A regular calculation, such as Sales minus Profit, is passed as part of the query that Tableau asks of a data source and the computation is handled by the data source itself with only the result set being returned to Tableau.

A table calculation is a secondary calculation that performed on top of the returned result set. This computation is done within Tableau. An example is as Running Total of Sales. A table calculation is indicated by this delta symbol on the pill.

The View Data command lets you display the values for all rows in the data source that underlie a set of marks in the view.

Summarized data is shown on the **Summary** tab. Summarized data is a text table of the aggregated data for the fields shown in the view.

All data for the selected marks are displayed on the Full Data tab.

Aggregations are colored light blue in the formula.

Field names are colored orange in the formula. Note those square brackets – this isn't necessary for a single-word field name, but if we were using a field like Shipping Cost, we'd need those brackets because of the space.

standard operators (like addition, multiplication, and comparisons like less than or equal to) are supported in Tableau.

Table Calculations are performed on the returned results of a view.

Partitioning fields scope the data. They define the groups of data the table calculation is performed on – like using Pane to partition our data by Year earlier.

Addressing fields are any dimensions not used in partitioning, and they determine the direction of the calculation. The order of addressing fields also matters.

Have you ever seen an "AGG" at the beginning of a pill? **AGG stands for 'aggregation'** and tells us that the aggregation is built into the calculation rather than performed after the fact.

So we need to aggregate Segment, but how? By using the attribute aggregation – we'll wrap Segment in ATTR – make sure to include the parentheses. what about **ATTR attribute aggregation**? This is another common thing seen with aggregate calculations. Let's say we want to aggregate separately for different dimensions values – for example, we want to average Corporate Sales but take the median of Sales for the other Customer Segments. Attribute checks to see if there is only one value for a given field for all rows in result set. If there is only one value for the data selected, Attribute returns that value. If there is more than one value for that subset of data, it returns an asterisk. Another way of thinking of the attribute function is as the equivalent to the logical test "if min equals max then return that value". We could also use MIN or MAX if we know there’s only one value, though these may artificially pick single value when 3 there are many values that could be there. But all of these aggregations are useful for **aggregating dimensions** to get around errors like the one we saw.

LOD
{[FIXED | INCLUDE | EXCLUDE] <*dimension declaration* > **:** <*aggregate expression*>}

```
{ INCLUDE [Customer Name] : SUM([Sales]) }
```

Note that aggregate expressions cannot contain ATTR or table calculations.

The dimension declaration identifies the dimensions that define the level of detail to use when performing the aggregation. If no dimensions are declared (either with a FIXED keyword or no keyword specified), the LOD Expression is considered Table Scoped, which is a complete aggregate of the data. {sum(sales)}

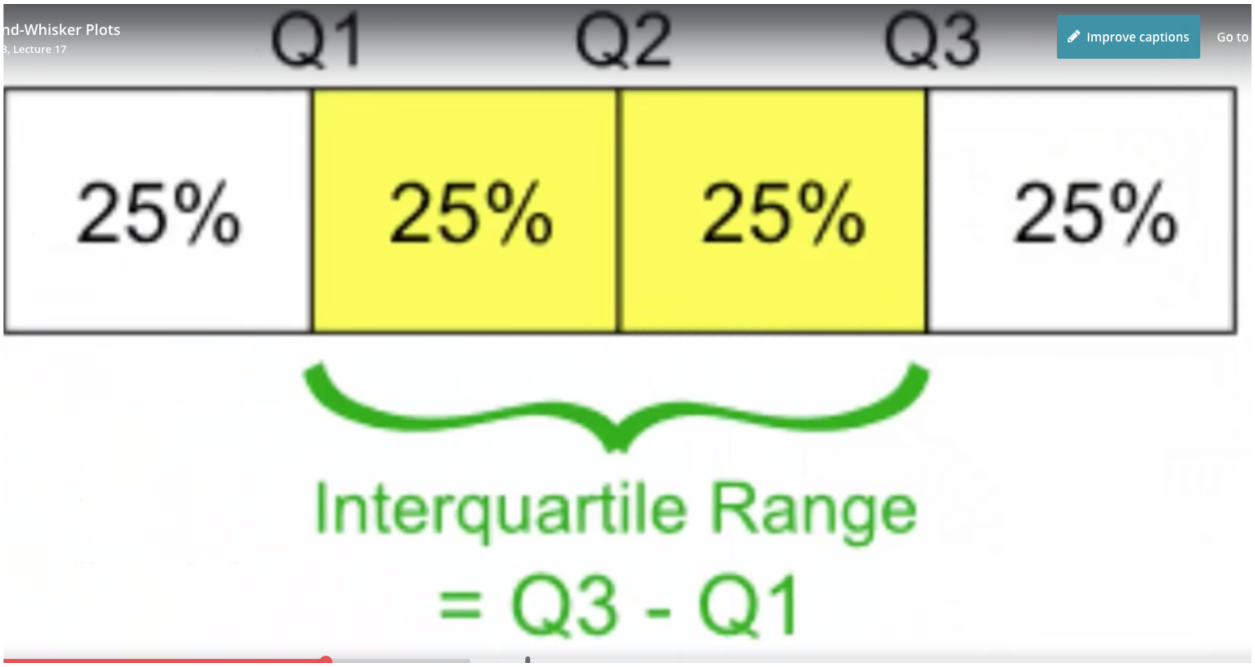
Level of Detail Expressions that use EXCLUDE or INCLUDE keywords will always result in measures. FIXED keyword expressions will default to be dimensions or measures based on the aggregate expression. If the aggregate expression would yield a measure, the overall expression will be a measure, such as Number of Orders per Customer. If the aggregate expression would yield a dimension (such as string, Boolean, or date), the overall expression will be a dimension, such as Second Purchase.

There are several different kinds of filters in Tableau and they are executed in the following order, from top to bottom. If you’re familiar with SQL, you can think of dimension filters as the WHERE clause in a query and measure filters as the HAVING clause. FIXED calculations happen before dimension filters. Effectively, any filter on the shelf will be ignored (if it’s not a context or data-level filter) when computing the FIXED results. This behavior is desired and necessary to make FIXED behave as expected.

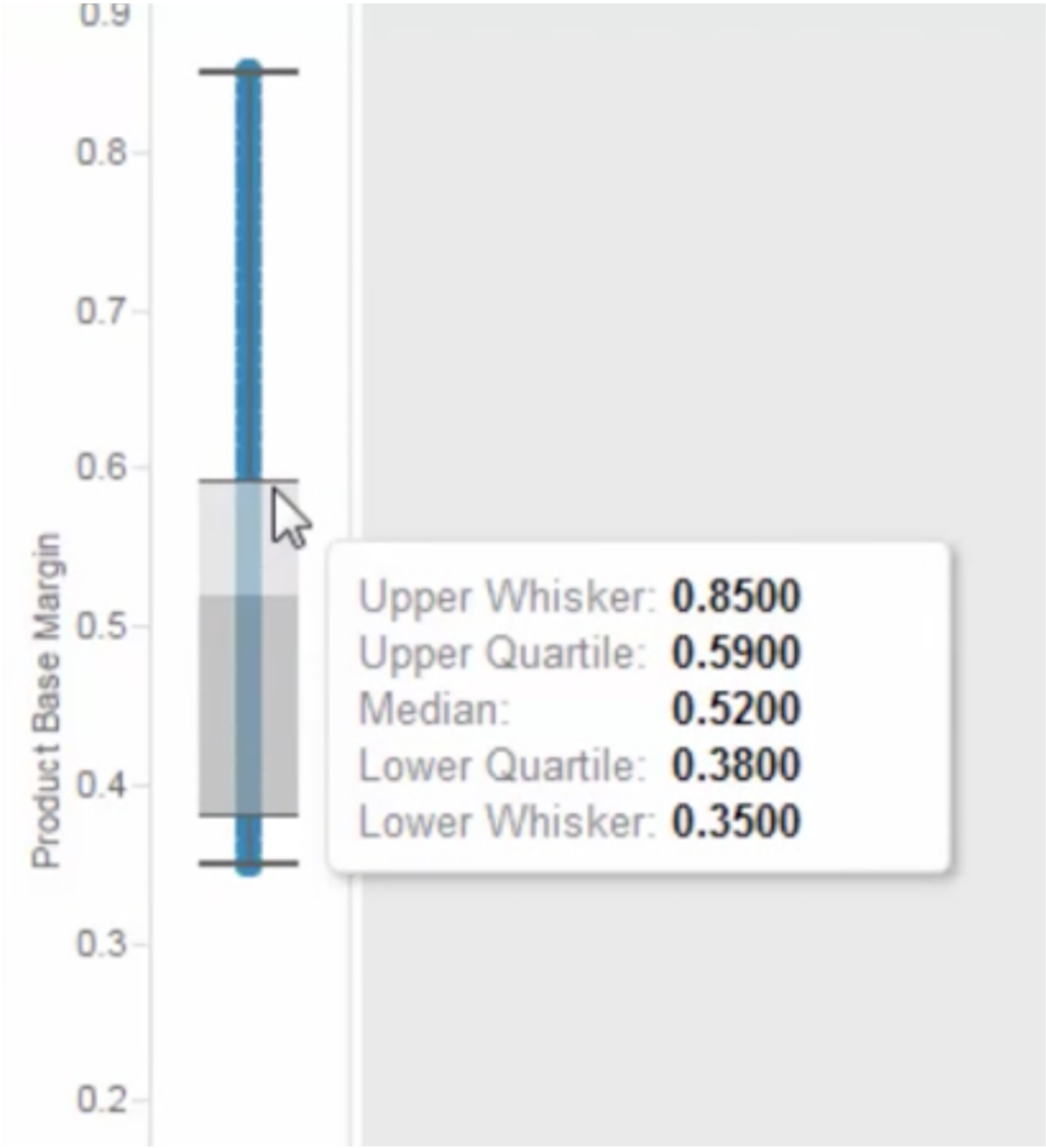
Box and Whisker Plot
IQR- Interquartile Range = Q3-Q1

