Hands-on analysis using BigQuery

## Introduction

You have been working at the headquarters for a national chain of grocery stores as a Junior Data Analyst for the last 6 months. Management has expressed concern about the amount of inventory kept in storage at some of their regional sites. Your immediate supervisor has asked you to do some analysis on inventory and sales data to make recommendations for changes to inventory.

Management has been collecting data about its inventory, products and sales. These datasets can be found here: [ [Inventory,](https://drive.google.com/u/0/uc?id=1FCo85-_jwlOkqucuiizlP3WyCEu1uXBH&export=download) [Products](https://drive.google.com/u/0/uc?id=1Qm1FFskt30Cc1I8XbptukBnT3T1ZHeJ-&export=download), [Sales](https://drive.google.com/u/0/uc?id=1hQ3_EqXPANEPhyY-VDKa9LTuB73E_DVT&export=download)]. You think this data can help you make the inventory recommendations requested by your supervisor.

### **What you'll do**

For this lab, we will use BigQuery, which is a fully-managed, serverless database management system (DBMS) that supports querying using ANSI SQL. Some open-source options in the data analyst's toolbox include MySQL Community Edition, MongoDB, and PostgreSQL. The visual design tool for each DBMS might look very different. However, the queries we will perform in this Qwiklab will be similar regardless of the DBMS used.

In addition to BigQuery, you'll also use Google Sheets. There are other options in the data analyst's toolbox, as well, such as Microsoft Excel and Libra Office 3. No matter which tool you use, the steps you'll follow are similar.

### **Goals for this lab**

#### You will use SQL commands to perform database actions like:

* Create a new SQL table using CREATE TABLE.
* Import a CSV file into a database table.
* Use the SELECT statement to view database records.
* Use the ORDER BY keyword to sort data in different ways.
* Use the WHERE clause to filter data in different ways.

#### You'll also use spreadsheets to perform actions like:

* Applying filters to explore data.
* Creating a Pivot table to summarize data.
* Using visualizations to present data analysis results.
* Using formulas to conduct basic data analysis.
* Applying formatting.

### **Start your lab by signing in to the Console**

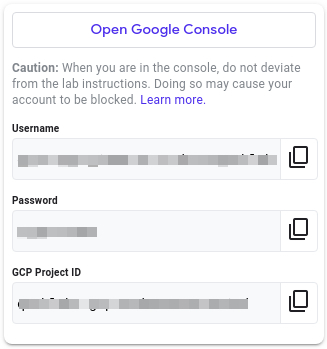
**Before you click the Start Lab button** read these instructions. Labs are timed and you cannot pause them. The timer, which starts when you click **Start Lab** , shows how long Google Cloud resources will be made available to you.

This Qwiklab hands-on lab lets you do the lab activities yourself in a real cloud environment, not in a simulation or demo environment. It does so by giving you new, temporary credentials that you use to sign in and access Google Cloud for the duration of the lab.

1. Click the **Start Lab** button.

Start Lab

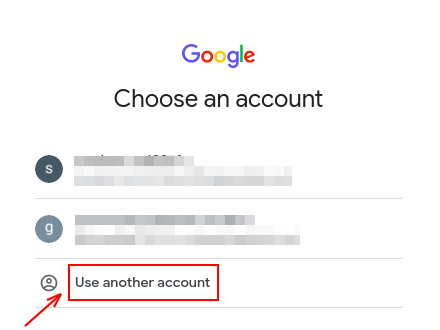
On the left is a panel populated with the temporary credentials that you'll need to use for this lab.



1. Copy the username, then click **Open Google Console**. The lab spins up resources, and then opens another tab that shows the **Choose an account** page.

**Tip:** Open the tabs in separate windows, side by side.

**Note:**Using a new Incognito window (Chrome) or another browser for the Qwiklabs session is recommended. Alternatively, you can log out of all other Google / Gmail accounts before beginning the labs.

1. On the **Choose an account** page, click **Use another account**. 
2. The **Sign in** page opens. Paste the username that you copied from the **Connection Details** panel. Then copy and paste the password.

**Important:** You must use the credentials from the **Connection Details** panel. Please do **not** use your Qwiklabs credentials. If you have your own GCP account, do **not** use it for this lab in order to avoid incurring charges.

1. Click through the subsequent pages:
2. Accept the terms and conditions.
3. Do **not** add recovery options or two-factor authentication, since this is a temporary account.
4. Do **not** sign up for free trials.

After a few moments, the GCP console opens in this tab.

### **What you need**

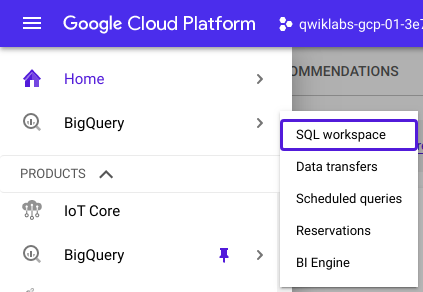
To complete this lab, you need:

* Access to a standard internet browser (Chrome browser recommended).
* Time to complete the lab.

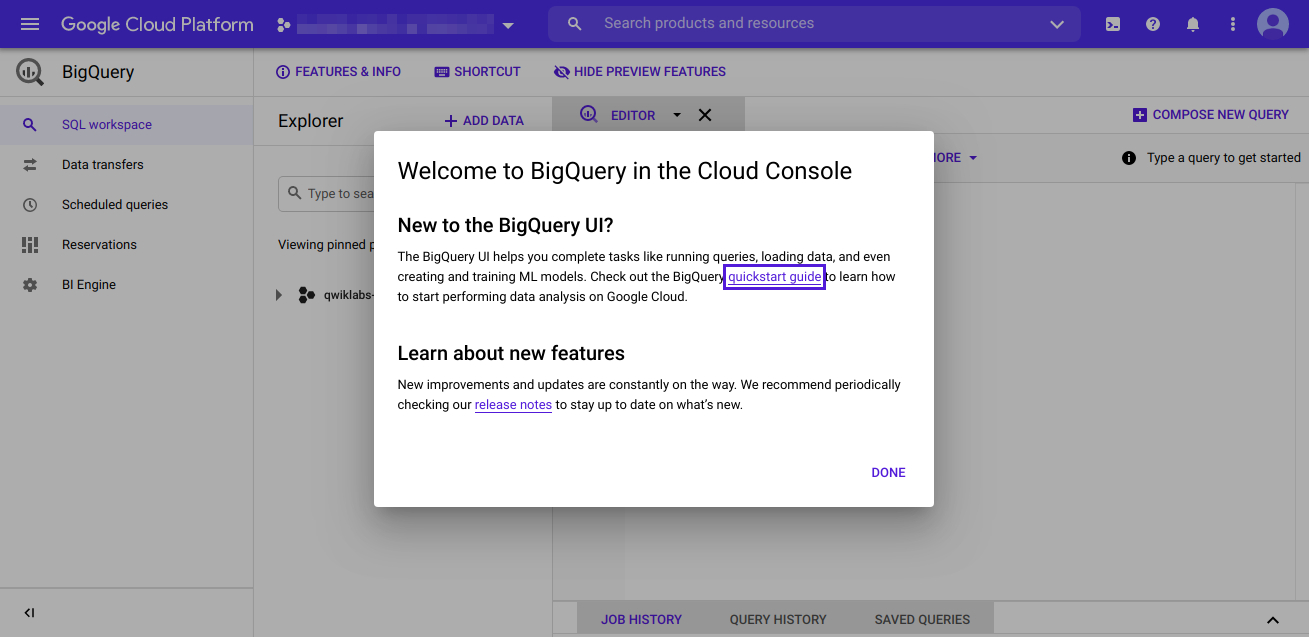
**Note:**If you already have your own personal Google Cloud account or project, do not use it for this lab.

