Getting started with Tableau

Open Tableau.

# The Tableau interface

The Tableau interface has 10 parts:

1. Data sources
2. Dimensions and Measures. **Dimensions** are categories, usually text fields, but they can also be numeric fields that are meant to classify, not to be summed (e.g. ZIP code, job classification, etc.). They are **BLUE**. **Measures** are numbers that can have functions applied to them: SUM, AVERAGE, MIN, MAX, etc. They are **GREEN**.
3. The *Show Me* "card".
4. The *Columns* and *Rows* "shelves". Tableau has a **column "shelf"** and a **row "shelf".** These are similar to the row and column areas in an Excel pivot table.
5. The *Marks* "card". Controls color, size, label text, tooltip text, and shape.
6. The *Filters* shelf.
7. The *Pages* shelf: filters the visualization by stepping or animating based on a particular field.
8. The view area (the drawing canvas).
9. Sheets and Dashboard tabs.
10. The session tabs: connect to data, show all tabs in a worksheet, or see all workbooks for a user.

# Example 1: Summing

Under *Connect*, under *To a File*, click on *Microsoft Excel*.

Find *NYCTrash.xlsx* and open it. This is the garbage collection data for the city of New York for September, 2011. Examine the data. Each borough is divided into *Community Districts*.

We want to compare the collection numbers for each of the 5 boroughs.

Drag the *Borough* field to the Rows shelf. We get a list of the 5 boroughs.

Drag the *Refuse Tons Collected* to the Columns shelf. We get a bar chart, with one bar per borough.

There are other ways to represent this data:

Go to *Show Me* and pause the mouse over the first option, *Text Table* (the name of the visualization appears at the bottom of the *Show Me* window).

Try the following:

* Heat map
* Highlight tables
* Pie Chart (you will have to skip over two that are grayed out). Note that the legend for the pie chart is behind the *Show Me* window.
* Horizontal bars. NOTE: to get a column chart (vertical bars), you must move *Borough* to the Columns shelf and *SUM(Refuse Tons…)* to the Rows shelf.
* Stacked bars
* Tree map
* Circle view
* Box-and-whisker
* Packed bubbles

Which view makes the most sense? Probably Bar or Column chart.

Note that the data is sorted in alphabetical order based on the name of the borough.

Let's sort it by the length of the bar. Click on the *Sort Descending* button on the toolbar:



Our chart is now sorted:

A screenshot of a cell phone

Description automatically generated

To learn more about how Tableau is sorting its data, click on the right side of the *Borough* pill:



From the menu, click on *Sort…*.

A screenshot of a cell phone

Description automatically generated

The following will appear:

A screenshot of text

Description automatically generated

Swap the labels and the values by clicking on the *Swap* button on the toolbar:

A close up of a logo

Description automatically generated

## Grid Lines

Let's add grid lines. To add grid lines, right-click on the chart background, and choose *Format*. The *Format* box will appear on the left.

Click on the *Lines* button:

A screenshot of a cell phone

Description automatically generated

Click on the *Rows* tab. Click on the down-arrow button on the *Grid Lines* box and choose a solid line, choose a thin line, and select a color.

# Example 2: Counting

Connect to the file *NYCRatSightings.xlsx* by doing the following:

Click on the Tableau button on the far left end of the tool bar.



Click on *Connect*. Then click on *Microsoft Excel*. Locate *NYCRatSightings.xlsx*. Click on the *New Worksheet* button on the bottom left of the window.

Drag the *Rat\_Sightings(Count)* field to the center of the display area (below *Rows* and *Columns* shelves and to the right of the *Marks* card).

Graphical user interface

Description automatically generated

The number 35,655 should appear in the data area. Note that the green pill says "CNT(Rat\_Sightings)". It is computing the COUNT of the number of records.

Drag the *Boroughs* field to the Rows shelf. You will get a table:

Graphical user interface, text, application

Description automatically generated

Replace *Borough* with *City* on the Rows shelf. You will see a list of city names. Sort in ascending order. Note that there is an entry for **BROOLKYN** (misspelling of Brooklyn):

Graphical user interface, application

Description automatically generated

Drag "CNT(Rat\_Sightings)" off of the Marks card. Right-click on the *Unique Key* field and drag it to the Columns shelf. When you let go of the mouse, the following will appear:

Graphical user interface, application

Description automatically generated

When you right-click and drag a field to a shelf, you will get this list of functions to choose from. Choose *CNT(Unique Key)* as indicated above and click on *OK*. When I do this, I get a bar chart, with a lot of very short bars, and a few very long bars. Let's sort the bars by clicking on the *Sort Descending* button on the toolbar

Graphical user interface

Description automatically generated with medium confidence

Click on *Show Me* and choose *Text Table*. When you do this, the *CNT(Unique Key)* pill moves from the Columns shelf to the Marks card.

Graphical user interface, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Replace *City* with *Borough* on the Rows shelf.



# Example 3

Connect to the *Global Superstore Orders 2016.xlsx*. Drag the *Orders* table (there is also a *People* table—ignore it) to the blank area on the right. Graphical user interface, application

Description automatically generated

Click on the *New Worksheet* button on the bottom left.



In Tableau:

1. Drag the *profit* field to the middle of the Tableau work area.

The total profit will appear as a text value.

Graphical user interface

Description automatically generated

1. Drag it (the green "pill") back again.
2. Drag the *market* field to the middle of the Tableau work area.

A table listing the unique market values will appear.

Text

Description automatically generated with low confidence

Drag it back again.

1. Drag *profit* to the rows shelf. You get a vertical bar representing the total profit.

Chart, bar chart

Description automatically generated

1. Drag *profit* to the columns shelf. You get a horizontal bar representing the total profit.

Graphical user interface, application

Description automatically generated

1. Drag *market* to the rows shelf. You will get a horizontal bar chart.

Text

Description automatically generated with low confidence

1. Move *market* to the columns shelf and profit to the rows shelf. You will get a column chart.

Chart, bar chart

Description automatically generated

1. Drag *segment* to the columns shelf. Our data is now broken down by market and then by segment.

Chart, bar chart

Description automatically generated

1. Drag *market* (from the *Tables* area on the left, not from the *Columns* shelf) to the *Color Card.*

Each market is now a different color.

Chart, bar chart

Description automatically generated

Note that this isn't very useful because they are already grouped by Market. But change the grouping by putting *Segment* before *Market* on the Columns shelf. Now it looks like this:

*Chart, bar chart

Description automatically generated*

1. Reverse *Segment* and *Market* on the *Columns* shelf so that *Market* is again to the left of *Segment*. Drag *segment* to the *Color* card. It now looks like this:

Chart, bar chart

Description automatically generated

1. Drag *Segment* (from the *Tables* area, not from the *Marks* area) to the *Label* card. It adds the name of the segment to the top of the bar.

Chart, bar chart

Description automatically generated

This is not what we want. Remove *Segment* from the *Label* card.

1. Drag *Profit* to the *Label* card. It adds the amount of profit to the top of the bar.
2. Chart, bar chart

   Description automatically generated

You can convert this into a horizontal bar chart if you flip-flop the pills on the *Rows* and *Columns* shelves:

Timeline

Description automatically generated