Box Plot
You can configure lines, called whiskers, to display all points **within 1.5 times the interquartile range** (in other words, all points within 1.5 times the width of the adjoining box), or all points at the maximum extent of the data, as shown in the following image:

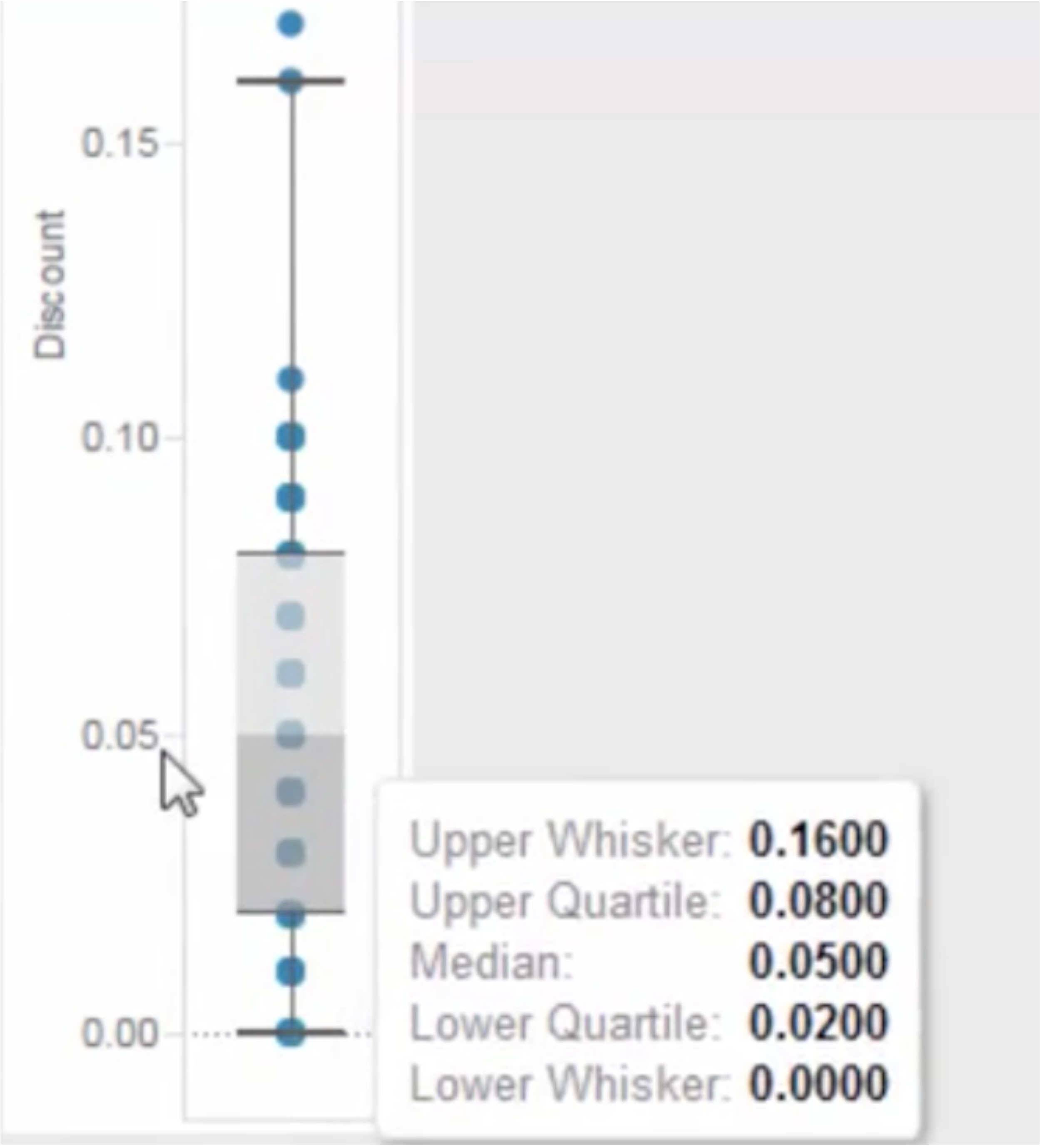
Upper whisker = Q3+1.5(Q3-Q1)
Upper quartile = Q3
Median = 50% quartile
Lower quartile = Q1
Lower whisker = Q1-1.5(Q3-Q1)



Interquartile Range
= Q3 - Q1



Upper Whisker: **0.8500**
Upper Quartile: **0.5900**
Median: **0.5200**
Lower Quartile: **0.3800**
Lower Whisker: **0.3500**



Upper Whisker: **0.1600**
Upper Quartile: **0.0800**
Median: **0.0500**
Lower Quartile: **0.0200**
Lower Whisker: **0.0000**

Quartiles divide a ordered data into four parts. The values that divide each part are called the first quartile (25th percentile), second quartile (50th percentile, also called the median), and third quartile (75the percentile). The difference between the 3rd quartile and the 1st quartile is called the interquartile range (IQR). Think of it as the range for the middle 50% of values.

Aggregate calculations summarize data to the level of detail in the view. The Average function ignores null values. That means if 3+3+0/3=2, however, 3+3+null=3+3/2=3.

COUNTD counts the unique the number of unique values. It ignores the blank or null values. It doesn't count null or blank values. It is not a table calculation. It is a database calculation. So the total is based on the dataset, not the numbers in the table.

Blending doesn't duplicate data if multiple matches are found within the linking field.
Blending combines all rows in the primary data source with matches on the linking field from the secondary data source.
Unlike a LEFT JOIN multiple matches on the linking field do not result in duplication of values.

Trend lines are meant for showing sales or profit or any other important variable performance over year and to know whether performance is increasing or decreasing!

Sometime they are called as best fit lines since they pass through the data points at steady rate!

A p-value is a measure of statistical confidence. In other words, a p-value is a measure of how confident you are that the model will still be valid if you have a lot more data.

The lower the p-value, the greater the statistical confidence.

Traditionally, values below .05 are considered "statistically significant."

The best fit line minimizes the total squared differences between the predicted and actual y-values.

R-Squared: measures how well the model fits the data. It is between 0 and 1. Percentage of variation in Y is explained in the model. The R-Squared values tells you how well the trend fits your data. A value of zero means that your model has no ability to predict the y value, a very low R-squared is not particularly meaningful,
while a value of 1 means that the model perfectly predicts the y value. In this case, our model explains about .8614 or 87% of the variance of y.

Forecasting - using prior values to predict future values.
Prediction Interval - range of values in which future observations will fall, with a certain probability.
Blue pill is for discrete, green pill is for continuous data.

Connecting Live versus Extracting
Connecting live is great when we have constantly changing data or when we want to leverage the high performance database we're connected to. Alternatively, we may choose to import data into Tableau's fast data engine with an extract. This takes the data offline, and allows us to minimize performance impact critical systems, while still allowing for regular, scheduled refreshes to keep the data up to date.

Dimensions and Measures
Fields are broken up into dimensions and measures that represent the column headers in the excel sheet. What are dimensions and measures?
Dimensions are categorical fields, in this case, fields such as date, customer, and Category. These are fields that we want to slice and dice our numerical data by. Dimensions are often discrete. **Discrete fields** create labels in the chart and **are color coded blue** in the data pane and in the view. **Measures, on the other hand, are our metrics. They are the numbers we want to analyze. Measures are often continuous. Continuous fields create axes in the chart and their pills are color coded green.**

Show Me
But we don't necessarily know the best way to view the data. Tableau Desktop provides a simple tool called "Show Me" to help in cases where we know the data we want to look at, but don't know how to create an effective view. **"Show Me" contains a list of common chart types that can help you start your analysis.**
Note: **it's possible to build an enormous variety of charts in Tableau – Show Me is the one-click options, not a comprehensive list of possibilities.**
Let's see Show Me at work by selecting different dimensions and measures while holding down the control key. We're curious about our Sales, and how they're doing in different Countries. Notice how different chart types come available based on what measures and dimensions we've chosen.

Dashboard
Multiple individual views can be combined into a single dashboard.

we can turn on the filter icon here in the border, and the entire map has now been turned into a filter.

Story Points

Story Points lets you assemble a series of specific views to walk the audience through an analysis.

When we do, the word Update appears above the navigator. Clicking “update” will save this state of the viz, **so everyone will see exactly this information.** We’ll title this “But there are problem areas”. This is one of the key aspect of Story Points, the ability to snapshot a specific insight of a visualization while still maintaining interactivity.

Collaboration

The most effective way to share a workbook is to publish it with Tableau Server or Tableau online. Published workbooks are fully interactive, up-to-date, secure, and can be accessed by browser or mobile app.

Menus & Toolbar

At the top, we have the menus. The layout may look slightly different on a Mac. The menus contain a lot of powerful options.

Below is the toolbar, with buttons like save and undo – **there’s no automatic save in Tableau**, so make sure to save your work periodically.

Many of these buttons are contextual to what’s going on in the sheet. For example, the clear sheet button is unavailable here, because there’s nothing on the sheet. If we have a sheet that has data, the clear sheet button is no longer greyed out. And if we click on the dropdown, we see there are options to clear specific aspects.

The logo button here brings us back to the start experience, where we can access saved data sources, recently opened or pinned workbooks, etc.

Data Pane If we’re on the data tab, the top lists all open data sources, and depending on which one is selected, the fields from that data source are listed below, broken out into dimensions and measures. The data pane will also show any sets or parameters we may have.

