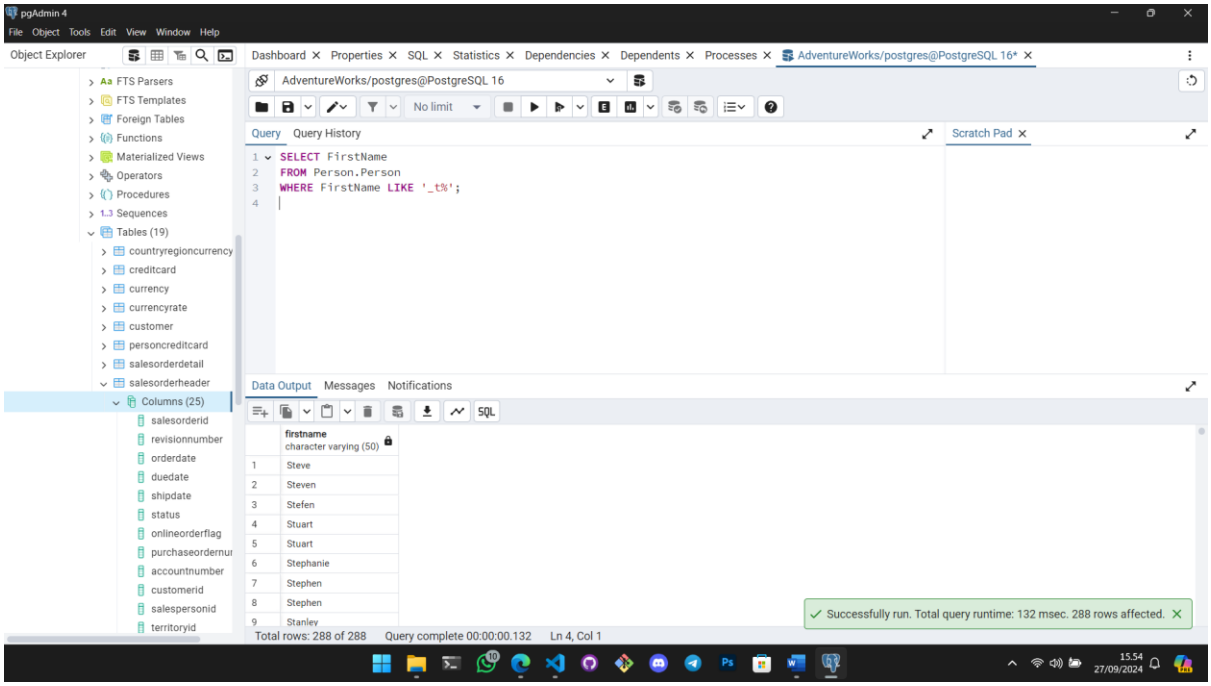
3. Find the sum of scrappedqty from the workorder table for productid 529