### **Open BigQuery Console**

In the Google Cloud Console, select **Navigation menu** > **BigQuery**:

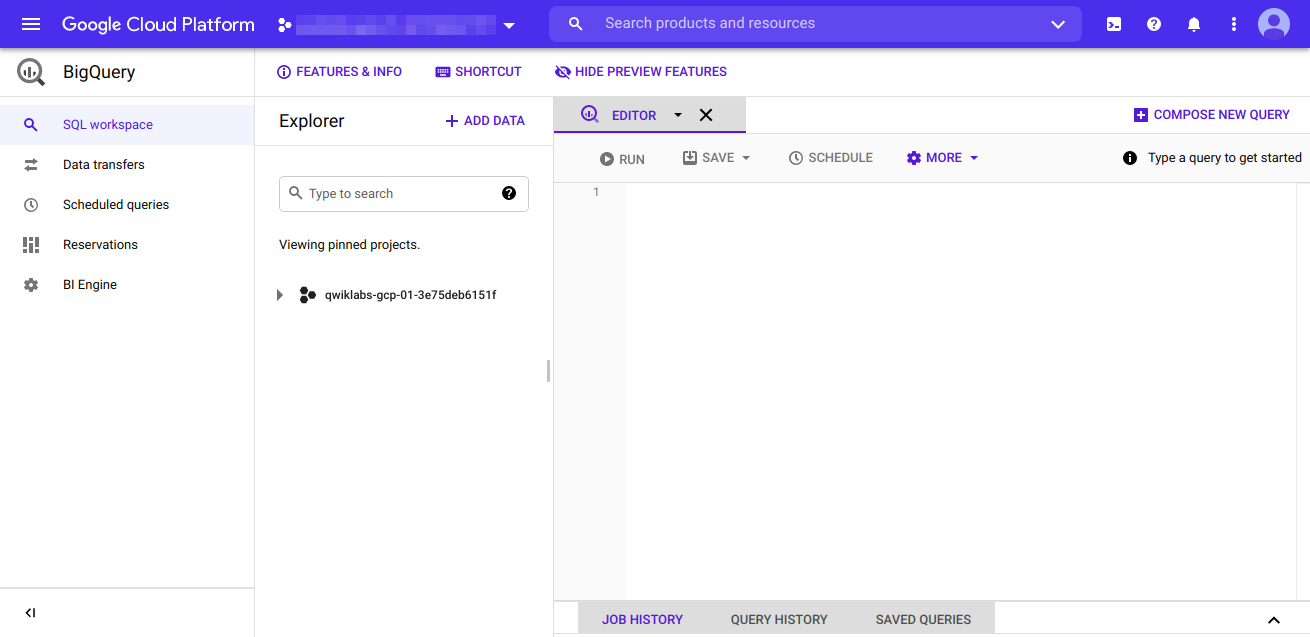


The **Welcome to BigQuery in the Cloud Console** message box opens. This message box provides a link to the quickstart guide and the release notes.



Click **Done**.

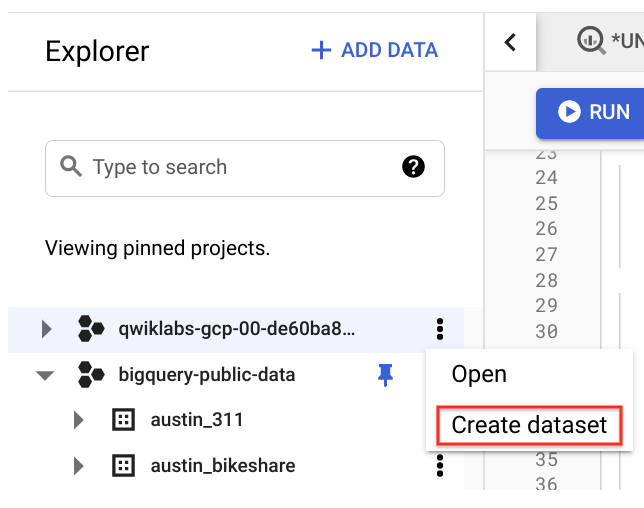
The BigQuery console opens.

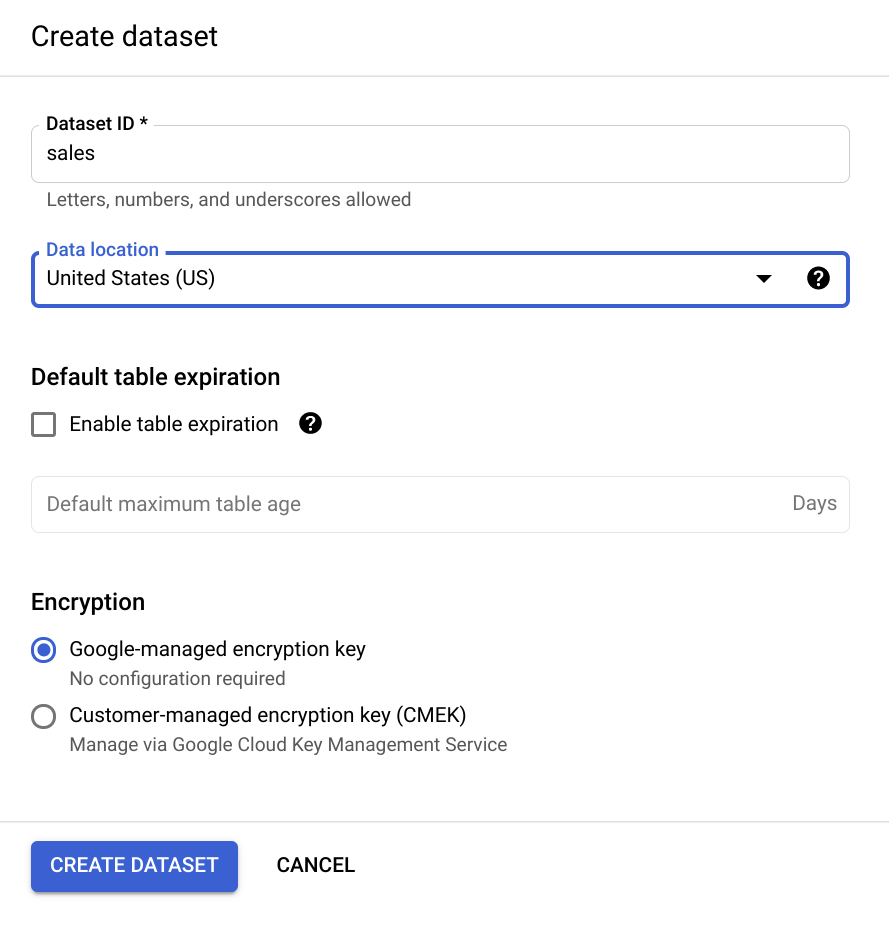


## Creating the database, creating the table, and importing the data into a DBMS

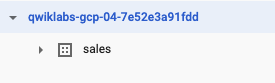
If you have not yet done so, download all the files to your desktop now: [ [Inventory,](https://drive.google.com/u/0/uc?id=1FCo85-_jwlOkqucuiizlP3WyCEu1uXBH&export=download) [Products](https://drive.google.com/u/0/uc?id=1Qm1FFskt30Cc1I8XbptukBnT3T1ZHeJ-&export=download), [Sales](https://drive.google.com/u/0/uc?id=1hQ3_EqXPANEPhyY-VDKa9LTuB73E_DVT&export=download)]. Looking at the sizes of the downloaded files, you can see that Sales.csv is 8MB. Although this is below the 20MB limit for Google Sheets, it is still quite large. It could be too slow to import, open and analyze in Google Sheets. Therefore, we should import it into BigQuery in order to inspect it.

To get started, you will need to create the sales database where you will store your table data. First select project then click on the View action icon (three dots) and select **Create Dataset**. Then on the popup menu, make the following entries and selections.