If we open the map layers or the format pane, these temporarily cover the data pane. To get back to the underlying data pane, simply close whatever's on top of it.
The data pane can also be minimized like so, and then reexpanded.
Down here at the bottom is **the status bar**. **This shows the number of marks in the view as well as other summary information.**

Analytics Pane If we click to the Analytics tab, we can bring out pieces of our analysis directly as drag and drop elements. If they're not relevant to the type of view, certain elements will be greyed out, such as totals on a timeline. If we select something like a trend line, we can bring it to any of these drop areas to control aspects of its properties, like model type and which measure it should apply to.
3 Sheet Tabs
New sheet tabs are found here at the bottom. We can create sheets, dashboards, and stories with these tabs. We can do thing like rename the sheets, duplicate sheets, copy formatting, and many other things. **If the workbook has a lot of sheets, we can navigate easily with the controls in the bottom right corner.**

Shelves and Cards
Finally, and perhaps most importantly, we have the shelves. A view can be built by dragging and dropping fields from the data pane into the canvas directly, or onto the shelves. **Shelves are sometimes also referred to as cards.** There's the Columns shelf and the Rows shelf up here, the Pages shelf, Filter shelf, and Marks Card.
It's also possible to type directly into the Columns or Rows shelf if you know the field you want to use.

Marks Card The Marks Card is made up of several other shelves, each of which can have fields placed on them and can be clicked on to edit their characteristics. **Changing the mark type can change the shelves on the marks card, such as selecting shape brings up the shape shelf.** Depending on the composition of the view, there can be multiple marks cards, one for each measure.

Legends Legends, such as for color, size, and shape, will automatically be created when a field is placed on the color, size or shape shelf. However, **legends can be removed by clicking on the menu, and selecting hide card.** To bring a legend back, right click anywhere off the canvas itself, select Legend and choose the one you want.

Layout for Dashboards & Stories If we're on a dashboard or story instead of a simple sheet, the layout changes a bit. Instead of the data pane on the left, we have the dashboard pane. The main area has a list of all the sheets available. There are also dashboard objects such as images and text boxes. There's also sizing options, and **the device preview helps design dashboards for mobile devices.**

Ways to Distribute

There are several ways to distribute the work done in Tableau Desktop that vary in security, interactivity, and data freshness.

Images and PDFs do not contain the underlying data and are static, both for any interactivity and for data updates because they're simply snapshots taken at a particular time.

Workbooks, either packaged to contain the data or not, can be shared with others who have Tableau Reader or Tableau Desktop. Tableau Reader can view and interact with workbooks. Tableau Desktop can view and edit workbooks. **Packaged workbooks may contain the underlying data and are not encrypted. Non-packaged workbooks require the recipient to have access to the same data source as they do not contain the data themselves.** Workbooks can also be published to Tableau Online or Tableau Server. These workbooks are secure, fully interactive, and can be scheduled to have data refreshes or maintain a live connection to the data source Published workbooks can be accessed via any major web browser as well as on mobile devices with the native Tableau App.

Exporting Images and PDFs Views can be exported by going to Worksheet > Export and selecting Image, Data, or Crosstab to Excel. Selecting Image brings up options for what should be included, such as legends and titles, as well as the layout. And the file can be saved in various formats.

Dashboards and Stories can be exported by going the Dashboard or Story menu and selecting Export Image. **The Dashboard image will be the**

current state of the Dashboard; the Story image will contain the current story point in its current state.
The entire workbook or a specific sheet can be printed to PDF by going to File > Print to PDF and choosing the appropriate settings. Printing the entire workbook includes each story point.

Workbook File Types Workbooks in Tableau Desktop can be saved as .twb files or .twbx files. **.twb workbooks are not packaged with the data itself – they contain just the information needed for the data connection and construction of the view.** Opening a twb file requires access to the same data source used to create it. **.twbx workbooks are packaged – they contain the information in a twb and can contain any data and local files such as background images and custom geocoding.** As noted previously, **twbx files are not encrypted and have no data security** – opening a twbx file with data shows all the underlying data. Opening Workbook Files – Desktop and Reader Packaged workbooks can be opened with Tableau Desktop or Tableau Reader. Opening a file in Desktop is the same experience as authoring a workbook – there is full functionality including the ability to edit or create new worksheets from the data. **Tableau Reader can open packaged workbook files to be viewed**, and preserves full interactivity of the workbook contents including dashboards, stories, and any filtering or other actions, but the workbook cannot be edited.
Publishing Securely – Tableau Online and Tableau Server
Finally workbooks can be published to Tableau Online or Tableau Server. These workbooks are secure, fully interactive via browser or mobile device, and can be set up for automatic data refreshes or be connected live some data sources.

Configuring Parallel Queries in Tableau Desktop
https://kb.tableau.com/articles/HowTo/Configuring-Parallel-Queries-in-Tableau-Desktop

Data Engine Vectorization: As of Tableau 9, the Data Engine will take advantage of vector instructions on current processors. This enables speeding up some calculations. The logs show which level of vectorization is supported by your processor under "Vectorization Support". **9.0 preview: Query performance improvements** https://www.tableau.com/about/blog/2015/1/90-preview-query-performance-improvements-36406

Tableau's Pace of Innovation
https://www.tableau.com/fast-pace-innovation

Shadow Extracts

Shadow extracts are only created when you work with workbooks that are based on non-legacy Excel or text, or statistical files.
Tableau creates and saves a shadow extract (.ttde files) in order to load data more quickly. After Tableau creates five shadow extracts, Tableau deletes the oldest shadow extracts to create space when it adds a new one.

Although shadow extracts contain underlying data and other information similar to the standard Tableau extract, shadow extracts are saved in a different format that cannot be used the same way as Tableau extracts. Shadow extracts cannot be used to recover data.

https://kb.tableau.com/articles/issue/low-disk-space-because-of-ttde-files

ttde are Shadow Extract files used for caching speed within clients and not directly referenced by users

Shadow extracts create a faster text-based data source experience. A workbook with a large text or Excel file re-opens with improved speed.

Tableau is able to connect to a wide range of data sources, including text and Microsoft Excel files. Both text and Excel files are highly popular file formats for storing data.In the past Tableau used Microsoft Jet/Ace drivers to query text and Excel files. This approach had a number of drawbacks, such as lack of portability to other operating systems and a 4GB parsing limit. Furthermore, running analytical queries over these data sources was inherently slow because the system had to parse the file for every query.Shadow extracts have been introduced to speed up the query execution and overcome the Jet limitations. When a text or excel file is connected, Tableau extracts the data from the file, and stores them in temporary tables in the TDE. Subsequently, all queries are executed by the TDE instead of parsing the entire file each time. This greatly improves the query execution time, however, we need to pay a one-time cost of creating the temporary database. Last but not least, the system can persist extracts in workbooks to avoid recreating temporary tables at every load.
In order to effectively extract data from text and Excel files, and overcome the Jet/Ace limitations, Tableau uses an in-house parser for parsing text files and LibXL for parsing

Excel files. These parsers are both more efficient, do not have the 4GB limitation, and are cross-platform. The text parser accepts a schema file as additional input if one is available. Otherwise, it attempts to discover the metadata by performing type and column name inference.

When you work with Tableau Desktop 9.0 and later, your My Tableau Repository might contain a folder called Shadow Extracts. By default, the Shadow Extracts folder can contain up to five shadow extract files, which have a .ttde extension.

The contents of the Shadow Extracts folder change depending on the workbook you work with, and shadow extract files are only created when you work with workbooks that are based on non-legacy Excel or text, or statistical files.

Tableau creates and saves a shadow extract file in order to load your data more quickly. After Tableau creates five shadow extract files, Tableau deletes the oldest shadow extract file to create space when it adds a new one.

Although shadow extract files contain underlying data and other information similar to the standard Tableau extract, shadow extract files are saved in a different format, which means that they cannot be used the same way Tableau extracts are.

When used: when processing non-legacy excel files, statistical files, or text files. Why? Processing some file formats is CPU/IO intensive.

Location: - For Windows: Users\AppData\Local\Tableau\Caching\TemporaryExtracts - For Mac: ~Library/Caches/com.tableau.Caching/TemporaryExtracts - Sometimes also in the "My Tableau Repository\Shadow Extras" folder.

Shadow extracts is data that Tableau stores when using file (like non-legacy excel) to make loading data faster. They have .ttde extension. Tableau will store up to 5 files with .ttde extension. Altough they're named extracts, its file format is not the same as .tde (extract) files.

http://kb.tableau.com/articles/issue/low-disk-space-because-of-ttde-fileshttp://www.icancrack.com/index.php/52/what-is-shadow-extract-in-tableau

Creating Extracts

If the workbook has already been saved as a .twbx packaged workbook, the extract will automatically save as part of the packaged workbook. If the workbook has been saved as a .twb or hasn't been saved yet, we'd be asked where to save the Tableau Data Extract file (.tde).
We can also create an extract right when we connect to data for the first time.

Extracts create a snapshot of your data at whatever point they were created. They are typically faster than a live data connection, especially when connecting to a live database, and are my general recommendation. Just remember that extracts have to be refreshed periodically so that you are working with the latest data possible.

Using Extracts
Notice that the icon here has changed from a single cylinder to two cylinders with an arrow on it. This indicates that the data has been extracted from its native environment into the fast data engine. The original data is untouched.
Tableau Desktop - the ability to replace a data source.
We have two data sources – the blue check means this is the primary connection currently being used in the viz.

Join
Join from different databases, one table will be highlighted in yellow color.
Join from same data connection, there won't be color differentiation.
Adding data connection will automatically join the newly added table in the new data source. Whereas adding data source won't have this automatic setting.