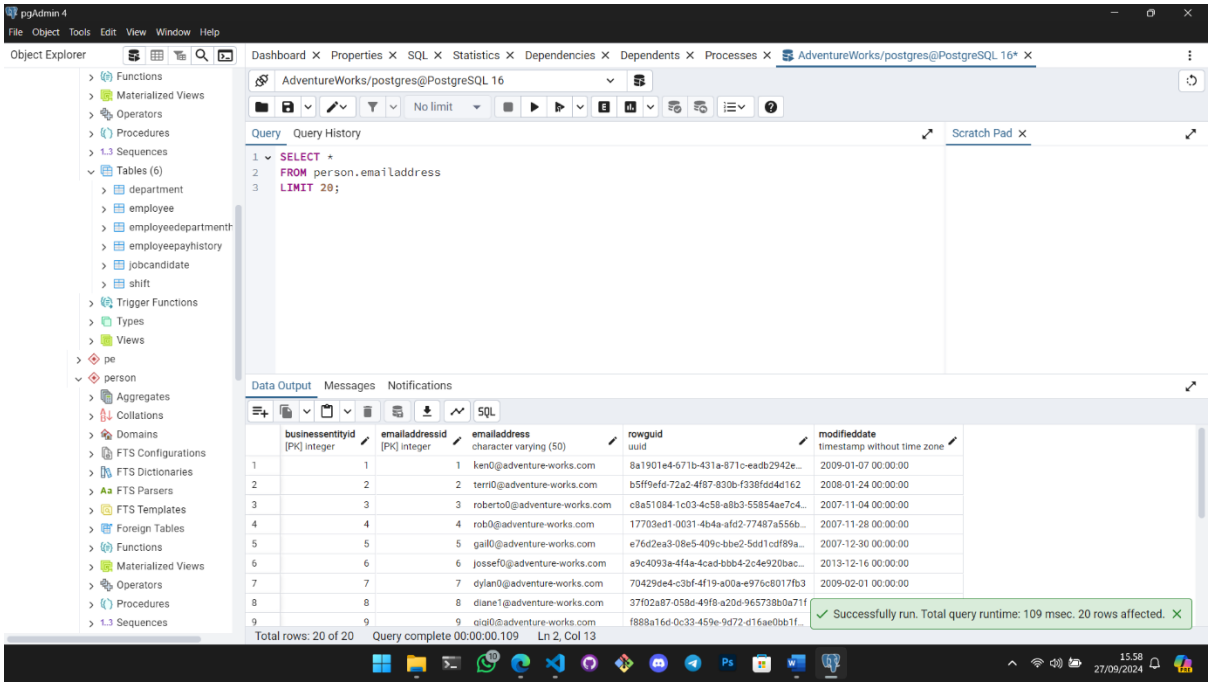
4. Find all vendor names that with a name that starts with letter G



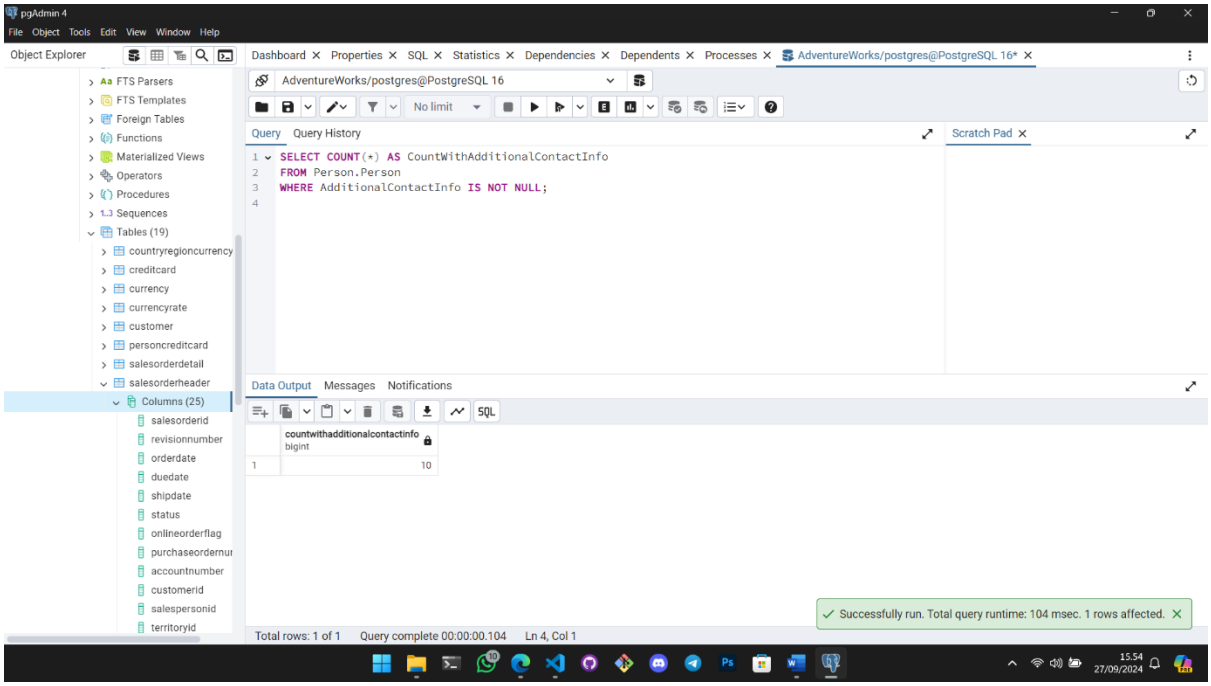
5. Search the person table for every firstname that has t as a second letter