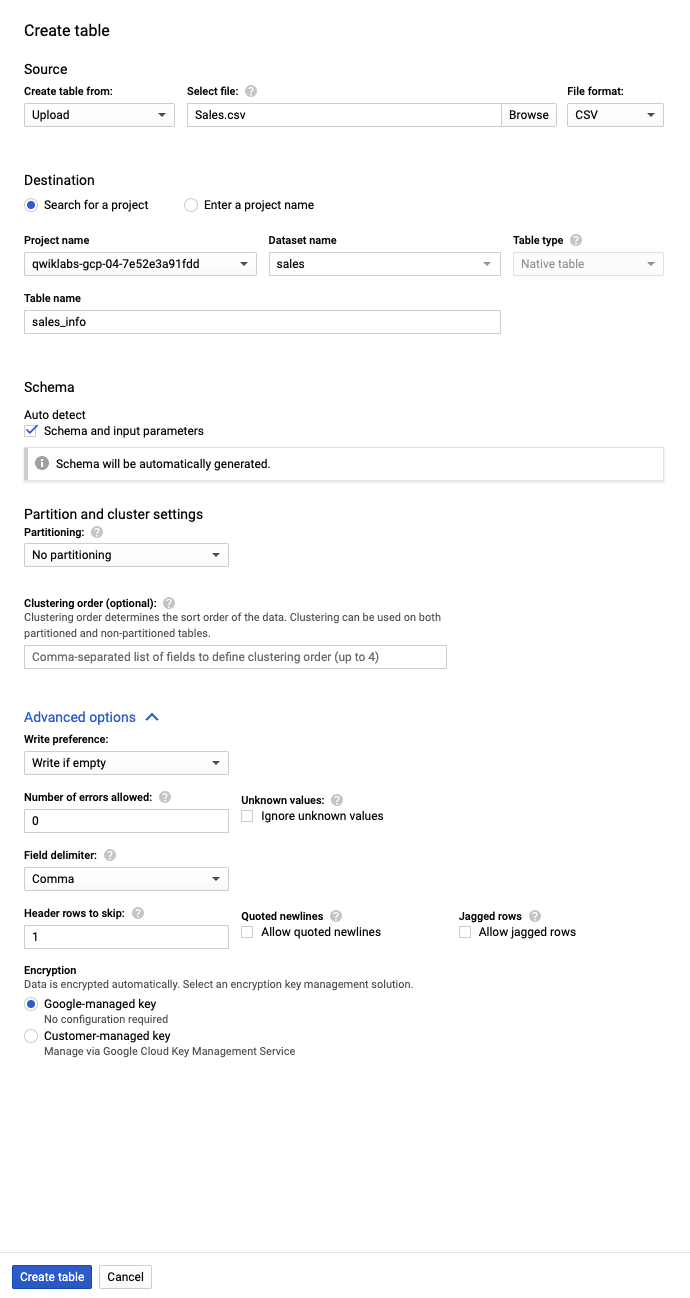


Then click **Create dataset**. You can verify that the sales database was created by selecting the project name on the left menu. The sales database should appear underneath the dropdown.



Next let's import the sales data into a new sales\_info table within the sales database. The process of importing a CSV file into a database table will vary depending on the DBMS. Using BigQuery, importing CSV data can be done in a few clicks using the web interface. Other database management systems may require you to run a query to do the import.

On the left menu pane, select the sales database name and then click on the View action icon (three dots) and select **Open**. Then on the menu that appears on the right, click **CREATE TABLE**. On the next popup menu, make the following entries and selections.



When you click the Browse button (shown above), you will be prompted to navigate to your downloaded file. If you didn't rename the file it should be Sales.csv. To make the advanced option selections (shown above), click the **Advanced options** dropdown.

Next click **Create table**.

Now the dataset has been imported into the sales\_info table within the sales database.

The SELECT statement is used to select data from a table. The data returned is stored in a result table, called the result-set. To verify that the import was successful, make the following query containing the SELECT statement:

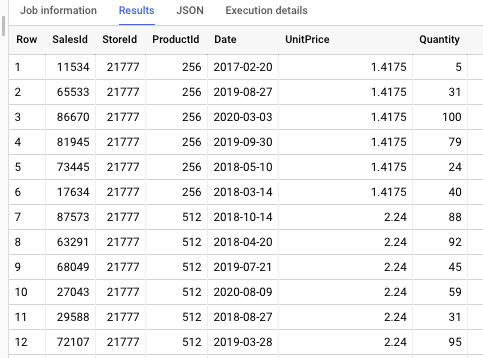
SELECT

\*

FROM

sales.sales\_info;

After you enter the query in the Query Editor, click on the run button to run your query. You should see something like this:



Notice that BigQuery requires you to specify the database name and the table name you are querying separated by a period (.) like this: sales.sales\_info. The status bar on the bottom right indicates that there are 200,000 rows in the sales\_info table.



## Inspecting the data using SQL and deciding on a forecasting method

When making predictions about inventory, two approaches are typically used – monthly forecasting and weekly forecasting. Monthly forecasting means that sales data is captured daily and grouped into months to produce a forecast. Similarly, weekly forecasting involves grouping daily sales into weeks to create a forecast. Well then, which is better?

You may think that for a given year 52 data points is better than 12, but this isn't always the case. In fact, monthly forecasts work best for most inventory products because they tend to generate lower forecast errors and, therefore, are more reliable. This is true mainly because of two reasons: the larger grouping of data used in monthly forecasting better absorbs changes in customer purchase timing and monthly timeframes handle seasonality better. For these reasons, we will decide on using a monthly forecasting method.

Now, let's inspect the data to see how many years of sales it includes. A good way to inspect the data in a column containing numeric values is to sort it. The ORDER BY keyword is used in SQL to sort the result-set in ascending or descending order. The ORDER BY keyword sorts the records in ascending order by default. To sort the records in descending order, use the DESC keyword.

The Date column contains numeric (or date) data, so let's inspect it by sorting in ascending order. Click on **COMPOSE NEW QUERY**.To do this, make the following query containing the ORDER BY keyword:

SELECT

\*

FROM

sales.sales\_info

ORDER BY

Date;

Scroll down to see how everything looks. It looks like we have daily sales data of products for a few different years, but simply scrolling is not a good way to see exactly how many years of data we have. If a dataset contains many rows of data, it may also be a good idea to inspect the minimum and maximum values of a column rather than scrolling down the returned values to locate them. In SQL, the MIN() function identifies the smallest value of the selected column, and the MAX() function identifies the largest value of the selected column.

To do this, click on **COMPOSE NEW QUERY** and then make the following query containing the MIN() and MAX() functions to inspect the minimum and maximum values of this column:

SELECT

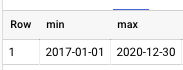
MIN(Date) AS min,

MAX(Date) AS max

FROM

sales.sales\_info;content\_copy

You should see the following:



In SQL, the AS command is used to rename a column or table with an alias. In the above query, the columns were renamed as min and max to make it easier to see which value in the result-set is the minimum and which is the maximum. These results indicate that the sales data ranges from 1/1/2017 and 12/30/2020.

Since we decided to use the monthly forecasting method, we will need to calculate the total quantity sold of each product grouped by month and year. In SQL, EXTRACT returns values corresponding to different date parts from a column of dates. To do this, click on **COMPOSE NEW QUERY** and then make the following query to return the total quantity sold of each ProductId grouped by the month and year it was sold.