An outer join brings in all names listed in all tables, and fills in nulls wherever there isn't information for a given column for that row.

Union
When you create a union in Tableau, a column will be added that tells you what sheet the data came from.

Tableau follows the behavior of a **UNION ALL**, that is, all rows will be returned in the union, even if there are duplicate values for some rows (such as Kai). In

Tableau, new fields are generated with the Table Name, which indicates metadata about the union's source. Finally, it's worth noting that if the field names don't match across the unioned data sources, Tableau will function similarly to an **Outer Join**, **appending the new rows AND new columns, with nulls as needed. If the new columns should be considered the same field, the columns can be merged in the data grid.**

Filter
The final option discussed in this post is the ability to filter the entire data source before you start working with it in Tableau. These filters can be created with any combination of fields by clicking the "Add" button under "Filters".

Data Blending https://onlinehelp.tableau.com/current/pro/desktop/en-us/multiple_connections.html
Data blending is one way of combining data from multiple data sources into a single view. Instead of joining the data at the row level like a cross-database join, data blending sends separate queries to the separate data sources and aggregates the results to a common level back in Tableau.

Whenever we are connected to multiple data sources in Tableau, the first data source we bring out to the view becomes the primary, denoted by 3 this blue check mark. We now see there's an orange check mark next the Coffee Chain data source and on the pill in the view, this indicated that it's from our secondary data source.
Tableau is querying for Sum of Sales by State to each data source, then displaying those values together in the view. We see sales information for every state from Office City because Tableau returns information for all field members (states) in the primary data source, regardless of whether or not there is data from the secondary data source.

It's important to note that primary and secondary sources are determined on a worksheet-by-worksheet basis and are not maintained globally throughout the workbook. When we are on a new sheet, the data sources within the Data Pane do not have orange and blue check marks to indicate them as primary and secondary. The relationships we established in the previous worksheet are not carried over.

Data blending in Tableau is very similar to a left join. In that, it takes a primary table and matches values from that primary table using some key field with a secondary table. One of the important differences is that it does not duplicate rows in the primary table even if it finds more than one match in the secondary table. *If the primary table has more matching rows, the data from secondary table won't be duplicated either. Only one row will have the data from the secondary table.*
But, again, it is similar to a left join in that, it will not drop any rows from the primary table even if no match is found in the secondary table. So, it's a way to combine data sets, very similar to a left join but with a difference that it will not duplicate data in your primary table.

Because the **Fines** field is a measure, you see the row values for the **Fines** field aggregated before the data in the right table is combined with the data in the left table.
There are multiple corresponding values in the rows in the right table, as indicated by the asterisk (*).

Superstore is a data set of sales for a global retail chain that sells furniture, office supplies, and technology goods. **Each row of data represents a single item in a transaction.**

Sheets in Excel are treated the same as tables in databases, and we can choose to connect to a single table or join multiple tables. Simply drag a sheet to the canvas. Tables can be renamed simply by double clicking on the name.

A workbook can have multiple data sources. Just click the Add Data source button. Note: clicking "Add" will add another a connection to the same data source, setting up a cross database join.
To add another data source entirely, use the new data source option in the toolbar.
To connect to a locally saved extract, when connecting to data, choose "other files" then simply navigate to the extract on the machine.
To connect to a data source your organization has published to Tableau Online or Tableau Server, choose "Tableau Server" and enter your credentials. This 3 will show you every data source you have access to. As a note, you won't see the side pane when connecting to a data source from Tableau Server or Tableau Online, because the data source is pre-defined.

Live VS Extract
Connecting live leaves the data in the database or source file. This is best when we want to leverage a high performance database's capabilities, or to get up-to-the-second changes in data visualized in Tableau. That being said, sometimes connecting live can result in a slow experience, depending on the database.

The other option is to extract the data into Tableau's high performance in-memory data engine. This can help when connecting to a slow database or to take query load off critical systems. We can also choose to only import some of the data and bring in specific elements. To access those options, click Edit.

Connecting to data from the web, Saving the workbook

Once we do our analysis and save the workbook, we can navigate to the workbook and then to its data sources. Note that while the data is available in the workbook, it is not published as an independent data source. Connecting to data from the web is done in the context of a workbook and is not a substitute for publishing a data connection to the site. Data Freshness Note that when we connect to data from the web, files are static and database connections are extracts. If the data needs to be kept fresh, either with live connections or scheduled extract refreshes, it's best to build workbooks off published data sources.

Editing Metadata

Tableau cannot write changes back to the data source. However, we can modify the metadata for use in Tableau in the data pane.

Fields we don't need can be hidden, We can rename fields, We can create a hierarchy by simply dragging a subordinate field onto another field. To add another field, simply drag it into the correct spot. Or drag to rearrange the order. This creates the ability to drill down in the view. A handy thing to know is that Tableau has this search function, which is helpful if fields are in folders. We can change the data type. And we can assign default colors simply by bringing a field such as market to color and editing colors. That choice will be remembered the next time we use the field. If we right click on a measure and go to Default Properties , we see that there are several features nested here such as adding a comment, editing the default number format, and changing the default aggregation.

The important thing to note here is that all of this is part of the definition of the data connection, not actual changes to the underlying data. When using a published data source from Tableau Server or Tableau Online, we will not be able to edit or remove existing metadata such as hierarchies, aliases, or calculations, but we can extend metadata, such as authoring new calculations for use in the workbook we're building. This will not write back to modify the original data source.

Saving Data Sources

Imagine you've put in a lot of effort managing your metadata: re-aliasing fields, creating calculations, setting default colors, etc. You want to keep all those efforts to use again later. You can locally save a data source, making it easily available in your copy of Tableau Desktop. Or, to securely share a data source with others, it can be published to Tableau Online or Tableau Server.

If we want to save a data source, we right click on the data source itself and say "add to saved data sources". This will save the connection as a .tds (note: this is not the data itself, just the information about the connection). That data source will now be in the Saved Data Sources list in in your local copy of tableau, making it readily available the next time you want to use it.

Why Publish Data Sources?

Alternatively, a data source can be published to Tableau Server or Tableau Online. Once published, a data source can be used by anyone with the correct permissions. They don't have to install drivers, or do data prep or curation themselves. All workbooks that use the published data source will be automatically updated when the data is refreshed or changed, maintaining a single source of truth.

a live data source (indicated by a single cylinder) or an extracted, in-memory data source (indicated by a double cylinder).

Publishing to Tableau Online
Because Tableau Online is in the cloud, and therefore outside the firewall, the process is a little more complex.
If we're publishing an extract to Tableau Online, refreshing the extract is handled with the Tableau Bridge client.

Data Preparation
there are several features in Tableau Desktop to help automatically reshape Text and Excel files into this format for better analysis in Tableau.

Data interpreter
"Review the Results", we see which fields are being used as headers, in red, and which are considered data, in green.

Pivot
This pivot feature essentially merges the information from the original columns and rows into two new columns – Pivot field names, and Pivot field values.

Metadata Grid
the metadata grid. This can be a useful view, as the vertical layout can be easier to navigate, especially if you have a large number of fields, and it's also useful when tables have been joined, or to see the original field names.

Connecting to PDFs
Tableau looks at absolute 3 page numbers, which may not correspond to the pagination in the document. The "rescan PDF" option, under the data connection dropdown, lets us re-pick what pages to look at. There's only one table on the page, but there are 3 options to the left. Tableau detected 3 possible ways to pull in that table. If we bring out each one at a time, we can see what they contain.

Note: when unioning tables across pages like we did before, be sure to union the correct table version from each page, not multiple versions of the same page.

As a reminder, Tableau will have trouble with PDFs containing: Sub tables, Hierarchies in headers, "single rows" that are actually multiple rows of content that should be interpreted as a single row. Finally, note that colors and shading can change how data is interpreted because of how PDFs must be parsed to cells and tables of data.

Connecting to Cubes

cube data sources are unsupported on Macs. Cube data sources, such as Microsoft Analysis Services and Oracle Essbase, process and transform data from a relational database into pre-aggregated results with defined hierarchical structures.
The pre-defined nature of a cube data source does affect some functionality in Tableau. If possible, we suggest connecting directly to the underlying relational database.

When using a cube, we cannot take an extract. There is no equivalent of the “Other Databases” option for unsupported cube data sources. And at this time Macs do not support connecting to cubes, so everything in this video is applicable only to Tableau Desktop in Windows, though workbooks using cube data can be published and will function in the browser and on mobile devices.

KPIs Cube data sources allow for the definition of KPIs as a data type. Tableau does not recognize these KPI data types, but we can recreate them in Tableau just as we would when using a relational data source.
Grouping When using a relational data source, there is a paperclip icon in the tooltip to group fields. This option is not available when working with cube data sources. Grouping can be done with a Calculated Member written in the MDX language, which is covered in the Analysis with Cubes and MDX video.
Unlike relational databases, we do not have the ability to preview the data when connecting to Cube data.
but you’ll notice that the sales field doesn’t have an aggregation like SUM or AVG next to it. The aggregation is part of the cube definition and cannot be switched within Tableau. Data structure and aggregation changes will require working with the cube’s architect. When we right click on the Sales in the Rows shelf, there is no aggregation menu option. However, quick table calculations are still available.
Hierarchies and Filters In the Dimensions pane, you will notice that hierarchies, including time fields, are also predefined in the cube and cannot be changed here. Cube hierarchies have a different type of filter than those in relational databases – they show all of the levels of the hierarchy and allow for “ragged” selections, where totals from different hierarchy levels are visible side by side. Tableau shows all of the levels expanded by default, but if we double click on a hierarchy level, it will collapse down.When using cubes, a continuous date is always at the lowest level of detail and Tableau cannot roll it up to a Yearly or Monthly continuous view.
If we want our filters to look more like those that Tableau creates in a relational data source, we can use Sets. Expand the Product Groups hierarchy and Right-click on Product Category. Choose “Create Set”.

