# Information

This document shows simple select queries which may be useful for sales analytics – there are 3 simple made up scenarios.

This is not a complex analysis. A lot of time spent on this analysis was finding the dataset, cleaning it, importing it and validating it.

This is to display that I know how to download and import data, clean data, and perform simple select statements for filtering the data. All these functions can be performed on Excel too.

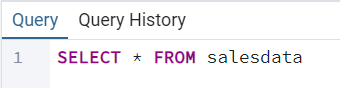
## Importing the dataset

I used an online tool to convert the dataset into a CREATE TABLE and INSERT INTO code. I know how to create and insert data into a table manually, but because this data set is not mine it was easier to convert it to SQL code.

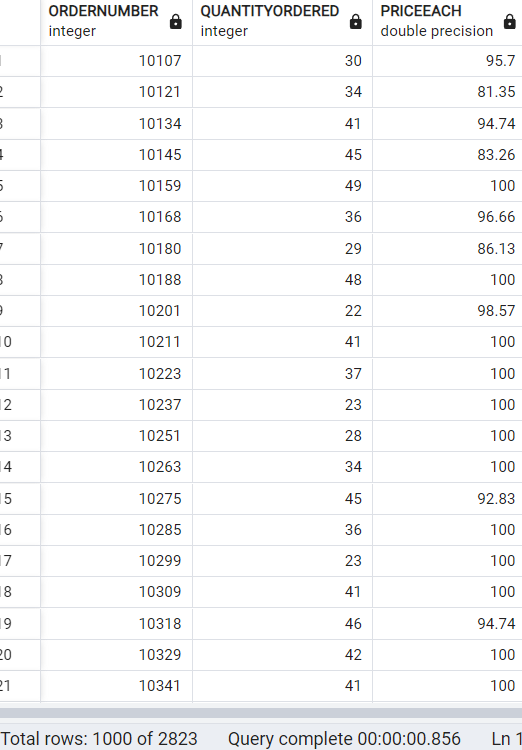
pgAdmin4 has a tool for importing straight from the Excel file (Excel as CSV) but this is not how I do it.

# Select Queries

* The select statement used below uses the \* to select all the data from the dataset. The dataset is named salesdata, which is a ‘table’.

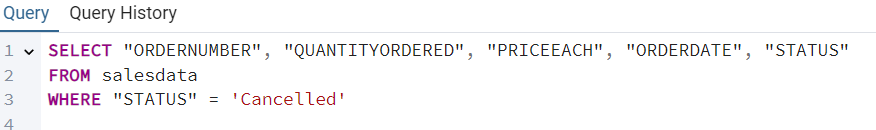


This is the output:



The entire thing is too big to fit into this Word document but notice the total rows: 1000 of 2823 at the bottom. The Excel sheet shows 2824 rows telling me that all the data in the query is in fact there. The extra 1 in the Excel sheet is counting the column header. 

### Scenario: I want to select which orders were cancelled



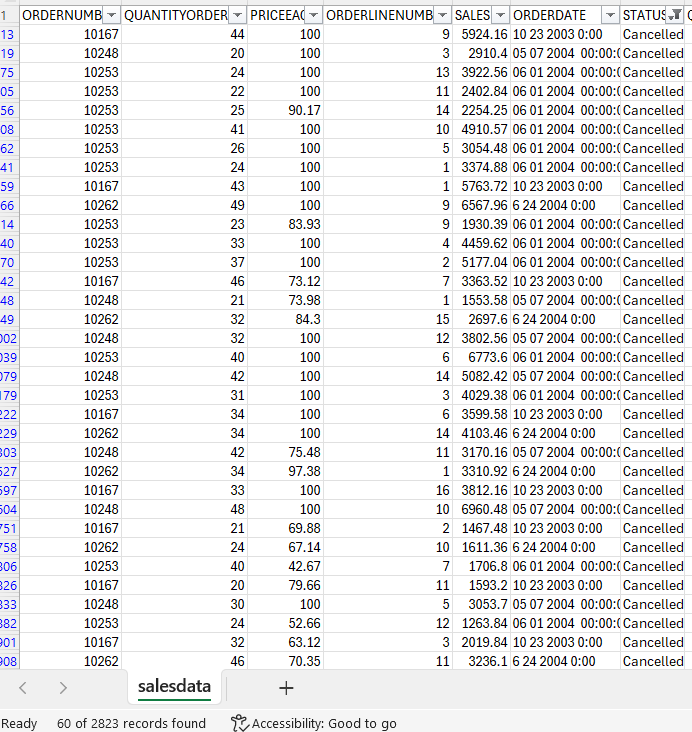
The above query tells the SQL server to select order number, quantity ordered, price each, order date and status from the dataset where the status is cancelled.

I selected the columns before status because it seems like relevant information.

A screenshot of a computer

Description automatically generated

This is a sample of the output – like before, there is relevant details at the bottom “Total rows: 60 of 60”, this shows that there are 60 cancelled orders. This query is similar to a simple filter on an Excel sheet shown below.



### Scenario: I want to see which customer ordered the most

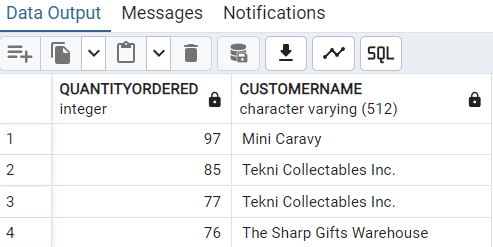
A screenshot of a computer screen

Description automatically generated

This function selects the quantity ordered and customer name from the sales data and orders it by quantity ordered descending.

The reason for the ORDER BY function is because by default it will be ordered ascending, but for readability it’s easier to find who ordered the most by checking top down.

The value from this kind of analysis could be seeing which customers are easiest to gain a long-term solid relationship with, conversely those with less orders may be possibly converted to long-term customers and may be worth investing into.



### Scenario: I want to see how many unique customers I have

A close up of a text

Description automatically generated



The output displays all the unique customers, showing that the company is working with 92 customers.

# End

There are many more possibilities, and I will continue working with this dataset – however, this is analysis #1 and I spent some time brushing up on the basics to begin further analysis and possibly bringing in more complex datasets.

Today was mostly spent searching for, importing, cleaning and filtering the dataset.