SELECT

EXTRACT(YEAR

FROM

date) AS Year,

EXTRACT(MONTH

FROM

date) AS Month,

ProductId,

ROUND(MAX(UnitPrice),2) AS UnitPrice,

SUM(Quantity) AS TotalSold

FROM

sales.sales\_info

GROUP BY

Year,

Month,

ProductId

ORDER BY

Year,

Month,

ProductId;content\_copy

You should see the following:



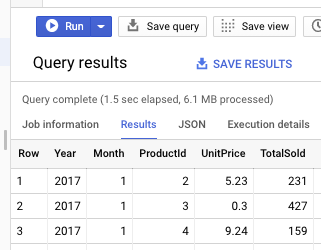
This query gives us the total number of products sold (TotalSold) for every unique product (ProductId) with its individual unit price (UnitPrice) for every month and year. In SQL, the GROUP BY statement groups rows that have the same values into summary rows. In the above query, we grouped the rows that have the same Month, Year and ProductID into the sum of their Quantity and the maximum of their UnitPrice columns.

The status bar on the bottom right indicates that there are 47,269 rows in this data subset.

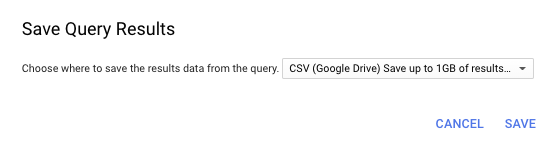


Since the data subset is less than 50,000 rows, a spreadsheet can easily handle any data presentation and further analysis we would like to perform. So, let's export the subset data for use in a spreadsheet.

To do this, click **SAVE RESULTS**.

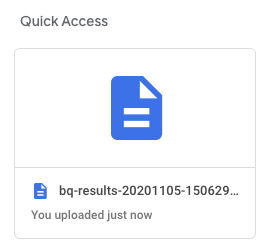


On the popup window, choose the **CSV(Google Drive)** option.

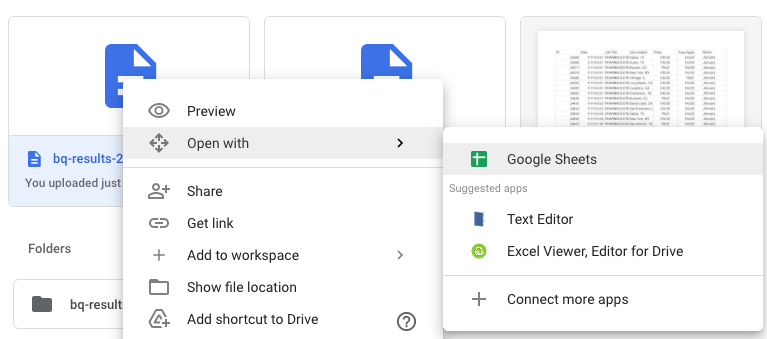


## Importing your data into a spreadsheet

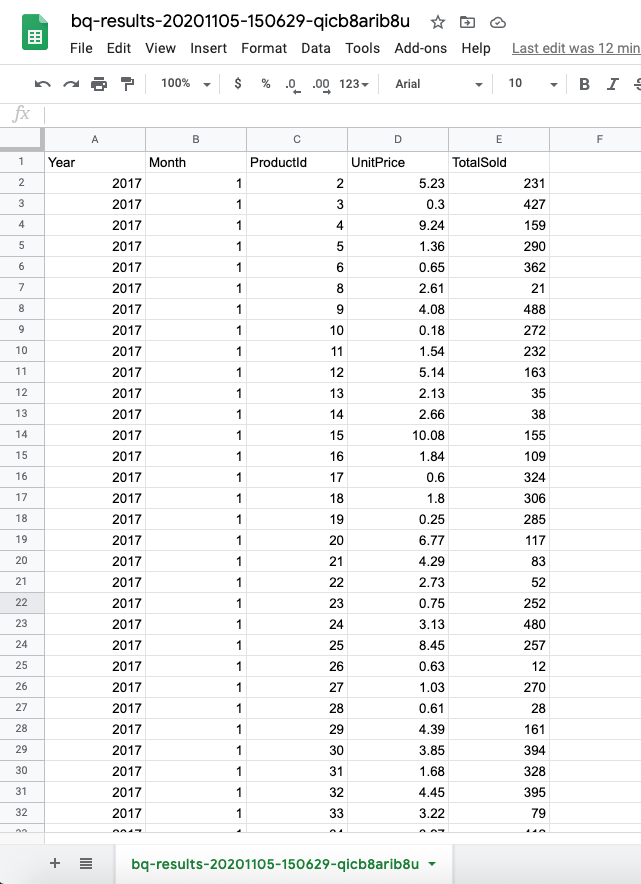
To access the exported query results and import it into Google Sheets, you must open Google Drive. To do this, open a new browser tab and in the address bar type the following URL **drive.google.com/drive/my-drive**. Under the **Quick Access** window, you should see the data file you just exported.



Right click on the file's icon and select **Open with -> Google Sheets**.



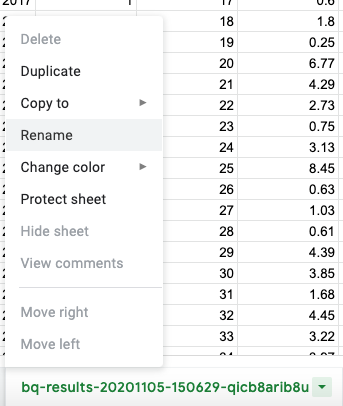
The data should open in a spreadsheet as shown in the following.



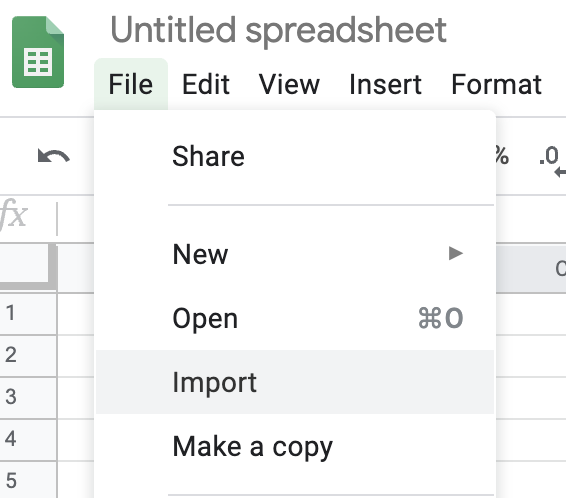
You should see 5 columns and 47,270 rows of data. Although 5 columns are easy to see, sometimes you'll work with spreadsheets that have hundreds or even thousands of columns. A quick way to check the number of columns in a spreadsheet is to select the row labeled 1. The status bar at the bottom will reflect the total count. To count the number of rows, select the column labeled A, then, in the status bar, select Count from the dropdown menu.

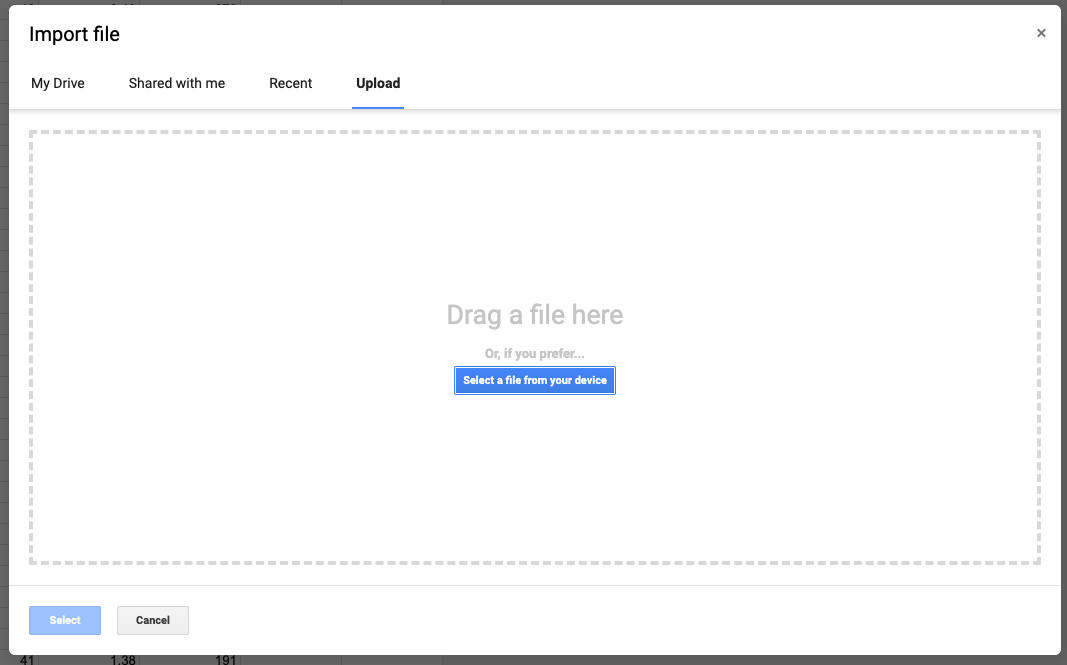
Rename this spreadsheet Sales by right-clicking the Sheet tab and choosing **Rename** from the dropdown menu.