Working with Marks
There are two ways to start analyzing data in Tableau. You can either jump in and start exploring, or you can ask questions of your data and attempt to answer them.

Building the View
Measures, are automatically aggregated to the granularity of the view. **The granularity is set by the dimensions and how they’re asked to interact with the marks.**

Dimensions define the number of marks.

Level of Detail
If we want to make marks at a certain granularity, but without encoding by color or shape, we can bring the field to the Level of Detail shelf.

It’s important to know how adding dimensions to the marks card impacts the visual; the better understanding we have over this behavior, the more precisely we can build graphs manually or leverage what Tableau can do automatically.

Highlighting

Highlighting can be a great way to call attention to specific marks. In Dashboards, highlighting actions can provide interactivity-even across worksheets using that field. In Stories, highlighting can be saved to preserve a specific selection by updating the point.

We can also show a highlighter for one or more dimensions that are not encoded by color or shape just by turning on the data highlighter. Right click on a dimension in the view, such as Market on the detail shelf, and select Show highlighter.

Show Me

Another way to get started with visual analysis is to use Show Me. Control click multiple fields you want to use, we'll use the same ones we were working with, Order Priority, Market, Shipping Cost, and Profit, then open Show Me. Here we can see the one-click graphical options for representing the fields we selected. **The orange box indicates visual best practices.**

Remember, **Show Me is just a starting point for creating visual analysis in Tableau.** Once we have the basis of the visualization, we can use the marks card and many other features to modify the view to make it exactly what we want.

Hierarchy Drill Down

Drilling down can mean a couple things in Tableau. Usually, it means expanding out a hierarchy, thereby drilling down to the next level of detail. Or it can mean to go all way down to see the raw data.

Date Hierarchies

Date fields, depending on their level of granularity, are automatically brought into the view as hierarchies.

The default pill type for a date field is discrete, and a discrete date hierarchy will have a different pill for each date part. However, if we right click and drag out the date, or change the pill type from the dropdown menu, we can select a continuous date instead. Continuous dates still have the same drill down functionality, but each drill down updates the pill to the next level down instead of creating a new pill, because continuous dates by definition cannot be considered as separate date part. To drill back up, open the dropdown menu by clicking on the caret in the pill and selecting the desired level.

Sorting

Ways to Sort

First, and perhaps easiest, is the **Quick Sort on the axis**.

For views with an axis, hovering brings up the Quick Sort icon. One click sorts the bars descending, another click switches to ascending, and a third click clears the sort. This kind of sorting helps identify the top or bottom values in a pane (within the context of how the view is built).

Second, toolbar buttons allow easy sorting. Click on the pill we want to sort by, in this case Profit, then select the appropriate sort from the toolbar.

Third, We can also **sort from the field labels.** If we want to sort the regions, we can hover over the field label (which is the name above the pane) and choose to sort alphabetically or by any of the measures. If we want to sort the markets as well, we can hover over that field label and sort again.

Fourth, sort by dragging field in the view.

Fifth, sort from the Pill.

When we sort by the field label/pill, we're sorting across panes, so to speak, and each pane (Ship Mode) will have the same order of the next dimension that's being sorted (Order Priority). **By contrast, a nested sort, which is the default behavior when sorting from the quick sort icon, simply sorts each bar within each pane.** The longest bar in the First Class pane, High priority orders, is first, independent of the fact that for Standard Class orders, Medium priority orders are first.

Grouping

- Grouping from the Header
- Grouping from the Data Window
- Grouping Marks – Visual Grouping such as scatterplots
    - Visual grouping has a downside that the grouping is static.
- using calculations
    - The example we just did is dynamic as the data changes, but the calculation itself is static. Parameters can be used to make the calculation interactive.
    - Create a parameter. Next, we need to create the calculated field – otherwise, this parameter isn't connected to anything and changing the value won't make an impact.
    -
- working with bins.
    - Sometimes an analysis calls for grouping values of a measure into bins.
    - Here, we'll create bins for our Sales measure. Right click on that field Sales in the data pane and select Create > Bins.
    - We see the range of values for this field automatically, and Tableau has suggested a default size for the bins, but this is easily edited. Let's make it 1,500. When we click OK, we've created a new dimension because dimensions are how we break out measures into categories, which is exactly what we want the bins to do. Now we can use this new field. We'll bring the bin field to Columns and the Sales field to Rows. As a note, **bins labels indicate the inclusive lower limit.**

Sets

Sets in Tableau can be thought of as combinations of data, like filter results.

Sets can be created by the user in Tableau Desktop from specific data in the view or created via a calculation. They will appear in the data pan at the bottom, labeled with the set icon of two overlapping circles.
Some data sources may already contain sets when they're brought into Tableau, these will be indicated with a database icon together with the circles.

Sets made from marks are called "**constant**". Constant sets are quick andeasy, though as the name suggests they are not dynamic.

One option we have here is to exclude. This will make a set of all the marks we did not select. Sometimes it's much easier to select the members we don't want in the set. We can also X out any of these dimensions here that are currently being used to define the set. Remember, sets are like filter results. If we remove a dimension here, we're removing it from the requirements to be in the set.

Creating Sets from a Computation
If the analysis requires a set whose members are dynamic and update as the underlying data changes, the set needs to be computed rather than constant.

Combining Sets
Sets based on the same dimension can be easily combined.
If we have two sets, A and B, the possible combinations are: All Members in Both Sets (**known as A union B**). Here, any orders that are EITHER negative Profit OR high Shipping Cost will be in the combined set. Shared Members in Both Sets (**known as A intersection B**). Here, any orders that are BOTH negative Profit AND high Shipping Cost will be in the combined set. **Except Shared Members** is direction specific, so has it two options. Any negative Profit order that is NOT high Shipping Cost OR any high Shipping Cost order that is NOT negative Profit. We'll go with this one, and select OK.
Once made, Combined Sets function just like any other set. Combined Sets built off of dynamic sets continue to be dynamic.

Editing Sets
Sets can be edited, but to what degree is dependent on how they were created.
IN/OUT versus Members
Sets can be used like any other field, but they have the additional feature of In/Out versus Members. By default, when a set is brought into the view, say to color, the marks are colored for which fall within the set and which are not in the set.
Sets and Filters
Sets can be thought of similarly to filter results. If there is a subset of the data that can be thought of as a useful set, or a frequently used combination of filters, it may be worthwhile to create a formal set from those filter results, and then use the members of that set instead of constantly evaluating the filters.

Sets in Hierarchies and Calculations **Like Groups, Sets can be added to hierarchies.**

Pill Types and Filtering
Note that the options presented when filtering varies by pill type.
Filtering Discrete Dimensions

Filtering Measures or Continuous Dimensions
If our field is a Measure specifically, first, we're offered the Filter Field and **asked to specify a level of aggregation.** For a discussion **on aggregate versus record level filtering**, please see the video "Where Tableau Filters". For now, **we'll click All values, which is record level filtering,** and now we're brought to our options for all quantitative filters.

Filtering Continuous Dates
When a date field, as identified by this
calendar icon, is brought to the filter shelf, we're given a Filter Field just like with a Measure and asked to specify how we want to filter the date. **A discrete date is treated like a dimension, and there are specific discrete date parts listed here. Note that they're blue.**

Continuous dates, however, have their own date specific options. **Here we can pick range of dates or relative date.** These will bring us to the options for continuous dates. **Relative dates let us set a specific unit of time.** Maybe we'd like to look at the last 2 years. We can set ranges, including something like week to date. By default the filter is anchored to today, which is dynamic, but we can change it to a static date. **Range of dates, starting, and ending are similar to measures, but with calendar date pickers.** The interactive filter for a continuous date comes up as a range, we can use the slider, or click on the date to bring up a calendar picker or type in a date. Ranges are inclusive – the data for boundary dates will be shown.

The title can be toggled on/off or edited. **It's often good to put in an action word in the title,** something like "Choose a Sub-Category". a filter layout should be
based on both the screen real estate but also thinking about how the filter will be used. The filter layout should mesh with the purpose of the dashboard.
Cascading interactive filters are a great way to get interactivity into your analysis, or to tidy up long lists of values to make it more intuitive for the end user. However, they do require more work for the queries back to the database and may impact performance. As a note, hiding the quick filter control does not remove the filter; it's necessary to remove it from the filter shelf or from the menu of the filter.
Interactive Filters are a great way to let end users interact with a visualization. Do note, however, that excessive numbers of Interactive Filters may slow down the performance of a dashboard or workbook.

Summary/Aggregate Filtering

Summary or aggregate filtering is essentially saying “filter out any marks in my view that are outside the constraints I set”.

Record Level Filtering

When we bring a measure like Profit to the filter shelf, selecting “All Values” essentially will exclude any record that falls outside the filter we build. If we set the minimum to 0 like before, we’re actually computing our view using only rows that have a profit of 0 or greater.

Data Source Filtering

To create a Data Source Filter, right click on the data source itself and click Edit Data Source Filters. (This can also be done when we’re originally creating the connection). We’ll “Add” and a list of fields is brought up. We’ll select the one we want to filter on, Market, and click ok. Now we can select the Market we want – we’ll choose Europe. Note that we don’t have anything on the filter shelf, but we only have countries here that are in the European Market. Data Source Filters apply to all sheets using the data connection, so be careful when using these.

Context Filters

One thing to note about the filter shelf is that the filters are computed independently of each other. If a given filter will trim the dataset down, it may be worth Adding to Context. A Context Filter will be computed first, then all other filters will run only on its results.