6. Return the first 20 records from emailaddress table



7. How many person records have an additionalcontactinfo field that has a value

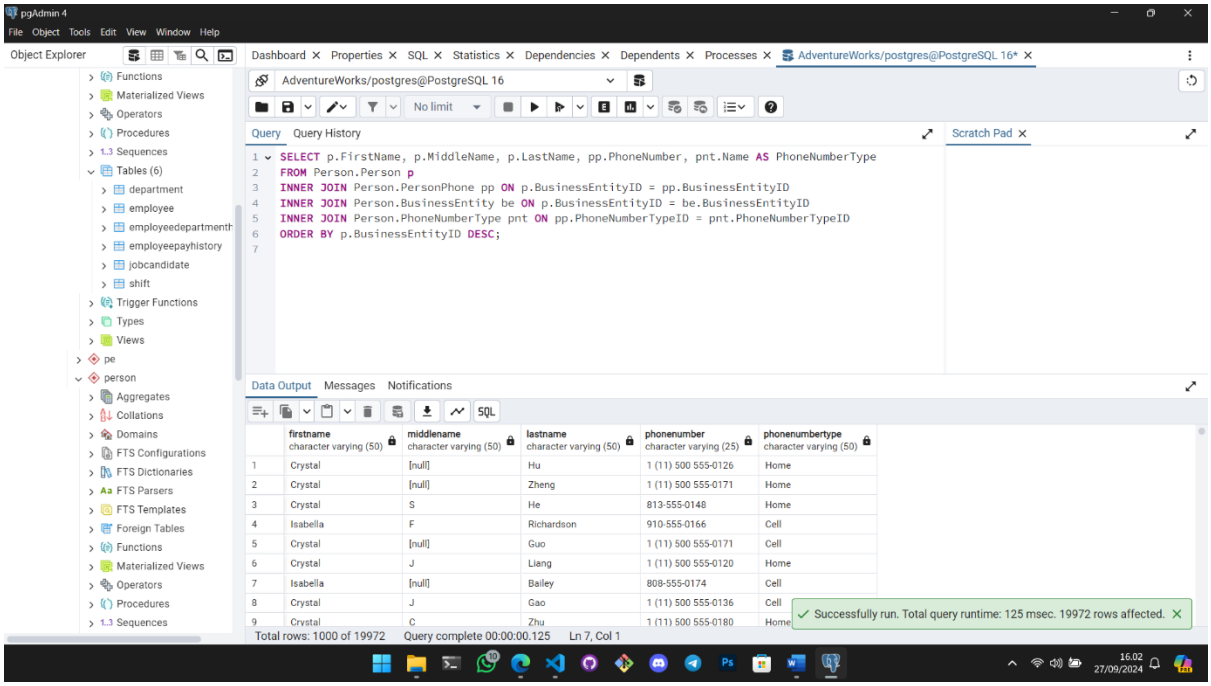


Assigment SQL 2.3

Open the AdventureWorks database and get a query window for that database. Make sure you are in the right database by looking at the text at the top of the query window.
Run the following queries.
There is more than one way to write most of these queries, so check to make sure the results are the same versus did you get the exact format I chose.