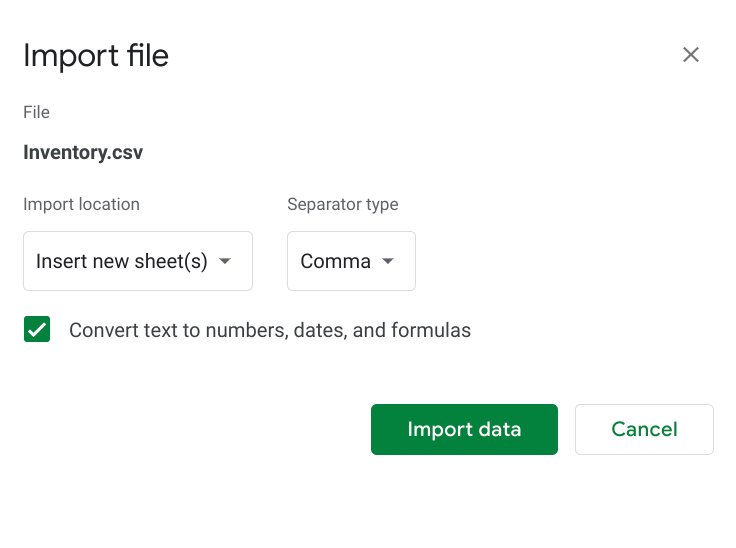
Now let's import the inventory data into the spreadsheet. To do this, in Google Sheets, you will go to the top menu bar and click **File → Import**.



Click on the **Upload** tab. Click **Select a file from your device** and then navigate to your downloaded file. If you did not rename the file name, the name of the dataset file should be Inventory.csv.



Next, on the popup window, make the following selections and then click on **Import data**:



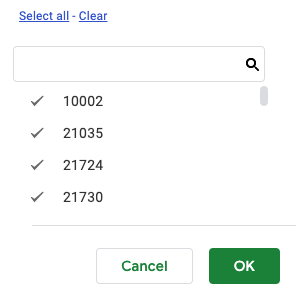
You can see if the import was a success by verifying that you have 6 columns and 1,001 rows of data. Repeat this process to import Products.csv into the spreadsheet. After the import, you should see 4 columns and 1,002 rows of data.

## Cleaning the data

To clean the data, you will need to know what data values and data types there are in each column of each spreadsheet. This is called inspecting the data. A good way to inspect the data in each column is by applying a filter. The filter will make it easier to find data that needs to be cleaned.

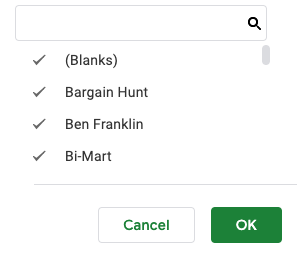
Let's apply a filter across the entire Inventory sheet. Click on any cell in the sheet. From the top menu bar select, **Data → Create a filter**.

You should see filters at the top range of every column containing data. Click the filter on the StoreId column (column B). At the bottom of the menu you will see a list of values.

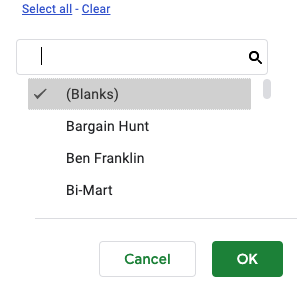


Scrolling down the list of values, you can see that this column does not contain any missing or erroneous data. There are no blanks and the values are all 5-digit numbers. This column looks clean.

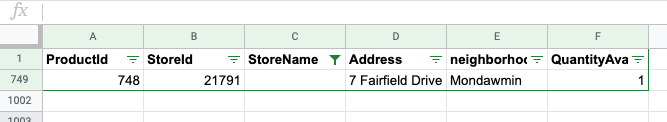
Now, click the filter on the StoreName column. You can see that this column has some blanks.



Filter this column on only the blanks by clicking on **Clear** from the filter menu and then clicking on (Blanks).

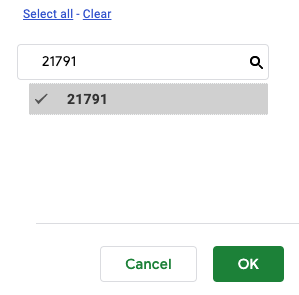


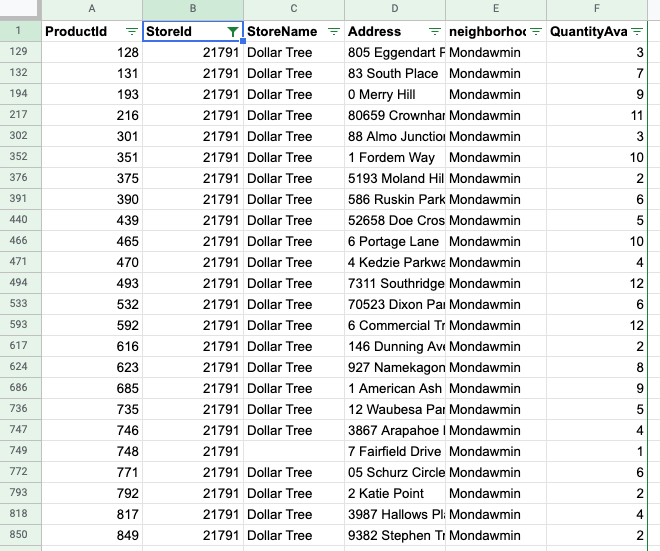
You should get the following filtered row.



One way to handle this missing data is by deleting the entire row. However, filling in the missing data whenever possible is always better than deleting it. So, let's check whether we can determine what the missing store name should be by filtering the store IDs on 21791.

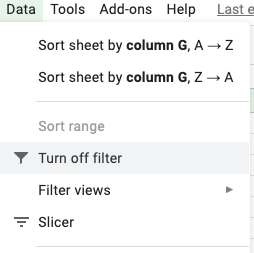
To do this, first remove the filter from the StoreId column by selecting its filter. Then from the filter menu click **Select all** and then **OK**. Next, click the filter on the StoreId column. Click **Clear**, then in the text box type 21791 and then click on 21791. Next click **OK**.





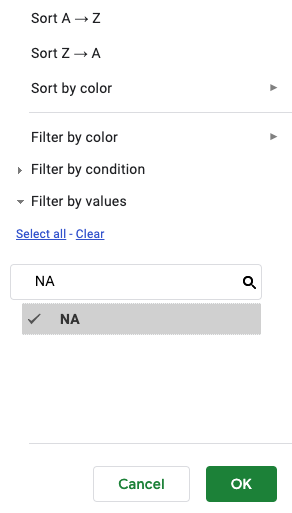
From this filter we can determine that the missing store name should be Dollar Tree since all store names with store ID 21791 is Dollar Tree. Scroll down to the cell with the missing value and type in Dollar Tree.