For example, if we knew that we only wanted to look at products within the Machines sub-category, we may want to make that a context filter by right clicking and selecting “Add to Context”. The filter is now grey, and any subsequent filters will only be on machine products. Context filters are generally best to use when they’ll return a significantly smaller set of results, the guideline is about a tenth or less of the original data. Context filters ideally shouldn’t be something that would change frequently, so let’s hide this interactive filter.

Apply to Worksheets

When a pill is placed on the filter shelf, by default that filter applies only to the current worksheet. If it makes sense to have the filter applied more broadly, we can change the scope by clicking the pill’s dropdown chosing, “Apply to Worksheets” and applying the filter to “All using related data sources”, “All using this data source”, or “Selected Worksheets”.

This can be especially useful on a Dashboard, where a filter that is relevant to multiple views can be set to apply to all or some of them simultaneously. By default, the interactive filter only applies to the view it was brought out with. But if we go to the menu and say “apply to worksheets” > “All using this data source”. Now all the views filter

Which Field to Filter On

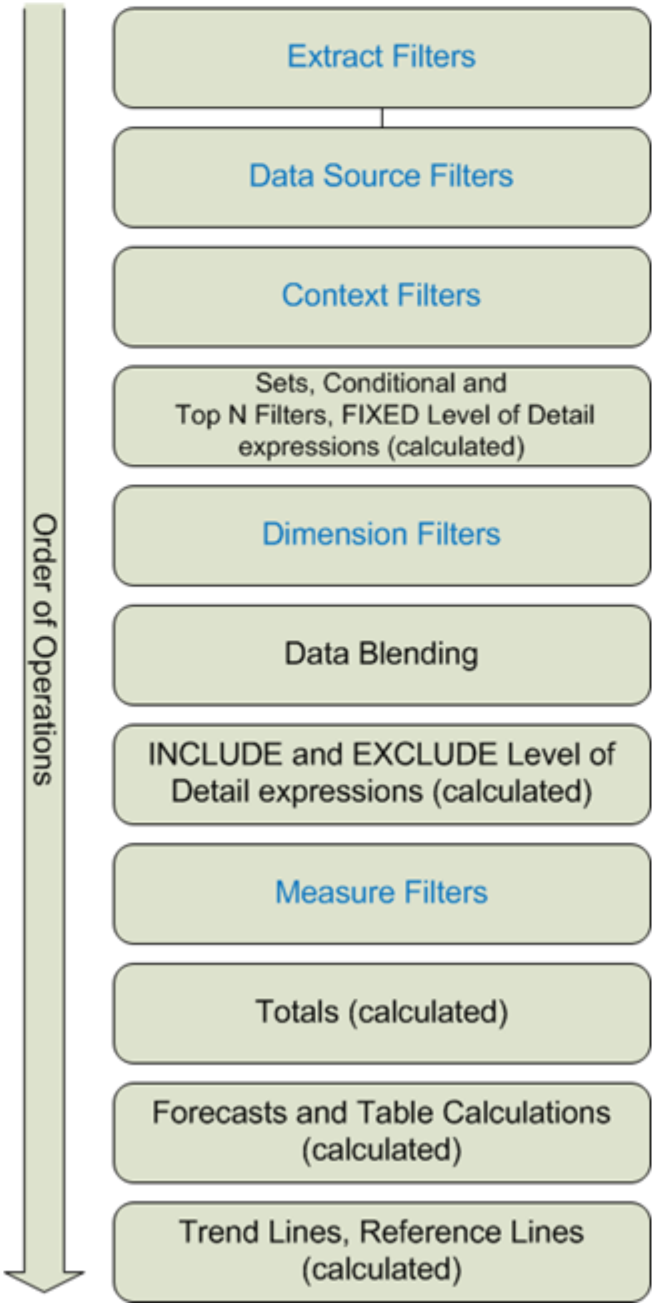
Knowing how we set up our filter
will impact how we can manipulate the view and maintain the filter as we designed it.

Filter Order of Operations

https://onlinehelp.tableau.com/current/pro/desktop/en-us/help.htm#order_of_operations.html?Highlight=order%20of%20operations%20for%20filters

What order filters are put onto the filter shelf has no impact on the query sent to the data source, filters are grouped together into the WHERE clause of the query. However, filters ARE applied in a fixed order of operations.

Filters created during the Extract process limit what data is brought into the extract itself. Then, Data Source filters are applied, limiting what is made available in Tableau. Once the data is in Tableau, Context Filters are applied first and all subsequent filters are run against their temporary table of results. Filters involving FIXED LOD expressions are evaluated next. Then filters on dimensions (like ship mode). Then INCLUDE or EXCLUDE LOD filters. Then filters on measures (like shipping cost). And finally Table Calculation filters, which are applied last (because these are performed only on the data in the view).


Order of Operations
Extract Filters
Data Source Filters
Context Filters
Sets, Conditional and Top N Filters, FIXED Level of Detail expressions (calculated)
Dimension Filters
Data Blending
INCLUDE and EXCLUDE Level of Detail expressions (calculated)
Measure Filters
Totals (calculated)
Forecasts and Table Calculations (calculated)
Trend Lines, Reference Lines (calculated)

Performance and Filter Queries

Inefficient filtering is one of the most common causes of poorly performing workbooks and dashboards. For detailed information, see the related materials below this video. Some things to consider about filtering and its impact on performance: **Context filters are slow to create or change due to the temporary table being created behind the scenes, but once they are in place they can increase performance since further queries are only run off a subset the subset of data as opposed to the entire data set.** Cascading interactive filters can help narrow down the list of options, like here, we only see states that are in the country we selected. **However, when a filter requires Tableau to find all potential field values, this requires a complex query that can take time to evaluate.** Here, using a wildcard may be more performant. The presence and maintenance of indexes in the data source can dramatically improve filtering performance.

What is a Parameter

Parameters add interactivity and flexibility to a workbook. We can think of **a parameter as a variable in an equation whose value can be controlled by the end user.** Parameter controls, like Shipping Cost Cutoff here, are sometimes confused with Filters, like Ship Mode. However, **Filters trim down the data in the view,** showing data only for the specific Ship Modes. **Parameters simply provide a single output to another element**, such as providing the reference value for the calculated field on Color.

Using Parameters

Because parameters simply control a variable's value, they are only useful once that value is incorporated into something else such as a filter, set, reference line, or calculated field.
Changing this orphan parameter has no impact, because that parameter is not tied to anything, it's like adjusting a dial with no wires attached.

parameters are workbook-wide and can be used in multiple places.

Parameter Types

parameters support a variety of data types. Back on the data pane, right click and select "create Parameter". Data type options include **float (decimal), integer, string (text), Boolean (true/false), date, and date & time.** Which data type is selected impacts where the parameter can be used and the display formats available.

Copy & Paste Formatting

Right click on the sheet tab of the formatted view and say "Copy Formatting". Right click on the destination tab and select "Paste Formatting". Note what is copied and pasted – worksheet formatting only. The mark type, colors, sizing, and reference lines are view-specific, and don't transfer. The axes formatting, and things like
lines, did copy. **A quick way to remember this is that formatting done in the format pane will copy and paste. Formatting done in the view will not.**

Workbook Level Formatting

Finally, we can format the font at the level of the workbook. Click on the format menu and select Workbook. We can control all the fonts in the workbook for a unified, consistent experience across the entire workbook.

Tooltips

By default, the various fields used to create a viz are added to the tooltip. Hovering over a mark provides that underlying information.

Tooltip selection

Command Buttons

Actions from tooltips

Conditional tooltips

Conditional tooltips

When the mouse moves over different bars here, watch the Order priority at the bottom. See how it changes to match the colors in the bar chart? How is that done? This isn't quite as straightforward as some of the other uses of tooltips, but it can be done fairly easily.

The key is to create a new field for each possible option that needs its own color; here, that's a field for critical, a field for high, etc. If we right click and edit, we can see how this was made. The calculation says, if the order priority field is low, then return the order priority, otherwise, and then just two double quotes with nothing inside them. This means that if the order priority is NOT low, the calculation will return nothing. The other 3 calculations are the same thing, just with other options here.

If we go to the tooltip, we can see that these four were inserted, like we did before, immediately after each other. Because only one order priority can be true for each order, the three that are false will show as nothing and only the true value will appear. Then it's simply a matter of coloring each field as desired. I used the More colors > pick screen color, to match the color legend.

ATTR(expression)

Returns the value of the given expression if it only has a signle value for all rows in the group, otherwise it displays an asterisk (*). Null values are ignored. Example: ATTR([Market])

Reference Lines

Reference lines do exactly what it sounds like they'd do – they add a point of reference to a view. Reference bands or boxes can also be added to shade a specific area or distribution in the view, and multiple reference lines can be added to the same chart.

Box Plots

Box Plots are fairly straightforward – **they're a common way to show statistical distribution. We can set the whiskers to be 1.5 times the interquartile range, or to the max of the data.** We can tweak the visual, hiding the data that would be under the box plot, and formatting the boxes and whiskers. Box plots are available in Show Me.

Adding Trend Lines

By default, trend lines are per pane and per color.

Trend Line Significance

In statistics, p-value is the number assigned to the concept of significance. **If the p-value**

is less than a cutoff value, usually 0.05, the results are interpreted as significant. A large p-value (on a scale of 0-1) can indicate that the apparent trend in the data is due to chance, not the factors in the model.In this example, the trend line has a very small p-value, which is good.
R-squared values can range from 0-1 and higher values are considered better.
We see in our example that the R-squared is very high, at 0.956. This means our model fits the data very well – an R-squared of 1 indicates a perfect fit, but beware that if your R-squared is incredibly high, for example 0.999, your model may be misleading. A common indicator of an artificially high R-squared value is low degrees of freedom, or just having too many observations.