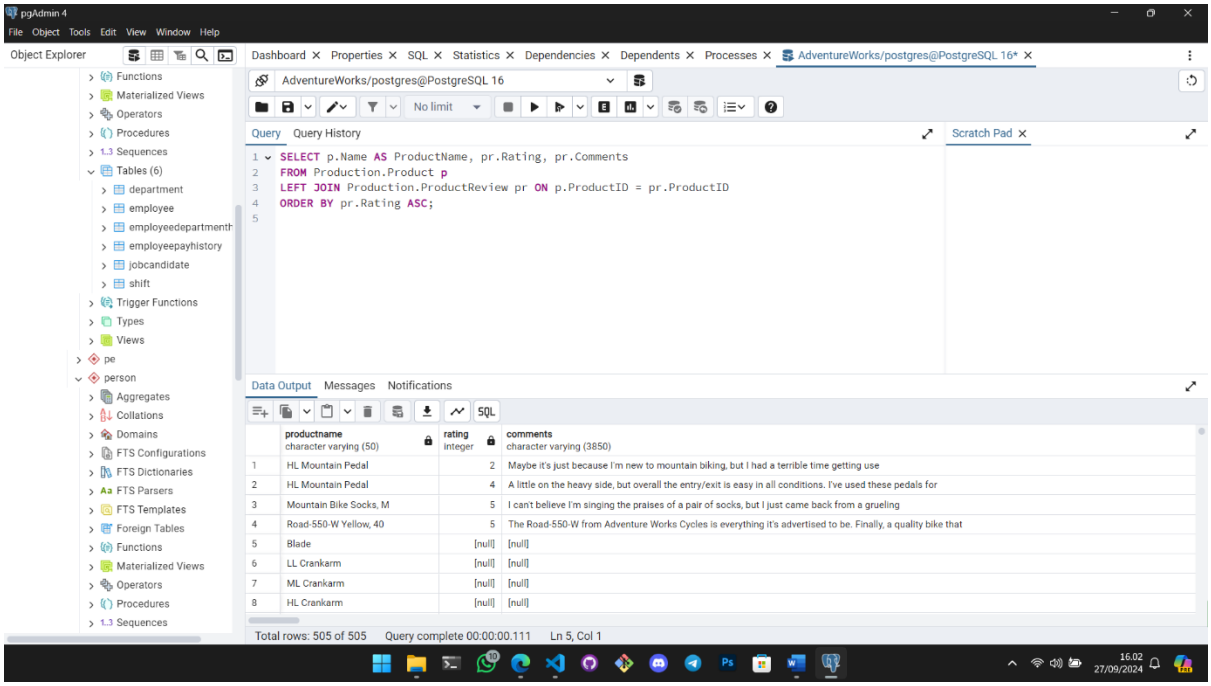
1. Join (with inner join) together person, personphone, businessentity and phonenumber type in the persons schema. Return first name, middle name, last name, phone number and the name of the phone number type (home, office, etc.) Order by business entity id descending.

```
SELECT p.FirstName, p.MiddleName, p.LastName, pp.PhoneNumber, pnt.Name AS PhoneNumberType
FROM Person.Person p
INNER JOIN Person.PersonPhone pp ON p.BusinessEntityID = pp.BusinessEntityID
INNER JOIN Person.BusinessEntity be ON p.BusinessEntityID = be.BusinessEntityID
INNER JOIN Person.PhoneNumberType pnt ON pp.PhoneNumberTypeID = pnt.PhoneNumberTypeID
ORDER BY p.BusinessEntityID DESC;
```



2. Join product and productreview in the schema table. Include every record from product and any reviews they have. Return the product name, review rating and comments. Order by rating in ascending order.

```
SELECT p.Name AS ProductName, pr.Rating, pr.Comments
FROM Production.Product p
LEFT JOIN Production.ProductReview pr ON p.ProductID = pr.ProductID
ORDER BY pr.Rating ASC;
```



3. Use a right join to combine workorder and product from production schema to bring back all products and any work orders they have. Include the product name and workorder orderqty and scrappedqty fields. Order by productid ascending.

```
SELECT p.Name AS ProductName, wo.OrderQty, wo.ScrappedQty
FROM Production.WorkOrder wo
RIGHT JOIN Production.Product p ON wo.ProductID = p.ProductID
ORDER BY p.ProductID ASC;
```