Continue inspecting the rest of the columns in this sheet. When you are done, turn off the filters in this spreadsheet. To do this, from the top menu bar select, **Data → Turn off filter.**

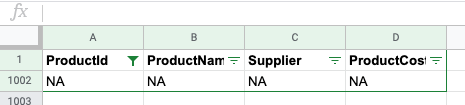


Let's inspect the Products spreadsheet by applying filters across all the columns. Next, click the filter on the ProductId column. Scrolling down to the end of the list of values, you can see that this column contains a value of NA. This is probably an error since all values in this column should be numbers.

Filter this column on only the value NA.



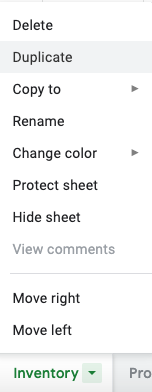
You should get the following filtered row.



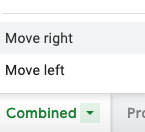
Let’s handle this error by deleting this row. Once you have applied the fix, continue inspecting the rest of the columns in this sheet. Turn off the filters when you are done.

## Combining the data

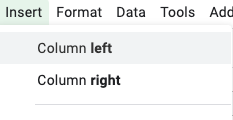
Now that the data has been cleaned, let's combine it into one summary sheet. Let's start by duplicating the Inventory sheet. To do this, click the down arrow on the Inventory tab and then select **Duplicate** from the dropdown menu.



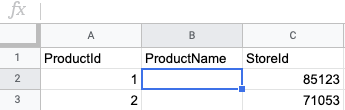
This should create a copy of the data from the Inventory sheet onto a new sheet named Copy of Inventory. Rename this sheet Combined. Then move it to the end by clicking the down arrow on the Inventory tab and then selecting **Move right**.



On the Combined sheet, insert a column to the left of StoreId. To do this, select the StoreId column by clicking its column letter. Then from the top menu bar, click **Insert → Column left**.



Give this column the header name ProductName by typing this into cell B1.



For each product ID in column A, fill in its corresponding product name in column B. To do this type the following formula in cell B2 and hit ENTER.

=VLOOKUP(A2,Products!A:B,2,FALSE)

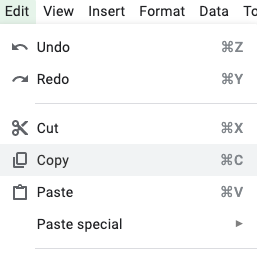
Syntax for VLOOKUP is:

=VLOOKUP (What you want to look up, where you want to look for it, the column number in the range containing the value to return, return an Approximate or Exact match – indicated as 1/TRUE, or 0/FALSE).

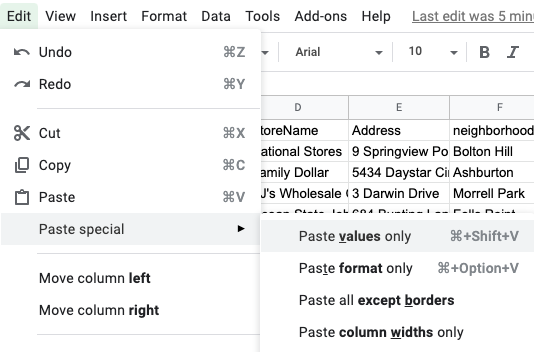
Copy this formula all the way down by clicking on cell B2 and double clicking the fill handle. Now all product names should be filled in for each product ID.

It's good practice to remove the formulas from the data just in case something gets deleted from the sheet this formula depends on. It is also a good idea to remove formulas from data you intend to create pivot tables from. Formulas in the data can create unexpected results in your pivot tables.

To remove the formulas, copy and paste the data in this column as values. To do this, select the column letter B. Then from the top menu bar, select **Edit → Copy**.



Then from the top menu bar, select **Edit → Paste special → Paste values only**.



Now all the data in the ProductName column are values, not formulas.

Insert 1 more column to the left of StoreId on the Combined sheet. Give this column the header Supplier in cell C1. For each product ID in column A, fill in its corresponding supplier in column C using the VLOOKUP function in cell C2 and then copying this formula all the way down.

Next give column I of the Combined sheet the column header ProductCost in cell I1. For each product ID in column A, using VLOOKUP fill in its corresponding product cost in cell I2 and then copy this formula all the way down. Be sure to remove the formulas from your new columns as described previously.

Your spreadsheet should now look like the following.



Next give column J of the Combined sheet the column header UnitPrice in cell J1. For each product ID in column A, fill in its corresponding UnitPrice. To do this type the following formula in cell J2 and hit ENTER.

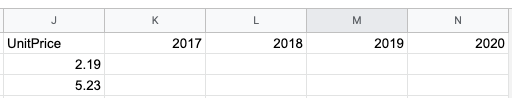
=VLOOKUP(A2,Sales!C:D,2,FALSE)

Copy this formula all the way and then remove the formulas.

Your spreadsheet should now look like the following.



On the Combined sheet, insert 4 columns to the right of UnitPrice with the column headers: 2017, 2018, 2019, and 2020.



In these columns, we will total the number of each product sold for each year in the data. To do this, we will need to use the SUMIFS function.

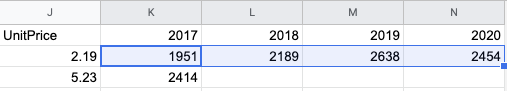
The SUMIFS function sums cells in a range using a supplied criterion. Unlike the SUMIF function, SUMIFS can apply more than one set of criteria, with more than one range. The first range is the range to be summed. The criteria are supplied in pairs (range/criteria) and only the first pair is required.

In cell K2, type the following formula and then hit ENTER.

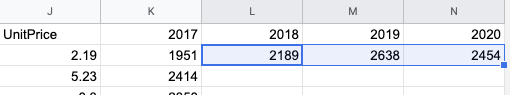
=SUMIFS (Sales!$E:$E,Sales!$A:$A,K$1,Sales!$C:$C,$A2)

Here the range to be summed is the TotalSold column in the Sales sheet and the criteria is to match on year 2017 and the product ID for that row. Click on cell K2 and then double click the fill handle to copy that formula all the way down.

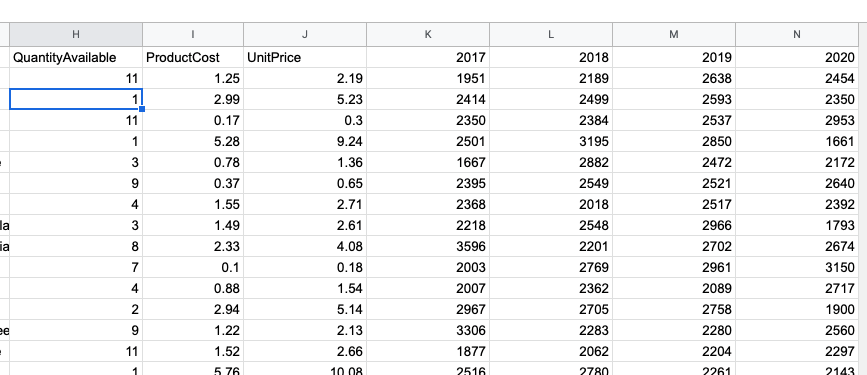
Since the anchors in the formula have been set appropriately, we can fill in the results for columns L through N by selecting cell K2 and dragging the fill handle over to cell N2.



Next, copy the formula all the way down by selecting cells L2 through N2 and then double clicking the fill handle.



The total sold per year should be filled in for each product. Be sure to remove the formulas from columns K through N.