Trend Line Residuals
To decide if a trend line accurately represents the data, it's not enough to have a small p-value or a large R-squared. Not all of our points will fall on the predicted trend line. The distance from a given point to its predicted value is the error, or residual.
In a correct model, those residuals should follow a random normal distribution around the zero line when plotted against the explanatory variable. If this residual plot isn't normally distributed, it indicates there are trends in how the data fail to line up with the predicted values, which means the model isn't the best.

Constraints on Forecasting in Tableau
Forecasting has some requirements. It needs at least one date (or dimension with integer values) and one measure, at least 5 data points, and if the data is seasonal, at least two seasons worth of data. Forecasting also has some restrictions. Because of the nature of the forecasting models, forecasting cannot be done against a cube, on a view containing table calculations, totals, or subtotals, or on disaggregated measures or dimensions.

Dashboards and Stories

Dashboards are a way to present one or more views, often with filters, legends, and interactivity tying the views together. Dashboards can include sheets, text, images, and webpages. Stories in Tableau are narrated walkthroughs of one or more sheets or dashboards, for example leading the audience through a discovery you made as you were analyzing the data. Each view or dashboard in a story is called a story point.

Any view that is brought into a dashboard or story is simply a window to the underlying worksheet. For the most part, changes made on one sheet – whether it's a view or a dashboard, will carry through to the other places that content is in use.

Modifying the story's filter doesn't make changes anywhere else,
but content that has been added to a story only tracks changes so long as it is unmodified on the story.
On the scatterplot, add Market to Shape. Verify that the dashboard and the story have those shapes. We hadn't modified the shape on the story, so it continues to track those changes from the scatterplot.

Dashboards are all about interactivity.

Similarly, we can make a view act as a filter for the dashboard. Click on the view to bring up the header bar, and click the filter icon. Now if we select a header such as Second Class, all the relevant views are filtered.

Dashboard Interactivity using Actions

There are three types of actions: Highlight, Filter, or URL. Actions on dashboards have several parts. Highlight and Filter Actions have a source and target. URL actions can include field values as dynamic inputs. An action can be activated by various mouse behaviors, such as hovering or clicking, or as a menu option in the tooltip.

If there is a web page on the dashboard, the URL action opens there. If there isn't a web page in the dashboard, the URL action will open in the default browser

Device Designer

In today's technology landscape, it cannot be assumed that a dashboard will only be consumed on a standard computer. The Device Designer feature in Tableau takes a master dashboard and lets you preview, and more importantly customize, what that dashboard would look like on a desktop, tablet, or phone.

The default acts as a template for the other device layouts so they don't have to start from scratch. Only sheets that are put onto the default dashboard will be available for the device-specific layouts. You can remove sheets from a layout, but only what is on the default can be used.

What Are Story Points

Story Points let you create compelling, interactive, data-driven stories. Stories consist of specific views or dashboards in sequential progression, for example letting the audience walk through a discovery you made as you were analyzing data.

Mapping

Latitude indicates how far up or down from the equator, Longitude indicates how far east or west from the Prime Meridian.
Any point on a map can be represented with latitude and longitude coordinates. In Tableau, coordinates need to be in decimal format. Positive latitudes indicate the northern hemisphere, positive longitudes indicate eastward from the Prime Meridian.
In this way, every point on the globe has unique latitude and longitude coordinates.
Incidentally, Tableau uses same projection as Google Maps, which is Web Mercator.

On the other hand, if your data doesn't have latitude and longitude but you have geographic place names such as city, country, or province, Tableau will determine their coordinates for you provide the fields latitude (generated) and longitude (generated).

Polygon Maps
Locations can be plotted on a map in two ways – as a point or mark to represent the entire area, or a polygon, covering the area. Tableau has polygon data, or filled maps, for many geographic locations built in. It's also possible to provide your own polygon data to create custom polygon maps, such as this map of National Parks in the UK. Check out the Polygon Maps video for more information, including creating custom territories on maps.

Spacial Files
Joining or Blending Geocoded Data
If we had banking data for these locations, we
could blend or join in that data as long as there's a shared field with our Geocoding data set, such as branch name or address.
Polygon Maps
One of the big distinctions between using geocoded data directly and importing custom geocoding is the ability to use polygons, or filled shapes. If we connect to a data source with polygon information, it's easy to map custom polygons.

Creating a Custom Geocoding Import File
There are three main types of custom geocoding: Extending an existing role, adding new roles to an existing hierarchy, or adding new hierarchies. Regardless of the type of custom geocoding, the CSV import file must be created very carefully. It must have consistent spelling, capitalization, and column names. For example, Latitude and Longitude must be spelled out fully and be capitalized.

Extending an Existing Role
Extending an existing role would be the option for adding towns that Tableau doesn't automatically recognize. Here's an Excel file which adds in some small towns from my home state of Florida. I have to indicate every level of the hierarchy above the cities, so state and country, then provide the Latitude and Longitude for these new cities. Because all three levels of the hierarchy are existing roles already in Tableau, I have to name my columns exactly as Tableau knows these roles – the online help has a table with the columns to include.

Online Help: Custom Geocode Your Data https://onlinehelp.tableau.com/current/pro/desktop/en-us/help.htm#custom_geocoding.html

Filled Maps
Tableau's filled map feature creates polygon maps quickly and easily.

Required Data for Polygon Maps
Certain fields are required to create a polygon map. I like to think of it like a series of connect-the-dot drawings where each polygon area is a separate drawing. The data set has

to contain specific information for Tableau to know what lines to draw and where. Latitude and Longitude are the coordinates of each point in the polygon, or each dot in the connectthe-dot. The point order field tells Tableau which dots to connect and in what order, like the numbers on a connect-the-dot. The polygon ID field will identify individual enclosed areas, letting Tableau know which points make up each drawing. Essentially, when to lift the pencil. This can be as simple as the name of each area – the key point is that each polygon needs a unique identifier. Depending on the complexity of the polygons, it might be helpful to have sub polygon IDs as well, multiple enclosed areas can roll up to a larger area. For example, Greece has many islands, each of which would be their own polygon, but on a world map we'd want to know all those islands roll up, together with the mainland, into a single country.

ATTR(). Attribute checks to see if there is only one value for a given field for all rows in result set. If there is only one value for the data selected, Attribute returns that value. If there is more than one value for that subset of data, it returns an asterisk. Another way of thinking of the attribute function is as the equivalent to the logical test "if min equals max then return that value". We could also use MIN or MAX if we know there's only one value, though these may artificially pick single value when 3 there are many values that could be there. But all of these aggregations are useful for aggregating dimensions to get around errors like the one we saw.

LOD

An LOD expression is established by curly braces. The first piece inside the braces is the keyword, then the dimension declaration, followed by a colon, then the aggregate expression. The aggregate expression portion of an LOD Expression should be familiar. This can be anything from a simple SUM([Sales]) to a complex calculation. Note that aggregate expressions cannot contain ATTR or table calculations.

The dimension declaration identifies the dimensions that define the level of detail to use when performing the aggregation. If no dimensions are declared (either with a FIXED keyword or no keyword specified), the LOD Expression is considered Table Scoped, which is a complete aggregate of the data. The keyword can be one of three options: FIXED, INCLUDE, or EXCLUDE

Dimension or Measure

Level of Detail Expressions that use EXCLUDE or INCLUDE keywords will always result in measures. FIXED keyword expressions will default to be dimensions or measures based on the aggregate expression. If the aggregate expression would yield a measure, the overall expression will be a measure, such as Number of Orders per Customer. If the aggregate expression would yield a dimension (such as string, Boolean, or date), the overall expression will be a dimension, such as Second Purchase.

Aggregation or Replication of Results As established in previous videos, the Level of Detail of the view determines the number of marks drawn in the visualization. If a calculation has a different Level of Detail, something needs to be done to reconcile the difference. This can be seen most easily with FIXED. FIXED

level of detail expressions compute values using the specified dimensions without reference to the view's level of detail. Because a FIXED LOD Expression is independent of the view's LOD, the results may be either "below" (that is, more granular than) or "above" (that is, more aggregated than) the LOD that established the marks on the sheet. FIXED – Aggregated Results If the LOD Expression's results are more granular than the view LOD, such as in this example, the values from the LOD Expression are aggregated to create the view. Here, the LOD of the view is simply Segment, but the LOD Expression was fixed at a more granular level of both Segment and Category. The LOD Expression's values for each category are aggregated into a single value per segment to be displayed in the view as the result. If we pre-emptively wrap a level of detail expression in an aggregation when we create it, Tableau will use the aggregation specified rather than choosing one for us when that expression is placed on a shelf. Here, for example, we want the Average per Category sale, so while we summed the sales inside the LOD Expression, we want the results to be displayed as an Average. This arrow diagram illustrates the process schematically, showing the value of the LOD Expression computed at a lower level than the Viz LOD then being aggregated back up to be displayed at the Viz LOD. FIXED – Replicated Results If the dimension declaration is less granular (more aggregated) than the view, the values from the LOD Expression are replicated to create the view. Here, the LOD of the view is both Category and Segment, but the LOD Expression was fixed at a less granular level of just Segment. The LOD Expression's values for each segment are replicated for each Category within a segment to be displayed in the view. In this instance, wrapping the expression in an aggregation isn't necessary, 3 but since the view may change, it doesn't hurt to do so. When no aggregation is needed (because the expression's level of detail is coarser than the view's), the aggregation we specified is ignored. This arrow diagram illustrates the process schematically, showing the value of the LOD Expression computed at a higher level than the Viz LOD then being replicated to be displayed at the Viz LOD. FIXED – Disjoint Results A final scenario is if the LOD Expression's dimensions are entirely disjoint from the dimensions in the view. In this instance, an intermediate step is calculated behind the scenes, whereby the LOD Expression's value is replicated for each combination of LOD Expression and LOD of the view, then aggregated to the level of the view. This is also the behavior for disjoint Nested LOD expressions, which will be covered in more detail later. INCLUDE INCLUDE level of detail expressions compute values using the specified dimensions in addition to whatever dimensions in the view. The INCLUDE keyword will almost always yield LOD Expression results that are below Viz LOD and thus need to be aggregated up to be displayed in the view. Establishing what type of aggregation is performed on the LOD Expression's results can be done in the calculation or on the pill in the view. EXCLUDE EXCLUDE level of detail expressions explicitly remove dimensions from the expression –that is, they subtract dimensions from the view level of detail. The EXCLUDE keyword will almost always yield LOD Expression results that are above the Viz LOD and thus need to be duplicated to be displayed in the view. Tableau will default to show an EXCLUDE LOD Expression pill in the view as ATTR rather than SUM because there is no aggregation happening other than what is defined inside the LOD Expression's aggregate expression. Note that it is also possible to have an LOD Expression match the Viz LOD, in which case the LOD Expression will seem to be an ordinary calculation simply performing the aggregate expression. However, if the viz LOD changes, or the LOD Expression is reused in a different view, the effects of the LOD Expression can be seen.

