



---

# Users Guide

*Version 4.5 August 01, 2018*

Copyright for ThoughtSpot publications. © 2018 ThoughtSpot, Inc. All rights reserved.

ThoughtSpot, Inc. 1 Palo Alto Square  
Building 1, Suite 200  
Palo Alto, CA 94306

All rights reserved. This product is protected by U.S. and international copyright and intellectual property laws. ThoughtSpot is a trademark of ThoughtSpot, Inc. in the United States and/or other jurisdictions. All other marks and names mentioned herein may be trademarks of their respective companies.

# Table of Contents

<b>Introduction.....</b>	<b>5</b>
<b>Getting Started</b>	
<b>Finding your way around .....</b>	<b>6</b>
<b>ThoughtSpot browser access .....</b>	<b>10</b>
<b>Your user preferences .....</b>	<b>12</b>
<b>Understanding privileges.....</b>	<b>15</b>
<b>Use stickers to organize.....</b>	<b>20</b>
<b>Use Search</b>	
<b>What are searches and answers?.....</b>	<b>23</b>
<b>Search basics</b>	
<b>Start a new search .....</b>	<b>25</b>
<b>Choose a data source.....</b>	<b>27</b>
<b>Search bar features .....</b>	<b>28</b>
<b>Search suggestions .....</b>	<b>30</b>
<b>Search results and column types .....</b>	<b>32</b>
<b>Last data refresh time.....</b>	<b>34</b>
<b>Work with Answers .....</b>	<b>36</b>
<b>Results that are tables .....</b>	<b>40</b>
<b>Use keywords in search</b>	
<b>Overview of keyword searchess.....</b>	<b>42</b>
<b>Time series analysis.....</b>	<b>45</b>
<b>Search using growth over time.....</b>	<b>49</b>
<b>Proximity searches "near" and "farther than".....</b>	<b>53</b>
<b>Work with filters</b>	
<b>Understand filters.....</b>	<b>55</b>
<b>Add a filter to a table.....</b>	<b>57</b>
<b>Add a filter to a chart .....</b>	<b>59</b>
<b>Create a bulk filter.....</b>	<b>61</b>
<b>Delete a filter .....</b>	<b>64</b>
<b>Filter on null, blank, or empty values.....</b>	<b>65</b>
<b>Other search actions</b>	
<b>Other search actions .....</b>	<b>67</b>
<b>Change result display options .....</b>	<b>68</b>
<b>Show underlying data .....</b>	<b>71</b>

See the search behind a result .....	73
Apply conditional formatting to a table.....	74
Download your search.....	79
Replay search.....	81
Work with charts	
Understand charts.....	83
Chart types	
Column charts .....	86
Bar charts .....	88
Line charts .....	89
Pie charts .....	90
Area charts .....	92
Scatter charts.....	94
Bubble charts .....	95
Pareto charts .....	97
Waterfall charts .....	98
Treemap charts .....	99
Line column charts .....	101
Funnel charts.....	104
Geo charts .....	105
Pivot table .....	109
Changing a chart	
Change the chart.....	115
Reorder the labels.....	116
Set the y-axis range .....	117
Hide and show values.....	118
Change chart colors .....	120
Additional chart options .....	124
Zoom into a chart.....	126
Work with formulas	
Understand formulas .....	128
Add a formula to a search .....	129
View or edit a formula in a search .....	132
Aggregate formulas	
Overview of aggregate formulas .....	134
Cumulative functions .....	137
Grouping functions .....	140

Moving functions.....	142
Filtered aggregation functions.....	146
Conversion functions .....	150
Date functions .....	151
Percent (simple number) calculations.....	154
Formula operators.....	155
Nested formulas .....	157
Formulas for chasm traps .....	158
Work with worksheets	
Understand worksheets .....	159
Save a search as a worksheet .....	161
Create a search from a worksheet .....	163
Worksheet example scenarios.....	165
Use pinboards	
Basic pinboard usage .....	166
Edit a pinboard .....	172
Pinboard filters .....	178
Schedule a pinboard job.....	181
Search actions within a pinboard .....	184
Copy a pinboard .....	185
Copy a pinboard or visualization link .....	187
Reset a pinboard or visualization .....	190
Start a slideshow.....	193
Work with data	
Understand data sources .....	195
Create and load CSV Files.....	197
Append data through the UI .....	200
View a data profile.....	201
Date and currency format settings.....	202
SpotIQ	
What is SpotIQ.....	204
101: Load and analyze data.....	206
101: Review and save insights .....	210
101: Do more with SpotIQ .....	215
Best SpotIQ Practices .....	222
Custom SpotIQ analysis .....	225
Advanced R Customizations.....	230

<b>Share your work</b>	
<b>About sharing .....</b>	<b>236</b>
<b>Share a pinboard .....</b>	<b>237</b>
<b>Share answers .....</b>	<b>239</b>
<b>Share uploaded data.....</b>	<b>241</b>
<b>Revoke access (unshare) .....</b>	<b>243</b>
<b>Slack and ThoughtSpot</b>	
<b>Slack and Spot.....</b>	<b>245</b>
<b>How to use Spot .....</b>	<b>246</b>
<b>More help and support.....</b>	<b>250</b>
<b>Keyword reference.....</b>	<b>255</b>
<b>TQL reference.....</b>	<b>262</b>
<b>tsload flag reference .....</b>	<b>272</b>
<b>tscli command reference.....</b>	<b>275</b>
<b>Date and time formats reference.....</b>	<b>295</b>
<b>Row level security rules reference .....</b>	<b>298</b>
<b>Formula function reference .....</b>	<b>307</b>
<b>Alert codes reference .....</b>	<b>319</b>
<b>User action codes reference .....</b>	<b>326</b>
<b>Error codes reference .....</b>	<b>328</b>
<b>Frequently asked questions .....</b>	<b>354</b>

# Introduction to the User Guide

This ThoughtSpot User Guide contains information on navigating and searching data with ThoughtSpot. It will assist you with starting new searches, managing your pinboards, and troubleshooting.

ThoughtSpot enables you to view and analyze your data through a search-based user interface. You can create your searches on the fly by typing them into a search bar, like you do when using an internet search engine. ThoughtSpot makes it easy to see your data, get your questions answered, create interactive graphs, and customize pinboards. You do not need to understand how the data is stored, attend days of training, or know SQL to do these things. Collaboration and security features make it easy for you to protect sensitive data and share information safely with others.

ThoughtSpot gives administrators the ability to modify data properties to meet business needs, for example by providing search synonyms for common terms, boosting the importance of a column in search results, or formatting how the data appears. So if you are not getting the answers you expect when using ThoughtSpot, check with your ThoughtSpot administrator to see if some settings may need to be changed.

- **Finding your way around**

ThoughtSpot is organized into several sections to make navigation easy. You can reach them by using the menu bar.

- **About the user profile**

The user icon lets you view your profile or sign out of ThoughtSpot.

- **Understanding privileges**

The things you can do in ThoughtSpot are determined by the privileges you have. These are set at the group level.

- **About stickers**

You can create stickers to make it easier for people to find data sources and pinboards.

# Finding your way around

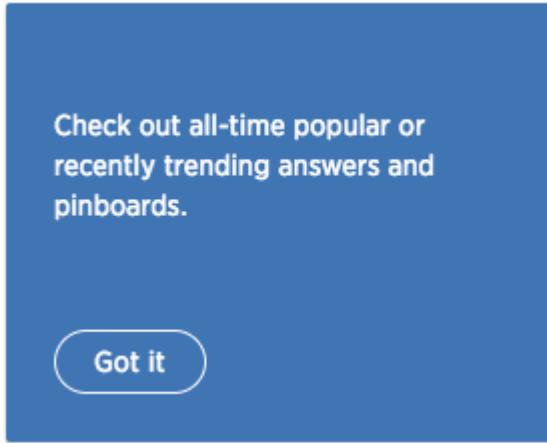
**Summary:** ThoughtSpot is organized into several sections to make navigation easy. You can reach them by using the menu bar.

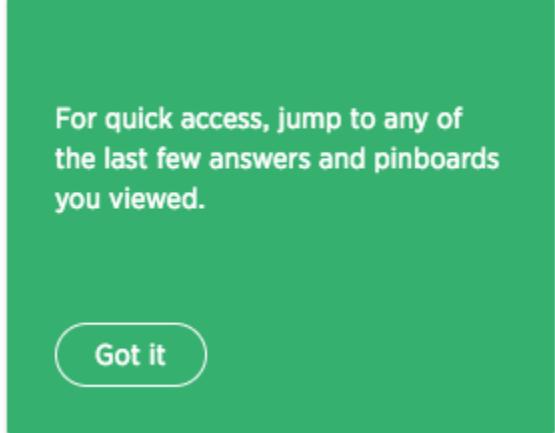
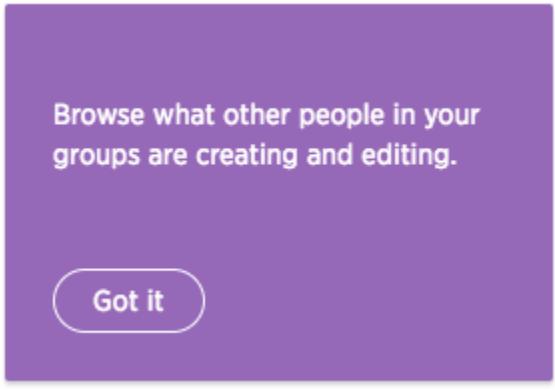
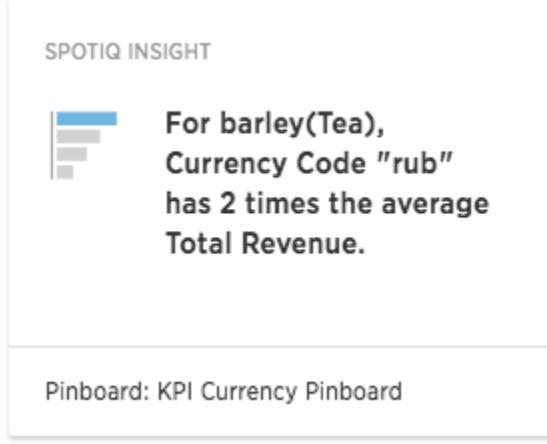
These are the different sections in ThoughtSpot:



## Home

Click the ThoughtSpot logo to go to the application home page. At the top of the home page is a search bar and below that several areas that show activity in your ThoughtSpot instance.

Keyword	Examples
All time popular Recently trending	Shows all time popular answers and pinboards by number of views. You can choose between all-time popular or recently in the last 15 days. Small icons illustrate the type of visualization you'll find when you click on an item.   <p>Check out all-time popular or recently trending answers and pinboards.</p> <p><b>Got it</b></p>

Keyword	Examples
Recently viewed	Answers and pinboards you've viewed recently.  <p>For quick access, jump to any of the last few answers and pinboards you viewed.</p> <p><b>Got it</b></p>
Recent team activity	Answers, pinboards, worksheets, and tables people in your company have created or edited recently.  <p>Browse what other people in your groups are creating and editing.</p> <p><b>Got it</b></p>
Did you know?	Contains auto analysis results from SpotIQ. Visible only to users that have the SpotIQ privilege.  <p>SPOTIQ INSIGHT</p> <p>For barley(Tea), Currency Code "rub" has 2 times the average Total Revenue.</p> <p>Pinboard: KPI Currency Pinboard</p>

The items shown in these areas are limited to answers, pinboards, worksheets, and other objects you have access to. For example, if you don't have the ability to use SpotIQ, that option does not appear.

## Search

**Search** is where you will spend most of your time. It allows you to search and explore your data. Choose your data sources and type in the search bar at the top. As you type your search, results will appear in the main part of the screen as either a table or a chart.

To learn more about this section, visit [About search](#).

## Answers

Answers are the result of a single search. You can save an answer you want to work more on later, or just keep it for your personal use. Answers are for you alone, until you share them with others.

## Pinboards

Pinboards are collections of related search results. You can create your own pinboard or add to an existing one. Once a pinboard has been saved, it can be shared with others or viewed as a slideshow. The **Pinboards** page shows a list of saved pinboards. Click on one to view, edit, or share it.

To learn more about pinboards, visit [About pinboards](#).

## Data

**DATA** contains a list of data sources (tables and worksheets). These are usually loaded and managed by your administrator. However, you may be able to import a spreadsheet (Excel or CSV) here, if you have the correct privileges.

To learn more about this section, visit [Working with data](#).

## Admin

**Admin** only appears if you have administrator privileges. This section is covered in the ThoughtSpot Administrator Guide.

## Help

**Help** is a support resource for ThoughtSpot. It contains short videos, a keyword reference, links to documentation, and other useful materials. You can also find the support contact information and software version number here.

## User

The **user icon** allows you to either view your preferences or log out. You can also change your icon here.

To learn more about this section, visit [About the user profile](#).

## Related information

[Set your ThoughtSpot locale](#)

# ThoughtSpot browser access

To set up and explore your data, access the ThoughtSpot application from a standard Web browser using a username and password. Before accessing ThoughtSpot, you need:

- The Web address (IP address or server name) for ThoughtSpot.
- A network connection.
- A Web browser.

A username and password for ThoughtSpot.

## Supported Web browsers

The following browsers are verified to work well with the ThoughtSpot application:

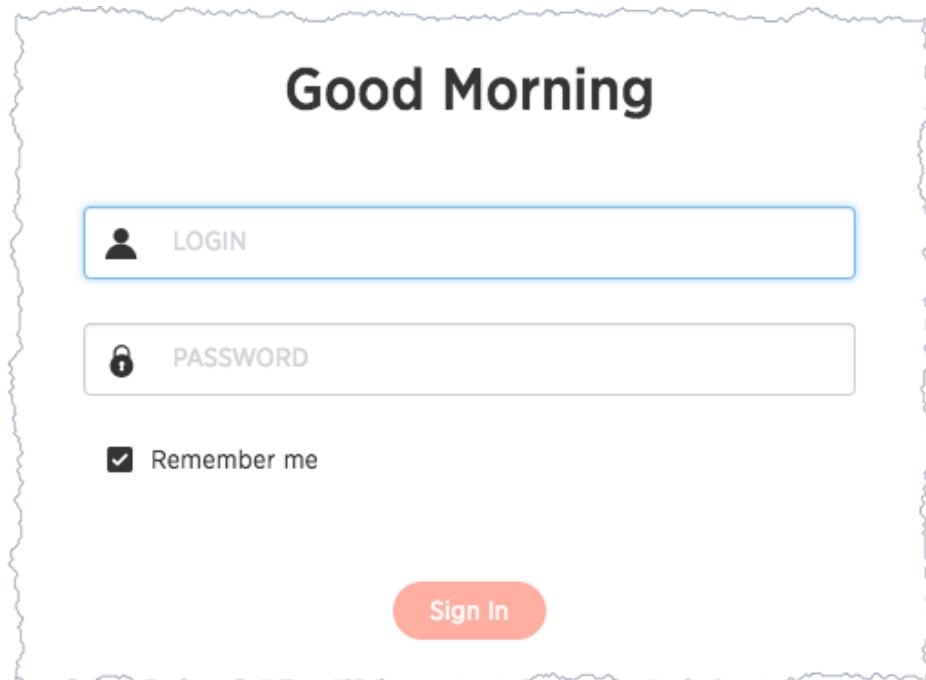
Browser	Version	Operating System
Google Chrome	20 and above	Windows 7 or greater, Linux, MacOS
Mozilla Firefox	14 and above	Windows 7 or greater, Linux, MacOS
Internet Explorer	11	Windows 7 or greater

**☒ Tip:** While Internet Explorer is supported, using it is not recommended. Depending on your environment, you can experience performance or UI issues when using IE.

## Log in

To log in to ThoughtSpot from a browser:

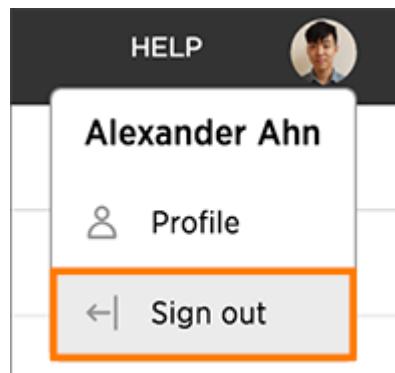
1. Open the browser and type in the Web address for ThoughtSpot: [http://HOSTNAME\\_OR\\_IP](http://HOSTNAME_OR_IP)
2. Enter your username and password and click **Sign In**.



## Log out

Once you're done with your search session, you can optionally log out of ThoughtSpot. To log out of ThoughtSpot from a browser:

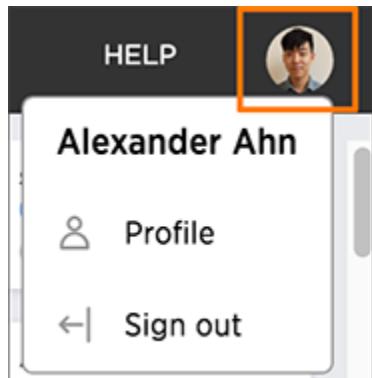
1. Click your user icon at the top right hand corner of the screen.
2. Click **Sign out**.



# About your user profile

**Summary:** The user icon lets you view your profile or sign out of ThoughtSpot.

To view the user actions, click on your user icon on the top right corner of the page.



Click **Profile** to go to your profile preference page, where you can change your icon, email, and locale preferences.

## Language and data format settings

The language the ThoughtSpot UX displays is based off of the locale in a user's profile. The locale preferences control the language and data formats (date and number formats) by geographic locations. In addition to American English (*en-US*), ThoughtSpot supports:

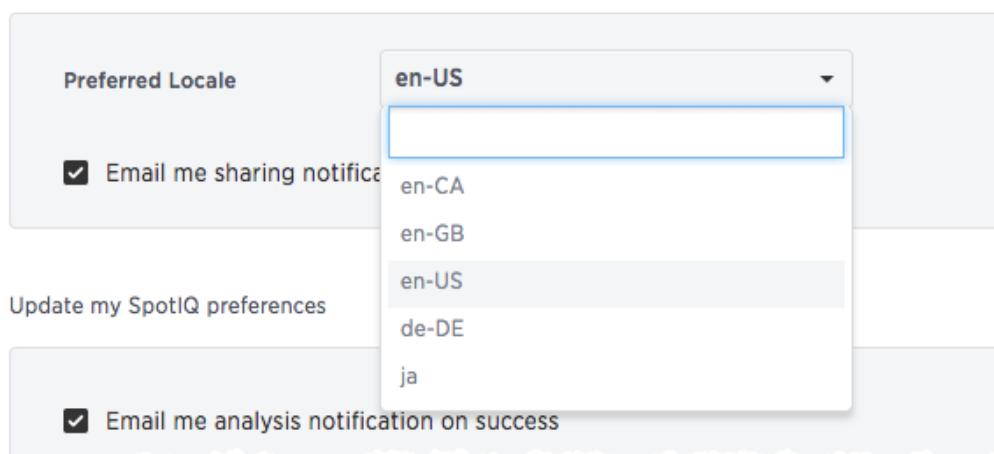
Locale	Language
<i>de-DE</i>	German
<i>en-CA</i>	Canadian English
<i>en-GB</i>	United Kingdom English
<i>en-US</i>	English (United States)
<i>es-US (beta)</i>	Spanish (United States)
<i>fr-CA (beta)</i>	French (Canada)
<i>fr-FR (beta)</i>	French (France)
<i>ja</i>	Japanese
<i>pt-BR (beta)</i>	Portuguese (Brazil)

Locale	Language
zh-CN (beta)	Chinese (S)

Date and number formats change to reflect your locale. So, if you set Japanese as your default locale in your profile settings, then the interface will update to reflect that after you refresh your page.

Keywords, operators, and error messages are included in the translated material. Formulas, however, are *not translated*. Also, all metadata remains as user inputted.

[Update my preferences](#)



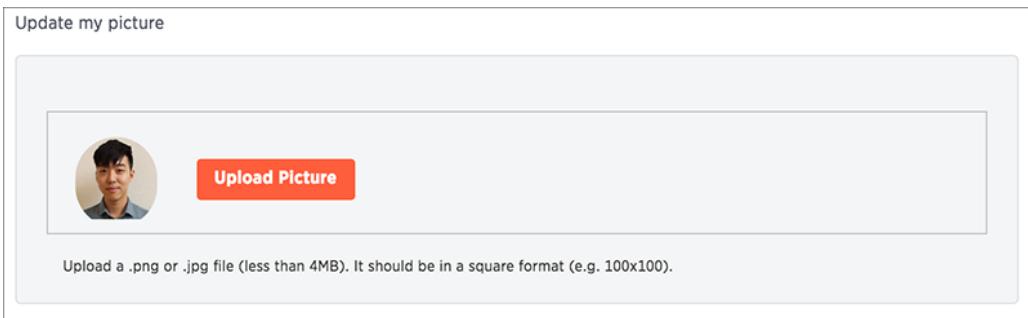
For example, if you are using ThoughtSpot in the US, the number formatting should look like this:

xxx,xxx.xx . And in Europe, it should look like this: xxx,xxx,xx .

**⚠ Warning:** It is possible that you set your locale but find strings in the UI still appear in English, this indicates an untranslated string. Please notify ThoughtSpot support.

## How other users see you

You can change your picture by clicking **Upload Picture**. The system accepts **GIF**, **BMP**, **JPEG**, and **PNG** files that do not exceed 4MB in size.



## Get email notifications

You can change your email notifications preferences by checking or unchecking **Email me sharing notifications** and clicking **Update Preferences**.

Update my preferences

Email me sharing notifications

**Update Preferences**

Clicking **Sign out** logs you out of ThoughtSpot, and takes you back to the sign in page.

# Understanding privileges

**Summary:** The things you can do in ThoughtSpot are determined by the privileges you have. Privileges are granted through group membership.

If you are trying to do something in ThoughtSpot, and cannot access the screens to accomplish it, you may not have the correct privileges. In this case, you should contact your administrator and explain what you want to accomplish. Your administrator may be able to grant you additional privileges.

**Note:** Permissions to see and edit tables, worksheets, and pinboards are not affected by privileges. They are given when these items are shared with you.

Here are the privileges that the administrator sets, and the capabilities they enable:

Privilege	Description
<b>Has administration privileges</b>	Can manage Users and Groups and has view and edit access to all data. Users with this privilege can also download a saved answer.
<b>Can upload user data</b>	Can upload their own data from the application's <b>DATA</b> page using <b>Actions &gt; Upload data</b> .
<b>Can download data</b>	Can download data from search results and pinboards.
<b>Can share with all users</b>	Can see the names of and share with users outside of the groups the user belongs to. Members of groups with this privilege can also share with groups marked as <b>NOT SHAREABLE</b> .
<b>Can manage data</b>	Can create a worksheet. Can also create an aggregated worksheet from the results of a search by selecting <b>Save as worksheet</b> . Can also use ThoughtSpot Data Connect, if it is enabled on your cluster.
<b>Can schedule pin-boards</b>	Can create pinboard schedules and edit their own scheduled jobs.
<b>Has Spot IQ privilege</b>	User can use SpotIQ's auto analyze function.
<b>Can Administer and Bypass RLS</b>	Users in groups with this privilege (directly or via group inheritance): <ul style="list-style-type: none"><li>• Are exempt from row-level security (RLS) rules.</li><li>• Can add/edit/delete existing RLS rules.</li><li>• Can check or uncheck Bypass RLS on a worksheet.</li></ul> Your installation configuration may enable or disable the availability of this privilege. By default, it is enabled. Administrators or groups with the <b>Has administration privilege</b> can grant this privilege.

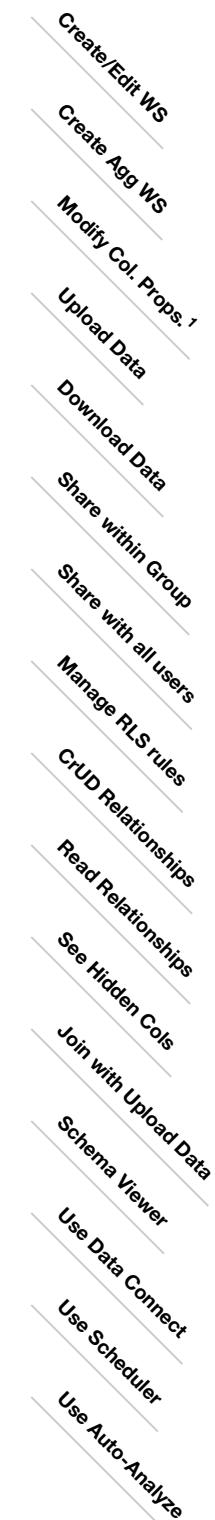
The following table shows the intersection of user privilege and ability:

The following table shows the intersection of user privilege and ability:

	Create/Edit WS	Create Agg WS	Modify Col. Props. <sup>1</sup>	Upload Data	Download Data	Share within Group	Share with all users	Manage RLS rules	CrUD Relationships	Read Relationships	See Hidden Cols	Join with Upload Data	Schema Viewer	Use Data Connect	Use Scheduler	Use Auto-Analyze
Has administration privilege	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Can upload user data	N	N	N	Y	N	Y	N	N	Y <sup>3</sup>	Y <sup>4</sup>	N	N	N	N	N	N
Can download data	N	N	N	Y	Y	N	N	N	Y <sup>4</sup>	N	N	N	N	N	N	N
Can manage data	Y	Y	Y	Y	N	Y	N	N	Y <sup>4</sup>	Y <sup>4</sup>	Y <sup>5</sup>	Y	N	Y	N	N
Can Auto-Analyze	N	N	N	N	N	N	N	N	Y <sup>4</sup>	N	N	N	N	N	Y	N
Can schedule pinboards	N	N	N	N	N	N	N	N	Y <sup>4</sup>	N	N	N	N	N	Y	N
Can share with all users	N	N	N	N	N	Y	Y	N	Y <sup>4</sup>	N	N	N	N	N	N	N
Can Administer and Bypass RLS	N	N	N	N	N	N	N	Y	N	N	N	N	N	N	N	N
None	N	N	N	N	N	Y	N	N	Y <sup>4</sup>	N	N	N	N	N	N	N

## Table notes:

1. Applies to non-owners only.
2. Any tables.
3. Author of at least one table in relationship.
4. Only when read permission for columns used in the relationship.
5. With edit permission.





# Use stickers to organize

**Summary:** Stickers enable you to create categories for classification of objects, including pinboards, answers, data sources, and worksheets.

You can create stickers to make it easier for people to find data sources and pinboards. Stickers are global in scope. This means that everyone can see the stickers and use them to tag objects. They can also filter lists of objects by sticker. Stickers are often used to designate subject areas, such as sales, HR, and finance, but you can use them any way you like.

Keep in mind these permissions when working with stickers:

- Only administrators can create stickers.
- Anyone can apply a sticker.
- Anyone can filter by a sticker.

## Create a sticker

Only administrator users can create stickers. Anyone can apply the stickers you create, or use them as filters when selecting from a list of sources or pinboards.

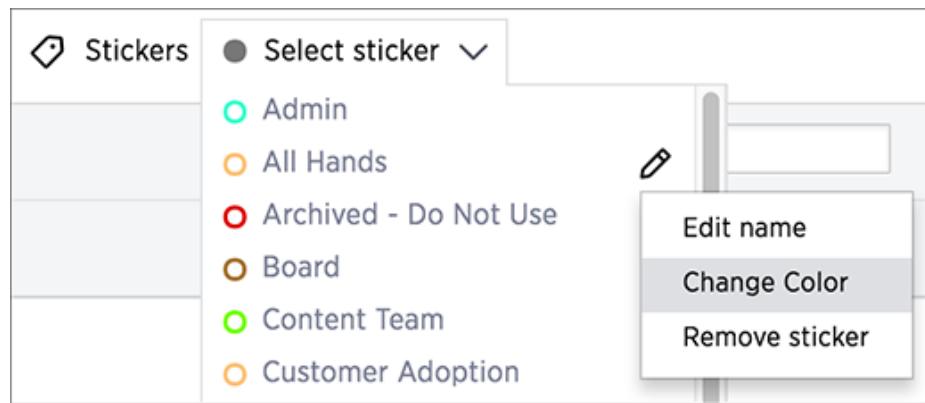
To create a sticker:

1. Navigate to the **Manage Data** or **Pinboards** screen using the icons in the top navigation bar.
2. Choose **Select sticker**, scroll to the bottom of the list, and click **+ Add**.

The screenshot shows a list of existing stickers on the left and a 'Select sticker' dropdown menu on the right. The dropdown menu contains a list of 20 stickers, each with a color-coded circle and a name. At the bottom of the dropdown is a blue 'Add' button with a plus sign. The list includes: Admin (teal), All Hands (orange), Archived - Do Not Use (red), Board (brown), Content Team (green), Customer Adoption (yellow), Customer Service (purple), Customer Usage (light orange), Demo (orange), Engineering (blue), IMS (yellow), Marketing (pink), Olympics (yellow), PM (yellow-green), and QA Testing (pink).

NAME	DESCRIPTION
Olympics Medals Analysis	
Expansion reviews	
GEO Area Maps	
Sparta - Dogfood	
SERVICE - Production Alerts	
3.4 Release Management	

3. Type the name for the new sticker.
4. You can change the name or color of a sticker by clicking the edit icon next to its name.



## Apply a sticker

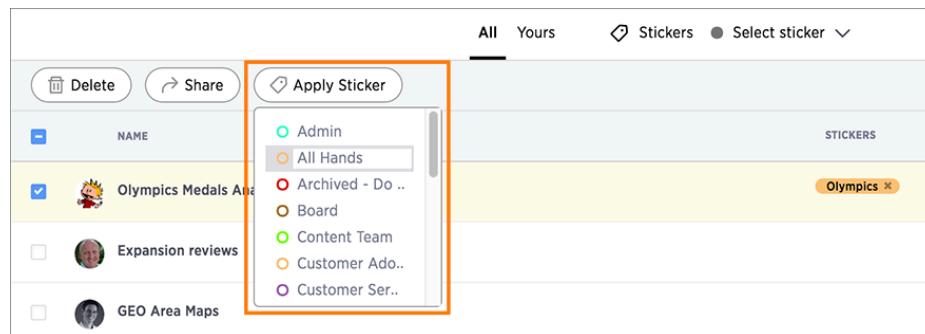
Only administrators create stickers, but anyone with edit privileges can tag an object with a sticker.

To tag an object with a sticker:

1. From the top menu, choose Answers, Pinboards, or Data.



2. Find the item(s) you want to tag in the list, and check the box next to its name.
3. Click the apply sticker icon and choose one from the list. You can apply as many stickers as you like to an object.



## Filter by stickers

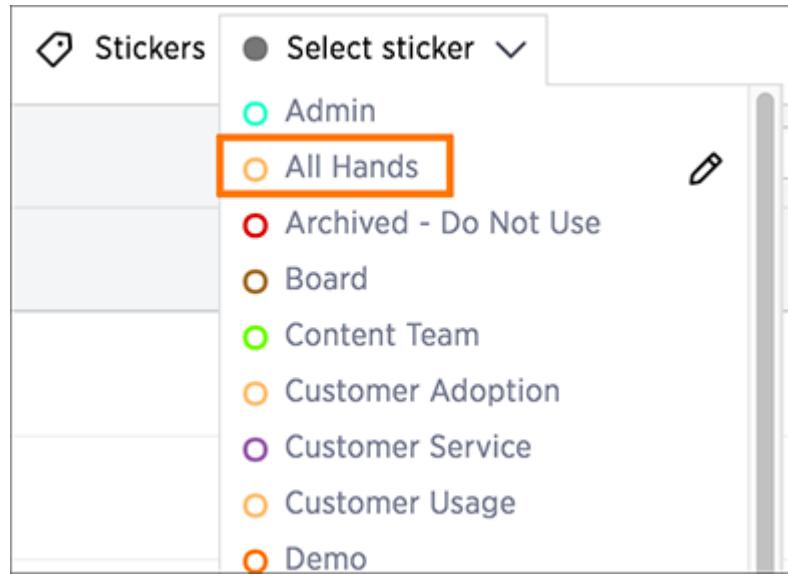
Whenever you are selecting objects from a list, you can filter by sticker to find what you're looking for. Anyone can use stickers to filter lists of pinboards or data sources. You can also filter by sticker when selecting data sources.

To filter by sticker:

1. From the top menu, choose **Answers**, **Pinboards**, or **Data**.



2. Click on **Select sticker**, and select a sticker to filter by. Click on its name.



# What are Searches and Answers?

**Summary:** You use search answer questions about your data without having to consult a data analyst.

Using ThoughtSpot's relational search is simple, so anyone can use it. In the search bar, type what you are interested in exploring, for example `revenue midwest sales rep`. Searches return a set of results in the form of a table or a chart. ThoughtSpot likes to call this set of results in response to a search an *answer*.

As you get better with ThoughtSpot's search, you will be able to get more out of your data by performing more complex searches. There are a few basic things you should understand before starting a new ThoughtSpot search.

## How do I search data?

You've probably seen one of these before:



Click in the box and start typing some letters. As soon as you begin typing, ThoughtSpot suggests some search terms Type slowly and use the suggestions to find what you're looking for.

Whenever you finish typing a word, you'll see an answer in the form of a chart or a table.

You should know that search in ThoughtSpot is more like an Amazon search than a conversation. For example. Instead of:

Find me all books by Lewis Carroll with the title Alice in Wonderland

You'd type:

carroll alice

Or consider Google. You wouldn't type in:

Find me the largest city by population

You'd type:

```
largest city by population
```

or simply:

```
largest city population
```

That's how search works in ThoughtSpot. You can use some helping words like "by" and "for" but they can also be left out, and the search will return the same answer.

## What kinds of things can I type?

Search is based on the tables that exist in your data. Tables are made of rows and columns, like spreadsheets. So you can search by typing in any of these words:

- The column name: like revenue, product name, or store
- Any of the values in the columns: like 20000, kitten chow, or richmond
- One of the special [keywords](#) ThoughtSpot understands: like yesterday, >, or contains

## Save an answer

An answer is the result of a single search. You can save an answer you want to work more on later, or just keep it for your personal use. You can also share an answer with other users. The **Answers** tab is where you can view and then the your saved search results.

## Related information

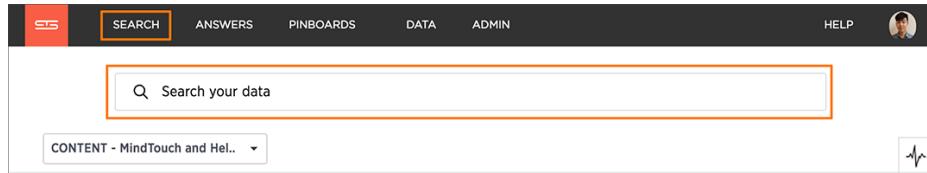
- [Choose a data source](#)
- [Search bar features](#)
- [Search suggestions](#)
- [Search results and column types](#)
- [Share an answer](#)

# Start a new search

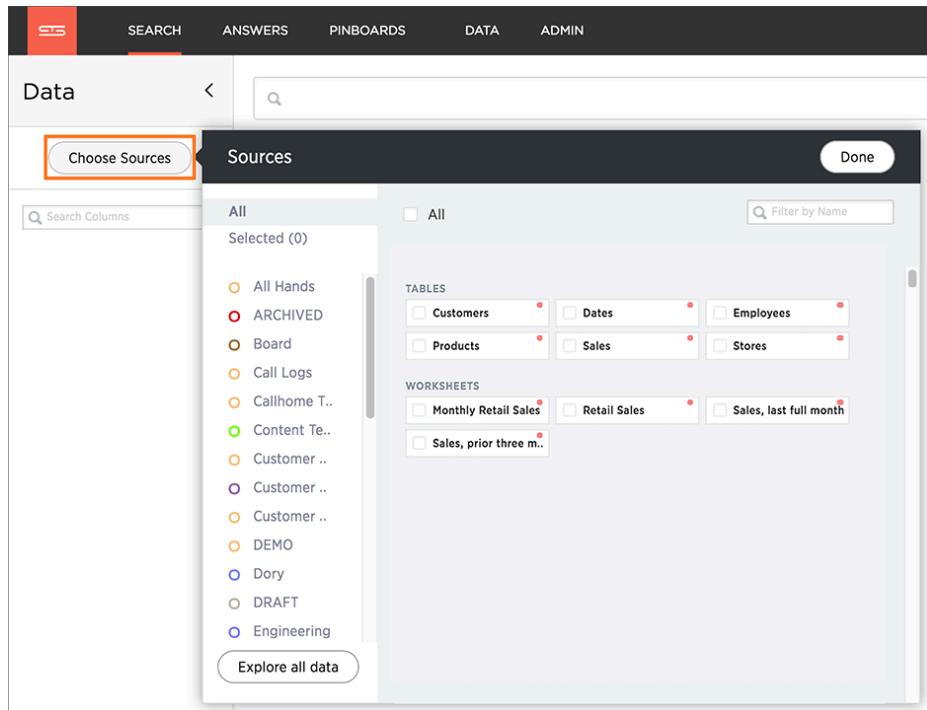
**Summary:** Starting a new ThoughtSpot search is simple, like starting a new Google search.

To start a new search:

1. Click **Search** on the top navigation bar. You can also click **Search your data** at the top of the home page.



2. Choose your sources by clicking **Choose Sources**.



To see details of all of the data, click **Explore all data**.

- a. Filter through all available sources by using the search bar or stickers.
  - b. Select your sources.
  - c. Click **Done**.
3. Add columns to the search bar, or double click columns in the Data column in the left panel.

The columns listed in the left panel are grouped together by data sources for discoverability and ease of access. You can also add multiple columns at once by clicking each column to select it, and then click + **Add Columns**.

The screenshot shows the ThoughtSpot Data interface. At the top, there's a navigation bar with tabs: SEARCH (highlighted in orange), ANSWERS, PINBOARDS, DATA, and ADMIN. To the right of the navigation bar are links for HELP and a user profile icon. Below the navigation bar, the word "Data" is displayed, followed by a back arrow and a search bar containing a magnifying glass icon.

On the left side, there's a sidebar titled "Choose Sources". It contains a search input field labeled "Search Columns" with a magnifying glass icon. Below it is a list of columns from the "fruit\_for\_help" dataset, each with a small icon and a dropdown arrow:

- Date
- Fruit (selected, highlighted in grey)
- Location
- Price per fruit (\$)
- Quantity sold
- Total sale
- Vendor

At the bottom of the sidebar is a button labeled "+ Add Columns" with a plus sign icon, which is outlined in orange. To the right of the sidebar, the main area displays the "fruit\_sales" section of the dataset. It features a logo consisting of two interlocking 'S' shapes and the text "Search your data".

Your search is given an automatic title based on your search columns, and is displayed as either a table or chart, depending on how it is best represented. You can [change the view](#) to fit your needs.

# Choose a data source

**Summary:** Before you start a new search, make sure you have chosen the right data sources.

To begin a new search, you must first select your data sources by clicking on the **Choose Sources** button. You will see a list of data sources that have been shared with you. The data sources are usually created by your administrator, though you can also [upload your own data](#).

The screenshot shows the ThoughtSpot interface with a dark header bar containing the logo, SEARCH, ANSWERS, PINBOARDS, DATA, and ADMIN buttons. Below the header is a navigation bar with 'Data' selected. A modal window titled 'Sources' is open, featuring a sidebar on the left with a 'Choose Sources' button highlighted by a red box. The main area displays a list of data sources under 'All' and 'Selected (0)'. The 'Selected (0)' section contains several colored circular icons followed by truncated names. The 'All' section lists various tables and worksheets, each with a checkbox and a red dot icon. A 'Filter by Name' search bar is at the top right of the main area. At the bottom of the sidebar is an 'Explore all data' button.

Searches happen in the context of the selected data sources. ThoughtSpot will also make suggestions from other data sources for you, if what you're typing can't be found in the selected data source.

# Search bar features

**Summary:** The search bar is designed to make it easy for you to identify your search terms.

A lot of work has gone into making ThoughtSpot's search bar intuitive and easy to use. However, it still helps to know some details of how the search bar works.

## Boxed search phrases

The search bar shows boxes around each search phrase, so you can easily see where it begins and ends. Your search phrases still appear as text when you are typing, but whenever you click out of the search bar, they are boxed. Search phrases have blue boxes, and [filters](#) have white boxes.



## Removing search phrases

When you click on a search phrase, it is highlighted, so you can easily replace it with one of the suggestions. When you hover over on a boxed phrase, you'll see an **x**, which you can click to remove it. When you delete a phrase, your search will automatically update.

If you find yourself looking at a table or chart, but it doesn't seem to contain all the data you expect, try looking in the search bar for white boxes (filters). If you remove all the filters, you will again see all the available data for that search.

## Spell check

The search bar includes spell check. If you spell or type a term incorrectly, your suggestions will include the correct spelling of the term or keyword.

The spell check in the search bar also performs a metaphone check for similar sounding words on text data values.

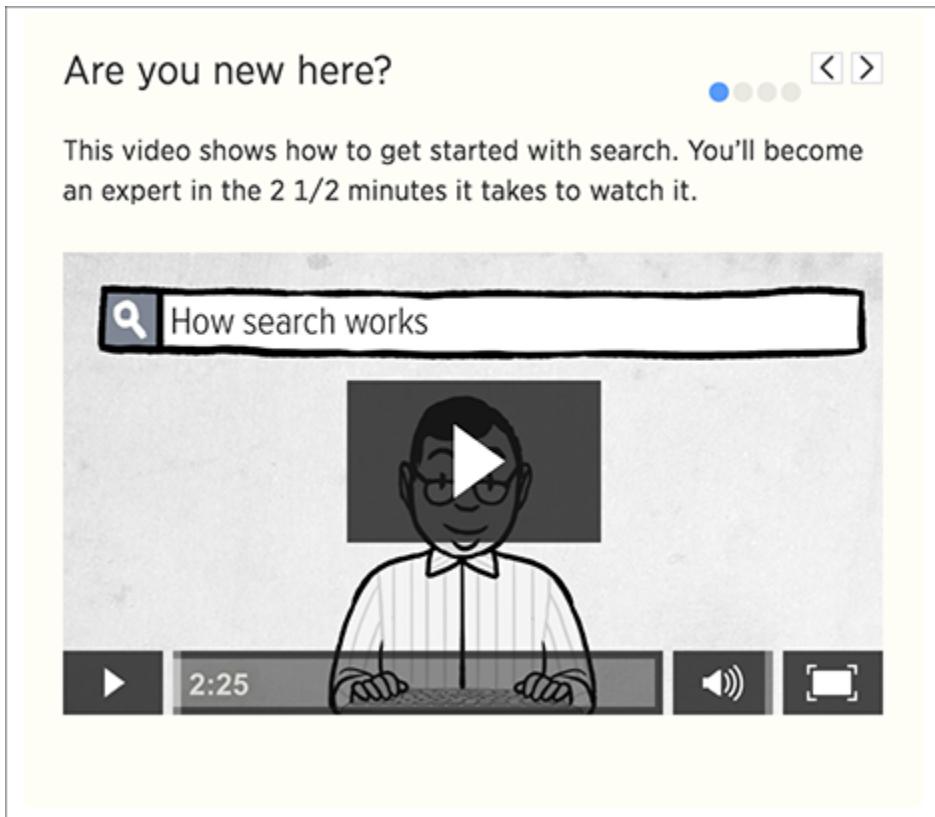
## Dictionary synonyms

A dictionary of common word synonyms is bundled with ThoughtSpot. The dictionary contains synonyms for common terms that occur in data. For example, if you type in “gross”, and that word is not found, ThoughtSpot will consider it a synonym for “revenue”. We use the WordNet library of terms, which takes word proximity into account, and weights words by similarity to the target term.

This frees you from having to know the exact term or column name used in a data source. You can still find the answer in many cases, if you type a word with the same meaning as a different word that occurs in the data.

## Search help

If you type an unrecognized search term, you will be offered tips on searching. The search help appears when you type a term that isn't understood by ThoughtSpot and then press the Enter key.



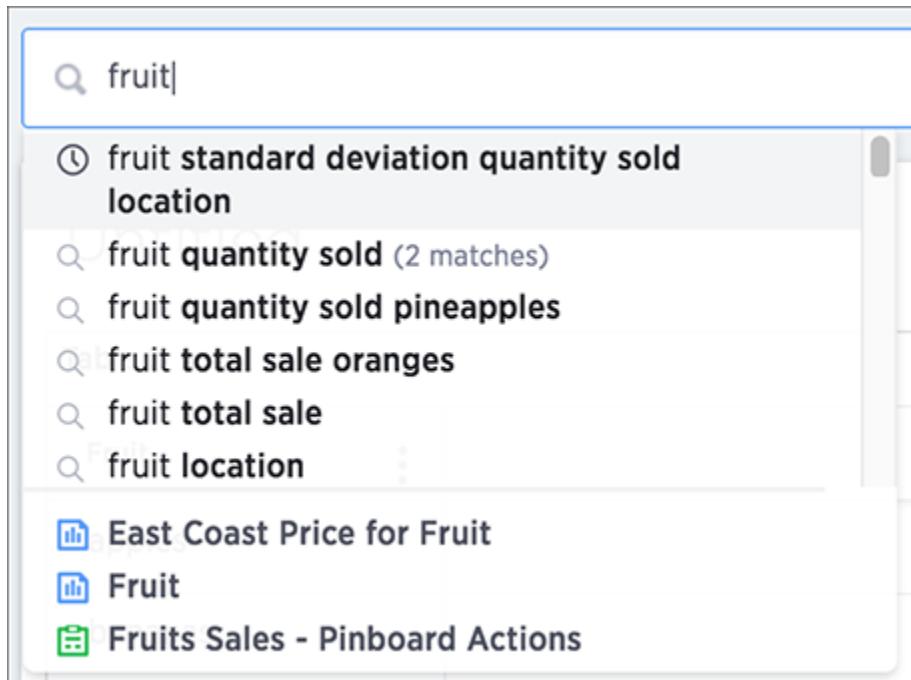
## Related searches

At the end of the **Search** page is the system provides searches you might find useful. These are quick searches you can load into the search bar by simply click on the provided link.

# Search suggestions

**Summary:** How ThoughtSpot learns from your searches and helps you.

Search suggestions include complete recent searches that are similar to the search you are constructing. You can also search for saved objects, such as pinboards.



## Usage-based ranking (UBR)

Search suggestions are relevant to the data and personalized to your search behavior. ThoughtSpot learns over time what columns are most important to you and to your company as a whole. Then, it uses this knowledge to rank the search terms it offers. The term for this is usage-based ranking (UBR).

ThoughtSpot keeps usage statistics on frequency of search terms in its local cache. If you frequently use type terms related to finance or to a particular product, ThoughtSpot provides you with related suggestions more frequently. Using this information, frequently used terms and phrases are offered in search suggestions more often than those that are not commonly used.

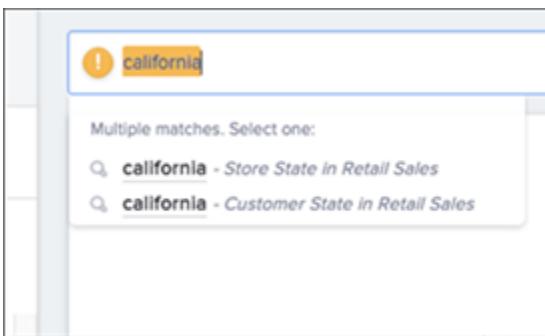
UBR based on search patterns makes ThoughtSpot more valuable over time. Search suggestions become more helpful the more searches you complete.

## Recent searches

As you type in the search bar, you are given search suggestions that include recent searches. This makes it easier to learn what you can do with ThoughtSpot. In addition to displaying your own recent searches, the search bar also learns from searches made by other people. This should provide you with interesting searches that you may not have been aware of. You'll see recent searches in a separate section of search suggestions, as recent searches appear higher in suggestions than other terms.

## Auto-disambiguation in search

When there are more than one possible meaning for a search term, you can provide disambiguation by selecting from a list of choices. Now, your choice is sticky. That means you won't have to select it again, in the scope of the current search.



## Out of scope columns

Search suggestions are not limited in scope to the columns that you select as the data sources. Suggestions can include columns that are out of scope, too. If you choose one of these suggestions, the columns will be added to the data sources scope for you.

## Object searches

Search suggestions also include relevant pinboards that have already been created. This means that if a pinboard that is similar to the search you're trying to do already exists, it will appear in search suggestions.

# Search results and column types

**Summary:** Searches in ThoughtSpot use columns that are defined as either attributes or measures.

ThoughtSpot identifies search columns (`sales`, `customer_name`, `total_sale`) as either attributes or measures. The columns you choose impact your search results. That's because several chart types expect that your search contains a certain number of attributes and a certain number of measures.

## Attributes

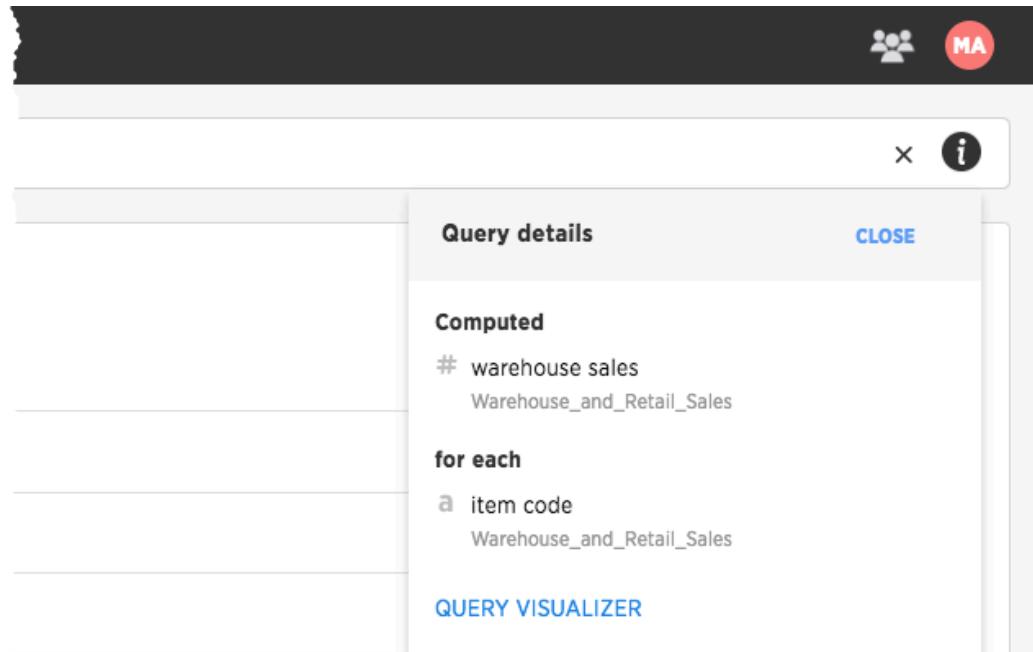
Attribute are primarily text or date values. Attributes make up the x-axis of your chart most of time. Some examples of attributes in terms of a person include name, eye color, occupation, social security number, address, employee ID, and phone number.

## Measures

Measures are numeric values you can do math on, with meaningful results. You will most likely find your measures on the y-axis of your chart. Some examples of measures in terms of a person are age, height, and weight.

## Query details

After, or while completing a search, you can see how ThoughtSpot displayed the answer. Click the **exclamation mark** on the right hand side of the search bar to open the **Query details** dialogue box.



This dialog shows measures ThoughtSpot computed, for each combination of attributes. Choose **QUERY VISUALIZER** to see search filters and how the tables were linked and used.

# Last data refresh time

**Summary:** You can see the last time at which data was refreshed without having to visit the \*\*DATA\*\* page.

Hover over the data when choosing a data source to see the last time it was updated. You will also see when it was created and by whom.

The screenshot shows the 'Sources' interface. On the left, there's a sidebar with categories like All, ARCHIVED, Board, Call Logs, etc., and a 'Content Team' section which is currently selected. In the main area, under 'IMPORTED DATA', there are several data sources listed with small checkboxes next to them. A tooltip is displayed over the checkbox for 'MindTouch\_most\_popular\_pages'. The tooltip contains the following information:

- NAME: Udemy\_user\_course..
- AUTHOR: Alicia
- CREATED: 9 months ago
- UPDATED: 1 month ago
- DATABASE: FalconUserDataDataBase
- SCHEMA: FalconUserDataSchema

You can also hover over a column in the **DATA** column to see when it was last refreshed. This popup will also show sample values from that column.

The screenshot shows the 'Data' interface. On the left, there's a sidebar with 'Choose Sources' and a search bar. Below it, a list of data sources is shown, with 'Last searched' checked. A tooltip is displayed over the 'Last searched' entry. The tooltip contains the following information:

- NAME: Search query
- SOURCE: searched
- MindTouch\_searches\_last\_90\_days
- UPDATED: a month ago
- 06/22/FY 2017 09 AM
- SAMPLE VALUES
- filters pinboard 2017 09 AM
- filter pinboard
- filters 06/22/FY 2017 09 AM

And lastly, you can hover over a column or column name in an answer to see last updated information.

Most Recently Searched

Search query (Select) ▾

Most Recently Searched table

Last searched ▾ HOURLY	Search query
06/22/FY 2017 07 AM	SOURCE: Search query (MindTouch_searches_last_90_days) UPDATED: a month ago
06/22/FY 2017 07 AM	date serach

# Work with Answers

**Summary:** Answers are the saved results of a search.

You can save the results of any search and share these with other people. Your saved searches appear on the **Answers** page. From this page, you can see answers you saved and answers other people have saved, provided these people shared them with you or a group you belong to.

## Apply a sticker

Only administrators create stickers, but anyone with edit privileges can tag an object with a sticker. To add a sticker to an answer:

1. Find the item(s) you want to tag in the list.
2. Check the box next to its name.
3. Click the apply sticker icon and choose one from the list.

You can apply as many stickers as you like to an object.

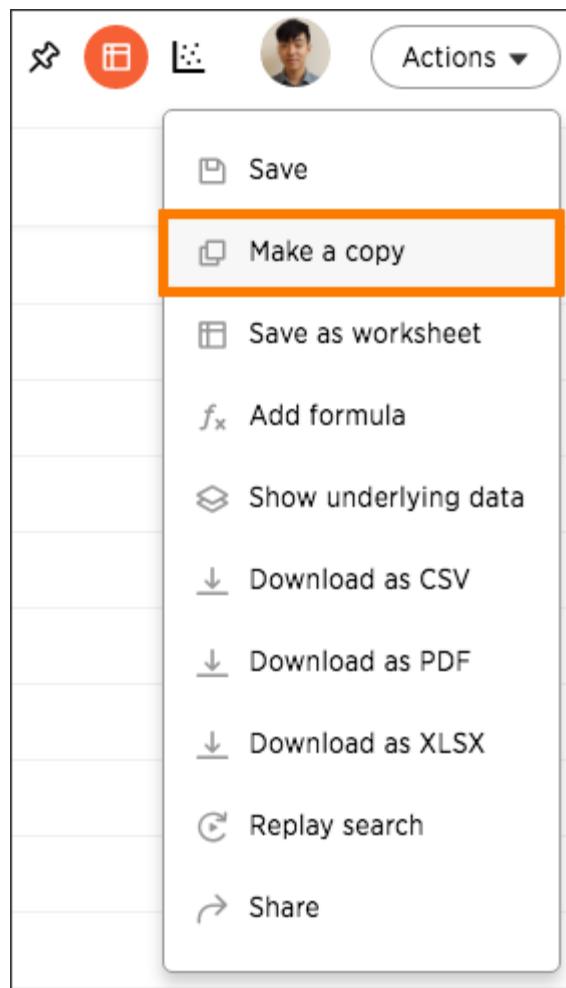
## Copy an answer

You can make a copy of an answer if you would like to make edits without changing the original answer. When copying, you can type in a different name for your copy.

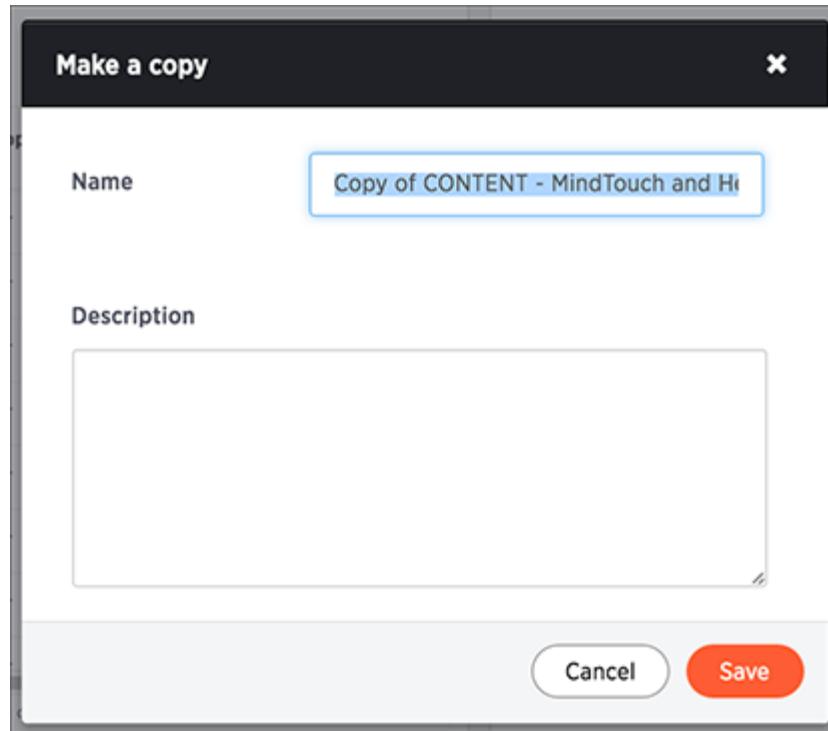
1. Click on **Answers**, on the top navigation bar.



2. On the answer list page, click the answer you would like to copy.
3. Click **Actions** and select **Make a copy**.



4. Give your answer a new name and description. Then click **Save**.



## Other features

At the bottom of the **Answers** page, you can find two additional areas, **SpotIQ Insights** and **Related Searches**. These areas are derived from your original answer and you can select them to deepen your search.

A screenshot of the ThoughtSpot interface. At the top, there's a navigation bar with 'Search', 'Answers', 'Pinboards', 'SpotIQ', 'Data', and 'Admin'. On the far right, there are user icons. Below the navigation bar, there's a chart titled 'Total Revenue by Color' with a legend showing 'magenta - 252M (9.07%)'. The main area is divided into sections: 'SpotIQ Insights' and 'Related Searches'. The 'SpotIQ Insights' section contains four cards, each with a bar chart icon and text: 'Total Revenue by Lineorder PartKey' (For olive(Color), Lineorder PartKey "155942" has significantly higher Total Revenue out of 54 values. Score: 0.27125), 'Total Revenue by Lineorder PartKey' (For cyan(Color), Lineorder PartKey "66877" has significantly higher Total Revenue out of 80 values. Score: 0.0986083), 'Total Revenue by Lineorder PartKey' (For burnished(Color), Lineorder PartKey "181577" has significantly higher Total Revenue out of 61 values. Score: 0.055599..), and 'Total Revenue by Lineorder PartKey' (For lace(Color), Lineorder PartKey "166759" has significantly higher Total Revenue out of 52 values. Score: 0.0536317). The 'Related Searches' section lists several search terms: 'Total Revenue by Color, Lineorder PartKey', 'Total Revenue by Color, OrderKey', 'Total Revenue by Color, Brand1', 'Total Revenue by Color, Year Month Number', 'Total Revenue by Color, Lineorder Supkey', 'Total Revenue by Color, Lineorder CustKey', 'Total Revenue by Color, Type', and 'Total Revenue by Color, Year Month'.

The **SpotIQ Insights** section is only visible to users with the **Has SpotIQ** privilege. Once you expand an insight on this list, an **Edit** button allows you modify the insight and make it your own. Your installation configuration determines how many insights can appear in this section. The default is 6.

# Results that are tables

**Summary:** Tables display your answer in a format similar to an Excel spreadsheet.

Your search results are not limited by the number of attributes or columns in order to be presented as a table. You can have a table with just one attribute or measure. When you choose to display your answer as a table, ThoughtSpot will create the columns for you and any relevant headlines.

Sometimes when you view a table, the results will be aggregated (combined). For example, if you only type “revenue”, you’ll see the total sum of all the revenue the table contains as one combined number. If you include the keyword **monthly**, the results will be aggregated by month. You can rearrange the column order of your table among other search actions.

Every table gives you the option to rearrange the column order and change the column widths.

## Rearrange column order

You can rearrange the column order of your table after adding all of your search terms.

To rearrange the column order:

1. While viewing your answer as a table, click the column header you would like to move.
2. Drag it across to its new position.

## Resize column widths

You can resize the column widths of your table after adding all of your search terms. Any adjustments you make to the column widths of your table are saved when you pin the table to a pinboard. To resize the column widths:

1. While viewing your answer as a table, hover over a column border in the column header row.
2. Click and drag the border to create your preferred column width.

## Sort columns

You can sort a table by column values by clicking on the column title. If you hold down the SHIFT key you can click on multiple heads and sort on them in turn.

**Tip:** This same functionality is available on tables you see elsewhere in ThoughtSpot. For example, a table in the **DATA** page is also sortable in this manner.

## About headlines (summary information)

Headlines display summary information of a table result. Headlines contain summary information for each column of a table. They appear at the bottom of the table in individual boxes.

Top Clicked Search Result for Frequent Searches					
Search query	(Select)	Hits:	> 2		
Search query	Last searched		Top clicked result	Hits	
filters pinboard	06/22/FY 2017		about pinboard filtersin 02_administration,	14	
filter pinboard	06/22/FY 2017		pinboard filtersin 01_the_basics/3.3	11	
filters	06/22/FY 2017		--	11	
stickers	06/07/FY 2017		create stickersin 02_administration/admini	10	
formula	06/13/FY 2017		formula referencein 02_administration/adr	9	
additive	06/07/FY 2017		change the additive setting for a columnin	8	
pivot	06/20/FY 2017		--	6	
filter	06/10/FY 2017		filter by a stickerin 02_administration/adm	5	
macos	05/04/FY 2017		--	5	
aggregation	06/07/FY 2017		change the aggregation setting for a colun	4	
load	06/20/FY 2017		--	4	
change data type	06/05/FY 2017		change the data type of a columnin 02_ad	4	

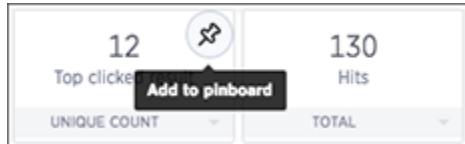
4 changes since 1 - 15 of 183

23 Search query UNIQUE COUNT	04/19/FY 2017 - 06/22/FY 2017 Daily (Last searched)	12 Top clicked result UNIQUE COUNT	130 Hits TOTAL
------------------------------------	--	--	----------------------

ThoughtSpot automatically creates up to 20 headlines for each table. Your ThoughtSpot configuration can be changed to accommodate more if needed.

You can modify how you'd like the value to be displayed by clicking the dropdown on a headline and selecting a different type of aggregation. The usual available aggregations are total, average, standard deviation, variance, minimum, and maximum. There are also unique count and total count values available for the appropriate columns.

To add a headline to a pinboard, hover over it and click the **Add to pinboard** icon.

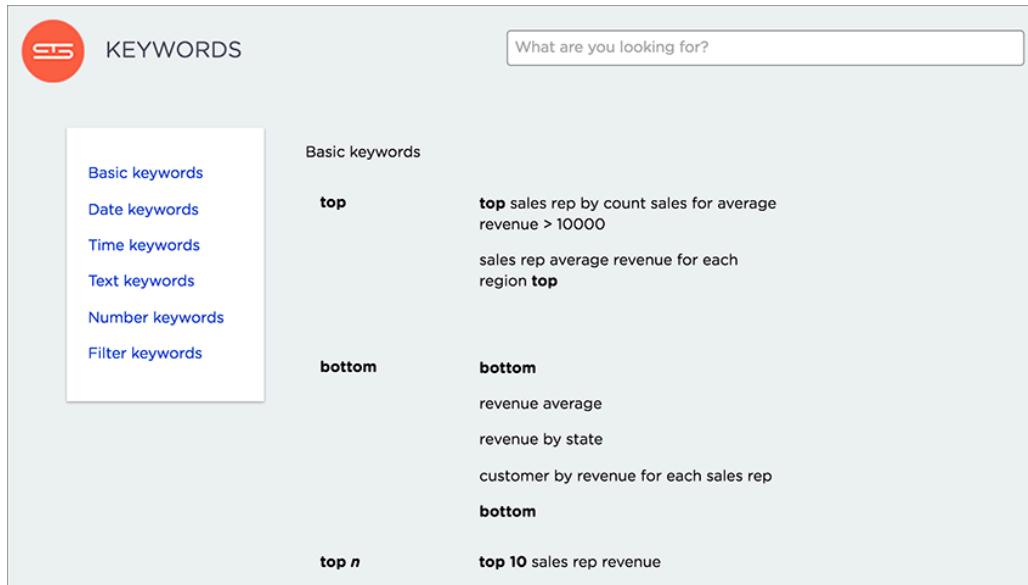


You can decide whether or not to display headlines in your table results. To customize headlines, choose **Actions > Customize summaries**.

# Overview of keyword searches

**Summary:** Use keywords when asking a question to narrow and further define your search.

In addition to column names, values, and filters, the search bar also accepts keywords. You can access a list of keywords and other reference materials in the [references](#) section or in the help center. Open the help center by clicking **Help** on the top navigation bar.

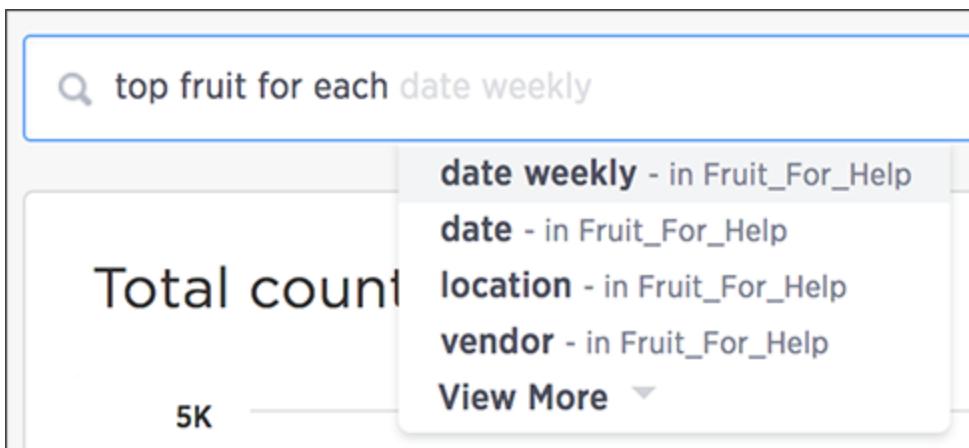


The screenshot shows the ThoughtSpot help center interface. At the top left is the ThoughtSpot logo (a red circle with 'TS'). To its right is the word 'KEYWORDS'. To the right of that is a search bar with the placeholder 'What are you looking for?'. Below the header, there's a sidebar on the left containing links: 'Basic keywords', 'Date keywords', 'Time keywords', 'Text keywords', 'Number keywords', and 'Filter keywords'. The main content area is titled 'Basic keywords'. It contains several examples of keyword usage. For 'top', it shows 'top sales rep by count sales for average revenue > 10000' and 'sales rep average revenue for each region top'. For 'bottom', it shows 'bottom revenue average', 'bottom revenue by state', and 'bottom customer by revenue for each sales rep'. For 'top n', it shows 'top 10 sales rep revenue'.

Within the help center, you'll find videos and documentation that pertain to the current version of ThoughtSpot. Here is where you can also find a list of keywords. You can expand each section to see which keywords are available and examples on how to use them.

## Example: Simplify search with the top keyword

The `top` keyword can be a powerful addition to your search when you remember to use it. By using one keyword, you can greatly simplify your searches. The `top` keyword is one such keyword. Its syntax looks like `top n` for each `x`.



Some common search examples it can simplify are finding the:

- top two performing sales rep
- top revenue average revenue by state
- top two students ranked by source for each class

## Types of keywords

Keywords serve a variety of predefined purposes, and are divided into different groups. The groups are as follows:

- **Basic keywords**

You can use keywords such as `top` and “`bottom`” to, for example, only see results for the best or worst performing sales rep.

- **Date keywords**

Date keywords give you the freedom to narrow your search by days, weeks, months, quarters, or years. There are also a number of date related keywords such as `after`, `before`, and `year-over-year`. Use the new date keywords to describe dates in the future. This is useful for exploring things that are scheduled for a future date, such as shipments due to go out in the next week.

- **Time keywords**

Time keywords are the most useful when trying to figure out how many visitors you've received within the last `n minutes or hours`.

- **Text keywords**

You can use text keywords to find similar words or phrases that contains a certain word. For example, `product name contains green`.

- **Number keywords**

These keywords allows you to define your search by sum, average, count, max, min, and other accumulations.

- **Comparative keywords**

Filter keywords work in the same way as filters on table columns or chart axes.

- **Location keywords**

The keywords `near` and `farther than` allow you to search with spatial filters on your geo type columns with latitude/longitude data. This lets you focus your search based on distance and locations. You can specify a specific radius using miles, kilometers, or meters.

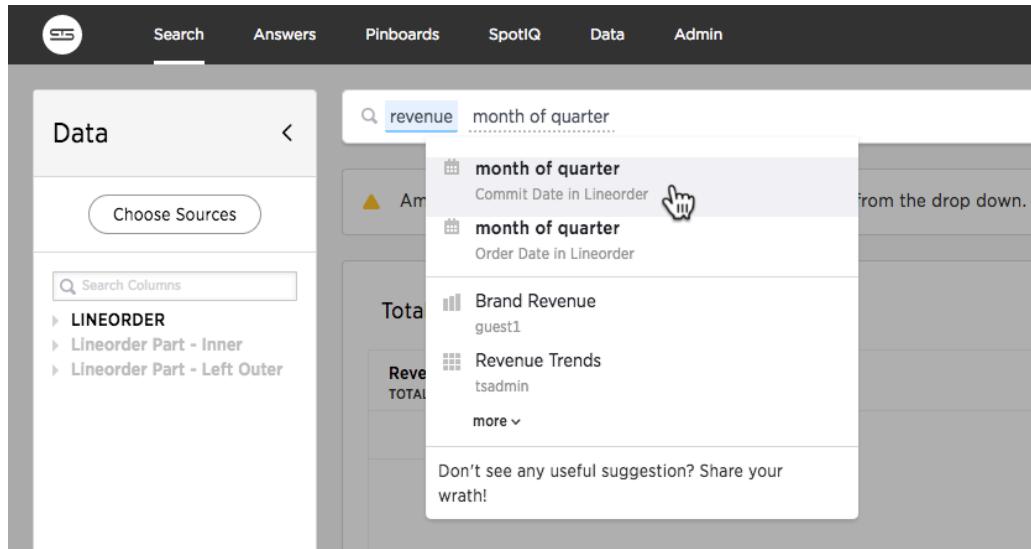
## Related information

- [Keyword reference](#)
- [Search using growth over time](#)
- [Geographical proximity keywords “near” and “farther than”](#)

# Time series analysis

**Summary:** You can compare across time periods without using a formula.

A *time series* is a set of data points ordered by time. For example, within the same week a sale on Tuesday comes before a sale on Wednesday in such a data set. You can use ThoughtSpot's *time series analysis* feature to search for answers about series data.



You might use this feature to compare a time period across other time periods. For examples sales for each month across several years. You may also want to calculate an aspect such as growth over the same time period across other periods. You can also do relative analysis such as sales for the last 3 months of each year across years.

You can use one or more of the following period keywords to create this type of analysis:

## Period keywords

Keyword	Example
quarter (date)	quarter (commit date)
quarter of year (date)	quarter of year (commit date)
month of quarter (date)	month of quarter (commit date)
week of year (date)	week of year (commit date)
week of quarter (date)	week of quarter (commit date)
week of month (date)	week of month (commit date)

day of year (date)	day of year (commit date)
day of quarter (date)	day of quarter (commit date)
day (date)	day (order date)
day of month (date)	day of monthly (order date)
day of week (date)	day of week (order date)
hour (datetime)	hour (timestamp)

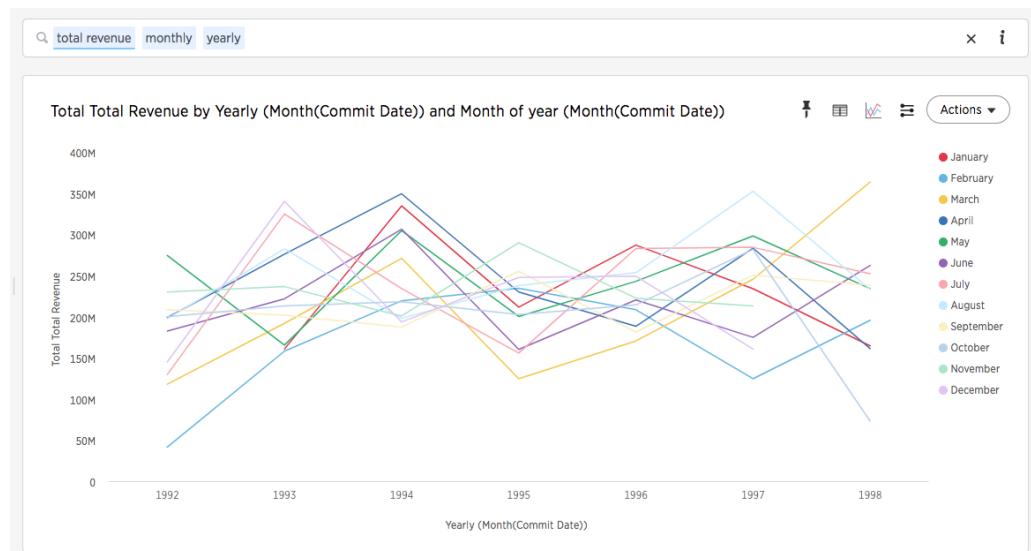
All of these keywords sort the data using datetime semantics, that is chronologically in a time sequence. By default, the **Search** bar suggests these keywords less frequently than others.

You can use these new keywords in combination with the existing data keywords which are:

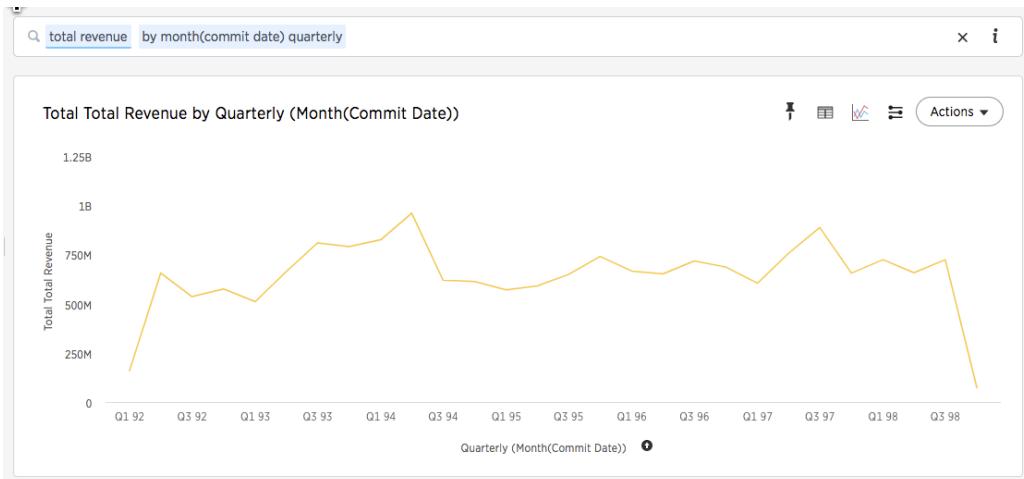
- Detailed
- Hourly
- Daily
- Weekly
- Monthly
- Quarterly
- Yearly

## Examples of time series analysis

Typically, when you search for answers about series data, the visualizations that answer your questions are line charts. These charts frequently but not always include a stack to indicate a period.



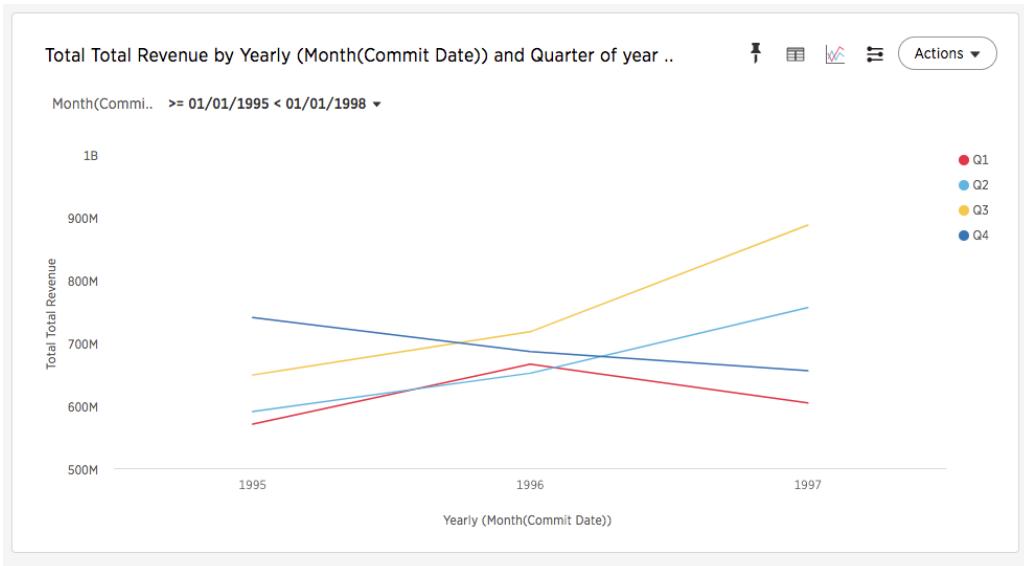
When you search for an aspect of data of time series, a typical response is a line chart showing the aspect as it rises and falls over time.



You can also add a relative date filter for example,

```
total revenue quarterly yearly by year month(commit date) >= 0
1/01/1995
month(commit date) before 01/01/1998
```

This type of query also yields a stacked line chart:



The child date time attribute is on the X-axis and the parent in the legend. For example, if you search `revenue month yearly` the child, `monthly`, appears on the x-axis and the parent, `yearly`, in the legend.

## Granularity for date filters

You can refine a simple date filters by adding hierarchical date filter to your query. The ability to specify two bucket granularities such as “hour of day” or “week of year” are two examples. The syntax of this type of query is

```
small_bucket of big_bucket [INTEGER_CONDITION]
```

The `INTEGER_CONDITION` is optional but it must be an integer. For example, this query is valid:

```
revenue by day of week <= 2
```

This query is invalid:

```
revenue by day of week = Tuesday
```

You can specify one or more granular filters.

These tips and gotchas apply to time granularity:

- The system-defined fiscal rules are respected. This means, for example, if the fiscal year begins in February, `month of year = 2` matches dates in March.
- Fiscal shorthands such as `Q1`, `Q2` and so on are not supported, so `day of week = d1` is not valid.
- `INTEGER_CONDITION` with `=` or `!=` accept a list of filter values, so, `day of week = 1 2 3` is valid.
- `INTEGER_CONDITION` with `=` or `!=` require legal values, so `day of week >` accepts any integer on the right hand side while `day of week =` requires a value in the legal `1-7` range.
- Simple date filters allow you to use edit the filter through the answer to refine your search, adding a hierarchical date filter in the search bar disables this ability.