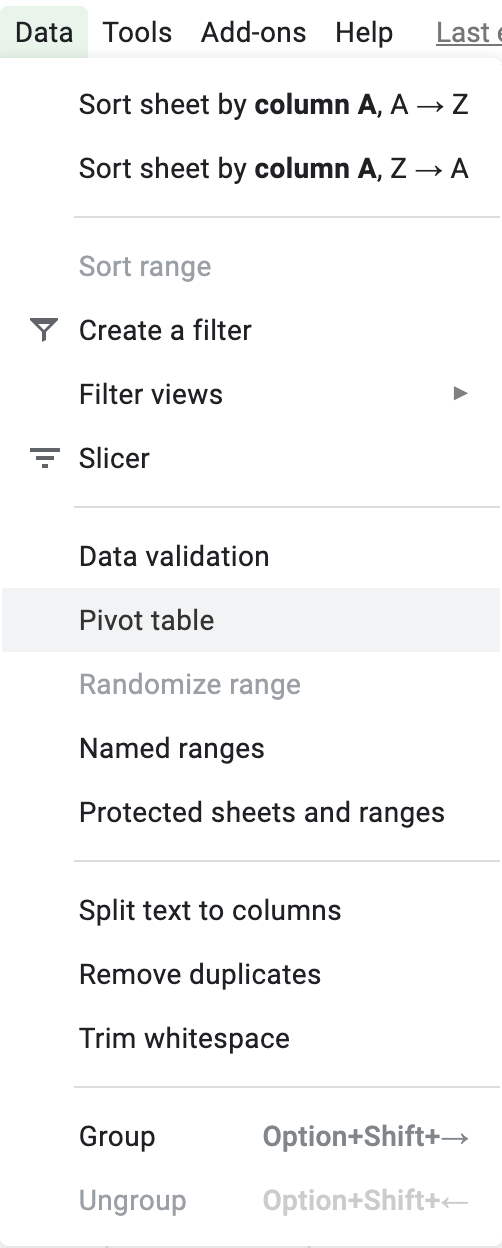
## Summarizing the data using pivot tables

Now that the data has been combined, let's begin summarizing it. First, insert a new sheet. To do this, from the top menu bar, click **Insert → New sheet**. Then, rename the new sheet Summary.

Now it's time to include important data summaries on the Summary sheet. Recall that data summaries simplify the data and make it easier for others to understand. Helpful data summaries should answer questions such as:

* How much of each product do we currently have in inventory?
* How much of each product do we sell per year on average?
* How much of each product do we sell per month on average?

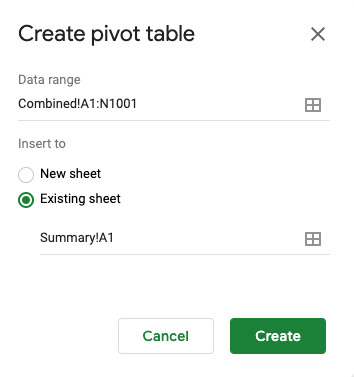
Now, insert a pivot table for the combined data. On the Combined sheet, click any cell that contains data. Then, from the top menu bar, select **Data → Pivot table**.



In the **Insert to** section of the popup window, select the **Existing sheet** option and click the **Select data range button**. 

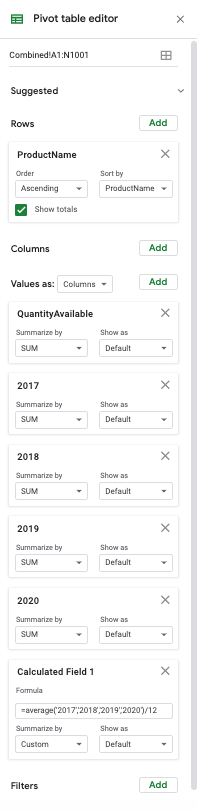
Then, in the **What location?** popup window, click in the input box. Click on cell A1 of the Summary sheet.

Click **OK**, and you'll be taken back to the previous popup window.



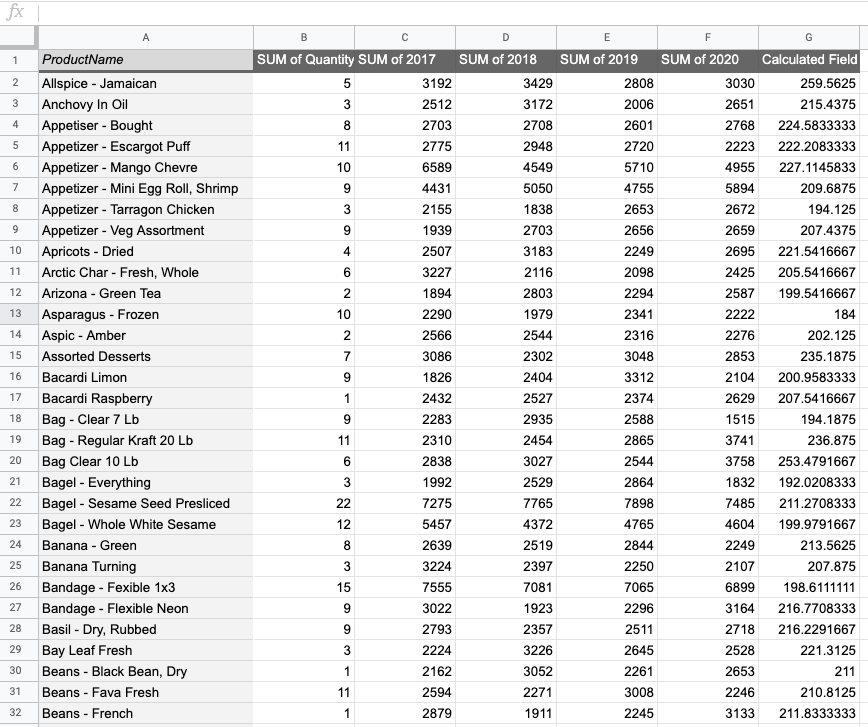
Now, click **Create**. This will create the pivot table framework for the Summary sheet.

In the pivot table editor on the right, click on add Rows and add Values, and make the following entries and selections.



formula =average('2017','2018','2019','2020')/12

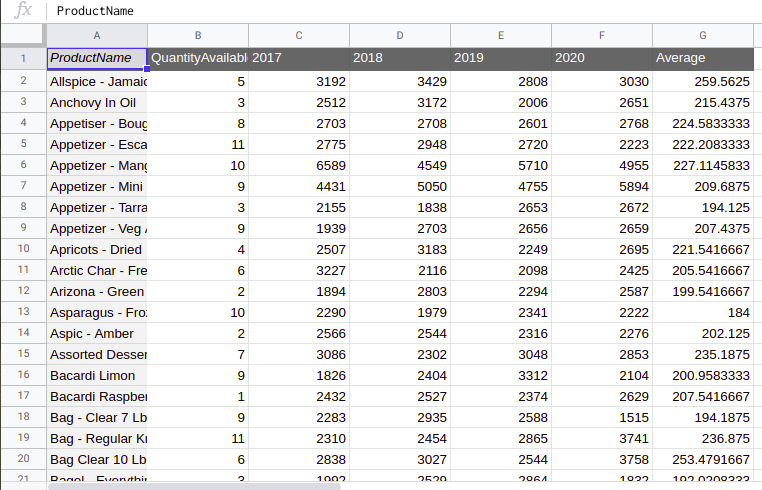
Your pivot table should begin in cell A1 and look like this:



Now change the following column names:

* SUM of QuantityAvailable to QuantityAvailable
* SUM of 2017 to 2017
* SUM of 2018 to 2018
* SUM of 2019 to 2019
* SUM of 2020 to 2020
* Calculated Field 1 to Average

Now, your pivot table should begin in cell A1 and look like this:



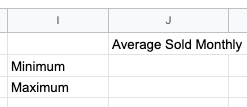
From this pivot table, you can see how much of each product the grocery store chain currently has in inventory. For example, you see that the chain currently has 5 of Allspice – Jamaican in inventory. You can also see how much of each product the chain sells per year on average. For instance, the chain sold a total of 2,948 in 2018 and 2,720 in 2019 of Appetizer – Escargot Puff. In addition, you can see that the chain sold an average of 221.54 per month of Apricots – Dried.

## Summarizing the data using custom formulas

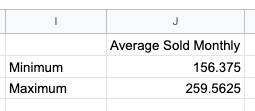
Another useful summary might answer this question:

* Which products do we sell the most and the least on average per month?

Create a custom table to find the answer. In cell I2 of the Summary sheet, type Minimum, and in cell I3, type Maximum. In cell J1, type Average Sold Monthly, and press Enter.



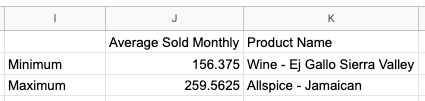
1. To calculate the product that sold the least, in cell J2 type the function =MIN(G2:G827).
2. To calculate the product that sold the most, in cell I3 type the function =MAX(G2:G827).



Next, let's find the products associated with these minimum and maximum values. In cell K1 type Product Name. Then in cell K2 type the following formula.

=VLOOKUP (J2,{G$2:G$827,A$2:A$827},2,0)

Hit ENTER. Then use the fil handle to copy this formula down to cell K3. Your custom table should look like this.

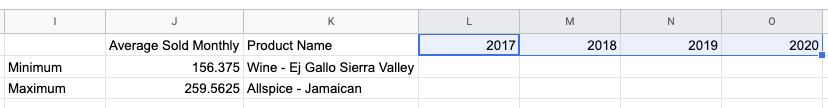


From this custom table, you can see that the grocery store chain sold the least on average per month of Wine – Ej Gallo Sierra Valley and the most on average per month of Allspice – Jamaican.

## Creating charts and graphs

Now let's turn the results of your custom table into a graph. It might be useful to see how the total sold changed over time for the minimum and maximum product sold. Recall that line charts are a good way to show trends.

To do this, first find the total sold per month of these 2 products from the pivot table. In cell L1, type 2017. Then in cell M1, type 2018. Next, select cells L1 through M1 and drag the fill handle to cell O1 to fill in the rest of the years.

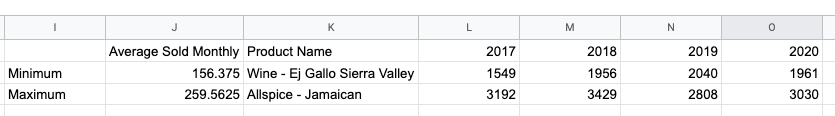


In cell L2, type the following formula:

=VLOOKUP($K2,$A$2:$F$827,3,FALSE)

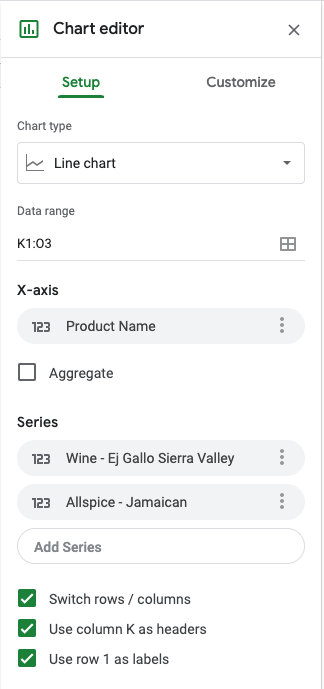
Hit ENTER. Then copy the formula down to cell L3. Complete the rest of the table by copying and editing this formula in cells M2 through O3.

Your table should look like this:

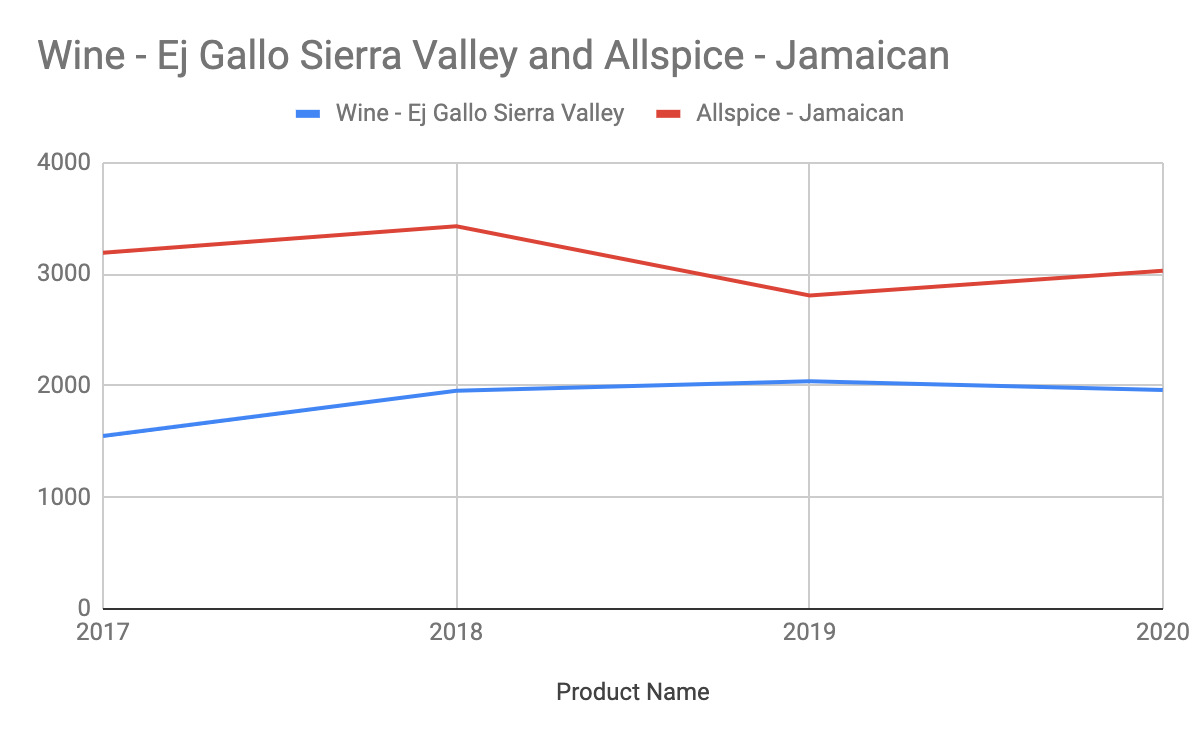


Now you are ready to insert your line chart. To do this, select cells K1 through O3. Then, from the top menu bar, select **Insert → Chart**.

If a line chart is not automatically inserted, change the graph type to a line chart. To do this, on the Chart editor, make the following selections:



The line chart created should look like this:



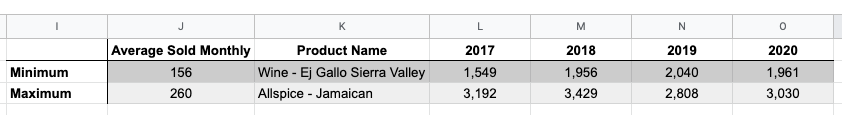
Change the title of the horizontal axis by double clicking Product Name and changing the text to Year. Then double click the graph title to give it a more descriptive name. From the chart you can see that neither Wine – Ej Gallo Sierra Valley or Allspice – Jamaican show a clear increasing or decreasing trend.

## Applying formatting

Your analysis is almost complete. All that's left is to apply formatting. Formatting helps draw the attention of the stakeholders you are presenting this data to.

Apply formatting to the custom summary table and pivot table. Try many different formatting options until you find the ones that will make this table look great. If you try something and don't like it, undo it. In addition, don't forget to give your pivot table columns more descriptive names.

Here's an example of what your custom summary table might look like:



## Congratulations!

You've successfully cleaned and analyzed this dataset using SQL and spreadsheets to ensure making good decisions regarding the grocery store chain's inventory purchases. Mastering the skills presented in this Qwiklab will help you in your data analytics career.

You can now close this window, manually end the lab.