Pill Types

Why Pill Type Matters

When we bring a field into the view from the data pane, Tableau creates a pill. Whether or not that pill is a dimension or a measure, or continuous or discrete, impacts every level of functionality in the analysis, from the way the data displays to the deeply technical, behind the scenes approach to how data is processed. Knowing the features of how Tableau works with pills is vital to staying in control of the analysis.

Dimensions and Measures

Measures are usually metrics, or numerical data, like shipping cost. Inside of Tableau, **measures are aggregations – they're aggregated up to the granularity set by the dimensions in the view. The value of a measure therefore depends on the context of the dimensions.** For example, the result for the sum of shipping cost is dependent on if we mean overall, just under a million and a half, or per Order Priority level, from 65 to about 3 to discrete".

Dimensions are usually categorical fields such as Order Priority and Region. Specifically, in Tableau, **dimensions set the granularity, or the level of detail in the view. We typically want to group our data by some combination of categories. What dimensions we use to build the view will determine how many marks we have** – here we have 4 order priorities - critical to low – and we have 4 marks.

When we hover over the pill, it's now blue, and the icon is blue as well. We can also convert it to a dimension simply by dragging it into the Dimensions area of the data pane. Note that those two operations – **converting to discrete and converting to a dimension, are not synonymous.** For the rest of this video, we'll mostly be talking about continuous and discrete data instead of measures and dimensions. If you're ever in doubt, remember: **dimensions come out onto the view as themselves, measures come out onto the view as aggregates, discrete pills are BLUE, continuous pills are GREEN.**

Axis vs Label

Let's start with axes and labels. **When a continuous pill is brought into the view, it creates an axis. This will automatically fill the entire view along that direction. When a discrete pill is brought into the view, it creates a label with panes for each value.** This will take up as much or as little room as required.

Color and Maps

Whether a pill is continuous or discrete also has an impact on default behavior for colors. **A continuous pill on the color shelf will create a gradient. A discrete pill on the color shelf will create a color palette, where each unique value is assigned a color** – here I made a copy of my Sales field and changed it to discrete, but the underlying data is identical. On a map, the default behavior depends on whether the pill is continuous or discrete, but also if it's dimension or a measure. **If the geographic type allows it, a measure on color defaults to a filled map. A dimension on color defaults to a symbol map. Whether these colors are gradients or palettes depends on if the pill is continuous or discrete.**

Date Types

Dates can be brought into the view as either continuous or discrete. If the date icon is blue in the data window, then the default when we bring out that pill will be discrete, but if we right click drag, we can select specifically what sort of date we want. We can change this property in the pill's menu – **the options at**

the top are discrete date parts, the lower section are continuous date truncations. Continuous date truncations are treated as an ongoing progression along an axis. Here, there is a single trend line, and December 2012 leads into January 2013. Discrete date 4 parts are treated like categories. Each year is a different category, and there is a trend line here per year. There is a break between December 2012 and January 2013 because month and year are simply labels.

Filtering

Filtering on a discrete pill, such as Category, brings up options related to the specific list of values for that pill. Filtering on a continuous measure pill, such as Profit, asks first if we want to filter at the row-level or aggregate level, then brings up options for continuous ranges.

Measure Names and Measure Values

Have you ever wondered what those "Measure Names" or "Measure Values" fields are? These are fields that Tableau automatically creates to enable certain types of analyses.

Text Tables

what if we want to see multiple measures instead of just one? For instance, if we want to see sales and profit for each of our regions. How would we do that?
Ok, we'll bring Region to rows, and sales and profit to text. Hmm, this isn't what I was imagining. I wanted to have my measures next to each other, in a column each like in Excel, not as two marks in the same cell, like they are here. But Tableau uses different logic than Excel**—the functionality of Measure Names and Measure values allows for greater flexibility when dealing with different data structures.**
Remember with our first text table, the columns were set by the dimension, order priority, with a column for each dimension member. To break sales and profit out into their own columns, we need a dimension. **That's what Measure Names is for—it's a dimension whose members are our measures.**
We'll remove sales and profit from text, and bring out Measure Names to Columns. It says "No Measure values"—measure names and Measure Values work hand-in-hand. **Measure names can be thought of like the labels for the measures, so we need to have values associated with those labels, too. We'll bring Measure Values to text.** By default, all measures are brought into the view, and they show up on a new "measure values" card. We can simply drag off the ones we don't want.

A Shortcut

To bring out only specific measures to the Measure Values card, we can ctrl+click the measures we want, along with a dimension, then, using Show me, select text table. This automatically brings out Measure Names and Measure Values into the appropriate places, with just the measures we selected.

Other Chart Types

Measure Names and Measure Values are useful for more than text tables. Here we have a line chart with sales broken out by segment. If we want to compare all the segments together on the same axis, we can simply move that dimension to color, and we now have all three lines together.

But what if we want to look at three measures on the same axis? We can dual axis two measures, but not three. So how can we get three measures in the same line chart?

With Measure Names and Measure values.

First, let's bring Measure Names to the filter shelf. We only want to look at Sales, Profit, and Shipping Cost, as those are measures with units of dollars. It wouldn't make sense to ut "discount", which is a percentage, on the same axis as Profit, which is dollars.

Now, we'll bring Measure Values to Rows, and Measure Names to Color. Note that we only get the three we selected in our filter. And we have three measures on the same axis!

Measure Names as Field Labels

Measure Names can also be used to provide labels for text tables. We have Region on Rows, and Sales on Text. The word "Region" appears above the regions, letting us know what those are, but there's nothing above the numbers. How can we easily display the field label for the measure?

If we bring out Measure Values and drop it on top of the sales values, Measure Values will replace Sales on Text, Measure Names will automatically filter for only Sales, and Sales will move to the Measure Values card—and more to our point, the label "Sales" appears above the column of numbers.

To come full circle, if we wanted to add more measures to this table, we could simply add them to the Measure Values card.

Tableau 201: How to Make a Heat Map

https://www.evolytics.com/blog/tableau-201-make-heat-map/

Heat maps are a visualization where marks on a chart are represented as colors. As the marks "heat up" due their higher values or density of records, a more intense color is displayed. These colors can be displayed in a matrix / crosstab, which creates a highlight table, but can also be displayed on a geographical map or even a customized image – such as a webpage used to show where users are clicking.

That being said, heat maps are defined somewhat differently in Tableau, and this post shares how to create a Tableau heat map. If you are interested in creating a traditional heat map using a custom image, see the post, How to Make Small Multiple Stadium Maps in Tableau.

To first get more specific about how Tableau defines heat map, let's take a look at the requirements to draw a heat map under Tableau's Show Me options.

"For heat maps try 1 or more dimensions and 1 or 2 measures"

This is very close to the requirements for drawing a highlight table with Show Me:

"For highlight tables try 1 or more dimensions and 1 measure"

How to Create Sparklines in Tableau
April 04, 2017 in data visualization
After being invented by Edward Tufte in 1983, sparklines gradually became a popular choice to represent time-series data after Microsoft Excel introduced sparklines in 2010. I love sparklines. They are compact, clean, easy to understand and perfect for dashboards.

What is Sparkline https://www.doingdata.org/blog/how-to-create-a-sparkline-chart-in-tableau

A sparkline is a compact trend line chart without axis that fits in a small area and shows values over time. So the key here is that sparklines show trends that means that they visualize time-series data, sparklines are line charts, they are without axis which means they do not start at zero, and they are compact which means that they are intended to fit in a small area and are mostly used for indicative purpose of the trend and changes over the period of time. Like all other charts, sparklines have their own advantages and disadvanteges, and are suitable for some visualizations more than others. We will discuss about properties and best usage of sparklines in the end section of this article.

How to Create Sparklines in Tableau
We are going to use Tableau Superstore data for representation. Creating trends over time is fairly easy with just a few drags and drops. If we play a bit with trend charts.

Step 1: Creating Sparkline

Drag Order Date to Columns
Right Click on Order Date Pill and select Month (Month Year)
Drag Sales to Rows
Right click on Sales Axis and select Edit Axis
Uncheck Include Zero
Select Independent axis rows for each row or column and click OK
Drag Segment on Rows
Hide Header for Sales Axis and Month of Order Date Axis
Step 2: Formatting Sparkline

Resize and make sparklines compact
Reduce the thickness of the sparkline

Add tool-tips
Add low and high data-points in red and green using table calculation and dual-axis
Remove borders, zero lines, grid lines
There you go: