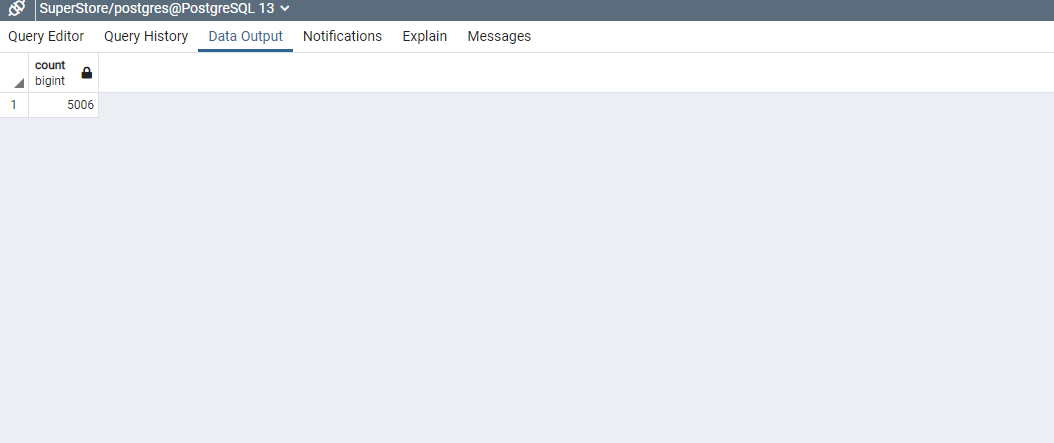
**Super Store Data Analysis Using PostgreSQL**

This project is about analyzing the Super Store dataset. I imported the dataset into PostgreSQL then I did the analysis below of the dataset.

1. --The query returns the number of records in the order table

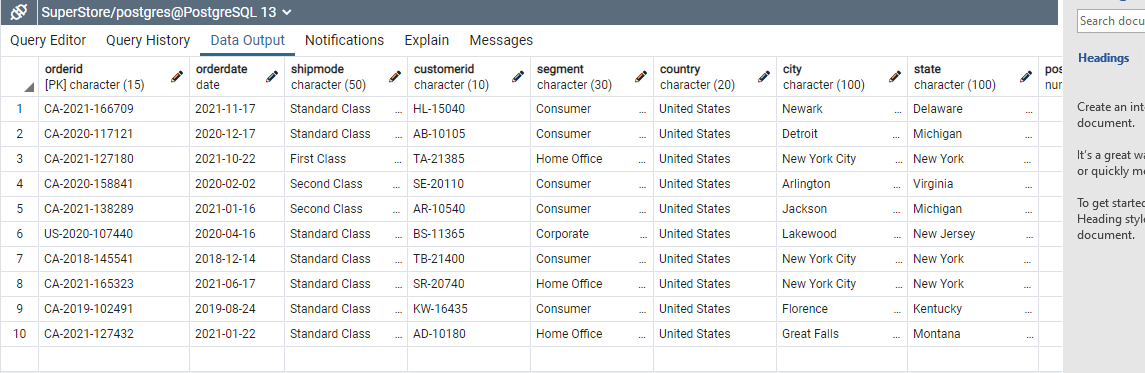
**SELECT COUNT(\*) FROM orders;**

****

1. --The query returns the top 10 records in the orders table

**SELECT \* FROM orders**

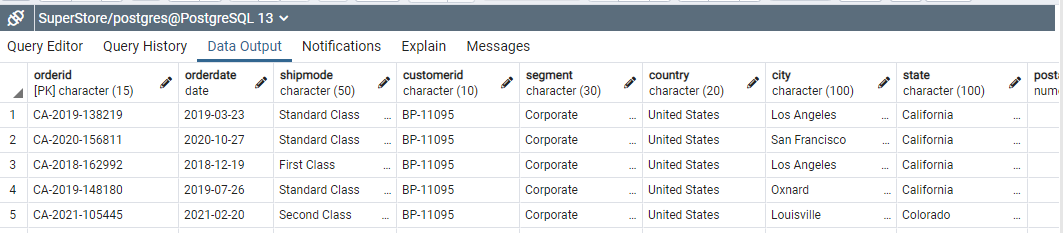
**LIMIT 10;**

****

1. --The query returns records for customer name 'Bart Pistole' for region 'west'

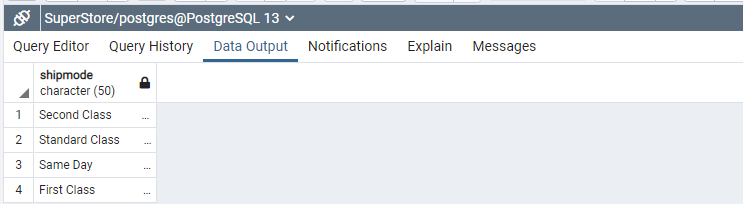
**SELECT \* from orders**

**WHERE customername = 'Bart Pistole' AND region = 'West';**



1. --The query returns the unique ship modes in orders table

**SELECT DISTINCT(orders.shipmode) FROM orders;**

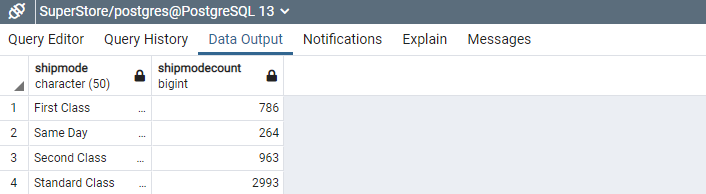


1. --The query below returns total number of ship modes in order table in ascending order

**SELECT orders.shipmode,COUNT(shipmode) AS shipmodecount FROM orders**

**GROUP BY orders.shipmode**

**ORDER BY orders.shipmode ASC;**

****

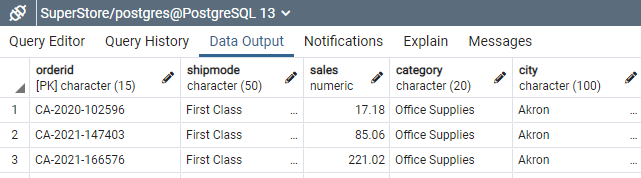
1. --The query returns records where ship mode is 'First Class' and city is 'Akron'

**SELECT orders.orderid,orders.shipmode,orders.sales,orders.category,orders.city FROM orders**

**WHERE orders.shipmode = 'First Class'AND orders.city = 'Akron'**

**GROUP BY orders.orderid,orders.category**

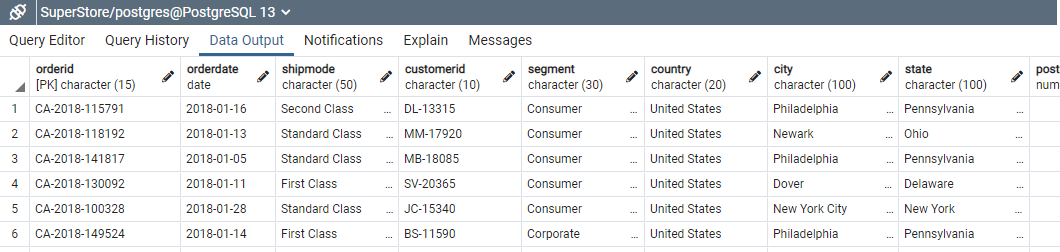
**ORDER BY orders.city ASC;**

****

1. --The query returns records for region 'East' for the year 2018

**SELECT \* FROM orders**

**WHERE orders.region ='East' AND orders.orderdate BETWEEN '2018-01-01' AND '2018-01-31'**;



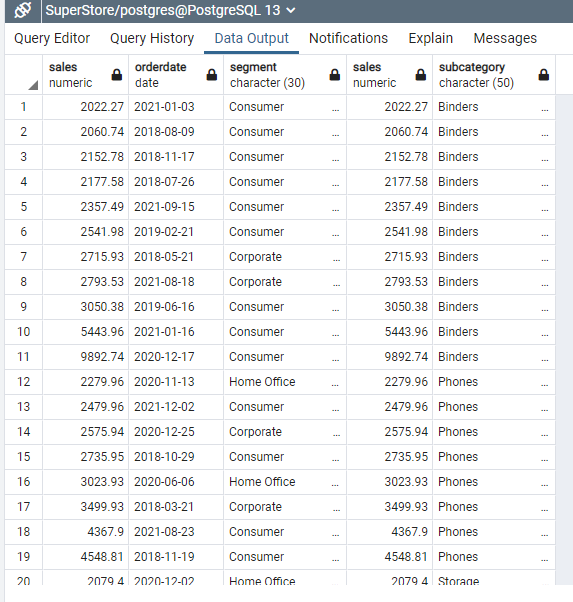
1. --The query returns sales greater than 2000 in different sub categories

**SELECT orders.sales,orders.orderdate,orders.segment,orders.sales,orders.subcategory FROM orders**

**WHERE orders.sales > '2000.00' AND orders.subcategory IN('Phones','Art','Binders','Storage','Labels')**

**GROUP BY orders.sales,orders.orderdate,orders.segment,orders.sales,orders.subcategory**

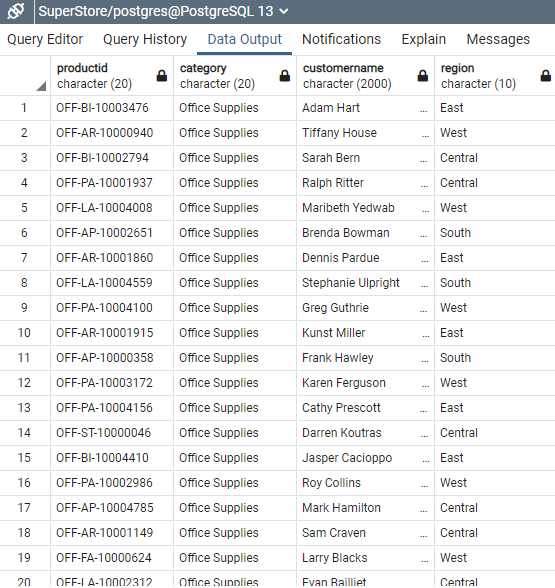
**ORDER BY orders.subcategory ASC;**

****

1. --The query will return productid,category,customername, region for productid that matches "OFF"

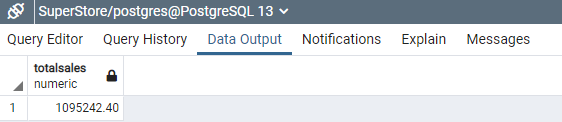
**SELECT DISTINCT(orders.productid),orders.category,orders.customername,orders.region from orders**

**WHERE orders.productid LIKE '%OFF%';**

****

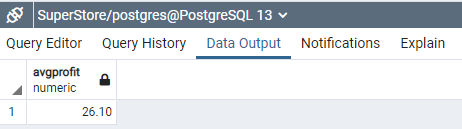
1. -- The query returns the total sales in the orders table

**SELECT SUM(orders.sales) AS TotalSales FROM orders;**

****

1. -- The query returns the average profit in the orders table

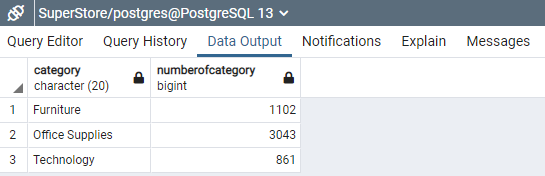
**SELECT ROUND(AVG(orders.profit),2) AS AVGProfit FROM orders;**

****

1. -- The query returns the count of each category in the orders table

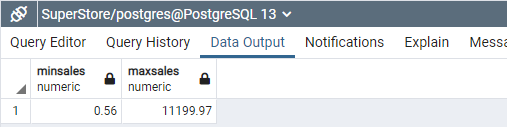
**SELECT orders.category,COUNT(orders.category) AS NumberOfCategory FROM orders**

**GROUP BY orders.category;**



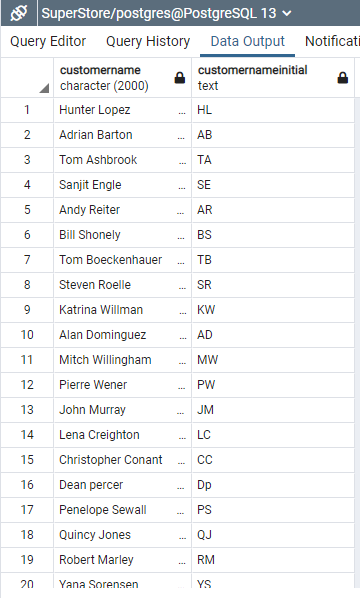
1. -- The query returns the min and max sales in the orders table

**SELECT MIN(orders.sales) AS MinSales, MAX(orders.sales) AS MaxSales FROM orders;**

****

1. -- The query returns the intials of customers from customer id

**SELECT orders.customername, SUBSTRING(orders.customerid,1,2) as CustomerNameInitial FROM orders;**



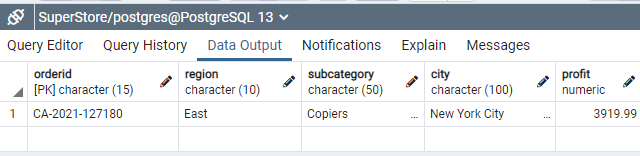
1. -- The query returns the subcategory which has a profit greater or equal to 3000 in New York city

**SELECT orders.orderid,orders.region,orders.subcategory,orders.city,orders.profit FROM orders**

**WHERE orders.city = 'New York City' AND orders.profit >= '3000.00'**

**GROUP BY orders.subcategory,orders.orderid**

**ORDER BY orders.subcategory ASC;**



1. -- The query returns productid with unit price greated than 50000 for a particular customer

**SELECT COUNT(orders.orderid) AS NumberOrders,orders.productid,orders.customername, SUM(orders.unitprice) AS TotalPrice FROM orders**

**GROUP BY orders.productid,orders.customername**

**HAVING SUM(orders.unitprice) >= 50000;**