# Search using growth over time

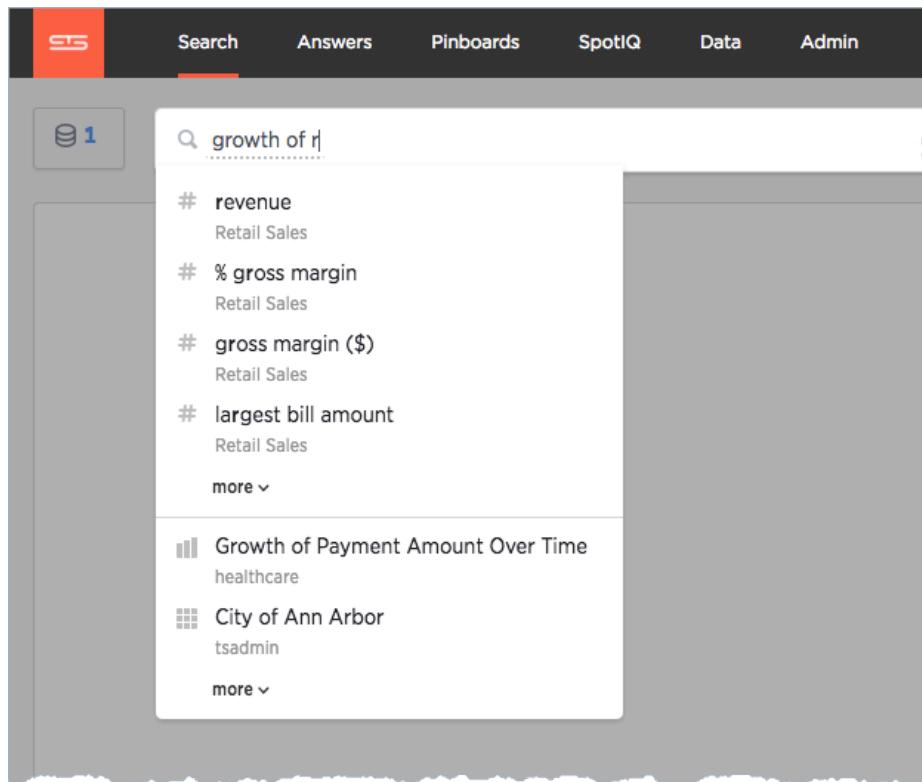
**Summary:** You can show growth over time by using the growth of keyword in your search.

The `growth of` keyword compares the data from different date periods, and returns a percentage of growth.

## Growth of measure by year

To search using growth over time:

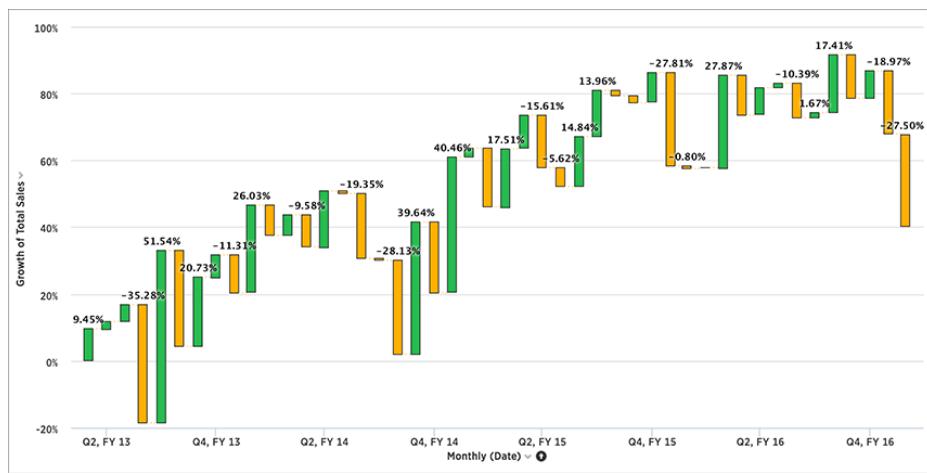
1. Type `growth of` into the search bar, and choose a measure you're interested in seeing the growth of.



You will see a list of suggestions to choose from based on your sources. You can also type a different column name containing numeric data to compare.

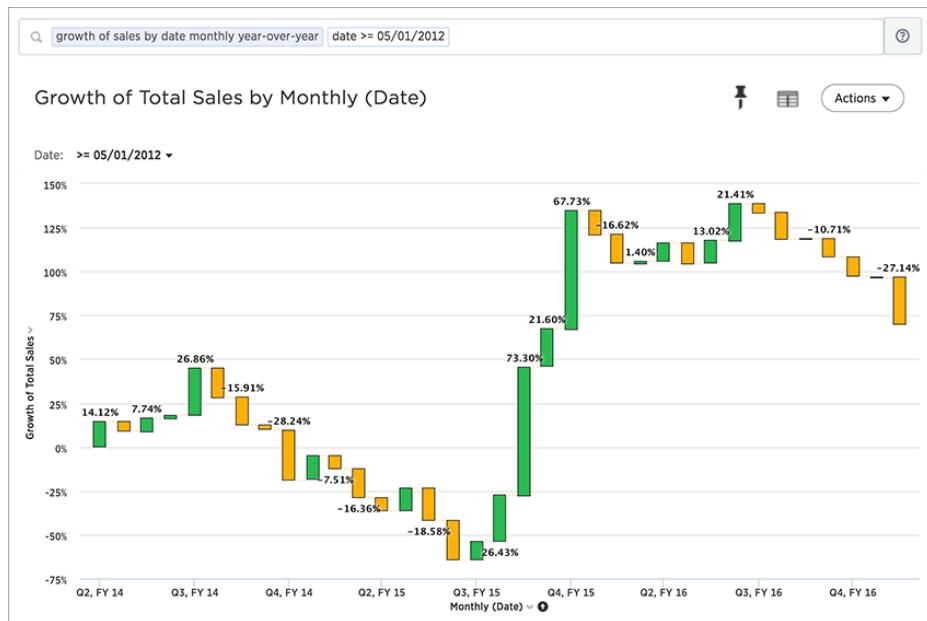
2. Type `by`, followed by a date column name.

The growth is calculated as a positive or negative percentage, for each period relative to the last period in the series. A line chart is a good way to display your data, but the waterfall chart is especially effective to show growth.



3. You can also do year-over-year analysis, which compares each time period to the corresponding time period in the prior year.

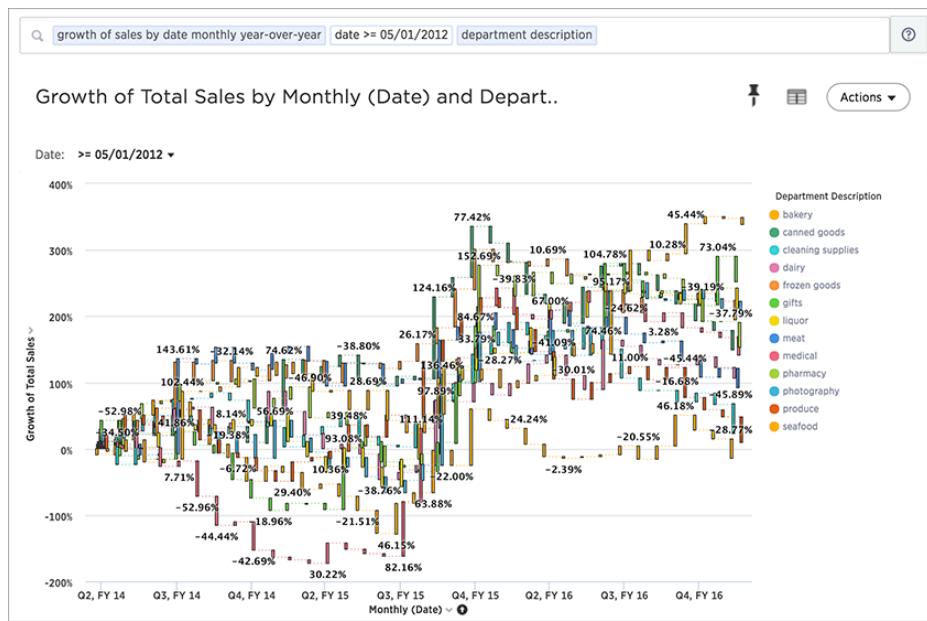
This type of growth analysis is more common in retail and other seasonal businesses. To do this, type monthly year-over-year after your growth of phrase in the search bar.



This compares data between the same month from different years. For example, it will allow you to compare sales from June of this year to the sales from June of last year. Note that for the first year, values are labeled as “{Blank}” in the Data View since there are no previous data to compare them to.

Add an attribute to your search see the breakdown of how each grouping of the attribute contributed to the overall growth of your measure.

4. Click **Configure Chart** and add your attribute to the Legend field.



It looks like Q4, FY 15 was a much more productive period for a number of categories compared to a year ago.

## Display periods (daily) over time (year-over-year)

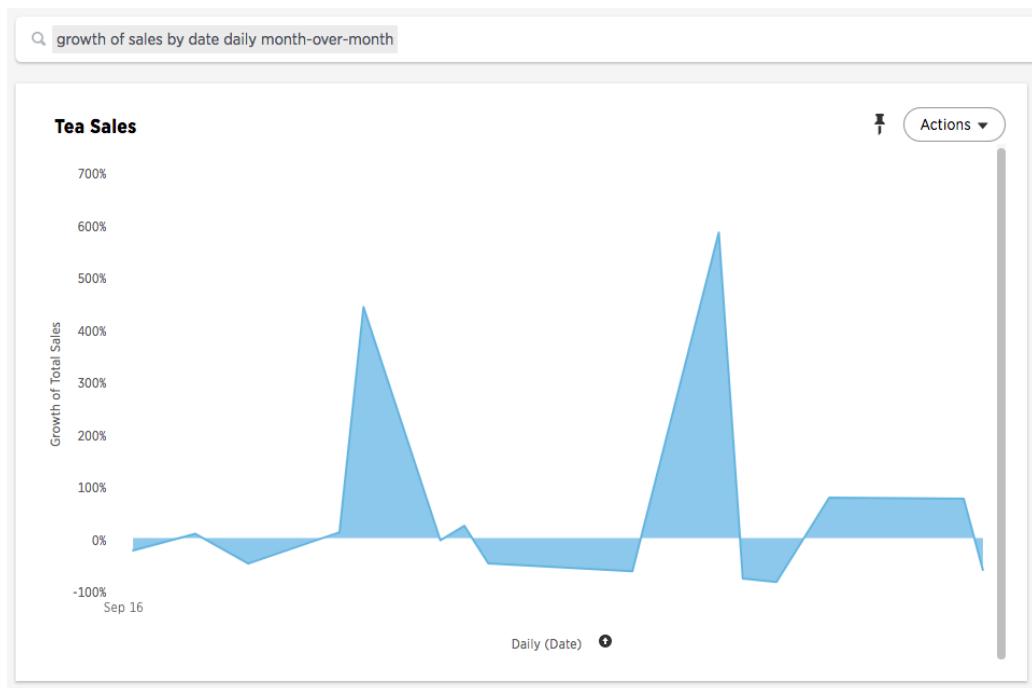
Growth query now supports more hierarchical date buckets and period calculations. The growth query syntax is as follows:

```
growth of <measure_column> by <date_column> <bucket> <period-over-period>
```

This table shows the possible buckets and the period-over keywords you can combine:

quarterly	monthly	weekly	daily	hourly
year-over-year	year-over-year	year-over-year	year-over-year	
	quarter-over-quarter	quarter-over-quarter	quarter-over-quarter	
		month-over-month	month-over-month	
			week-over-week	
				day-over-day

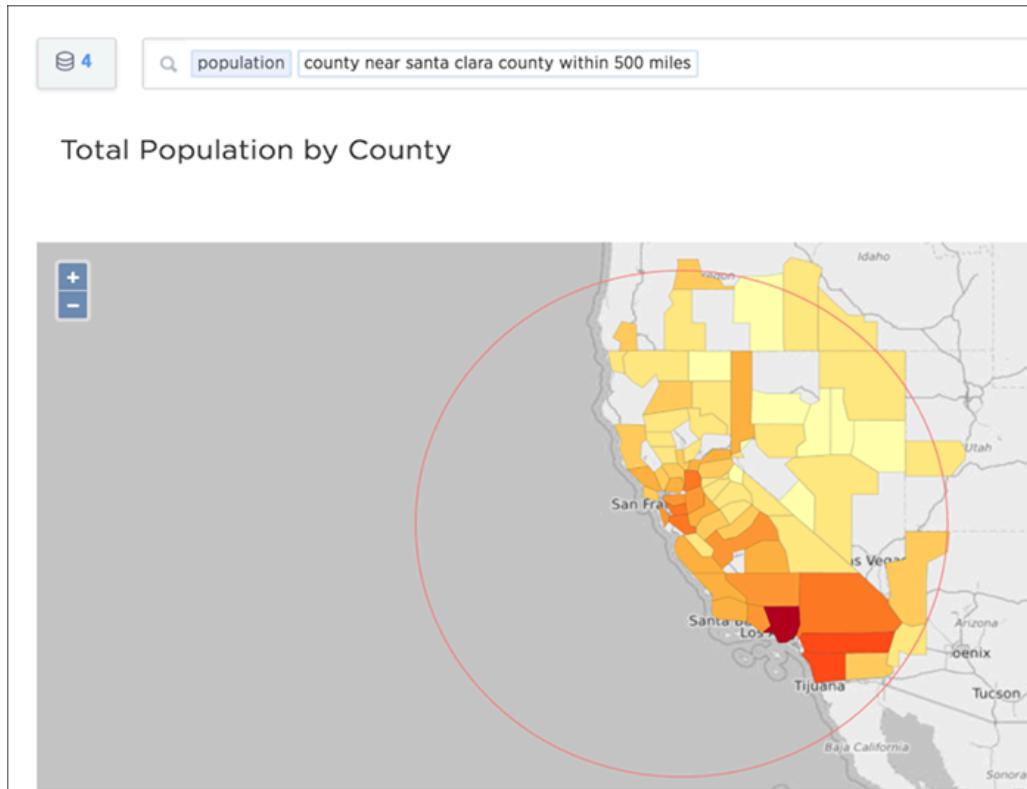
For example, the following query `growth of sales by date daily month-over-month`:



# Proximity searches "near" and "farther than"

**Summary:** If your table contains Latitude and Longitude data, you can use proximity searches that find entities related to each other by location.

Proximity keywords allow you to search and show the results on a map. Using proximity keywords causes ThoughtSpot to display a circle that represents your set distance on the geo charts.



The proximity keywords are:

- `near`  
`near...within n miles      km      meters`
- `farther than n miles      km      meters from`

The distance is calculated as a straight-line distance (not road distance) radius using miles, kilometers, or meters from the central point. If you do not specify a distance, the system uses 10 km as the default.

Given a latitude, a longitude, and an optional distance, the search returns all instances of a geotype column that falls within the parameters. These keywords are limited to 33 latitude/longitude pairs. They work on duplicate counties. Finally, you can filter on them. Some examples of valid searches are:

```
landings latitude longitude city near tokyo  
landings latitude longitude city near tokyo within 50 miles
```

You can bracket your search only with actual values found in the data. So "longitude between `-125.000000` and `-115.316670` worked for me, but not `longitude between -125 and -115`.

## Proximity search configuration requirements

All your data must be in the same set. The worksheet or one of the tables must contain a column of type `longitude` and a column of type `latitude`. The latitude and longitude data have to be on the same base tables for the feature to work. You can't, for instance, have a dimension table with all your cities and their associated geo coordinates and join to it from your fact table and expect proximity search to work.

Also, your administrators must have configured these columns using the appropriate GeoType.

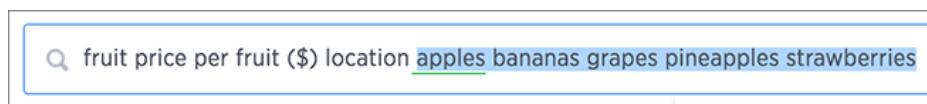
# Understand filters

**Summary:** Filters narrow down the search result to only include the data you want to see.

When you add a value to your search, it becomes a filter. To add a filter from the search bar:

1. Click in the search bar and type the values you want to include in the search.

Typing a value in the search bar acts as a filter.



You can also use keywords like `yesterday`, `after`, and `next month` to filter your search. To see more keywords, refer to the [keyword reference](#).

2. Click outside of the search bar or push enter to apply your filter.

Simple filters can be applied to an answer, while pinboard filters can be applied to all visualizations of a pinboard. You can find out more about [pinboard filters](#) in the pinboards section.

## Where filters appear in ThoughtSpot

As you have seen with search, filters appear in white boxes in the search bar.



In an answer or a pinboard, filters appear just below the title. For pinboards, your filters apply to all worksheet-based visualizations in the pinboard.

The screenshot shows a pinboard titled "Olympics Medals Analysis". Just below the title, there is a filter input field. The field contains the text "Medal: gold ▼" and has a small red "X" button to its right, which is used to clear the selection. This illustrates how filters are applied directly to the pinboard's visualizations.

If you ever find that your search or pinboard does not appear to contain all the data you want to see, check for any existing filters and remove them by clicking the **X** to see all the data.

**Note:** Filtering on NULL and empty values is a special case. You can find out more about how these values are represented and how to filter for them in [About filtering on null, blank, or empty values](#).

## Simple filters

Simple filters can be applied to searches in a few different ways. You can use the search bar or the **Change Configuration** menu to add a filter to a search. You can apply simple filters to your search, whether it shows a table or a chart. Your filters remain part of the search even when you change the visualization type.

When adding a filter from the **Change Configuration** menu, numeric columns and text columns provide you with a checkbox selector for values. If the column contains a date, you'll see a calendar selector when applying a filter. This is also where you'll go to apply bulk filters.

## Bulk filters

If you have a large worksheet or table with thousands or millions of rows, you may want to create bulk filters. You can paste in a list of filter values, without having to click the box next to each value in the filter selector.

Bulk filters can be very useful when you have a very large worksheet or table. You can use them to filter a large list of values easily. For example, this is useful if you want to only search on a list of products that your manager sent to you in an email. You can cut and paste those values into the bulk filter box to quickly generate a report or chart that includes only those items of interest.

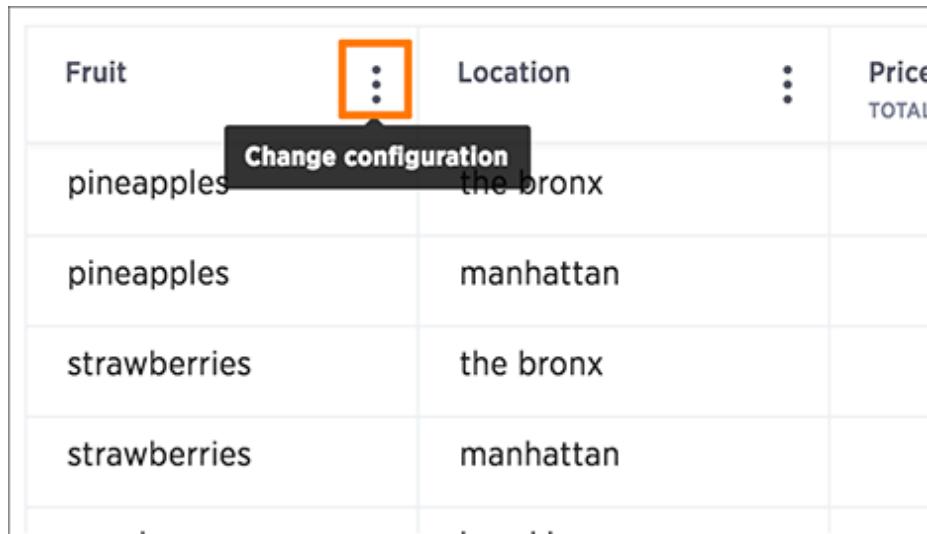
You can [create a bulk filter](#) by pasting a list of values, separated by commas, semicolons, new lines, or tabs, into the bulk filter box. This allows you to easily search a large list of filters repeatedly.

## Add a filter to a table

**Summary:** You can add a simple filter from a column header while viewing your answer as a table.

To add a filter from column headers:

1. While viewing your answer as a table, click **Change configuration** on the column header you want to filter.

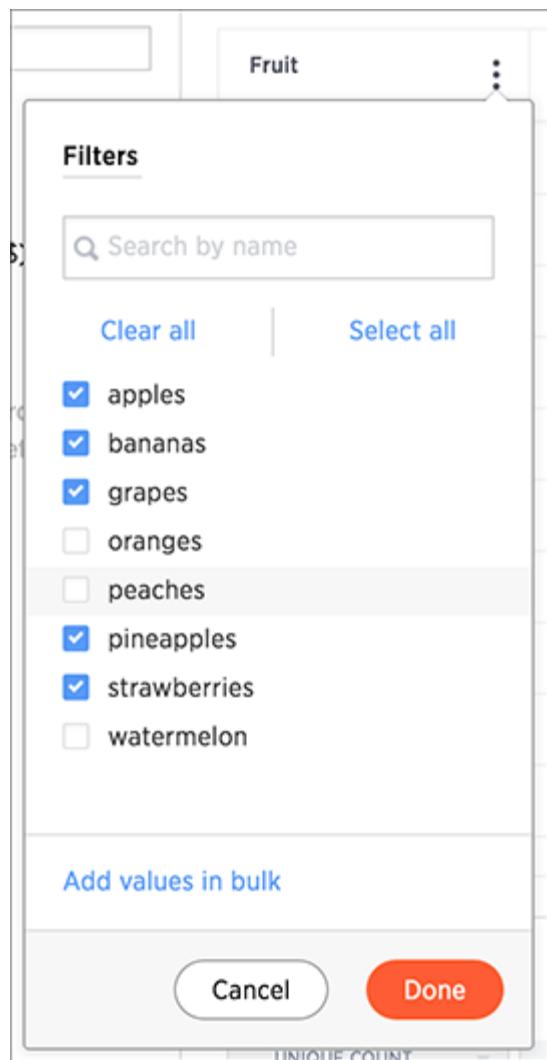


The screenshot shows a table with four rows of data. The columns are labeled 'Fruit', 'Location', and 'Price TOTAL'. The first row contains 'pineapples' in the Fruit column and 'the bronx' in the Location column. A black rectangular box highlights the 'Change configuration' button in the Fruit column header. The second row contains 'pineapples' in the Fruit column and 'manhattan' in the Location column. The third row contains 'strawberries' in both the Fruit and Location columns. The fourth row contains 'strawberries' in the Fruit column and 'manhattan' in the Location column. The 'Price TOTAL' column is empty.

Fruit	Location	Price TOTAL
pineapples	the bronx	
pineapples	manhattan	
strawberries	the bronx	
strawberries	manhattan	

2. Select **Filters**.
3. Select the values to include in your answer.
4. Click **Done**.

If there are too many values, you can use the filter search bar to find the ones you want.



# Add a filter to a chart

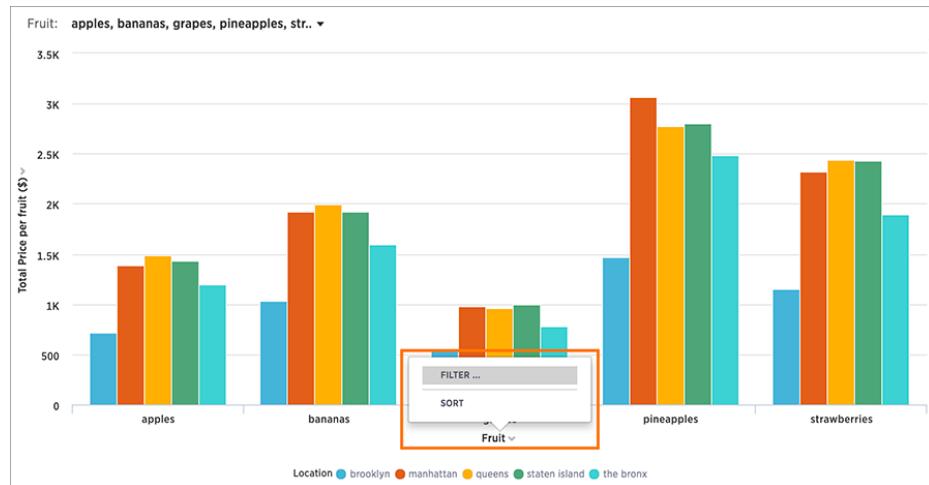
**Summary:** You can add a simple filter from a chart axis while viewing your answer as a chart.

You can filter a chart in two ways:

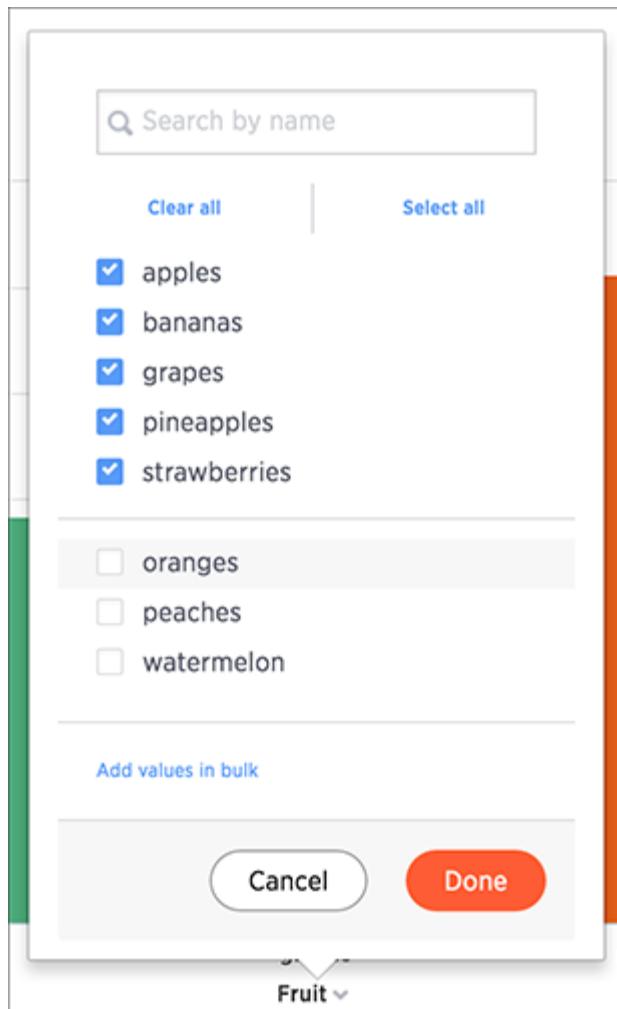
1. Click on the legend labels to toggle the values on or off.
2. Follow the steps shown here to filter on the axis values.

To add a filter from the chart axes:

1. While viewing your answer as a chart, click the chart axis you want to filter on, and select **Filter**.



2. Select the values you would like to include in your answer. Then click **Done**.



# Create a bulk filter

You can create a bulk filter on a worksheet or table. You create the filter by pasting a list of values, separated by commas, semicolons, new lines, or tabs, into the bulk filter box. A filter allows you to easily search a large list of values repeatedly.

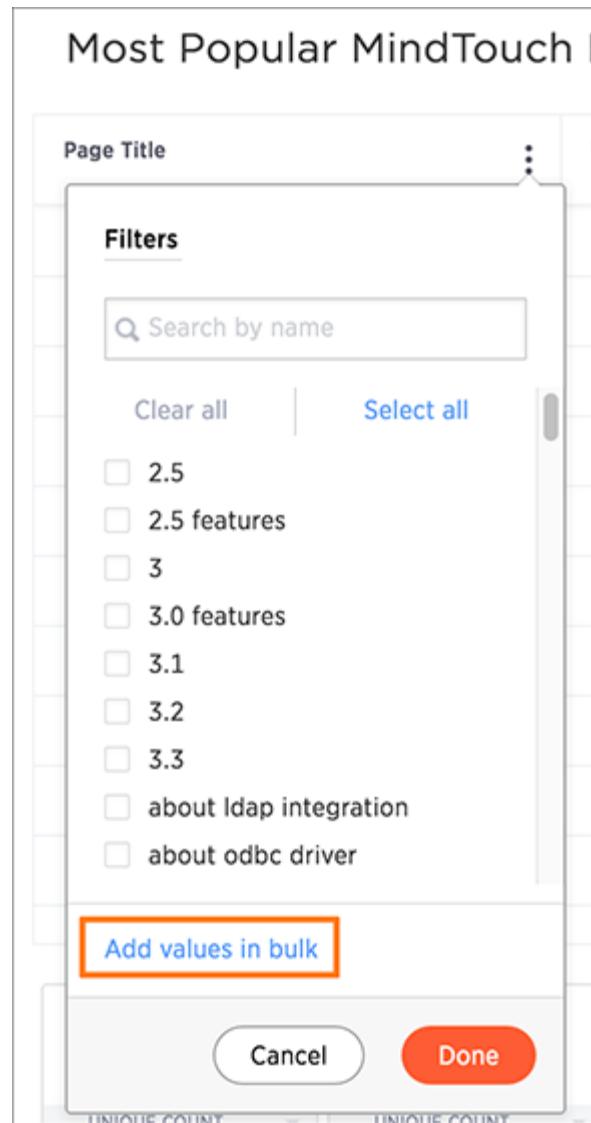
Your filter can have up to 10k values in it. When you add a filter, the system does not check that the value exists in the data. This allows you to anticipate values that *may be* added in your filter.

Before you can create a bulk filter, you must have **Can edit** permission on the table or worksheet. In this example, you will cut and paste values to create a bulk filter. You could also get a list of text values from another application such as an email or cells from an Excel or Google Sheets spreadsheet.

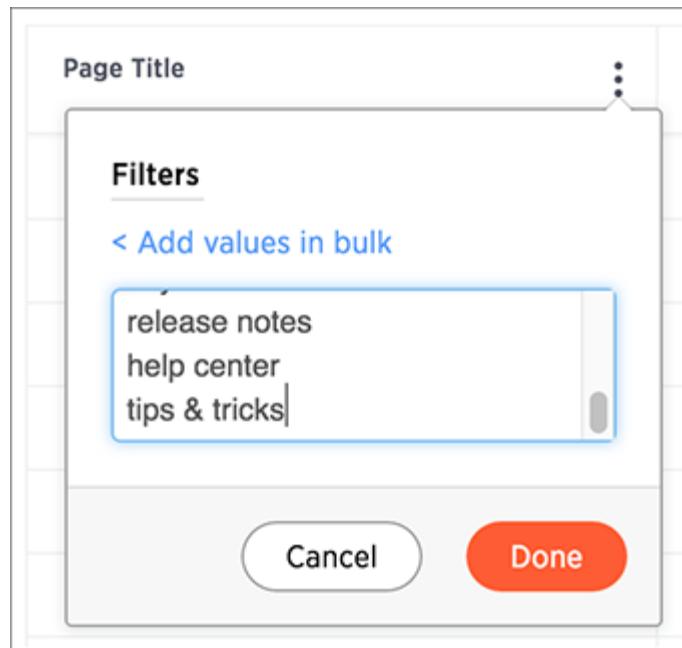
1. When viewing a table, select multiple cells by clicking and dragging.
2. Right click and choose **Copy to Clipboard**.

Most Popular MindTouch Pages November 2015			
Page Title	URL	Views <small>TOTAL</small>	
welcome to the help center	<a href="https://help.thoughtspot.com/help_center">https://help.thoughtspot.com/help_center</a>	2,422	
installation and setup	<a href="https://help.thoughtspot.com/02_administrators/installation_and_setup">https://help.thoughtspot.com/02_administrators/installation_and_setup</a>	2,000	
administration	<a href="https://help.thoughtspot.com/02_administrators">https://help.thoughtspot.com/02_administrators</a>	1,151	
the basics	<a href="https://help.thoughtspot.com/01_the_basics">https://help.thoughtspot.com/01_the_basics</a>	879	
knowledge base	<a href="https://help.thoughtspot.com/03_knowledge_base">https://help.thoughtspot.com/03_knowledge_base</a>	743	
keywords	<a href="https://help.thoughtspot.com/help_center/keywords">https://help.thoughtspot.com/help_center/keywords</a>	727	
release notes	<a href="https://help.thoughtspot.com/03_knowledge_base/release_notes">https://help.thoughtspot.com/03_knowledge_base/release_notes</a>	677	
help center	<a href="https://help.thoughtspot.com/help_center">https://help.thoughtspot.com/help_center</a>	630	
tips & tricks	<a href="https://help.thoughtspot.com/help_center/tips_tricks">https://help.thoughtspot.com/help_center/tips_tricks</a>	620	
administrator guide 2.2	<a href="https://help.thoughtspot.com/02_administrators/administrator_guide_2.2">https://help.thoughtspot.com/02_administrators/administrator_guide_2.2</a>	581	

3. Click the **Filters** icon in the column header.
4. Click **Add values in bulk**:



5. Paste the values into the bulk filter box.



6. Click **Done**.

## Delete a filter

**Summary:** You can delete a filter from an answer to return to the original unfiltered search result.

There are multiple ways to delete a filter. To delete a filter you can click the **x** on the filter term in the search bar.



You can also click the **x** on the filter bar above the answer.

A screenshot of an answer card titled "Total Price per fruit (\$) by Fruit, Location". The card includes a filter bar with the text "Fruit: apples, bananas, oranges, peaches, pinea.." and a red box around the "x" button. Below the filter bar is a table with the following data:

Fruit	Location	Price per fruit (\$)
pineapples	the bronx	2,487.00
pineapples	manhattan	3,066.00

# Filter on null, blank, or empty values

**Summary:** Filtering on NULL and empty values is a special case.

Filtering on null, blank, or empty values can be tricky if your data contains both of these. You can use this method to see what's really going on with these types of values, and to get the filtering behavior you want.

## How NULL and blank values are displayed

When you view a table or chart, you may see values that appear as **{blank}**. These can actually be one of two types of values:

- NULL values, which are essentially missing values.
- blank or empty values, like an empty string of text or a string containing only whitespace (spaces, tabs).

Both of these types of values are represented as **{blank}**, but if you filter on **{blank}**, the filter will apply to only the NULL values. That is, only the NULL values will be included in your result. It can be hard to tell what's going on if you have a data source that contains both NULL and blank/empty values.

## To show NULL and blank values differently

If you need to differentiate between NULL and blank values, you can [Add a formula](#) to make them appear differently in charts and tables. In this example, we'll use **<text\_column>** to refer to the text column which contains both NULL and blank values:

```
if ( strlen ( <text_column> ) = 0 ) then if ( isnull ( <text_column> ) ) then 'null' else 'empty' else <text_column>
```

This formula will show “null” where the value contained in the column is actually NULL. When the value is blank or empty, it will show up as “empty”.

## To allow filtering on both NULL and blank values

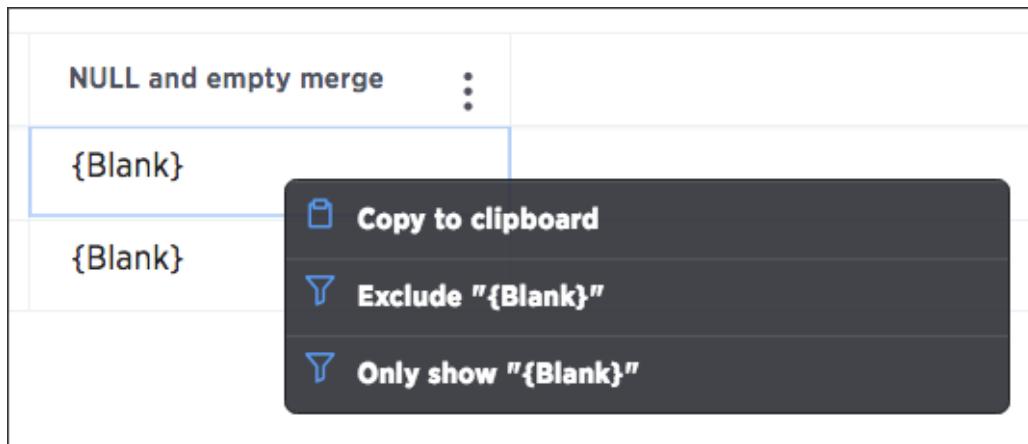
If you want to keep the same display format for NULL and blank values, but be able to filter on both using “{blank}”, your [formula](#) will be slightly different. You can use a formula like:

```
if ( strlen ( <text_column> ) = 0 ) then null else <text_column>
```

Use the filter you created instead of the original text column in your search to get the result you desire.

## Filtering on your formula

After creating the above formula that fits what you want to do, you can filter on the formula column you created in the search bar by typing the value **{blank}**, which will act as a filter. Or you can filter by left clicking on a **{blank}** value in your search result table, then right clicking and selecting **Show only "{Blank}"**.



# Other search actions

There are other search actions you can perform by interacting directly with your answer. These actions can be performed in an ad hoc way, or you can pin your answer to a pinboard to save your configurations.

- **Change view options**

You can change the view of your answer so it appears as a table or a chart. Sorting your search allows you to order your answer, making it easier to read. You can change the date bucketing on tables and charts for columns with date values.

- **Show underlying data**

Viewing the underlying data of your answer gives you an un-aggregated view of the underlying data.

- **Drill down**

Drilling down allows you to see more information about the columns used within your search.

- **Exclude and include row values**

You can include or exclude row values from your answer.

- **Apply conditional formatting**

You can apply conditional formatting to tables or charts to highlight values in the data. This makes values above, below, or within a particular threshold stand out.

- **Work with saved answers**

You can make a copy of an answer if you would like to make edits without changing the original answer.

- **Download your search**

You can download your search as either a table or chart.

- **Replay search**

You can instantly generate a step-by-step replay showing the creation of a table or chart.

# Change result display options

**Summary:** You can change how your search results (answers) appear.

Your search answer (results) have various display options. You can change the visualization used to display your results, their sort order, and how they are grouped or grouped.

## Change the view

You can change the view of your answer so it appears as a table or a chart. To change the view of your answer:

1. View your answer.
2. Click **Change Visualization**.
3. Select either table or a chart type.



## Sort your search

Sorting your search allows you to order your answer, making it easier to read. To sort your search:

- If you are in the data (table) view, click the column header you would like to sort on.

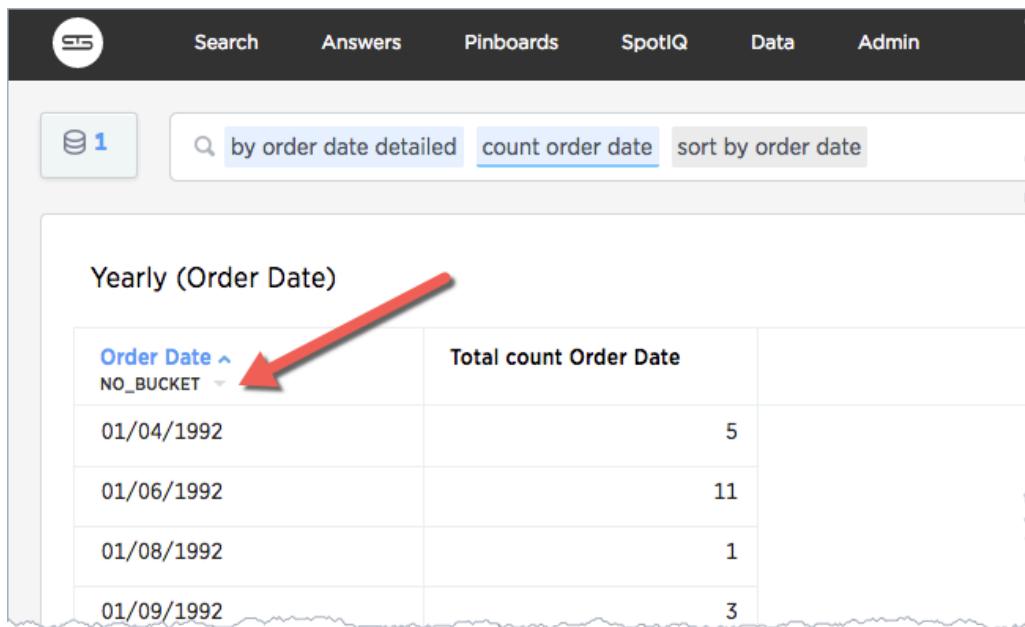
By default, sorting applies in descending order. You can click on a column header to sort again to sort in ascending order.

**Tip:** Hold shift and click another column to add a secondary sort. You can even add tertiary sorting and so on by continuing to use this trick.

- If you are in the chart view, click the axis label of your chart and select **Sort**.

## Change the date bucketing (grouping)

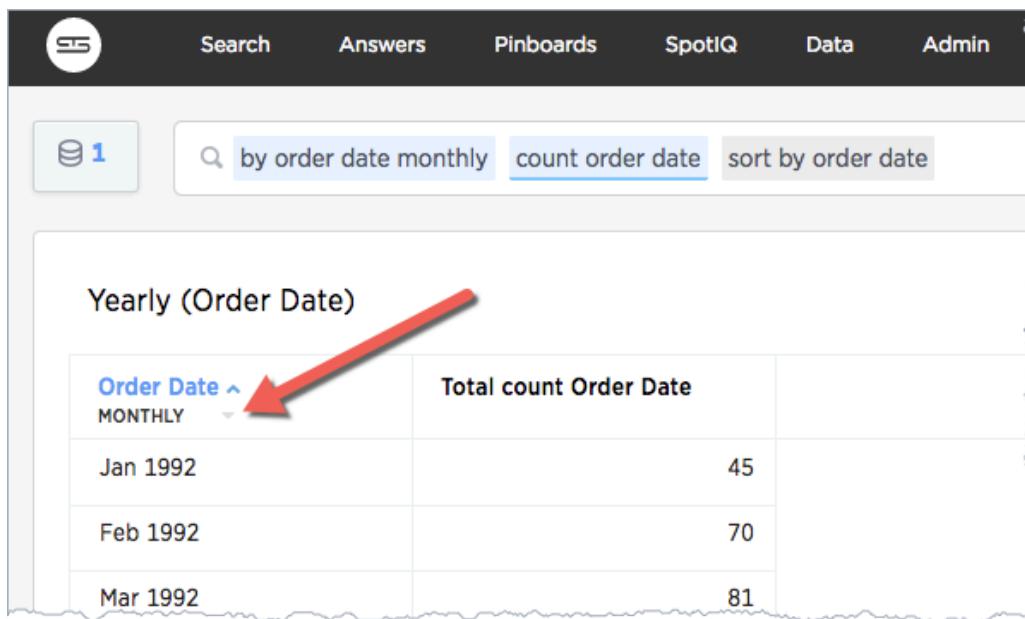
You can change the date *bucketing* on tables and charts for columns with date values. Bucketing is a method for grouping a column in your data. For example, if you have a column of orders, you can use the date of each order (**NO\_BUCKET**).



The screenshot shows a ThoughtSpot interface with a search bar containing "by order date detailed". Below the search bar is a table titled "Yearly (Order Date)". The table has two columns: "Order Date" and "Total count Order Date". The "Order Date" column is sorted by date. A red arrow points to the dropdown menu next to the first item "NO\_BUCKET".

Order Date	Total count Order Date
01/04/1992	5
01/06/1992	11
01/08/1992	1
01/09/1992	3

Alternatively, you can group these **MONTHLY**:



The screenshot shows a ThoughtSpot interface with a search bar containing "by order date monthly". Below the search bar is a table titled "Yearly (Order Date)". The table has two columns: "Order Date" and "Total count Order Date". The "Order Date" column is sorted by month. A red arrow points to the dropdown menu next to the first item "MONTHLY".

Order Date	Total count Order Date
Jan 1992	45
Feb 1992	70
Mar 1992	81

ThoughtSpot chooses a default date bucket for you when you search. This default takes the entire search result into account. For example, if your search includes last month, dates are bucketed daily instead of monthly.

On a table view, you use the date bucket dropdown below the column header. On a chart, you can change this via the axis label.

The screenshot displays two interfaces for changing date bucketing:

- Table View:** Shows a dropdown menu for "Order Date" with options like NO\_BUCKET, HOURLY, DAILY, WEEKLY, MONTHLY, QUARTERLY, and YEARLY. The "YEARLY" option is highlighted.
- Chart View:** Shows a dropdown menu for "Order Date" with options like DETAILED, HOURLY, DAY, WEEK, MONTH, QUARTER, and YEARLY. The "YEARLY" option is also present here.

The bucket values are a subset of date and time period keywords. To change the date bucketing:

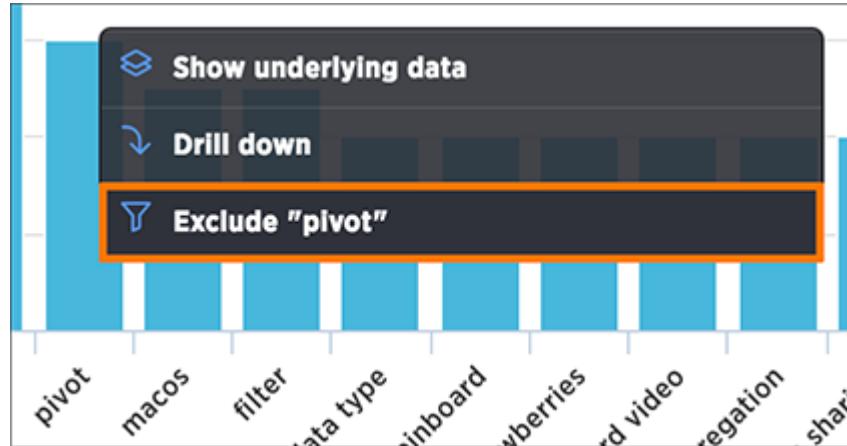
1. Go to the table or chart you want to change.
2. Locate the column or axis value with the date value you want to change.
3. Click the column's bucket menu.
4. Select a date bucket.

## Exclude and include row values

You can include or exclude row values from your answer.

To exclude or include row values:

1. Right click on the visualization or table cell of interest
2. Select **Exclude “value”** or **Include “value”** if available.



# Show underlying data

**Summary:** Viewing the underlying data of your answer gives you an un-aggregated view of the underlying data.

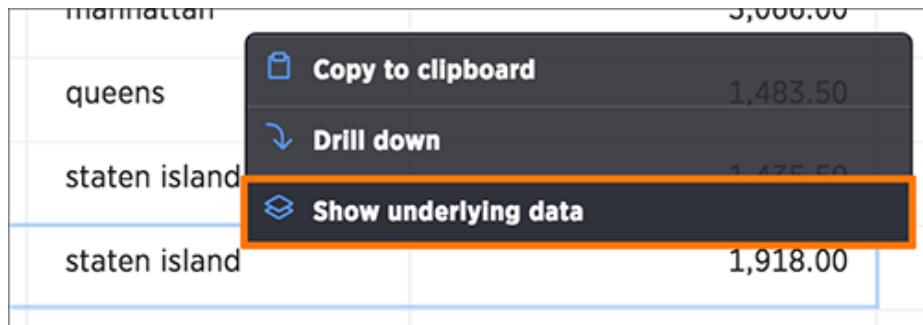
You can see the most granular details of a given result set, i.e. it shows the un-aggregated view. This feature lets you understand what an answer consists of. For example, if you search for `customer region revenue`, the answer shows the aggregate revenue value for each customer region. You can then click on any row and then on **Show underlying data**, to see each value that `revenue` constitutes of, for any given region.

Only the first 1,000 rows are shown when viewing the underlying data. You can even download the results shown when you choose **Show underlying data** from a chart. The download file limit is a million rows.

**Note:** Viewing underlying data does not work for answers that are derived from chasm trap searches. Nor does this feature work for pivot tables.

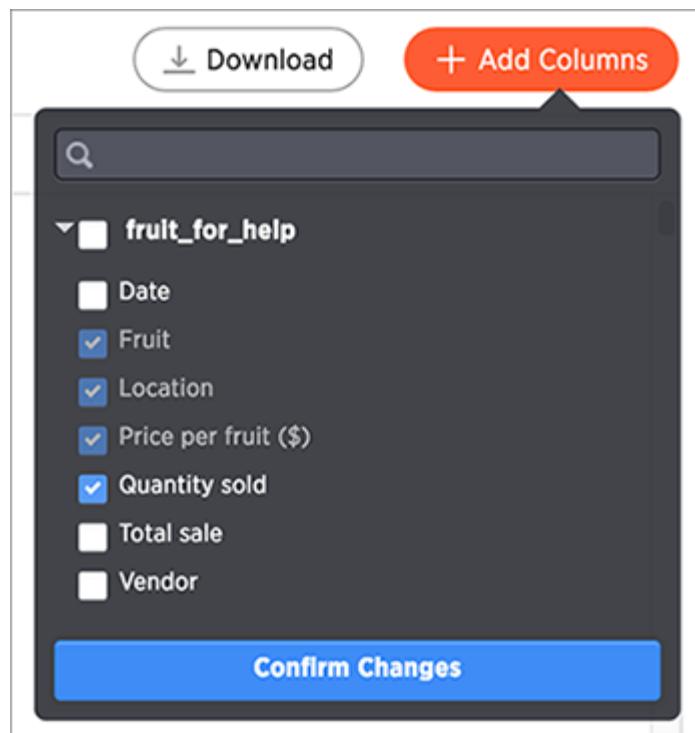
To show underlying data:

1. Right click on the visualization or table cell of interest
2. Select **Show underlying data**.



A new window opens that displays a summary and the underlying data.

3. Optionally, choose to + Add Column to add additional data.



4. Click **Confirm Changes**.
5. Click **Download** to download a CSV file of the data.

Showing underlying data			
Summary		Fruit: bananas	
		Location: staten island Total Price per fruit (\$): 1.918.00	
Underlying Data			
Location	Fruit	Price per fruit (\$)	
staten island	bananas	2.00	
staten island	bananas	2.00	
staten island	bananas	2.00	
staten island	bananas	2.00	

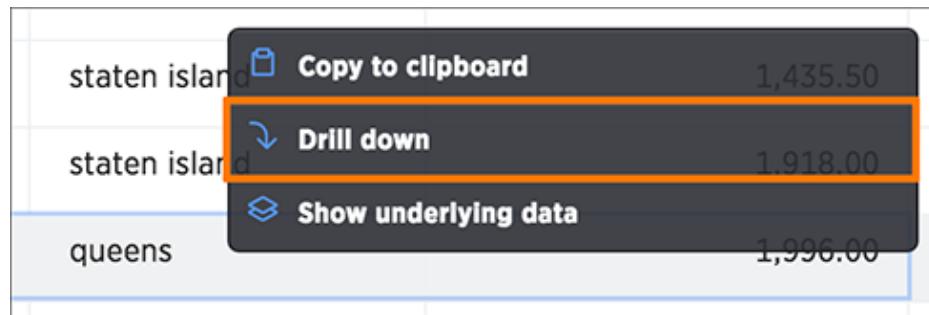
## See the search behind a result

**Summary:** You can use your answer to learn more about the data used in your search.

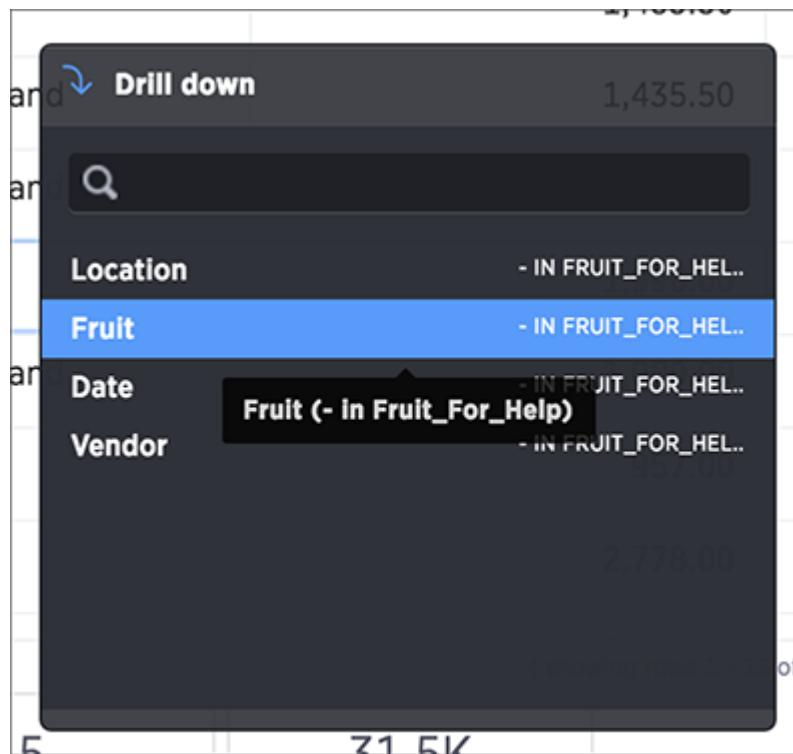
Drilling down allows you to see more information about the columns used within your search. If you own the pinboard you can drill down to the data beneath. Users you share the pinboard with can also drill down provided they *also* have access to the data on which the board was based.

To drill down:

1. Right click on the visualization or table cell of interest, and select **Drill down**.



2. Click on any of the listed data to recreate the search with that data included.



# Apply conditional formatting

**Summary:** Use conditional formatting to make portions of your results stand out

You can apply conditional formatting to tables or charts to highlight values in the data. This makes values above, below, or within a particular threshold stand out.

## Understand conditional formatting

Many companies create pinboards with key metrics they want to track in daily or weekly staff meetings. Using conditional formatting, they can see at a glance how they are performing relative to these metrics.

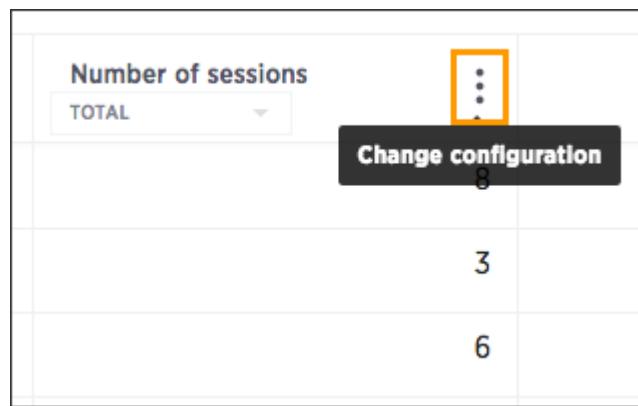
You can add visual cues for KPIs (Key Performance Indicators) or threshold metrics to charts and tables, to easily show where you are falling short or exceeding targets. These visual cues are called conditional formatting, which applies color formatting to your search result. For tables, you can add conditional formatting to set the background color of cells in a table based on the values they contain. For charts, you can add conditional formatting to show the threshold(s) you defined, and the data that falls within them will be shown using the same color.

## Apply conditional formatting to a table

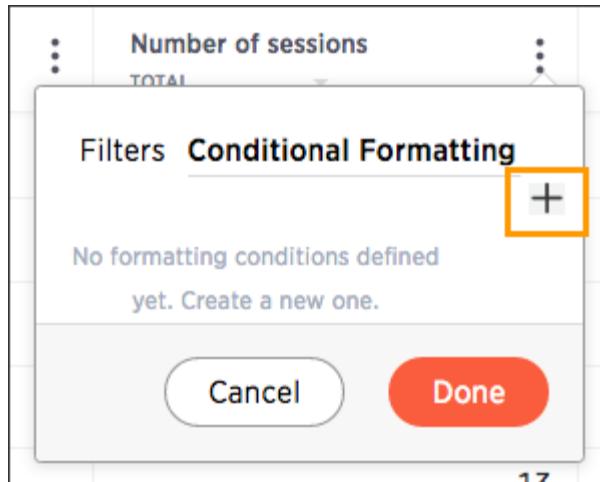
You can use conditional formatting to show table cells with a background color determined by the value they contain.

To apply conditional formatting to a table:

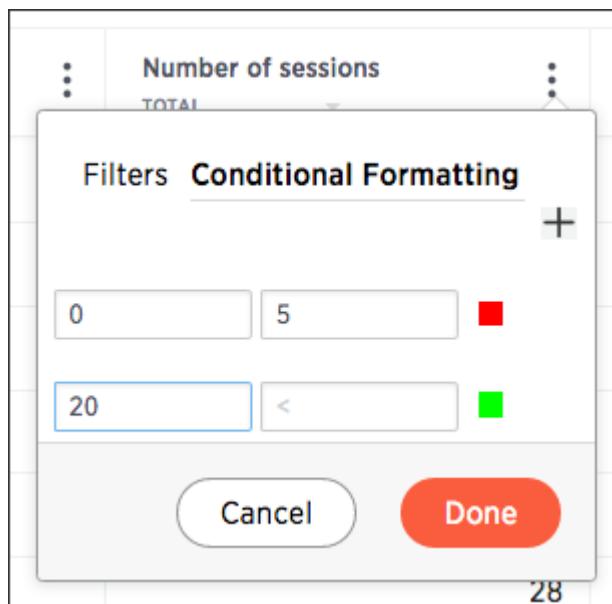
1. In the column header of your table for the column you want to apply formatting to, click the three dot **Change Configuration** icon.



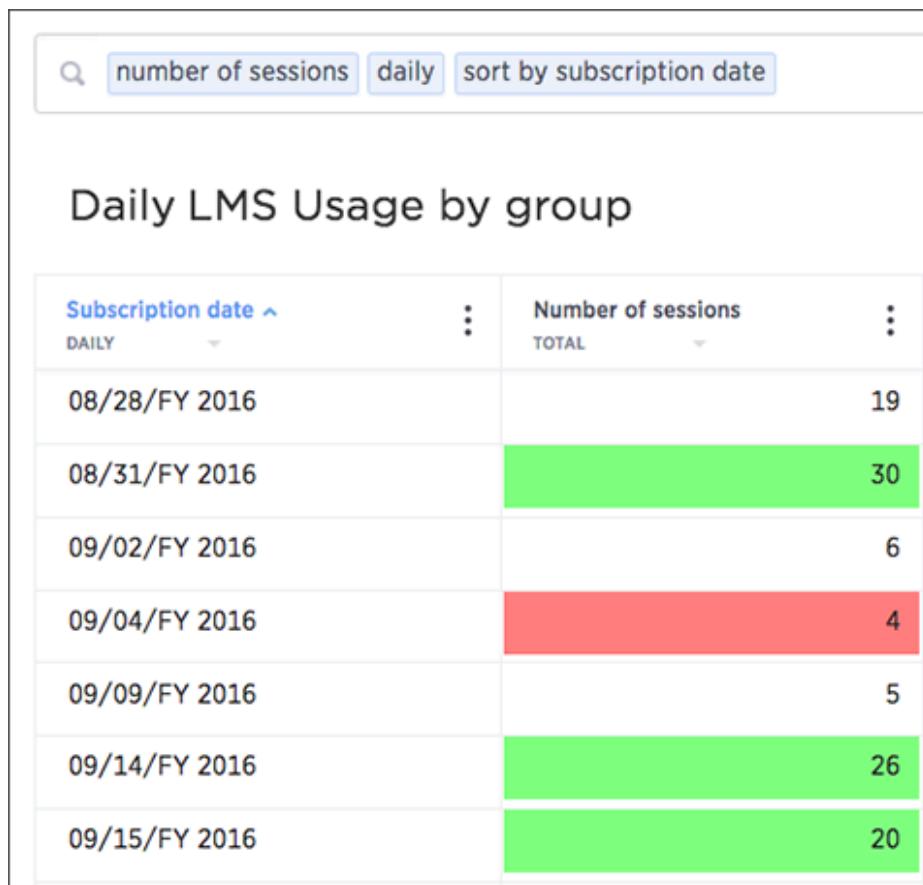
2. Select **Conditional Formatting**.
3. Click the + icon in the **Conditional Formatting** menu.



4. Define the sets of values and the color to use for each set.



5. Click **Done** after defining all of your conditional formatting sets.



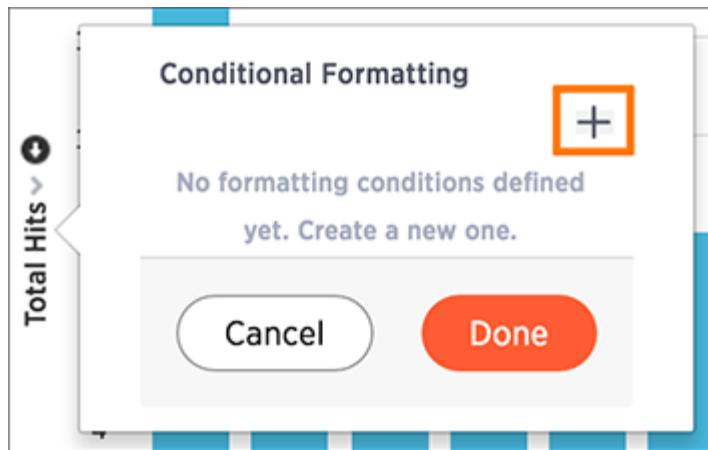
6. Note that if you change to a chart type, you will need to apply conditional formatting again. Conditional formatting is tied to the specific visualization.

## Apply conditional formatting to a chart

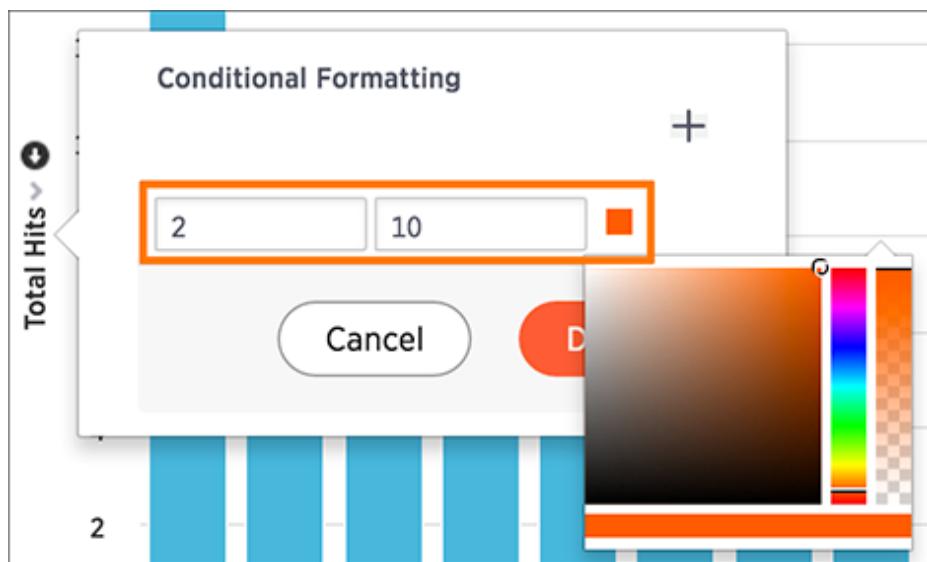
You can use conditional formatting to show charts with a target value or range drawn as a line in the chart, and the legend colors determined by where values fall relative to the target.

To apply conditional formatting to a chart:

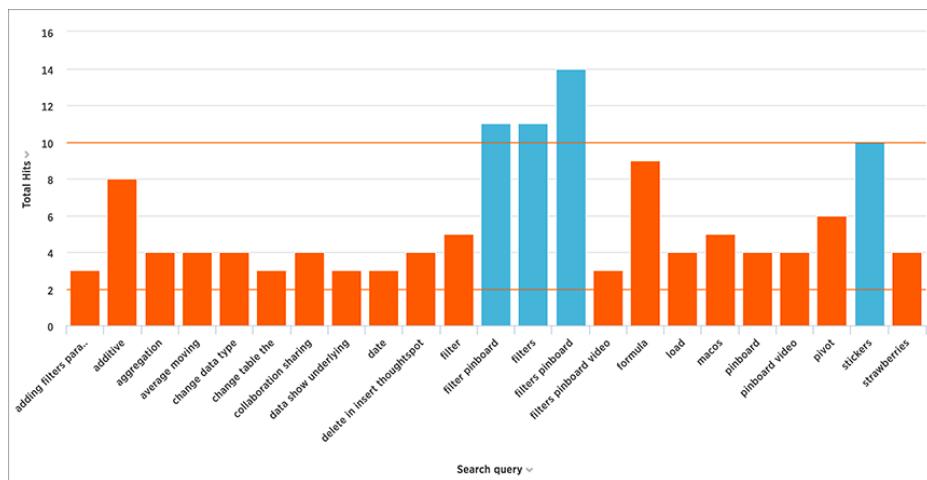
1. Click the axis label of your chart. Select **Conditional Formatting**.
2. Click the + icon in the Conditional Formatting menu.



3. Define the ranges of values and the color to use for each range you want to track. To add another range, click the + icon and repeat.



4. Click **Done** after defining all of your conditional formatting ranges.



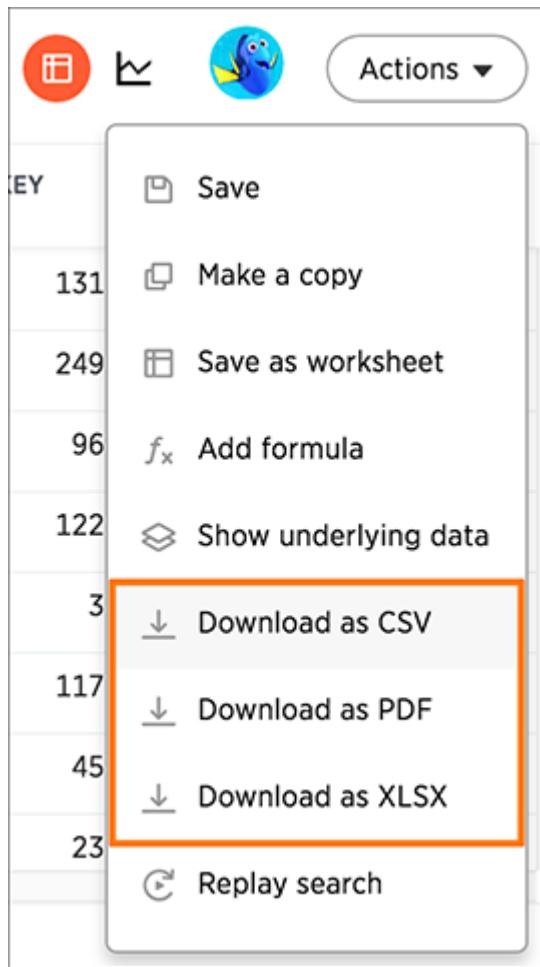
5. Note that if you change the chart type, you will need to apply conditional formatting again. It is tied to the specific visualization.

# Download your search

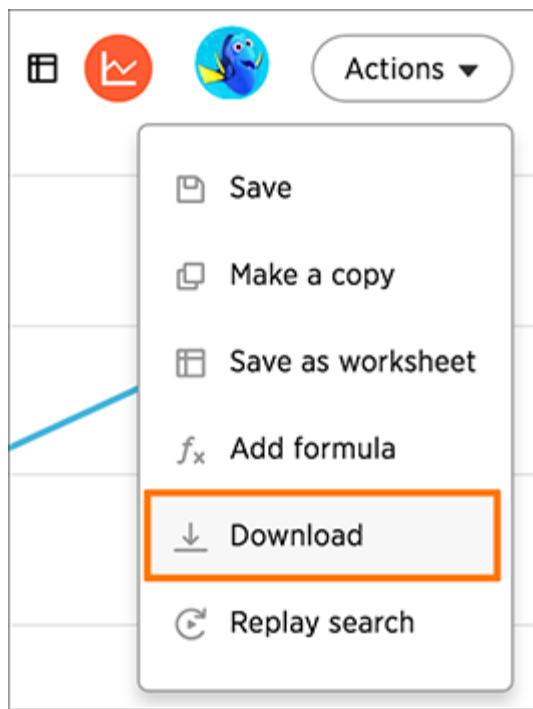
**Summary:** You can download your search as either a table or chart.

You have the option to download your table as a CSV (comma separated values), PDF, or XLSX (Excel) file. When you download a chart, it will be a PNG file. To download your search:

- Click **Actions**. For a table, choose between **Download as CSV**, **Download as PDF**, or **Download as XLSX**.



- For a chart, select **Download**.



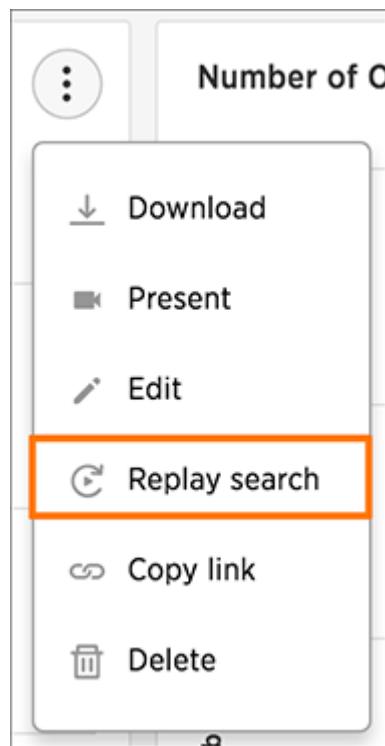
Currently, ThoughtSpot limits the maximum number of rows that can be downloaded to 10M (default is 1M). Your system limit is configured by your ThoughtSpot administrator.

# Replay search

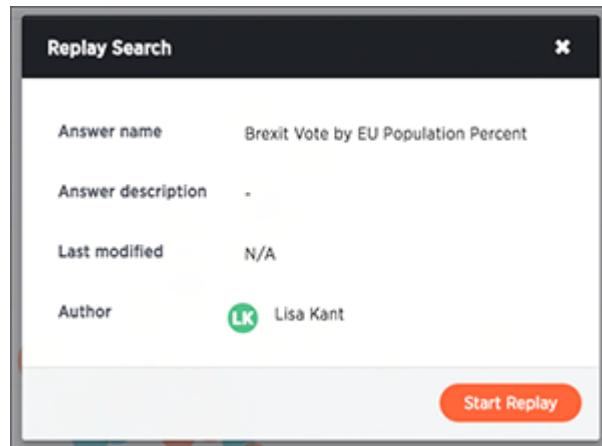
**Summary:** You can instantly generate a step-by-step replay showing the creation of a table or chart.

The replay feature shows how to create the chart or table you are viewing. Use it to teach yourself, or take a screencam of it and create your own ThoughtSpot training for your team.

1. When viewing a chart or table, click **Actions** and select **Replay search**.



2. Then, select **Start Replay** to view the video.



The replay will start automatically. You can pause and resume it by clicking on it.

# Understand charts

**Summary:** Charts display your search answer in a visual way.

Your search needs at least [one attribute and one measure](#) to be presented as a chart. When you choose to display your answer as a chart, ThoughtSpot will assign it the best fit chart type.

**Note:** Colors are maintained across searches within a session. For example, when doing a search on revenue by state, each state will keep its color assigned to it even if you change the search or chart type.

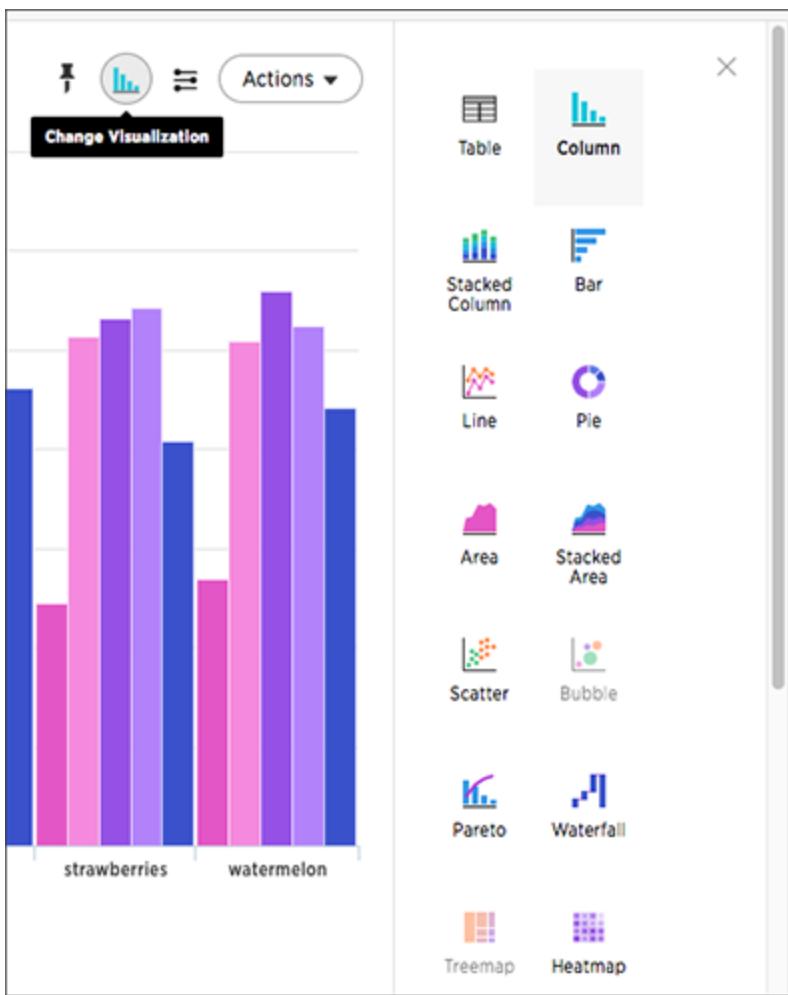
You can choose from a large number of chart types in ThoughtSpot. Each chart type provides you with a different visualization for your answer.

You can also [adjust the axes, labels, and view of the chart](#).

## About chart types

You can choose from a large number of chart types in ThoughtSpot. Each chart type provides you with a different visualization for your answer.

You can change the chart type of your answer by clicking **Change Visualization**.



**Note:** Some chart types may be unavailable for you to select depending on the columns of your search. For example, if your search does not contain at least one geographical column then you will not be able to select any of the geo chart types. Unavailable chart types are grayed out. Hovering over one will tell you what columns are needed before you can choose it.

- **Column charts**

The column chart is one of ThoughtSpot's simplest, yet most versatile chart type. More often than not, the column chart will be chosen as your default chart type.

- **Bar charts**

The bar chart is very similar to the column chart. The only difference is that it is oriented the other way.

- **Line charts**

Like the column chart, the line chart is one of ThoughtSpot's simplest, yet most versatile chart type. More often than not the line chart will be chosen as your default chart type.

- **Pie charts**

The pie chart is a classic chart type that displays your search in a circle. The pie chart ThoughtSpot shows is in the shape of a doughnut.

- **Area charts**

The area chart is based on the line chart, but has filled in regions.

- **Scatter charts**

The scatter chart is useful for finding correlations or outliers in your data.

- **Bubble charts**

The bubble chart displays three dimensions of data with each containing a set of values.

- **Pareto charts**

The pareto chart is a type of chart that contains both columns and a special type of line chart.

- **Waterfall charts**

The waterfall chart is used to show how an initial value is affected by a series of intermediate positive or negative values.

- **Treemap charts**

The treemap chart displays hierarchical data as a set of nested rectangles.

- **Heatmap charts**

The heatmap chart displays individual data values in a matrix following a color scale.

- **Line column charts**

The line column chart combines the column and line charts.

- **Funnel charts**

The funnel chart shows a process with progressively decreasing proportions amounting to 100 percent in total.

- **About geo charts**

There are three geo charts that let you visualize geographical data in ThoughtSpot.

- **About pivot tables**

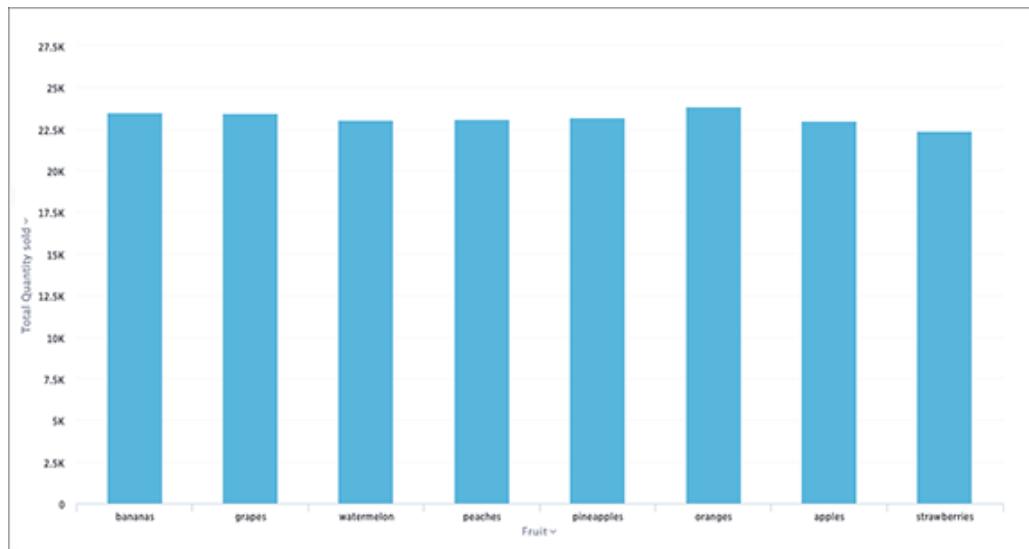
Pivot tables in ThoughtSpot use the well known drag-and-drop interface. Creating a pivot table enables exploring alternate visualization of data in a wide table. The basic idea is that some data is easier to consume when laid out horizontally, while others, vertically.

# Column charts

**Summary:** A column chart is the most versatile chart type

The column chart is one of ThoughtSpot's simplest, yet most versatile chart type. More often than not, the column chart will be chosen as your default chart type.

Column charts are vertical bar charts that display your data using rectangular bars. The length of the bar is proportional to the data value.



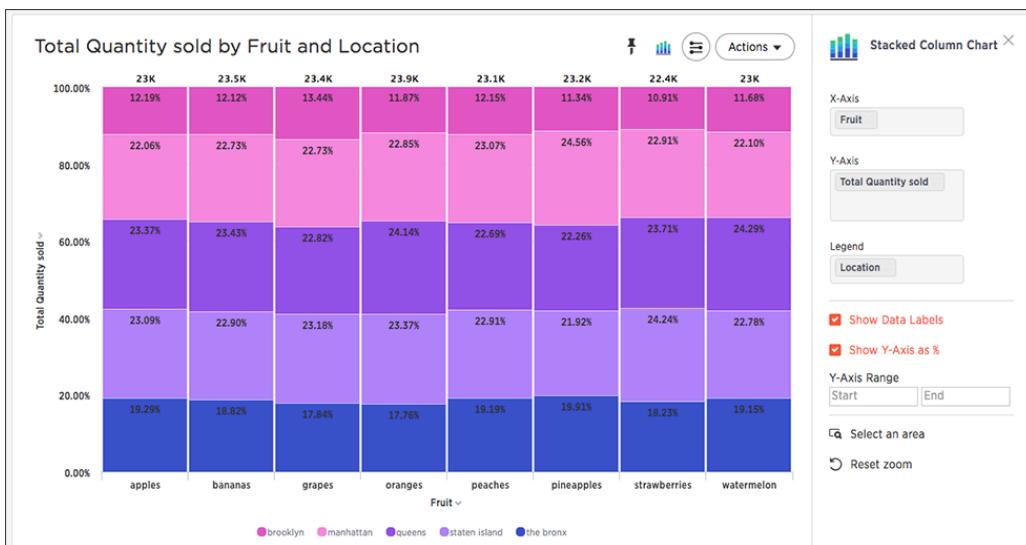
Your search needs at least one attribute and one measure to be represented as a column chart.

## Stacked column charts

The stacked column chart is similar to the column chart, but with one major difference. It includes a legend, which divides each column into additional sections by color.

Stacked column charts are typically used when you want to compare aggregated data and the data that it includes together. This chart type benefits when you add the **Additional chart options** option. Turning it on will show the sum of the stacks at the top of each stack.

It is important to note that stacked column charts plot the y-axis as a percentage by default. You can choose to toggle **Show Y-Axis as %** on or off in the Configuration Options. This feature is also available for stacked area charts.

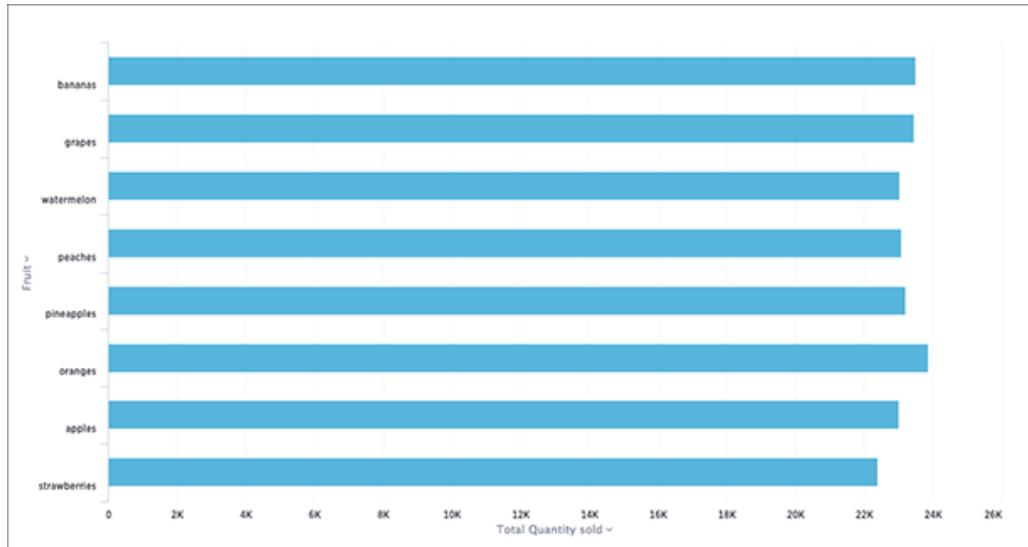


Your search needs at least two attributes and one measure to be represented as a stacked column chart.

# Bar charts

The bar chart is very similar to the column chart. The only difference is that it is oriented the other way.

Bar charts display your data using horizontal rectangular bars. The length of the bar is proportional to the data value.



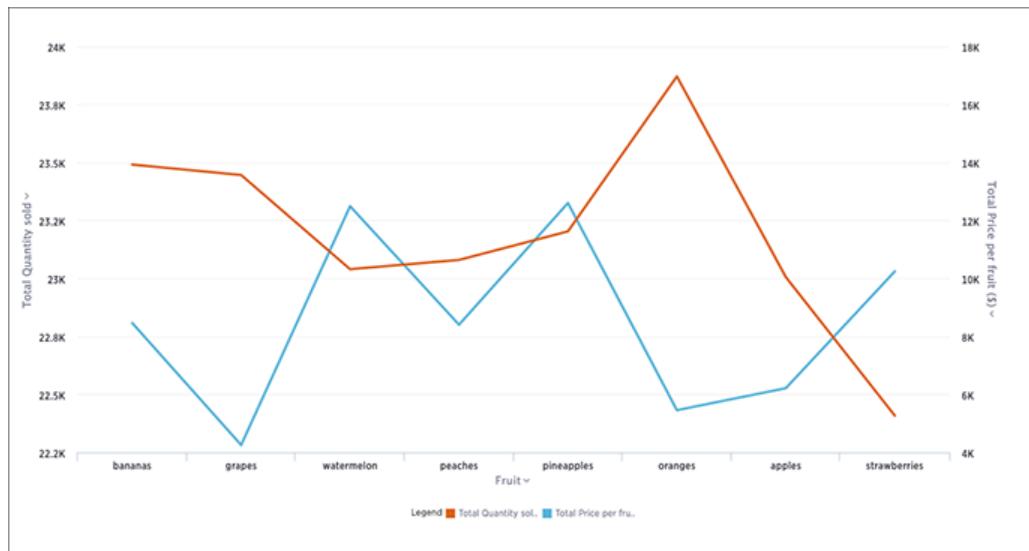
Your search needs at least one attribute and one measure to be represented as a bar chart.

# Line charts

**Summary:** Line charts are good at showing trends over intervals of time.

Like the column chart, the line chart is one of ThoughtSpot's simplest, yet most versatile chart type. More often than not the line chart will be chosen as your default chart type.

Line charts displays your data as a series of data points connected by straight line segments. The measurement points are ordered by the x-axis value.



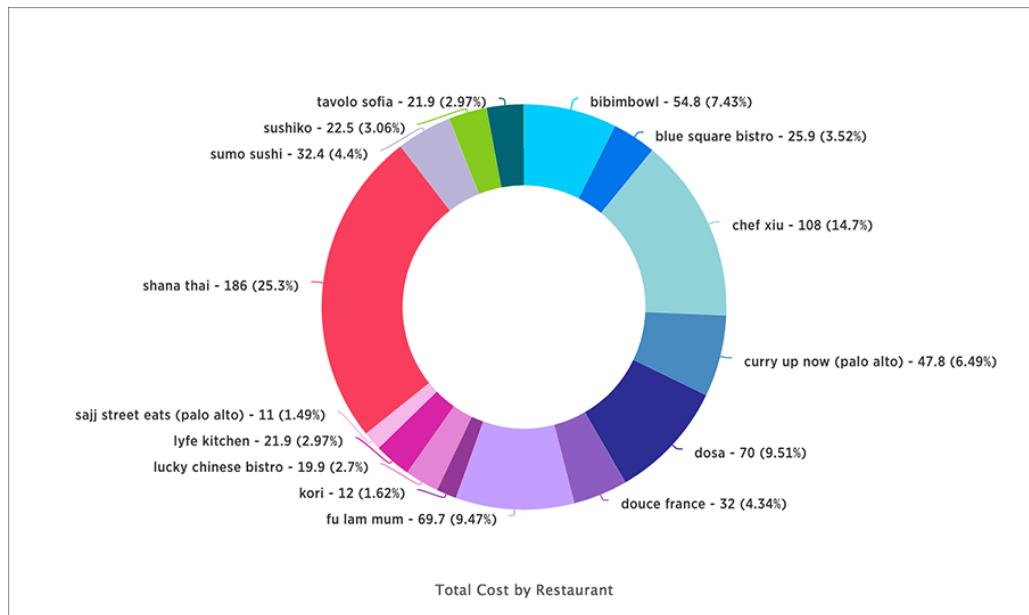
Your search needs at least one attribute and one measure to be represented as a line chart.

# Pie charts

The pie chart is a classic chart type that displays your search in a circle. The pie chart ThoughtSpot shows is in the shape of a doughnut.

## How pie charts divide data

Pie charts divide your data into sectors that each represent a proportion of a whole circle. You can display the exact values of each slice, in addition to the percentage values by toggling on **Additional chart options** found under Change chart configuration.

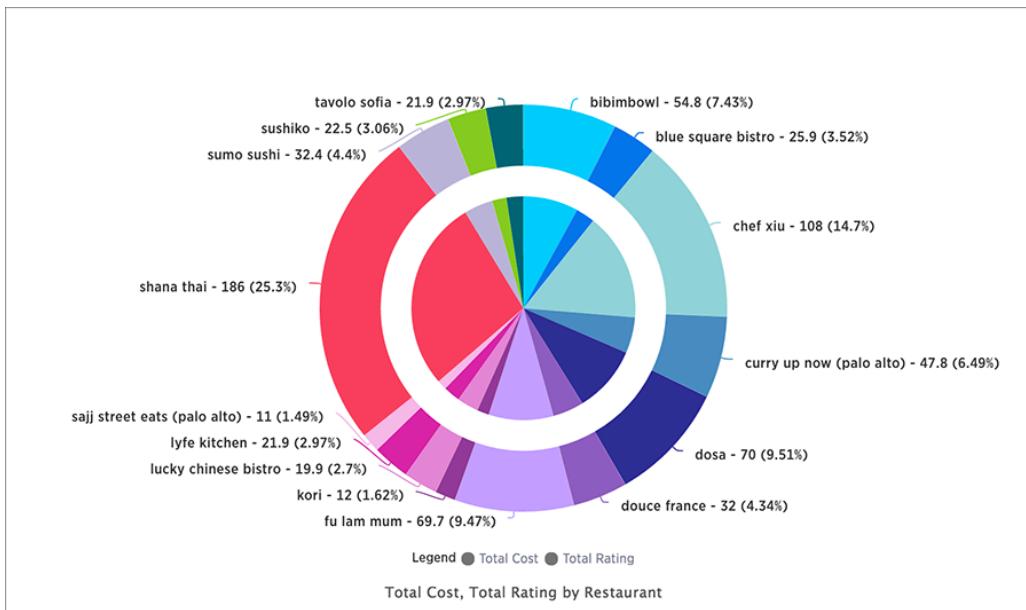


Your search needs at least one attribute and one measure to be represented as a pie chart. Also, there must be fewer than 50 values in the attribute column.

## Pie in pie charts

The pie in pie chart can be created from a regular pie chart in order to compare more than one component of an attribute. Pie in pie charts show two concentric pie charts comparing different measures.

To see a pie in pie chart, assign two different measures to the y-axis under **Configure Chart**.

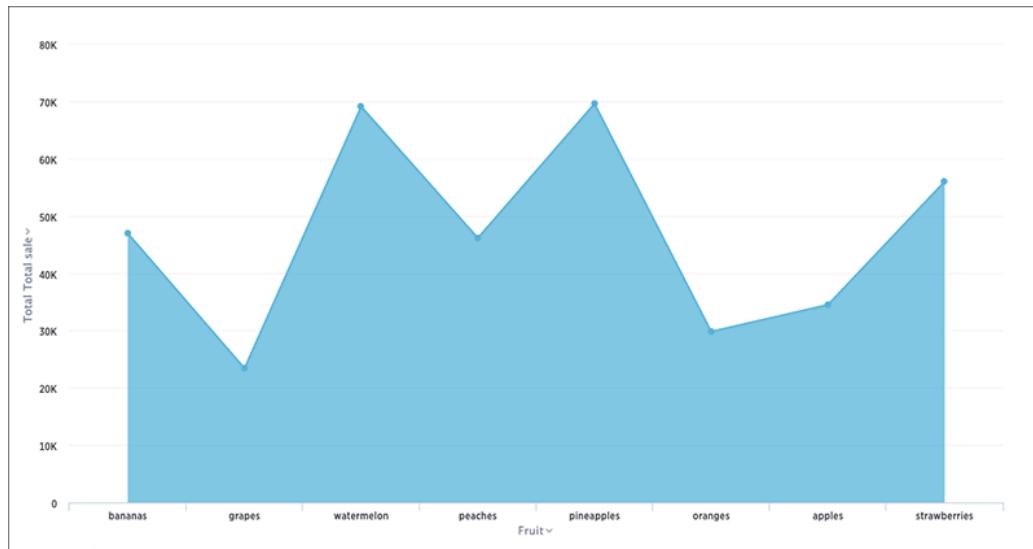


# Area Charts

The area chart is based on the line chart, but has filled in regions.

## Understand area charts

Area charts display quantitative data graphically. The area between the x-axis and the line are colored in to help you compare different portions of the chart.



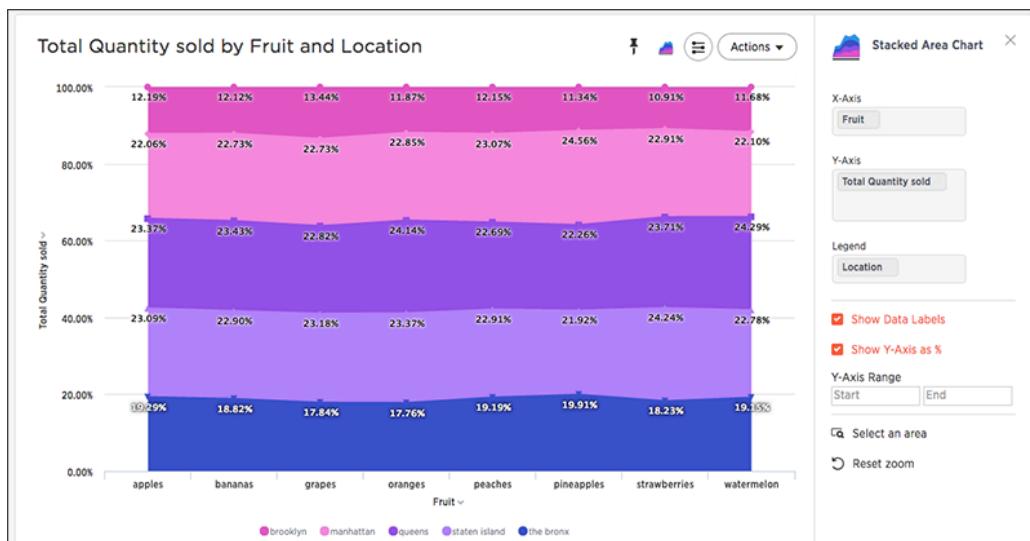
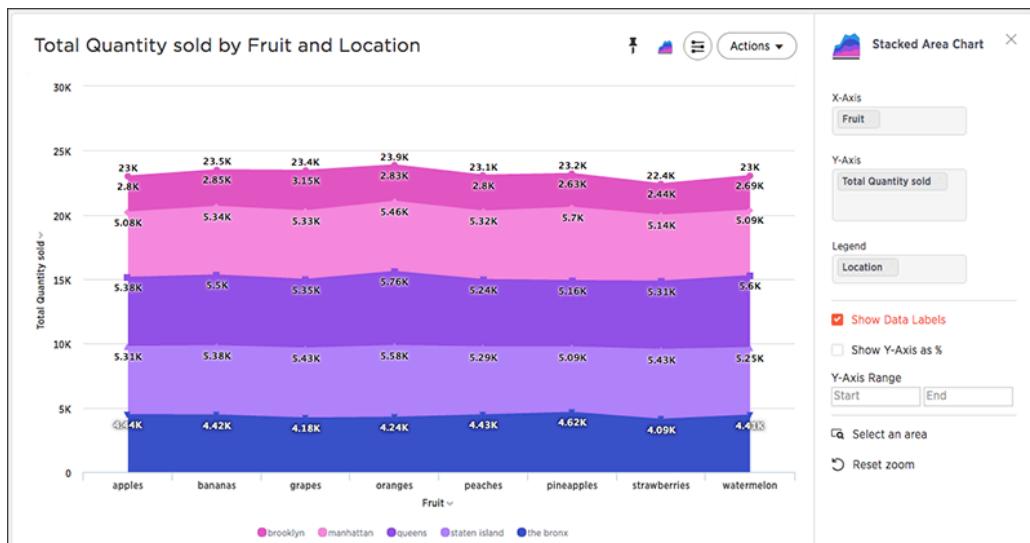
Your search needs at least one attribute and one measure to be represented as an area chart.

## Stacked area charts

The stacked area chart is an area chart with an attribute in the legend, which divides the area into layers.

Stacked area charts show the relative contribution to the accumulated total of a measure over time.

Stacked area charts plot the y-axis as a percentage by default. You can choose to toggle **Show Y-Axis as %** on or off in the **Configuration Options** to create your own mountain-style charts.

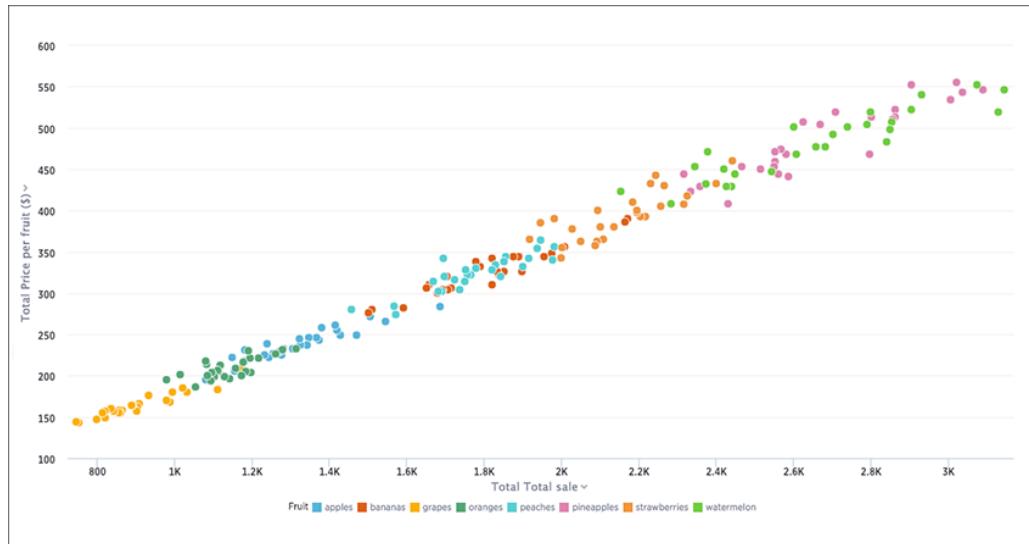


Your search needs at least two attributes and one measure to be represented as a stacked area chart.

# Scatter charts

**Summary:** The scatter chart is useful for finding correlations or outliers in your data.

Scatter charts display your data as a collection of points, which can either be evenly or unevenly distributed. Each point is plotted based on its own axes values. This helps you determine if there is a relationship between your searched columns.

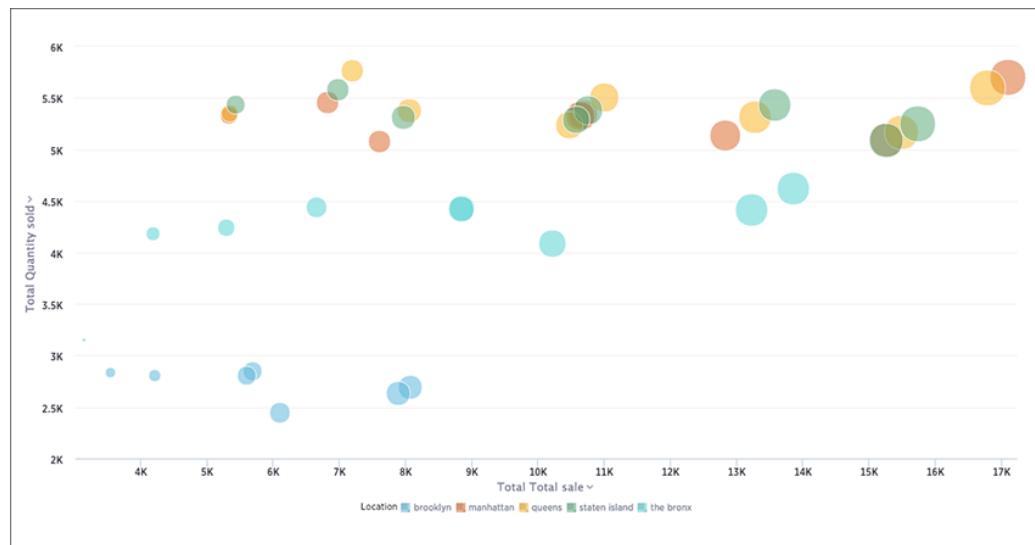


Your search needs at least one attribute and one measure to be represented as a scatter chart.

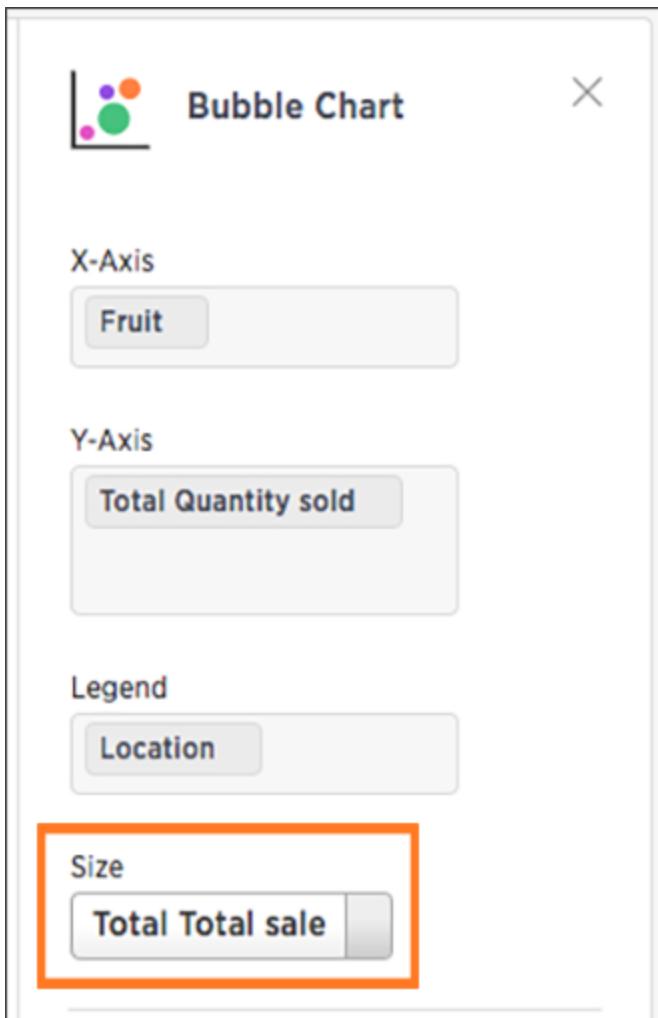
# Bubble charts

**Summary:** The bubble chart displays three dimensions of data with each containing a set of values.

The bubble chart is a variation of the scatter chart, with the data points replaced with bubbles. These bubbles add a third data dimension to your answer.



The size of each bubble depends on the measure you choose under **Edit chart configuration**.

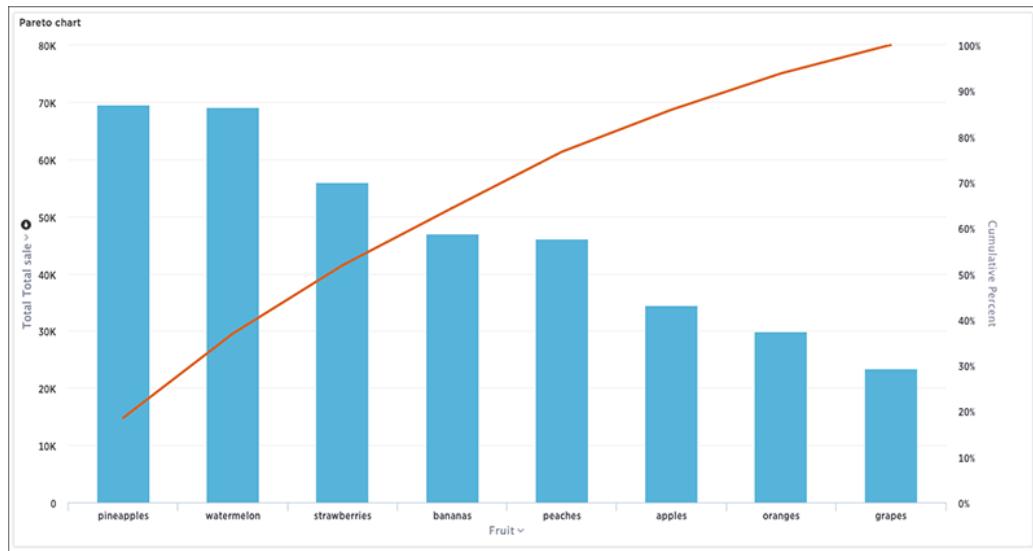


Your search needs at least one attribute and two measures to be represented as a bubble chart.

# Pareto charts

**Summary:** The pareto chart is a type of chart that contains both columns and a special type of line chart.

The individual values of a pareto chart are represented in descending order by columns, and the cumulative percent total is represented by the line. The y-axis on the left is paired with the columns, while the y-axis on the right is paired with the line. By the end of the line, the cumulative percent total reaches 100 percent.

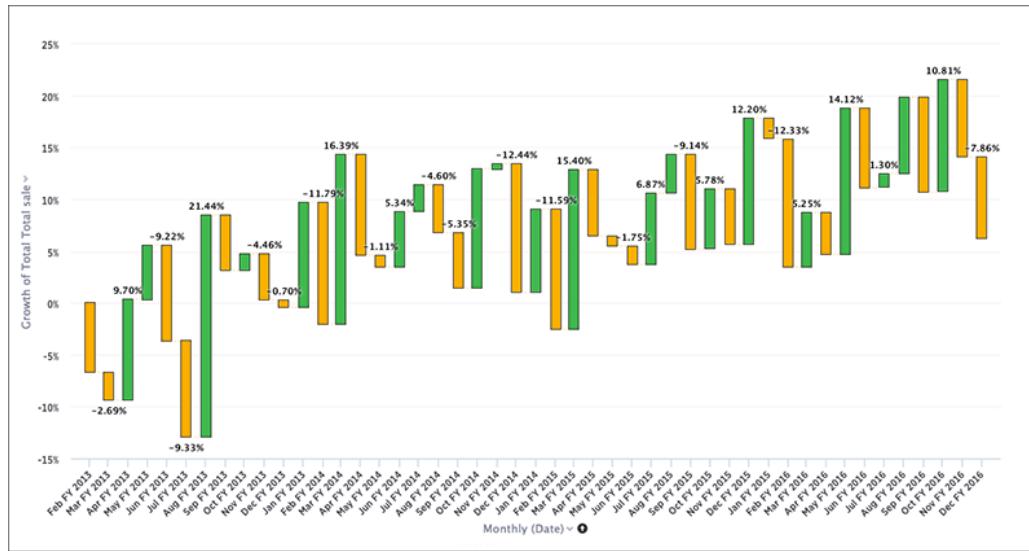


Your search needs at least one attribute and one measure to be represented as a pareto chart.

# Waterfall charts

**Summary:** The waterfall chart is used to show how an initial value is affected by a series of intermediate positive or negative values.

Waterfall charts are good for visualizing positive and negative growth, and therefore work well with the growth over time keyword. The columns are color-coded to distinguish between positive and negative values.

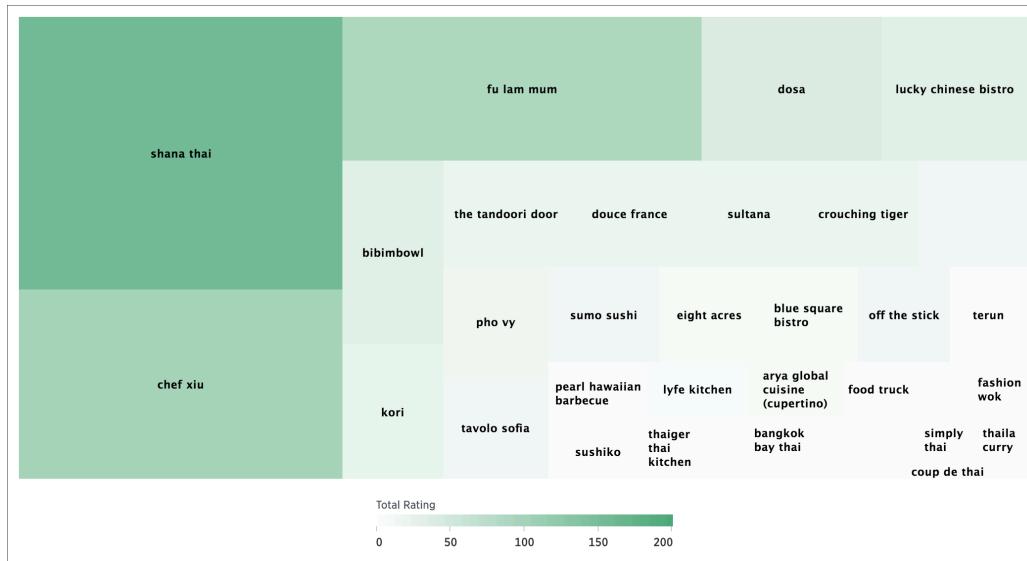


Your search needs at least one attribute and one measure to be represented as a waterfall chart.

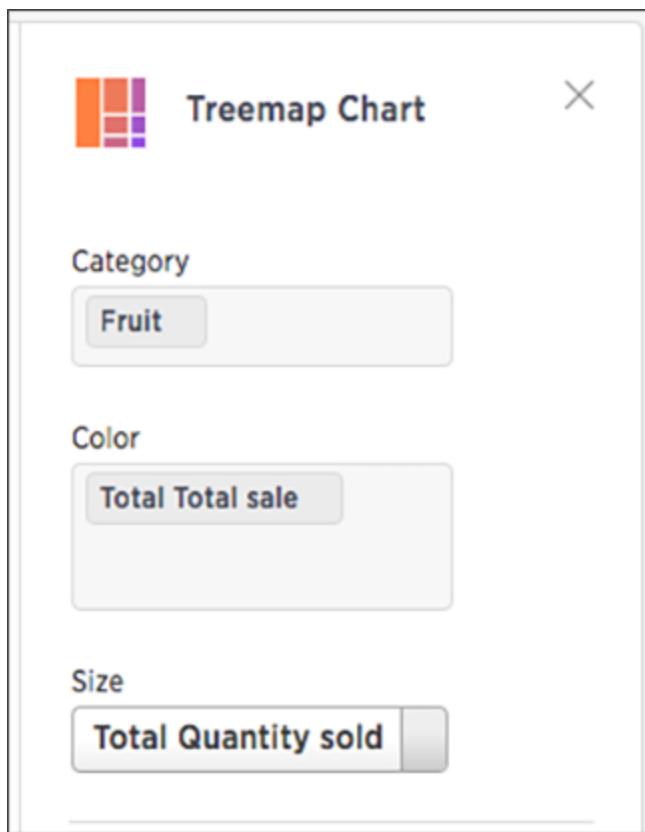
# Treemap charts

**Summary:** The treemap chart displays hierarchical data as a set of nested rectangles.

Treemap charts use color and rectangle size to represent two measure values. Each rectangle, or branch, is a value of the attribute. Some branches can contain smaller rectangles, or sub-branches. This setup makes it possible to display a large number of items in an efficient way.



You can rearrange the columns of your search into category, color, and size under [Edit chart configuration](#).



Your search needs at least one attribute and two measures to be represented as a treemap chart.

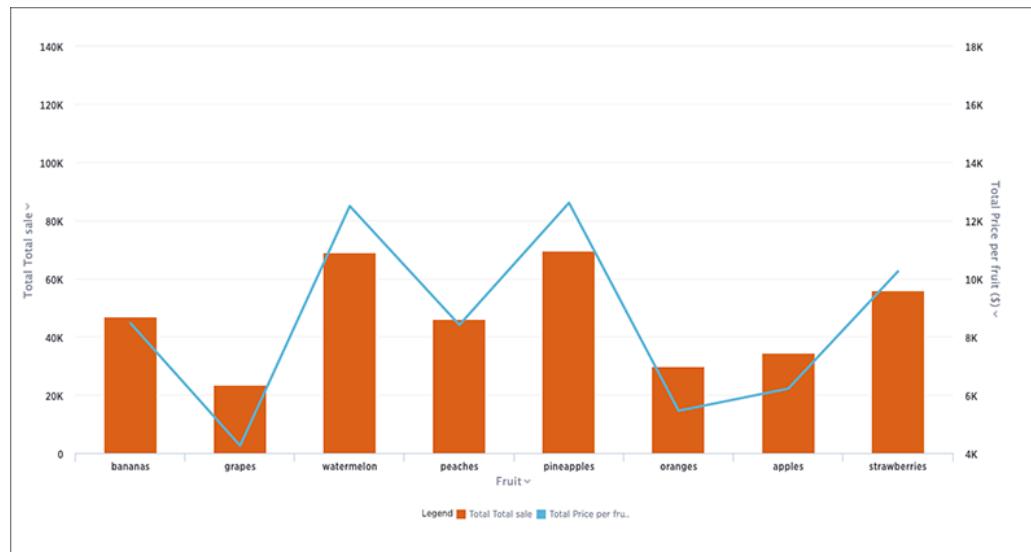
# Line column charts

**Summary:** The line stacked column chart combines stacked column and line charts.

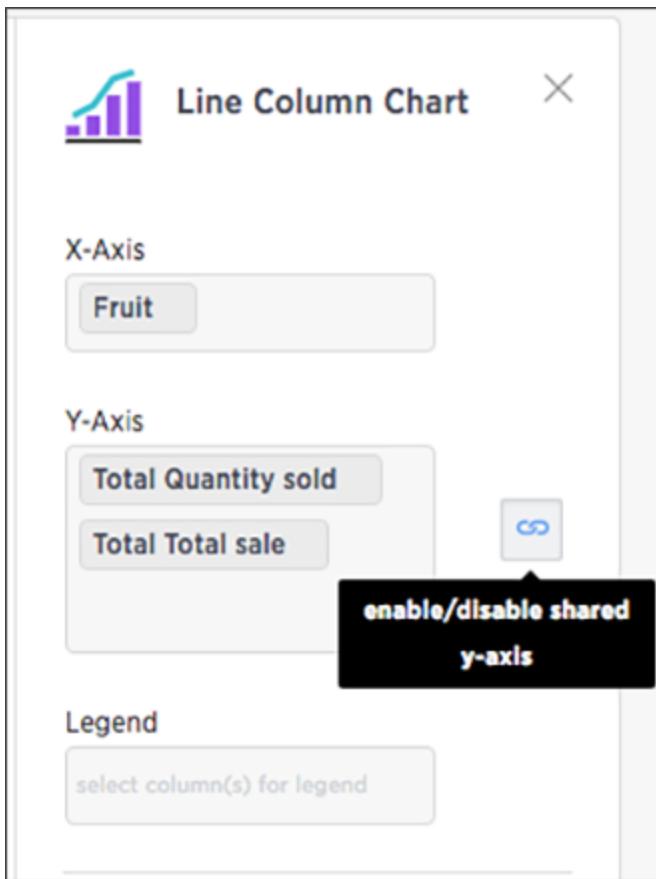
The line column chart combines the column and line charts. Your search needs at least one attribute and two measures to be represented as a line column chart.

## Understand line column charts

Line column charts display one measure as a column chart and the other as a line chart. Each of these measures has its own y-axis.

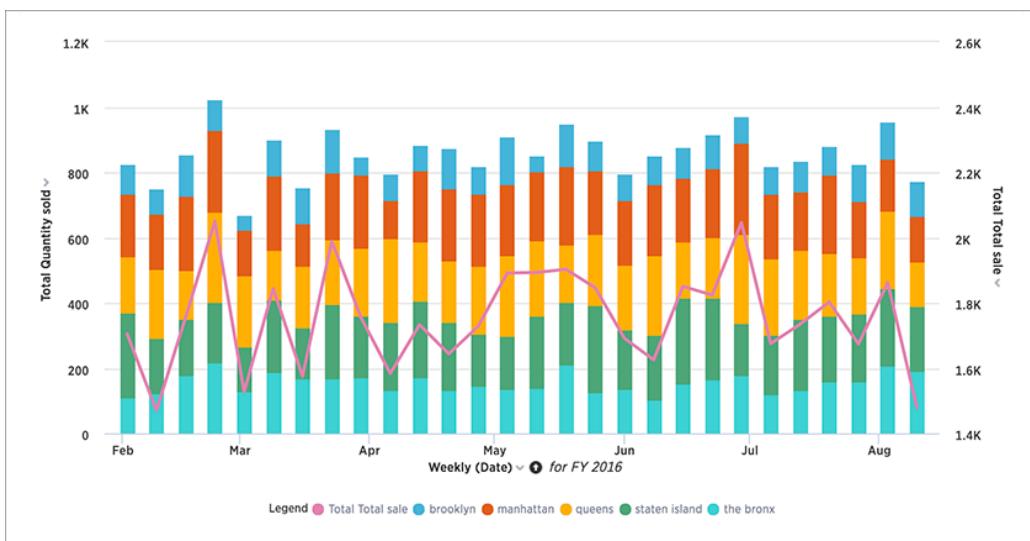


You can choose to enable shared y-axis by clicking the link icon found under **Edit chart configuration**.

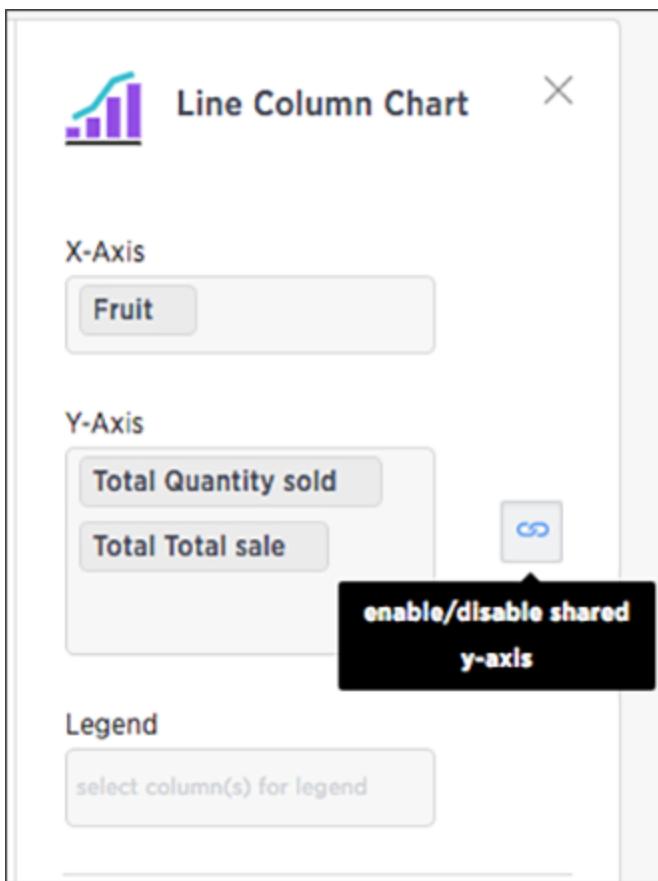


## Line stacked column charts

This chart is similar to the line column chart, except that it divides its columns with an attribute in the legend. There are two y-axes, one for each measure.



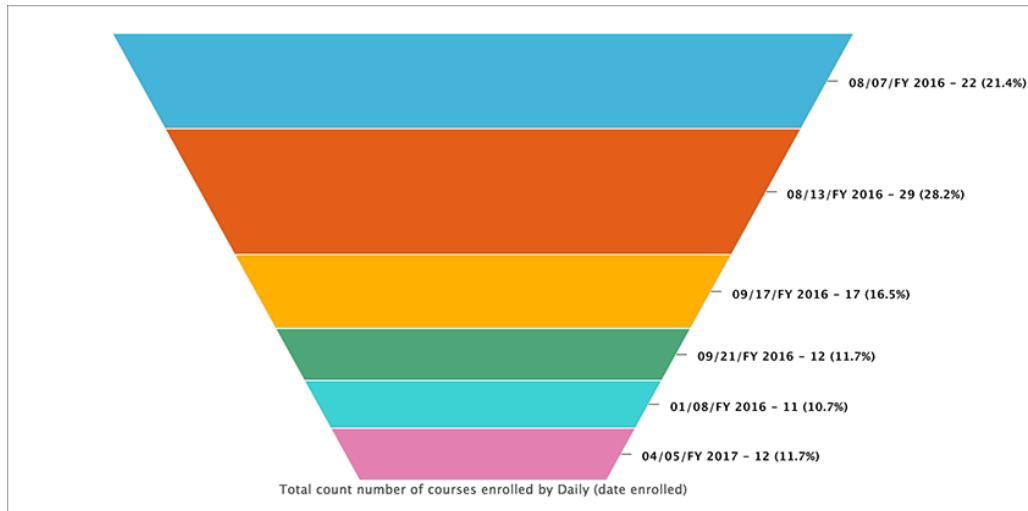
You can choose to enable shared y-axis by clicking the link icon found under **Edit chart configuration**.



# Funnel charts

**Summary:** The funnel chart shows a process with progressively decreasing proportions amounting to 100 percent in total.

A funnel chart is similar to a stacked percent column chart, and is often used to represent stages in a sales process. You can visualize the progression of data as it passes from one phase to another. Data in each of these phases is represented as different proportions.



Your search needs at least one attribute and one measure to be represented as a column chart. The attribute must contain 50 or fewer values.

# Geo charts

**Summary:** There are three geo charts that let you visualize geographical data in ThoughtSpot.

Geo charts show data on a map by location. They are geo area, geo bubble, and geo heatmap charts. And these geo charts can display six types of geographical data, which are:

- Country
- State
- County
- Zipcode
- Point (latitude/longitude)
- Other sub-nation regions (for international countries)

In addition to the United States, ThoughtSpot also supports maps for the Canadian province/count maps, United Kingdom, Germany, Sweden, South Africa, and France.

## Geo data that can be displayed

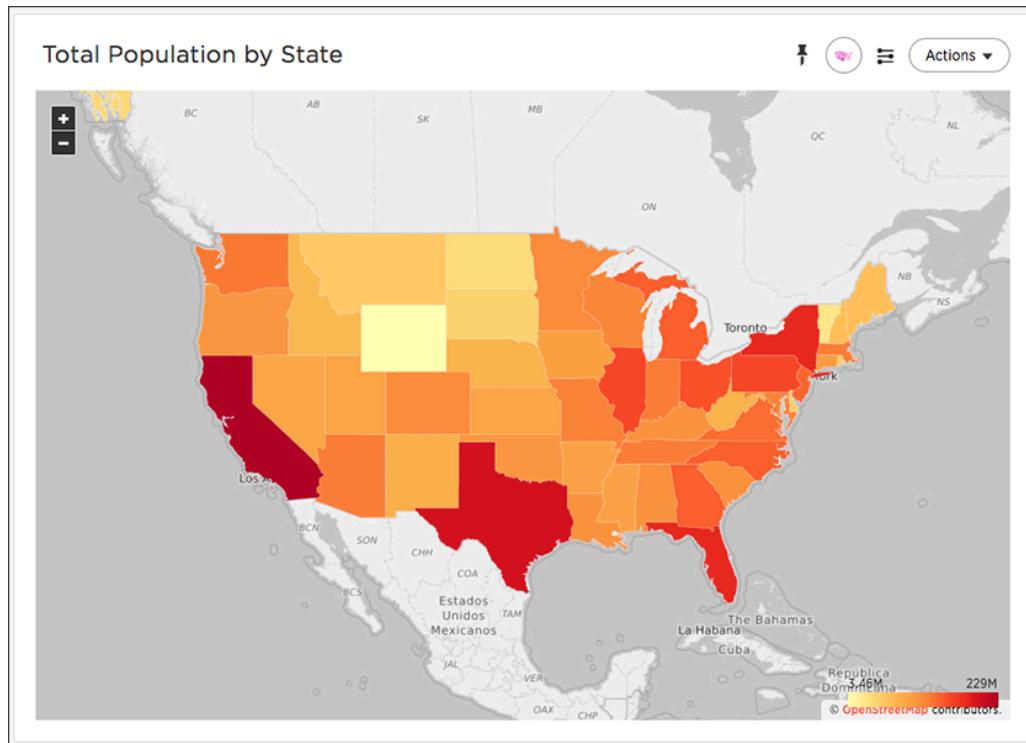
Here is a table that shows which GeoType data can be displayed using which geo chart type.

GeoType	Geo chart type	Notes
Country	Geo area (default), geo bubble, geo heatmap	Can also be regions.
County	Geo area (default), geo bubble, geo heatmap	Only for counties in the United States.
Point	Geo bubble (default), geo heatmap	Must use both latitude and longitude columns.
State	Geo area (default), geo bubble, geo heatmap	Only for states in the United States.
Zipcode	Geo bubble (default), geo heatmap	Zip codes and zip codes +4 in the United States.
Other sub-nation regions	Geo area (default), geo bubble, geo heatmap	The display will depend on the type of administrative region chosen.

For data to be displayed using geo charts, your administrator must configure it as geographical data. If you are expecting to be able to get a map visualization, but it isn't available, contact your administrator so they can make the configuration.

## Area charts

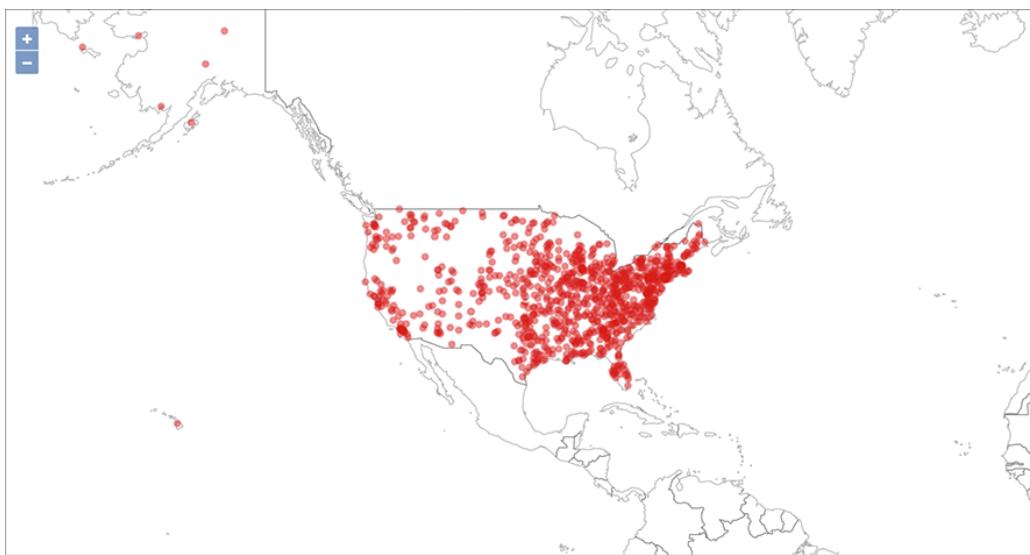
Geo area charts highlight the regions of interest. Point data (longitude/latitude) doesn't work on geo area charts. Also, only geo area charts display boundaries for counties.



Your search needs one geographical column of granularity to be represented as a geo area chart.

## Geo bubble charts

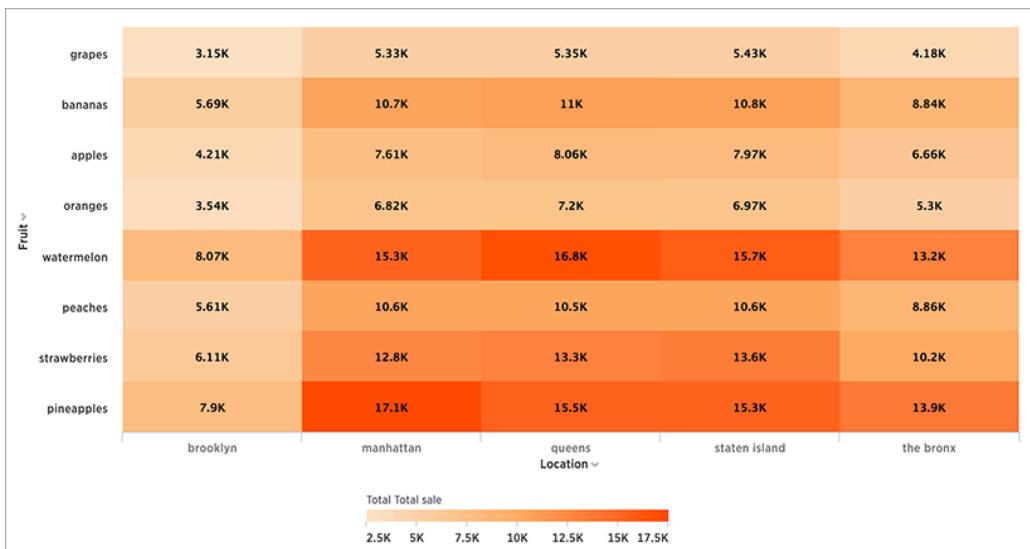
Geo bubble charts, like bubble charts, display the value of the measure by the size of the bubble. Zip code data makes the most sense for geo bubble charts.



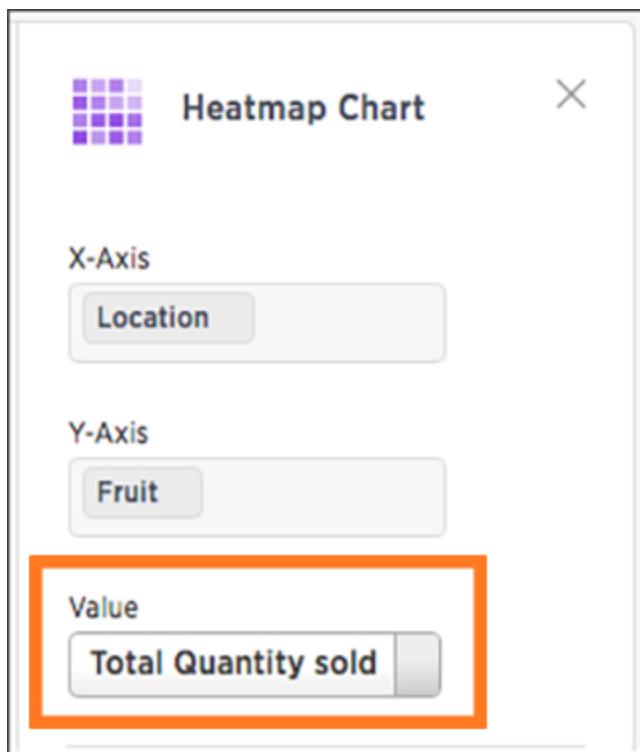
Your search needs one geographical column or a pair of latitude and longitude columns to be represented as a geo bubble chart.

## Heatmap charts

Heatmap charts are similar to treemap charts in that they both use a similar system of color-coding to represent data values. However, the heatmap does not use size to measure data, and instead requires an additional attribute.



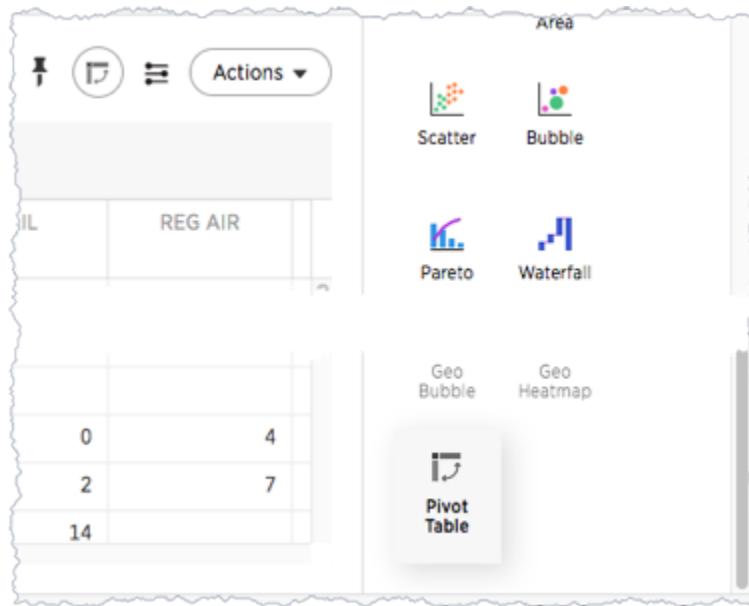
The value of each cell depends on the measure you choose under **Edit chart configuration**.



# Pivot table

**Summary:** A pivot table is a chart type.

Creating a pivot table enables exploring alternate visualization of data in a wide table. It is useful especially when improve data display when some data is best viewed horizontally, while others, vertically. Pivot tables are a chart time table that use a drag-and-drop interface.



If a **Pivot Table** is available for your answer, you see this type of visualization as an option under the visualization. Add rows, measures, and columns to the search bar:

Pivot table with multiple measures			
Total Revenue	Total Discount	Market Segment	Customer Region
Quarterly (Order..)	Order Priority	▶ automobile	
		Total Revenue	Total Discount
▼ Q1 1992	1-urgent	38.6M	68
	2-high		
	3-medium	23.1M	22
	4-not specified	5.12M	10

Pivot table with multiple measures			
Total Revenue	Total Discount	Market Segment	Customer Region
Quarterly (Order..)	Order Priority	▶ automobile	
		Total Revenue	Total Discount
▼ Q1 1992 Total	1-urgent	98.1M	137
	2-high		
	3-medium	23.1M	22
	4-not specified	5.12M	10

You can restructure your pivot table by moving these values under **Configure Chart** or by dragging and dropping column headings on the table itself. If you right click on a the row headings, the system displays a contextual sort menu:

Quarterly (Order..)	Order Priority	▶ automobile		▶ building		▶ furnitu	
		Total Revenue	Total Discount	Total Revenue	Total Discount	Total Revenue	
▼ Q1 1992	1-urgent	332	47.3M				
	2-high	67	3.29M				
	3-medium	99	9.3M				
	4-not spe	87	27.6M				
	5-low	64					
► Q2 1992		15	7.13M				
		195	34.2M				
		62.9M	84	171M			

## Expanding or contract columns and rows

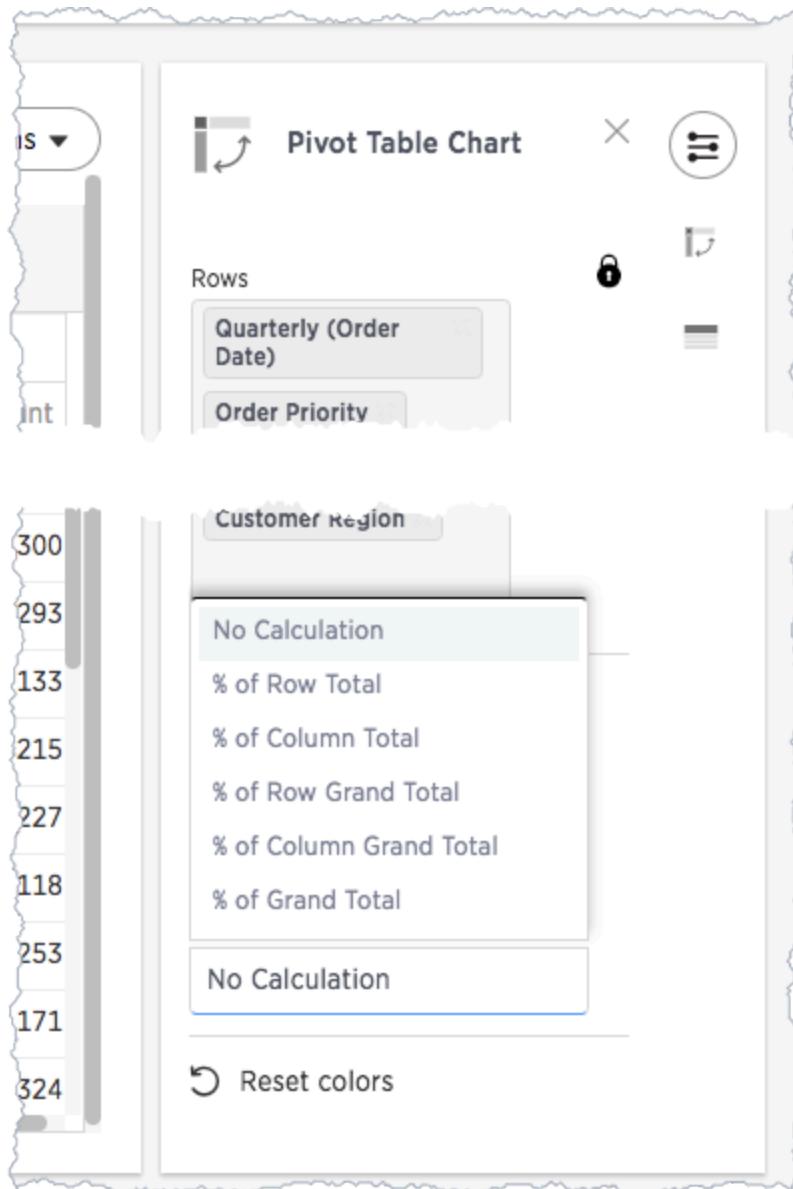
Click a column or row to expand it. Additionally, you can expand or collapse all by right clicking the arrow on the top left of a cell.



When you pin a pivot table to a pinboard, it will retain your expansion settings.

## Display totals

You can show **% Row Grand Total** and, **% of Column Grand Total**, or **% of Grand Total** on pivot tables. Grand totals aggregate data of the entire pivot grid. They show values summarized across all available data.



You only see 100% value when grand total rows is enabled. This is because when it is disabled, the rows and columns they have no parental total column and the percentage cannot be calculated. All intermediate total (columns or rows) display the percentage values calculated with respect to their parent. For each inner summary (column or row) the parental total values are assumed to be 100% internally.

To see the total column make sure you also have **Pivot Summaries** checked.

Total count Medal		Country	Medal							
Sport	▼ United States			United States Total	▼ United Kingdom			United Kingdom Total	B	
	Gold	Silver	Bronze		Silver	Bronze	Gold			
Wrestling	5.45%	4.69%	3.49%	13.63%	0.44%	1.09%	0.33%	1.85%		
Weightlifting	3.64%	3.64%	2.50%	9.77%	0.68%	0.68%	0.23%	1.59%		
Water Motorsports	{Null}	{Null}	{Null}	{Null}	{Null}	{Null}	80.00%	80.00%		
Volleyball	6.87%	3.88%	3.88%	14.63%	{Null}	{Null}	{Null}	{Null}		
Taekwondo	5.68%	5.68%	5.68%	17.05%	19.32%	10.23%	19.32%	48.86%		

## Format row headers

For row headers, you can switch between the default view and a more compact (tree) layout:

Pivot table with multiple measures			
Total Revenue	Total Discount	Market Segment	Customer Region
Quarterly (Order..)	Order Priority	► automobile	► automobile
		Total Revenue	Total Discount
▼ Q1 1992	1-urgent	38.6M	68
	2-high		
	3-medium	23.1M	22
	4-not specified	5.12M	10

Pivot table with multiple measures			
Total Revenue	Total Discount	Market Segment	Customer Region
Quarterly (Order..)	Order Priority	► automobile	► automobile
		Total Revenue	Total Discount
▼ Q1 1992 Total		98.1M	137
	1-urgent	38.6M	68
	2-high		
	3-medium	23.1M	22
	4-not specified	5.12M	10

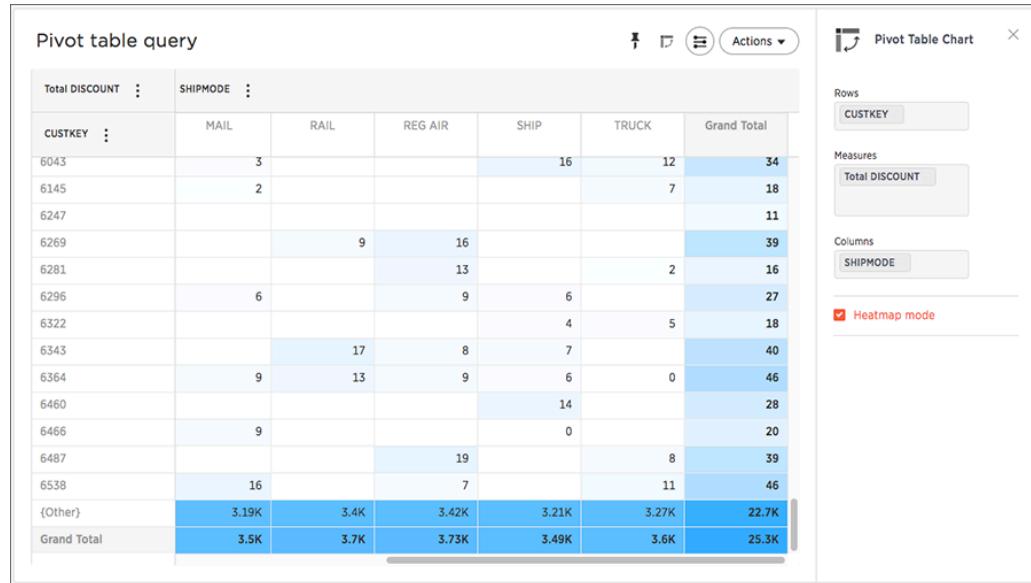
From the header menu users, change the type of total shown for an aggregated measure:

A screenshot of the ThoughtSpot interface showing a 'Pivot Table' card. A context menu is open over a row labeled 'Total Total Adhoc Impressions'. The menu options include: TOTAL, AVG, MIN, MAX, TOTAL COUNT, UNIQUE COUNT, STD DEVIATION, VARIANCE, FILTER ..., and SORT. The 'TOTAL' option is checked.

You can also **Remove** a headline.

## Heatmap of large contributors

The heatmap functionality allows you to see which measures contribute more than others (heat) within the table.



Toggle on the **Heatmap mode** found under **Configuration Options** to add color coordination to your data.

## Pivot table limitations

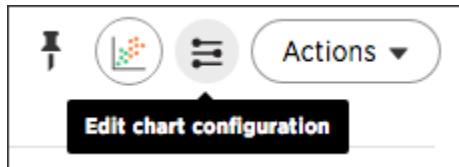
The pivot table chart type has these limitations:

- Only the first 10,000 data values are used. If you would like to increase this limit, please contact ThoughtSpot Support.
- Pivot table is not available if the dataset contains more than 100,000 rows.
- Cardinality is not configurable.
- Show underlying data menu is unavailable.
- Conditional formatting is not functional.

# Change a chart

**Summary:** You can adjust the axes, labels, and view of the chart.

Every chart gives you the option to move columns between the x- and y-axes, hide or Additional chart options, and zoom.



The configure chart option (the top icon which looks like a small bar chart) gives you the ability to edit the chart axes and legends. Click on the icon to view the chart axes, add a legend, lock a visualization, and more.

- **Reorder labels on the axis or legend**

When there are multiple columns on the x- or y-axis or legend of a chart, you can reorder them by using the **Configure Chart** icon.

- **Set the y-axis range**

You can manually set the y-axis range by using the **Edit chart configuration** icon.

- **Hide and show values**

You can hide and show values on the chart using the legend.

- **Change chart colors**

You can easily change the legend colors in a chart.

- **Additional chart options**

You can configure charts to show the y-axis data values.

- **Zoom into a chart**

You can zoom into your chart by selecting an area with your mouse.

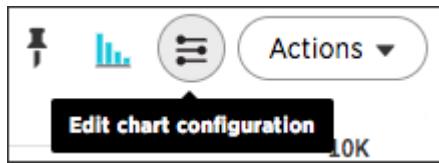
# Reorder labels on the axis or legend

When there are multiple columns on the x- or y-axis or legend of a chart, you can reorder them by using the **Configure Chart** icon.

The order in which columns appear on the axis or legend is based on the sequence they are added. The first field is used as the primary sorting field for the chart. Adding another field adds it after the first one on the axis label or legend. If you want to change the order, you can remove the fields and re-add them in the reverse order.

This example shows you how to reorder the x-axis columns.

1. While viewing your answer as a chart, click **Edit chart configuration** near the top right.



2. In the X-Axis box, delete the values. Then re-add them in the new preferred order.



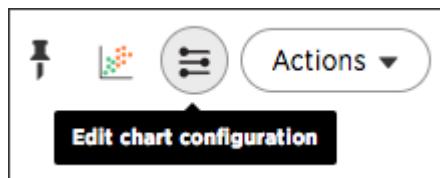
3. Click **Done**.

Your chart reorganizes itself to reflect the new label order.

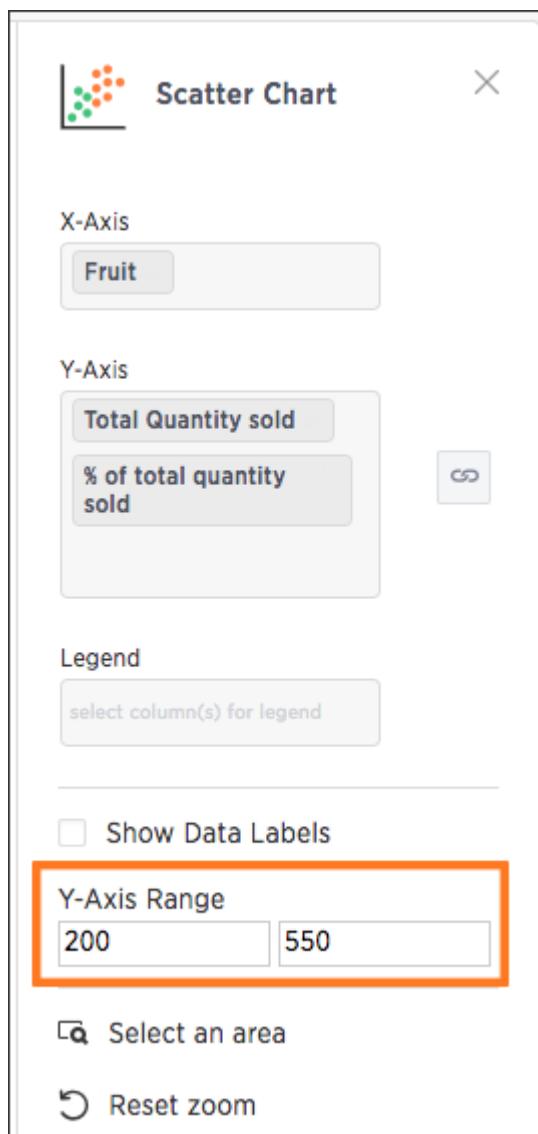
# Set the y-axis range

You can manually set the y-axis range by using the **Edit chart configuration** icon. The chart configuration panel includes the option to manually set the y-axis range. To set the y-axis range:

1. While viewing your answer as a chart, click **Edit chart configuration** near the top right.



2. Under **Y-Axis Range** enter your preferred start and end values.



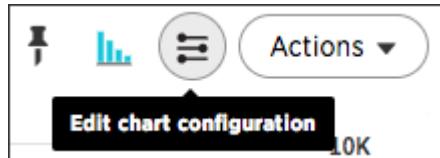
Your chart will reorganize itself to reflect the new y-axis range.

# Hide and show values

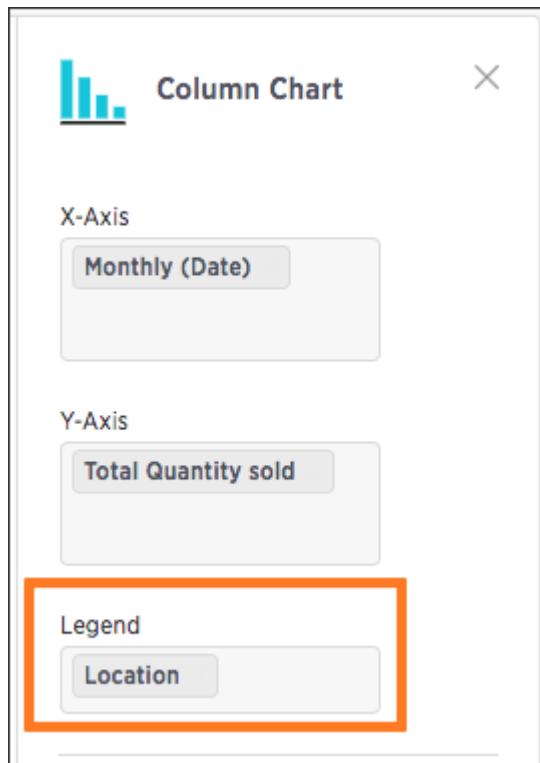
**Summary:** You can hide and show values on the chart using the legend.

Clicking on a legend item will hide it from the chart. You can click it again to show it. To hide and show values:

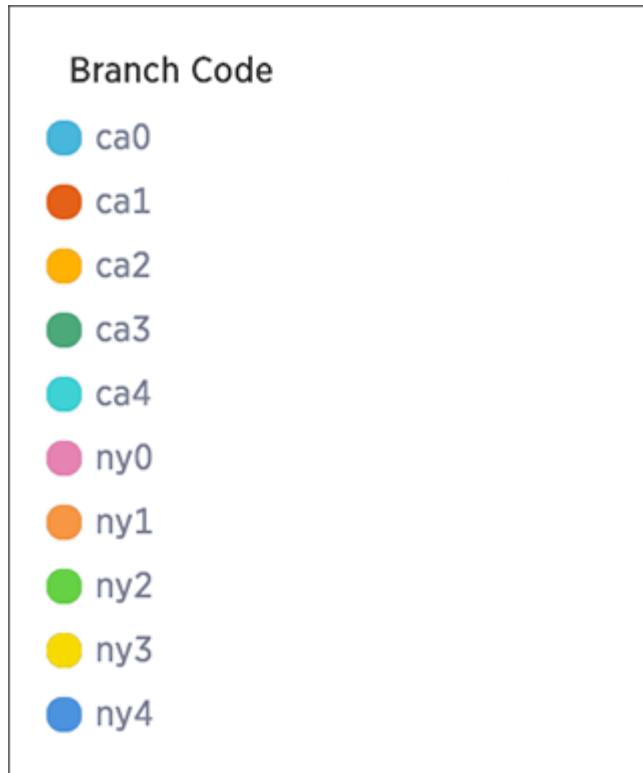
1. While viewing your answer as a chart, click **Edit chart configuration** on the right.



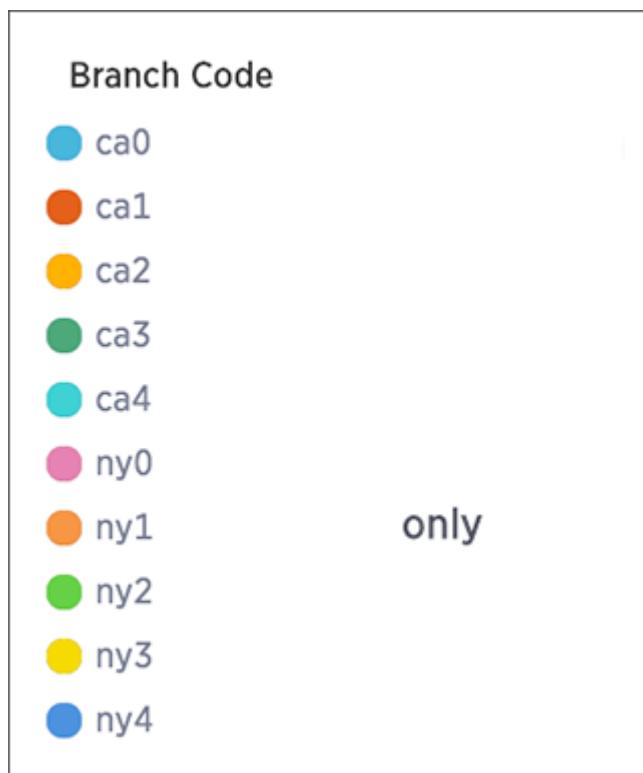
2. Make sure you have a column in the **Legend** field.



3. On the chart, click the legend value you would like to hide. Click the hidden legend value again to show it.



You can also click **only** next to a legend value to only show that value on the chart.



# Change chart colors

**Summary:** You can easily change the legend colors in a chart.

To change the colors used in a chart, you can use the color picker from the chart. The color you set is used in:

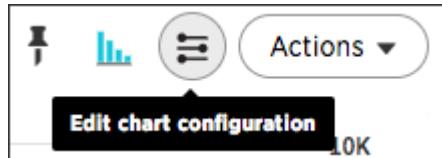
- legends on charts
- tables for conditional formatting
- tags and editing the colors

To change the chart colors, you need at least one column in the chart's legend. If the chart does not have a legend (not all do) and it is a single-measure chart, you can [apply a conditional format](#) to change the color. This type of conditional formatting does not work with Line charts.

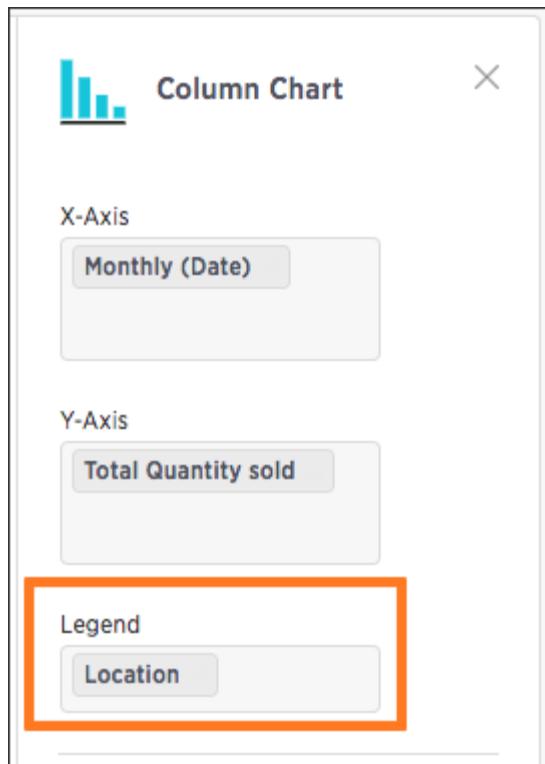
Colors are maintained across searches within a session. For example, when doing a search on revenue by state, each state will keep its color assigned to it even if you change the search or chart type.

## Add a column to the legend

1. While viewing your answer as a chart, click **Edit chart configuration** on the right.

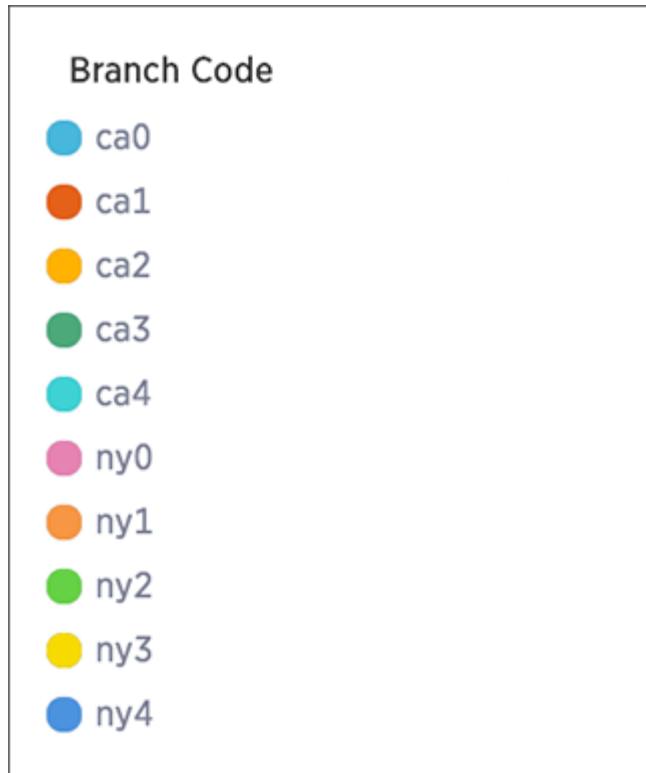


2. Make sure you have a column in the **Legend** field.



## Set a color through the legend

1. On the chart, click the color icon of the legend value you would like to change the color of.



2. Use the color picker to choose a new color to represent that legend value.

You can also enter a HEX value directly.



3. Click outside of the picker to dismiss the picker.
4. **Actions > Save** your changes.

# Additional chart options

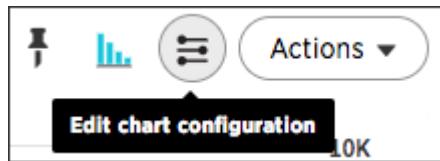
**Summary:** You can configure charts to show the y-axis data values.

Some charts have additional options under **Configuration Options** that are covered in each chart type topic.

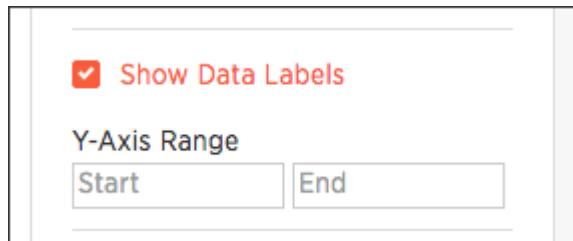
## Show data labels

To show data labels:

1. While viewing your answer as a chart, click **Edit chart configuration**.

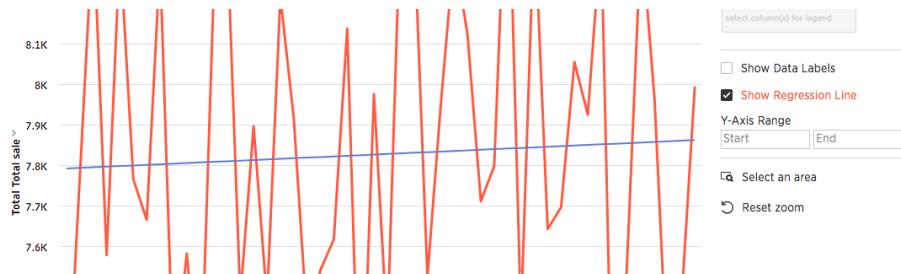


2. Select **Show Data Labels**.



## Add regression line

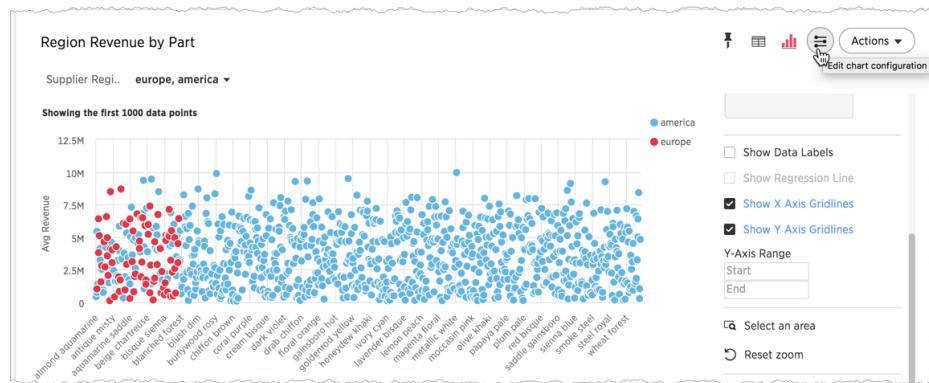
1. While viewing your answer as a chart, click **Edit chart configuration**.
2. Select **Show regression line**.



## Display gridlines

Charts with X and Y axis can display gridlines. To display them:

1. Choose **Edit chart configuration**.
2. Select one or both of the gridline options.



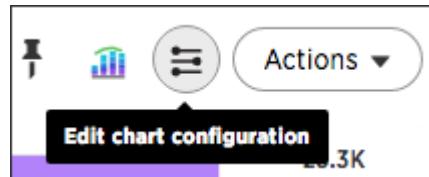
## Lock the chart

You can use the lock icon to prevent ThoughtSpot from recommending other chart options or configurations. For example, you want a line chart, you can lock that display and stop TS from recommending an alternative visualization such as a funnel/pie/column and so forth. A lock applies for all subsequent answers that have the same column set.

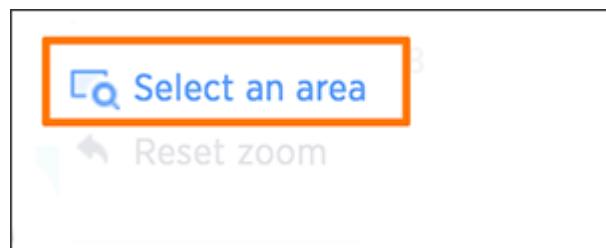
# Zoom into a chart

You can zoom into your chart by selecting an area with your mouse. To zoom into a chart:

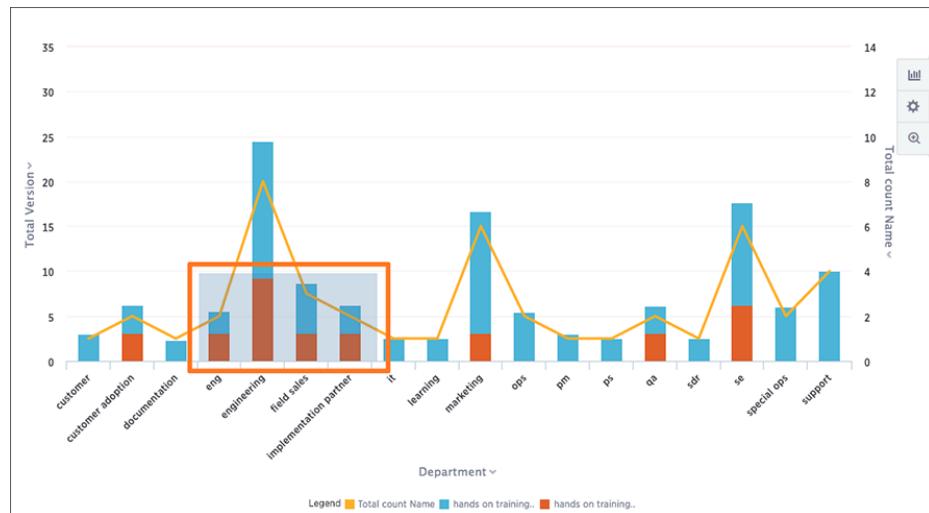
1. While viewing your answer as a chart, click **Edit chart configuration** on the right.



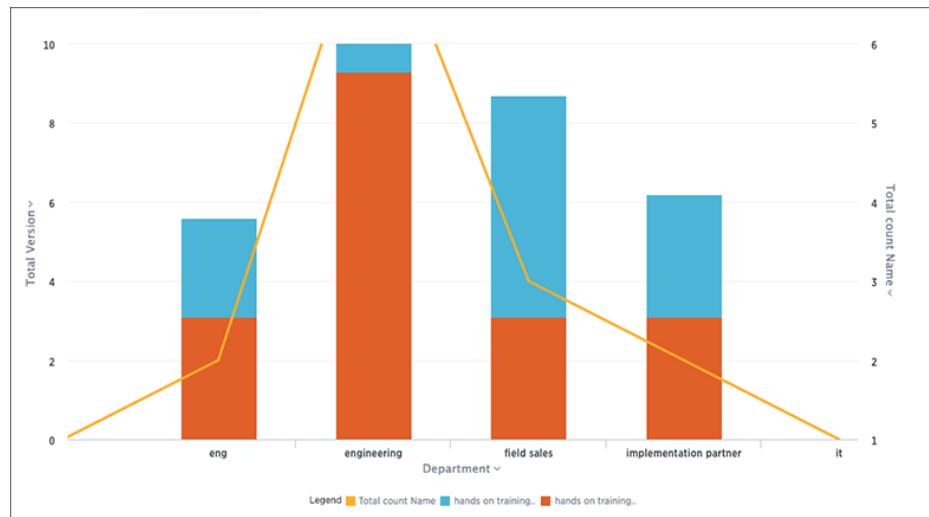
2. Click **Select an area**.



3. Select an area on your chart to zoom into by clicking and dragging your mouse.



Your chart will be reconfigured to only show the selected area.



4. If you would like to return to the original chart view, click **Reset zoom** under **Zoom options**.



# Understand formulas in searches

**Summary:** To provide richer insights, you can add a formula to your search.

The Formula Builder includes many types of operators, such as logical (if, then, else), math, date, and text string functions.

You can create a formula from directly within a search. If you have the privilege that allows you to create or edit worksheets, you can also create a formula within a worksheet. Formulas in worksheets act as derived columns, so that anyone who uses the worksheet as a data source will see the formula as just another column.

Adding a formula within a search works much the same way as adding a formula to a worksheet. However, you will be able to edit the formula directly from within the answer. If you add the answer to a pinboard and share it with the **Edit** privilege, other people can see the formula results, too. To make edits to the formula, they also need to have the **Edit** privilege on the underlying data.

- **Add a formula to a search**

You can add a formula directly within a search. Some common reasons for using a formula in a search are to perform mathematical functions, check for and replace null values, or add if...then...else logic.

- **View or edit a formula in a search**

You can always go back and view or edit a formula that has been added to a search. Do this by clicking the edit icon next to its name in the **Columns** listing.

- **About aggregate formulas**

When working with formulas, it is useful to understand the difference between regular (or row-wise) formulas and aggregation formulas.

- **About conversion formulas**

Some formulas require the input to be of a particular data type. If you find that you want to pass a value to the function, but it is of the wrong data type, you can convert it using a conversion formula.

- **About date formulas**

Date formulas allow you to apply date related functions to your formulas.

- **About percent (simple number) calculations**

You can use simple number functions to perform useful percent calculations.

- **About conditional formulas (operators)**

Conditional formulas, or operators, allow you to apply `if / then / else` conditions in your formulas.

- **About nested formulas**

Nested formulas, or formula on formula, allow you to reference a formula within another formula.

- **About formula support for chasm trap schemas**

You can create a formula that involves aggregated measures coming from multiple fact tables of a chasm trap.

# Add a formula to a search

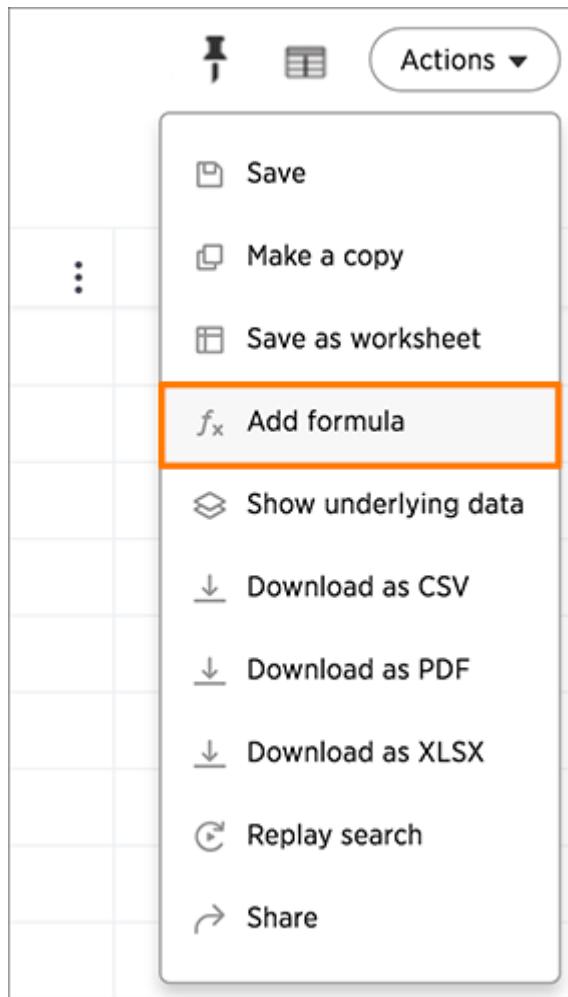
You can add a formula directly within a search. Some common reasons for using a formula in a search are to perform mathematical functions, check for and replace null values, or add conditional logic.

To create a formula in a search:

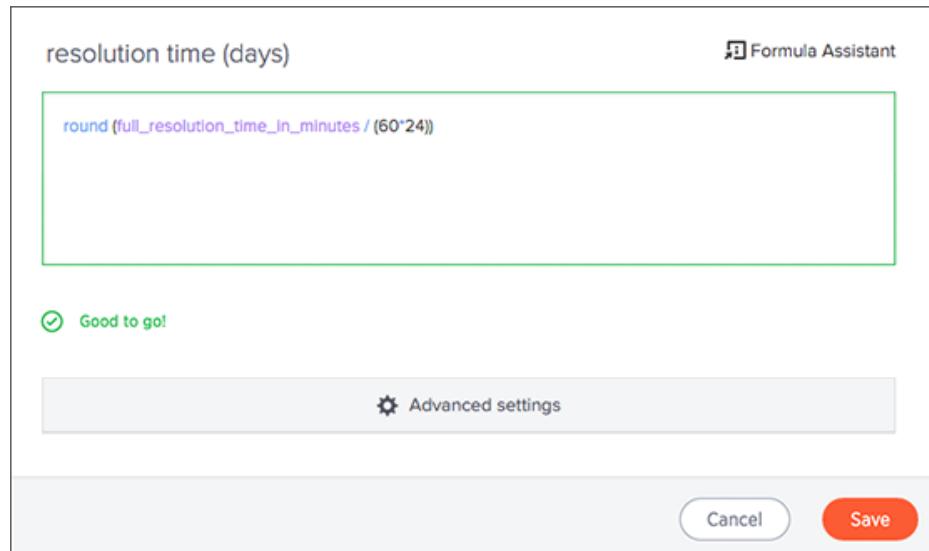
1. Start a new search, or choose an existing answer from a pinboard to edit.
2. If the answer shows a chart, switch to **Data View**.



3. In the upper right hand side of the table, click **Actions** and select **Add formula**.



4. Type your formula in the Formula Builder.



**Note:** Formulas elements are color coded by type and can include the formula operators and functions (blue), the names of columns (purple), and/or constants (black).

5. You can see a list of formula operators with examples by clicking on **Formula Assistant**.

	28
	7
ended into double play	0
	0

6. If you want to change what your formula returns, use the **Advanced settings**. Depending on your formula, you may be able to change:

- Data type
- ATTRIBUTE or MEASURE
- Aggregation type

The screenshot shows the ThoughtSpot Formula Assistant interface. At the top, there is a title bar with the text "resolution time (days)" and a "Formula Assistant" button. Below the title bar is a code editor containing the formula: `round(full_resolution_time_in_minutes / (60*24))`. A green rectangular border highlights this code editor area. Below the code editor, there is a status message "Good to go!" with a green checkmark icon. Underneath this is a section titled "Advanced settings". This section includes dropdown menus for "Data type" (set to "Numeric"), "Measure or attribute" (set to "MEASURE"), and "Aggregation". The "Aggregation" dropdown is open, showing a list of options: TOTAL, AVG, MAX, MIN, STD DEVIATION, TOTAL, TOTAL COUNT, UNIQUE COUNT, and VARIANCE. A red "Save" button is located at the bottom right of this dropdown menu.

7. Name the formula by clicking on its title and typing the new name. Click **Save**.

## Related information

[Formula function reference](#)

# View or edit a formula in a search

You can always go back and view or edit a formula that was added to a search. Do this by clicking the edit icon next to its name in the **Columns** listing. Anyone who has edit privileges on an answer can also edit any formulas it contains. To view or edit an existing formula in an answer:

1. Navigate to the pinboard that contains the answer with the formula, and open it.
2. Scroll down to the bottom of the **Columns** listing. You will see a data source called **Formulas**.
3. Expand **Formulas**, and you'll see a list of all the formulas in this answer.

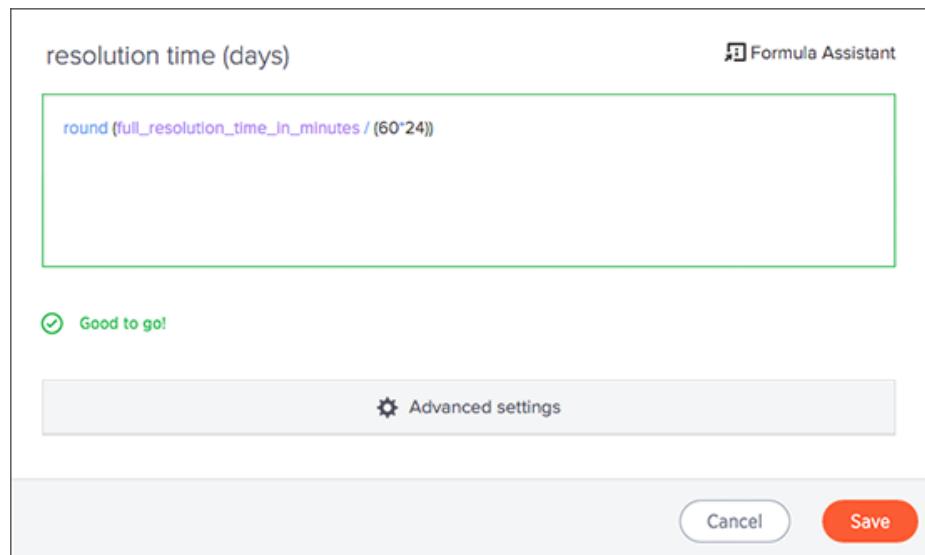
The screenshot shows the ThoughtSpot Data interface. On the left, there's a sidebar with a 'Data' title and a 'Choose Sources' button. Below it is a 'Search Columns' input field containing 'fruit'. A list of columns is shown, with 'fruit\_for\_help' expanded, revealing sub-columns: Date, Fruit, Location, Price per fruit (\$), Quantity sold, Total sale, and Vendor. The 'Formulas' section is also expanded, showing 'top fruit'. This part is highlighted with a red box. To the right, there's a search bar with 'fruit date quantity sold top fruit' and a results table titled 'Total Quantity sold, Top Fruit by Fruit,..'. The table has columns for Fruit, Date, and Quantity sold. The data includes rows for bananas, strawberries, pineapples, and another row for strawberries.

Fruit	Date	Quantity sold
bananas	Feb FY 2016	403
strawberries	May FY 2016	548
strawberries	Jul FY 2015	456
pineapples	Dec FY 2016	410
pineapples	Mar FY 2015	410
strawberries	Mar FY 2012	460

4. Click the edit icon next to the formula name.

This screenshot shows the 'Formulas' section of the Data interface. It lists two formulas: 'Resolution Time' and 'Qa Acceptance Atte...', with an edit icon next to 'Resolution Time'. The 'edit' icon is highlighted with a red box.

5. Type your formula in the Formula Builder.



6. Click **Save** to save the formula with your changes.

# Overview of aggregate formulas

When working with formulas, it is useful to understand the difference between regular (or row-wise) formulas and aggregation formulas.

## Standard and aggregation formulas

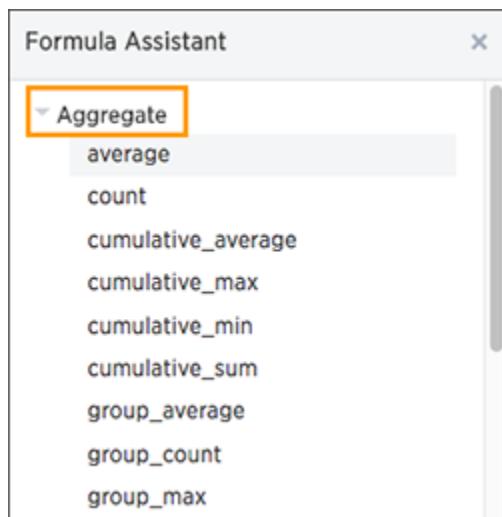
Formulas can be broken down into two types standard and aggregation formulas. Standard formulas act on individual rows and returns one result per row. Standard formulas use functions like:

- `add`
- `subtract`
- `multiply`
- `divide`
- `contains`
- `if...then...else`

Aggregation formulas combine rows together and returns a single result for a group of rows. Aggregation formulas use functions, some examples of functions you find in aggregation formulas are:

- `average`
- `cumulative_sum`
- `moving_average`
- `stddev` (standard deviation)

You can tell which formulas are aggregation formulas by looking at the function list in the **Formula Assistant**. Aggregation functions have their own section.



## Advanced aggregation functions

Some more advanced aggregation functions are widely used in business intelligence, since they provide better insight into data. Some of the more advanced aggregation functions are:

- [Group aggregation functions](#) apply a specific aggregate to a value, and group the results by an attribute in the data.

- [Cumulative functions](#) measure from the start of your data to the current point. They're often applied on time-based data.
- [Moving formulas](#) measure within a window (usually time-based) that you define.

Data from any rows that are not included in the search result will not be incorporated, and you cannot create a filter on aggregated data.

## Using division with aggregation in a search

Whenever your search result combines rows, your formula will get aggregated automatically. For example, if your search contains words like “region”, “monthly”, or “department”, the results will be grouped (aggregated) by that category. The administrator can change the default aggregation that gets applied through a configuration, and you can also change it using the dropdown list in the column header of the search result.

For example, this search would typically return a sum of total sales by department:

```
sum sales department
```

This search would return an average of sales by month:

```
average sales monthly
```

When you’re using division in your formula, and the search is aggregated like this, you may have to change the order of operations to get the result you expect. This is best understood by using a real world example.

Suppose you want to calculate the gross margin by department for a grocery store. The formula for gross margin is:

```
profit / sales
```

But if you use that as your formula, you won’t get the expected calculation. Why? It’s because the formula will be evaluated in this order: For each row, divide profit by sales and then total up all the results. As you can see, the results do not look like gross margin values, which should be between 0 and 1.

Gross margin by department	
Department Description :	Gross Margin without Sum ( ) :
photography	2,012.54
bakery	1,890.42
cleaning supplies	1,493.94
frozen goods	2,453.36
meat	1,152.35
dairy	800.57
gifts	1,736.05

Instead, you'd need to use a formula that uses the order of operations you want:

```
sum (profit) / sum (sales)
```

Now the result is as expected, because the formula totals the profits for all rows, and then divides that by the total of sales for all rows, returning an average gross margin:

Gross margin by department		
Department Description :	Gross Margin without Sum ( ) :	Gross Margin with Sum ( ) :
photography	2,012.54	0.56
bakery	1,890.42	0.54
cleaning supplies	1,493.94	0.40
frozen goods	2,453.36	0.69
meat	1,152.35	0.31
dairy	800.57	0.21
gifts	1,736.05	0.55

# Cumulative functions

**Summary:** Cumulative formulas are aggregate formulas.

Cumulative formulas that allow you to calculate the average, max, min, or sum of your data over time. Although we usually talk about cumulative formulas over time, you can use them over any other sequential data. Each of the cumulative formulas accepts a measure and one or more optional grouping by an attribute (like region or department):

```
formula (measure, [attribute, attribute, ...])
```

Only the measure value is required. If you supply both a measure and attributes, the formula returns the aggregate of the measure accumulated by the attribute(s) in the order specified. You should experiment with only a measure and then with an attribute to see which output best meets your use case.

The cumulative formulas are:

Function	Description
<code>cumulative_average</code>	Takes a measure and one or more attributes. Returns the average of the measure, accumulated by the attribute(s) in the order specified. For example: <code>cumulative_average (revenue, order date, state)</code>
<code>cumulative_max</code>	Takes a measure and one or more attributes. Returns the maximum of the measure, accumulated by the attribute(s) in the order specified. For example: <code>cumulative_max (revenue, state)</code>
<code>cumulative_min</code>	Takes a measure and one or more attributes. Returns the minimum of the measure, accumulated by the attribute(s) in the order specified. For example: <code>cumulative_min (revenue, campaign)</code>
<code>cumulative_sum</code>	Takes a measure and one or more attributes. Returns the sum of the measure, accumulated by the attribute(s) in the order specified. For example: <code>cumulative_sum (revenue, order date)</code>

## Calculate a cumulative sum

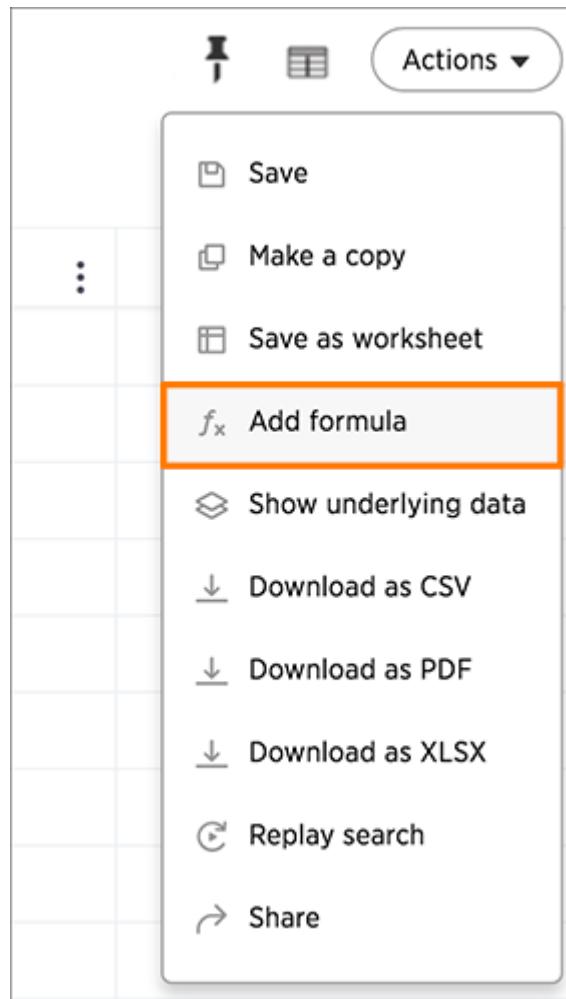
This example demonstrates using the `cumulative_sum` formula, also known as a running total. To use the cumulative function in a search:

1. Start a new search.

The screenshot shows a ThoughtSpot Data View interface. On the left, there's a sidebar with a 'Data' tab, a 'Choose Sources' button, and a 'Search Columns' input field containing 'revenue region commitdate'. The main area displays a table titled 'Total REVENUE by REGION, Quarterly (COMMITDA..)'. The table has three columns: 'REGION', 'COMMITDATE QUARTERLY', and 'REVENUE TOTAL'. The data shows revenue for Africa across four quarters of 1993. At the bottom of the table, it says '( showing rows 1 - 14 of 135 )'. Below the table, there are summary statistics: '18.1B REVENUE', '5 REGION', and 'Q1 FY 1993 - Q3 FY 1999 Quarterly (COMMITDATE)'.

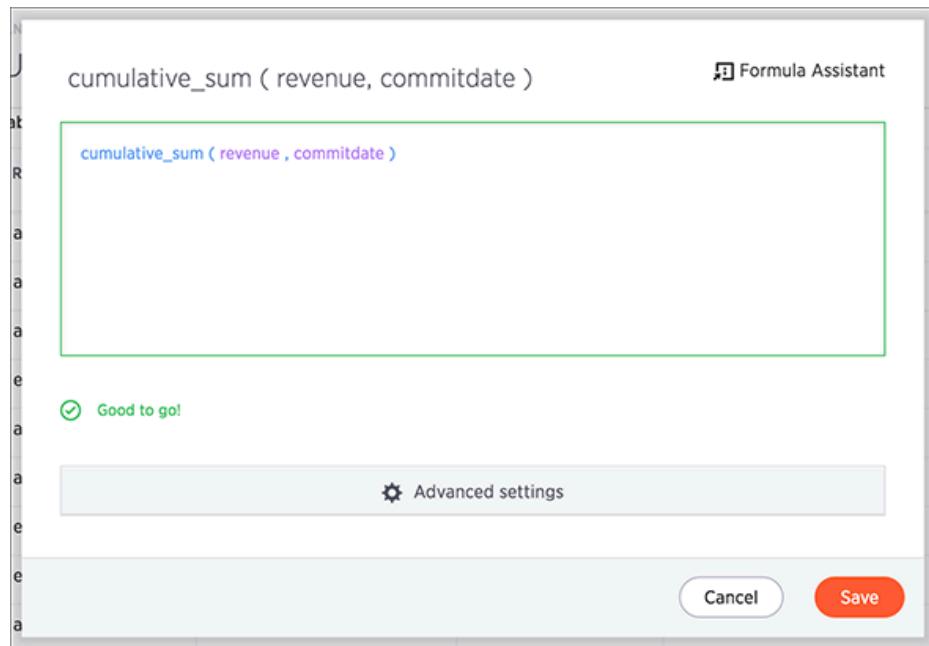
REGION	COMMITDATE QUARTERLY	REVENUE TOTAL
africa	Q1 FY 1993	38,235,729
africa	Q2 FY 1993	67,504,844
africa	Q3 FY 1993	116,890,921
africa	Q4 FY 1993	180,153,392
africa	Q1 FY 1994	166,139,486
africa	Q2 FY 1994	189,965,026
africa	Q3 FY 1994	138,520,689
africa	Q4 FY 1994	147,903,759

2. In the upper right hand side of the table, click **Actions** and select **Add formula**.



3. Enter the cumulative\_sum formula, providing a measure and one or more attributes.

The example will return the sum of revenue, accumulated by the commit date.



4. Name the formula by clicking on its title and typing the new name.
5. Click **Save**.

The formula will appear in the search bar and in the table as its own column.

REGION	COMMITDATE	REVENUE	cumulative_sum ( revenue, commitdate )
africa	Nov FY 1993	86,601,468	309,232,962
africa	Nov FY 1994	28,881,900	926,291,987
africa	Jan FY 1998	16,773,589	3,079,460,707
america	Apr FY 1993	41,296,350	85,157,963
america	Sep FY 1995	12,512,017	1,481,630,414
america	Oct FY 1996	69,654,810	1,939,222,892
america	Nov FY 1996	45,631,320	1,984,854,212

( showing rows 1 - 9 of 401 )

18.1B REVENUE TOTAL    5 REGION UNIQUE COUNT    Feb FY 1993 - Oct FY 1999 Monthly (COMMITDATE)    7.1M cumulative\_sum ( revenue, commitdate ) MIN

A headline box displaying the cumulative sum within the entire table will appear at the bottom. You can click on it to toggle between different aggregations.

# Group aggregation functions

What if you want to aggregate a value by a specific attribute (for example, show revenue by product)? This is known as a grouped aggregation, but some people call it a pinned measure or level-based measure. You can do this for any aggregation using the grouping functions.

Each of the grouping functions accepts a measure and one or more optional attributes:

```
formula (measure, [attribute, attribute, ...])
```

Only the measure value is required. If you supply both a measure and an attribute, the function returns the aggregate of the measure grouped by the attribute(s). You should experiment with only a measure and then with an attribute to see which output best meets your use case.

## List of group functions

Group aggregation functions have names with formats like `group_<aggregation>`. The group aggregation functions are the following:

Function	Description
<code>group_average</code>	Takes a measure and one or more attributes. Returns the average of the measure grouped by the attribute(s).  <code>group_count (revenue, customer region)</code>
<code>group_count</code>	Takes a measure and one or more attributes. Returns the count of the measure grouped by the attribute(s).  <code>group_count (revenue, customer region)</code>
<code>group_max</code>	Takes a measure and one or more attributes. Returns the maximum of the measure grouped by the attribute(s).  <code>group_max (revenue, customer region)</code>
<code>group_min</code>	Takes a measure and one or more attributes. Returns the minimum of the measure grouped by the attribute(s).  <code>group_min (revenue, customer region)</code>

Function	Description
<code>group_stddev</code>	Takes a measure and one or more attributes. Returns the standard deviation of the measure grouped by the attribute(s).  <code>group_stddev (revenue, customer region)</code>
<code>group_sum</code>	Takes a measure and one or more attributes. Returns the sum of the measure grouped by the attribute(s).  <code>group_sum (revenue, customer region)</code>
<code>group_unique_count</code>	Takes a column name and one or more attributes. Returns the number of unique values in a column, grouped by the attribute(s).  <code>group_unique_count (product, supplier)</code>
<code>group_variance</code>	Takes a measure and one or more attributes. Returns the variance of the measure grouped by the attribute(s).  <code>group_variance (revenue, customer region)</code>

# Moving functions

**Summary:** Moving formulas are aggregate formulas that allow you to calculate the average, max, min, or sum of your data over a predetermined interval, or window, with an adjustable range.

Moving formulas can be used to smooth out any irregularities in your data to easily recognize trends. The larger the interval you set, the more the peaks and valleys are smoothed out. While the smaller the interval, the closer the moving averages are to the actual data points.

Each of the moving formula accepts a measure, two integers to define the window, and one or more optional attributes.

```
formula (measure, integer, integer, [attribute, attribute, ...])
```

Only the measure and integer values are required. If you supply both required and optional values, the formula returns the aggregate of the measure over the given window. You should experiment with only a measure and integers leaving out the attribute and then adding it back in. This will help you decide which output best meets your use case.

The moving formulas are the following:

- `moving_average`, for example `moving_average (revenue, 2, 1, customer region)`

Takes a measure, two integers to define the window to aggregate over, and one or more attributes. Returns the average of the measure over the given window. The attributes are the ordering columns used to compute the moving average. The window is (`current - Num1...Current + Num2`) with both end points being included in the window. For example, `1,1` will have a window size of 3. To see periods in the past, use a negative number for the second endpoint, as in the example `moving_average(sales, 1, -1, date)`.

- `moving_max`, for example `moving_max (complaints, 1, 2, store name)`

Takes a measure, two integers to define the window to aggregate over, and one or more attributes. Returns the maximum of the measure over the given window. The attributes are the ordering columns used to compute the moving maximum. The window is (`current - Num1...Current + Num2`) with both end points being included in the window. For example, `1,1` will have a window size of 3. To see periods in the past, use a negative number for the second endpoint, as in the example `moving_max(sales, 1, -1, date)`.

- `moving_min`, for example `moving_min (defects, 3, 1, product)`

Takes a measure, two integers to define the window to aggregate over, and one or more attributes. Returns the minimum of the measure over the given window. The attributes are the ordering columns used to compute the moving minimum. The window is (`current - Num1...Current + Num2`) with both end points being included in the window. For example, `1,1` will have a window size of 3. To see periods in the past, use a negative number for the second endpoint, as in the example `moving_min(sales, 1, -1, date)`.

- `moving_sum`, for example `moving_sum (revenue, 1, 1, order date)`

Takes a measure, two integers to define the window to aggregate over, and one or more attributes. Returns the sum of the measure over the given window. The attributes are the ordering columns used to compute the moving sum. The window is (current - Num1...Current + Num2) with both end points being included in the window. For example, `1,1` will have a window size of 3. To see periods in the past, use a negative number for the second endpoint, as in the example `moving_sum(sales, 1, -1, date)`.

## Calculate a moving average

This example demonstrates using the `moving_average` formula. To use the moving function in a search:

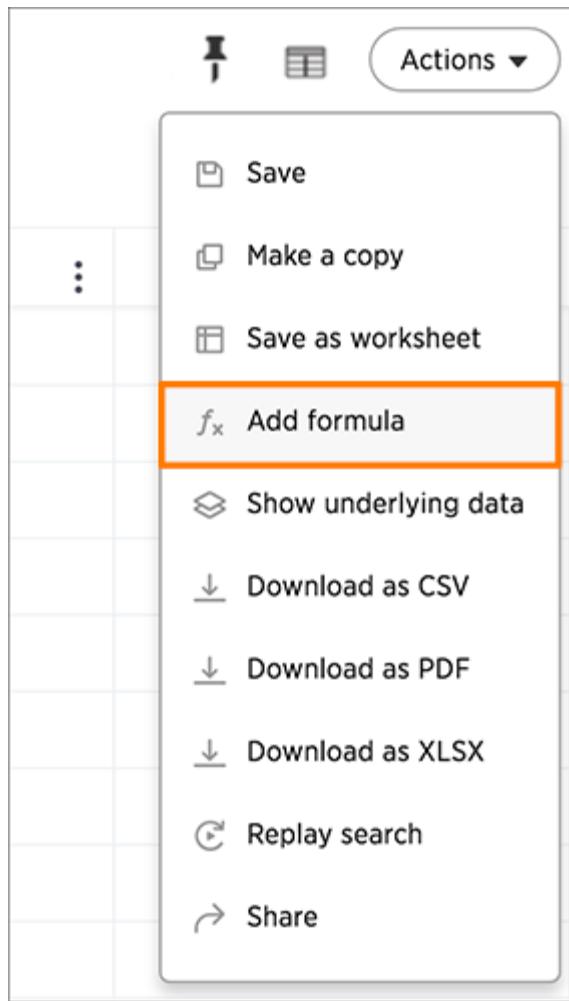
1. Start a new search.

The screenshot shows a search interface with a sidebar on the left containing 'Data' and 'Choose Sources' buttons, and a list of source types: CUSTOMER, DATE, LINEORDER, PART, and Formulas. The main area displays a table titled 'Total REVENUE by REGION, Quarterly (COMMITDA..)' with the following data:

REGION	COMMITDATE QUARTERLY	REVENUE TOTAL
africa	Q1 FY 1993	38,235,729
africa	Q2 FY 1993	67,504,844
africa	Q3 FY 1993	116,890,921
africa	Q4 FY 1993	180,153,392
africa	Q1 FY 1994	166,139,486
africa	Q2 FY 1994	189,965,026
africa	Q3 FY 1994	138,520,689
africa	Q4 FY 1994	147,903,759

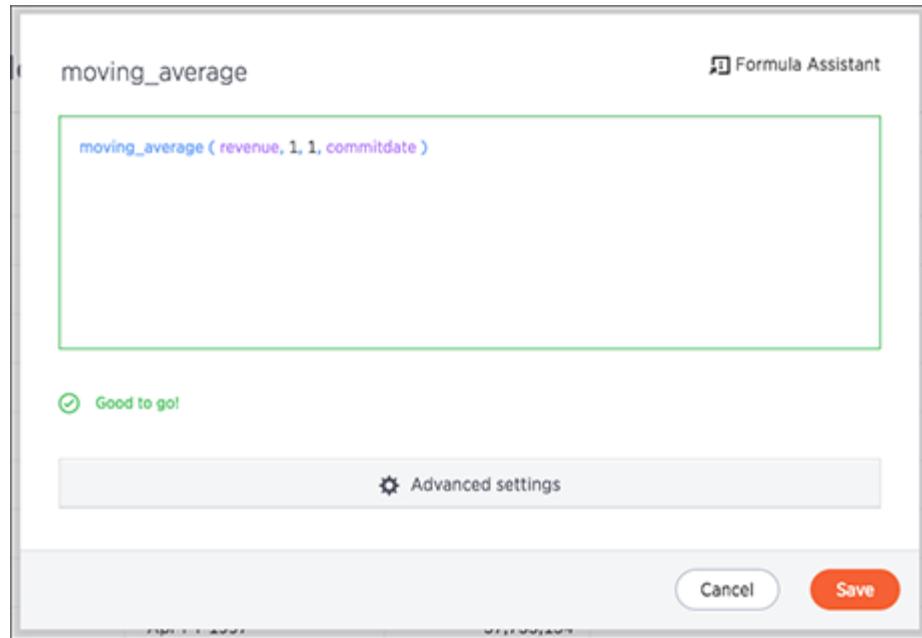
At the bottom of the table, there are summary statistics: '18.1B REVENUE TOTAL', '5 REGION UNIQUE COUNT', and 'Q1 FY 1993 - Q3 FY 1999 Quarterly (COMMITDATE)'.

2. In the upper right hand side of the table, click **Actions** and select **Add formula**.



3. Enter the moving\_average formula, providing a measure, a window, and one or more attributes.

The example will return the average of revenue, within the commit date window size of 3. The window includes the previous, current, and next rows. The attributes are the ordering columns used to compute the moving average. The window is (current - Num1...Current + Num2) with both end points being included in the window. For example, “1,1” will have a window size of 3. To see periods in the past, use a negative number for the second endpoint, as in the example “moving\_average(sales, 1, -1, date)”.



4. Name the formula by clicking on its title and typing the new name. Click **Save**.

The formula will appear in the search bar and in the table as its own column.

REGION	COMMITDATE	REVENUE	moving_average (revenue, 1, 1, commitdate)
africa	Nov FY 1993	86,601,468	67,580,044.33
africa	Nov FY 1994	28,881,900	44,018,897.67
africa	Jan FY 1998	16,773,589	20,810,803.00
america	Apr FY 1993	41,296,350	50,056,679.00
america	Sep FY 1995	12,512,017	27,881,766.00
america	Oct FY 1996	69,654,810	54,408,279.67
america	Nov FY 1996	45,631,320	52,882,097.00
america	Apr FY 1997	21,160,168	25,207,722.67

( showing rows 1 - 7 of 401 )

18.1B REVENUE TOTAL    5 REGION UNIQUE COUNT    Feb FY 1993 - Oct FY 1999 Monthly (COMMITDATE)    10.2M moving\_average (revenue, 1, 1, commitdate) MIN

A box displaying the moving average within the entire table will appear at the bottom.

5. Click on the box to toggle between aggregation types.

# Filtered aggregation functions

**Summary:** You can create a filtered aggregation in the search bar.

Filters are useful for queries where the results should reflect a new, filtered value. On this page you learn about comparative versus derivative filters, and the functions, and the keywords that you can use with them.

## Comparative and derivative filters

Comparative filters compare two segments of some whole against each in the **Search** bar. For example, a company that has locations across the United States, may want to compare total revenue in the West to the East segment. In a comparative filter, one of the segments you are comparing is filtered.

An example of a comparative filter is comparing west revenue with total revenue. In these cases, one measure is a *filtered measure*, for example, `revenue region = west` is a filtered measure.

Derivative filters add a column to your results which is derived from other columns in the same results. For example, you search for revenue and cost and want to calculate profit in your result.

Some examples of comparative and derivative filters in the real world are:

- revenue of this\_soap versus all\_soaps (Comparative filter)
- tax as a percentage of revenue (derivative)
- count revenue as a percentage of state revenue (comparative with a derivative)

If you plan to create these types of filters, you need to understand how to create filter functions.

## Use filter functions

Filter functions take two arguments, the column ( measure or attribute) to aggregate and the filter condition:

```
FUNCTION_NAME(condition, <column name>)
```

ThoughtSpot functional library will include the following functions:

- `sum_if`
- `average_if`
- `count_if`
- `unique_count_if`
- `max_if`
- `min_if`
- `stddev_if`
- `variance_if`
- `vs`
- `all`

The table below illustrates some examples of these functions in use:

Function	Examples
<code>sum_if(region='west', revenue)</code>	Only aggregate the revenue for the values corresponding to west region.
<code>count_if(region = 'west', region)</code>	Only aggregate the region for the values corresponding to west region.
<code>count_if(revenue &gt; 100, red)</code>	Count the number of times red appears when revenue was greater than 100 (row level revenue data, not aggregated).

A condition can have multiple filters like `sales region = west OR region = east`. You can also just type a value such as `east` as in `sales east` as a filter. If there are no rows matching the criteria, the condition returns a 0 (zero). A 0 can result in situations where there are logic errors in the formula, so be sure to double-check your work.

The screenshot shows the ThoughtSpot interface with a search bar containing 'revenue by\_color\_filter'. The results pane displays a formula named 'by\_color\_filter' with the expression 'sum\_if (color="red", revenue)'. A green box highlights this formula. The sidebar on the left shows various data sources and formulas, with 'Revenue' selected. A message at the bottom says 'Good to go!'

Once you have aggregated with a filter, you can do further comparisons with the `vs` and `all` keywords.

## Using the vs and all keywords

You can use the `vs` and `all` keywords to expand the usefulness of your comparison filters. It compares a measure across different sets of filters and or groupings. The basic format of a comparison search is:

```
<common search tokens> (A vs B) <common search tokens>
```

For example:

```
revenue region last 10 years vs all
```

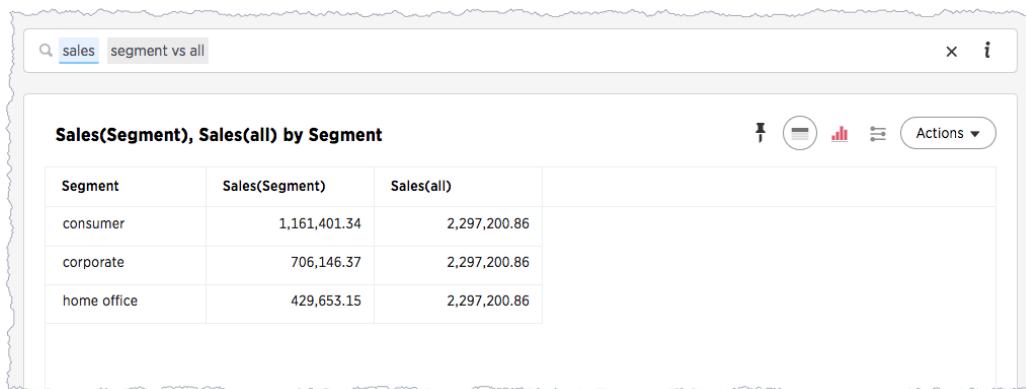
Try this syntax on using the Superstore example data. The first `vs` example compares two segments with a single search token:



The screenshot shows a search bar with the query "consumer vs corporate" and a selected token "sales". Below the search bar is a table titled "Sales(consumer), Sales(corporate)". The table has two columns: "Sales(consumer)" and "Sales(corporate)". The values are 1,161,401.34 and 706,146.37 respectively.

Sales(consumer)	Sales(corporate)
1,161,401.34	706,146.37

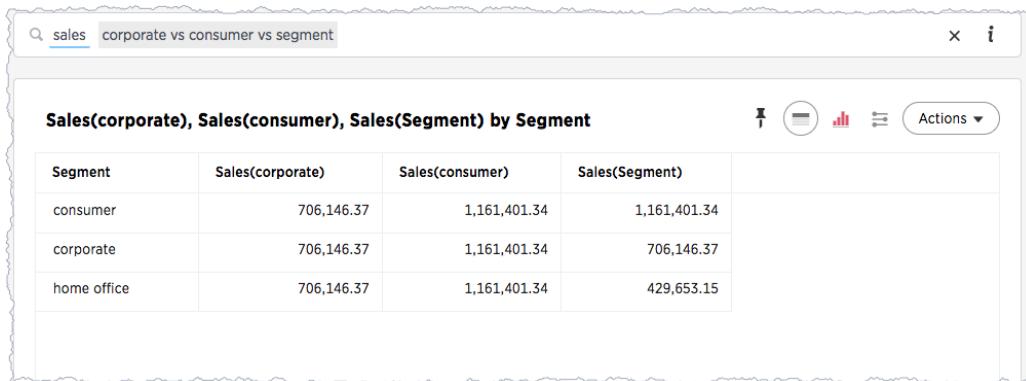
The system automatically applies the `sales` token to both sides and groups each segment. You can use the `all` keyword to break out the segments and avoid grouping.



The screenshot shows a search bar with the query "sales" and a selected token "segment vs all". Below the search bar is a table titled "Sales(Segment), Sales(all) by Segment". The table has three columns: "Segment", "Sales(Segment)", and "Sales(all)". The rows show data for consumer, corporate, and home office segments.

Segment	Sales(Segment)	Sales(all)
consumer	1,161,401.34	2,297,200.86
corporate	706,146.37	2,297,200.86
home office	429,653.15	2,297,200.86

You can also provide multiple `vs` instances:



The screenshot shows a search bar with the query "sales" and a selected token "corporate vs consumer vs segment". Below the search bar is a table titled "Sales(corporate), Sales(consumer), Sales(Segment) by Segment". The table has four columns: "Segment", "Sales(corporate)", "Sales(consumer)", and "Sales(Segment)". The rows show data for consumer, corporate, and home office segments.

Segment	Sales(corporate)	Sales(consumer)	Sales(Segment)
consumer	706,146.37	1,161,401.34	1,161,401.34
corporate	706,146.37	1,161,401.34	706,146.37
home office	706,146.37	1,161,401.34	429,653.15

Of course, you can compare across different columns as well:

The screenshot shows a search interface with a search bar at the top containing 'sales city vs state'. Below the search bar is a table titled 'Sales(City), Sales(State) by City, State'. The table has four columns: 'City', 'State', 'Sales(City)', and 'Sales(State)'. The data in the table is as follows:

City	State	Sales(City)	Sales(State)
allentown	pennsylvania	853.25	116,511.91
bartlett	tennessee	88.96	30,661.87
chandler	arizona	1,076.75	35,282.00
cleveland	ohio	6,346.18	78,258.14
conway	arkansas	301.96	11,678.13
cottage grove	minnesota	144.36	29,863.15

Other supported formats you can try:

- `sales accessory6 accessory12 vs all`
- `sales monthly accessory6 vs last year`
- `sales staples file caddy vs all monthly` answers what the share of sales belonging to the file caddy by month
- `sales (germany ariel vs laundry) july 2017 time` answers what is the category share of Germany Ariel for July 17?

# Type conversion functions

Some formulas require the input to be of a particular data type. If you find that you want to pass a value to the function, but it is of the wrong data type, you can convert it using a conversion formula.

The following are the conversion formulas:

Function	Description	Examples
<code>to_bool</code>	Returns the input as a boolean (true or false).	<code>to_bool (0) = false</code> <code>to_bool (married)</code>
<code>to_date</code>	Accepts a date represented as an integer or text string, and a second string parameter that can include strftime date formatting elements. Replaces all the valid strftime date formatting elements with their string counterparts and returns the result. Does not accept epoch formatted dates as input.	<code>to_date (date_sold, '%Y-%m-%d')</code>
<code>to_double</code>	Returns the input as a double.	<code>to_double ('3.14') = 3.14</code> <code>to_double (revenue * .01)</code>
<code>to_integer</code>	Returns the input as an integer.	<code>to_integer ('45') + 1 = 46</code> <code>to_integer (price + tax - cost)</code>
<code>to_string</code>	Returns the input as a text string.	<code>to_string (45 + 1) = '46'</code> <code>to_string (revenue - cost)</code>

# About date formulas

Date formulas are useful when you want to compare data collected between two date periods. Date formulas allow you to apply date related functions to your formulas. The date functions include:

Function	Description	Examples
<code>add_days</code>	Returns the result of adding the specified number of days to the given date.	<code>add_days (01/30/2015, 5) = 02/04/2015</code> <code>add_days (invoiced, 30)</code>
<code>date</code>	Returns the date portion of a given date.	<code>date (home visit)</code>
<code>day</code>	Returns the number (1-31) of the day for the given date.	<code>day (01/15/2014) = 15</code> <code>day (date ordered)</code>
<code>day_number_of_week</code>	Returns the number (1-7) of the day in a week for the given date with 1 being Monday and 7 being Sunday.	<code>day_number_of_week (01/30/2015) = 6</code> <code>day_number_of_week (shipped)</code>
<code>day_number_of_quarter</code>	Returns the number of the day in a quarter for a given date.	<code>day_number_of_quarter (01/30/2015)</code>
<code>day_number_of_week</code>	Returns the number of the day in a week for a given date.	<code>day_number_of_week(01/15/2014) &gt; 3</code>
<code>day_number_of_year</code>	Returns the number (1-366) of the day in a year for the given date.	<code>day_number_of_year (01/30/2015) = 30</code> <code>day_number_of_year (invoiced)</code>
<code>day_of_week</code>	Returns the day of the week for the given date.	<code>day_of_week (01/30/2015) = Friday</code> <code>day_of_week (serviced)</code>
<code>diff_days</code>	Subtracts the second date from the first date and returns the result in number of days, rounded down if not exact.	<code>diff_days (01/15/2014, 01/17/2014) = -2</code> <code>diff_days (purchased, shipped)</code>

Function	Description	Examples
<code>diff_time</code>	Subtracts the second date from the first date and returns the result in number of seconds.	<code>diff_time (01/01/2014, 01/01/2014) = -86,400</code> <code>diff_time (clicked, submitted)</code>
<code>hour_of_day</code>	Returns the hour of the day for the given date.	<code>hour_of_day (received)</code>
<code>is_weekend</code>	Returns true if the given date falls on a Saturday or Sunday.	<code>is_weekend (01/31/2015) = true</code> <code>is_weekend (emailed)</code>
<code>month</code>	Returns the month from the given date.	<code>month (01/15/2014) = January</code> <code>month (date ordered)</code>
<code>month_number</code>	Returns the number (1-12) of the month for the given date.	<code>month_number (09/20/2014) = 9</code> <code>month_number (purchased)</code>
<code>month_number_of_quarter</code>	Returns the month (1-12) number for the given date in a quarter.	<code>month_number_of_quarter (02/20/2018) &gt; 9</code>
<code>now</code>	Returns the current timestamp.	<code>now ()</code>
<code>start_of_month</code>	Returns <code>MMM yyyy</code> for the first day of the month. Your installation configuration can override this setting so that it returns a different format such as <code>MM/dd/yyyy</code> . Speak with your ThoughtSpot administrator for information on doing this.	<code>start_of_month (01/31/2015) = Jan FY 2015</code> <code>start_of_month (shipped)</code>
<code>start_of_quarter</code>	Returns the date for the first day of the quarter for the given date.	<code>start_of_quarter (09/18/2015) = Q3 FY 2015</code> <code>start_of_quarter (sold)</code>
<code>start_of_week</code>	Returns the date for the first day of the week for the given date.	<code>start_of_week (06/01/2015) = 05/30/2015 Week</code> <code>start_of_week (emailed)</code>

Function	Description	Examples
<code>start_of_year</code>	Returns the date for the first day of the year for the given date.	<code>start_of_year ( 02/15/2015 ) = FY 2015</code> <code>start_of_year (joined)</code>
<code>time</code>	Returns the time portion of a given date.	<code>time (3/1/2002 10:32) = 10:32</code> <code>time (call began)</code>
<code>week_number_of_month</code>	Returns the week number for the given date in a month.	<code>week_number_of_month(03/23/2017) = 3</code>
<code>week_number_of_quarter</code>	Returns the week number for the given date in a quarter.	<code>week_number_of_quarter(04/03/2017)&gt; 2</code>
<code>week_number_of_year</code>	Returns the week number for the given date in a year.	<code>week_number_of_year(04/03/2017) = 20</code>
<code>year</code>	Returns the year from the given date.	<code>year (01/15/2014) = 2014</code> <code>year (date ordered)</code>

## Calculate date formulas

Calculating date formulas is useful when you want to compare data from different date periods. Here are some examples of using date formulas:

### Example 1

The following example shows you how to create formulas that you can use to compare data from this week to last week.

- The formula for this week is: `week ( today () ) - week (date)`
- The formula for last week is: `diff_days ( week ( today ) ) , week ( date ) )`

### Example 2

The following example shows you how to calculate the percent increase from the last date period to this period in terms of revenue.

1. Create the formula: `this week revenue = sum ( if ( this week ) then revenue else 0 )`
2. Then create the formula: `last week revenue = sum ( if (last week ) then revenue else 0 )`
3. Use **nested formulas** to calculate the percent increase by creating a parent formula: `percent increase = ( ( last week revenue - this week revenue) / last week revenue ) \* 100`

# Percent (simple number) calculations

You can use simple number functions to perform useful percent calculations. Simple number functions include addition, subtraction, multiplication, and division.

Function	Description	Examples
*	Returns the result of multiplying both numbers.	<ul style="list-style-type: none"><li><code>3 * 2 = 6</code></li><li><code>price * taxrate</code></li></ul>
+	Returns the result of adding both numbers.	<ul style="list-style-type: none"><li><code>1 + 2 = 3</code></li><li><code>price + shipping</code></li></ul>
-	Returns the result of subtracting the second number from the first.	<ul style="list-style-type: none"><li><code>3 - 2 = 1</code></li><li><code>revenue - tax</code></li></ul>
/	Returns the result of dividing the first number by the second.	<ul style="list-style-type: none"><li><code>6 / 3 = 2</code></li><li><code>markup / retail price</code></li></ul>

## Calculate percentages

Calculating percentages is useful when you want to see, for example, the percentage revenue generated from each channel (online, stores, partner, etc.). For example, you can figure out the percentage revenue generated from each channel (online, stores, partners, etc.).

Create a formula called `Percent Revenue` in the **Formula Assistant**:

```
( sum ( revenue ) / group sum ( revenue ) ) \* 100
```

Now you can search by each channel, using the percent revenue formula that you just created.

# Formula operators

Formula operators allow you to apply `if / then / else` conditions in your formulas. You can leverage operators in your formulas to have them return true, false, or a predetermined value. The operators include:

Operator	Description	Examples
<code>and</code>	Returns true when both conditions are true, otherwise returns false.	<code>(1 = 1) and (3 &gt; 2) = true</code> <code>lastname = 'smith' and state = 'texas'</code>
		<b>Note:</b> Not available for row level security (RLS) formulas.
<code>if...then...else</code>	Conditional operator.	<code>if (3 &gt; 2) then 'bigger' else 'not bigger'</code> <code>if (cost &gt; 500) then 'flag' else 'approve'</code>
<code>ifnull</code>	Returns the first value if it is not null, otherwise returns the second.	<code>ifnull (cost, 'unknown')</code>
<code>isnull</code>	Returns true if the value is null.	<code>isnull (phone)</code>
<code>not</code>	Returns true if the condition is false, otherwise returns false.	<code>not (3 &gt; 2) = false</code> <code>not (state = 'texas')</code>
<code>or</code>	Returns true when either condition is true, otherwise returns false.	<code>(1 = 5) or (3 &gt; 2) = true</code> <code>state = 'california' or state = 'oregon'</code>

## Calculate the conditional sum

Calculating the conditional sum is useful when you want to see, for example, the total revenue for a product by region.

Conditional sum formulas follow this syntax: `if (some condition) then (measure) else 0`. You can use this syntax to limit your search in cases when you don't want to add a column filter. For example: `if (product = shoes) then revenue else 0`

The following example shows you how to figure out the number of customers who bought both products, in this case an ipad and galaxy tablet. You can then find out the revenue generated by both products.

1. Create the following formula in the Formula Builder:

```
ipadcount = sum ( if ( product = 'ipad' ) then 1 else 0 ) > 0
```

This formula will provide you with the number of ipads that were bought.

2. You can then create another formula that looks like this:

```
galaxycount = sum ( if ( product = 'galaxy' then 1 else 0 ) > 0 )
```

And this formula will provide you with the number of galaxys that were bought.

3. Using [nested formulas](#), you can combine these two formulas.

For example: `f1 = ipadcount + galaxycount`

4. Now, you can search using the `f1` formula to find out the revenue generated by both products.

# Nested formulas

Nested formulas, or formula on formula, allow you to reference a formula within another formula. This graphic illustrates how you would define a formula and then reference it from another:



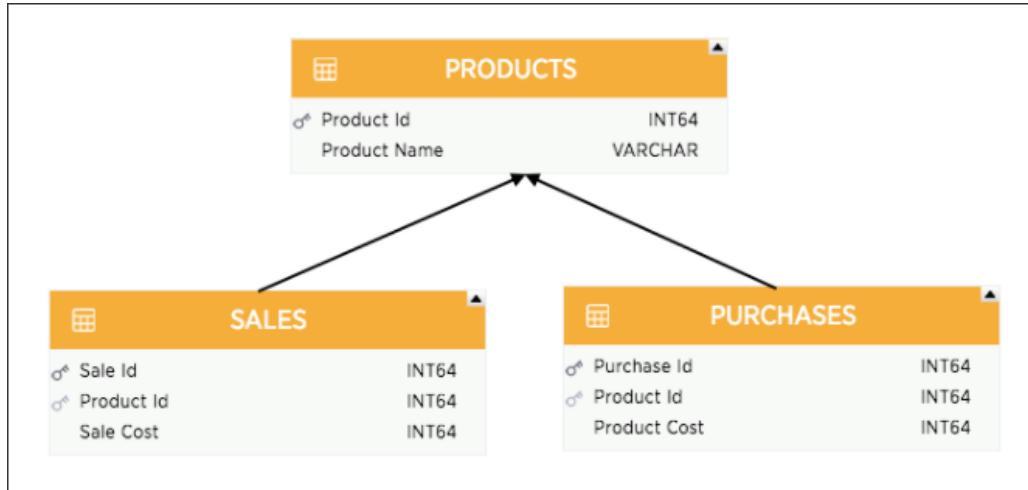
These formulas become columns, and are even suggested within the Formula Builder. There is no limit to the nesting depth you can create.

ThoughtSpot doesn't allow users to create invalid formulas, such as cyclic (loops) or inconsistent aggregation formulas. To support this, formulas have dependency awareness. For example, you cannot delete a formula that is nested in another. Also, formula dependency awareness ensures that you do not break any dependent formulas while editing a formula with dependent formulas. For example, common cases of invalidating occur when users try to change a data type.

Nesting enables encapsulation and decomposition of formula logic. Meaning, you may have a staff member create some small useful formulas that others can use. This also minimizes duplication of formulas.

# Formula support for chasm trap schemas

You can create a formula that involves aggregated measures coming from multiple fact tables of a chasm trap.



Just as you would create any other formula, you can create a formula that spans across a chasm trap. The Formula Builder will provide column suggestions for formulas that span across chasm traps.

# Refine a search with a worksheet

**Summary:** If you want to search on top of another search, try saving your search as a worksheet. Then, you can use the saved worksheet as a data source for a new search.

Worksheets can be created from tables or from searching. You can also create a worksheet from a search. This is an advanced form of searching. A worksheet created from a search is called an *aggregated worksheet*. Aggregated worksheets are effectively the same as any worksheet.

When you do a search on a data source, ThoughtSpot is only able to aggregate one column by one other column. Because of this, you may come across searches you can't do in one pass, because they are essentially nested queries. But you can create the equivalent of nested queries using an aggregated worksheet, which is essentially an answer that you save as a worksheet.

You can use search saved as a worksheet just like any other data source. You can even link it to other sources by defining a relationship. When you save an answer as a worksheet, and then use it as a source in a new search, it is similar to doing a nested query in SQL, only much easier to create.

To create a worksheet from a search, you must belong to a group that has the privilege **Has administration privileges** or **Can Manage Data**. If you are not able to create aggregated worksheets, contact your administrator and request the **Can Manage Data** privilege.

## Aggregated worksheet workflow

Suppose you have created a search on the sales fact table that shows the top ten Sales Reps by revenue for the first quarter. Then you want to do some further investigations on that set of data, like ranking them by how much they discounted a specific product using data from the orders fact table. Unless you save your first answer as a worksheet, certain explorations like this won't be possible. If you want to do this, here are the steps at a high level:

1. Create the first search, and [save it as an aggregated worksheet](#).
2. Link your worksheet to any other data sources you'll need.
3. Create a new search that includes your aggregated worksheet and the other sources you linked with it.
4. You may want to create a new worksheet that includes these data sources.

This will make it easy for people to search using the same group of aggregated worksheet and tables that you created.

## Best practices for using aggregated worksheets

Only users with administrative privileges are able to create aggregated worksheets and link them. Users that create aggregated worksheets should keep in mind best practices for creating a worksheet and the boundaries around the final worksheet size.

You can't link an aggregated worksheet with a sharded table. If you do this and try to search on it, you will get an error.

To be able to join an aggregated worksheet with a base table, your installation must be configured to allow the behavior. The aggregated worksheet cannot have more than 5 tables involved. Moreover, the number of rows in the final aggregated worksheet cannot be greater than 1000.

The order of the objects being linked (joined) matters, this is because joins are directional. The table/aggregated worksheet with the foreign key needs to occur in the first (left) position. The table with the primary key should be in the second (right) position.

For the best performance, the final aggregated worksheet should have 50 or fewer columns and no more than 10 million rows. Exceeding these boundaries can make your worksheet creation slow or error prone.

You can use an ETL (extract, transform, load) process to circumvent these limitations.

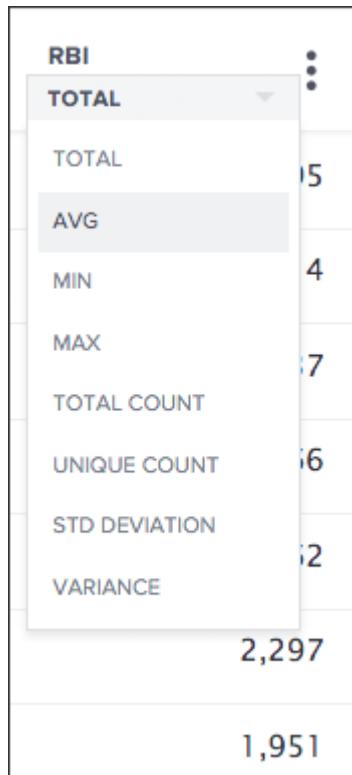
# Save a search as a worksheet

This procedure walks you through creating a worksheet from a search. To create a worksheet from a search (i.e. an aggregated worksheet):

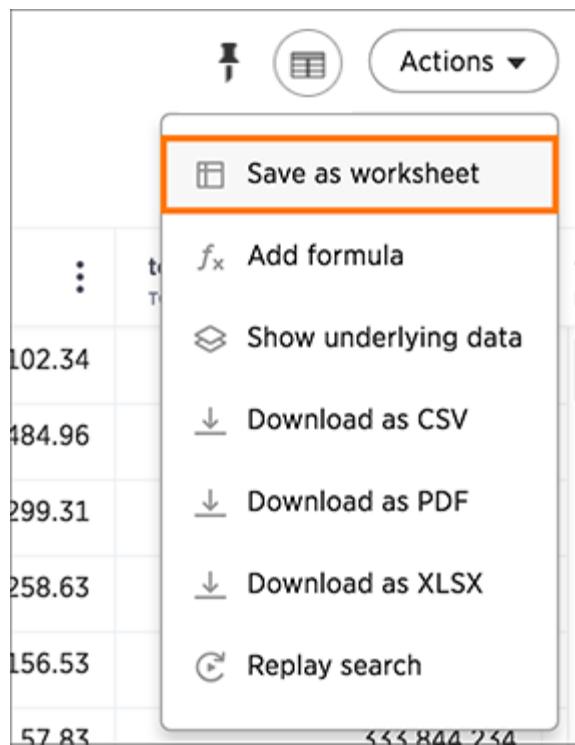
1. Start a new search, or edit an existing visualization from a pinboard.

Any filters or aggregations created during this search will be reflected in the worksheet.

2. If you want to use a different aggregation than the default one for any column, set it from the column header.



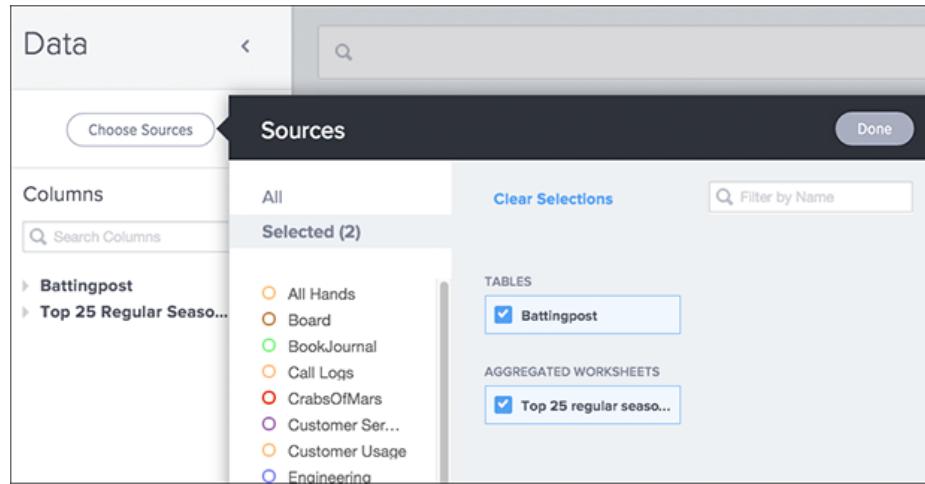
3. Save the answer as a **Worksheet**.



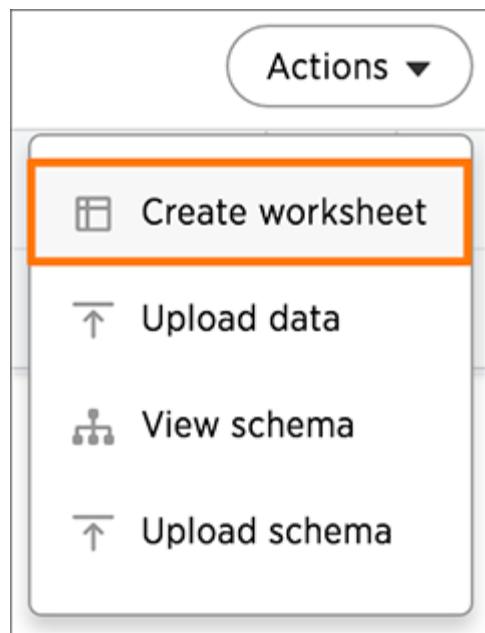
# Search from an aggregated worksheet

After creating a worksheet from a search (also called an aggregated worksheet) and linking it to related data, you're ready to create your new search. To do a search on the aggregated worksheet, along with any data sources you linked:

1. Click the search icon, and select **Data Source**. Choose your linked sources.



2. Do a search using columns from the linked sources, including any aggregations you created.
3. Test the result, to make sure it's what you expect. If your search shows no data found or doesn't look right to you, it is possible that one of the links between your sources was made on the incorrect column. Check the relationships you created and try linking using a different column, to see if that gives the expected search results.
4. Once you have the expected answer, you can create a worksheet to make it easier for you and other people to use. To do this, click the **DATA** icon.
5. Click the **Actions** icon from the upper right side of the screen
6. Select **Create worksheet**.



7. Choose both views using **Choose Sources**.
8. You may need to rename some columns appropriately for searching.
9. Share the worksheet with the appropriate users and groups.

# Worksheet example scenarios

Here are three common examples of when you would need to use aggregated worksheets.

## Example 1

The first example involves creating an aggregated worksheet with a default filter. Say you want to create a worksheet that only shows data for a particular US state. In your search, enter `customer state = texas`. Then click **Actions**, and select **Save as worksheet**. Give your worksheet a name, then click **Save** to create your worksheet.

Now you have a worksheet that only contains data that pertains to Texas. You can share this worksheet with others to search across. Another popular example of this concept includes creating a worksheet with only active employee data.

## Example 2

The second example involves joining two aggregated worksheets. Say you want to plot the revenue of the top five states over time. Search for `revenue`, `store state`, and `top 5`. Save this answer as an aggregated worksheet called `Top 5 states`. Then start another search with the tokens `revenue`, `store state`, and `date`. Save this answer as an aggregated worksheet called `Total monthly purchases`.

Now you want to join these two worksheets. Navigate to the **DATA** tab and make a relationship between the two worksheets, involving store state.

To start a new search, select your two aggregated worksheets as data sources, selecting the appropriate columns: `store state` from `Top 5 States`, and `date` and `total sales` from `Total monthly Purchases`. You will only see data for the top five revenue states.

## Example 3

The third example involves creating a search to find customers who bought product A, but did not buy product B. This example can also be done using [conditional formulas](#).

First, you would have to perform searches for total sales by customer for both A and B, and create aggregated worksheets for both. Then join these two aggregated worksheets back in an outer join looking for conditions where the A and B join values are null.

This approach can become clunky, but depending on the requirement, can also be easily implemented.

# Basic pinboard usage

**Summary:** Create a new pinboard to group and manage related search results. Pinboards are the ThoughtSpot term for a dashboard.

Pinboards act like live dashboards, and are collections of your related charts, tables, and headline.

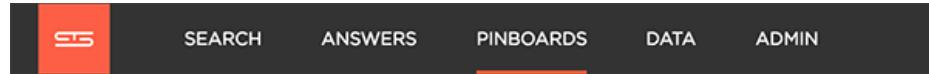
## What are pinboards

You can pin charts and tables to any pinboard which you created, and those that have been shared with you with the **Edit** privilege. When you create a pinboard, you can share it with other people with either the **View** or **Edit** privilege. Pinboards are interactive, allowing you to perform actions like filtering, excluding values, and drilldown on the visualizations.

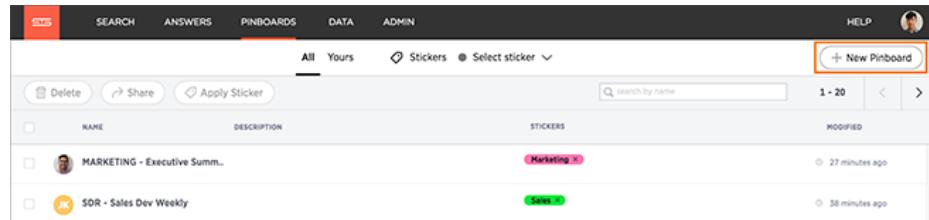
## Create a pinboard

To create a pinboard:

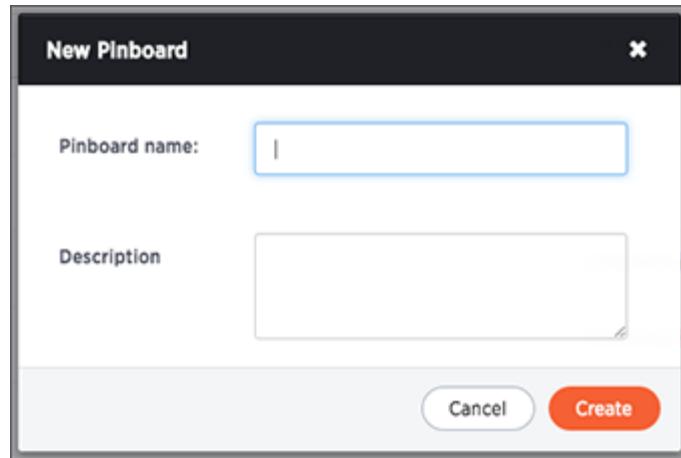
1. Click on **Pinboards**, on the top navigation bar.



2. Click **+ New Pinboard** on the pinboards list page.



3. In the **New Pinboard** dialog box, give your pinboard a name and description. Then click **Create**.



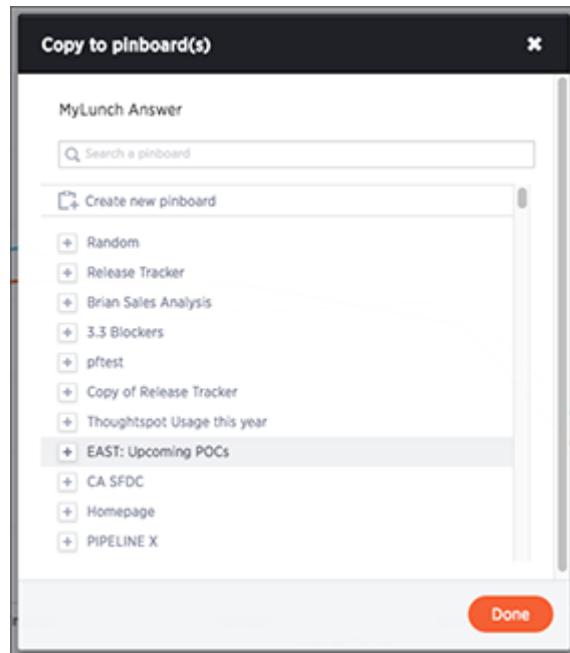
## Add an answer to a pinboard

Instead of saving an answer you can add it to a pinboard by pinning it. To add an answer to a pinboard:

1. While viewing your answer of interest, click the **Pin** icon on the top right of the answer.



2. In the **Copy to pinboard(s)** dialog box, click the + icons next to the pinboards you would like to add your answer to.

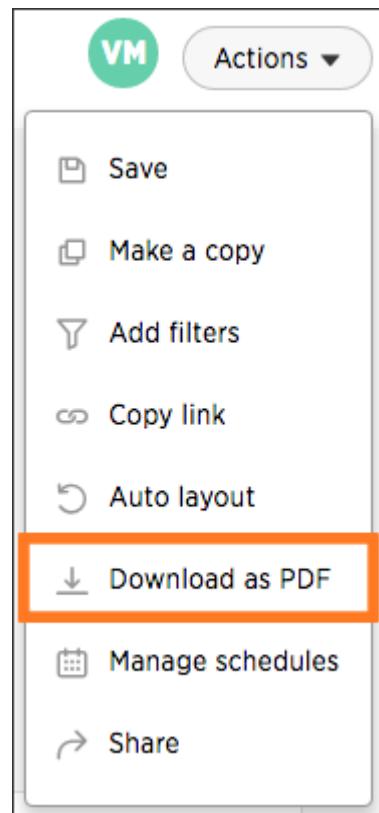


You can also select **Create new pinboard** at the top of the list to create a new pinboard, then add your answer to it.

## Download a pinboard

You can download a pinboard as a PDF file, without having to download each visualization separately. Downloading a pinboard works just as it would when downloading an answer. To download a pinboard:

1. Click **Actions**, and select **Download as PDF**.



## Details about a pinboard

You can see who authored a pinboard, when it was created, and when it was last modified by choosing **Actions > Pinboard Info**.

### Pinboard Info

Name  
AWS 3TB Performance testing

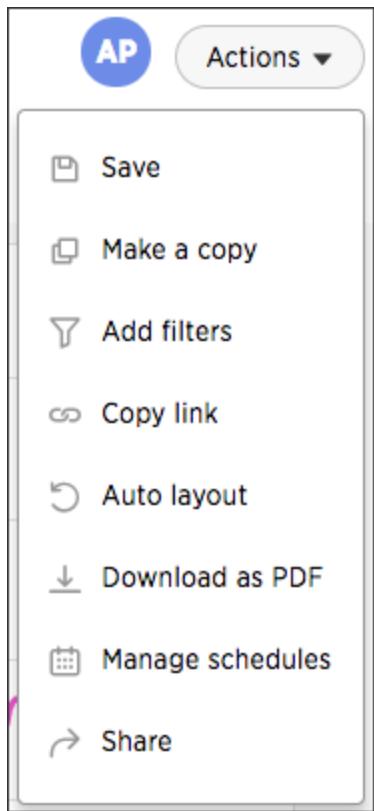
Author  
sunil.iyer@thoughtspot.int

Created	Modified
Mar 5, 2018, 12:13 PM	Mar 16, 2018, 2:00 PM

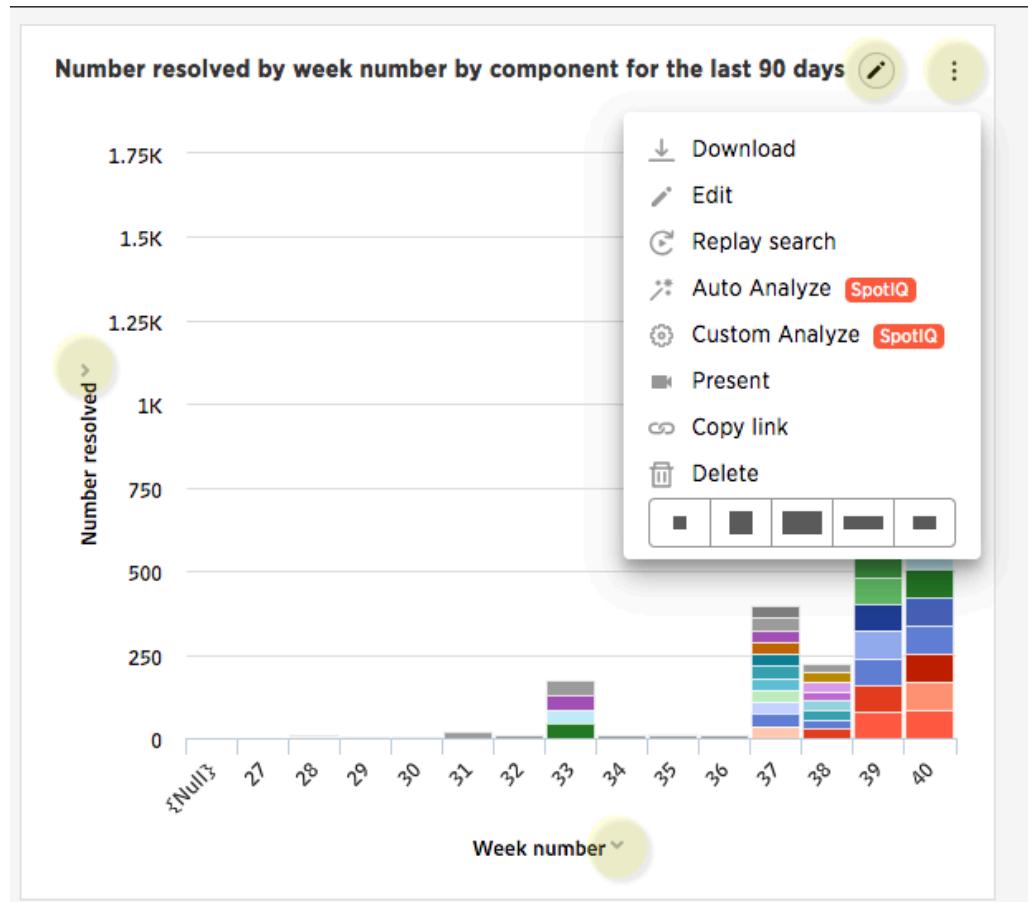
[CLOSE](#)

## Other pinboard actions

Other pinboard actions include actions you normally perform on your search as well as actions that you can only perform on a pinboard. Most of these actions are found under the **Actions** button.



Other actions are available on a pinboard display itself. Some actions are static others appear only when you over over specific locations of the pinboard. This diagram displays active action areas on a pinboard.



If you make a change to a pinboard, the system recognizes this and puts a **Save changes** button up in the header to alert you. Simply press this button to make your changes permanent.

# Edit the layout of a pinboard

**Summary:** Editing the layout of a pinboard lets you snap visualizations into place, choose between set visualization sizes, and reset your layout.

Visualizations within a pinboard are easy to move around and snap into place on a relative flow layout. Your pinboard layout is also responsive to your browser resolution. This helps keep the layout of your pinboard neat and organized.

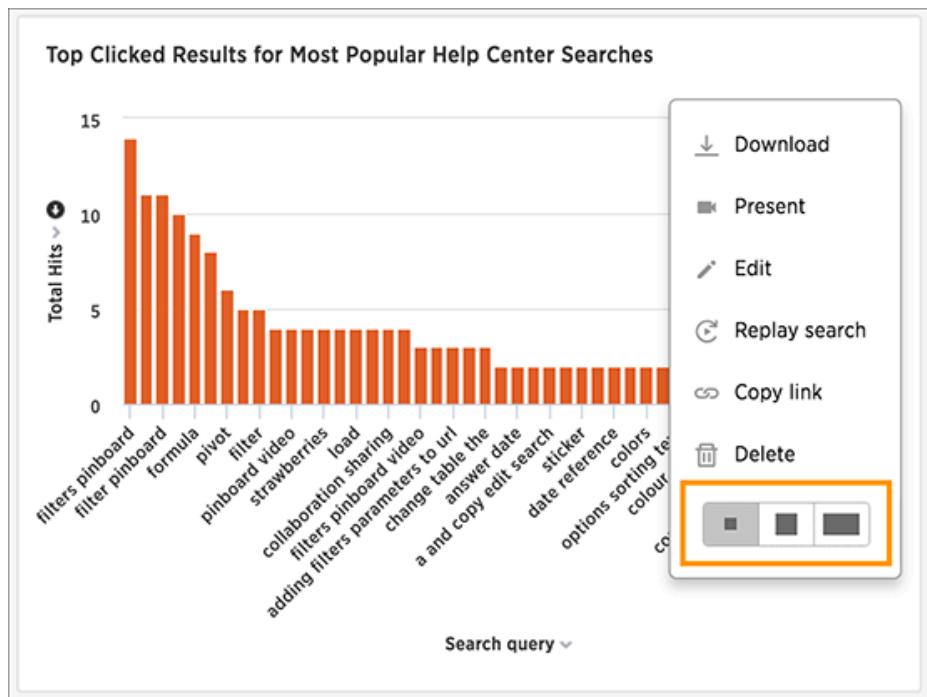
## Reorder the visualization

In addition, the size picker at the bottom of a visualization dropdown lets you toggle between predetermined sizes for each visualization. Charts and graphs can be toggled between a small, medium, and full width size, while headlines can be only one size (small). One row of the pinboard can hold a predetermined number of visualizations of each size.

1. Click on **Pinboards**, on the top navigation bar.

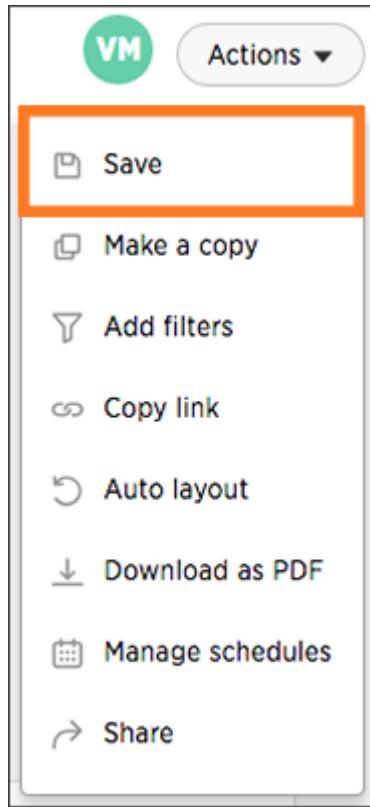


2. On the pinboard list page, click the pinboard you would like to edit.
3. Resize your visualizations by choosing between the predetermined sizes under the visualization dropdown menu.



4. Drag and drop your visualizations on the layout grid to reorder your pinboard.

5. If you are unhappy with your layout or you would like ThoughtSpot to configure your layout for you, go ahead and [reset your layout](#).
6. Save your pinboard by clicking **Actions** and **Save**.



## Change the orientation of a chart or table

You can change the size of a chart or table on a pinboard. To do this:

1. Navigate to the chart or table.
2. Select the menu.

The system displays the dropdown menu.

The image shows a table titled "Total Count by Yearly (Created)" with three rows: "FY 2017" (5,907), "FY 2018" (3,845), and "FY 2014" (2,801). A dropdown menu is open on the right side of the table, listing various actions: "Download as CSV", "Download as XLSX", "Download as PDF", "Edit", "Replay search", "Auto Analyze", "Custom Analyze", "Present", "Copy link", and "Delete". Below the menu are five small square icons.

Created YEARLY	Total count	Issue Key
FY 2017	5,907	
FY 2018	3,845	
FY 2014	2,801	

( showing rows 1-4 of 6 )

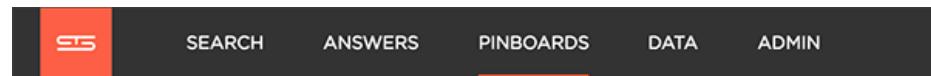
The last item shows the possible orientation

3. Select a new orientation.
4. Choose **Actions > Save**.

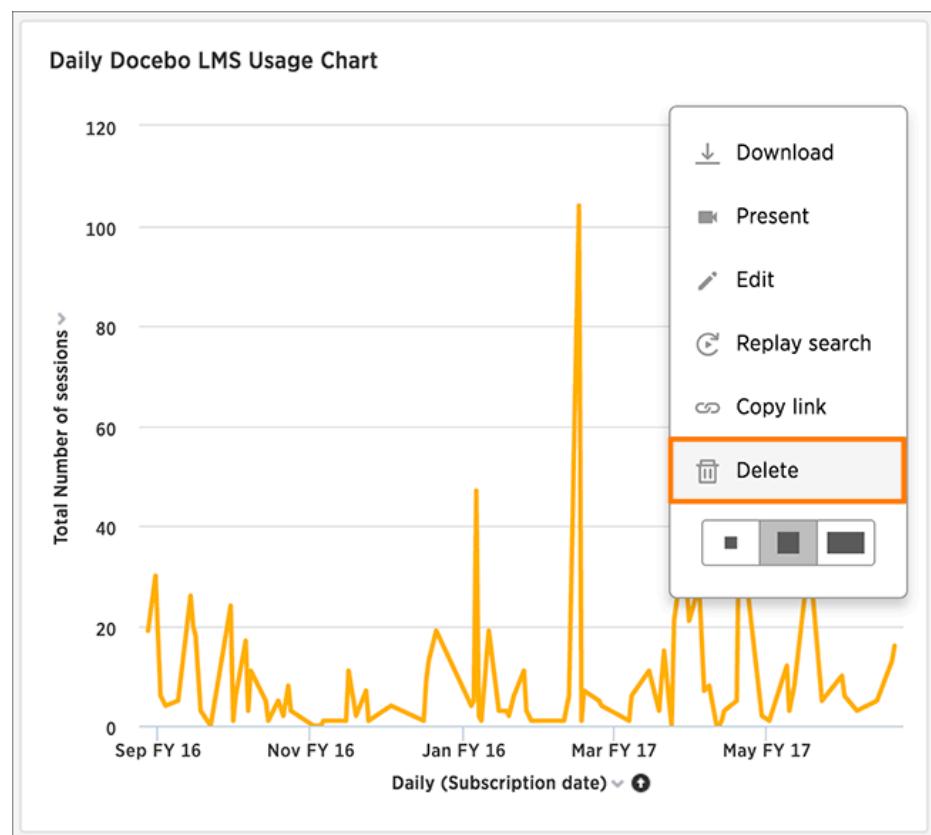
## Remove a visualization from your pinboard.

To delete a visualization:

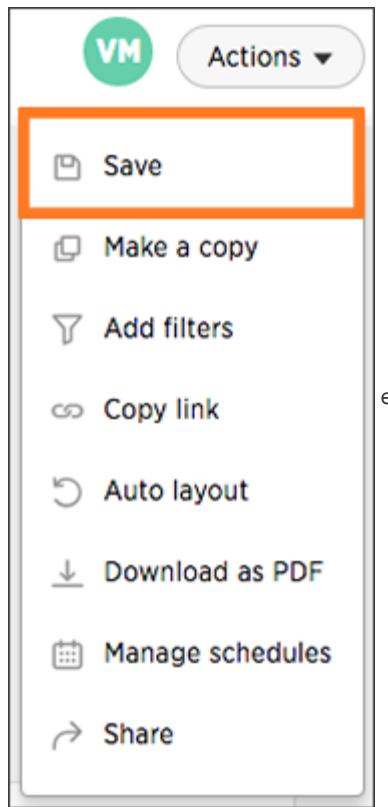
1. Click on **Pinboards**, on the top navigation bar.



2. On the pinboard list page, click the pinboard you would like to edit.
3. Click **Delete** under the dropdown menu of the visualization you would like to delete.

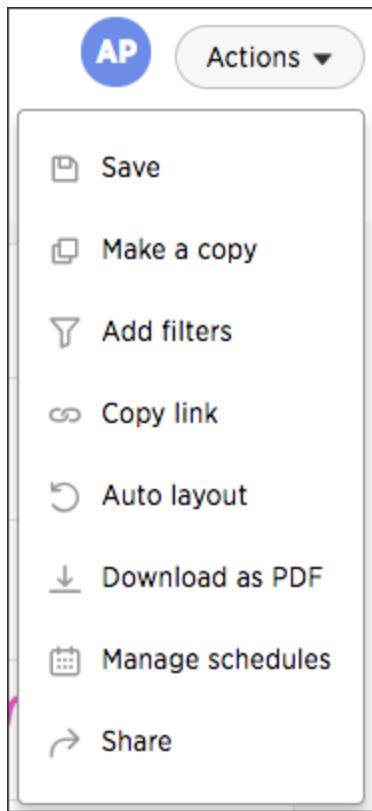


4. Save your pinboard by clicking **Actions** and **Save**.

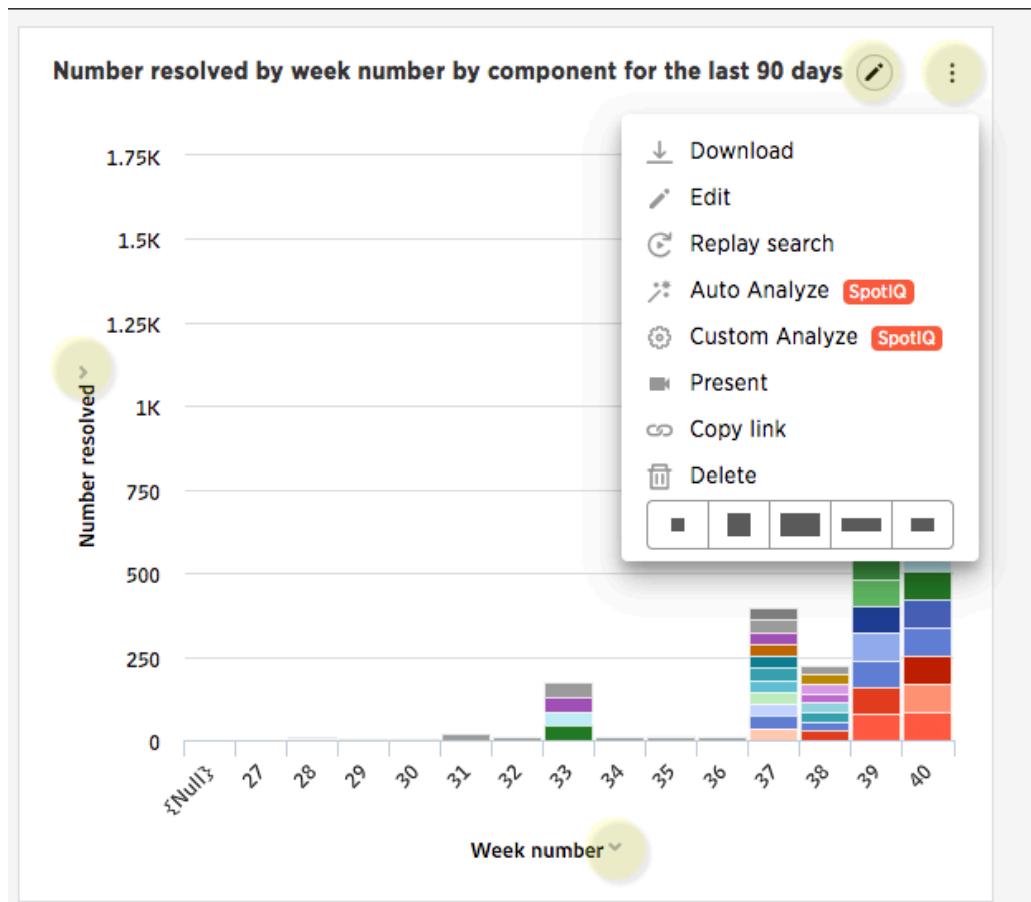


## Other pinboard edits

Other pinboard actions include actions you normally perform on your search as well as actions that you can only perform on a pinboard. Most of these actions are found under the **Actions** button.



Other actions are available on a pinboard display itself. Some actions are static others appear only when you over over specific locations of the pinboard. This diagram displays active action areas on a pinboard.



# Pinboard filters

Filters, including bulk filters, can be applied to pinboards just as with tables and charts. These kinds of filters apply to an entire pinboard, making it easy to see only the data that you are interested in across the tables and charts within a pinboard.

## When to use a pinboard filter

Pinboard filters can be very useful when you want to apply the same filters to more than one related visualization. You can narrow the focus of your pinboard for specific purposes or audiences.

When you apply a filter, the pinboard is not automatically saved with your filter applied. This is to encourage ad hoc filtering. Therefore, people with read-only access can create pinboard filters. You must have edit access to the pinboard, and view access (or higher) to the underlying data source in order save a pinboard filter.

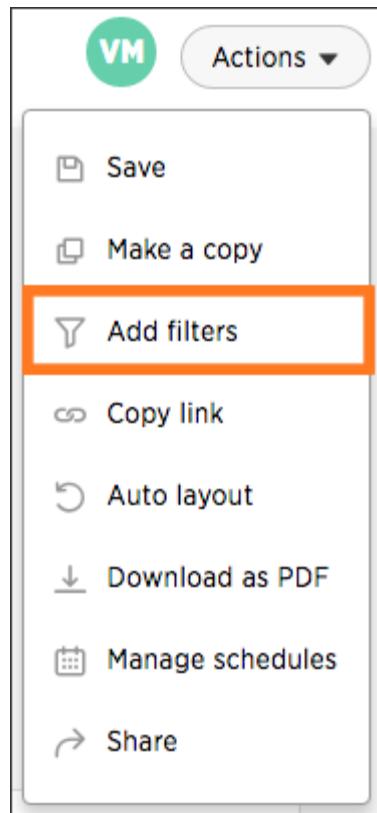
Note that pinboard filters only apply to the tables and charts that are based on worksheets. If a pinboard also includes tables and charts that were created from underlying tables or on user uploaded data, the filters don't apply to them.

## Create a pinboard filter

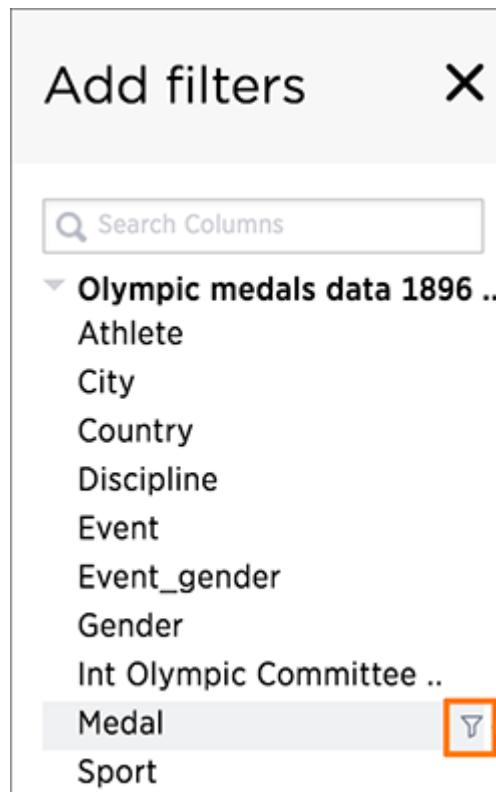
You can create a filter in the pinboard view for any pinboard. This allows you to easily manipulate the visualizations and view the modified presentation in one place.

Here is an example that shows you how to add a single filter to a pinboard. To create a pinboard filter:

1. Click the **Actions** button, and click **Add filters**.



2. In the populated columns menu, click the **Add filter** icon next to the columns you would like to use as filters.



3. Choose the values you are interested in by typing them in, selecting the appropriate checkboxes, or using **Add values in bulk**.
4. Click **Done**.

When you apply a filter, the pinboard is not automatically saved with your filter applied to every object in the pinboard that was created from a worksheet. Pinboard filters do not apply to tables and charts built directly on top of base tables.

## Edit a filter from the answer

Your applied filter will sit at the top of the pinboard, where you can edit it in place:

**Add filters** X

Search Columns

▼ Olympic medals data 1896 ..

- Athlete
- City
- Country
- Discipline
- Event
- Event\_gender
- Gender
- Int Olympic Committee ..
- Medal
- Sport
- Year

Olympics Medals Analysis

Medal: gold ▾

Search by name

Clear all | Select all

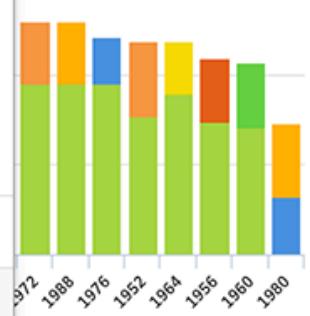
gold

bronze

silver

Add values in bulk

Cancel Done



Or delete it:

Olympics Medals Analysis

Medal: gold ▾ X

# Schedule a pinboard job

**Summary:** You can schedule a pinboard job for any pinboard by using the Add a schedule prompt page.

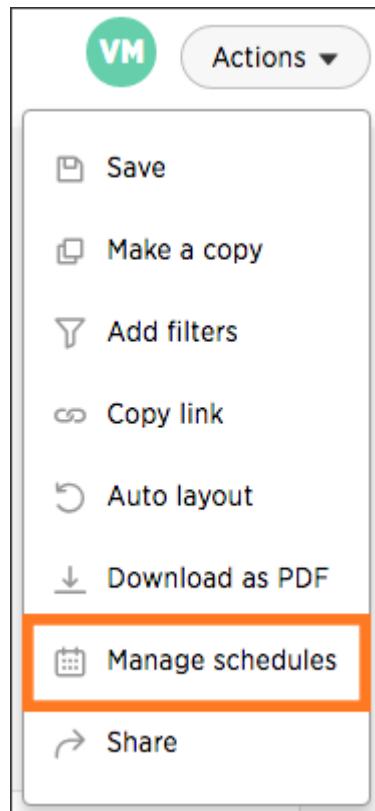
You can add multiple schedules with different configurations for a single pinboard. However, each job is limited to one pinboard schedule. To add a schedule, you must have administrator or can schedule pinboard privilege, and at least edit-only and view-only access to the pinboard.

To schedule a pinboard:

1. Log in to ThoughtSpot from a browser.
2. Click on **Pinboards**, on the top navigation bar.



3. Select the pinboard you would like to create a schedule for.
4. Click **Actions** and select **Manage schedules** to view all of the schedules set for the pinboard.



5. Click **+ Schedule** to add a new schedule.

Schedules					
	NAME	DESCRIPTION	STATUS	RECIPIENT	CREATED
<input type="checkbox"/>	sc test	test	▷ Scheduled	1 Recipient	🕒 2 hours ago
<input type="checkbox"/>	sc max jobs test	max jobs	▷ Scheduled	1 Recipient	🕒 2 hours ago
<input type="checkbox"/>	sc header	header	▷ Scheduled	1 Recipient	🕒 2 hours ago

The system displays the **Add a schedule for PINBOARD** page. On this page you configure both the schedule

- Set the values for your schedule.

### Add a schedule for rdata

Schedule	Recipients
Repeats: <input type="button" value="Select an Option"/> Server time zone: UTC <b>Name*</b> : <input style="border: 2px solid red; width: 150px; height: 20px;" type="text"/> Description: <input style="width: 250px; height: 40px;" type="text"/> Type: <input type="radio"/> CSV <input checked="" type="radio"/> PDF Gating condition: <input type="button" value="+ Add condition"/> <input type="button" value="Delete"/>	Recipients <input type="button" value="Add Recipients"/> Users or groups: <input type="text"/> <input type="button" value="Add &amp; Save"/> Emails: <input type="text"/> <input type="button" value="Add"/>

Field	Description
<b>Repeats</b>	You can rebuild a pinboard every <i>n</i> minutes, hourly, daily, weekly, or monthly. For some of these, you can also choose specific times of the day or days of the week. Make sure to note the <b>Server time zone</b> which is the timezone which will be used.
<b>Name</b>	Provide a short name for this schedule, <b>Monthly Report Source</b> is an example of a good name.
<b>Description</b>	Enter a description to remind yourself and to inform others. For example, if this is an important schedule for a meeting or a report, you should mention that.
<b>Type</b>	CSV files provide all data for tables, with one attachment per table. Use CSV files to perform further analysis offline. PDF files show all visualizations in the pinboard. Each chart takes up a whole page in the file, while only the first 100 rows of a table are included. Use PDF files to skim the data.

<b>Gating condition</b>	Write a statement that returns a single boolean value (true or false). For example, <code>sum (revenue) &gt; 100</code> is a valid condition but <code>is_weekend (commit_date)</code> is not. You can use any valid formula in your statement. Other than checking your formula syntax, ThoughtSpot does not validate your formula returns a single boolean.
-------------------------	---

At the scheduled time, ThoughtSpot checks the gating condition, and, if the condition returns true, processes the pinboard. For a list of valid formulas, see the [Formula function Reference](#).

**Recipients**

You are limited to 1000 recipients per job. You can add **Users or groups** to enter ThoughtSpot users or groups. Use **Emails** to add recipients that are not ThoughtSpot users. Any users your ThoughtSpot admin must have set your whitelist domains. Contact ThoughtSpot Support to set your whitelist domains.

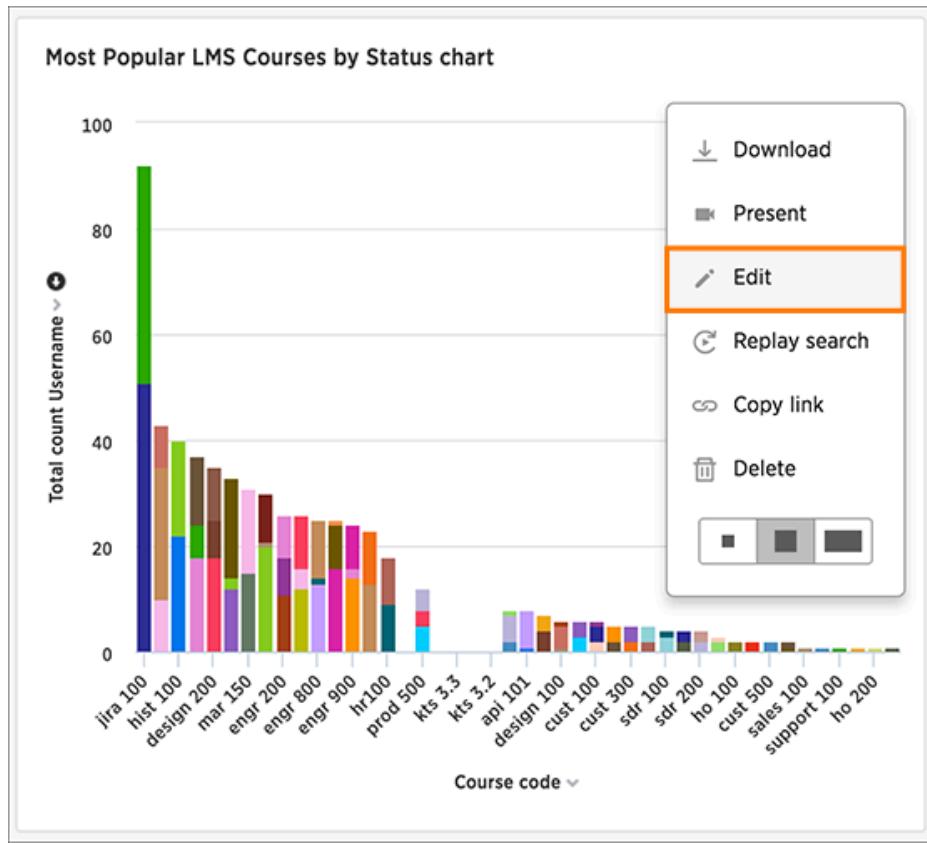
If you experience problems with sending email, your cluster may not have SMTP enabled. Contact your system administrator to resolve this issue.

7. Click **Schedule** to save your schedule configuration.

# Search actions within a pinboard

You can perform many of the same search actions on individual visualizations within a pinboard as you can within a search. You can interact directly with a visualization of a pinboard to perform ad hoc searches or edit it. These search actions include the following:

- Editing a search - You can edit the original search and reconfigure the answer. In the dropdown of a visualization, click **Edit**.



You will be taken to an edit mode, where you can view and change sources, search different columns, change the view, save the answer as a worksheet, add a formula, and change the chart.

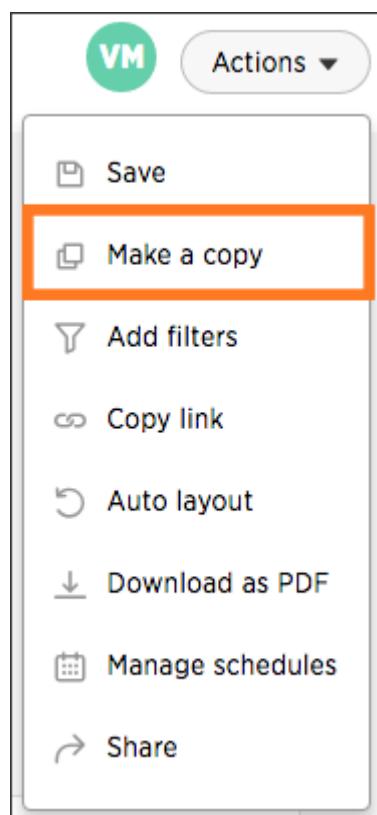
# Copy a pinboard

You can make a copy of a pinboard if you would like to make edits without changing the original pinboard. Making a copy of a pinboard allows you to make your own edits without overwriting the original. When saving a copy, you can type in a new name.

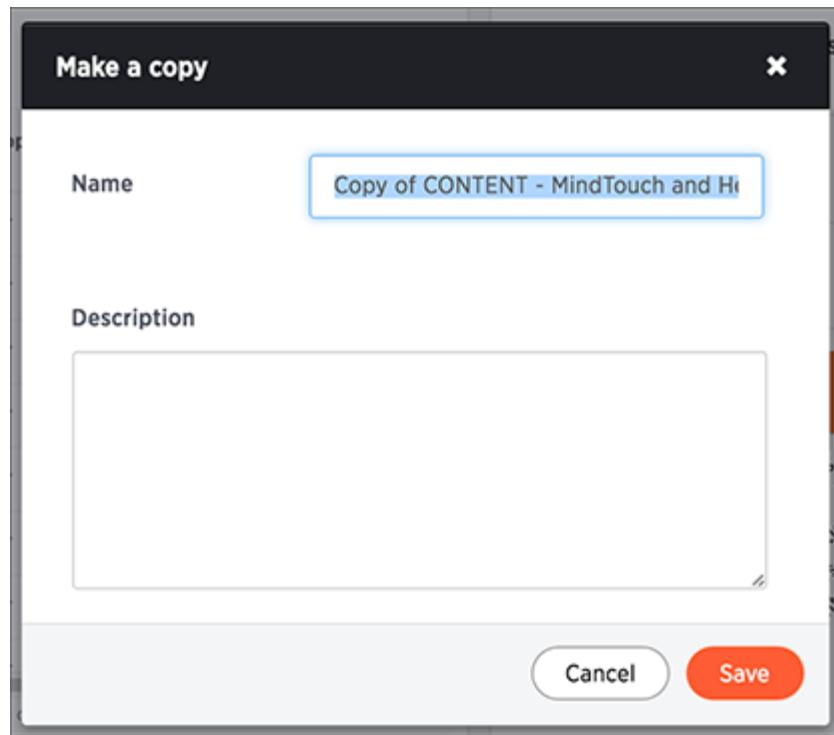
1. Click on **Pinboards**, on the top navigation bar.



2. On the pinboard list page, click the pinboard you would like to copy.
3. Click **Actions** and select **Make a copy**.



4. Give your pinboard a new name and description. Then click **Save**.



# Copy a pinboard or visualization link

**Summary:** In pinboards, there is a copy link option that lets you copy the link to access the pinboard and visualizations directly.

You can copy and paste the copied link to include in a presentation or spreadsheet, or email or Slack it to other people in your company. Note that when clicking the link, the person must be authenticated to ThoughtSpot to see the visualization (e.g. by logging in or LDAP).

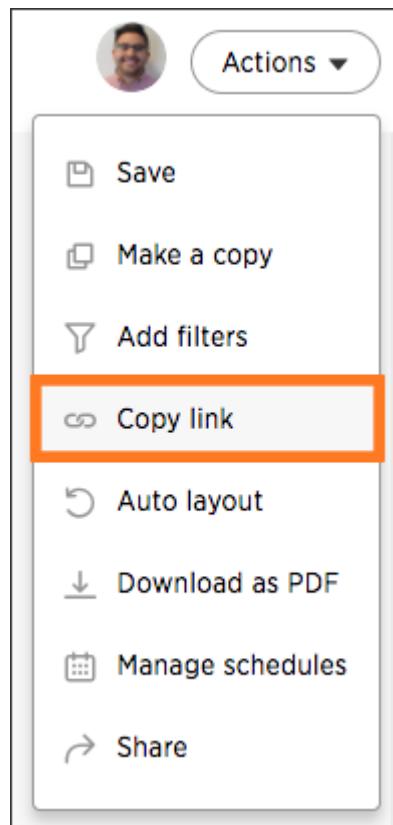
You can also use this link for embedding the chart or table in another Web page, Web portal, or application when using the ThoughtSpot JavaScript API with Embedding or the REST API. For details, read the ThoughtSpot Application Integration Guide.

To copy the link for a pinboard:

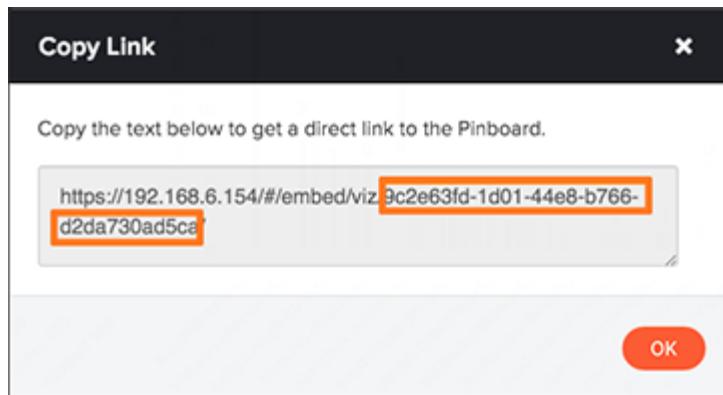
1. Click on **Pinboards**, on the top navigation bar.



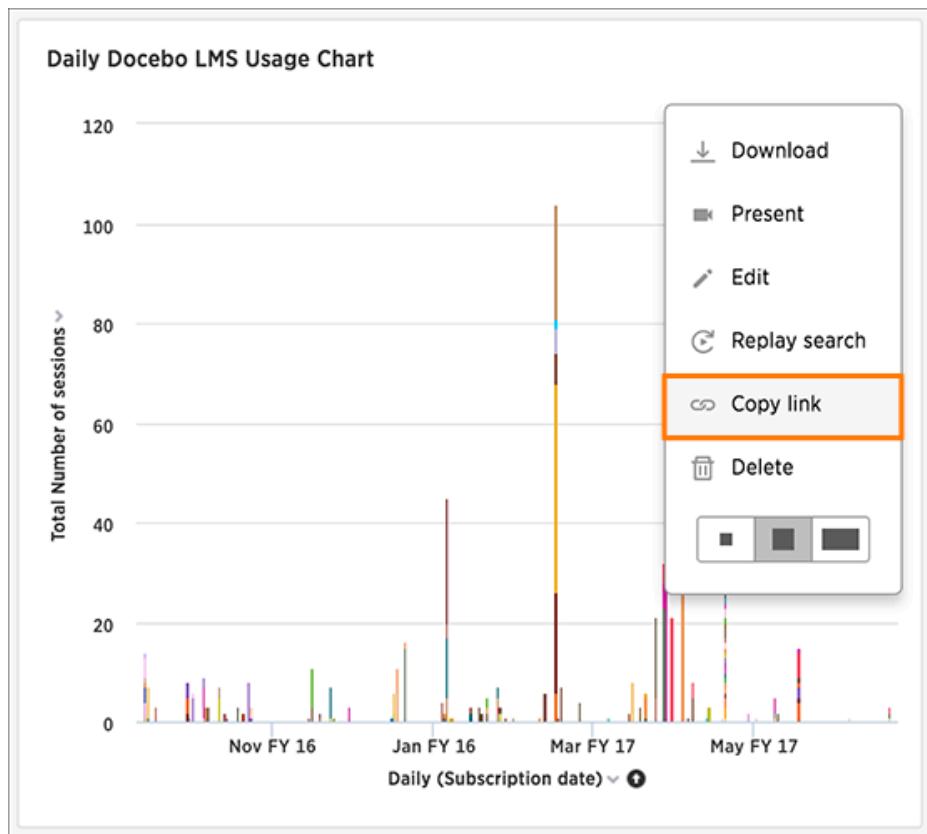
2. On the pinboard list page, click the pinboard you would like to get a link for.
3. Click **Actions** and select **Copy link**.



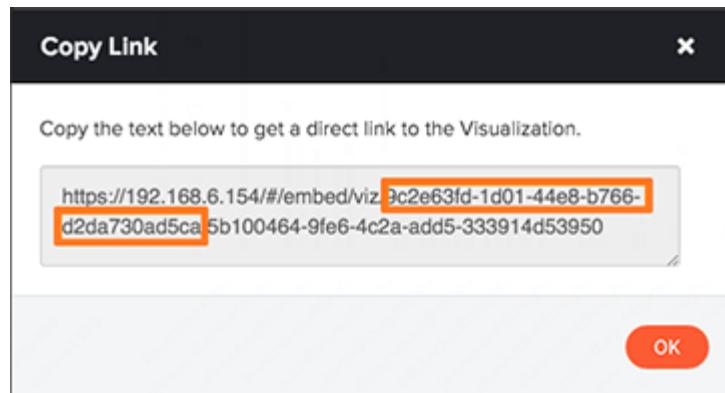
4. Copy the pinboard link. The highlighted portion is the pinboard ID.



5. To copy an individual visualization link, click **Copy link** under the dropdown menu of the visualization you would like to get a link for.



6. Copy the visualization link. The highlighted portion is the visualization ID.



# Reset a pinboard or visualization

You can undo your edits to a pinboard or visualization by resetting it

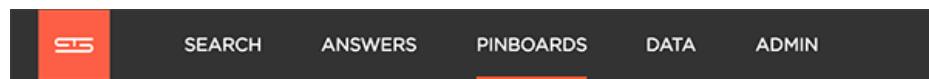
## Reset the layout of a pinboard

Reset the layout of a pinboard to undo your changes or have ThoughtSpot create a layout for you.

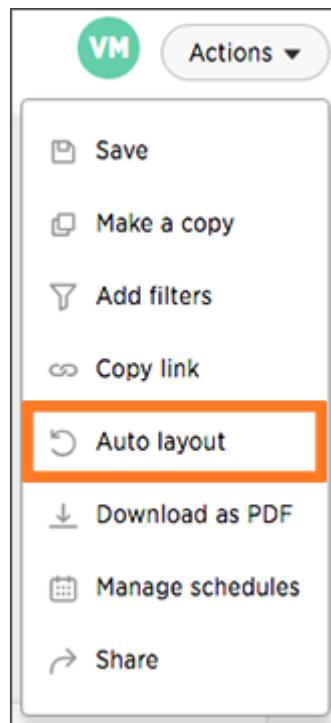
You can reset the layout of your pinboard if you would like to undo your edits or have ThoughtSpot optimize your layout space for you. This action will keep the sizes of your visualizations, but not the order.

To reset the layout of a pinboard:

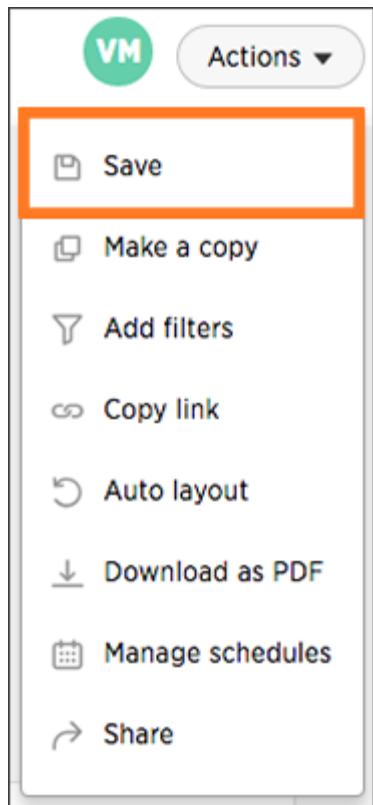
1. Click on **Pinboards**, on the top navigation bar.



2. On the pinboard list page, click the pinboard you would like to edit.
3. Click **Actions** and select **Auto layout**.



4. Save your pinboard by clicking **Actions** and **Save**.

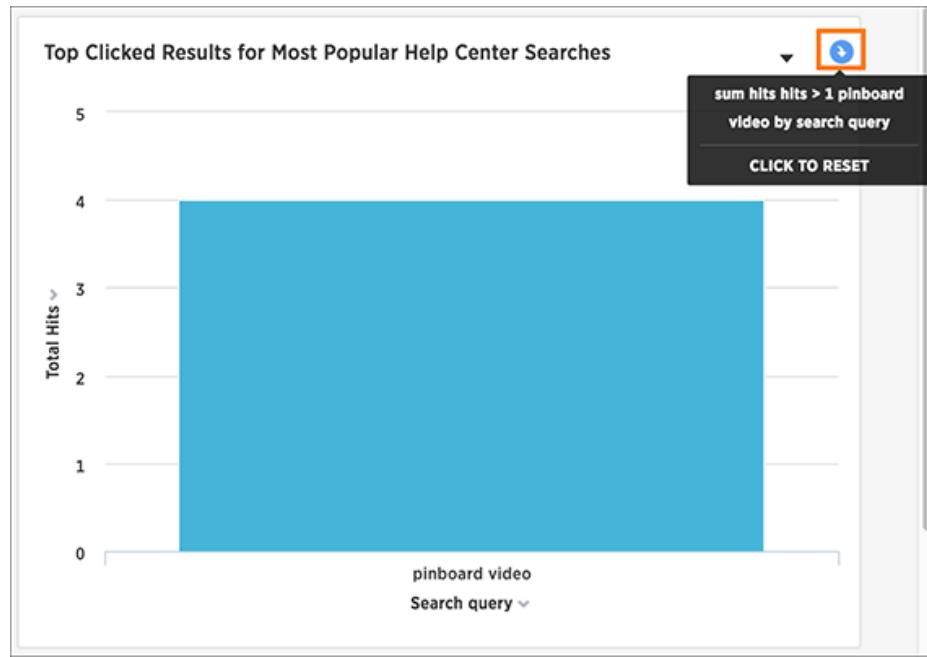


## Reset a visualization

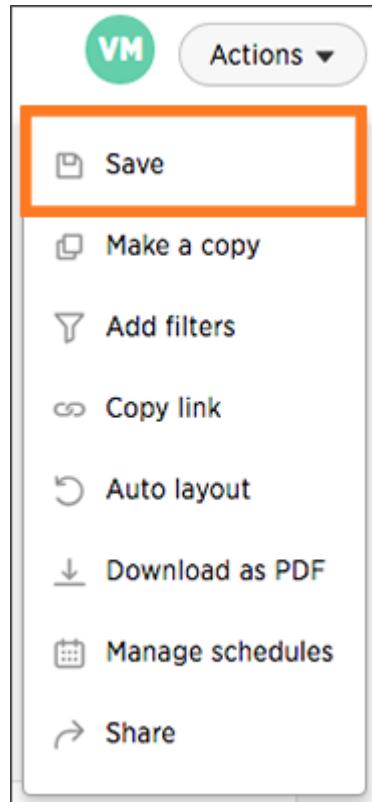
After performing ad hoc actions or edits to a visualization of your pinboard, you can reset the visualization to its original form.

To reset a visualization:

1. On an altered visualization, click the reset icon.



2. Save your pinboard by clicking **Actions** and **Save**.



# Start a slideshow

**Summary:** Displaying your pinboard as a slideshow is a good way to present its contents to others.

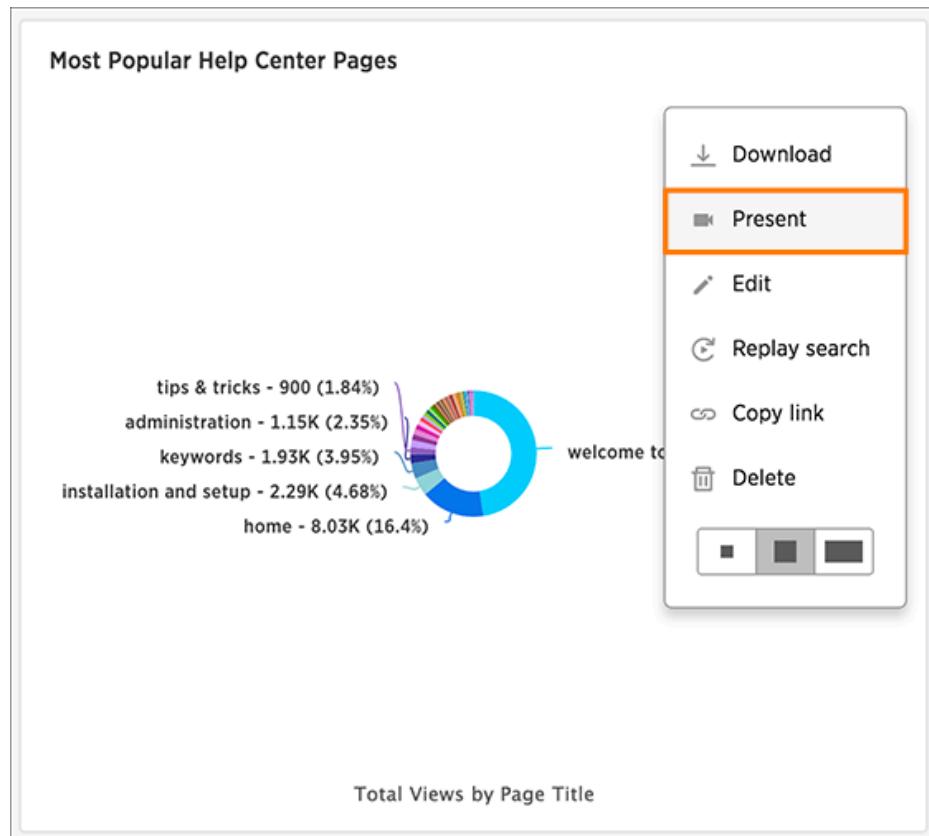
Presenting your pinboard displays your visualizations in order from left to right and top to bottom.

To start a slideshow:

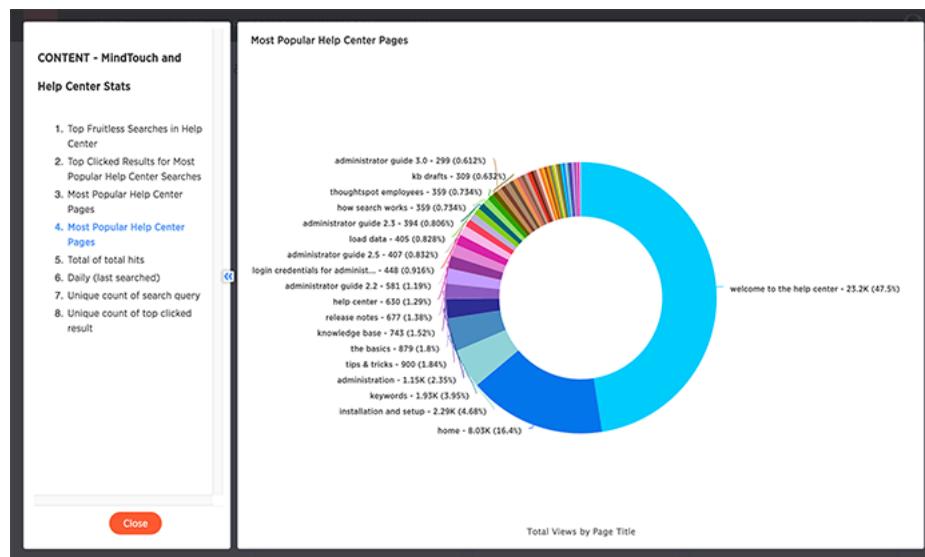
1. Click on **Pinboards**, on the top navigation bar.



2. On the pinboard list page, click the pinboard you would like to present.
3. Click **Present** under the dropdown menu of the visualization you would like to start the slideshow with.



4. Use the left and right arrow keys to navigate between your pinboard's visualizations.



5. Click the x at the top right of the screen or push the Esc key to exit the slideshow view.

# Understand data sources

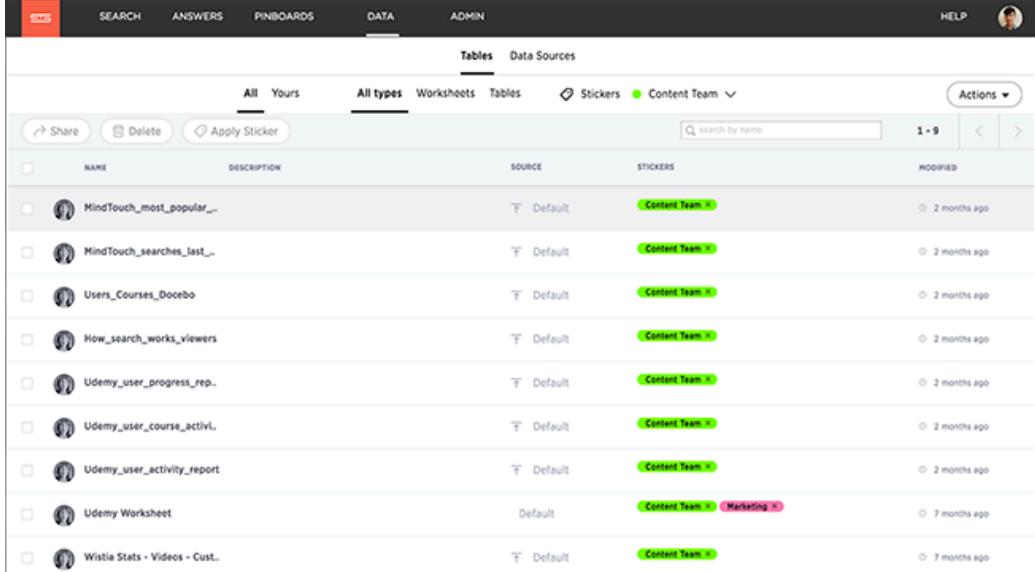
**Summary:** Use the DATA tab to manage data sources.

The **DATA** tab lists all of the tables and data sources available to you. There are three types of data sources that you may see in the data list. They are tables, worksheets, and user uploaded data. You will most likely only see worksheets and user uploaded sources. These are the most commonly used data sources for searching.

Name	Icon	Definition	Created By
Table	 Default	Raw tables loaded by an Administrator.	Administrators
Worksheet		Collection of related tables, optimized for searching. (Like a view).	Anyone
User Imported	 Imported	Table uploaded by a user through the Web browser.	Anyone

Clicking on the name of a table or data source shows you detailed information about it. You won't be able to change these settings or edit the table unless it was shared with you with the **Edit** privilege. To see how to edit a data source, refer to the *ThoughtSpot Administrator Guide*.

Use the filters at the top of the page to find the data you are interested in. From the **DATA** tab, you can also delete or apply stickers to tables and data sources in bulk by selecting them and clicking the appropriate action button.



The screenshot shows the ThoughtSpot interface with the 'DATA' tab selected. At the top, there are tabs for 'Tables' and 'Data Sources'. Below that, a search bar and a filter section with dropdowns for 'All', 'Yours', 'All types', 'Worksheets', 'Tables', 'Stickers', and 'Content Team'. A 'Actions' dropdown and a search bar are also present. The main area displays a list of data sources with columns for 'NAME', 'DESCRIPTION', 'SOURCE', 'STICKERS', and 'MODIFIED'. Each row contains a small profile picture, the data source name, a brief description, the source type (e.g., 'Default'), any applied stickers (e.g., 'Content Team'), and the last modified date.

NAME	DESCRIPTION	SOURCE	STICKERS	MODIFIED
MindTouch_most_popular...		Default	Content Team	2 months ago
MindTouch_searches_last...		Default	Content Team	2 months ago
Users_Courses_Docebo		Default	Content Team	2 months ago
How_search_works_viewers		Default	Content Team	2 months ago
Udemy_user_progress_re...		Default	Content Team	2 months ago
Udemy_user_course_activi...		Default	Content Team	2 months ago
Udemy_user_activity_report		Default	Content Team	2 months ago
Udemy Worksheet		Default	Content Team, Marketing	7 months ago
Wistia Stats - Videos - Cust...		Default	Content Team	7 months ago

## Related information

- [View your data profile](#)

Basic column data profile information is available under Profile on the Data page.

- [About sharing](#)

Whenever you are working in ThoughtSpot, you are in your own private environment until you share your work with others. This applies to searches, pinboards, and any data you upload.

# Create and load a CSV file

**Summary:** The simplest way to load data is to upload a CSV or Excel file from the ThoughtSpot Web interface.

Loading data through the Web browser is recommended for smaller tables (under 50MB) with simple relationships between them. This method is recommended for small, one time data loads. Using this method, the data schema is created for you automatically.

Any user who belongs to a group that has the privilege **Has administration privileges** or **Can upload user data** can upload their own data from the browser.

Your data should be in a CSV (comma separated values) before you load it. A CSV file is a text file made up of data fields separated by a delimiter and optionally enclosed with an enclosing character. If your data contains multiple tables, you'll have a separate CSV for each table.

## Create a CSV file

The first step in loading data is to obtain or create one or more CSV files that contain the data to be loaded into ThoughtSpot. CSV is a common format for transferring data between databases. ThoughtSpot requires this format.

Most applications such as Microsoft Excel or Google Sheets can output CSV formatted files. If your source is an Excel spreadsheet or Google Sheet:

1. Save, export, or download the file in CSV format. The exact procedure you use will depend on the source application.
2. Review the file's format before uploading it to ThoughtSpot.

Your source data may be in another database. If this is the case, your company's ETL (extract, transform, load) process will typically generate CSV files. If your source is another database:

3. Connect to the source database.
4. Extract each table you wish to import into ThoughtSpot as a CSV file.

The column delimiter should be a `,` (comma), `|` (pipe), or tab.

Large organizations typically have a data administrator or department that builds ETL processes. If the data you want is in another system, speak with your ThoughtSpot administrator about getting CSV files extracted from this system.

## Formatting the CSV

A CSV file contains a delimiter that marks the separation between fields in the data. The delimiter is usually comma, but it can be any character. The file also contains fields optionally enclosed with double quotes. Use these guidelines when creating the CSV file:

- If the CSV contains column headers, they must match the column names in the database exactly.
- Often a `|` (pipe) or tab is used as the delimiter, because it may be less likely to occur within the data values.
- When a field contains a double quote, it must be escaped with the character specified in the

escape character argument in `tsload`.

- When a field contains the delimiter, the field must be enclosed in double quotes.

ThoughtSpot supports a wide range of [date and timestamp formats](#) in the CSV file. Blank values in user uploaded CSV files are interpreted as NULL values. These include the values (case insensitive):

- `NULL`
- `\N`
- `NA`
- `N/A`
- [space]

If you are appending data to an existing schema or table, columns in the CSV file must be in the same order as defined in the target table.

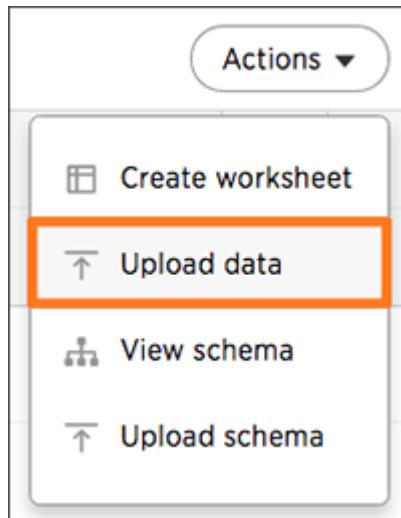
## Load the CSV File

Any user who belongs to a group that has the privilege **Has administration privileges** or **Can upload user data** can upload their own data from the browser. To load the CSV or Excel file into ThoughtSpot:

- Log into ThoughtSpot from a browser.
- Click on **DATA**, on the top navigation bar.



- Click the **Actions** button in the upper right corner, and select **Upload Data**.

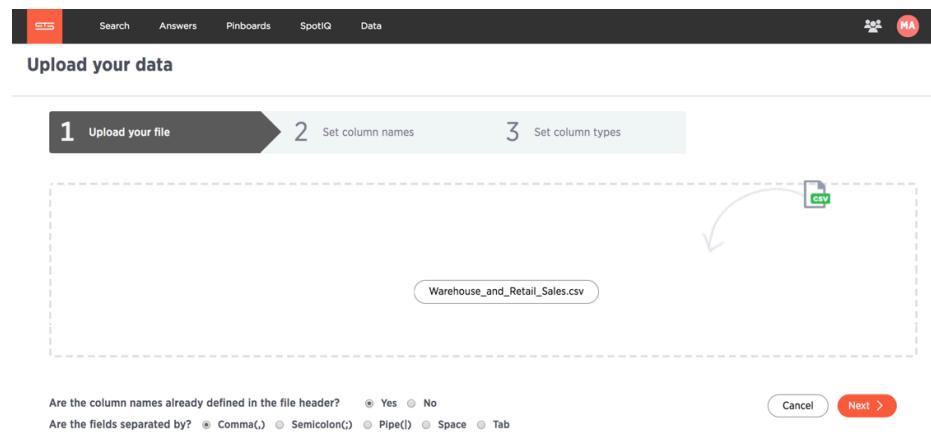


- Upload the CSV or Excel file by doing one of these options:
  - Click on **Browse your files** and select the file.
  - Drag and drop the file into the drop area.
- Answer the question **Are the column names already defined in the file header?**
- Answer the question **Are the fields separated by?** Click **Next**.
- Click on the column header names to change them to more useful names, if you'd like. Click **Next**.
- Review the automatically generated data types for each column, and make any changes you

want. There are four data types: Text, Integer, Decimal, and Date.

#### 9. Click **Import**.

When an upload is complete, the system reports the results and offers you some further actions.



- Click **Link to Existing Data** if you want to link the data you uploaded to the data in another table or worksheet.
- Click **Search** if you want to begin a new search.
- Click **Auto analyze** if you want to use the SpotIQ feature to find insights in your new data.

## Troubleshoot uploads

Boolean data must use **TRUE / FALSE** values. Other values such as **Y / N** are not supported.

# Append data through the UI

**Summary:** Use append to add more data to an existing data source.

If you have permissions to upload data and permissions to a data source, you can add to that source by uploading more data with CSV file. You can append data to your existing system tables through the ThoughtSpot application, even if the tables were initially loaded using Data Connect or `tsload`. The CSV file must have the same structure as the table it is being loaded into, including number and type of columns, in the same order as the target table.

To append data into ThoughtSpot:

1. Log in to ThoughtSpot from a browser.
2. Click on **Data**, on the top navigation bar.



3. Click the on the table you would like to append data to.
4. Click the **Load data** button.

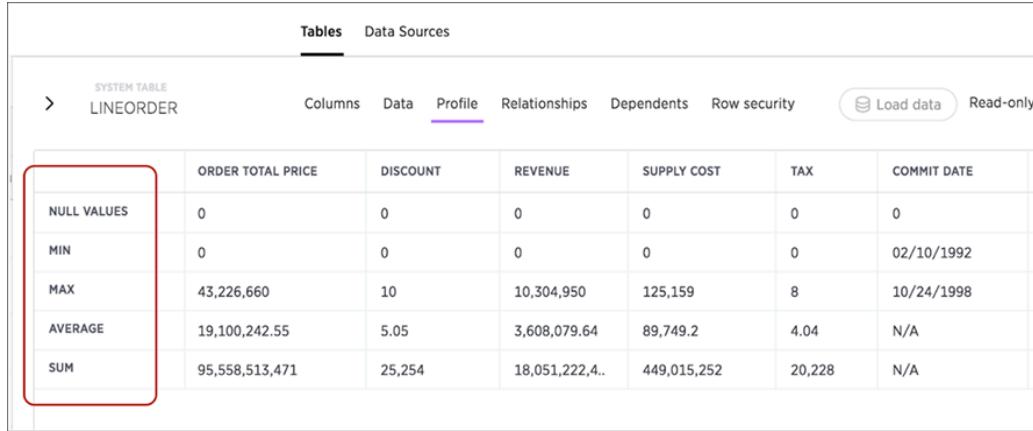
A screenshot of the 'Load data' configuration screen. At the top, there's a table header with columns: COLUMN NAME, DESCRIPTION, DATA TYPE, COLUMN TYPE, ADDITIVE, AGGREGATION, and HIDDEN. Below the header, there are three rows of data:

COLUMN NAME	DESCRIPTION	DATA TYPE	COLUMN TYPE	ADDITIVE	AGGREGATION	HIDDEN
Company	Click to edit	VARCHAR	ATTRIBUTE	<input type="radio"/> NO	NONE	<input type="radio"/> NO
Number of emplo..	Click to edit	INT64	MEASURE	<input checked="" type="radio"/> YES	SUM	<input type="radio"/> NO

5. Upload the CSV or Excel file by doing one of these options:
  - Click on **Browse your files** and select the file.
  - Drag and drop the file into the drop area.
6. Answer the question **Are the column names already defined in the file header?**
7. Answer the question **Do you want to append to the existing data or overwrite it?**
8. Answer the question **Are the fields separated by?** Click **Next**.
9. Click on the column header names to change them to more useful names, if you'd like. Click **Next**.
10. Review the automatically generated data types for each column, and make any changes you want. There are four data types: `Text`, `Integer`, `Decimal`, and `Date`.
11. Click **Import**.
12. Click **Link to Existing Data** if you want to link the data you uploaded to the data in another table or worksheet. Or click **Search** if you want to begin a new search.

# View a data profile

After you upload a CSV file, it is available as a table in ThoughtSpot. The **Profile** tab appears after you have selected a specific table from **DATA**.



	ORDER TOTAL PRICE	DISCOUNT	REVENUE	SUPPLY COST	TAX	COMMIT DATE
NULL VALUES	0	0	0	0	0	0
MIN	0	0	0	0	0	02/10/1992
MAX	43,226,660	10	10,304,950	125,159	8	10/24/1998
AVERAGE	19,100,242.55	5.05	3,608,079.64	89,749.2	4.04	N/A
SUM	95,558,513,471	25,254	18,051,222,4..	449,015,252	20,228	N/A

The data profile includes null values, min, max, average, and sum information for each table column. This **Profile** view should help you get a better sense of what's there before searching on the data.

# Set your display language (locale)

The language the ThoughtSpot UX displays is based off of the locale in a user's profile. The locale preferences control the language and data formats (date and number formats) by geographic locations. In addition to American English (*en-US*), ThoughtSpot supports:

Locale	Language
<i>de-DE</i>	German
<i>en-CA</i>	Canadian English
<i>en-GB</i>	United Kingdom English
<i>en-US</i>	English (United States)
<i>es-US (beta)</i>	Spanish (United States)
<i>fr-CA (beta)</i>	French (Canada)
<i>fr-FR (beta)</i>	French (France)
<i>ja</i>	Japanese
<i>pt-BR (beta)</i>	Portuguese (Brazil)
<i>zh-CN (beta)</i>	Chinese (S)

Date and number formats change to reflect your locale. So, if you set Japanese as your default locale in your profile settings, then the interface will update to reflect that after you refresh your page.

Keywords, operators, and error messages are included in the translated material. Formulas, however, are *not translated*. Also, all metadata remains as user inputted.

[Update my preferences](#)

The screenshot shows a user interface for setting preferences. At the top, there is a link to 'Update my preferences'. Below it, there are two sections: 'Preferred Locale' and 'Update my SpotIQ preferences'. The 'Preferred Locale' section contains a dropdown menu with the following options: en-US (selected), en-CA, en-GB, en-US, de-DE, and ja. The 'Update my SpotIQ preferences' section contains two checkboxes: 'Email me sharing notifications' (checked) and 'Email me analysis notification on success' (checked). The 'en-US' option in the dropdown menu is highlighted, indicating it is the selected locale.

For example, if you are using ThoughtSpot in the US, the number formatting should look like this:  
`XXX,XXX.XX`. And in Europe, it should look like this: `XXX.XXX,XX`.

**⚠ Warning:** It is possible that you set your locale but find strings in the UI still appear in English, this indicates an untranslated string. Please notify ThoughtSpot support.

# What is SpotIQ?

SpotIQ is a ThoughtSpot feature that helps you find insights into your data. Insights are trends, correlations, explanations of increases, explanations of decreases, and outliers (values unique from what is typical in your data).

Using SpotIQ, you can find interesting answers in your data that you might not have found on your own. SpotIQ also learns from your responses to your insights. Meaning, how you interact with SpotIQ results helps to build better results.

## Who can use SpotIQ?

The SpotIQ feature is made for users who are not data magicians; however, data magicians can use it to do power data magic. This means SpotIQ is for everyone.

If you can see the **SpotIQ** on your ThoughtSpot dashboard, you have access to SpotIQ:



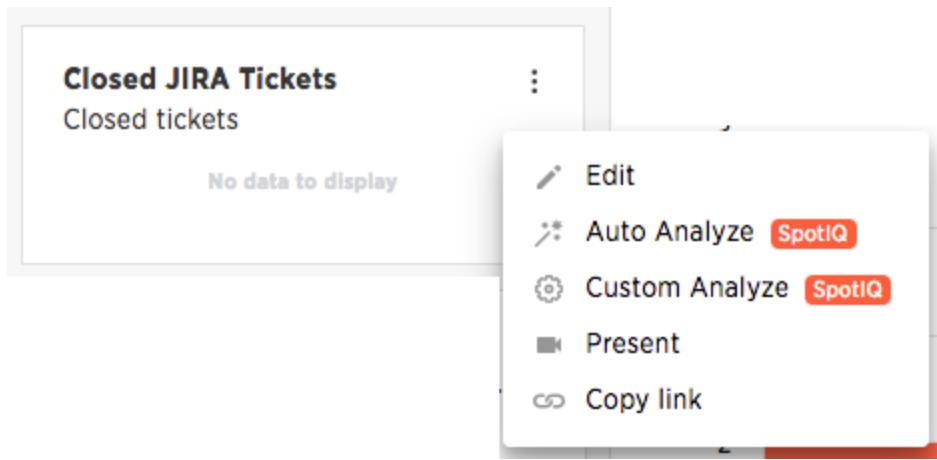
If you can't see this option, ask your ThoughtSpot Administrator to give you the **Has Spot IQ** privilege.

## Where to find SpotIQ actions

The **SpotIQ** page shows you all the analysis results in the system you have access to, these results are called SpotIQ *insights*. You can create your own results from several different locations in your system:

- search results such as query, saved answer, or visualization.
- tables or worksheets
- data uploads
- SpotIQ insights

The menu items that use SpotIQ have a lozenge that indicates this:



## How do you use SpotIQ?

The best way to learn how to use SpotIQ is to try it for yourself. The [SpotIQ 101 tutorial](#) walks you through a simple example.

# SpotIQ 101: Load and analyze data

This 101 walks you through a few simple, SpotIQ workflows. When you have completed the walk through you will be able to use the core features of SpotIQ in your ThoughtSpot installation.

If you want a detailed overview of what SpotIQ does before you try it out, first [read the introduction to SpotIQ](#).

## Prerequisites

Before you begin, make sure you can login into ThoughtSpot application. To complete this tutorial, you need the ability to log into ThoughtSpot and the ability to upload a CSV file. You also need the ability to use the SpotIQ application. These features require that your user account must have the following privileges:

- **Can upload user data**
- **Has Spot IQ privilege**

Your ThoughtSpot administrator can give you these privileges. If you can see the SpotIQ option on your ThoughtSpot dashboard, you have access to SpotIQ:



## Get the sample data and some insights

You can use SpotIQ with any of the data in your system.

This tutorial uses a dataset containing a list of sales and movement data by item and department.

1. Download the [FoodDollarDataReal](#) CSV file.
2. Save or move the file to a place on your local drive..

## Upload the data

1. If you haven't, log into the ThoughtSpot application.
2. Click the **DATA** tab and choose **Actions > Upload Data**.
3. Browse to the sample data file you downloaded or drag the file into the upload area.
4. Choose **YES** for the **Are the column names already defined in the file header** setting.
5. Choose **Comma** for the **Are the files separated by?** setting.

Upload your data

**1 Upload your file**   **2 Set column names**   **3 Set column types**

FoodDollarDataReal.csv

Are the column names already defined in the file header?  Yes  No  
Are the fields separated by?  Comma(,)  Semicolon(;)  Pipe(,)  Space  Tab

Cancel **Next >**

6. Click **Next** for the **Set column names** page.
7. Set the **category\_num** column to **TEXT** on the **Set column types** page.

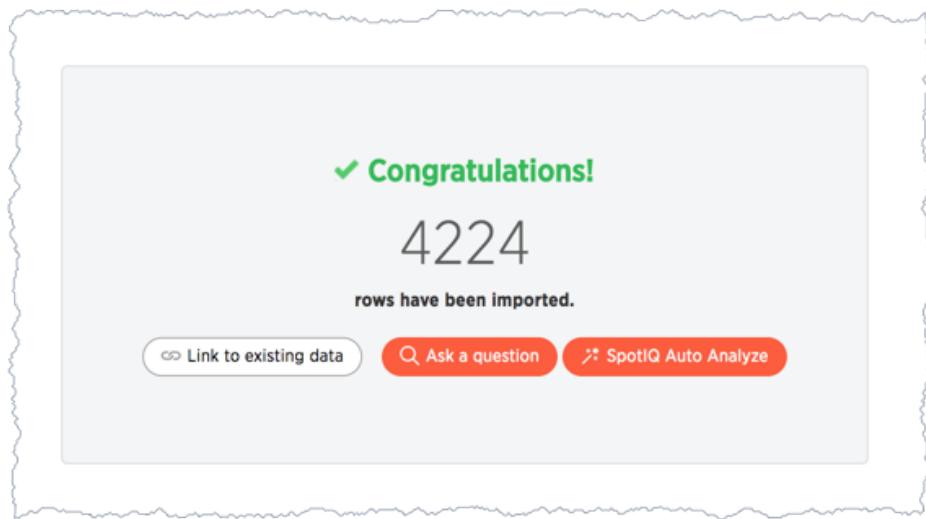
**✓ Upload your file**   **✓ Set column names**   **3 Set column types**

table_num	TABLE_NAME	category_num	year	Units	value_added	imports
1	Food dollar	LARGE INTEGER	DATE	TEXT	DECIMAL	DECIMAL
1	Food dollar	TRUE/FALSE	6/15/05	Cents per Domestic ..	90.9	3.5
1	Food dollar	INTEGER	6/15/05	Cents per Domestic ..	3.1	0.4
1	Food dollar	LARGE INTEGER	6/15/05	Cents per Domestic ..	10	0.8
1	Food dollar	DECIMAL	6/15/05	Cents per Domestic ..	18.4	0.8
1	Food dollar	TEXT	6/15/05	Cents per Domestic ..	3.3	0.6
1	Food dollar	DATE	6/15/05	Cents per Domestic ..	3.6	0.2
1	Food dollar	DATE TIME	6/15/05	Cents per Domestic ..	11.5	-0.1
1	Food dollar	7	6/15/05	Cents per Domestic ..	14.6	0.2
1	Food dollar	8	6/15/05	Cents per Domestic ..		

< Back   Cancel **Upload**

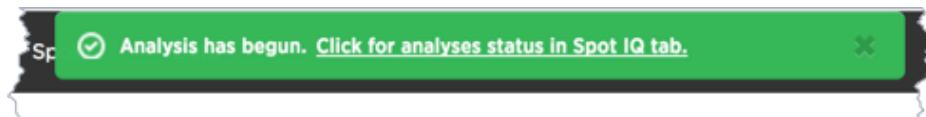
8. Click **Upload**.

The system presents you with a few choices.



- Choose **SpotIQ Auto Analyze** to build SpotIQ insights.

Building insights can take time. How long depends on the data you are analyzing. The ThoughtSpot application displays an informational message. The message disappears after a moment.



## Work with the INSIGHTS list

Each time SpotIQ does an analysis, it generates a set of results. ThoughtSpots keeps the results until the user that requested the analysis (or an admin), deletes them. You can run SpotIQ on the same object multiple times. Each analysis generates new results.

- To check for the results of your analysis, select the **SpotIQ** page.

The **SpotIQ** page allows you to see all results with data you have permission for. So the results lists shows **All** results or just **Yours**.

- Select **Yours**.

NAME	DESCRIPTION	STICKERS	MODIFIED
FoodDollarDataReal	Table analysis, done at 17 October 2017 13:41	MA	13 minutes ago
NominalData	Table analysis, done at 17 October 2017 13:26	MA	28 minutes ago
Warehouse_and_Retail_Sales	Table analysis, done at 16 October 2017 18:29	MA	3 hours ago

- Look for results from your **FoodDollarDataReal** data.

SpotIQ labels each result with a **NAME**, **DESCRIPTION**, **STICKERS**, and **MODIFIED**. The **NAME** comes from the object that was analyzed which is referenced again in the **DESCRIPTION**. The combination of **NAME**, **DESCRIPTION**, and **MODIFIED** is unique.

4. Take a minute and review the **DESCRIPTION** and **MODIFIED** time.

Not every SpotIQ analysis creates results. You can see information about each analysis as well as the results.

5. Select the **analyzes** link at the top of the page.

Check the **STATUS** and also how long the **RESULT** took to generate. Since you got results, you can see that the analysis succeeded.

6. Select the **Analysis for FoodDollarDataReal** and choose **Delete**.

The information disappears from the **analyzes** list.

7. Choose **Results**.

You should still see the results for your **FoodDollarDataReal** run. Deleting information about an analysis run does not delete the actual results. You must delete each individually.

## Where to go next

At this point, you've created a set of insights using SpotIQ, in the next section you [Review insights](#) you created.

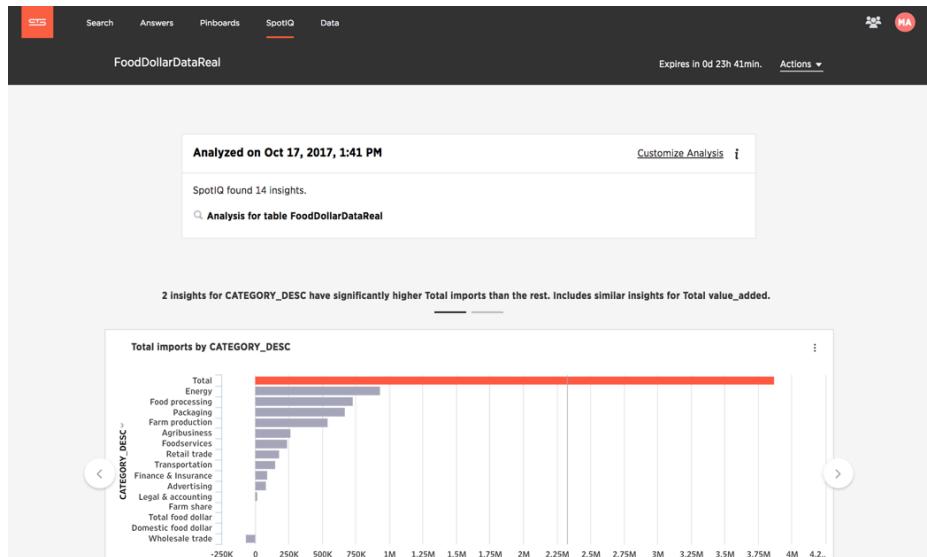
# SpotIQ 101: Do more with SpotIQ

At this point, you have done [the first set of SpotIQ tutorial tasks](#), you've uploaded some data and run your first SpotIQ analysis. You should be able to look at the **SpotIQ** list and see when your analysis ran. This is important knowledge because data changes, you'll want to run multiple analyzes on the same data.

In this section, you look at the results and see what insights SpotIQ discovered. You'll learn about the basic types of analysis SpotIQ runs on data.

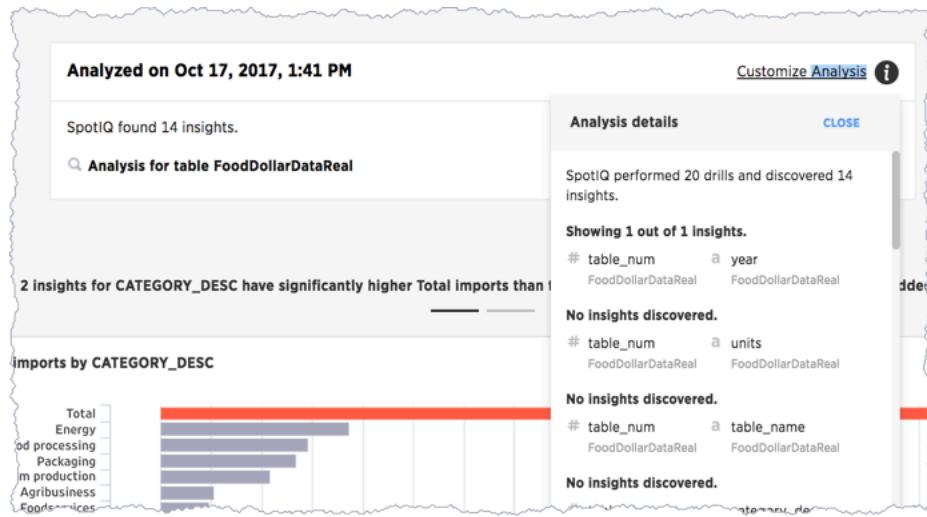
## View results of an analysis

1. If you haven't, log into the ThoughtSpot application.
2. Select the **SpotIQ > Results** page.
3. Look for results from your **FoodDollarDataReal** analysis and click on the **NAME**. The application opens the SpotIQ insights.



The first panel provides information about the analysis.

4. Click on the small **i** to see a summary of the SpotIQ analysis.



You can tell that SpotIQ ran through 20 combinations and discovered 14 insights. The first looked for an insight in a combination of the `table_num` and `year` columns in the `FoodDollarDataReal` data and discovered one insight. The next to column combination `table_num` and `units` did not reveal any insights.

5. Close the details panel and look at the corresponding insight SpotIQ produced.

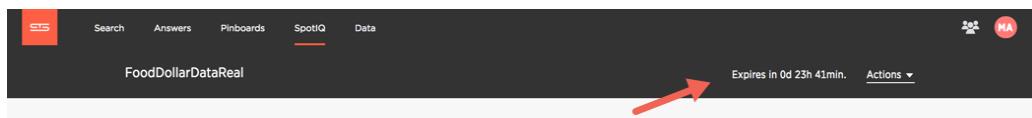
You can see that SpotIQ found three sets of insights:

- 2 insights for `CATEGORY_DESC` have significantly higher Total imports than the rest. Includes similar insights for `Total value_added`.
- Insights from Cross Correlation Analysis.
- Insights from Trend Analysis.

There is a reason there are three sets. You'll learn the answer to that later.

## Time and your insight data

You'll notice that when you first look at an insight there is an expiration date in the upper right corner:

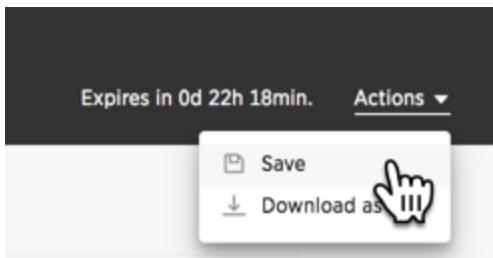


If the expiration time elapses, ThoughtSpot automatically removes the reports.

Insights also take space in the ThoughtSpot installation. A lot of people creating insights can add up to a lot of data lying around. Just as water rising in a bath tub can overflow or just cause a problem you have to wade through, too much data can create a lot of old reports to sift through.

Another reason to expire an insight is time. Typically, business data is changing every day if not more frequently. Any particular set of insights are valid for data during a specific period of time. This doesn't mean old insights can't provide information you can use, just that you are unlikely to get the same insight twice.

If a set of insights look good, you can **Save** them and they are not removed when they expire. Do this now, click **Actions > Save** from the menu:



Now your insights won't expire as you work with them!

## Data outliers

SpotIQ attempts to look for three central insights. The first insight it looks for are data outliers. An outlier is a value that is “far away” or that differs from the other data.

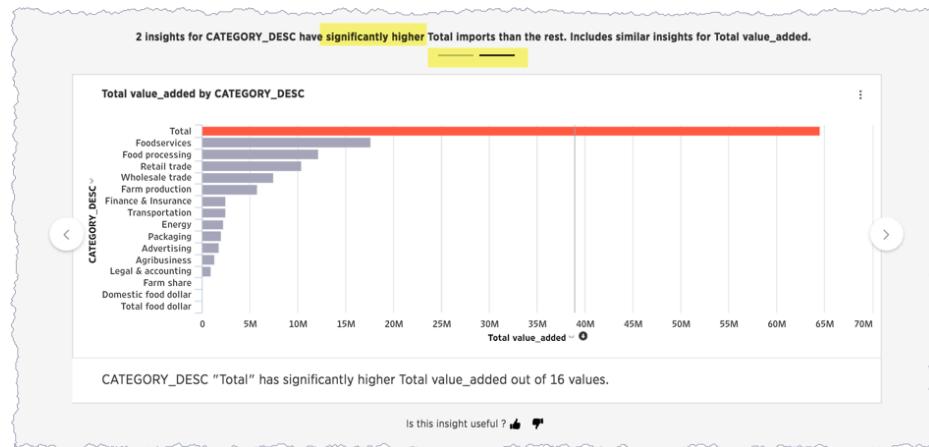
Outliers can result from measurement or recording errors or they could contain valuable information. For example, one store might sell significantly more towels in one week out of the year because there was a flood that week.

1. Scroll to the first carousel of data that SpotIQ created for you.

A carousel groups a set of visualizations on a page. You can use the arrows attached to a carousel to “page” through its contents.

2. Count the number of items in the carousel.

You should see that SpotIQ found two outliers.



In this case, SpotIQ is indicating that the **Total Imports** value is significantly higher than the other values in your set.

Two facts about this outlier shouldn't surprise, you. The first fact is that a total exceeds all the other items in the data. That just makes sense so this isn't a true outlier.

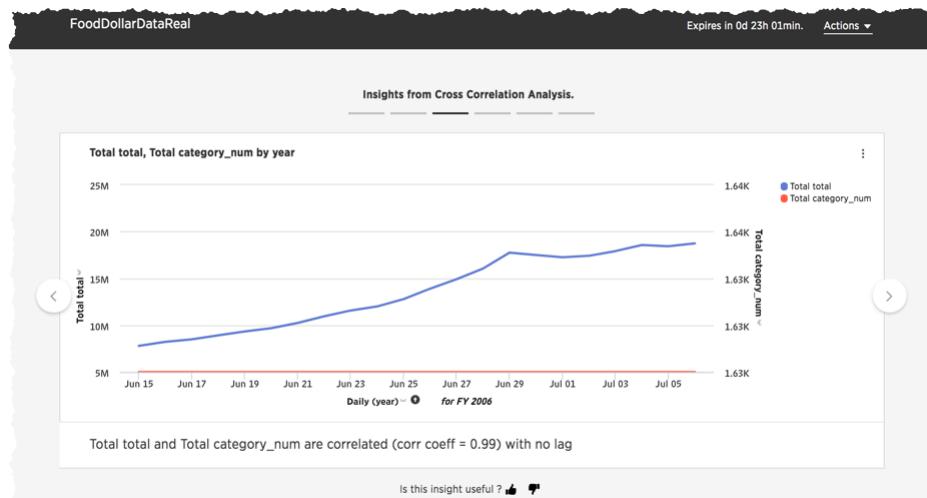
The second fact is that SpotIQ insights are not always interesting. In this case the data included totals in the row data which caused nonsense outliers. You can use SpotIQ to fix this issue and go further. More about this later in the tutorial.

## Looking for relationships (cross-correlation analysis)

The second category of insights SpotIQ attempts to find is cross-correlations. Cross-correlation looks for and measures relationships between two quantitative, continuous information sets – in our case two fields (columns) in our dataset. Usually, you'll see a cross-correlation in your insights if your data includes time or date data.

The result of a correlation analysis shows how strong and in what direction a relationship between two data sets moves. The range is -1 to 1, the closer the cross-correlation value is to 1, the more closely the information fields (data elements) are correlated.

1. Scroll down your insight page, till you find the correlation results:
2. Page through the correlation category to the third insight:



You can see that the overall total and the total category number values correlate very strongly. The correlation also looks for a *lag* value.

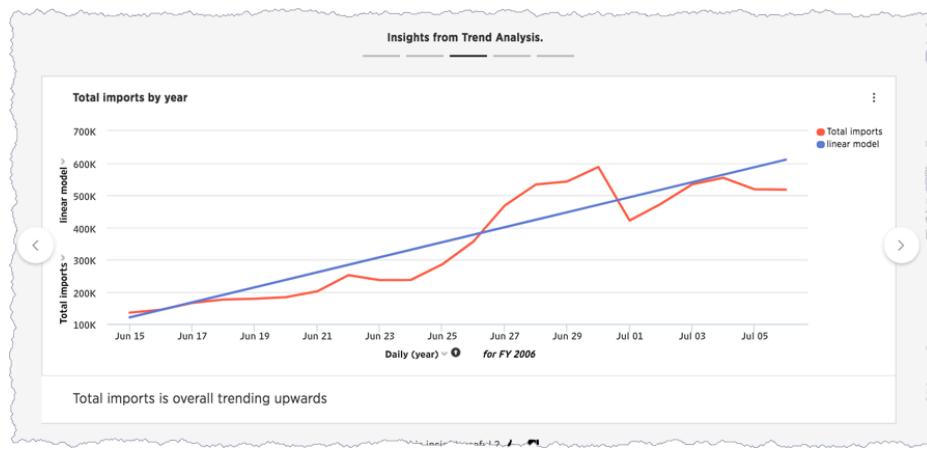
*In the business world the dependence of a variable Y (the dependent variable) on another variable X (the explanatory variable) is rarely instantaneous. Often, Y responds to X after a certain lapse of time. Such a lapse of time is called a lag.* [Good Data Help](#).

## Trends over time

The third set of insights SpotIQ looks for is a trendline. A trendline is a straight or curved line that indicates the general pattern or direction of a time series data (information in sequence over time). As you may have guessed, trendlines rely on your data having a date or time column.

You can use a trendline to determine the general direction of a trend in your data. Are sales are climbing or are customers are declining over time?

1. Scroll down to the **Insights from Trend Analysis** section.
2. Look for the **Total imports by year** result.



At this point, you may have noticed that SpotIQ grouped each type of insights. Outliers, correlations, and this last one, trendlines. This grouping makes it easier to locate and review similar types of insights.

## Where to go next

In this section, you explored some of the functionality of the **SpotIQ** page. You learned that SpotIQ combines columns in your data to look for three categories of insights:

- outliers
- cross-correlations
- trendlines

In [the next section](#), you'll learn how to schedule SpotIQ to deliver insights on a regular basis.

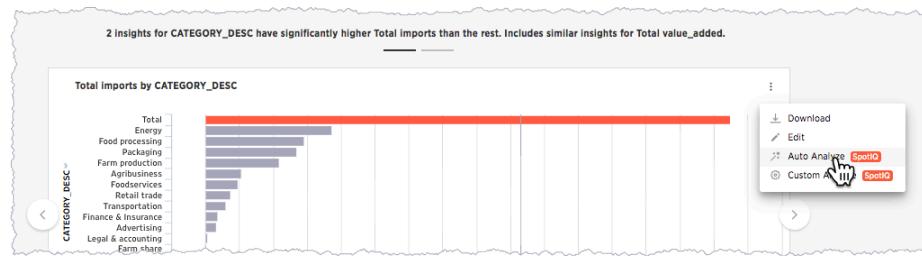
# SpotIQ 101: Review insights

You've finished the [second part of the tutorial](#). In this, the third and final part, you'll learn how you can modify a SpotIQ analysis to extend or customize the analysis. You'll edit an insight to customize the chart it uses or modify the **Search** syntax behind it.

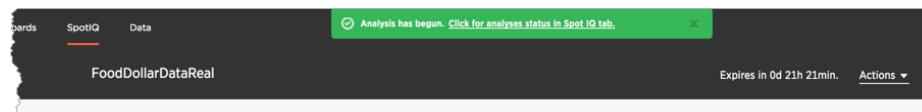
## Customize an analysis

In the second part of the tutorial, you reviewed the first output from SpotIQ. The outlier insights weren't really that useful. Let's try to dig deeper into that.

1. Scroll down to the outlier section.
2. Select the first insight.
3. Choose the menu and select **Auto Analyze**.

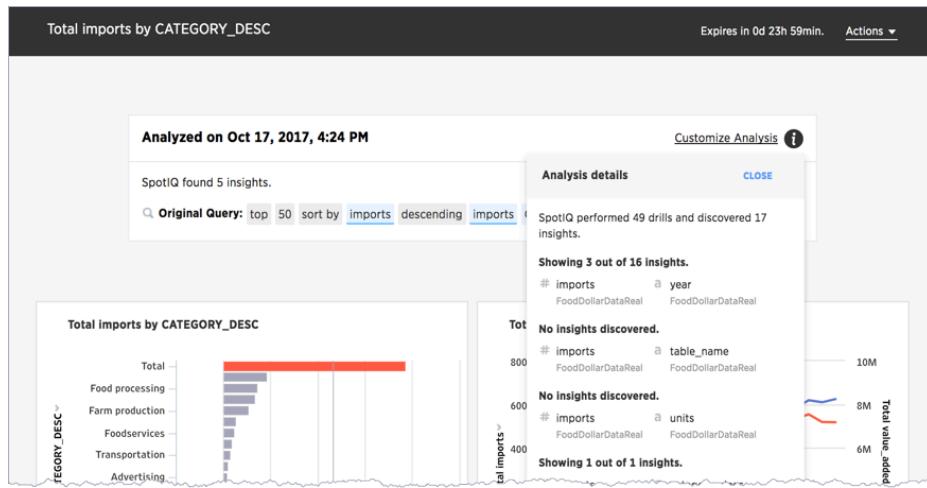


The system responds by starting a new insight run based on the single chart. Each time you request a change to the analysis settings, it starts another run.



4. Return to the SpotIQ page and select your new analysis.

5. Open the analysis and review the analysis details.



If you recall, the first run against this data SpotIQ ran through 20 combinations and discovered 14 insights. In this second run against one table, SpotIQ ran 49 combinations and developed 17 insights. Only 4 insights were discovered.

Running SpotIQ again does not necessarily result in fewer runs or even, as in this case a reduced number of insights. It changes the analysis though, by selecting the single insight from your first run, you gave SpotIQ information it could use to focus the next run.

Continue with the customization.

#### 6. Choose **Customize Analysis**.

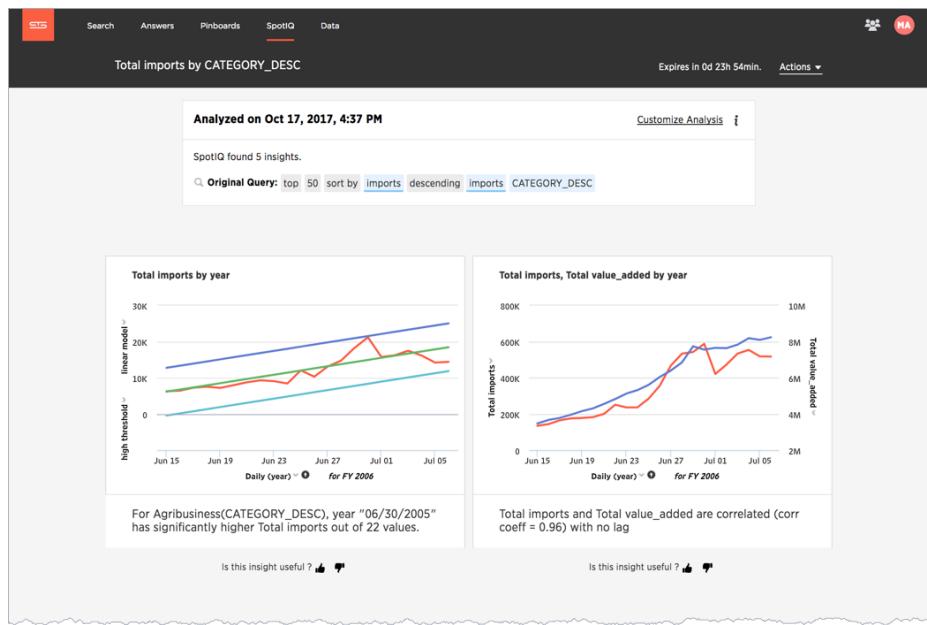
The **Customize Analysis** dialog appears.

7. Select the **Customize algorithms** tab.
8. Uncheck the **Outlier Detection using Z-Score** option.
9. Click **ANALYZE**.

SpotIQ starts a new analysis.

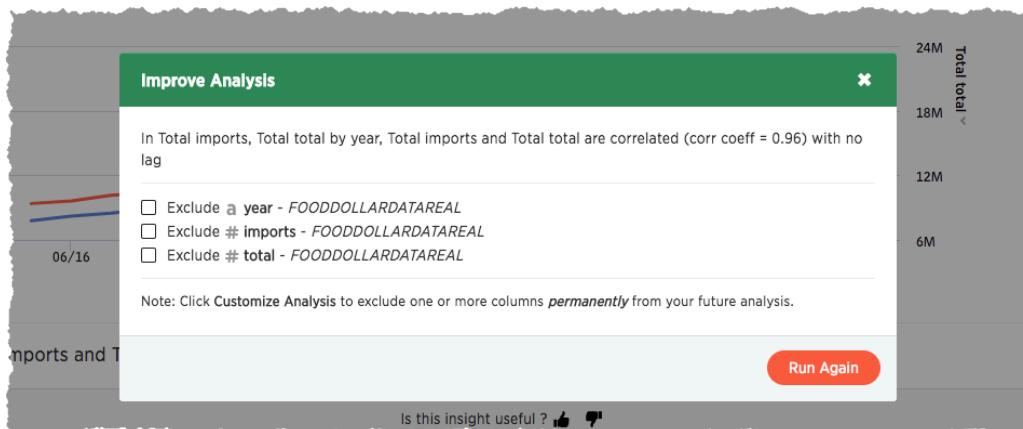
10. Open the new analysis from the SpotIQ page.

You'll find that the new analysis has the same name as the last one you ran. The new results include an outlier and a cross-correlation. You'll discover with these new charts a definite outlier **6/30/2005**.



## A temporary customization for a single insight

Finally, you might have noticed the thumbs at the bottom of each insight. Use the thumbs up to tell SpotIQ you like the data insight. Clicking thumb's down display an **Improve Analysis** dialog:



You can exclude a value from this individual analysis and rerun the analysis. Unlike the **Customize Analysis** action, this action works only on this analysis and does not persist.

Try this for yourself later at some point.

## Modify an insight search and visualization

The lines used in the cross-correlation are somewhat ambiguous. A different visualization may help in this case. You can also modify the charts you see in your results. Try this now.

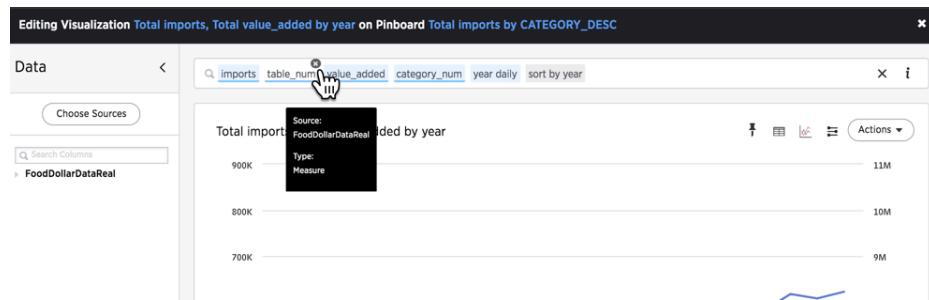
1. Select the menu on the **Total imports, Total value\_added by year** chart.
2. Click on the chart menu and choose **Edit**.



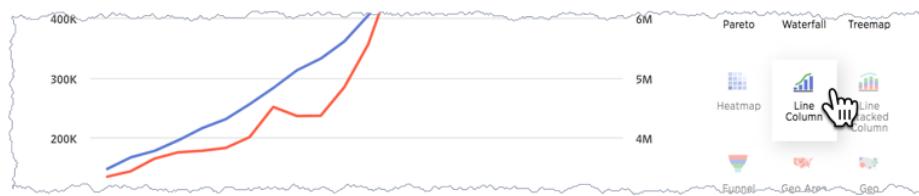
The **Edit Visualization** appears. The layout is familiar to any user that has run a ThoughtSpot search or worked with a result. This dialog reveals the **Search** syntax used to generate the insight. You can use this dialog to do all the things you would normally do with a search.

3. Edit the search bar and remove `table_num` from the search.

The `table_num` isn't strictly necessary in the data.

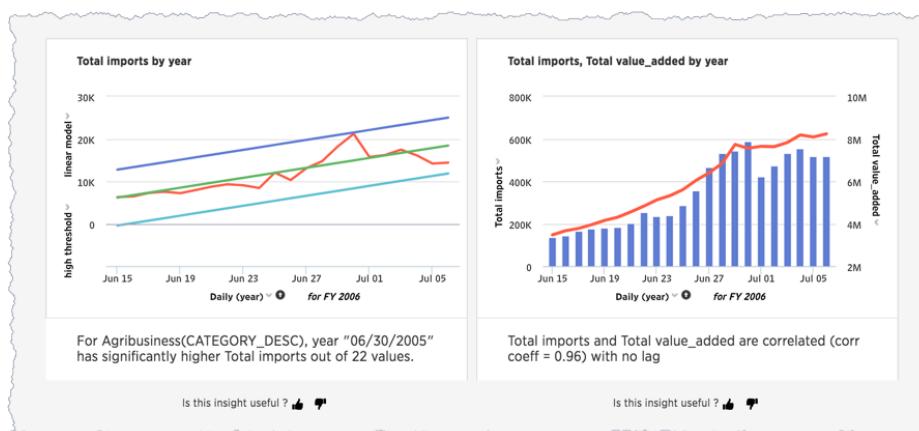


4. Select the chart icon and choose **Line Column** to change the visualization.



5. Choose **Actions > Update**.
6. Close the dialog by pressing the **Close**.

The chart with your changes now appears in the **SpotIQ** list. The outlier stands out very clearly now in both charts.



## Present your insights

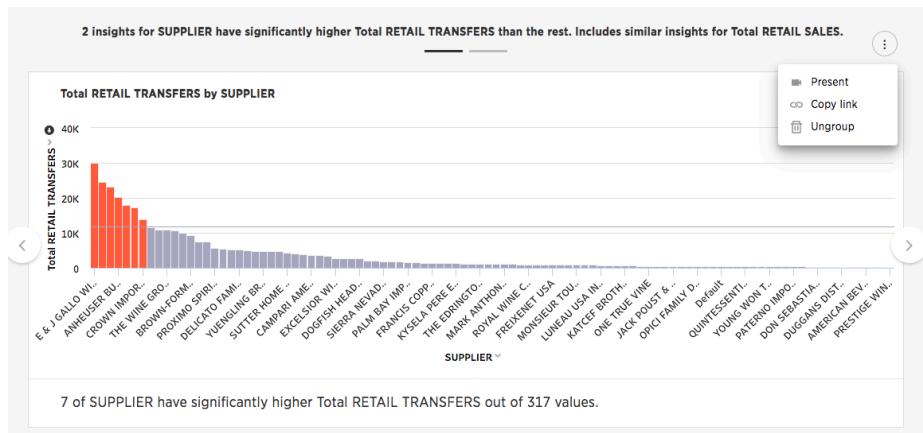
You can present SpotIQ insights in several ways. You can present an individual result or all of them at once. You can package all the insights in a PDF. This is useful for passing a packet of results. Since any insight reflects the data at a specific point in time, always consider saving the insight as a PDF.

1. Locate the **Actions** menu on the corner of the **SpotIQ** page.
2. Choose **Actions > Download as PDF** from the menu.

The system downloads a file containing all the insights. Keep in mind this file can be very large if there are a lot of insights found in your data.

You can use the **Present** action to view visualizations that appear in a carousel, and present them as a group.

1. Scroll down to the first carousel with your new charts.
2. Roll over the right corner of the menu to reveal the menu.



3. Choose **Present** from the menu.

Click through the carousel with the next / previous arrow buttons, or use the left side menu of chart names.

Also, each individual chart has a **Download**, **Present**, or **Copy link** action you can take.

## Schedule a recurring analysis

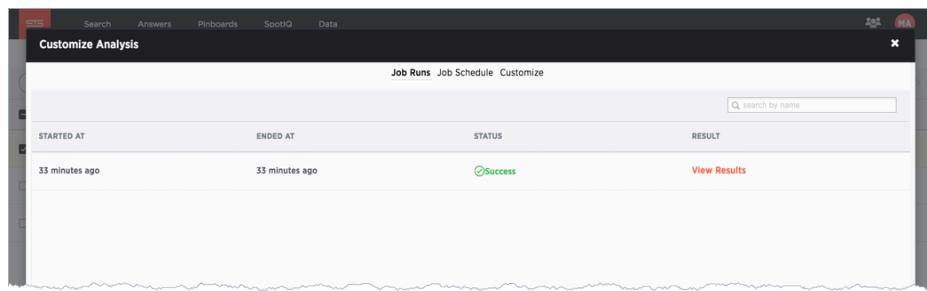
If you find an analysis configuration that is useful. You can run that configuration periodically. This is a good thing to do of course as, in a business, data changes all the time.

1. Go to the **SpotIQ** page.
2. Choose the **Analysis** tab.

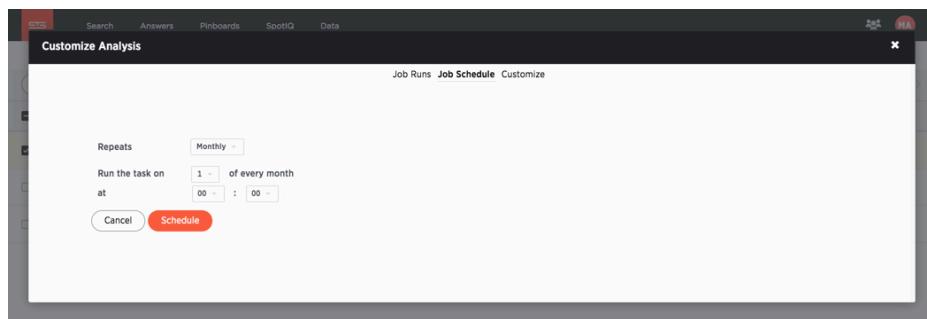
	NAME	STATUS	MODIFIED	RESULT
<input checked="" type="checkbox"/>	Analysis for Total imports by CATEGORY_DESC	✓ Done	32 minutes ago	Last run succeeded in 10.00 seconds. <a href="#">View Results</a>
<input type="checkbox"/>	Analysis for Total imports by CATEGORY_DESC	✓ Done	45 minutes ago	Last run succeeded in 5.00 seconds. <a href="#">View Results</a>
<input type="checkbox"/>	Analysis for table FoodDollarDataReal	✓ Done	45 minutes ago	Last run succeeded in 10.00 seconds. <a href="#">View Results</a>

3. Click on the latest **Total imports by CATEGORY\_DESC** analysis.

The **Customize Analysis** dialog appears.



4. Choose the **Job Schedule** option.
5. Fill out the schedule to run the job **Monthly**.



Every month SpotIQ will update your analysis with new data.

Keep in mind that SpotIQ jobs take resources in your system. So, you really don't want to leave casual jobs running in the system. Go ahead and delete this latest job. Now that you know how to schedule a job, it isn't needed anymore.

1. Click the checkbox for the job you just scheduled.
2. Click **Delete** to remove it.

NAME	STATUS	MODIFIED	RESULT
<input checked="" type="checkbox"/> Analysis for Total imports by CA...	<span>▷ Scheduled</span>	① 3 minutes ago	Last run succeeded in 7.00 seconds. <a href="#">View Results</a>
<input type="checkbox"/> Analysis for table FoodDollarDat...	<span>✓ Done</span>	① 30 minutes ago	Last run succeeded in 6.00 seconds. <a href="#">View Results</a>
<input type="checkbox"/> Analysis for Total imports by CA...	<span>✓ Done</span>	① 39 minutes ago	Last run succeeded in 10.00 seconds. <a href="#">View Results</a>

## Congratulations!

Congratulations, at this point you've learned the basics of SpotIQ with ThoughtSpot. Throughout the application, you'll find the SpotIQ icon on search answers, pinboards, visualizations, and data. Wherever you see this option, you can click to run suggested or custom analyses, and get new insights on the SpotIQ page.

# Best SpotIQ Practices

The SpotIQ feature works without you having to do anything but pushing a button. However, like any other feature, there things you can do to optimize the feature. This page contains some best practices you can use to make SpotIQ more effective when you use it.

## When to invoke SpotIQ

If you followed the tutorial, you know that right after uploading data is a good time to run **Auto Analyze**. SpotIQ can very quickly help you find insights in your data.

Start from a **Search**. Enter a single measure in the bar; one you want to explore of course! Then, choose **Actions > Auto Analyze** on the results. Choosing the single measure focuses SpotIQ.

Use **Custom Analyze** to focus or tweak the SpotIQ results. While you are tempted to keep all the columns, eliminating some can also result in a better analysis.

## Do your data modeling

You can increase the SpotIQ's effectiveness by ensuring you are practicing good data modeling. This is true if you are user uploading the occasional data file or a data management professional. Modeling data requires that you can:

1. Click on the **DATA** icon, to get to the data management listing.
2. Click on a data source you own or can edit. This brings up the **Columns** screen, where you'll make your modeling settings.
3. Modify one or more column settings.
4. Save your changes.

If you worked through the SpotIQ tutorial, you can try experiment on the **FoodDollarDataReal** data you uploaded.

Make sure you set the **INDEX PRIORITY** for columns in your data source. Use a value between 8-10 for important columns to improve their search ranking. Use 1-3 for low priority columns. **INDEX PRIORITY** impacts user-based ranking which helps SpotIQ focus its analysis.

SpotIQ uses measures for correlations. For trendlines and outliers, if SpotIQ has a measure, it then drills by attributes in turn.

**ATTRIBUTE** = text or dates that you can't sum  
**MEASURE** = values you can do math on, with a meaningful result

**Attributes**

- \* Fruit
- \* Grocery
- \* Macintosh

**Measures**

- \* Price
- \* Age
- \* Weight

**What about?**  
A style number or product ID.



You should also set **AGGREGATION** on your columns. SpotIQ applies the default aggregations from your data when it pulls measures for analysis.

## Situations to avoid

Like any AI, some situations SpotIQ is not yet equipped to handle. You should know what these are so you can avoid them. If your data contains a measure that uses a `MOVING_*` or `GROUP_*` formula, SpotIQ may return results that simply aren't meaningful. When doing a correlation analysis, SpotIQ may not find meaningful data if you have a measure with anything other than `SUM`.

## Set SpotIQ preferences

You can set preferences for SpotIQ in your user profile. These preferences allow you to control how you receive analysis notifications. They also allow you to exclude nulls or zero value measures from analysis.

[Update my SpotIQ preferences](#)

- Email me analysis notification on success
- Email me analysis notification on failure
- Email me analysis pinboard as attachment
- Exclude null values from my analysis
- Exclude zero measure values from my analysis

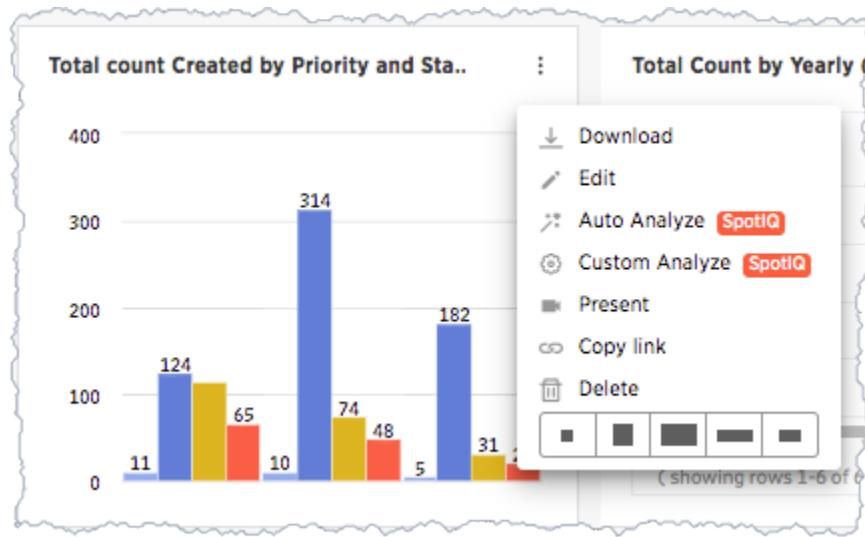
The exclusions impact each SpotIQ analysis. It eliminates points with such values during statistical calculations for example, for mean, standard deviation SpotIQ excludes values from any equation and uses only the remaining points.

# Custom SpotIQ analysis

Everywhere in ThoughtSpot where you can **Auto Analyze** with ThoughtSpot, you can also configure a customized SpotIQ analysis. The customized analysis can be a one-time customization or can apply to all future analysis of that particular data source.

## Finding customization controls

Every menu or location where you can **Auto Analyze** with ThoughtSpot, you can also choose **Custom Analyze**. It looks like this:



Regardless of where you are in the ThoughtSpot UI, the SpotIQ **Customize Analysis** dialog has the same layout. The columns, of course, are specific to the data you launched the dialog from.

---

This dialog has three tabs, **Included columns**, **Excluded columns**, and **Customize Algorithms**. The **Customize Algorithms** is by far the most advanced panel.

## Customize Analysis

Included columns Excluded columns Customize algorithms

**Select Algorithms**

Outlier Detection using Z-Scores  
 Outlier Detection using Linear Regression  
 Trend Analysis  
 Cross-Correlation Analysis

**Refine Parameters**

Minimum rows required for analysis

Multiplier for Outlier Detection

Maximum P-Value

Minimum Correlation Coefficient

Maximum Correlation Lag

Restrict analysis to current resultset only  
 Exclude null values from current analysis

Insight Count Settings

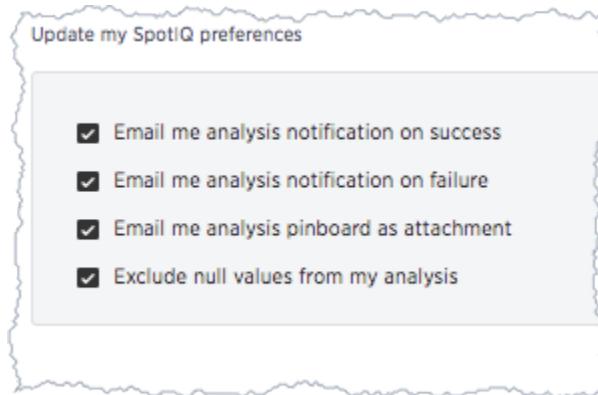
**Trigger Analysis**

The **SpotIQ** insights page has an option for this dialog and also an option to customize a single analysis.

The screenshot shows the ThoughtSpot interface for a 'FoodD' dataset. On the left, there's a sidebar with a tree view under 'Total' and a chart showing 'Food processing' as the top category. The main area has a 'Customize Analysis' dialog open, listing columns like 'CATEGORY\_DESC', 'TABLE\_NAME', 'Units', etc., with checkboxes for inclusion. A blue arrow points from the 'Customize Analysis' button in the dialog to the 'Actions' dropdown in the top right. Another blue arrow points from the 'Run Again' button in the 'Improve Analysis' modal back to the 'Customize Analysis' dialog.

The **Improve Analysis** dialog always excludes a value from all future analysis.

Finally, you can also set some analysis options from your **Profile**.



You'll notice only one of these options is actually related to analysis. The others relate to how you receive notifications about SpotIQ operations.

## Eliminate null values

Null values should be eliminated from your analysis.

**From next analysis**

**From future analyses**

In the Search bar, add a filter `column!= Null` to the search bar. Or choose **Exclude null values from current analysis**, using the **Customize algorithms** panel.

Set **Exclude null values from my analysis** on your user **Profile** to exclude them from any future SpotIQ analyses you do.

## Exclude uninteresting column(s) in insight

You should always exclude columns you are not interested from your analysis.

From next analysis	From future analyses
Exclude a column using <b>Improve Analysis</b> on a visualization. Or set the value through the <b>Excluded columns</b> tab of the <b>Customize Analysis</b> panel.	Choose <b>Also exclude from all my future analysis</b> on the <b>Excluded columns</b> tab of the <b>Customize Analysis</b> panel. Alternatively, exclude using the <b>Improve Analysis</b> dialog on an insight that uses the column you want to exclude.

## Include an interesting column

You can always includes columns that interest you in your analysis.

From next analysis	From future analyses
Choose <b>Customize Analysis</b> and select to <b>Include columns</b> that you want to appear.	Ensure **Index Priority** is between 8-10 on the column through data modeling.

## Remove known date outliers

Your data may contain known outliers. For example, you are in the middle of a quarter and only want to analyze the previous quarter. Anything from the present quarter could contain an outlier.

From next analysis	From future analyses
In the Search bar, add a filter <code>date&lt; last time period</code> to the search bar.	Not applicable.

## Too few insights

Your SpotIQ analysis may not provide you as many insights as you think it should.

From next analysis	From future analyses
Choose <b>Customize Analysis</b> and <b>Customize algorithms</b> . Decrease the <b>Multiplier for Outlier Detection</b> to a value closer to zero.	Not applicable.

# Advanced R Customizations

Admins and users with the **Has SpotIQ** privilege can run an R script using the **Custom Analysis** feature of SpotIQ. This means you can run an R script from any point in ThoughtSpot where you find the **Custom Analysis** menu option.

The R language is an open source programming language used for statistical computing. As such, knowledgeable users can use R to perform sophisticated analysis in a ThoughtSpot environment.

This section explains the feature and how to use it. It is not meant as an R primer. To learn more about R and how to use it, visit the [R Project for Statistical Computing](#).

## Understand R script requirements in ThoughtSpot

ThoughtSpot provides R running as a service within a ThoughtSpot cluster. Permissions are restricted. This means the R script does not have permission to issue system commands.

The ThoughtSpot cluster has pre-installed the basic R packages. If your script requires a specific package, you must request your ThoughtSpot cluster admin to install the package on your behalf.

ThoughtSpot internally transforms and binds an R script prior to sending it to the cluster's R service. The system expects each script have a well-defined structure which is the following:

```
#####R SCRIPT#####
<Fill script body>
#####COLUMN BINDINGS (ONE PER LINE)#####
<Fill column bindings here>
```

The scripts contains the column bindings with the answer results appearing as parameters in the R script. ThoughtSpot expects for each `.param n` in R your script must provide a corresponding binding. The following pseudo code illustrates an R script in a form suitable for ThoughtSpot:

```
#####R SCRIPT#####
df <- data.frame(.param0,.param1, ...);
...
write.csv(..., file=#output_csv#, ...);
```

Notice that `.param0` refers to first column in column binding and `.param1` refers to the second. Should you need a third binding, you would use `.param2` and so forth.

The output of the script is either PNG or CSV. This example script uses `#output_csv#` to emit data in a CSV (tabular) format. Use `#output_png#` to emit data in PNG format.

Presently, error reporting is limited for R scripts in SpotIQ. You should validate your R script independent of your ThoughtSpot environment. Once you are sure they are free of syntax or other errors, then try the script in ThoughtSpot.

# Try a Custom Analysis with R

The following illustrates how to run an R analysis on data that has a sales column and a zip code column.

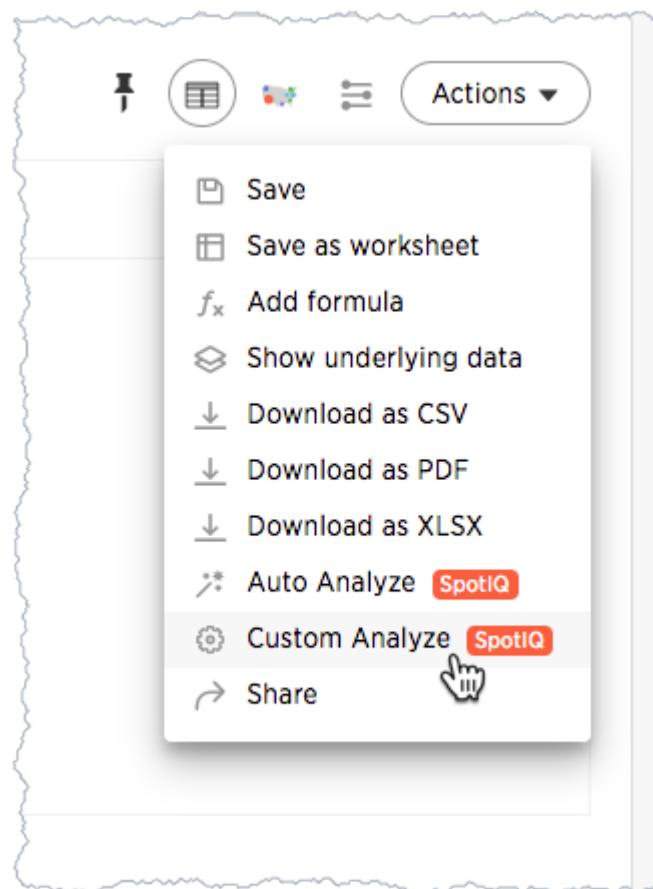
1. Log into ThoughtSpot and go to the **Search** bar.
2. Use **Choose Sources** to locate a source with sales and zip code data. This example uses **Phone Sales** data.
3. Enter `sales zip code` in the search bar.

If your source contains the proper data, you should see something similar to the following:

The screenshot shows the ThoughtSpot interface. At the top, there's a navigation bar with tabs for Search, Answers, Pinboards, SpotIQ, Data, and Admin. The Search tab is active. Below the navigation bar, there's a search bar containing the query "sales zip code". To the left of the search bar, there's a sidebar labeled "Data" with a "Choose Sources" button and a dropdown menu showing "Phone Sales", "% Gross Margin", "Age Group", and "City". The main content area displays a table titled "Total Sales by Zip Code" with two rows of data:

Zip Code	Sales
	TOTAL
91706	267,380.23
70601	789,295.87

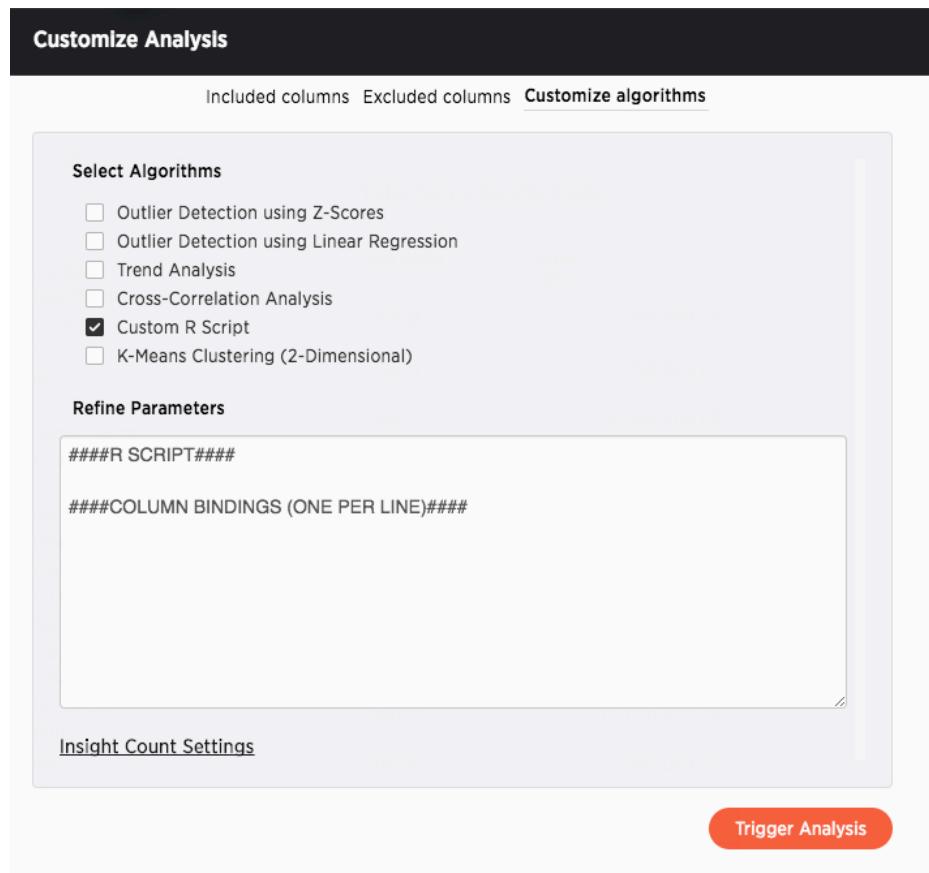
4. Choose **Actions > Custom Analyze**.



ThoughtSpot opens the **Customize Analysis** dialog.

5. Choose the **Customize algorithms** tab.
6. In the **Select Algorithms** section, click the **Custom R Script** box.

Selecting this option unsets all the other options on this tab and displays the **Refine Parameters** field.



7. Enter this sample script in the field.

```
####R SCRIPT#####
library(ggplot2)
set.seed(20)
df <- data.frame(.param0, .param1)
cluster <- kmeans(df[1:2], 3, nstart = 20)
cluster$cluster <- as.factor(cluster$cluster)
png(file=#output_file#,width=400,height=350,res=72)
print(ggplot(df, aes(.param0, .param1, color = cluster)) + geom_point())
####COLUMN BINDINGS (ONE PER LINE)#####
Sales
Zip Code
```

This script binds `.param0` to `Sales` and `.param1` to the `Zip Code` column. You can see from the script that the output should be PNG (`#output_png#`).

8. Check your work.

**Select Algorithms**

- Outlier Detection using Z-Scores
- Outlier Detection using Linear Regression
- Trend Analysis
- Cross-Correlation Analysis
- Custom R Script
- K-Means Clustering (2-Dimensional)

**Refine Parameters**

```
####R SCRIPT#####
library(ggplot2)
set.seed(20)
df <- data.frame(.param0, .param1)
cluster <- kmeans(df[1:2], 3, nstart = 20)
cluster$cluster <- as.factor(cluster$cluster)
png(file=#output_png#,width=400,height=350,res=72)
print(ggplot(df, aes(.param0, .param1, color = cluster$cluster)) + geom_point())
####COLUMN BINDINGS (ONE PER LINE)#####
Sales
Zip Code
```

**Insight Count Settings**

**Trigger Analysis**

9. Click **Trigger Analysis**.

SpotIQ runs your analysis in the background.

10. Go to the SpotIQ page and click on the results of your newly triggered analysis.

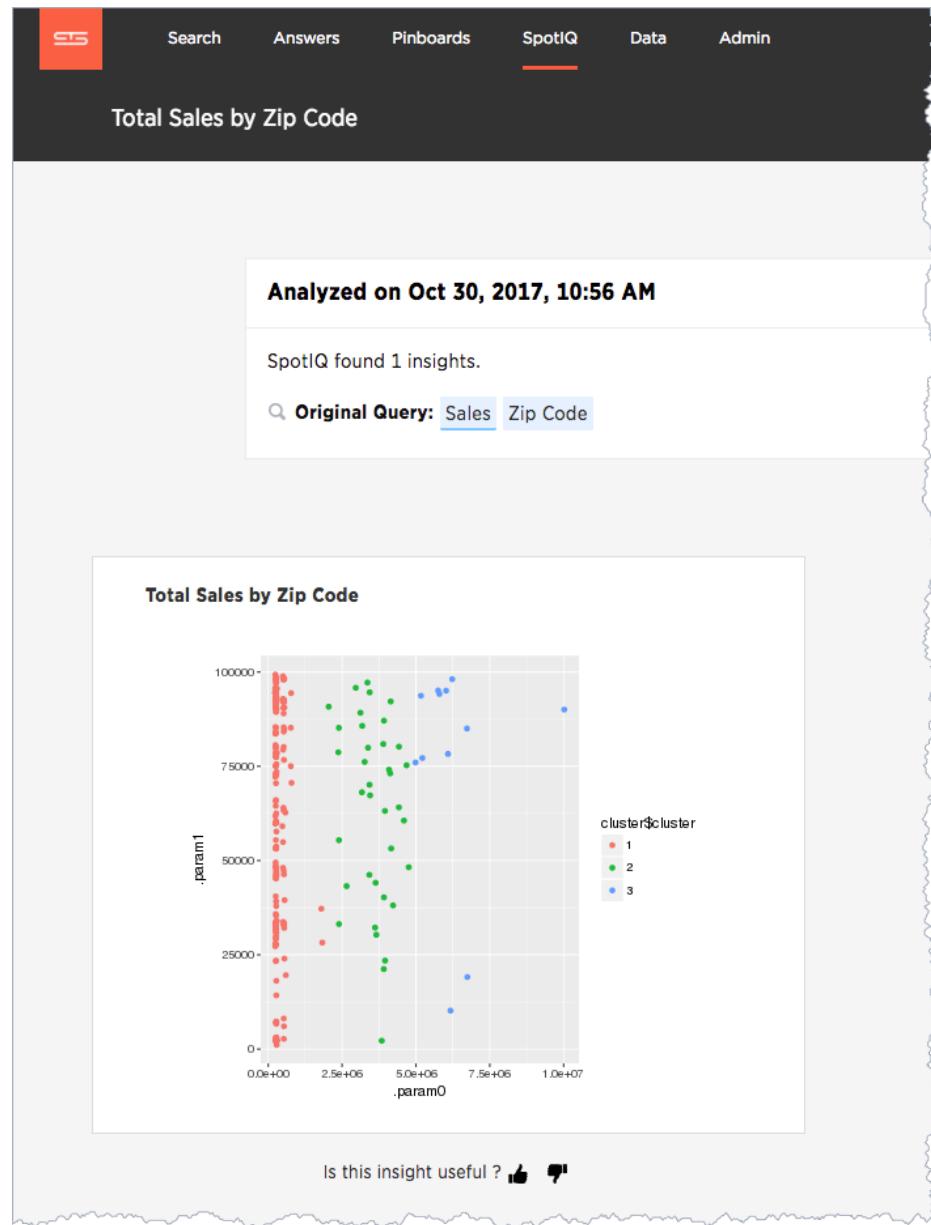
Search Answers Pinboards SpotIQ Data Admin

Results Analyses

All Yours Share Delete Apply Sticker

NAME	DESCRIPTION	STICKERS	MODIFIED
Total Sales by Zip Code	Custom R analysis of Sales, Zip Code don..		2 minutes ago

You should see the results in PNG format similar to the following:



You can run another R script directly on this result to get CSV results. Try this on your own. Here is the script to give you CSV output:

```
#####R SCRIPT#####
set.seed(20);
df <- data.frame(.param0,.param1);
cluster <- kmeans(df[1:2], 3, nstart = 20);
df$Cluster <- as.factor(cluster$cluster);
colnames(df)[1] <- 'Sales';
colnames(df)[2] <- 'Zip Code';
write.csv(df, file=#output_csv#, row.names=FALSE);
#####COLUMN BINDINGS (ONE PER LINE)#####
Sales
Zip Code
```

## Syntax help in the dialog

Use the **i** icon to see help for the R syntax.



# Overview of sharing

Whenever you are working in ThoughtSpot, you are in your own private environment until you share your work with others. Sharing applies to searches, pinboards, and any data you upload.

## What you can share

This is a list of objects a regular, non-administrator user can share. Administrators have more granular control over data security.

You can share with groups and with individual people. You can share several different types of objects:

Object type	Description	Default security model
Uploaded data	Data that was uploaded using a Web browser.	Only the user who uploaded the data (and any user with administrator privileges) has access to it by default. They can share a table (or selected columns) with other people or groups. See <a href="#">Share uploaded data</a>
Pinboards	A pinboard of saved search results.	Anyone who can view a pinboard can share it. See <a href="#">share a pinboard</a>
Answers	The result of a single search.	Anyone who can view an answer can share it. See <a href="#">share answers</a>

## Who can share and to whom

You do not have to be an administrator or the owner, to share saved answers or pinboards. Any user can share them, based on the access levels the user has.

If you upload a spreadsheet, you can share **Can View** or **Can Edit** privileges with other people, who can further share them with others. This last point is important. Like sharing a secret, sharing an object or your data can quickly spread in this way to people you do not intend it to. So, if your data or work is sensitive to your business be thoughtful in how you share it.

Finally, you can revoke access to an object (table, worksheet, or pinboard) that you have previously shared. Unsharing an object is very similar to sharing it.

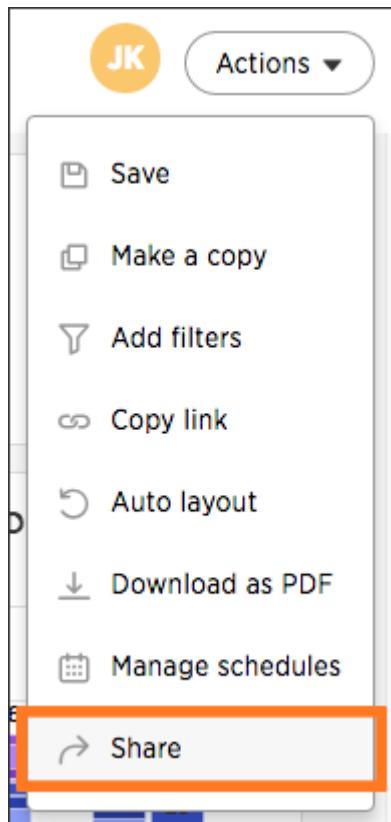
# Share a pinboard

**Summary:** Whenever you view a pinboard you have the option of sharing it with others.

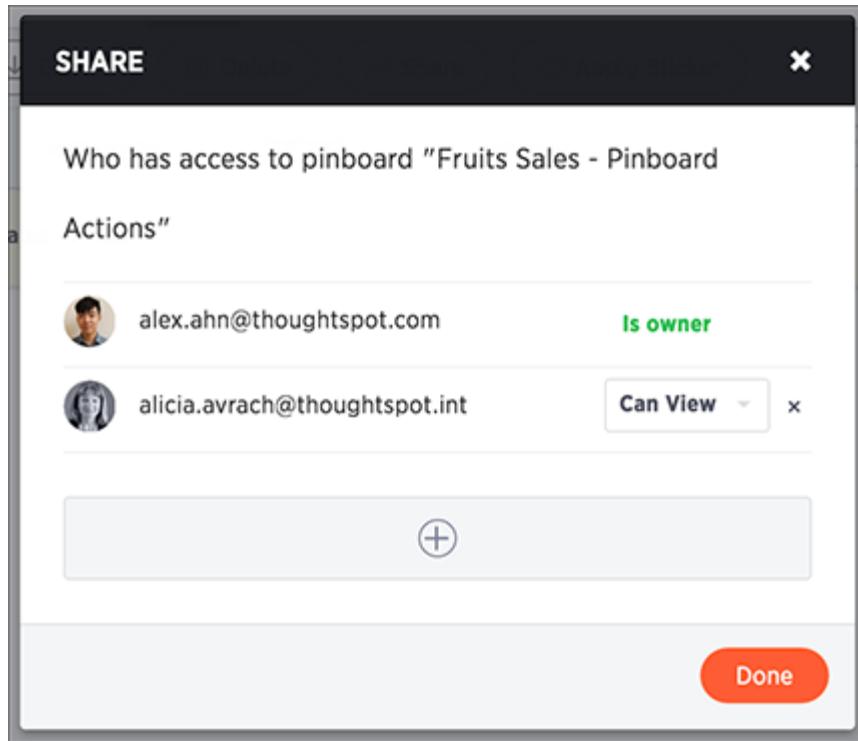
When you share a pinboard what you are really sharing is a live link to the pinboard, when you click **Share with....** So whenever someone else views it, they will see the most recently saved version with the most recent data. You do not have to be an administrator or the owner to share saved pinboards. Any user can share them, based on the access levels the user has.

To share a pinboard:

1. Configure the pinboard to look as you'll want it to appear when shared.
2. Click the **Share** icon.



3. Click **+ Add users or groups** and select users or groups that you want to share with.



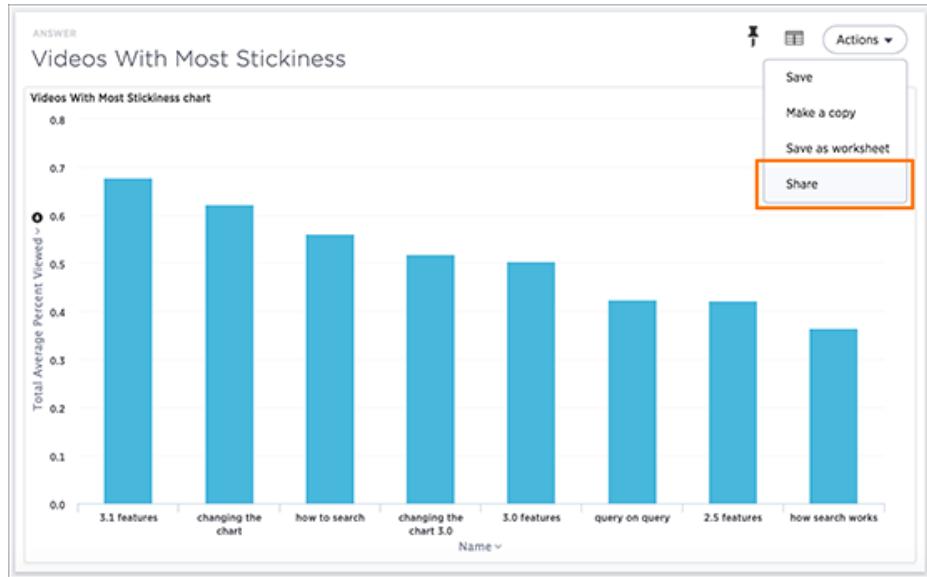
4. Configure the level of access by selecting from the dropdown list. You will only see options available, based on your own access level. For example, if you have only **View** access, you cannot share as **Edit**. You can select:
  - **Can View** to provide read-only access. If the person doesn't have access to the underlying data, they can only view a shared pinboard. If they change anything on the pinboard, their changes are not saved. In order to persist the changes, the user would need to make a copy of the modified pinboard.
  - **Can Edit** to allow modification. Enables renaming or deleting the shared pinboard. If a person with edit privileges modifies a shared pinboard, their changes will be saved to it.
5. Click **Add and Save**.
6. Click **Add Permissions**.

# Share answers

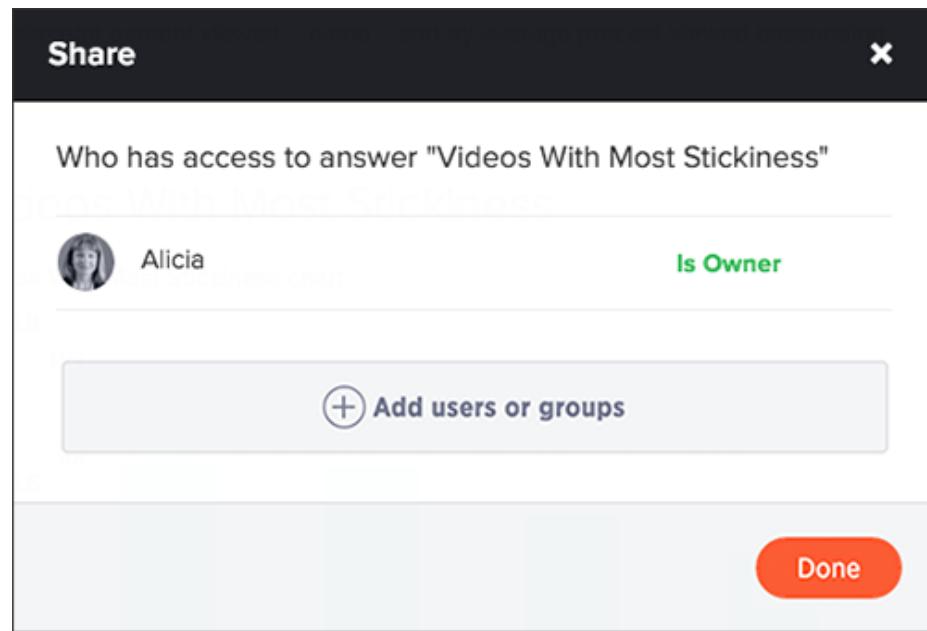
**Summary:** You do not have to be an administrator or the owner to share saved answers. Any user can share them, based on the access levels the user has.

Whenever you view an answer, you have the option of sharing it with others. It will be shared in its current state, so if you have modified the answer by interacting with the table or chart, the modified version is what will be shared.

1. Configure the answer to look as you'll want it to appear when shared.
2. Save the answer by clicking **Actions** and **Save**.
3. Click **Actions** and then **Share**.



4. Click **+ Add users or groups** and select users or groups that you want to share with.



5. Configure the level of access by selecting from the dropdown list. You can select:
  - **Can View** to provide read-only access. If the user doesn't have access to the underlying worksheet, they can only view the shared answer.
  - **Can Edit** to allow modification. Enables renaming or deleting the shared answer. If a user with edit privileges modifies a shared answer, their changes will be saved to it.
6. Click **Add and Save**.
7. Click **Done**.

# Share uploaded data

If you upload a spreadsheet, you can share **Can View** or **Can Edit** privileges with other people, who can further share them with others.

## Understand data sharing

Data that you uploaded from a Web browser is only visible to you and to the administrator. You can share the entire uploaded table, or only some of its columns.

By default, if you share only some table columns, users with access to those columns cannot see the other column's data in their searches. However, the data can become visible if a worksheet or pinboard that *also contain* those columns is shared with these users.

If you want to prevent shares of these types from also revealing the columns, you can ask your administrator to enable a stricter behavior.

## How to share data

Share uploaded data by following these steps:

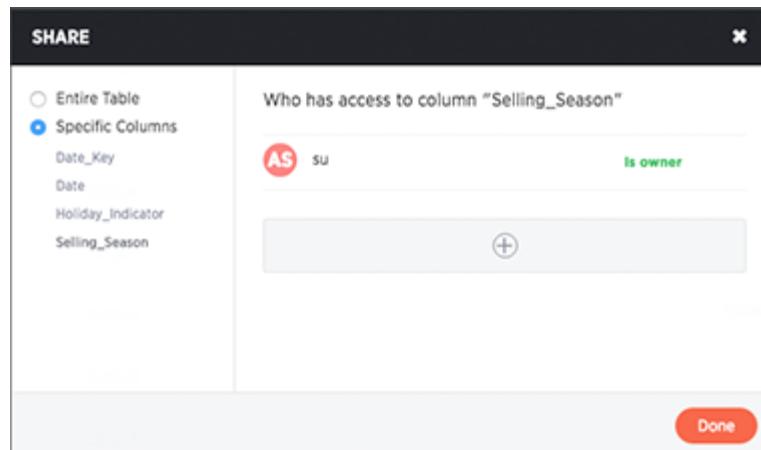
1. Click on **DATA**, on the top navigation bar.



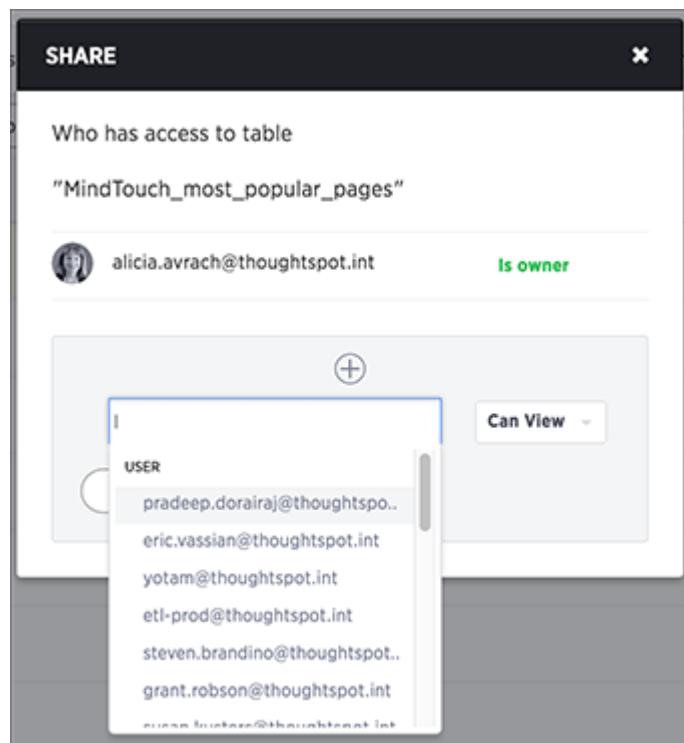
2. Click on the name of the uploaded data you want to share.
3. Click the **Share** icon.

A screenshot of the ThoughtSpot Tables page. At the top, there are tabs for 'Tables' (which is underlined) and 'Data Sources'. Below that is a filter bar with 'All', 'Yours', 'All types', 'Worksheets', 'Tables', and a 'Stickers' icon. A prominent orange box highlights the 'Share' button, which has a circular arrow icon. Below the filter bar is a table with columns for NAME, DESCRIPTION, and SOURCE. Three rows are listed: 'Fish1 WS' (with a green 'MM' icon), 'Fish Worksheet' (with a green 'MA' icon), and 'fish\_seas\_oceans' (with a yellow 'SS' icon and checked). The last row is highlighted with a yellow background.

4. Select **Entire Table** or **Specific Columns**.



5. If you selected **Specific Columns**, select the column(s) to share.
6. Click **+ Add users or groups** and select the users and groups that you want to share with.



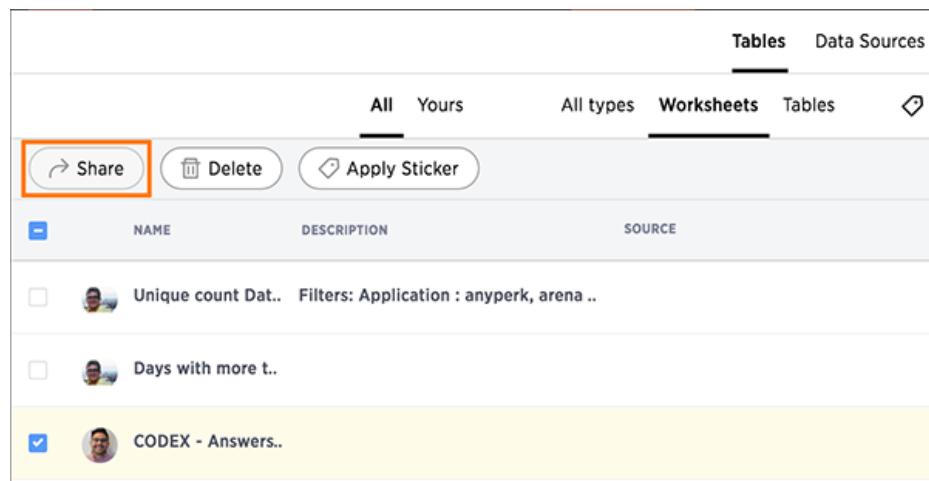
7. Configure the level of access by selecting from the dropdown list. You can select:
  - **Can View** to provide read-only access. This enables viewing the table data and defining worksheets on the table.
  - **Can Edit** to allow modification. This enables renaming, modifying, or deleting the entire table and adding or removing its columns.
8. Click **Add and Save**.
9. Click **Done**.

# Revoke access (unshare)

You may need to revoke access to an object (table, worksheet, or pinboard) that you have previously shared. Unsharing an object is very similar to sharing it.

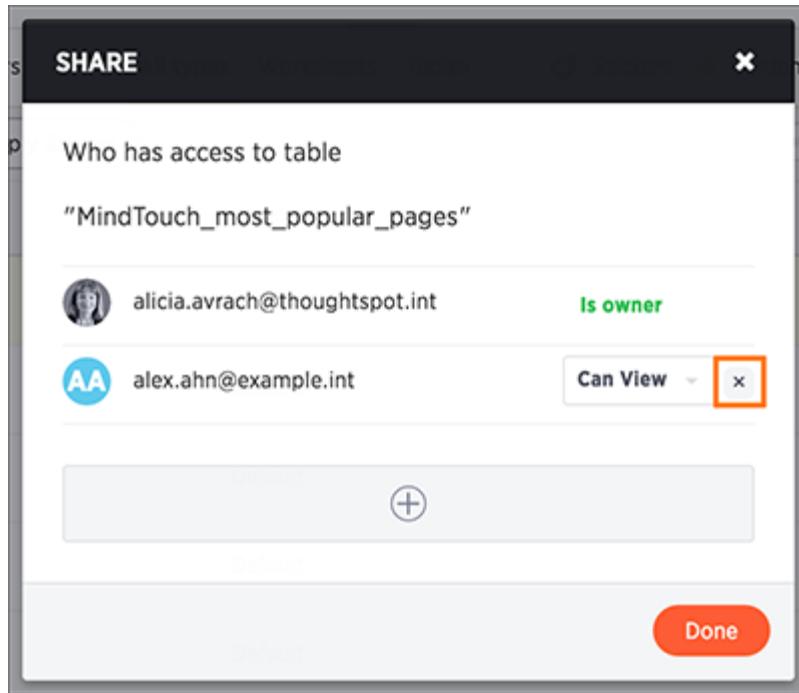
To unshare one or more objects:

1. Go to the area where the object(s) you want to unshare is located. From the top menu bar:
  - If the object is a table or worksheet, click **DATA**.
  - If the object is a pinboard, click **Pinboards**.
  - If the object is an answer, click **Answers**.
2. Find the object(s) in the list, and check the corresponding box(es).
3. Click the **Share** icon.



The screenshot shows a list of objects under the 'Worksheets' tab. At the top, there are buttons for 'Share', 'Delete', and 'Apply Sticker'. The 'Share' button is highlighted with a red box. Below the buttons is a table with columns: NAME, DESCRIPTION, and SOURCE. There are three rows of data. The first two rows have empty checkboxes in the NAME column. The third row has a checked checkbox in the NAME column, and its background is highlighted with a yellow gradient. The 'SOURCE' column for the third row shows 'CODEX - Answers..'. The top navigation bar includes tabs for 'Tables' and 'Data Sources', and buttons for 'All', 'Yours', 'All types', 'Worksheets' (which is selected), and 'Tables'.

4. Click the **X** next to the users and groups that you want to remove from sharing.



5. Click **Done**.

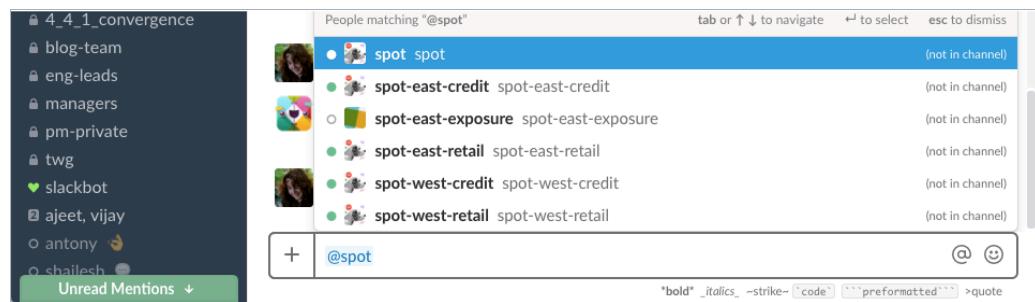
# Slack and Spot

Slack is an enterprise software platform that allows teams to communicate effectively through a messaging application. Slack also allows users to communicate with applications like ThoughtSpot through chat.

Spot is a ThoughtSpot integration with Slack.

## Does your Slack have Spot?

Spot has to be integrated with your Slack team before you can use it. Your team admin or ThoughtSpot admin can do this. To test if your Slack team has a Spot integration, mention **@spot** and see if he barks back:



In this particular channel, **@spot** is there for you but like his brothers **@spot-east-credit** is not in the channel. Scroll down to Frequently asked questions to get started with Spot.

If **@spot** doesn't come when you "call" you are spotless. Ask your administrator to see if you can get one.

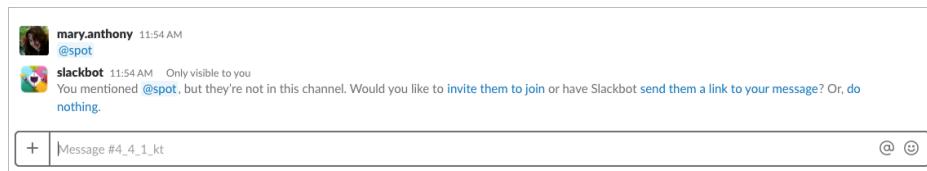
# How to use Spot

You can't ask Spot to do tricks until you invite him to a channel and authorize yourself as a Spot trainer.

## Call Spot and make him do tricks

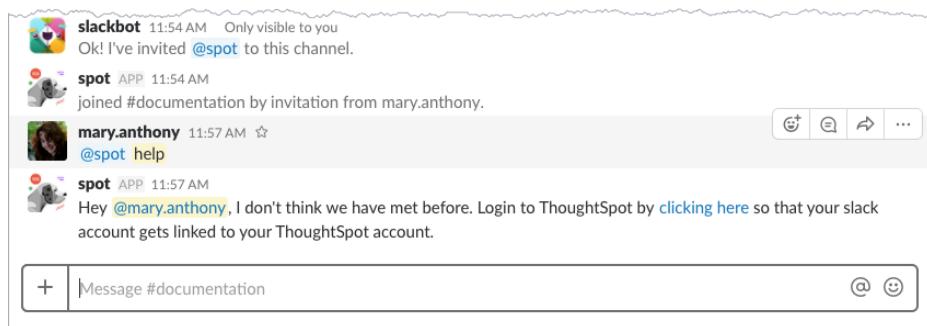
1. Launch Slack if you haven't already.
2. Go to a channel where you want to invite Spot.
3. Invite **@spot** like you would invite anyone else.

Type **@spot** and press **RETURN**.



4. Click invite them to join.
5. Ask him for help.

If this is the first time you've commanded him, Spot tells you:

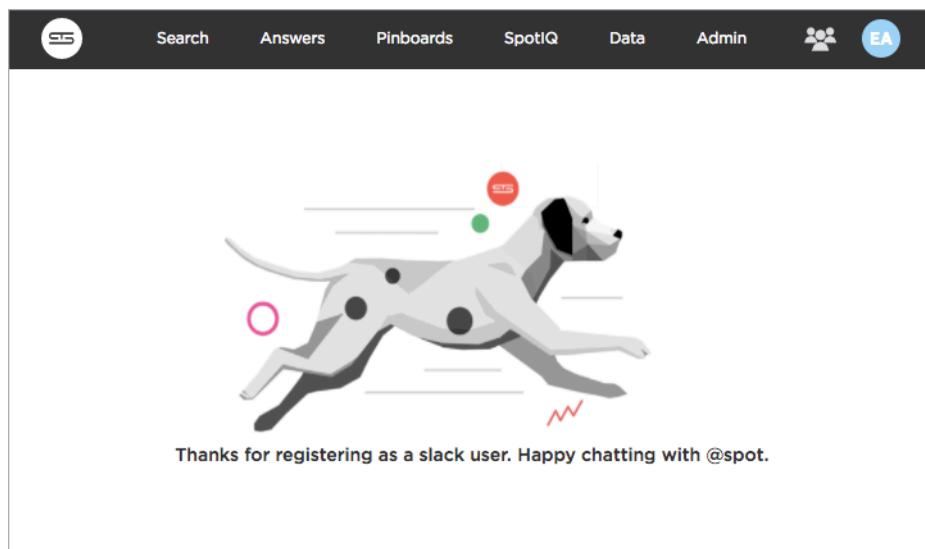


6. Take the [clicking here](#) link.

Spot takes you out of Slack and to the ThoughtSpot application.

7. Enter your username and password.

If you do this successfully, the application tells you:



8. Go back to Slack and ask Spot to list what he does by typing `@spot help`.

mary.anthony 11:58 AM  
@spot help

spot APP 11:58 AM  
(61 kB) ▶

@spot is at your service! Here's what I can do.

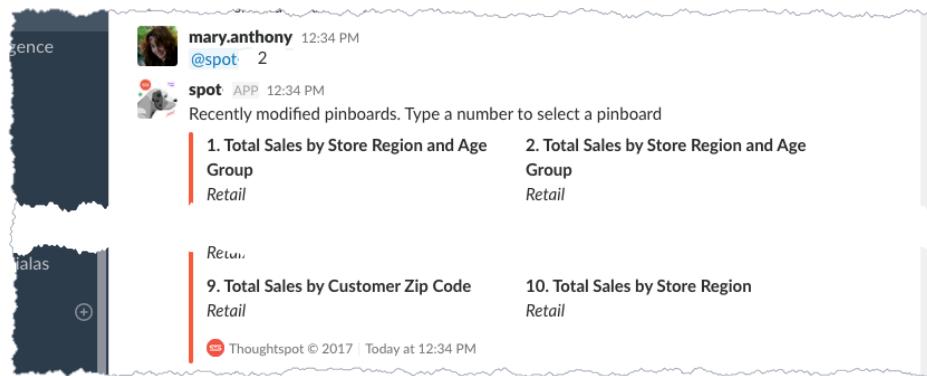
<b>1. &lt;anything&gt;</b> Will try to match a chart or pinboard with your query, Eg. <i>total weekly active users</i>	<b>2. list pinboards</b> Will list most recently modified pinboards
<b>3. subscriptions</b> Will give you a list of the items subscribed by you.	<b>4. subscribe</b> Search for a chart, display it. Type <code>subscribe</code> to subscribe to it.
<b>5. more</b> Eg: When seeing a list of objects, type <code>more</code> to see more of them.	

Thoughtspot © 2017 Today at 11:58 AM

+ Message #documentation @ 😊

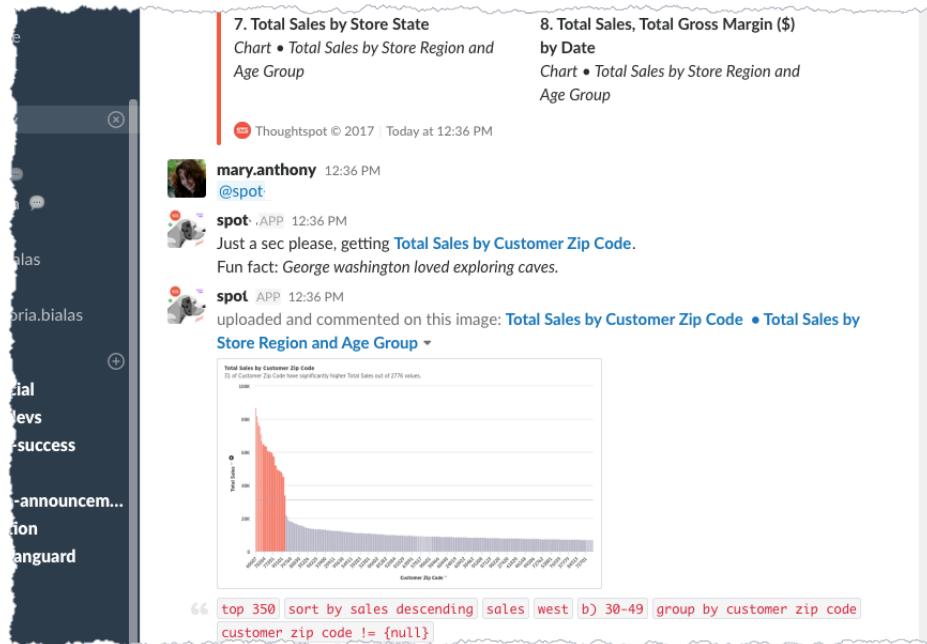
Spot can run a query directly that's the `**1. **` command.

9. Try simply listing your subscriptions.



**Note:** Spot only looks for objects with the **spot** sticker on them. Work your way from a pinboard to asking for a particular chart.

10. Work your way from a pinboard to asking for a particular chart.



At this point, you should go onto explore your Spot's talents on your own!

## Frequently Asked Questions

Whenever possible, Spot answers questions for you directly in channel. Here, though, are some frequent questions owners ask.

### Can I name my Spot whatever I want?

Sure. This is done by your administrator when the integration is created. Um, I guess the answer is really “ask your parents?”

## Spot isn't fetching what I want, why?

Spot can only fetch “toys” that have the spot sticker. Your administrator should do this when he sets up ThoughtSpot.

## My spot looks different, why is that?

The screenshots in this document were taken on desktop integration of Slack. If you are using mobile device, your interactions with Spot will look different.

## How does Spot know you?

The first time you message Spot, it will send back a link which you can click to link your ThoughtSpot account to Slack account. Once the accounts are linked all charts you query from Spot will be tied to your permissions/authorization.

## How to subscribe to charts posted to a channel?

Any chart retrieved using spot can be subscribed to, and can have any number of channel/user subscribers.

## How do I get the list of Spot commands?

Messaging `help` to `spot` gives you a list of commands and how to use them.

# More help and support

**Summary:** There are many locations on the web where you can find help for ThoughtSpot.

There are multiple locations inside and outside the ThoughtSpot application where you can get more help. This page describes some of the ways to get more help inside or outside of the application interface.

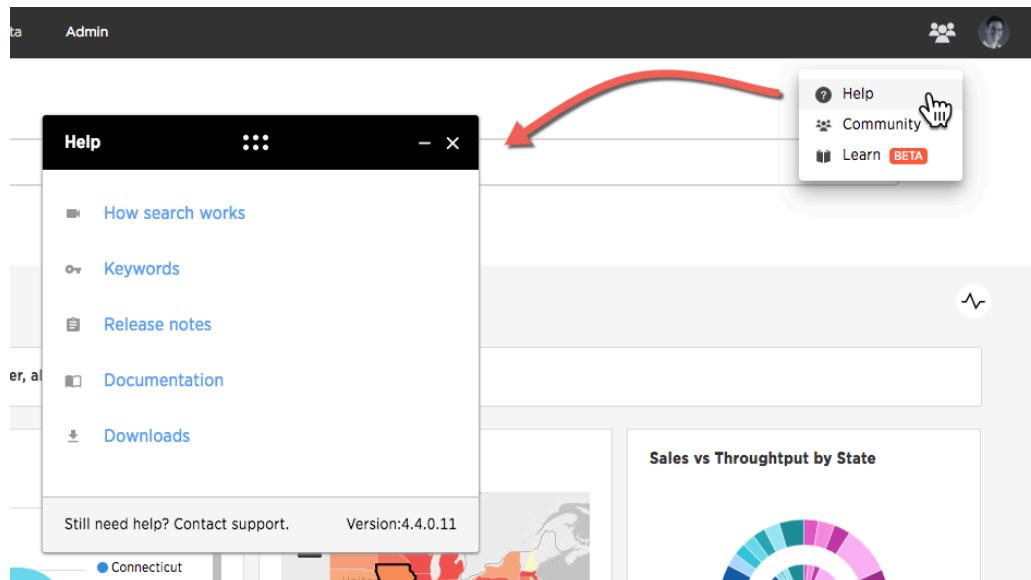
Each version of ThoughtSpot has its own documentation set. ThoughtSpot releases new documentation sets for every major release version (`x.y`) of ThoughtSpot. Documentation features added with minor versions (`x.y.z`) are added to the corresponding major release. This means, for example, that new documentation for 4.4.1 minor release version appears in the 4.4 major version documentation.

**Tip:** ThoughtSpot supplies release notes for every major release and every minor release version (`x.y.z`).

## Help menu

You can navigate directly to this site on the web by entering [docs.thoughtSpot.com](https://docs.thoughtSpot.com) in your browser's address bar. By default, the latest version (4.4) is presented first. ThoughtSpot versions prior to 4.4 have documentation located at [help.thoughtSpot.com](https://help.thoughtSpot.com).

You can navigate to this documentation from inside the ThoughtSpot application as well. In this case, the application links directly to the documentation version that matches your product version.



From this dialog you can get to the following:

Link	Takes you to ...

**How search works** a short video that explains how to use the **Search** bar

**Keywords** a product keyword reference

**Release notes** The notes for the ThoughtSpot version.

**Documentation** this documentation set

**Downloads** a list of software downloads such as connectors

Along the bottom of the **Help** dialog you'll see a link to contact Support as well as the version of ThoughtSpot you are using.

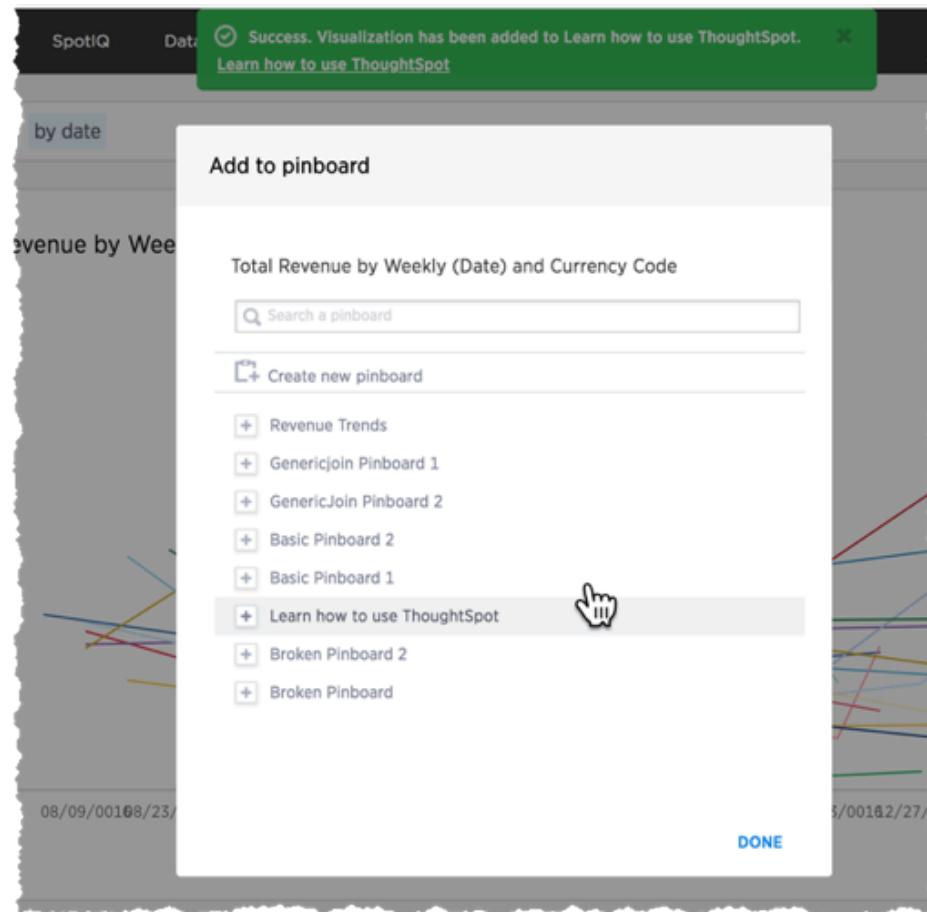
## Learn

Choosing **Learn** from the **Help** menu takes you to the *Learn* center. The center contains one or more example queries. You can only play replay on data that you have access to.

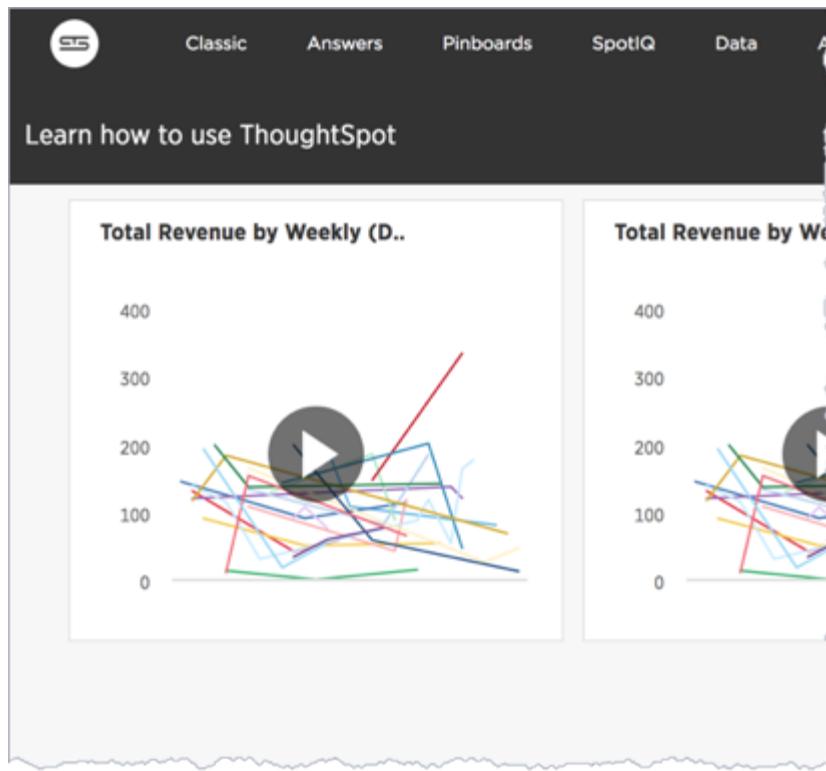
Click on an item in Learn to replay it in ThoughtSpot. During a replay, you can click **Pause** or **Resume** or click **ESC** to exit the replay.

Any administrator can add to the items in the **Learn** center. To add a replay, do the following:

1. Search in your search bar.
2. Choose the pin from the answer that results. The system lists the pinboards you can pin to.
3. Select **Learn how to use ThoughtSpot** pinboard.



4. Navigate to the *Learn* center to replay your search.



## Search the help in search bar

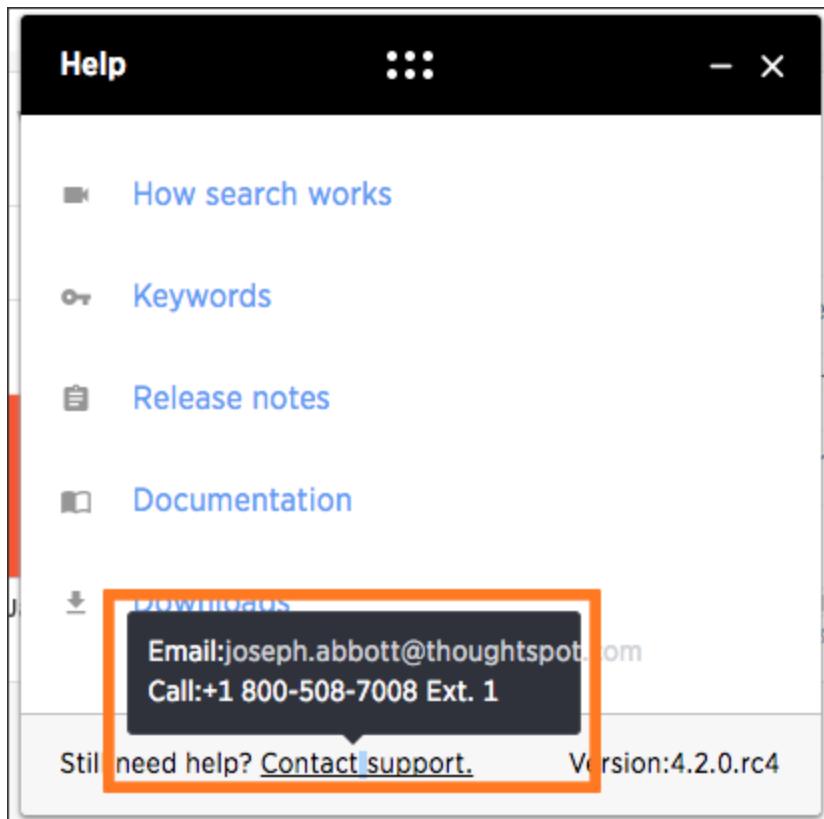
You can search the help directly from the search bar with the `how to` and `help` keywords.

Function	Examples
help	<code>help keywords</code>
how to	<code>how to search</code>

The `how to` keyword takes you into the documentation only. The `help` keyword allows you to jump directly into a product workflow.

## Support contact information

If you still can't find what you're looking for, you can contact support.



# Keyword reference

You can use keywords when asking a question to help define your search. This reference lists the various keywords. You can also see this list of keywords and examples from within the help center.

## General

Keyword	Examples
top	<ul style="list-style-type: none"> <li>top sales rep by count sales for average revenue &gt;10000</li> <li>sales rep average revenue for each region top</li> </ul>
bottom	<ul style="list-style-type: none"> <li>bottom revenue average revenue by state</li> <li>customer by revenue for each sales rep bottom</li> </ul>
<i>n</i>	top 10 sales rep revenue
<i>n</i>	bottom 25 customer by revenue for each sales rep
sort by	<ul style="list-style-type: none"> <li>revenue by state sort by average revenue</li> <li>revenue by customer sort by region</li> </ul>

## Date

Keyword	Examples
after	order date after 10/31/2014
before	order date before 03/01/2014
between ... and ...	order date between 01/30/2012 and 01/30/2014
daily year-over-year	growth of revenue by order date daily year-over-year
daily	shipments by region daily
day	count monday restaurant
day of week	revenue by day of week last 6 months
day of week	count shipments Monday
<i>n</i> days for each month	sales last 2 days for each month
<i>n</i> days for each quarter	revenue last 15 days for each quarter

Keyword	Examples
<i>n</i> days for each week	total sold last 2 days for each week
<i>n</i> days for each year	revenue last 300 days for each year
growth of ... by ... daily	growth of sales by order date daily
growth of ... by ... monthly	growth of sales by date shipped monthly sales > 24000
growth of ... by ... quarterly	growth of sales by date shipped quarterly
growth of ... by ... weekly	growth of sales by receipt date weekly for proski2000
growth of ... by ... yearly	growth of sales by date closed yearly
growth of ... by ...	growth of sales by order date
<i>n</i> hours for each day	sales last 2 hours for each day
last day by	customers last day by referrer
last month by	customers last month by day
last <i>n</i> days	visitors last 7 days
last <i>n</i> quarters	visitors last 2 quarters by month by campaign
last <i>n</i> weeks	visitors last 10 weeks by day
last quarter	customers last quarter sale >300
last week	customers last week by store
last year	top 10 customers last year by sale by store for region west
month to date	sales by product month to date sales > 2400
month year	commission by sales rep February 2014
<i>month</i>	commission January
month	revenue by month last year
monthly year-over-year	growth of revenue by receipt date monthly year-over-year
monthly	commission > 10000 monthly

Keyword	Examples
<i>n</i> months for each quarter	cost last 2 months for each quarter
<i>n</i> months for each year	last 8 months for each year
<i>n</i> days ago	sales 2 days ago
<i>n</i> months ago	sales 2 months ago by region
<i>n</i> months	visitors last 6 months for homepage visits > 30 by month
<i>n</i> quarters ago	sales 4 quarters ago by product name contains deluxe
<i>n</i> weeks ago	sales 4 weeks ago by store
<i>n</i> years ago	sales 5 years ago by store for region west
<i>n</i> years	opportunities next 5 years by revenue
<i>n</i> years	visitors last 5 years by revenue for sum revenue >5000
next day	shipments next day by order
next month	appointments next month by day
next <i>n</i> days	shipments next 7 days
next <i>n</i> months	openings next 6 months location
next <i>n</i> quarters	opportunities next 2 quarters by campaign
next <i>n</i> weeks	shipments next 10 weeks by day
next quarter	opportunities next quarter amount > 30000
next week	shipments next week by store
next year	opportunities next year by sales rep
quarter to date	sales by product quarter to date for top 10 products by sales
quarterly year-over-year	growth of revenue by date shipped quarterly year-over-year
quarterly	sales quarterly for each product
<i>n</i> quarters for each year	last 2 quarters for each year
today	sales today by store

Keyword	Examples
week to date	sales by order date week to date for pro-ski200
week	revenue by week last quarter
weekly year-over-year	growth of revenue by date shipped weekly year-over-year
weekly	revenue weekly
$n$ weeks for each month	sales last 3 weeks for each month
$n$ weeks for each quarter	last 2 weeks for each quarter
$n$ weeks for each year	last 3 weeks for each year
year to date	sales by product year to date
year	revenue by product 2014 product name contains snowboard
yearly	shipments by product yearly
yesterday	sales yesterday for pro -ski200 by store

## Time

Keyword	Examples
detailed	ship time detailed
last minute	count homepage views last minute
last hour	count unique visits last hour
$n$ minutes	count visitors last 30 minutes
$n$ hours	count visitors last 12 hours
hourly	visitors by page name hourly
$n$ minutes ago	sum inventory by product 10 minutes ago

Keyword	Examples
n hours ago	sum inventory by product by store 2 hours ago

## Text

Keyword	Examples
begins with	product name begins with 'pro'
contains	product name contains "alpine" description contains "snow shoe"
ends with	product name ends with 'deluxe'
not begins with	product name not begins with "tom's"
not contains	product color not contains 'tan' product color not contains 'red'
not ends with	product name not ends with "trial"
similar to	course name similar to 'hand'
not similar to	course name not similar to 'hand'

## Number

Function	Examples
sum	sum revenue
average	average revenue by store
count	count visitors by site
max	max sales by visitor by site
min	min revenue by store by campaign for cost > 5000
standard deviation	standard deviation revenue by product by month for date after 10/31/2010
unique count	unique count visitor by product page last week

Function	Examples
variance	variance sale amount by visitor by product for last year

## Comparative

Function	Examples
all	all
between... and	revenue between 0 and 1000
vs, versus	revenue east vs west
>	sum sale amount by visitor by product for last year sale amount > 2000
<	unique count visitor by product by store for sale amount < 20
>=	count calls by employee lastname >= m
<=	count shipments by city latitude <= 0
=	unique count visitor by store purchased products = 3 for last 5 days
!=	sum sale amount region != canada region != mexico

## Location

Keyword	Examples
near	revenue store name county near san francisco
near... within n miles km meters	revenue store name county near alameda within 50 miles
farther than n miles km meters from	average hours worked branch farther than 80 km from scarborough

Location keywords only work for searches where the data source includes latitude/longitude data.

## Period

Keyword	Example
quarter ( <i>date</i> )	quarter (commit date)
quarter of year ( <i>date</i> )	quarter of year (commit date)
month of quarter ( <i>date</i> )	month of quarter (commit date)
week of year ( <i>date</i> )	week of year (commit date)
week of quarter ( <i>date</i> )	week of quarter (commit date)
week of month ( <i>date</i> )	week of month (commit date)
day of year ( <i>date</i> )	day of year (commit date)
day of quarter ( <i>date</i> )	day of quarter (commit date)
day ( <i>date</i> )	day (order date)
day of month ( <i>date</i> )	day of monthly (order date)
day of week ( <i>date</i> )	day of week (order date)
hour ( <i>datetime</i> )	hour (timestamp)

## Help

Function	Examples
help	help keywords
how to	how to search

# TQL reference

TQL is the ThoughtSpot language for entering SQL commands. This reference lists TQL commands you can use to do things like creating a schema or verifying a data load.

## About using TQL

You can use TQL either [through the ThoughtSpot application's web interface](#) or the [command line interface](#) in the Linux shell.

Use `--query_results_apply_top_row_count <number>` flag to limit the number of result rows returned by a query. For example:

```
$ tql --query_results_apply_top_row_count 100
```

As a best practice, you should enclose object names (database, schema, table, and column) in double quotes, and column values in single quotes. When referring to objects using fully qualified object names, the syntax is:

```
"database"."schema"."table"
```

To get help on SQL when using TQL, enter `help` on the command line.

You can use TQL to view and modify schemas and data in tables. Remember to add a semicolon after each command. Commands are not case sensitive but are capitalized here for readability.

**Note:** Worksheets and pinboards in ThoughtSpot are dependent upon the data in the underlying tables. Use caution when modifying tables directly. If you change or remove a schema on which those objects rely, the objects could become invalid.

## View schemas and data

Syntax	Description
<code>SHOW DATABASES</code>	Lists all available databases.  Examples:  <code>SHOW DATABASES;</code>

Syntax	Description
<code>USE &lt;database&gt;</code>	Switches the context to the specified database. This is required if queries do not use fully qualified names (database.schema.table) for specifying tables.  Examples:  <code>USE "fruit_database";</code>
<code>SHOW SCHEMAS</code>	Lists all schemas within the current database. For example:  <code>SHOW SCHEMAS;</code>
<code>SHOW TABLES</code>	Lists all tables within the current database by schema. For example:  <code>SHOW TABLES;</code>
<code>SHOW TABLE &lt;table&gt;</code>	Lists all the columns for a table. For example:  <code>SHOW TABLE "locations";</code>
<code>SCRIPT SERVER</code>	Generates the TQL schema for all tables in all databases on the server. For example:  <code>SCRIPT SERVER;</code>
<code>SCRIPT DATABASE &lt;database&gt;</code>	Generates the TQL schema for all tables in a database. For example:  <code>SCRIPT DATABASE "fruit_database";</code>
<code>SCRIPT TABLE &lt;table&gt;</code>	Generates the TQL schema for a table. For example:  <code>SCRIPT TABLE "vendor";</code>

Syntax	Description
<pre>SELECT   &lt;cols_or_expr&gt;   FROM &lt;table_list&gt;   [WHERE &lt;predicates&gt;]   [GROUP BY &lt;expr&gt;]   [ORDER BY &lt;expr&gt;]</pre>	<p>Shows specified set of table data.</p> <p>If you do not specify the TOP number of rows to select, the top 50 rows will be returned by default. The number of rows to return can be set using the TQL command line flag:</p> <pre>--query_results apply_top_row_count</pre> <p>You can use the following aggregation functions:</p> <ul style="list-style-type: none"> <li>• sum</li> <li>• count</li> <li>• count distinct</li> <li>• stddev</li> <li>• avg</li> <li>• variance</li> <li>• min</li> <li>• max</li> </ul> <p>You can use the following date functions:</p> <ul style="list-style-type: none"> <li>• absyear</li> <li>• absmonth</li> <li>• absday</li> <li>• absquarter</li> <li>• date</li> <li>• time</li> </ul> <p>For example:</p> <pre>SELECT TOP 10 "quantity" FROM "sales_fact";</pre> <pre>SELECT COUNT(*) FROM "vendor";</pre> <pre>SELECT "vendor", SUM("quantity") FROM "sales_fact" GROUP BY "vendor";</pre> <pre>SELECT "vendor", SUM("amount") FROM "vendor", "sales_fact"   WHERE "sales_fact"."vendorid" = "vendor"."vendorid"     AND "amount" &gt; 100 GROUP BY "vendor" ORDER BY "amount" DESC;</pre> <pre>SELECT "vendor", SUM("quantity") FROM "sales_fact" GROUP BY "vendor" LIMIT 10;</pre>

## Schema creation

Syntax	Description
<code>CREATE DATABASE &lt;database&gt;</code>	Creates a database. For example:  <code>CREATE DATABASE "fruit_database";</code>
<code>CREATE SCHEMA &lt;schema&gt;</code>	Creates a schema within the current database. For example:  <code>CREATE SCHEMA "fruit_schema";</code>
<code>CREATE TABLE &lt;table&gt; (&lt;column_definitions&gt; [&lt;constraints&gt;]) [PARTITION BY HASH (&lt;number&gt;) [KEY ("&lt;column&gt;")]]</code>	Creates a table with the specified column definitions and constraints.  Use <code>PARTITION BY HASH</code> to shard a table across all nodes. If no <code>KEY</code> is specified, the table will be randomly sharded.  Do not specify relationship constraints ( <code>FOREIGN KEY</code> or <code>RELATIONSHIP</code> ) in the <code>CREATE TABLE</code> statement. Instead, define these using <code>ALTER TABLE</code> statements at the end of your TQL script, after creating your tables. This method guarantees that tables are created before they are referenced in the constraint definitions. For example:  <code>CREATE TABLE "vendor" ("vendorid" int, "name" varchar(255));</code>  <code>CREATE TABLE "sales_fact" ("saleid" int, "locationid" int, "vendorid" int, "quantity" int, "sale_amount" double, "fruitid" int, CONSTRAINT PRIMARY KEY("saleid")) PARTITION BY HASH(96) KEY ("saleid");</code>

## Schema modification

Syntax	Description
<code>DROP DATABASE &lt;database&gt;</code>	Drops a database and all of its schemas and tables. For example:  <code>DROP DATABASE "fruit_database";</code>

Syntax	Description
<code>DROP SCHEMA &lt;schema&gt;</code>	Drops a schema within the current database, and drops all of the tables in the schema. For example:  <code>DROP SCHEMA "fruit_schema";</code>
<code>DROP TABLE &lt;table&gt;</code>	Drops a table. For example:  <code>DROP TABLE "location";</code>
<code>TRUNCATE TABLE &lt;table&gt;</code>	Removes all data from a table, but preserves its metadata, including all GUIDs, relationships, etc. This can be used to force a new schema for a table without losing the metadata.  However, this operation removes all existing data from the table and must be used with caution. You must reload the data following a <code>TRUNCATE</code> , or all dependent objects (worksheets and pinboards) in ThoughtSpot will become invalid. For example:  <code>TRUNCATE TABLE "location";</code>
<code>ALTER TABLE &lt;table&gt; ADD   DROP   RENAME COLUMN     &lt;column&gt;</code>	Alters a table to add, drop, or rename a column.  When you add a column to an existing table, you must provide a default value to use for existing rows. For example:  <code>ALTER TABLE "cart" ADD COLUMN "nickname" varchar(255) DEFAULT 'no nickname';</code>  <code>ALTER TABLE "cart" DROP COLUMN "nickname";</code>  <code>ALTER TABLE "cart" RENAME COLUMN "nickname" TO "shortname";</code>
<code>ALTER TABLE &lt;table&gt; DROP CONSTRAINT PRIMARY KEY;</code>	Drops the primary key from a table.  Note that if you then add a new primary key, the same upsert behavior will be applied as with adding any primary key. This can result in data deletion, so make sure you understand how the upsert will affect your data ahead of time. For example:  <code>ALTER TABLE "sales" DROP CONSTRAINT PRIMARY KEY;</code>  <code>ALTER TABLE "sales" ADD CONSTRAINT PRIMARY KEY ("PO_number");</code>

Syntax	Description
<pre>ALTER TABLE &lt;table&gt; DROP [FOREIGN KEY] RELATIONSHIP &lt;name&gt;;</pre>	<p>Drops the named foreign key or relationship between two tables. For example:</p> <pre>ALTER TABLE "sales_fact" DROP FOREIGN KEY "FK_PO_number"; ALTER TABLE "fruit_dim" DROP RELATIONSHIP "REL_dates";</pre>

<pre>ALTER TABLE &lt;table&gt; DROP [CONSTRAINT FOREIGN KEY [&lt;table_name&gt;]    RELATIONSHIP [WITH &lt;table_name&gt;];</pre>	<p>You must use this syntax when dropping relationships between tables created before ThoughtSpot version 3.2. This is because relationships could not be named in older versions.</p> <p>Drops the foreign key or relationship between two tables where you cannot reference it by relationship name. If the relationship was created without a name, use:</p> <ul style="list-style-type: none"> <li>• the name of the referenced table, for a foreign key.</li> <li>• the name of the related table, for a relationship.</li> </ul> <p>If you drop a foreign key without specifying the referenced table, all foreign keys from the table you are altering will be dropped.</p> <p>Examples:</p> <pre>ALTER TABLE "shipments" DROP CONSTRAINT FOREIGN KEY "orders";</pre> <pre>ALTER TABLE "wholesale_buys" DROP RELATIONSHIP WITH "retail_sales";</pre> <p>Drops all relationships that have wholesale_buys as a source.</p> <pre>ALTER TABLE "wholesale_buys" DROP RELATIONSHIP;</pre> <p>Drops all foreign keys from wholesale_buys.</p> <pre>ALTER TABLE "wholesale_buys" DROP CONSTRAINT FOREIGN KEY;</pre>
---	---

Syntax	Description
<pre>ALTER TABLE &lt;table&gt; [SET DIMENSION   SET FACT [PARTITION BY       HASH [(&lt;shards&gt;)] [KEY(&lt;column&gt;)]]]</pre>	<p>Changes the partitioning on a table by doing one of:</p> <ul style="list-style-type: none"> <li>• re-sharding a sharded table</li> <li>• changing a replicated table to a sharded table</li> <li>• changing a sharded table to a replicated (unsharded) table</li> </ul> <p>By default, ThoughtSpot does not shard dimension tables.</p> <p>To change the partitioning on a table, or to change a dimension table to a sharded table, use <code>ALTER TABLE...SET FACT PARTITION BY HASH...;</code></p> <p>To make a sharded table into a dimension table (replicated on every node), use <code>ALTER TABLE...SET DIMENSION;</code> command.</p> <p>Examples of this statement:</p> <pre>ALTER TABLE "sales_fact" SET FACT PARTITION BY HASH (96) KEY ("PO_number"); ALTER TABLE "fruit_dim" SET DIMENSION;</pre>

<pre>ALTER TABLE &lt;table&gt; MODIFY COLUMN &lt;column&gt; &lt;new_data_type&gt;;</pre>	<p>Changes the data type of a column. This can have implications on sharding and primary key behavior. See <a href="#">About data type conversion</a>. For example:</p> <pre>ALTER TABLE fact100 MODIFY COLUMN product_id int;</pre>
--	--

## Modify data

Syntax	Description
<pre>INSERT INTO &lt;table&gt; VALUES ...</pre>	<p>Inserts values into a table. Only use this for testing. Do not use <code>INSERT</code> on a production system. For example:</p> <pre>INSERT INTO "vendor" VALUES 'helen rose', 'jacob norse', 'eileen ruff', 'manny gates';</pre>

<pre>ALTER TABLE &lt;table&gt; SET LOAD PRIORITY &lt;value&gt; &lt;new_data_type&gt;;</pre>	<p>Sets the load priority for a table. Load priority determines the order in which a table is loaded on a cluster restart. You can set any value from <code>1-100</code>. The system default for all tables is <code>50</code>. For example:</p> <pre>ALTER TABLE 'sales_facts' SET LOAD PRIORITY 1</pre>
---	---

Syntax	Description
<pre>UPDATE &lt;table&gt; ... SET ... [WHERE ...]</pre>	<p>Updates rows in a table that match optionally provided predicates. Predicates have the form <code>column = value</code> connected by the <code>AND</code> keyword.</p> <p>Sets the column values to the specified values. For example:</p> <pre>UPDATE "location" SET "borough" = 'staten island', "city" = 'new york' WHERE "borough" = 'staten isl' AND city = 'NY';</pre>

<pre>DELETE FROM &lt;table&gt; [WHERE...]</pre>	<p>Deletes rows from a table that match optionally provided predicates. Predicates have the form <code>column = value</code> connected by the <code>AND</code> keyword.</p> <p>For example:</p> <pre>DELETE FROM "vendor" WHERE "name" = 'Joey Smith' AND "vendorid" = '19463';</pre>
---	---

## Constraints and relationships

Constraints and relationships in ThoughtSpot are used to define the relationships between tables (i.e. how they can be joined). However, constraints are not enforced, as they would be in a transactional database. You can define the following constraints when creating a table with `CREATE TABLE`, or add them to an existing table using the `ADD CONSTRAINT` syntax:

Syntax	Description
<pre>PRIMARY KEY</pre>	<p>Designates a unique, non-null value as the primary key for a table. This can be one column or a combination of columns. If values are not unique, an upsert will be performed if a row includes a primary key that is already present in the data. Some examples:</p> <pre>CREATE TABLE "schools" ( "schoolID" varchar(15), "schoolName" varchar(255), "schoolCity" varchar(55), "schoolState" varchar(55), "schoolNick" varchar(55), CONSTRAINT PRIMARY KEY ("schoolID") ) ;  ALTER TABLE "cart" ADD CONSTRAINT PRIMARY KEY ("cart_id");  ALTER TABLE "cart" DROP CONSTRAINT PRIMARY KEY "cart_id";</pre>

Syntax	Description
FOREIGN KEY	<p>Defines a relationship where the value(s) in the table are used to join to a second table. Uses an equality operator. The foreign key must match the primary key of the table that is referenced in number, column type, and order of columns.</p> <p>When creating a foreign key, give it a name. You can reference the foreign key name later, if you want to remove it.</p> <p>Examples of this statement:</p> <pre>ALTER TABLE "batting" ADD CONSTRAINT "FK_player" FOREIGN KEY ("playerID") REFERENCES "players" ("playerID");  ALTER TABLE "batting" ADD CONSTRAINT "FK_lg_team" FOREIGN KEY ("lgID" , "teamID") REFERENCES "teams" ("lgID" , "teamID");  ALTER TABLE "shipment" ADD CONSTRAINT "FK_PO_vendor" FOREIGN KEY ("po_number", "vendor") REFERENCES "orders" ("po_number", "vendor");  ALTER TABLE "shipment" DROP CONSTRAINT "FK_PO_vendor";</pre>
RELATIONSHIP	<p>Defines a relationship where the value(s) in the table can be used to join to a second table, using an equality condition (required) and one or more range conditions (optional). These conditions act like a WHERE clause when the two tables are joined. They are applied using AND logic, such that all conditions must be met for a row to be included.</p> <p>You may add multiple relationships between tables. When creating a relationship, give it a name. You can reference the relationship name later, if you want to remove it.</p> <p>Examples of this statement:</p> <pre>ALTER TABLE "wholesale_buys" ADD RELATIONSHIP "REL_fruit" WITH "retail_sales" AS "wholesale_buys"."fruit" = "retail_sales"."fruit" AND ("wholesale_buys"."date_order" &lt; "retail_sales"."date_sold" AND "retail_sales"."date_sold" &lt; "wholesale_buys"."expire_date");  ALTER TABLE "wholesale_buys" DROP RELATIONSHIP "REL_fruit";</pre>

## Data types

ThoughtSpot supports a simplified list of data types:

Syntax	Description	Examples
Character	<ul style="list-style-type: none"><li>VARCHAR(<i>n</i>)</li></ul>	Specify the maximum number of characters, as in VARCHAR(255). The size limit is 1GB for VARCHAR values.
Floating point	<ul style="list-style-type: none"><li>DOUBLE</li><li>FLOAT</li></ul>	DOUBLE is recommended.
Boolean	<ul style="list-style-type: none"><li>BOOL</li></ul>	Can be <code>true</code> or <code>false</code> .
Integer	<ul style="list-style-type: none"><li>INT</li><li>BIGINT</li></ul>	INT holds 32 bits. BIGINT holds 64 bits.
Date or time	<ul style="list-style-type: none"><li>DATE</li><li>DATETIME</li><li>TIMESTAMP</li><li>TIME</li></ul>	DATETIME, TIMESTAMP, and TIME are stored at the granularity of seconds  TIMESTAMP is identical to DATETIME, but is included for syntax compatibility.

# tsload flag reference

For recurring data loads and for scripting loads, use `tsload` (the ThoughtSpot Loader). This reference section lists all the flags that can be used to modify the behavior of `tsload`.

## General tsload flags

Flag	Description	Notes
<code>--target_database &lt;database&gt;</code>	Specifies the pre-existing target database into which tsload should load the data.	
<code>--target_schema &lt;schema&gt;</code>	Specifies the target schema. Default is “falcon_default_schema”.	
<code>--target_table &lt;table&gt;</code>	Specifies the tables that you want to load into the database. The tables must exist in the database specified by <code>--target_database</code> .	
<code>--empty_target</code>	Specifies that any data in the target table is to be removed before the new data is loaded. If supplied, any rows that exist in the table specified by <code>--target_database</code> and <code>--target_table</code> will be deleted before this data load. To perform an “upsert” on the existing data, omit this flag or specify <code>--noempty_target</code> .	
<code>--max_ignored_rows &lt;number&gt;</code>	Specifies the maximum number of rows that can be ignored if they fail to load. If the number of ignored rows exceeds this limit, the load will be aborted.	
<code>--bad_records_file &lt;path_to_file&gt;/&lt;file_name&gt;</code>	Specifies the file to use for storing rows that failed to load. Input rows that do not conform to the defined schema in ThoughtSpot will be ignored and inserted into this file.	
<code>--date_format &lt;date_formatmask&gt;</code>	Specifies the format string for date values. The default format is <code>yearmonthday</code> e.g. “Dec 30th, 2001” and is represented as <code>20011230</code> . Use the date format specifications supported in the <a href="#">strftime library function</a> .	
<code>--date_time_format &lt;date_formatmask&gt; &lt;time_formatmask&gt;</code>	Specifies the format string for datetime values. The default is <code>yearmonthday hour:minute:second</code> e.g. Dec 30th, 2001 1:15:12 and is represented as <code>20011230 01:15:12</code> . Use the datetime format specifications supported in the <a href="#">strftime library function</a> .	
<code>--time_format &lt;time_formatmask&gt;</code>	Specifies the format string for time values. The default is <code>hour:minute:second</code> . Use the time format specifications supported in the <a href="#">strftime library function</a> .	

Flag	Description	Notes
--v=[0 1 2 3]	Specifies the verbosity of log messages.	Provide a value for verbosity level. By default, verbosity is set to the minimum, which is 0. This value is similar to a volume control. At higher levels your log receives more messages and that log more frequently. This is used for debugging. You should not change this value unless instructed by ThoughtSpot Support.
--skip_second_fraction	Skips fractional seconds when loading data.	If supplied, the upserts logic may be affected, especially if the date time being loaded is a primary key, and the data has millisecond granularity. Load the data twice, once as a string with a primary key, and again with second granularity date time. There is no support to store fractional seconds in the ThoughtSpot system.

## File loading tsload flags

The following flags are used when loading data from an input file:

Flag	Description	Notes
--source_file <path_to_file>/<file_name>	Specifies the location of the file to be loaded.	
--source_data_format [csv delimited]	Specifies the data file format.	Optional. The default is csv.
--field_separator "<delimiter>"	Specifies the field delimiter used in the input file.	
--trailing_field_separator	Specifies that the field separator appears after every field, including the last field per row.	Example row with trailing field separator: a,b,c,The default is false.
--null_value "<null_representation>"	Specifies how null values are represented in the input file.	These values will be converted to NULL upon loading.
--date_converted_to_epoch [true false]	Specifies whether the “date” or “datetime” values in the input file are represented as epoch values.	
--boolean_representation [true_false   1_0   T_F   Y_N]	Specifies the format in which boolean values are represented in the input file.	The default is T_F. You can also use this flag to specify other values. For example, if your data used Y for true and NULL for false, you could specify: --boolean_representation Y_NULL

Flag	Description	Notes
<code>--has_header_row</code>	Indicates that the input file contains a header row.	If supplied, the first row of the file is ignored. If not supplied, the first row of the file is loaded as data.
<code>--escape_character "&lt;character&gt;"</code>	Specifies the escape character used in the input file.	If no value is specified, the default is "(double quotes).
<code>--enclosing_character "&lt;character&gt;"</code>	Specifies the enclosing character used in the input file.	If the enclosing character is double quotes, you need to escape it, as in this example: <code>--enclosing_character "\""</code>
<code>--use_bit_boolean_values = [true   false]</code>	Specifies how boolean values are represented in the input file.	If supplied, the input CSV file uses a bit for boolean values, i.e. the false value is represented as 0x0 and true as 0x1. If omitted or set to false, boolean values are assumed to be T_F, unless you specify something else using the flag <code>--boolean_representation [true_false   1_0   T_F   Y_N]</code> .

# tscli command reference

The `tscli` command line interface is an administration interface for the ThoughtSpot instance. Use `tscli` to take snapshots (backups) of data, apply updates, stop and start the services, and view information about the system. This reference defines each subcommand and what you can accomplish with it.

The command returns 0 upon success and a non-zero exit code upon failure. Because the `tscli` command is typically running a command on multiple nodes, an error may be called at different points. As much as possible, the command attempts to save errors to the `stderr` directory as configured on a node.

## How to use the tscli command

The `tscli` command has the following syntax:

```
tscli [-h] [--helpfull] [--verbose] [--noautoconfig]
      [--autoconfig] [--yes] [--cluster <cluster>]
      [--zoo <zookeeper>] [--username username] [--identity_file identity_file]
      {alert,backup,backup-policy,callhome,cluster,command,dr-mirror,etl,event,
       feature,fileserver,firewall,hdfs,ldap,logs,map-tiles,monitoring,nas,node,
       os,saml,scheduled-pinboards,smtp,snapshot,snapshot-policy,spot,ssl,storage,
       support}
```

The `tscli` command has several subcommands such as `alert`, `backup`, and so forth. You issue a subcommand using the following format:

```
tscli [subcommand]
```

Subcommands have their own additional options and actions such as `tscli backup create` or `tscli backup delete` for example. To view help for a subcommand:

```
tscli [subcommand] -h
```

A subcommand itself may have several options.

# tscli subcommands

This section lists each subcommand and its syntax.

## alert subcommand

```
tscli alert [-h] {count,info,list,off,on,refresh,silence,status,unsilence}
```

Use this subcommand to do the following:

- `tscli alert info` Lists all alerts.
- `tscli alert list` Lists the generated alerts.
- `tscli alert off` Disables all alerts from the cluster in the cluster's timezone.
- `tscli alert on` Enables alerts from the cluster.
- `tscli alert silence --name <alert_name>`

Silences the alert with `alert_name`. For example, `DISK_ERROR`. Silenced alerts are still recorded in postgres, however emails are not sent out.

- `tscli alert status` Shows the status of cluster alerts.
- `tscli alert unsilence-name alert_name`

Unsilences the alert with `* alert_name *`. For example, `DISK_ERROR`.

## backup subcommand

```
tscli backup [-h] {create,delete,ls,restore}
```

Use this subcommand to do the following:

- `tscli backup create [-h] [--mode {full,light,dataless}] [--type {full,incremental}] [--base BASE] [--storage_type {local,nas}] [--remote name out]`

Pulls a snapshot and saves it as a backup where:

- `--mode {full,light,dataless}`

Mode of backups. To understand these different modes see [Understand backup modes](#).

- `--type {full,incremental}` Type of backup.(Incremental `incremental` is not implemented yet) (default: full)
- `--base BASE`

Based snapshot name for incremental backup. (Not Implemented yet) (default: None)

- `--storage_type {local,nas}`

Storage type of output directory. (default: local)

- `--remote`

Take backup through orion master. (default: True)

- `tscli backup delete * name *` Deletes the named backup.
- `tscli backup ls` List all backups taken by the system.
- `tscli backup restore` Restore cluster using backup.

## backup-policy

```
tscli backup-policy [-h] {create,delete,disable,enable,ls,show,status,update}
```

Use this subcommand to do the following:

- `tscli backup-policy create` Prompts an editor for you to edit the parameters of the backup policy.
- `tscli backup-policy delete name` Deletes the backup policy with `name`.
- `tscli backup-policy disable name` Disables the policy `name`.
- `tscli backup-policy enable name` Enables the policy `name`.
- `tscli backup-policy ls` List backup policies.
- `tscli backup-policy show name` Show the policy `name`.
- `tscli backup-policy status name` Enables the policy `name`.
- `tscli backup-policy update * name *` Prompts an editor for you to edit the policy `name`.

## callhome

```
tscli callhome [-h] {disable,enable,generate-bundle}
```

Use this subcommand to do the following:

- `tscli callhome disable` Turns off the periodic call home feature.
- `tscli callhome enable --customer_name customer_name`

Enables the “call home” feature, which sends usage statistics to ThoughtSpot. This feature is enabled by default.

The parameter `customer_name` takes the form `Shared/*`customer_name`*/stats`.

- `tscli callhome generate-bundle -d directory --since DAYS`
  - `--d D` Dest folder where tar file will be created. (default: None)
  - `--since DAYS`

Grab callhome data from this time window in the past. Should be a human readable duration string, e.g. `4h` (4 hours), `30m` (30 minutes), `1d` (1 day). (default: None) Generates a tar file of the cluster metrics and writes it to the specified directory where `DAYS` is how far back you'd like to generate the tar file from in days. For example, `30`. If this parameter is not specified, the command will collect the stats from the last `7` days by default.

## cluster

```
tscli cluster [-h] {abort-reinstall-os,check,create,get-config,load,reinstall-os,report,restore,resume-reinstall-os,resume-update,set-config,set-min-resource-spec,show-resource-spec,start,status,stop,update,update-hadoop}
```

Use this subcommand to do the following:

- `tscli cluster abort-reinstall-os` Abort in-progress reinstall.
- `tscli cluster check --includes {all,disk,zookeeper,hdfs,orion-cgroups,orion-oreo}` check the status nodes in the cluster.

You must specify a component to check.

- `tscli cluster create release`

Creates a new cluster from the release file specified by `* release *`. This command is used by ThoughtSpot Support when installing a new cluster, for example, `tscli cluster create 2.0.4.tar.gz`

- `tscli cluster get-config` Get current cluster network and time configuration. Prints JSON configuration to stdout. If for some reason the system cannot be connected to all interfaces, the command returns an error but continues to function.
- `tscli cluster load` Load state from given backup onto existing cluster
- `tscli cluster reinstall-os` Reinstall OS on all nodes of the cluster.
- `tscli cluster report` Generate cluster report.
- `tscli cluster restore --release release_tarball backupdir`

Restores a cluster using the backup in the specified directory `backupdir`. If you're restoring from a dataless backup, you must supply the release tarball for the corresponding software release.

- `tscli cluster resume-reinstall-os` Resume in-progress reinstall.
- `tscli cluster resume-update` Resume in-progress updates.
- `tscli cluster set-config` Set cluster network and time configuration. Takes JSON configuration from stdin.
- `tscli cluster set-min-resource-spec` Sets min resource configuration of the cluster
- `tscli cluster show-resource-spec` Prints default or min.
- `tscli cluster start` Start cluster.
- `tscli cluster status` Gives the status of the cluster, including release number, date last updated, number of nodes, pending tables time, and services status.
- `tscli cluster stop` Pauses the cluster (but does not stop storage services).
- `tscli cluster update` Update existing cluster.
- `tscli cluster update-hadoop` Updates Hadoop/Zookeeper on the cluster.

## command

```
tscli command [-h] {run}
```

Command to run a command on all nodes.

```
tscli command run [-h] [--nodes NODES] --dest_dir DEST_DIR [--copyfirst COPYFIRST] [--timeout TIMEOUT] command
```

- `--nodes NODES` Space separated IPs of nodes where you want to run the command.  
(default: `all`)
- `--dest_dir DEST_DIR` Directory to save the files containing output from each nodes.  
(default: None)
- `--copyfirst COPYFIRST` Copy the executable to required nodes first. (default: `False`)
- `--timeout TIMEOUT` Timeout waiting for the command to finish. (default: `60`)

## dr-mirror

```
tscli dr-mirror [-h] {start,status,stop}
```

- `tscli dr-mirror start` Starts a mirror cluster which will continuously recover from a primary cluster.
- `tscli dr-mirror status` Checks whether the current cluster is running in mirror mode.
- `tscli dr-mirror stop` Stops mirroring on the local cluster.

## etl

```
tscli etl [-h] {change-password,disable-lw,download-agent,enabl e-lw,show-lw}
```

- `tscli etl change-password --admin_username admin_user --username Informatica_user`

Changes the Informatica Cloud account password used by ThoughtSpot Data Connect.  
Required parameters are:

- `--admin_username admin_user` specifies the Administrator username for ThoughtSpot.
- `--username Informatica_user` specifies the username for the Informatica Cloud.
- `tscli etl disable-lw` Disables ThoughtSpot Data Connect.
- `tscli etl download-agent` Downloads the ThoughtSpot Data Connect agent to the

cluster.

- `tscli etl enable-lw [-h] --username USERNAME --thoughtspot_url THOUGHTSPOT_URL --admin_username ADMIN_USERNAME [--groupname GROUPNAME] --org_id ORG_ID [--pin_to PIN_TO] [--proxy_host PROXY_HOST] [--proxy_port PROXY_PORT] [--proxy_username PROXY_USERNAME] [--max_wait MAX_WAIT]`

You should contact ThoughtSpot Support for assistance in setting this up. Required parameters are:

- `--username USERNAME` Username for Informatica Cloud (default: None)
- `--thoughtspot_url THOUGHTSPOT_URL` URL to reach thoughtspot. (default: None)
- `--admin_username ADMIN_USERNAME` Admin username for ThoughtSpot (default: None)
- `--groupname GROUPNAME`
- `--org_id ORG_ID` specifies the Informatica `id` of the organization (company). For ThoughtSpot, this is `001ZFA`. `org_id` shouldn't include the prefix `Org`. For example, if on Informatica cloud, the `orgid` is `Org003XYZ`, then use only
- `--pin_to PIN_TO` specifies the IP address to pin to. If you specify an IP to pin to, that node becomes sticky to the Informatica agent, and will always be used. Defaults to the public IP address of the localhost where this command was run.
- `--proxy_host PROXY_HOST` Proxy server host for network access (default: :)
- `--proxy_port PROXY_PORT` Proxy server port (default: :)
- `--proxy_username PROXY_USERNAME` Proxy server username (default: :)
- `--max_wait MAX_WAIT` Maximum time in seconds to wait for Data Connect agentto start (default: None)
- `tscli etl show-lw` Shows the status of ThoughtSpot Data Connect. It also returns the Informatica username and OrgId.

## event

```
tscli event [-h] {list}
```

This subcommand has the following actions:

- ```
tscli event list [-h] [--include INCLUDE] [--since SINCE] [--from FROM] [--to TO] [--limit LIMIT] [--detail] [--summary_contains SUMMARY_CONTAINS] [--detail_contains DETAIL_CONTAINS] [--attributes ATTRIBUTES]
```
- `--include INCLUDE` Options are all, config, notification. Default config. (default: config)
  - `--since SINCE` Grab events from this time window in the past. Should be a human readable duration string, e.g. `4h` (4 hours), `30m` (30 minutes), `1d` (1 day). (default: None)
  - `--from FROM` Begin timestamp, must be of the form: `yyyymmdd-HH:MM` (default: None)
  - `--to TO` End timestamp, must be of the form: `yyyymmdd-HH:MM` (default: None)
  - `--limit LIMIT` Max number of events to fetch. (default: 0)
  - `--detail` Print events in detail format. This is not tabular. Default is a tabular summary. (default: False)
  - `--summary_contains SUMMARY_CONTAINS` Summary of the event will be checked for this string. Multiple strings to check for can be specified by separating them with `|` (event returned if it matchesALL). Put single quotes around the param value to prevent undesired glob expansion (default: None)

- `--detail_contains DETAIL_CONTAINS` Details of the event will be checked for this string. Multiple strings to check for can be specified by separating them with `|` (event returned if it matches ALL). Put single quotes around the param value to prevent undesired glob expansion (default: None)
- `--attributes ATTRIBUTES` Specify attributes to match as key=value. Multiple attributes to check for can be specified by separating them with `|` (event returned if it matches ALL). Put single quotes around the param value to prevent undesired glob expansion (default: None)

## feature

```
tscli feature [-h] {get-all-config}
```

This subcommand has the following actions:

`tscli feature get-all-config` Gets the configured features in a cluster. The command will return a list of features, such as custom branding, Data Connect, and call home, and tell you whether they are enabled or disabled.

## fileserver

```
tscli fileserver [-h] {configure,download-release,purge-config,show-config,upload}
```

This subcommand has the following actions:

- `tscli fileserver configure [-h] --user USER [--password PASSWORD]` Configures the secure file server username and password for file upload/download and the call home feature. You only need to issue this command once, to set up the connection to the secure file server. You only need to reissue this command if the password changes. The parameter `PASSWORD` is optional. If a password is not specified, you will be prompted to enter it.
- `tscli fileserver download-release [-h] [--user USER] [--password PASSWORD] release` Downloads the specified release file and its checksum. Specify the release by number, to the second decimal point (e.g. 3.1.0, 3.0.5, etc.). You may optionally specify the `--user` and `--password` to bypass the credentials that were specified when configuring the file server connection with `tscli fileserver configure`. Before using this command for the first time, you need to set up the file server connection using `tscli fileserver configure`.
- `tscli fileserver purge-config` Removes the file server configuration.
- `tscli fileserver show-config` Shows the file server configuration.
- `tscli fileserver upload [-h] [--user USER] [--password PASSWORD] --file_name FILE_NAME* --server_dir_path * SERVER_DIR_PATH*`

Uploads the file specified to the directory specified on the secure file server. You may optionally specify the `--user` and `--password` to bypass the credentials that were specified when configuring the file server connection with `tscli fileserver configure`. Before using this command for the first time, you need to set up the file server connection using `tscli fileserver configure`.

Accepts these flags

- `--user USER` Username of fileserver (default: None)
- `--password PASSWORD` Password of fileserver (default: None). This is required and the command prompts you for it if you do not supply it.
- `--file_name FILE_NAME` Local file that needs to be uploaded (default: None)
- `--server_dir_path SERVER_DIR_PATH` Directory path on fileserver. (default: None) The `SERVER_DIR_PATH` parameter specifies the directory to which you want to upload the file. It is based on your customer name, and takes the form `/Shared/support/* customer_name *`.

## firewall

```
tscli firewall [-h] {close-ports,disable,enable,open-ports,stat  
us}
```

- `tscli firewall close-ports`

Closes given ports through firewall on all nodes. Takes a list of ports to close, comma separated. Only closes ports which were previously opened using “open-ports”. Ignores ports which were not previously opened with “open-ports” or were already closed.

- `tscli firewall disable` Disable firewall.
- `tscli firewall enable` Enable firewall.
- `tscli firewall open-ports * ports *`

Opens given ports through firewall on all nodes. Takes a list of ports to open, comma separated. Ignores ports which are already open. Some essential ports are always kept open (e.g. `ssh`), they are not affected by this command or by `close-ports`.

- `tscli firewall status` Shows whether firewall is currently enabled or disabled.

## hdfs

```
tscli hdfs [-h] {leave-safemode}
```

This subcommand has the following actions:

`tscli hdfs leave-safemode` Command to get HDFS namenodes out of safemode.

## ldap

```
tscli ldap [-h] {add-cert,configure,purge-configuration}
```

This subcommand has the following actions:

- `tscli ldap add-cert name certificate`

Adds an SSL certificate for LDAP. Use only if LDAP has been configured without SSL and you wish to add it. Use `* name *` to supply an alias for the certificate you are installing.

- `tscli ldap configure`

Configures LDAP using an interactive script. You can see detailed instructions for setting up LDAP in [About LDAP integration](#).

- `tscli ldap purge-configuration` Purges (removes) any existing LDAP configuration.

## logs

```
tscli logs [-h] {collect,runcmd}
```

This subcommand has the following actions:

- `tscli logs collect [-h] [--include INCLUDE] [--exclude EXCLUDE] [--since SINCE] [--from FROM] [--to TO] [--out OUT] [--maxsize MAXSIZE] [--sizeonly] [--nodes NODES]`

Extracts logs from the cluster. Does not include any logs that have been deleted due to log rotation.

These parameters have the following values:

- `--include INCLUDE`

Specifies a comma separated list of logs to include. Each entry is either a “selector” or a glob for matching files. Selectors must be among: `all`, `orion`, `system`, `ts`. Anything starting with `/` is assumed to be a glob pattern and interpreted via `find(1)`. Other entries are ignored. Put single quotes around the param value to prevent undesired glob expansion (default: `all`)

- `--exclude EXCLUDE`

Comma separated list of logs to exclude. Applies to the list selected by `-include`. Params are interpreted just like in `-include` (default: None)

- `--since SINCE`

Grab logs from this time window in the past. Should be a human readable duration string, e.g. 4h (4 hours), 30m (30 minutes), 1d (1 day). (default: None)

- `--from FROM` Timestamp where collection begins, must be of the form: `yyyymmdd-HH:MM` (default: None)
- `--to TO` Timestamp where collection ends, must be of the form: `yyyymmdd-HH:MM` (default: None)
- `--out OUT` Tarball path for dumping logs from each node (default: `/tmp/logs.tar.gz`)
- `--maxsize MAXSIZE` Only fetch logs if size is smaller than this value. Can be specified in megabytes/gigabytes, e.g. 100MB, 10GB. (default: None)
- `--sizeonly` Do not collect logs. Just report the size. (default: False)
- `--nodes NODES` Comma separated list of nodes from where to collect logs. Skip this to use all nodes. (default: None)

- `tscli logs runcmd [-h] --cmd CMD [--include INCLUDE] [--exclude EXCLUDE] [--since SINCE] [--from FROM] [--to TO] [--outfile OUTFILE] [--outdir OUTDIR] [--cmd_infmt CMD_INFMT] [--cmd_outfmt CMD_OUTFMT] [--nodes NODES]`

Runs a Unix command on logs in the cluster matching the given constraints. Results are reported as text dumped to standard out, the specified output file, or as tarballs dumped into the specified directory.

- `--cmd CMD`

Unix-Command to be run on the selected logs. Use single quotes to escape spaces etc. Language used to specify CMDSTR has following rules.

- A logfile and its corresponding result file can be referred by keywords `SRCFILE` & `DSTFILE`. eg. `cp SRCFILE DSTFILE`
- Without any reference to `DSTFILE` in CMDSTR, `> DSTFILE` will be appended to CMDSTR for output redirection. eg `du -sch SRCFILE` gets auto- transtalated to `du -sch SRCFILE > DSTFILE`
- Without any reference to `SRCFILE`, content of log is streamed to CMDSTR via pipe. eg. `tail -n100 | grep ERROR` gets auto- transtalated to `cat SRCFILE | tail -n100 | grep ERROR > DSTFILE` (default: None)

- `--include INCLUDE`

Comma separated list of logs to include,each entry is either a “selector” or a glob for matching files.Selectors must be among: `all` , `orion` , `system` , `ts` . Anything starting with / is assumed to be a glob pattern and interpreted via `find(1)` . Other entries are ignored. TIP: put single quotes around the param value to prevent undesired glob expansion (default: all)

- `--exclude EXCLUDE`

Comma separated list of logs to exclude. Applies to the list selected by `--include` . Params are interpreted just like in `--include` (default: None)

- `--since SINCE`

Grab logs from this time window in the past. Should be a human readable duration string, e.g. `4h` (4 hours), `30m` (30 minutes), `1d` (1 day). (default: None)

- `--from FROM` Timestamp where collection begins, must be of the form: `yyyymmdd-HH:MM` (default: None)

- `--to TO` Timestamp where collection ends, must be of the form: `yyyymmdd-HH:MM` (default: None)

- `--outfile OUTFILE` File path for printing all the results. By default printed to stdout (default: None)

- `--outdir OUTDIR` Directory path for dumping results with original dir structure from each node. Used as an alternative to printing output to outfile/stdout (default: None)

- `--cmd_infmt CMD_INFMT` Specify if the inputfile should be compressed/ uncompressed before running `CMD` . `C` =compressed, `U` =uncompressed. Don’t use this flag if `CMD` works on both (default: None)

- `--cmd_outfmt CMD_OUTFMT` Specify if `OUTFILE` generated by `CMD` will be compressed/uncompressed. `C` =compressed, `U` =uncompressed. Don't use this flag if output file will be of same format as input file (default: None)
- `--nodes NODES` Comma separated list of nodes where to run command. Skip this to use all nodes. (default: None)

## map-tiles

```
tscli map-tiles [-h] {disable,enable}
```

This subcommand supports the following actions:

- `tscli map-tiles enable [-h] [--online] [--offline] [--tar TAR] [--md5 MD5]`  
Enables ThoughtSpot's map tiles, which are used when constructing geomap charts. If you don't have interest access, you must download the map tiles tar and md5 files. Then you must append the following to the `tscli` command.
  - `--online` Download `maptiles` tar from internet. (default: True)
  - `--offline` Using `maptiles` tar from local disk. (default: False)
  - `--tar TAR` Specified tar file for map-tiles. (default: )
  - `--md5 MD5` Specified md5 file for map-tiles. (default: )
- `tscli map-tiles disable` Disable map-tiles functionality.

## monitoring

```
tscli monitoring [-h] {set-config,show-config}
```

This subcommand has the following actions:

- `tscli monitoring set-config [-h] [--email EMAIL] [--clear_email] [--heartbeat_interval HEARTBEAT_INTERVAL] [--heartbeat_disable] [--report_interval REPORT_INTERVAL] [--report_disable]` Sets the monitoring configuration.
  - `--email EMAIL` Comma separated list (no spaces) of email addresses where the cluster will send monitoring information.
  - `--clear_email` Disable emails by clearing email configuration. (default: False)
  - `--heartbeat_interval HEARTBEAT_INTERVAL` Heartbeat email generation interval in seconds. Should be greater than 0.
  - `--heartbeat_disable` Disable heartbeat email generation. (default: False)
  - `--report_interval REPORT_INTERVAL` Cluster report email generation interval in seconds. Should be greater than 0.
  - `--report_disable` Disable cluster report email generation. (default: False)
- `tscli monitoring show-config` Shows the monitoring configuration.

## nas

```
tscli nas [-h] {ls,mount-cifs,mount-nfs,unmount}
```

This subcommand has the following actions:

- `tscli nas ls [-h]` List mounts managed by NAS mounter service.
- `tscli nas mount-cifs [-h] --server SERVER [--path_on_server PATH_ON_SERVER] --mount_point MOUNT_POINT --username USERNAME --password PASSWORD [--uid UID] [--gid GID] [--options OPTIONS]`

Mounts a CIFS device on all nodes.

- `--server SERVER` IP address or DNS name of CIFS service. For example, `10.20.30.40` (default: None)
  - `--path_on_server PATH_ON_SERVER` Filesystem path on the CIFS server to mount (source). For example: `/a` (default: `/`)
  - `--mount_point MOUNT_POINT`
- Directory on all cluster nodes where the NFS filesystem should be mounted (target). This directory does not need to already exist. If this directory already exists, a new directory is not created and the existing directory is used for mounting. For example: `/mnt/external` (default: None)
- `--username USERNAME` Username to connect to the CIFS filesystem as (default: None)
  - `--password PASSWORD` CIFS password for `--username` (default: None)
  - `--uid UID`
- `UID` that will own all files or directories on the mounted filesystem when the server does not provide ownership information. See `man mount.cifs` for more details. (default: `1001`)
- `--gid GID`
- Gid that will own all files or directories on the mounted filesystem when the server does not provide ownership information. See `man mount.cifs` for more details. (default: `1001`)
- `--options OPTIONS` Other command-line options to forward to `mount.cifs` command (default: `noexec`)

- `tscli nas mount-nfs [-h] --server SERVER [--protocol PROTO --path_on_server PATH_ON_SERVER] --mount_point MOUNT_POINT [--options OPTIONS]`

Mounts a NFS device on all nodes. Parameters are:

- `--server SERVER` IP address or DNS name of NFS service. For example, `10.20.30.40` (default: None)
- `--path_on_server PATH_ON_SERVER` Filesystem path on the NFS server to mount (source). For example: `/a/b/c/d` (default: `/`)

- `--mount_point MOUNT_POINT`

Directory on all cluster nodes where the NFS filesystem should be mounted (target). This directory does not need to already exist. If this directory already exists, a new directory is not created and the existing directory is used for mounting. For example: `/mnt/external` (default: None)

- `--options OPTIONS` Command-line options to forward to mount command (default: `noexec`).
- `--protocol PROTO` One of `nfs` or `nfs4`. The default is `nfs`.

- `tscli nas unmount [-h] --dir DIR`

Unmounts all devices from the specified `DIR` (directory) location. This command returns an error if nothing is currently mounted on this directory via `tscli nas mount` (default: None)

## node

```
tscli node [-h] {check,ls,reinstall-os,resume-reinstall-os,status} ...
```

This subcommand has the following actions:

- `tscli node check [-h] [--select {reinstall-preflight}] [--secondary SECONDARY ]`

Run checks per node. Takes the following parameters:

- `--select {reinstall-preflight}` Select the type of node check (default: `reinstall-preflight`)
- `--secondary SECONDARY` Secondary drive for `reinstall-preflight` (default: `sdd`)

- `tscli node ls [-h] [--type {all,healthy,not-healthy}]` Filter by node state (default: `all`)

- `tscli node reinstall-os [-h] [--secondary SECONDARY ] [--cluster]` Reinstall OS on a node. This takes the following parameters:

- `--secondary SECONDARY` Secondary drive to be used to carry to reinstall (default: `sdd`)
- `--cluster` Is the node part of a cluster (default: `False`)

- `tscli node resume-reinstall-os` Resume in-progress reinstall

## rpackage

```
tscli rpackage [-h] {add,delete,list} ...
```

Manages R packages available to SpotIQ.

- `tscli rpackage add [-h] [--repo REPO] [--timeout TIMEOUT] [--dest_dir DEST_DIR] [--nodes NODES] package_name` Command to add an R `package_name` to the cluster. This command has the following options:
  - `--repo REPO` Specify the url of a specific repo to download packages
  - `--timeout REPO` Timeout waiting for the R Package to be installed (default: 60)
  - `--dest_dir REPO` Directory where output of this command will be placed (default: None)
  - `--nodes NODES` Space separated IPs of nodes where you want to run the command. (default: all).
- `tscli rpackage add [-h] [--timeout TIMEOUT] [--dest_dir DEST_DIR] [--nodes NODES] package_name` Command to delete an installed R package from the cluster. This command has the following options:
  - `--timeout REPO` Timeout waiting for the R Package to be removed (default: 60)
  - `--dest_dir REPO` Directory where output of this command will be placed (default: None)
  - `--nodes NODES` Space separated IPs of nodes where you want to run the command. (default: all).
- `tscli rpackage list [-h] [--detailed]` List all R packages installed on the cluster.

## saml

```
tscli saml [-h] {configure,purge-configuration}
```

This subcommand has the following actions:

- `tscli saml configure [-h]` Configures SAML. To see a list of prerequisites refer to [Configure SAML](#).
- `tscli saml purge-configuration` Purges any existing SAML configuration.

## scheduled-pinboards

```
tscli scheduled-pinboards [-h] {disable,enable}
```

This subcommand has the following actions:

- `tscli scheduled-pinboards disable [-h]` Disable scheduled pinboards for this cluster.
- `tscli scheduled-pinboards enable [-h]` Enables scheduled pinboards, which is disabled in prod clusters by default.

**❶ Note:** When you enable scheduled pinboards, you should also configure a whitelist of intended email domains. Contact ThoughtSpot Support for help configuring a whitelist.

## smtp

```
tscli smtp [-h] {reset-canonical-mapping, set-canonical-mapping, set-mailfromname, set-mailname, set-relayhost, show-canonical-mapping, show-mailfromname, show-mailname, show-relayhost}
```

This subcommand takes supports the following actions:

- `tscli smtp reset-canonical-mapping` Deletes the current postmap mapping.
- `tscli smtp set-canonical-mapping [-h] new_key new_value` Sets a new Postmap mapping.
- `tscli smtp set-mailfromname mailfromname` Sets the name, an email address, from which email alerts are sent, for the cluster.
- `tscli smtp set-mailname mailname` Sets the mailname, a domain, where email alerts are sent, for the cluster.
- `tscli smtp set-relayhost [-h] [--force FORCE] relayhost` Sets the Relay Host for SMTP (email) sent from the cluster.
  - `--force FORCE` Set even if relay host is not accessible. (default: `False`)
- `tscli smtp show-canonical-mapping` Shows the current postmap mapping.
- `tscli smtp show-mailfromname` Shows the mailname, from which email alerts are sent, for the cluster.
- `tscli smtp show-mailname` Shows the mailname, where email alerts are sent, for the cluster.
- `tscli smtp show-relayhost` Shows the for SMTP (email) sent from the cluster. If there is no Relay Host configured, the command returns `NOT FOUND`.

## snapshot

```
tscli snapshot [-h] {backup,create,delete,ls,restore,update-ttl}
```

Learn more about snapshots and backups see the [Understand the backup strategies](#) documentation.  
This subcommand supports the following actions:

- `tscli snapshot backup [-h] [--mode {full,light,dataless}] [--type {full,incremental}] [--base BASE] [--storage_type {local,nas}] [--remote name out]`

Pull snapshot out as a backup. This takes the following parameters:

- `--mode {full,light,dataless}` Mode of backups. (default: `full`)
- `name` Name of snapshot to pull out as a backup. To list all snapshots, run `tscli snapshot ls`.
- `out` Directory where backup will be written, must not already exist.

- `--type {full,incremental}` Type of backup.(Incremental backup is not implemented yet) (default: `full`)
  - `--base BASE` Based snapshot name for incremental backup. (Not Implemented yet) (default: None)
  - `--storage_type {local,nas}` Storage type of output directory. (default: `local`)
  - `--remote` Take backup through Orion master. (default: `True`)
  
  - `tscli snapshot create [-h] name reason ttl`
- Creates a new snapshot with the `name` and `reason` provided. This command does not accept `.` (periods), but does accept `-` (dashes). The `ttl` parameters is the number of days after which this snapshot will be automatically deleted. A value of `-1` disables automatic deletion.
- `tscli snapshot delete [-h] name` Deletes the named snapshot.
  - `tscli snapshot ls [-h]` List available snapshots.
  - `tscli snapshot restore [-h] [--allow_release_change] [--only_service_state] name` Restore cluster to an existing snapshot. This takes the following parameters:
    - `--allow_release_change` Allow restoration to a snapshot at a different release. (default: `False`)
    - `--only_service_state` Restore only service state. (default: `False`)
  
  - `tscli snapshot update-ttl [-h] [--disable DISABLE] [name ttl]`

Updates manual snapshot garbage collection policy.

- `name` Specifies which snapshot to update.
- `ttl` Extends the manual snapshot `ttl` (time-to-live) value. Use a positive value to increase `ttl`. Use negative value to decrease it.
- `--disable DISABLE` Disable manual snapshot garbage collection. Setting this value to `True` will override any `ttl` value. (default: False)

## snapshot-policy

```
tscli snapshot-policy [-h] {disable,enable,show,update}
```

This subcommand supports the following actions:

- `tscli snapshot-policy disable [-h]` Disable snapshot policy.
- `tscli snapshot-policy enable -h` Enable specified snapshot policy.
- `tscli snapshot-policy show [-h]` Show snapshot policy.
- `tscli snapshot-policy update [-h] [--config CONFIG]` Update periodic snapshot config. This takes the following parameter:
  - `--config CONFIG` Text format of periodic backup policy config (default: None)

## spot

```
tscli spot [-h] {enable}
```

Enables Spot integration. This subcommand supports the following actions:

```
tscli spot enable [-h] --token TOKEN --thoughtspot_url THOUGHTSPOT_URL [--cache_timeout CACHE_TIMEOUT ]
```

- `--token TOKEN` Slack authroization token for Spot bot. This is required. You receive this token when your Slack administrator adds the Spot application.
- `--thoughtspot_url THOUGHTSPOT_URL` URL for the ThoughtSpot application. This is required.
- `--cache_timeout CACHE_TIMEOUT` Internal cache timeout (default: `60000`)

## ssl

```
tscli ssl [-h] {add-cert,clear-min-tls-version,off,on,rm-cert,set-min-tls-version,status,tls-status}
```

This subcommand supports the following actions:

- `tscli ssl add-cert [-h] key certificate` Adds an SSL certificate, key pair.
- `tscli ssl clear-min-tls-version [-h]` Clears any customizations for the minimum TLS version to support.
- `tscli ssl off`

Disables SSL. Disabling SSL will stop users from seeing a security warning when accessing ThoughtSpot from a browser if there is no SSL certificate installed.

- `tscli ssl on [-h]` If SSL is enabled and there is no certificate, users will see a security warning when accessing ThoughtSpot from a browser.
- `tscli ssl rm-cert` Removes the existing SSL certificate, if any.
- `tscli ssl set-min-tls-version [-h] {1.0,1.1,1.2}` Sets the minimum supported TLS version. Sets the minimum SSL version to be supported by the ThoughtSpot application. Please ensure that client browsers are enabled for this version or newer.
- `tscli ssl status` Shows whether SSL authentication is enabled or disabled.
- `tscli ssl tls-status [-h]` Prints the status of TLS support.

## storage

```
tscli storage [-h] gc df
```

This subcommand supports the following actions:

- `tscli storage gc [-h] [--log_age LOG_AGE] [--force] [--localhost_only]`

Garbage collect unused storage. Before issuing this command, you must stop the cluster using `tscli cluster stop`. After garbage collection has completed, you can restart the cluster with `tscli cluster start`. The command frees space in these directories:

- `/tmp`
- `/usr/local/scaligent/logs/`
- `/export/logs/orion`
- `/export/logs/oreo`
- `/export/logs/hadoop`
- `/export/logs/zookeeper`
- `cores`

Accepts these optional flags:

- `--log_age LOG_AGE`

Delete logs older than these many hours. Use a non-zero value ideally. A zero value will cause all temporary files to be deleted, including say those which are just temporarily closed while they are being passed from one component to the next.  
(default: 4 )

- `--force` Forces deletion of all logs and temporary files regardless of age. This must only be run on a stopped cluster. (default: False)
- `--localhost_only` If used, only the logs on the localhost will be removed. If not specified, the command acts on the entire cluster.

- `tscli storage df [--mode disk|hdfs]`

Checks the disk usage on the relevant mounts. Returns output similar to the Linux system command `df -h <directory>`.

## support

```
tscli support [-h]
{restart-remote,rm-admin-email, rm-admin-phone, rm-feedback-email,
set-admin-email, set-admin-phone, set-feedback-email, set-remote,
show-admin-email, show-admin-phone, show-feedback-email,
show-remote, start-remote, stop-remote}
```

This subcommand supports the following actions:

- `tscli support bundle [-h] [--include INCLUDE] [--exclude EXCLUDE] [--list_selectors] [--since SINCE] [--from FROM] [--to TO] [--out OUT] [--nodes NODES]`
- `--include INCLUDE` Comma separated list of selectors to include, each entry is either a “selector” or a glob for matching files. To see the list of valid selectors, run this command with `--list_selectors`. You may also specify: “`all`” to get all selectors and logs, and “`basic`” to get only the basic selectors. Selectors may also

be selectors meant for logs collect: all, orion, system, ts, or the name of a service. Anything starting with / is assumed to be a glob pattern and interpreted via find(1). Other entries are ignored. TIP: put single quotes around the param value to prevent undesired glob expansion. Use “all” to collect all selectors and all logs (default: all\_but\_logs)

- `--exclude EXCLUDE` Comma separated list of selectors to exclude. Applies to the list selected by `--include`. Params are interpreted just like in `--include`. Use the special keyword “logs” to exclude logs collection all together. (default: None)
  - `--list_selectors` List the selectors available for `--include` and `--exclude`, and then exit. (default: `False`)
  - `--since SINCE` Grab logs from this time window in the past. Should be a human readable duration string, e.g. `4h` (4 hours), `30m` (30 minutes), `1d` (1 day). (default: None)
  - `--from FROM` Timestamp where collection begins, must be of the form: `yyyymmdd-HH:MM` (default: None)
  - `--to TO` Timestamp where collection ends, must be of the form: `yyyymmdd-HH:MM` (default: None)
  - `--out OUT` Tarball path for dumping the support bundle (default: `/tmp/support_bundle.tar.gz`)
  - `--nodes NODES` Comma separated list of nodes from where to collect logs. Skip this to use all nodes. (default: None)
- 
- `tscli support restart-remote` Restarts remote support.
  - `tscli support rm-admin-email` Removes the email address for contacting the customer administrator. Replaces it with the default ThoughtSpot Support email address.
  - `tscli support rm-feedback-email` Removes the email address for product feedback. Replaces it with the default ThoughtSpot Support email address.
  - `tscli support rm-admin-phone` Removes the phone number for contacting the customer administrator. Replaces it with the default ThoughtSpot Support phone number.
  - `tscli support rm-feedback-email` Removes the email for sending feedback out of the system. If you would like to set a blank email address, issue the command `tscli support set-feedback-email ''`.
  - `tscli support set-admin-email email` Sets the email address for contacting the customer administrator. If you would like to display a blank email address, issue the command `tscli support set-admin-email ''`.
  - `tscli support set-feedback-email email` Sets the email address for sending feedback. If you would like to display a blank email address, issue the command `tscli support set-feedback-email ''`.
  - `tscli support set-admin-phone phone_number` Sets the phone number for contacting the customer administrator. Specify a phone number using any value (e.g. `+1 800-508-7008 Ext. 1`). If you would like to display a blank phone number, issue the command `tscli support set-admin-phone`.
  - `tscli support set-remote [-h] [--addr ADDR] [--user USER]` Configures the cluster for remote support through SSH tunneling, where `ADDR` is the address of support, e.g. `t unnel.thoughtspot.com`, and `USER` is the support username.
  - `tscli support show-admin-email` Shows the email address for customer administrator, if set.
  - `tscli support show-feedback-email` Shows the email address for product feedback, if set.
  - `tscli support show-admin-phone` Shows the phone number for customer administrator, if set.

set.

- `tscli support show-remote` Shows the status and configuration of remote support.
- `tscli support start-remote` Starts remote support.
- `tscli support stop-remote` Stops remote support.

## tokenauthentication

```
tscli cli tokenauthentication [-h] {disable,enable}
```

- `tscli cli tokenauthentication enable` Generates a token.
- `tscli cli tokenauthentication disable` Purges token login configuration.

# Date and time formats reference

This is a references for the date and time contexts and formats you can use with ThoughtSpot. You define data formats in specific contexts and, depending on the context, your choices in data formatting differ. You must understand date and time when you load data in these contexts:

- using data upload from the browser
- through `tsload` command
- through data connect or another extract, transform, load (ETL) tool

Data loading formats do not change how data is displayed in tables and charts.

The context where you *can control* date and time formats is data modeling. Data modeling controls how data is displayed in search and their resulting answers.

## Data loading formats via `tsload`

When loading via the `tsload` command you must specify `date` and `timestamp` formats using the format specifications defined in the `strftime` library function. Data is imported based on the timezone of the node from which `tsload` is run.

For `date` data types, the default format is `%Y%m%d` which translates to `yearmonthday`. For example, `Dec 30th, 2001` is represented as `20011230`. For `time` and `datetime` data types, the default is `%Y%m%d %H:%M:%S` which translates to `yearmonthday hour:minute:second`, for example, `Dec 30th, 2001 1:15:12` is represented as `20011230 01:15:12`.

## Data modeling formats for browser data upload

These date and time formats are supported in a CSV file when uploading via the browser. You cannot specify the date format; ThoughtSpot will pick the format that fits your data best:

- 1/30/2014
- 2014-01-30
- 2014-1-30
- 30-Jan-2014
- 2014-Jan-30
- 2014-01-30 10:32 AM
- 2014-01-30 14:52
- 2014-01-30 10:32:22
- 2014-01-30 10:32:22 AM
- 2014-01-30 10:32:22.0
- 2014-01-30 10:32:22.000
- 2014-01-30 10:32:22.000 AM
- 1/30/2014
- 30-Jan-14
- 01-Mar-02 (assumes 2002)
- 30/1/2014 10:32 AM
- 30/1/2014 14:52
- 30/1/2014 10:32:22
- 30/1/2014 10:32:22 AM

- 30/1/2014 10:32:22.0
- 30/1/2014 10:32:22.0 AM
- 30/1/2014 10:32:22.000
- 30/1/2014 10:32:22.000 AM
- 30-Jan-14 10:32 AM
- 30-Jan-14 14:52
- 30-Jan-14 10:32:22
- 30-Jan-14 10:32:22 AM
- 30-Jan-14 10:32:22.0
- 30-Jan-14 10:32:22.0 AM
- 30-Jan-14 10:32:22.000
- 30-Jan-14 10:32:22.000 AM
- Fri Jan 30 2014 3:26 PM
- Fri Jan 30 2014 13:46
- Fri Jan 30 2014 10:32:22
- Fri Jan 30 2014 10:32:22 AM
- Fri Jan 30 2014 10:32:22.0
- Fri Jan 30 2014 10:32:22.0 AM
- Fri Jan 30 2014 10:32:22.000
- Fri Jan 30 2014 10:32:22.000 AM
- 14:52
- 10:32 AM
- 10:32:22
- 10:32:22 AM
- 10:32:22.0
- 10:32:22.000
- 10:32:22.0 AM
- 10:32:22.000 AM

## Data loading formats via data connect or another ETL tool

Data that is loaded via ETL arrives through ODBC or JDBC connection. After you extract the data from the source and before you load it into ThoughtSpot, you must transform any date or timestamp into a valid format for ThoughtSpot. Once transformed, no explicit data masking is required. See the data integration guide for more details of loading data via ODBC and JDBC.

## Data modeling formats

A user with administrative rights can configure data modeling for data on one or all files. You can set number, date, and currency display formats. These formats define how these value types display in tables and charts. See the Admin Guide for more information about data modeling settings. The following format strings are available for use:

| Format mask  | Description                        |
|--------------|------------------------------------|
| YYYY or yyyy | four digit year such as 2017       |
| YY or yy     | last two digits of year such as 17 |
| M            | month with no leading zero 1 - 12  |

| Format mask | Description                                                |
|-------------|------------------------------------------------------------|
| MM          | Two digit month 01 - 12                                    |
| MMM         | Three letter month such as Jan                             |
| D           | Day of year without a leading zero 0 - 365                 |
| DD          | Day of year with up to one leading zero 01 - 365           |
| DDD         | Day of year with up to two leading zeroes 001 - 365        |
| d           | Day of month with no leading zero 1 - 31                   |
| dd          | Two digit day of month 01 - 31                             |
| HH          | Two digit 24 hour representation of hour 00 - 23           |
| hh          | Two digit 12 hour representation of hour 01 - 12           |
| H           | 24 hour representation of hour with no leading zero 0 - 23 |
| h           | 12 hour representation of hour with no leading zero 1 - 12 |
| mm          | Minutes 00 - 59                                            |
| m           | Minutes with no leading zero 0 - 59                        |
| ss          | Seconds 00 - 59                                            |
| s           | Seconds with no leading zero 0 - 59                        |
| a           | AM/PM indicator                                            |

Valid delimiters include most non-alphabet characters. This includes but is not limited to:

- ＼ (forward slash)
- ／ (backward slash)
- | (pipe symbol)
- : (colon)
- (dash)
- \_ (underscore)
- = (equal sign)

Examples of valid format masks you can produce for display are as follows:

- MM/dd/yyyy
- MMM
- DD/MM/yyyy
- MM/dd/yyyy HH:mm
- DD/MM/yyyy HH:mm

# Row level security rules reference

ThoughtSpot allows you to create row level security rules using expressions. If an expression evaluates to “true” for a particular row and group combination, that group will be able to see that row. This reference lists the various operators and functions you can use to create rules.

For information on how to use the row level security functions and operators, see [About Rule-Based Row Level Security](#). There is a special variable called `ts_groups`, which you can use when creating row level security rules. It fetches a list of the groups that the currently logged in user belongs to. For each row, if the expression in the rule evaluates to ‘true’ for any one of these groups, that row will be shown to the user.

You can also see this list of operators and examples from within the Rule Builder by selecting **Rule Assistant**.

## Conversion functions

These functions can be used to convert data from one data type to another. Conversion to or from date data types is not supported.

| Function                | Description                                                                                                                                                                                                                                                                                                | Examples                                                                                |
|-------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------|
| <code>to_bool</code>    | Returns the input as a boolean (true or false).                                                                                                                                                                                                                                                            | <code>to_bool (0) = false</code><br><code>to_bool (married)</code>                      |
| <code>to_date</code>    | Accepts a date represented as an integer or text string, and a second string parameter that can include strftime date formatting elements. Replaces all the valid strftime date formatting elements with their string counterparts and returns the result. Does not accept epoch formatted dates as input. | <code>to_date (date_sold, '%Y-%m-%d')</code>                                            |
| <code>to_double</code>  | Returns the input as a double.                                                                                                                                                                                                                                                                             | <code>to_double ('3.14') = 3.14</code><br><code>to_double (revenue * .01)</code>        |
| <code>to_integer</code> | Returns the input as an integer.                                                                                                                                                                                                                                                                           | <code>to_integer ('45') + 1 = 46</code><br><code>to_integer (price + tax - cost)</code> |
| <code>to_string</code>  | Returns the input as a text string.                                                                                                                                                                                                                                                                        | <code>to_string (45 + 1) = '46'</code><br><code>to_string (revenue - cost)</code>       |

## Date functions

| Function                           | Description                                                                                                        | Examples                                                                                            |
|------------------------------------|--------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------|
| <code>add_days</code>              | Returns the result of adding the specified number of days to the given date.                                       | <code>add_days (01/30/2015, 5) = 02/04/2015</code><br><code>add_days (invoiced, 30)</code>          |
| <code>date</code>                  | Returns the date portion of a given date.                                                                          | <code>date (home visit)</code>                                                                      |
| <code>day</code>                   | Returns the number (1-31) of the day for the given date.                                                           | <code>day (01/15/2014) = 15</code><br><code>day (date ordered)</code>                               |
| <code>day_number_of_week</code>    | Returns the number (1-7) of the day in a week for the given date with 1 being Monday and 7 being Sunday.           | <code>day_number_of_week (01/30/2015) = 6</code><br><code>day_number_of_week (shipped)</code>       |
| <code>day_number_of_quarter</code> | Returns the number of the day in a quarter for a given date.                                                       | <code>day_number_of_quarter (01/30/2015)</code>                                                     |
| <code>day_number_of_week</code>    | Returns the number of the day in a week for a given date.                                                          | <code>day_number_of_week(01/15/2014) &gt; 3</code>                                                  |
| <code>day_number_of_year</code>    | Returns the number (1-366) of the day in a year for the given date.                                                | <code>day_number_of_year (01/30/2015) = 30</code><br><code>day_number_of_year (invoiced)</code>     |
| <code>day_of_week</code>           | Returns the day of the week for the given date.                                                                    | <code>day_of_week (01/30/2015) = Friday</code><br><code>day_of_week (serviced)</code>               |
| <code>diff_days</code>             | Subtracts the second date from the first date and returns the result in number of days, rounded down if not exact. | <code>diff_days (01/15/2014, 01/17/2014) = -2</code><br><code>diff_days (purchased, shipped)</code> |

| Function                             | Description                                                                                                                                                                                                                                                              | Examples                                                                                                 |
|--------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------|
| <code>diff_time</code>               | Subtracts the second date from the first date and returns the result in number of seconds.                                                                                                                                                                               | <code>diff_time (01/01/2014, 01/01/2014) = -86,400</code><br><code>diff_time (clicked, submitted)</code> |
| <code>hour_of_day</code>             | Returns the hour of the day for the given date.                                                                                                                                                                                                                          | <code>hour_of_day (received)</code>                                                                      |
| <code>is_weekend</code>              | Returns true if the given date falls on a Saturday or Sunday.                                                                                                                                                                                                            | <code>is_weekend (01/31/2015) = true</code><br><code>is_weekend (emailed)</code>                         |
| <code>month</code>                   | Returns the month from the given date.                                                                                                                                                                                                                                   | <code>month (01/15/2014) = January</code><br><code>month (date ordered)</code>                           |
| <code>month_number</code>            | Returns the number (1-12) of the month for the given date.                                                                                                                                                                                                               | <code>month_number (09/20/2014) = 9</code><br><code>month_number (purchased)</code>                      |
| <code>month_number_of_quarter</code> | Returns the month (1-12) number for the given date in a quarter.                                                                                                                                                                                                         | <code>month_number_of_quarter (02/20/2018) &gt; 9</code>                                                 |
| <code>now</code>                     | Returns the current timestamp.                                                                                                                                                                                                                                           | <code>now ()</code>                                                                                      |
| <code>start_of_month</code>          | Returns <code>MMM yyyy</code> for the first day of the month. Your installation configuration can override this setting so that it returns a different format such as <code>MM/dd/yyyy</code> . Speak with your ThoughtSpot administrator for information on doing this. | <code>start_of_month (01/31/2015) = Jan FY 2015</code><br><code>start_of_month (shipped)</code>          |
| <code>start_of_quarter</code>        | Returns the date for the first day of the quarter for the given date.                                                                                                                                                                                                    | <code>start_of_quarter (09/18/2015) = Q3 FY 2015</code><br><code>start_of_quarter (sold)</code>          |
| <code>start_of_week</code>           | Returns the date for the first day of the week for the given date.                                                                                                                                                                                                       | <code>start_of_week (06/01/2015) = 05/30/2015 Week</code><br><code>start_of_week (emailed)</code>        |

| Function                            | Description                                                        | Examples                                                                                  |
|-------------------------------------|--------------------------------------------------------------------|-------------------------------------------------------------------------------------------|
| <code>start_of_year</code>          | Returns the date for the first day of the year for the given date. | <code>start_of_year ( 02/15/2015 ) = FY2015</code><br><code>start_of_year (joined)</code> |
| <code>time</code>                   | Returns the time portion of a given date.                          | <code>time (3/1/2002 10:32) = 10:32</code><br><code>time (call began)</code>              |
| <code>week_number_of_month</code>   | Returns the week number for the given date in a month.             | <code>week_number_of_month(03/23/2017) = 3</code>                                         |
| <code>week_number_of_quarter</code> | Returns the week number for the given date in a quarter.           | <code>week_number_of_quarter(04/03/2017)&gt;2</code>                                      |
| <code>week_number_of_year</code>    | Returns the week number for the given date in a year.              | <code>week_number_of_year(04/03/2017) = 20</code>                                         |
| <code>year</code>                   | Returns the year from the given date.                              | <code>year (01/15/2014) = 2014</code><br><code>year (date ordered)</code>                 |

## Mixed functions

These functions can be used with text and numeric data types.

| Function           | Description                                                                   | Examples                                                            |
|--------------------|-------------------------------------------------------------------------------|---------------------------------------------------------------------|
| <code>!=</code>    | Returns true if the first value is not equal to the second value.             | <code>3 != 2 = true</code><br><code>revenue != 1000000</code>       |
| <code>&lt;</code>  | Returns true if the first value is less than the second value.                | <code>3 &lt; 2 = false</code><br><code>revenue &lt; 1000000</code>  |
| <code>&lt;=</code> | Returns true if the first value is less than or equal to the second value.    | <code>1 &lt;= 2 = true</code><br><code>revenue &lt;= 1000000</code> |
| <code>=</code>     | Returns true if the first value is equal to the second value.                 | <code>2 = 2 = true</code><br><code>revenue = 1000000</code>         |
| <code>&gt;</code>  | Returns true if the first value is greater than the second value.             | <code>3 &gt; 2 = true</code><br><code>revenue &gt; 1000000</code>   |
| <code>&gt;=</code> | Returns true if the first value is greater than or equal to the second value. | <code>3 &gt;= 2 = true</code><br><code>revenue &gt;= 1000000</code> |

| Function              | Description                        | Examples                                                                              |
|-----------------------|------------------------------------|---------------------------------------------------------------------------------------|
| <code>greatest</code> | Returns the larger of the values.  | <code>greatest (20, 10) = 20</code><br><code>greatest (q1 revenue, q2 revenue)</code> |
| <code>least</code>    | Returns the smaller of the values. | <code>least (20, 10) = 10</code><br><code>least (q1 revenue, q2 revenue)</code>       |

## Number functions

| Function           | Description                                                         | Examples                                                                     |
|--------------------|---------------------------------------------------------------------|------------------------------------------------------------------------------|
| <code>*</code>     | Returns the result of multiplying both numbers.                     | <code>3 * 2 = 6</code><br><code>price * taxrate</code>                       |
| <code>+</code>     | Returns the result of adding both numbers.                          | <code>1 + 2 = 3</code><br><code>price + shipping</code>                      |
| <code>-</code>     | Returns the result of subtracting the second number from the first. | <code>3 - 2 = 1</code><br><code>revenue - tax</code>                         |
| <code>/</code>     | Returns the result of dividing the first number by the second.      | <code>6 / 3 = 2</code><br><code>markup / retail price</code>                 |
| <code>^</code>     | Returns the first number raised to the power of the second.         | <code>3 ^ 2 = 9</code><br><code>width ^ 2</code>                             |
| <code>abs</code>   | Returns the absolute value.                                         | <code>abs (-10) = 10</code><br><code>abs (profit)</code>                     |
| <code>acos</code>  | Returns the inverse cosine in degrees.                              | <code>acos (0.5) = 60</code><br><code>acos (cos-satellite-angle)</code>      |
| <code>asin</code>  | Returns the inverse sine (specified in degrees).                    | <code>asin (0.5) = 30</code><br><code>asin (sin-satellite-angle)</code>      |
| <code>atan</code>  | Returns the inverse tangent in degrees.                             | <code>atan (1) = 45</code><br><code>atan (tan-satellite-angle)</code>        |
| <code>atan2</code> | Returns the inverse tangent in degrees.                             | <code>atan2 (10, 10) = 45</code><br><code>atan2 (longitude, latitude)</code> |
| <code>cbrt</code>  | Returns the cube root of a number.                                  | <code>cbrt (27) = 3</code><br><code>cbrt (volume)</code>                     |
| <code>ceil</code>  | Returns the smallest following integer.                             | <code>ceil (5.9) = 6</code><br><code>ceil (growth rate)</code>               |
| <code>cos</code>   | Returns the cosine of an angle (specified in degrees).              | <code>cos (63) = 0.45</code><br><code>cos (beam angle)</code>                |

| Function                 | Description                                                                                                                        | Examples                                                                             |
|--------------------------|------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------|
| <code>cube</code>        | Returns the cube of a number.                                                                                                      | <code>cube (3) = 27</code><br><code>cube (length)</code>                             |
| <code>exp</code>         | Returns Euler's number (~2.718) raised to a power.                                                                                 | <code>exp (2) = 7.38905609893</code><br><code>exp (growth)</code>                    |
| <code>exp2</code>        | Returns 2 raised to a power.                                                                                                       | <code>exp2 (3) = 8</code><br><code>exp2 (growth)</code>                              |
| <code>floor</code>       | Returns the largest previous integer.                                                                                              | <code>floor (5.1) = 5</code><br><code>floor (growth rate)</code>                     |
| <code>ln</code>          | Returns the natural logarithm.                                                                                                     | <code>ln (7.38905609893) = 2</code><br><code>ln (distance)</code>                    |
| <code>log10</code>       | Returns the logarithm with base 10.                                                                                                | <code>log10 (100) = 2</code><br><code>log10 (volume)</code>                          |
| <code>log2</code>        | Returns the logarithm with base 2 (binary logarithm).                                                                              | <code>log2 (32) = 5</code><br><code>log2 (volume)</code>                             |
| <code>mod</code>         | Returns the remainder of first number divided by the second number.                                                                | <code>mod (8, 3) = 2</code><br><code>mod (revenue, quantity)</code>                  |
| <code>pow</code>         | Returns the first number raised to the power of the second number.                                                                 | <code>pow (5, 2) = 25</code><br><code>pow (width, 2)</code>                          |
| <code>random</code>      | Returns a random number between 0 and 1.                                                                                           | <code>random () = .457718</code><br><code>random ()</code>                           |
| <code>round</code>       | Returns the first number rounded to the second number (the default is 1).                                                          | <code>round (35.65, 10) = 40</code><br><code>round (battingavg, 100)</code>          |
| <code>safe_divide</code> | Returns the result of dividing the first number by the second. If the second number is 0, returns 0 instead of NaN (not a number). | <code>safe_divide (12, 0) = 0</code><br><code>safe_divide (total_cost, units)</code> |
| <code>sign</code>        | Returns +1 if the number is greater than zero, -1 if less than zero, 0 if zero.                                                    | <code>sign (-250) = -1</code><br><code>sign (growth rate)</code>                     |
| <code>sin</code>         | Returns the sine of an angle (specified in degrees).                                                                               | <code>sin (35) = 0.57</code><br><code>sin (beam angle)</code>                        |

| Function                        | Description                                             | Examples                                                                                                                                                                                                                      |
|---------------------------------|---------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>spherical_distance</code> | Returns the distance in km between two points on Earth. | <code>spherical_distance<br/>(37.465191, -122.153617,<br/>37.421962, -122.142174) =<br/>4,961.96</code><br><code>spherical_distance<br/>(start_latitude,<br/>start_longitude,<br/>start_latitude,<br/>start_longitude)</code> |
| <code>sq</code>                 | Returns the square of a numeric value.                  | <code>sq (9) = 81</code><br><code>sq (width)</code>                                                                                                                                                                           |
| <code>sqrt</code>               | Returns the square root.                                | <code>sqrt (9) = 3</code><br><code>sqrt (area)</code>                                                                                                                                                                         |
| <code>tan</code>                | Returns the tangent of an angle (specified in degrees). | <code>tan (35) = 0.7</code><br><code>tan (beam angle)</code>                                                                                                                                                                  |

## Operators

| Operator                      | Description                                                              | Examples                                                                                                                         |
|-------------------------------|--------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------|
| <code>and</code>              | Returns true when both conditions are true, otherwise returns false.     | <code>(1 = 1) and (3 &gt; 2) = true</code><br><code>lastname = 'smith' and state ='texas'</code>                                 |
|                               |                                                                          | <b>Important:</b> Not available for row level security (RLS) formulas.                                                           |
| <code>if...then...else</code> | Conditional operator.                                                    | <code>if (3 &gt; 2) then 'bigger' else<br/>'not bigger'</code><br><code>if (cost &gt; 500) then 'flag' else<br/>'approve'</code> |
| <code>ifnull</code>           | Returns the first value if it is not null, otherwise returns the second. | <code>ifnull (cost, 'unknown')</code>                                                                                            |
| <code>isnull</code>           | Returns true if the value is null.                                       | <code>isnull (phone)</code>                                                                                                      |
| <code>not</code>              | Returns true if the condition is false, otherwise returns false.         | <code>not (3 &gt; 2) = false</code><br><code>not (state = 'texas')</code>                                                        |
| <code>or</code>               | Returns true when either condition is true, otherwise returns false.     | <code>(1 = 5) or (3 &gt; 2) = true</code><br><code>state = 'california' or state ='oregon'</code>                                |

## Text functions

| Function                            | Description                                                                                                                                                                                                                                                                                                                                                              | Examples                                                                                                                                         |
|-------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>concat</code>                 | Returns the one or more values as a concatenated text string. Be sure to use single quotes instead of double quotes around each of the strings.                                                                                                                                                                                                                          | <code>concat ('hay' , 'stack' ) = 'haystack'</code><br><code>concat (last_name , first_name )</code>                                             |
| <code>contains</code>               | Returns true if the first string contains the second string, otherwise returns false.                                                                                                                                                                                                                                                                                    | <code>contains ('broomstick', 'room') = true</code><br><code>contains (product, 'trial version')</code>                                          |
| <code>edit_distance</code>          | Accepts two text strings. Returns the edit distance (minimum number of operations required to transform one string into the other) as an integer. Works with strings under 1023 characters.                                                                                                                                                                              | <code>edit_distance ('attorney', 'atty') = 4</code><br><code>edit_distance (color, 'red')</code>                                                 |
| <code>edit_distance_with_cap</code> | Accepts two text strings and an integer to specify the upper limit cap for the edit distance (minimum number of operations required to transform one string into the other). If the edit distance is less than or equal to the specified cap, returns the edit distance. If it is higher than the cap, returns the cap plus 1. Works with strings under 1023 characters. | <code>edit_distance_with_cap ('pokemon go', 'minecraft pixelmon', 3) = 4</code><br><code>edit_distance_with_cap (event, 'burning man', 3)</code> |
| <code>similar_to</code>             | Accepts a document text string and a search text string. Returns true if relevance score (0-100) of the search string with respect to the document is greater than or equal to 20. Relevance is based on edit distance, number of words in the query, and length of words in the query which are present in the document.                                                | <code>similar_to ('hello world', 'hello swirl') = true</code><br><code>similar_to (current team, drafted by)</code>                              |

| Function                 | Description                                                                                                                                                                                                                                                                                                                                | Examples                                                                                                                     |
|--------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------|
| <code>similarity</code>  | Accepts a document text string and a search text string. Returns the relevance score (0-100) of the search string with respect to the document. Relevance is based on edit distance, number of words in the query, and length of words in the query which are present in the document. If the two strings are an exact match, returns 100. | <code>similarity ('where is the burning man concert', 'burning man') = 46</code><br><code>similarity (tweet1, tweet2)</code> |
| <code>spells_like</code> | Accepts two text strings. Returns true if they are spelled similarly and false if they are not. Works with strings under 1023 characters.                                                                                                                                                                                                  | <code>spells_like ('thouhgtspot', 'thoughtspot') = true</code><br><code>spells_like (studio, distributor)</code>             |
| <code>strlen</code>      | Returns the length of the text.                                                                                                                                                                                                                                                                                                            | <code>strlen ('smith') = 5</code><br><code>strlen (lastname)</code>                                                          |
| <code>strpos</code>      | Returns the numeric position (starting from 0) of the first occurrence of the second string in the first string, or -1 if not found.                                                                                                                                                                                                       | <code>strpos ('haystack_with_needles', 'needle') = 14</code><br><code>strpos (complaint, 'lawyer')</code>                    |
| <code>substr</code>      | Returns the portion of the given string, beginning at the location specified (starting from 0), and of the given length.                                                                                                                                                                                                                   | <code>substr ('persnickety', 3, 7) = snicket</code><br><code>substr (lastname, 0, 5)</code>                                  |

## Variables

These variables can be used in your expressions.

| Function                 | Description                                                                                                                                                                | Examples                           |
|--------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------|
| <code>ts_groups</code>   | Returns a list of all the groups the current logged in user belongs to. For any row, if the expression evaluates to true for any of the groups, the user can see that row. | <code>ts_groups = 'east'</code>    |
| <code>ts_username</code> | Returns the user with the matching neame.                                                                                                                                  | <code>ts_username != 'mark'</code> |

# Formula function reference

ThoughtSpot allows you to create derived columns in worksheets using formulas. You create these columns by building formulas using the **Formula Assistant**. An individual formula is constructed from  $n$  combination of operators and functions.

This reference lists the various operators and functions you can use to create formulas.

## Operators

| Operator                      | Description                                                              | Examples                                                                                                                 |
|-------------------------------|--------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------|
| <code>and</code>              | Returns true when both conditions are true, otherwise returns false.     | <code>(1 = 1) and (3 &gt; 2) = true</code><br><code>lastname = 'smith' and state = 'texas'</code>                        |
|                               |                                                                          | <b>Note:</b> Not available for row level security (RLS) formulas.                                                        |
| <code>if...then...else</code> | Conditional operator.                                                    | <code>if (3 &gt; 2) then 'bigger' else 'not bigger'</code><br><code>if (cost &gt; 500) then 'flag' else 'approve'</code> |
| <code>ifnull</code>           | Returns the first value if it is not null, otherwise returns the second. | <code>ifnull (cost, 'unknown')</code>                                                                                    |
| <code>isnull</code>           | Returns true if the value is null.                                       | <code>isnull (phone)</code>                                                                                              |
| <code>not</code>              | Returns true if the condition is false, otherwise returns false.         | <code>not (3 &gt; 2) = false</code><br><code>not (state = 'texas')</code>                                                |
| <code>or</code>               | Returns true when either condition is true, otherwise returns false.     | <code>(1 = 5) or (3 &gt; 2) = true</code><br><code>state = 'california' or state = 'oregon'</code>                       |

## Aggregate functions

These functions can be used to aggregate data.

| Function             | Description                                        | Examples                       |
|----------------------|----------------------------------------------------|--------------------------------|
| <code>average</code> | Returns the average of all the values of a column. | <code>average (revenue)</code> |

| Function                        | Description                                                                                                                             | Examples                                                     |
|---------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------|
| <code>average_if</code>         | Returns the average of all the columns that meet a given criteria.                                                                      | <code>average_if(city = "San Francisco", revenue)</code>     |
| <code>count</code>              | Returns the number of rows in the table containing the column.                                                                          | <code>count (product)</code>                                 |
| <code>count_if</code>           | Returns the number of rows in the table containing the column.                                                                          | <code>count_if(region = 'west', region)</code>               |
| <code>cumulative_average</code> | Takes a measure and one or more attributes. Returns the average of the measure, accumulated by the attribute(s) in the order specified. | <code>cumulative_average (revenue, order date, state)</code> |
| <code>cumulative_max</code>     | Takes a measure and one or more attributes. Returns the maximum of the measure, accumulated by the attribute(s) in the order specified. | <code>cumulative_max (revenue, state)</code>                 |
| <code>cumulative_min</code>     | Takes a measure and one or more attributes. Returns the minimum of the measure, accumulated by the attribute(s) in the order specified. | <code>cumulative_min (revenue, campaign)</code>              |
| <code>cumulative_sum</code>     | Takes a measure and one or more attributes. Returns the sum of the measure, accumulated by the attribute(s) in the order specified.     | <code>cumulative_sum (revenue, order date)</code>            |
| <code>group_average</code>      | Takes a measure and one or more attributes. Returns the average of the measure grouped by the attribute(s).                             | <code>group_average (revenue, customer region, state)</code> |
| <code>group_count</code>        | Takes a measure and one or more attributes. Returns the count of the measure grouped by the attribute(s).                               | <code>group_count (revenue, customer region)</code>          |
| <code>group_max</code>          | Takes a measure and one or more attributes. Returns the maximum of the measure grouped by the attribute(s).                             | <code>group_max (revenue, customer region)</code>            |
| <code>group_min</code>          | Takes a measure and one or more attributes. Returns the minimum of the measure grouped by the attribute(s).                             | <code>group_min (revenue, customer region)</code>            |
| <code>group_stddev</code>       | Takes a measure and one or more attributes. Returns the standard deviation of the measure grouped by the attribute(s).                  | <code>group_stddev (revenue, customer region)</code>         |

| Function                        | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                    | Examples                                                     |
|---------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------|
| <code>group_sum</code>          | Takes a measure and one or more attributes. Returns the sum of the measure grouped by the attribute(s).                                                                                                                                                                                                                                                                                                                                                                        | <code>group_sum (revenue, customer region)</code>            |
| <code>group_unique_count</code> | Takes a measure and one or more attributes. Returns the unique count of the measure grouped by the attribute(s).                                                                                                                                                                                                                                                                                                                                                               | <code>group_unique_count (product , supplier)</code>         |
| <code>group_variance</code>     | Takes a measure and one or more attributes. Returns the variance of the measure grouped by the attribute(s).                                                                                                                                                                                                                                                                                                                                                                   | <code>group_variance (revenue, customer region)</code>       |
| <code>max</code>                | Returns the maximum value of a column.                                                                                                                                                                                                                                                                                                                                                                                                                                         | <code>max (sales)</code>                                     |
| <code>max_if</code>             | Returns the maximum value among columns that meet a criteria.                                                                                                                                                                                                                                                                                                                                                                                                                  | <code>max_if( (revenue &gt; 10) , customer region )</code>   |
| <code>min</code>                | Returns the minimum value of a column.                                                                                                                                                                                                                                                                                                                                                                                                                                         | <code>min (revenue)</code>                                   |
| <code>min_if</code>             | Returns the minimum value among columns that meet a criteria.                                                                                                                                                                                                                                                                                                                                                                                                                  | <code>min_if( (revenue &lt; 10) , customer region )</code>   |
| <code>moving_average</code>     | Takes a measure, two integers to define the window to aggregate over, and one or more attributes. The window is (current - Num1...Current + Num2) with both end points being included in the window. For example, “1,1” will have a window size of 3. To define a window that begins before Current, specify a negative number for Num2. Returns the average of the measure over the given window. The attributes are the ordering columns used to compute the moving average. | <code>moving_average (revenue, 2, 1, customer region)</code> |
| <code>moving_max</code>         | Takes a measure, two integers to define the window to aggregate over, and one or more attributes. The window is (current - Num1...Current + Num2) with both end points being included in the window. For example, “1,1” will have a window size of 3. To define a window that begins before Current, specify a negative number for Num2. Returns the maximum of the measure over the given window. The attributes are the ordering columns used to compute the moving maximum. | <code>moving_max (complaints, 1, 2, store name)</code>       |

| Function                     | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                    | Examples                                                       |
|------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------|
| <code>moving_min</code>      | Takes a measure, two integers to define the window to aggregate over, and one or more attributes. The window is (current - Num1...Current + Num2) with both end points being included in the window. For example, “1,1” will have a window size of 3. To define a window that begins before Current, specify a negative number for Num2. Returns the minimum of the measure over the given window. The attributes are the ordering columns used to compute the moving minimum. | <code>moving_min (defects, 3, 1, product)</code>               |
| <code>moving_sum</code>      | Takes a measure, two integers to define the window to aggregate over, and one or more attributes. The window is (current - Num1...Current + Num2) with both end points being included in the window. For example, “1,1” will have a window size of 3. To define a window that begins before Current, specify a negative number for Num2. Returns the sum of the measure over the given window. The attributes are the ordering columns used to compute the moving sum.         | <code>moving_sum (revenue, 1, 1, order date)</code>            |
| <code>stddev</code>          | Returns the standard deviation of all values of a column.                                                                                                                                                                                                                                                                                                                                                                                                                      | <code>stddev (revenue)</code>                                  |
| <code>stddev_if</code>       | Returns a standard deviation values filtered to meet a specific criteria.                                                                                                                                                                                                                                                                                                                                                                                                      | <code>stddev_if( (revenue &gt; 10) , (revenue/10.0) )</code>   |
| <code>sum</code>             | Returns the sum of all the values of a column.                                                                                                                                                                                                                                                                                                                                                                                                                                 | <code>sum (revenue)</code>                                     |
| <code>sum_if</code>          | Returns sum values filtered by a specific criteria.                                                                                                                                                                                                                                                                                                                                                                                                                            | <code>sum_if(region='west', revenue)</code>                    |
| <code>unique_count</code>    | Returns the number of unique values of a column.                                                                                                                                                                                                                                                                                                                                                                                                                               | <code>unique_count (customer)</code>                           |
| <code>unique_count_if</code> | Returns the number of unique values of a column provided it meets a criteria.                                                                                                                                                                                                                                                                                                                                                                                                  | <code>unique_count_if( (revenue &gt; 10) , order date )</code> |
| <code>variance</code>        | Returns the variance of all the values of a column.                                                                                                                                                                                                                                                                                                                                                                                                                            | <code>variance (revenue)</code>                                |
| <code>variance_if</code>     | Returns the variance of all the values of a column provided it meets a criteria..                                                                                                                                                                                                                                                                                                                                                                                              | <code>variance_if( (revenue &gt; 10) , (revenue/10.0) )</code> |

## Conversion functions

These functions can be used to convert data from one data type to another. Conversion to or from date data types is not supported.

| Function                | Description                                                                                                                                                                                                                                                                                                | Examples                                                                                |
|-------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------|
| <code>to_bool</code>    | Returns the input as a boolean (true or false).                                                                                                                                                                                                                                                            | <code>to_bool (0) = false</code><br><code>to_bool (married)</code>                      |
| <code>to_date</code>    | Accepts a date represented as an integer or text string, and a second string parameter that can include strftime date formatting elements. Replaces all the valid strftime date formatting elements with their string counterparts and returns the result. Does not accept epoch formatted dates as input. | <code>to_date (date_sold, '%Y-%m-%d')</code>                                            |
| <code>to_double</code>  | Returns the input as a double.                                                                                                                                                                                                                                                                             | <code>to_double ('3.14') = 3.14</code><br><code>to_double (revenue * .01)</code>        |
| <code>to_integer</code> | Returns the input as an integer.                                                                                                                                                                                                                                                                           | <code>to_integer ('45') + 1 = 46</code><br><code>to_integer (price + tax - cost)</code> |
| <code>to_string</code>  | Returns the input as a text string.                                                                                                                                                                                                                                                                        | <code>to_string (45 + 1) = '46'</code><br><code>to_string (revenue - cost)</code>       |

## Date functions

| Function              | Description                                                                  | Examples                                                                                   |
|-----------------------|------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------|
| <code>add_days</code> | Returns the result of adding the specified number of days to the given date. | <code>add_days (01/30/2015, 5) = 02/04/2015</code><br><code>add_days (invoiced, 30)</code> |
| <code>date</code>     | Returns the date portion of a given date.                                    | <code>date (home visit)</code>                                                             |
| <code>day</code>      | Returns the number (1-31) of the day for the given date.                     | <code>day (01/15/2014) = 15</code><br><code>day (date ordered)</code>                      |

| Function                           | Description                                                                                                        | Examples                                                                                                 |
|------------------------------------|--------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------|
| <code>day_number_of_week</code>    | Returns the number (1-7) of the day in a week for the given date with 1 being Monday and 7 being Sunday.           | <code>day_number_of_week (01/30/2015) = 6</code><br><code>day_number_of_week (shipped)</code>            |
| <code>day_number_of_quarter</code> | Returns the number of the day in a quarter for a given date.                                                       | <code>day_number_of_quarter (01/30/2015)</code>                                                          |
| <code>day_number_of_week</code>    | Returns the number of the day in a week for a given date.                                                          | <code>day_number_of_week(01/15/2014) &gt; 3</code>                                                       |
| <code>day_number_of_year</code>    | Returns the number (1-366) of the day in a year for the given date.                                                | <code>day_number_of_year (01/30/2015) = 30</code><br><code>day_number_of_year (invoiced)</code>          |
| <code>day_of_week</code>           | Returns the day of the week for the given date.                                                                    | <code>day_of_week (01/30/2015) = Friday</code><br><code>day_of_week (serviced)</code>                    |
| <code>diff_days</code>             | Subtracts the second date from the first date and returns the result in number of days, rounded down if not exact. | <code>diff_days (01/15/2014, 01/17/2014) = -2</code><br><code>diff_days (purchased, shipped)</code>      |
| <code>diff_time</code>             | Subtracts the second date from the first date and returns the result in number of seconds.                         | <code>diff_time (01/01/2014, 01/01/2014) = -86,400</code><br><code>diff_time (clicked, submitted)</code> |
| <code>hour_of_day</code>           | Returns the hour of the day for the given date.                                                                    | <code>hour_of_day (received)</code>                                                                      |
| <code>is_weekend</code>            | Returns true if the given date falls on a Saturday or Sunday.                                                      | <code>is_weekend (01/31/2015) = true</code><br><code>is_weekend (emailed)</code>                         |
| <code>month</code>                 | Returns the month from the given date.                                                                             | <code>month (01/15/2014) = January</code><br><code>month (date ordered)</code>                           |
| <code>month_number</code>          | Returns the number (1-12) of the month for the given date.                                                         | <code>month_number (09/20/2014) = 9</code><br><code>month_number (purchased)</code>                      |

| Function                             | Description                                                                                                                                                                                                                                                              | Examples                                                                                            |
|--------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------|
| <code>month_number_of_quarter</code> | Returns the month (1-12) number for the given date in a quarter.                                                                                                                                                                                                         | <code>month_number_of_quarter (02/20/2018) &gt; 9</code>                                            |
| <code>now</code>                     | Returns the current timestamp.                                                                                                                                                                                                                                           | <code>now ()</code>                                                                                 |
| <code>start_of_month</code>          | Returns <code>MMM yyyy</code> for the first day of the month. Your installation configuration can override this setting so that it returns a different format such as <code>MM/dd/yyyy</code> . Speak with your ThoughtSpot administrator for information on doing this. | <code>start_of_month ( 01/31/2015 ) = Jan FY 2015</code><br><code>start_of_month (shipped)</code>   |
| <code>start_of_quarter</code>        | Returns the date for the first day of the quarter for the given date.                                                                                                                                                                                                    | <code>start_of_quarter ( 09/18/2015 ) = Q3 FY 2015</code><br><code>start_of_quarter (sold)</code>   |
| <code>start_of_week</code>           | Returns the date for the first day of the week for the given date.                                                                                                                                                                                                       | <code>start_of_week ( 06/01/2015 ) = 05/30/2015 Week</code><br><code>start_of_week (emailed)</code> |
| <code>start_of_year</code>           | Returns the date for the first day of the year for the given date.                                                                                                                                                                                                       | <code>start_of_year ( 02/15/2015 ) = FY 2015</code><br><code>start_of_year (joined)</code>          |
| <code>time</code>                    | Returns the time portion of a given date.                                                                                                                                                                                                                                | <code>time (3/1/2002 10:32) = 10:32</code><br><code>time (call began)</code>                        |
| <code>week_number_of_month</code>    | Returns the week number for the given date in a month.                                                                                                                                                                                                                   | <code>week_number_of_month(03/23/2017) = 3</code>                                                   |
| <code>week_number_of_quarter</code>  | Returns the week number for the given date in a quarter.                                                                                                                                                                                                                 | <code>week_number_of_quarter(04/03/2017)&gt;2</code>                                                |
| <code>week_number_of_year</code>     | Returns the week number for the given date in a year.                                                                                                                                                                                                                    | <code>week_number_of_year(04/03/2017) = 20</code>                                                   |

| Function          | Description                           | Examples                                                                  |
|-------------------|---------------------------------------|---------------------------------------------------------------------------|
| <code>year</code> | Returns the year from the given date. | <code>year (01/15/2014) = 2014</code><br><code>year (date ordered)</code> |

## Mixed functions

These functions can be used with text and numeric data types.

| Function              | Description                                                                   | Examples                                                                              |
|-----------------------|-------------------------------------------------------------------------------|---------------------------------------------------------------------------------------|
| <code>!=</code>       | Returns true if the first value is not equal to the second value.             | <code>3 != 2 = true</code><br><code>revenue != 1000000</code>                         |
| <code>&lt;</code>     | Returns true if the first value is less than the second value.                | <code>3 &lt; 2 = false</code><br><code>revenue &lt; 1000000</code>                    |
| <code>&lt;=</code>    | Returns true if the first value is less than or equal to the second value.    | <code>1 &lt;= 2 = true</code><br><code>revenue &lt;= 1000000</code>                   |
| <code>=</code>        | Returns true if the first value is equal to the second value.                 | <code>2 = 2 = true</code><br><code>revenue = 1000000</code>                           |
| <code>&gt;</code>     | Returns true if the first value is greater than the second value.             | <code>3 &gt; 2 = true</code><br><code>revenue &gt; 1000000</code>                     |
| <code>&gt;=</code>    | Returns true if the first value is greater than or equal to the second value. | <code>3 &gt;= 2 = true</code><br><code>revenue &gt;= 1000000</code>                   |
| <code>greatest</code> | Returns the larger of the values.                                             | <code>greatest (20, 10) = 20</code><br><code>greatest (q1 revenue, q2 revenue)</code> |
| <code>least</code>    | Returns the smaller of the values.                                            | <code>least (20, 10) = 10</code><br><code>least (q1 revenue, q2 revenue)</code>       |

## Number functions

| Function       | Description                                                         | Examples                                                |
|----------------|---------------------------------------------------------------------|---------------------------------------------------------|
| <code>*</code> | Returns the result of multiplying both numbers.                     | <code>3 * 2 = 6</code><br><code>price * taxrate</code>  |
| <code>+</code> | Returns the result of adding both numbers.                          | <code>1 + 2 = 3</code><br><code>price + shipping</code> |
| <code>-</code> | Returns the result of subtracting the second number from the first. | <code>3 - 2 = 1</code><br><code>revenue - tax</code>    |

| Function | Description                                                    | Examples                                                                     |
|----------|----------------------------------------------------------------|------------------------------------------------------------------------------|
| /        | Returns the result of dividing the first number by the second. | <code>6 / 3 = 2</code><br><code>markup / retail price</code>                 |
| ^        | Returns the first number raised to the power of the second.    | <code>3 ^ 2 = 9</code><br><code>width ^ 2</code>                             |
| abs      | Returns the absolute value.                                    | <code>abs (-10) = 10</code><br><code>abs (profit)</code>                     |
| acos     | Returns the inverse cosine in degrees.                         | <code>acos (0.5) = 60</code><br><code>acos (cos-satellite-angle)</code>      |
| asin     | Returns the inverse sine (specified in degrees).               | <code>asin (0.5) = 30</code><br><code>asin (sin-satellite-angle)</code>      |
| atan     | Returns the inverse tangent in degrees.                        | <code>atan (1) = 45</code><br><code>atan (tan-satellite-angle)</code>        |
| atan2    | Returns the inverse tangent in degrees.                        | <code>atan2 (10, 10) = 45</code><br><code>atan2 (longitude, latitude)</code> |
| cbrt     | Returns the cube root of a number.                             | <code>cbrt (27) = 3</code><br><code>cbrt (volume)</code>                     |
| ceil     | Returns the smallest following integer.                        | <code>ceil (5.9) = 6</code><br><code>ceil (growth rate)</code>               |
| cos      | Returns the cosine of an angle (specified in degrees).         | <code>cos (63) = 0.45</code><br><code>cos (beam angle)</code>                |
| cube     | Returns the cube of a number.                                  | <code>cube (3) = 27</code><br><code>cube (length)</code>                     |
| exp      | Returns Euler's number (~2.718) raised to a power.             | <code>exp (2) = 7.38905609893</code><br><code>exp (growth)</code>            |
| exp2     | Returns 2 raised to a power.                                   | <code>exp2 (3) = 8</code><br><code>exp2 (growth)</code>                      |
| floor    | Returns the largest previous integer.                          | <code>floor (5.1) = 5</code><br><code>floor (growth rate)</code>             |
| ln       | Returns the natural logarithm.                                 | <code>ln (7.38905609893) = 2</code><br><code>ln (distance)</code>            |
| log10    | Returns the logarithm with base 10.                            | <code>log10 (100) = 2</code><br><code>log10 (volume)</code>                  |
| log2     | Returns the logarithm with base 2 (binary logarithm).          | <code>log2 (32) = 5</code><br><code>log2 (volume)</code>                     |

| Function                        | Description                                                                                                                        | Examples                                                                                                                                                                                          |
|---------------------------------|------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>mod</code>                | Returns the remainder of first number divided by the second number.                                                                | <code>mod (8, 3) = 2</code><br><code>mod ( revenue , quantity )</code>                                                                                                                            |
| <code>pow</code>                | Returns the first number raised to the power of the second number.                                                                 | <code>pow (5, 2) = 25</code><br><code>pow (width, 2)</code>                                                                                                                                       |
| <code>random</code>             | Returns a random number between 0 and 1.                                                                                           | <code>random ( ) = .457718</code><br><code>random ( )</code>                                                                                                                                      |
| <code>round</code>              | Returns the first number rounded to the second number (the default is 1).                                                          | <code>round (35.65, 10) = 40</code><br><code>round (battingavg, 100)</code>                                                                                                                       |
| <code>safe_divide</code>        | Returns the result of dividing the first number by the second. If the second number is 0, returns 0 instead of NaN (not a number). | <code>safe_divide (12, 0) = 0</code><br><code>safe_divide (total_cost, units)</code>                                                                                                              |
| <code>sign</code>               | Returns +1 if the number is greater than zero, -1 if less than zero, 0 if zero.                                                    | <code>sign (-250) = -1</code><br><code>sign (growth rate)</code>                                                                                                                                  |
| <code>sin</code>                | Returns the sine of an angle (specified in degrees).                                                                               | <code>sin (35) = 0.57</code><br><code>sin (beam angle)</code>                                                                                                                                     |
| <code>spherical_distance</code> | Returns the distance in km between two points on Earth.                                                                            | <code>spherical_distance (37.465191, -122.153617, 37.421962, -122.142174) = 4,961.96</code><br><code>spherical_distance (start_latitude, start_longitude, start_latitude, start_longitude)</code> |
| <code>sq</code>                 | Returns the square of a numeric value.                                                                                             | <code>sq (9) = 81</code><br><code>sq (width)</code>                                                                                                                                               |
| <code>sqrt</code>               | Returns the square root.                                                                                                           | <code>sqrt (9) = 3</code><br><code>sqrt (area)</code>                                                                                                                                             |
| <code>tan</code>                | Returns the tangent of an angle (specified in degrees).                                                                            | <code>tan (35) = 0.7</code><br><code>tan (beam angle)</code>                                                                                                                                      |

## Text functions

| Function                            | Description                                                                                                                                                                                                                                                                                                                                                              | Examples                                                                                                                                         |
|-------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>concat</code>                 | Returns the one or more values as a concatenated text string. Be sure to use single quotes instead of double quotes around each of the strings.                                                                                                                                                                                                                          | <code>concat ('hay' , 'stack' ) = 'haystack'</code><br><code>concat (last_name , first_name )</code>                                             |
| <code>contains</code>               | Returns true if the first string contains the second string, otherwise returns false.                                                                                                                                                                                                                                                                                    | <code>contains ('broomstick', 'room') = true</code><br><code>contains (product, 'trial version')</code>                                          |
| <code>edit_distance</code>          | Accepts two text strings. Returns the edit distance (minimum number of operations required to transform one string into the other) as an integer. Works with strings under 1023 characters.                                                                                                                                                                              | <code>edit_distance ('attorney', 'atty') = 4</code><br><code>edit_distance (color, 'red')</code>                                                 |
| <code>edit_distance_with_cap</code> | Accepts two text strings and an integer to specify the upper limit cap for the edit distance (minimum number of operations required to transform one string into the other). If the edit distance is less than or equal to the specified cap, returns the edit distance. If it is higher than the cap, returns the cap plus 1. Works with strings under 1023 characters. | <code>edit_distance_with_cap ('pokemon go', 'minecraft pixelmon', 3) = 4</code><br><code>edit_distance_with_cap (event, 'burning man', 3)</code> |
| <code>similar_to</code>             | Accepts a document text string and a search text string. Returns true if relevance score (0-100) of the search string with respect to the document is greater than or equal to 20. Relevance is based on edit distance, number of words in the query, and length of words in the query which are present in the document.                                                | <code>similar_to ('hello world', 'hello swirl') = true</code><br><code>similar_to (current team, drafted by)</code>                              |

| Function                 | Description                                                                                                                                                                                                                                                                                                                                | Examples                                                                                                                     |
|--------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------|
| <code>similarity</code>  | Accepts a document text string and a search text string. Returns the relevance score (0-100) of the search string with respect to the document. Relevance is based on edit distance, number of words in the query, and length of words in the query which are present in the document. If the two strings are an exact match, returns 100. | <code>similarity ('where is the burning man concert', 'burning man') = 46</code><br><code>similarity (tweet1, tweet2)</code> |
| <code>spells_like</code> | Accepts two text strings. Returns true if they are spelled similarly and false if they are not. Works with strings under 1023 characters.                                                                                                                                                                                                  | <code>spells_like ('thouhgtspot', 'thoughtspot') = true</code><br><code>spells_like (studio, distributor)</code>             |
| <code>strlen</code>      | Returns the length of the text.                                                                                                                                                                                                                                                                                                            | <code>strlen ('smith') = 5</code><br><code>strlen (lastname)</code>                                                          |
| <code>strpos</code>      | Returns the numeric position (starting from 0) of the first occurrence of the second string in the first string, or -1 if not found.                                                                                                                                                                                                       | <code>strpos ('haystack_with_needles', 'needle') = 14</code><br><code>strpos (complaint, 'lawyer')</code>                    |
| <code>substr</code>      | Returns the portion of the given string, beginning at the location specified (starting from 0), and of the given length.                                                                                                                                                                                                                   | <code>substr ('persnickety', 3, 7) = snicket</code><br><code>substr (lastname, 0, 5)</code>                                  |

# Alerts code reference

This reference identifies the messages that can appear in the **System Health > Overview > Critical Alerts** and in the **Alerts** dashboard.

## Informational alerts

### TASK\_TERMINATED

**Msg:** Task {{.Service}}.{{.Task}} terminated on machine {{.Machine}}

**Type:** INFO

This alert is raised when a task terminates.

### DISK\_ERROR

**Msg:** Machine {{.Machine}} has disk errors

**Type:** INFO

Raised when a machine has disk errors.

### ZK\_AVG\_LATENCY

**Msg:** Average Zookeeper latency is more than {{.Num}} msec

**Type:** INFO

Raised when average Zookeeper latency is above a threshold.

### ZK\_MAX\_LATENCY

**Msg:** Max Zookeeper latency is more than {{.Num}} msec

**Type:** INFO

Raised when max Zookeeper latency is above a threshold.

### ZK\_MIN\_LATENCY

**Msg:** Min Zookeeper latency is more than {{.Num}} msec

**Type:** INFO

Raised when min Zookeeper latency is above a threshold.

### ZK\_OUTSTANDING\_REQUESTS

**Msg:** Number of outstanding Zookeeper requests exceeds {{.Num}}

**Type:** INFO

Raised when there are too many outstanding Zookeeper requests.

## ZK\_NUM\_WATCHERS

**Msg:** Number of Zookeeper watchers exceeds {{.Num}}

**Type:** INFO

Raised when there are too many Zookeeper watchers.

## MASTER\_ELECTION

**Msg:** {{.Machine}} elected as Orion Master

**Type:** INFO

Raised when a new Orion Master is elected.

## PERIODIC\_BACKUP

**Msg:** {{.Process}} periodic backup for policy {{.Name}} failed.

**Type:** INFO

Raised when periodic backup fails.

## PERIODIC\_SNAPSHOT

**Msg:** {{.Process}} periodic snapshot {{.Name}} failed.

**Type:** INFO

Raised when a periodic snapshot fails.

## HDFS\_CORRUPTION

**Msg:** HDFS root directory is in a corrupted state.

**Type:** INFO

Raised when HDFS root directory is corrupted.

## APPLICATION\_INVALID\_STATE

**Msg:** {{.Service}}.{{.Task}} on {{.Machine}} at location {{.Location}}

**Type:** INFO

Raised when Application raises invalid state alert.

## UPDATE\_START

**Msg:** Starting update of ThoughtSpot cluster {{.Cluster}}

**Type:** INFO

Raised when update starts.

## UPDATE-END

**Msg:** Finished update of ThoughtSpot cluster {{.Cluster}} to release {{.Release}}

**Type:** INFO

Raised when update completes.

## Errors

### TIMELY\_JOB\_RUN\_ERROR

**Msg:** Job run {{.Message}}

**Type:** ERROR

Raised when a job run fails.

### TIMELY\_ERROR

**Msg:** Job manager {{.Message}}

**Type:** ERROR

Raised when a job manager runs into an inconsistent state.

## Warnings

### DISK\_SPACE

**Msg:** Machine {{.Machine}} has less than {{.Perc}}% disk space free

**Type:** WARNING

Raised when a disk is low on available disk space. Valid only in the 3.2 version of ThoughtSpot.

### ROOT\_DISK\_SPACE

**Msg:** Machine {{.Machine}} has less than {{.Perc}}% disk space free on root partition

**Type:** WARNING

Raised when a machine is low on available disk space on root partition.

### BOOT\_DISK\_SPACE

**Msg:** Machine {{.Machine}} has less than {{.Perc}}% disk space free on boot partition

**Type:** WARNING

Raised when a machine is low on available disk space on boot partition.

## UPDATE\_DISK\_SPACE

**Msg:** Machine {{.Machine}} has less than {{.Perc}}% disk space free on update partition

**Type:** WARNING

Raised when a machine is low on available disk space on update partition.

## EXPORT\_DISK\_SPACE

**Msg:** Machine {{.Machine}} has less than {{.Perc}}% disk space free on export partition

**Type:** WARNING

Raised when a machine is low on available disk space on export partition.

## HDFS\_NAMENODE\_DISK\_SPACE

**Msg:** Machine {{.Machine}} has less than {{.Perc}}% disk space free on HDFS namenode drive

**Type:** WARNING

Raised when a machine is low on available disk space on HDFS namenode drive.

## MEMORY

**Msg:** Machine {{.Machine}} has less than {{.Perc}}% memory free

**Type:** WARNING

Raised when a machine is low on free memory.

## OS\_USERS

**Msg:** Machine {{.Machine}} has more than {{.Num}} logged in users

**Type:** WARNING

Raised when a machine has too many users logged in.

## OS\_PROCS

**Msg:** Machine {{.Machine}} has more than {{.Num}} processes

**Type:** WARNING

Raised when a machine has more too many processes.

## SSH

**Msg:** Machine {{.Machine}} doesn't have an active SSH server

**Type:** WARNING

Raised when a machine has more than 600 processes.

**DISK\_ERROR\_EXTERNAL**

**Msg:** Machine {{.Machine}} has disk errors

**Type:** WARNING

Raised when more than 2 disk errors happen in a day.

**ZK\_FD\_COUNT**

**Msg:** Zookeeper has more than {{.Num}} open file descriptors

**Type:** WARNING

Raised when there are too many open Zookeeper files.

**ZK\_EPHEMERAL\_COUNT**

**Msg:** Zookeeper has more than {{.Num}} ephemeral files

**Type:** WARNING

Raised when there are too many Zookeeper ephemeral files.

**HOST\_DOWN**

**Msg:** {{.Machine}} is down

**Type:** WARNING

Raised when a host is down.

**TASK\_UNREACHABLE**

**Msg:** {{.ServiceDesc}} on {{.Machine}} is unreachable over HTTP

**Type:** WARNING

Raised when a task is unreachable over HTTP.

**TASK\_NOT\_RUNNING**

**Msg:** {{.ServiceDesc}} is not running

**Type:** WARNING

Raised when a service task is not running on any machine in the cluster.

# Critical alerts

## TASK\_FLAPPING

**Msg:** Task {{.Service}}.{{.Task}} terminated {{.\_actual\_num\_occurrences}} times in last {{.\_earliest\_duration\_str}}

**Type:** CRITICAL

This alert is raised when a task is crashing repeatedly. The service is evaluated across the whole cluster. So, if a service crashes 5 times in a day across all nodes in the cluster, this alert is generated.

## OREO\_TERMINATED

**Msg:** Oreo terminated on machine {{.Machine}}

**Type:** CRITICAL

This alert is raised when the Oreo daemon on a machine terminates due to an error. This typically happens due to an error accessing Zookeeper, HDFS, or a hardware issue.

## HDFS\_DISK\_SPACE

**Msg:** HDFS has less than {{.Perc}}% space free

**Type:** CRITICAL

Raised when a HDFS cluster is low on total available disk space.

## ZK\_INACCESSIBLE

**Msg:** Zookeeper is not accessible

**Type:** CRITICAL

Raised when Zookeeper is inaccessible.

## PERIODIC\_BACKUP\_FLAPPING

**Msg:** Periodic backup failed {{.\_actual\_num\_occurrences}} times in last {{.\_earliest\_duration\_str}}

**Type:** CRITICAL

This alert is raised when a periodic backup failed repeatedly.

## PERIODIC\_SNAPSHOT\_FLAPPING

**Msg:** Periodic snapshot failed {{.\_actual\_num\_occurrences}} times in last {{.\_earliest\_duration\_str}}

**Type:** CRITICAL

This alert is raised when periodic snapshot failed repeatedly.

### APPLICATION\_INVALID\_STATE\_EXTERNAL

**Msg:** {{.Service}}.{{.Task}} on {{.Machine}} at location {{.Location}}

**Type:** CRITICAL

Raised when Application raises invalid state alert.

# User action code reference

This reference identifies the user action codes that can appear in the **System Health** pages and in logs or other reports.

|                                               |                                                                                                            |
|-----------------------------------------------|------------------------------------------------------------------------------------------------------------|
| <code>answer_unsaved</code>                   | User makes a change to tokens in the search bar.                                                           |
| <code>answer_saved</code>                     | User opens an existing saved answer and makes changes to tokens in the search bar.                         |
| <code>answer_pinboard_context</code>          | User opens an existing saved pinboard, edits a context viz and makes a change to tokens in the search bar. |
| <code>answer_aggregated_worksheet</code>      | User opens an existing saved aggregated worksheet and makes changes to tokens in the search bar.           |
| <code>answer_upgrade</code>                   | Requests made for the sole purpose of upgrade.                                                             |
| <code>pinboard_view</code>                    | User opens an existing saved pinboard.                                                                     |
| <code>pinboard_filter</code>                  | User adds, removes or applies values to a pinboard filter.                                                 |
| <code>pinboard_ad_hoc</code>                  | User drills down in a pinboard viz.                                                                        |
| <code>data_chart_config</code>                | Request for new data being generated following a chart config change.                                      |
| <code>data_show_underlying_row</code>         | Request to show underlying data for a data row(s).                                                         |
| <code>data_export</code>                      | Request to export data.                                                                                    |
| <code>pinboard_tspublic_runtime_filter</code> | Request to TSPublic/pinboarddata with runtime filters.                                                     |
| <code>answer_aggregated_worksheet_save</code> | User updates aggregated worksheet.                                                                         |
| <code>answer_add_new_filter</code>            | User adds a filter using the UI.                                                                           |
| <code>data_show_underlying_viz</code>         | Request to show underlying data for a data row(s).                                                         |
| <code>answer_view</code>                      | User opens an existing, saved answer.                                                                      |
| <code>answer_viz_context_view</code>          | User opens an existing saved pinboard, edits a context viz.                                                |
| <code>pinboard_insight_view</code>            | User opens SpotIQ tab pinboards.                                                                           |

|                                                  |                                                             |
|--------------------------------------------------|-------------------------------------------------------------|
| <code>pinboard_admin_view</code>                 | User opens admin tab pinboards.                             |
| <code>pinboard_embed_view</code>                 | User opens embed pinboard from a URL.                       |
| <code>pinboard_homepage_view</code>              | On loading of homepage pinboard.                            |
| <code>pinboard_learn_view</code>                 | On loading learn pinboard.                                  |
| <code>pinboard_tspublic_no_runtime_filter</code> | Request to TSPublic/pinboard data without run-time filters. |

# Error code reference

## Summary: List of error codes and messages.

This section lists error codes that can appear in ThoughtSpot, with summary information and actions to take. Error codes and messages are shown in ThoughtSpot when something goes wrong. These messages can appear in the application and in logs.

When you see an error code, you will also see a message with a brief summary of what has happened. If there is a remediation action you can take, it will be listed in this references. If there is no action listed, please contact ThoughtSpot Support.

**Tip:** Only the base code number is listed for each error. So keep this in mind when searching through these codes. For example, error code TS-00125 is simply listed as 125.

## Metadata Errors (100 - 499)

| Code     | Severity | Summary                                                                                             | Details                                                                                               | Action                                                                                                                              |
|----------|----------|-----------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------|
| TS-00100 | INFO     | Success. {1} has been added to {2}. \# {1} – name of visualization \# {2} – {name/link to pinboard} | None                                                                                                  | None                                                                                                                                |
| TS-00101 | ERROR    | Failure adding {1} to {2}                                                                           | Visualization could not be added to {2} \# {1} – name of visualization \# {2} – name/link to pinboard | None                                                                                                                                |
| TS-00102 | ERROR    | Failure adding {1} to {2} due to corruption                                                         | {1} could not be added to {2} as the pinboard has one or more invalid visualizations                  | Please try again after removing the invalid visualization(s) from {2} \# {1} – name of visualization \# {2} – name/link to pinboard |
| TS-00103 | INFO     | Success. Visualization has been deleted from {1}. 1 – name/link to pinboard                         | None                                                                                                  | None                                                                                                                                |

|          |       |                                                              |                                                                                                                               |                                                                                |
|----------|-------|--------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------|
| TS-00104 | ERROR | Failure deleting visual from {1}                             | Visualization could not be deleted from the pinboard. 1 – name/link to pinboard                                               | None                                                                           |
| TS-00105 | ERROR | Failure deleting visual from {1} due to corruption           | Visualization could not be deleted from {1} as the pinboard has one or more invalid visualizations. 1 – name/link to pinboard | Please try again after removing the invalid visualization(s) from the pinboard |
| TS-00106 | INFO  | Success. {1} created successfully. 1 – name/link to pinboard | None                                                                                                                          | None                                                                           |
| TS-00107 | ERROR | Failure creating {1}. 1 – name/link to pinboard              | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.                                             | None                                                                           |
| TS-00108 | INFO  | Sticker created successfully.                                | None                                                                                                                          | None                                                                           |
| TS-00109 | ERROR | Failure creating the sticker.                                | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.                                             | None                                                                           |
| TS-00110 | INFO  | Sticker deleted successfully.                                | None                                                                                                                          | None                                                                           |
| TS-00111 | ERROR | Failure deleting sticker.                                    | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.                                             | None                                                                           |
| TS-00112 | INFO  | Pinboards deleted successfully.                              | None                                                                                                                          | None                                                                           |
| TS-00113 | ERROR | Failure deleting pinboards                                   | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.                                             | None                                                                           |
| TS-00114 | INFO  | Answers deleted successfully.                                | None                                                                                                                          | None                                                                           |

|          |       |                                         |                                                                                   |      |
|----------|-------|-----------------------------------------|-----------------------------------------------------------------------------------|------|
| TS-00115 | ERROR | Failure deleting answers                | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None |
| TS-00116 | INFO  | Tables deleted successfully.            | None                                                                              | None |
| TS-00117 | ERROR | Failure deleting tables                 | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None |
| TS-00118 | INFO  | Relationship created successfully.      | None                                                                              | None |
| TS-00119 | ERROR | Failure creating relationship           | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None |
| TS-00120 | INFO  | Relationship updated successfully.      | None                                                                              | None |
| TS-00121 | ERROR | Failure updating the relationship       | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None |
| TS-00122 | INFO  | Relationship deleted successfully.      | None                                                                              | None |
| TS-00123 | ERROR | Failure deleting the relationship       | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None |
| TS-00124 | ERROR | Failure fetching details for table      | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None |
| TS-00125 | ERROR | Failure fetching details for the tables | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None |

|          |       |                                             |                                                                                   |      |
|----------|-------|---------------------------------------------|-----------------------------------------------------------------------------------|------|
| TS-00126 | ERROR | Failure fetching details for datasource     | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None |
| TS-00127 | ERROR | Failure fetching details for datasources    | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None |
| TS-00128 | ERROR | Failure fetching details for metadata items | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None |
| TS-00129 | ERROR | Failure opening the answer                  | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None |
| TS-00130 | ERROR | Failure opening the pinboard                | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None |
| TS-00131 | ERROR | Failure opening the worksheet               | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None |
| TS-00132 | INFO  | Table saved successfully.                   | None                                                                              | None |
| TS-00133 | ERROR | There was a problem saving the table        | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None |
| TS-00134 | INFO  | Visualization update successful             | None                                                                              | None |
| TS-00135 | ERROR | Visualization failed to update              | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None |

|          |       |                                                    |                                                                                                        |      |
|----------|-------|----------------------------------------------------|--------------------------------------------------------------------------------------------------------|------|
| TS-00136 | INFO  | {1} saved 1 – name of answer                       | None                                                                                                   | None |
| TS-00137 | ERROR | {1} could not be saved 1 – name of answer          | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.                      | None |
| TS-00138 | INFO  | {1} saved 1 - name of pinboard / link              | None                                                                                                   | None |
| TS-00139 | ERROR | {1} could not be saved 1 - name of pinboard / link | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.                      | None |
| TS-00140 | INFO  | {1} saved 1 – name of worksheet                    | None                                                                                                   | None |
| TS-00141 | ERROR | {1} could not be saved 1 – name of worksheet       | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.                      | None |
| TS-00142 | INFO  | {1} saved 1 – name of answer                       | None                                                                                                   | None |
| TS-00143 | ERROR | {1} could not be saved                             | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. 1 – name of answer   | None |
| TS-00144 | INFO  | {1} saved 1 – name/ link to pinboard               | None                                                                                                   | None |
| TS-00145 | ERROR | {1} could not be saved                             | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. 1 – name of pinboard | None |
| TS-00146 | INFO  | Worksheet saved                                    | None                                                                                                   | None |
| TS-00147 | ERROR | Worksheet could not be saved                       | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.                      | None |

|          |       |                                     |                                                                                   |      |
|----------|-------|-------------------------------------|-----------------------------------------------------------------------------------|------|
| TS-00148 | INFO  | Sticker updated                     | None                                                                              | None |
| TS-00149 | ERROR | The sticker could not be updated    | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None |
| TS-00150 | INFO  | Successfully assigned sticker       | None                                                                              | None |
| TS-00151 | ERROR | The sticker could not be assigned   | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None |
| TS-00152 | INFO  | Successfully unassigned sticker     | None                                                                              | None |
| TS-00153 | ERROR | The sticker could not be unassigned | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None |
| TS-00154 | ERROR | Failed to fetch metadata list       | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None |
| TS-00155 | ERROR | Failed to fetch table list          | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None |
| TS-00156 | ERROR | Failed to fetch relationship list   | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None |
| TS-00157 | ERROR | Failed to fetch answer list         | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None |

|          |       |                                           |                                                                                   |      |
|----------|-------|-------------------------------------------|-----------------------------------------------------------------------------------|------|
| TS-00158 | ERROR | Failed to fetch pin-board list            | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None |
| TS-00159 | ERROR | Failed to fetch worksheet list            | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None |
| TS-00160 | ERROR | Failed to fetch aggregated worksheet list | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None |
| TS-00161 | ERROR | Failed to fetch imported data list        | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None |
| TS-00162 | ERROR | Failed to fetch system table list         | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None |
| TS-00163 | ERROR | Failed to DB view list                    | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None |
| TS-00164 | ERROR | Failed to fetch data source list          | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None |
| TS-00165 | ERROR | Failed to fetch column list               | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None |
| TS-00166 | ERROR | Failed to label list                      | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None |

|          |       |                                                                   |                                                                                   |      |
|----------|-------|-------------------------------------------------------------------|-----------------------------------------------------------------------------------|------|
| TS-00167 | ERROR | Failed to fetch answer                                            | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None |
| TS-00168 | ERROR | Failed to fetch worksheet                                         | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None |
| TS-00169 | INFO  | Aggregated worksheet {1} created 1 – name of aggregated worksheet | None                                                                              | None |
| TS-00170 | ERROR | Failure creating Aggregated Worksheet.                            | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None |
| TS-00171 | INFO  | {1} updated 1 – name of aggregated worksheet                      | None                                                                              | None |
| TS-00172 | ERROR | {1} failed to update 1 – name of aggregated worksheet             | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None |
| TS-00173 | ERROR | {1} failed to update 1 – name of the formula                      | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None |
| TS-00174 | ERROR | Comments cannot be fetched                                        | Failed to save client state                                                       | None |
| TS-00175 | ERROR | Comment cannot be created                                         | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None |
| TS-00176 | ERROR | Comment cannot be updated                                         | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None |

|          |       |                                    |                                                                                                                           |      |
|----------|-------|------------------------------------|---------------------------------------------------------------------------------------------------------------------------|------|
| TS-00177 | ERROR | Comment cannot be deleted          | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.                                         | None |
| TS-00178 | INFO  | Rule saved successfully            | None                                                                                                                      | None |
| TS-00179 | ERROR | Rule could not be saved            | We're not sure what happened. Please email the trace file to {adminEmail}.                                                | None |
| TS-00180 | INFO  | Rule deleted successfully          | None                                                                                                                      | None |
| TS-00181 | ERROR | Rule could not be deleted          | We're not sure what happened. Please email the trace file to {adminEmail}.                                                | None |
| TS-00182 | INFO  | Item deleted successfully.         | None                                                                                                                      | None |
| TS-00183 | ERROR | Item could not be deleted.         | We're not sure what happened. Please email the trace file to {adminEmail}.                                                | None |
| TS-00184 | INFO  | Related link created successfully. | None                                                                                                                      | None |
| TS-00185 | ERROR | Related link could not be created. | Uh oh. We're not sure what happened. Please click 'Report Problem' to email a report to your administrator, {adminEmail}. | None |
| TS-00186 | INFO  | Related link updated successfully. | None                                                                                                                      | None |
| TS-00187 | ERROR | Related link could not be updated. | Uh oh. We're not sure what happened. Please click 'Report Problem' to email a report to your administrator, {adminEmail}. | None |
| TS-00188 | INFO  | Related link deleted successfully. | None                                                                                                                      | None |

|          |       |                                           |                                                                                                                           |      |
|----------|-------|-------------------------------------------|---------------------------------------------------------------------------------------------------------------------------|------|
| TS-00189 | ERROR | Related link could not be deleted.        | Uh oh. We're not sure what happened. Please click 'Report Problem' to email a report to your administrator, {adminEmail}. | None |
| TS-00190 | INFO  | Related link detail fetched successfully. | None                                                                                                                      | None |
| TS-00191 | ERROR | Related link detail could not be fetched. | Uh oh. We're not sure what happened. Please click 'Report Problem' to email a report to your administrator, {adminEmail}. | None |

## Data Service Errors (500 - 699)

| Code     | Severity | Summary                             | Details                                  | Action |
|----------|----------|-------------------------------------|------------------------------------------|--------|
| TS-00500 | ERROR    | Failed to fetch leaf level data     | Failed to fetch leaf level data.         | None   |
| TS-00501 | ERROR    | Failed to fetch excel data          | Failed to fetch excel data.              | None   |
| TS-00502 | ERROR    | Failed to fetch visualization data  | Failed to fetch visuzliation data.       | None   |
| TS-00503 | ERROR    | Failed to fetch visualizations data | Failed to fetch data for visualizations. | None   |
| TS-00504 | ERROR    | Failed to fetch chart data          | Failed to fetch table data.              | None   |
| TS-00505 | ERROR    | Failed to fetch table data          | Failed to fetch table data.              | None   |
| TS-00506 | ERROR    | Failed to fetch worksheet data      | Failed to fetch worksheet data.          | None   |
| TS-00507 | ERROR    | Failed to fetch filter data         | Failed to fetch filter data.             | None   |

| <b>Code</b> | <b>Severity</b> | <b>Summary</b>                                  | <b>Details</b>                                                                                                            | <b>Action</b> |
|-------------|-----------------|-------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------|---------------|
| TS-00508    | ERROR           | Failed to fetch headline data                   | Failed to fetch filter data.                                                                                              | None          |
| TS-00509    | ERROR           | Failed to fetch natural query                   | Failed to fetch natural query.                                                                                            | None          |
| TS-00510    | INFO            | File upload successful                          | None                                                                                                                      | None          |
| TS-00511    | ERROR           | Failed to upload file                           | Failed to upload                                                                                                          | None          |
| TS-00512    | ERROR           | The pinboard data could not be exported to pdf. | Uh oh. We're not sure what happened. Please click 'Report Problem' to email a report to your administrator, {adminEmail}. | None          |

## Dependency Errors (700 - 799)

| <b>Code</b> | <b>Severity</b> | <b>Summary</b>                     | <b>Details</b>                             | <b>Action</b> |
|-------------|-----------------|------------------------------------|--------------------------------------------|---------------|
| TS-00700    | ERROR           | Failure fetching table dependents  | Failed to fetch dependents for the table.  | None          |
| TS-00701    | ERROR           | Failure fetching column dependents | Failed to fetch dependents for the column. | None          |
| TS-00702    | ERROR           | Failure fetching incomplete items  | Failed to fetch incomplete items.          | None          |

## Admin Service Errors (800 - 899)

| <b>Code</b> | <b>Severity</b> | <b>Summary</b>                  | <b>Details</b>                  | <b>Action</b> |
|-------------|-----------------|---------------------------------|---------------------------------|---------------|
| TS-00800    | ERROR           | Failure fetching MemCache stats | Failed to fetch MemCache stats. | None          |
| TS-00801    | ERROR           | Failure MemCache Clear          | Failed to clear MemCache.       | None          |
| TS-00802    | ERROR           | Failure searching from MemCache | Failed to search from MemCache. | None          |
| TS-00803    | ERROR           | Failure fetching Loggers        | Failed to fetch Loggers.        | None          |
| TS-00804    | ERROR           | Failure setting LogLevel        | Failed to set Log Level.        | None          |

| Code     | Severity | Summary                       | Details                   | Action |
|----------|----------|-------------------------------|---------------------------|--------|
| TS-00805 | ERROR    | Failure getting debug info    | Failed to get debug info. | None   |
| TS-00806 | INFO     | Memcache cleared successfully | None                      | None   |
| TS-00807 | INFO     | Log level set successfully    | None                      | None   |
| TS-00808 | ERROR    | Failed to report problem      | None                      | None   |
| TS-00809 | INFO     | Problem reported successfully | None                      | None   |

## Permissions Errors (900 - 999)

| Code     | Severity | Summary                               | Details                               | Action |
|----------|----------|---------------------------------------|---------------------------------------|--------|
| TS-00900 | ERROR    | Failure fetching table permissions    | Failed to fetch table permissions.    | None   |
| TS-00901 | ERROR    | Failure fetching answer permissions   | Failed to fetch answer permissions.   | None   |
| TS-00902 | ERROR    | Failure fetching pinboard permissions | Failed to fetch pinboard permissions. | None   |
| TS-00903 | ERROR    | Failure getting metadata permissions  | Failed to get metadata permissions.   | None   |

## Import Data Errors (1000 - 1099)

| Code     | Severity | Summary                       | Details                       | Action |
|----------|----------|-------------------------------|-------------------------------|--------|
| TS-01000 | ERROR    | Data caching failed           | Data caching failed.          | None   |
| TS-01001 | ERROR    | Read Columns failed.          | Failed to read columns.       | None   |
| TS-01002 | ERROR    | Failed to read keys.          | Failed to read keys.          | None   |
| TS-01003 | ERROR    | Failed to read relationships. | Failed to read relationships. | None   |
| TS-01004 | ERROR    | Failed to load data.          | Failed to load data.          | None   |
| TS-01005 | ERROR    | Failed to create table.       | Failed to create table.       | None   |
| TS-01006 | ERROR    | Failed to fetch data rows.    | Failed to fetch data rows.    | None   |
| TS-01007 | ERROR    | Failed to delete files.       | Failed to fetch data rows.    | None   |

| <b>Code</b> | <b>Severity</b> | <b>Summary</b>                 | <b>Details</b>                 | <b>Action</b> |
|-------------|-----------------|--------------------------------|--------------------------------|---------------|
| TS-01008    | ERROR           | Failed to abort create table.  | Failed to abort create table.  | None          |
| TS-01009    | ERROR           | Failed to create schema.       | Failed to create schema.       | None          |
| TS-01010    | ERROR           | Failed to fetch table models.  | Failed to fetch table models.  | None          |
| TS-01011    | ERROR           | Failed to fetch sample values. | Failed to fetch sample values. | None          |

## Scheduled Jobs Errors (1100 - 1199)

| <b>Code</b> | <b>Severity</b> | <b>Summary</b>                | <b>Details</b> | <b>Action</b>                                                          |
|-------------|-----------------|-------------------------------|----------------|------------------------------------------------------------------------|
| TS-01100    | INFO            | The list of jobs.             | None           | Please click 'Report Problem' to email a report to your administrator. |
| TS-01110    | INFO            | Successfully created job.     | None           | None                                                                   |
| TS-01111    | ERROR           | The job could not be created. | None           | Please click 'Report Problem' to email a report to your administrator. |
| TS-01112    | INFO            | Successfully updated job.     | None           | None                                                                   |
| TS-01113    | ERROR           | The job could not be updated. | None           | Please click 'Report Problem' to email a report to your administrator. |
| TS-01114    | INFO            | Successfully deleted jobs.    | None           | None                                                                   |
| TS-01115    | ERROR           | The job could not be deleted. | None           | Please click 'Report Problem' to email a report to your administrator. |
| TS-01116    | INFO            | The job was paused.           | None           | None                                                                   |
| TS-01117    | ERROR           | The job could not be paused.  | None           | Please click 'Report Problem' to email a report to your administrator. |
| TS-01118    | INFO            | The job was resumed           | None           | None                                                                   |
| TS-01119    | ERROR           | The job could not be resumed. | None           | Please click 'Report Problem' to email a report to your administrator. |

## User Admin Service Errors (1200 - 1399)

| Code     | Severity | Summary                               | Details                               | Action |
|----------|----------|---------------------------------------|---------------------------------------|--------|
| TS-01200 | ERROR    | Failed to fetch users list            | Failed to fetch users list            | None   |
| TS-01201 | ERROR    | Failed to fetch groups list           | Failed to fetch groups list           | None   |
| TS-01202 | ERROR    | Failed to fetch users and groups list | Failed to fetch users and groups list | None   |
| TS-01203 | ERROR    | Successfully created user             | Successfully created user             | None   |
| TS-01204 | ERROR    | Failed to create user                 | Failed to create user                 | None   |
| TS-01205 | ERROR    | Successfully created group            | Successfully created group            | None   |
| TS-01206 | ERROR    | Failed to create group                | Failed to create group                | None   |
| TS-01207 | ERROR    | Successfully updated user             | Successfully updated user             | None   |
| TS-01208 | ERROR    | Failed to update user                 | Failed to update user                 | None   |
| TS-01209 | ERROR    | Successfully updated users            | Successfully updated users            | None   |
| TS-01210 | ERROR    | Failed to update users                | Failed to update users                | None   |
| TS-01211 | ERROR    | Successfully updated group            | Successfully updated group            | None   |
| TS-01212 | ERROR    | Failed to update group                | Failed to update group                | None   |
| TS-01213 | ERROR    | Successfully updated password         | Successfully updated password         | None   |
| TS-01214 | ERROR    | Failed to update password             | Failed to update password             | None   |
| TS-01215 | ERROR    | Successfully deleted users            | Successfully deleted users            | None   |
| TS-01216 | ERROR    | Failed to delete users                | Failed to delete users                | None   |
| TS-01217 | ERROR    | Successfully deleted groups           | Successfully deleted groups           | None   |
| TS-01218 | ERROR    | Failed to delete groups               | Failed to delete groups               | None   |
| TS-01219 | ERROR    | Successfully assigned users to groups | Successfully assigned users to groups | None   |
| TS-01220 | ERROR    | Failed to assign users to groups      | Failed to assign users to groups      | None   |

| Code     | Severity | Summary                               | Details                        | Action |
|----------|----------|---------------------------------------|--------------------------------|--------|
| TS-01221 | ERROR    | Failed to fetch profile pic           | Failed to fetch profile pic    | None   |
| TS-01222 | INFO     | Successfully uploaded profile pic     | None                           | None   |
| TS-01223 | ERROR    | Failed to upload profile pic          | Failed to upload profile pic   | None   |
| TS-01224 | ERROR    | Successfully assigned groups to group | Failed to assign user to group | None   |
| TS-01228 | ERROR    | Successfully created role             | Successfully created role      | None   |
| TS-01229 | ERROR    | Failed to create role                 | Failed to create role          | None   |
| TS-01230 | ERROR    | Successfully deleted role             | Successfully deleted role      | None   |
| TS-01231 | ERROR    | Failed to delete role                 | Failed to delete role          | None   |
| TS-01232 | ERROR    | Successfully updated role             | Successfully updated role      | None   |
| TS-01233 | ERROR    | Failed to update role                 | Failed to update role          | None   |

## Session Service Errors (1400 - 1599)

| Code     | Severity | Summary                             | Details                                                                              | Action |
|----------|----------|-------------------------------------|--------------------------------------------------------------------------------------|--------|
| TS-01400 | ERROR    | Failed to fetch session info        | Failed to fetch session info                                                         | None   |
| TS-01401 | ERROR    | Failed to login                     | Uh oh. We're not sure what happened.<br>Please email the trace file to {adminEmail}. | None   |
| TS-01402 | ERROR    | Failed to logout                    | Failed to logout                                                                     | None   |
| TS-01403 | ERROR    | Failed to save client state         | Failed to save client state                                                          | None   |
| TS-01404 | ERROR    | Failed to fetch login config        | Failed to fetch login config                                                         | None   |
| TS-01405 | ERROR    | Failed to fetch slack config        | Failed to fetch slack config                                                         | None   |
| TS-01406 | ERROR    | Health check failed                 | Health check failed                                                                  | None   |
| TS-01407 | ERROR    | Failed to fetch health portal token | Failed to fetch health portal token                                                  | None   |

| <b>Code</b> | <b>Severity</b> | <b>Summary</b>                                        | <b>Details</b>                                                                    | <b>Action</b> |
|-------------|-----------------|-------------------------------------------------------|-----------------------------------------------------------------------------------|---------------|
| TS-01408    | ERROR           | The health portal release name could not be retrieved | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None          |

## Data Management Service Errors (1600 - 1799)

| <b>Code</b> | <b>Severity</b> | <b>Summary</b>                                  | <b>Details</b>                                  | <b>Action</b> |
|-------------|-----------------|-------------------------------------------------|-------------------------------------------------|---------------|
| TS-01600    | ERROR           | Failed to fetch data source types               | Failed to fetch data source types               | None          |
| TS-01601    | ERROR           | Failed to fetch data source sample values       | Failed to fetch data source sample values       | None          |
| TS-01602    | ERROR           | Failed to delete data source                    | Failed to delete data source                    | None          |
| TS-01603    | ERROR           | Failed to execute DDL                           | Failed to execute DDL                           | None          |
| TS-01604    | ERROR           | Failed to update schedule                       | Failed to update schedule                       | None          |
| TS-01605    | ERROR           | Failed to reload tasks                          | Failed to reload tasks                          | None          |
| TS-01606    | ERROR           | Failed to stop tasks                            | Failed to stop tasks                            | None          |
| TS-01607    | ERROR           | Failed to get creation DDL                      | Failed to get creation DDL                      | None          |
| TS-01608    | ERROR           | Failed to load from data source                 | Failed to load from data source                 | None          |
| TS-01609    | ERROR           | Failed to create connection to data source      | Failed to create connection to data source      | None          |
| TS-01610    | ERROR           | Failed to create data source                    | Failed to create data source                    | None          |
| TS-01611    | ERROR           | Failed to connect to data source                | Failed to connect to data source                | None          |
| TS-01612    | ERROR           | Failed to get data source connection field info | Failed to get data source connection field info | None          |
| TS-01613    | ERROR           | Failed to get connection list for data source   | Failed to get connection list for data source   | None          |

| <b>Code</b> | <b>Severity</b> | <b>Summary</b>                                      | <b>Details</b>                                                          | <b>Action</b> |
|-------------|-----------------|-----------------------------------------------------|-------------------------------------------------------------------------|---------------|
| TS-01614    | ERROR           | Failed to get connection attributes for data source | Failed to get connection attributes for data source                     | None          |
| TS-01615    | ERROR           | Failed to get connections to data source            | Failed to get connections to data source                                | None          |
| TS-01616    | ERROR           | Failed to fetch data source config                  | Failed to fetch data source config                                      | None          |
| TS-01617    | ERROR           | Failed to parse sql.                                | Failed to parse sql.                                                    | None          |
| TS-01618    | ERROR           | Failed to execute sql.                              | Failed to execute sql.                                                  | None          |
| TS-01619    | INFO            | Successfully created connection to data source      | None                                                                    | None          |
| TS-01620    | INFO            | Successfully updated data upload schedule           | None                                                                    | None          |
| TS-01621    | ERROR           | Failed to execute sql.                              | Please check the failing command, executed {1} statements successfully. | None          |
| TS-01622    | ERROR           | Lightweight data-cache disabled                     | Lightweight data-cache disabled                                         | None          |
| TS-01623    | INFO            | Selected tables were queued for loading.            | Selected tables were queued for loading.                                | None          |
| TS-01624    | ERROR           | Data Type conversion error.                         | No mapping found for source datatype to ThoughtSpot datatype.           | None          |
| TS-01625    | INFO            | Successfully reload task started.                   | None                                                                    | None          |
| TS-01626    | INFO            | Successfully connected to data source.              | None                                                                    | None          |
| TS-01627    | INFO            | Successfully created data source.                   | None                                                                    | None          |
| TS-01628    | INFO            | Successfully stopped the tasks.                     | None                                                                    | None          |
| TS-01629    | INFO            | Successfully deleted the connection.                | None                                                                    | None          |
| TS-01630    | ERROR           | There was an error deleting this connection.        | None                                                                    | None          |
| TS-01631    | INFO            | Successfully executed the DDL.                      | None                                                                    | None          |

## Cluster Status Service Errors (1800 - 1899)

| Code     | Severity | Summary                                                              | Details              | Action |
|----------|----------|----------------------------------------------------------------------|----------------------|--------|
| TS-01800 | WARNING  | Failed to fetch cluster information from search service.             | None                 | None   |
| TS-01801 | WARNING  | Failed to fetch table detail information from search service.        | None                 | None   |
| TS-01802 | WARNING  | Failed to fetch cluster information from database service.           | None                 | None   |
| TS-01803 | WARNING  | Failed to fetch table detail information from databse service.       | None                 | None   |
| TS-01804 | WARNING  | Failed to fetch cluster information from cluster management service. | None                 | None   |
| TS-01805 | WARNING  | Failed to fetch detail information from cluster management service.  | None                 | None   |
| TS-01806 | WARNING  | Failed to fetch log from cluster management service.                 | None                 | None   |
| TS-01807 | WARNING  | Failed to fetch snapshot list from cluster man-agement service.      | None                 | None   |
| TS-01808 | WARNING  | Failed to fetch cluster information from alert management service.   | None                 | None   |
| TS-01809 | WARNING  | Failed to fetch cluster information from event service.              | None                 | None   |
| TS-01810 | WARNING  | Failed to fetch alerts information from alert management service.    | None                 | None   |
| TS-01811 | WARNING  | Failed to fetch events information from alert management service.    | None                 | None   |
| TS-01812 | INFO     | Thanks for your feedback!                                            | None                 | None   |
| TS-01813 | WARNING  | Sorry! Unable to submit the feedback at this moment!                 | None                 | None   |
| TS-01814 | INFO     | Successfully exported objects. File can be found at {1}.             | None                 | None   |
| TS-01815 | ERROR    | Sorry! Unable to export objects at this mo-ment!                     | What hap-pened? {1}. | None   |

| Code     | Severity | Summary                                         | Details             | Action |
|----------|----------|-------------------------------------------------|---------------------|--------|
| TS-01816 | INFO     | Successfully imported objects                   | None                | None   |
| TS-01817 | ERROR    | Sorry! Unable to import objects at this moment! | What happened? {1}. | None   |
| TS-01818 | INFO     | Successfully deleted data source object(s).     | None                | None   |

## Callosum API Errors (9000 - 9199)

| Code     | Severity | Summary                                                    | Details                                                                           | Action                                             |
|----------|----------|------------------------------------------------------------|-----------------------------------------------------------------------------------|----------------------------------------------------|
| TS-09000 | ERROR    | The data you are trying to delete has some dependencies    | Some objects depend on the data you are trying to delete                          | delete the dependencies before deleting this data. |
| TS-09001 | ERROR    | Uh oh. We're not sure what happened.                       | Please email the trace file to {adminEmail}.                                      | None                                               |
| TS-09002 | ERROR    | Could not authorize user                                   | Try logging in again                                                              | None                                               |
| TS-09003 | ERROR    | Uh oh. We're not sure what happened.                       | Please email the trace file to {adminEmail}.                                      | None                                               |
| TS-09004 | WARNING  | Still loading data, come back soon                         | None                                                                              | None                                               |
| TS-09005 | ERROR    | Uh oh. We're having trouble getting data for this request. | Please email the trace file to {adminEmail}.                                      | None                                               |
| TS-09006 | ERROR    | Uh oh. We're having trouble getting data for this request. | Please email the trace file to {adminEmail}.                                      | None                                               |
| TS-09007 | ERROR    | Uh oh. We're having trouble getting data for this request. | Please email the trace file to {adminEmail}.                                      | None                                               |
| TS-09008 | ERROR    | Something went wrong with your search                      | Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. | None                                               |

| Code                                     | Severity | Summary                                                          | Details                                                                                                               | Action |
|------------------------------------------|----------|------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------|--------|
| TS-09009                                 | ERROR    | The calculation engine has timed out. Please try again.          | Please email the trace file to {adminEmail}.                                                                          | None   |
| TS-09010                                 | ERROR    | Cannot open Object                                               | Object cannot be opened due to errors in some of its dependencies                                                     | None   |
| TS-0Blink Generated Errors (9500 - 9599) |          |                                                                  |                                                                                                                       |        |
| TS-09500                                 | WARNING  | Cannot connect to the calculation engine. Please try again soon. | None                                                                                                                  | None   |
| TS-09501                                 | WARNING  | The calculation engine has timed out. Please try again.          | None                                                                                                                  | None   |
| TS-09502                                 | WARNING  | Cannot connect to the search engine. Please try again soon.      | None                                                                                                                  | None   |
| TS-09503                                 | WARNING  | The search engine has timed out. Please try again.               | None                                                                                                                  | None   |
| TS-09504                                 | ERROR    | Cannot open {1}                                                  | {1} cannot be opened due to errors in the following dependencies<br>1 - Type of the object Table/Answer/Pinboard etc. | None   |
| TS-09505                                 | WARNING  | We're still indexing this data, try again soon                   | None                                                                                                                  | None   |
| TS-09506                                 | ERROR    | Object is not present in the system                              | {1} is not present in the system<br>1 - Type of the object Table/Answer/Pinboard etc.                                 | None   |
| TS-09507                                 | ERROR    | ThoughtSpot is unreachable. Please try again soon                | None                                                                                                                  | None   |

## Common Errors (10000 - 10099)

| Code     | Severity | Summary                                | Details                                                                                                       | Action                                        |
|----------|----------|----------------------------------------|---------------------------------------------------------------------------------------------------------------|-----------------------------------------------|
| TS-10000 | ERROR    | A system error has occurred            | Uh oh. We're not sure what happened. Please contact your administrator.                                       | None                                          |
| TS-10001 | ERROR    | Connection failed                      | The metadata store is not reachable.                                                                          | Please contact your administrator             |
| TS-10002 | ERROR    | The input is invalid                   | Input from the client to the server is invalid.                                                               | Please contact your administrator             |
| TS-10003 | ERROR    | Unfortunately, you can't do that       | You are not authorized to perform {1}. # {1} – action user is not authorized for                              | Please request access from your administrator |
| TS-10004 | ERROR    | The user could not be authorized       | User {0} is not authorized to perform {1}. # {0} – name of the user # {1} – action user is not authorized for | Please request access from your administrator |
| TS-10005 | ERROR    | The base object is missing             | An underlying object referenced by this object is missing in store.                                           | Please contact your administrator             |
| TS-10006 | ERROR    | The connection to Zookeeper has failed | Zookeeper is not reachable.                                                                                   | Please contact your administrator             |
| TS-10007 | ERROR    | There's invalid parameter(s)           | Invalid parameter values: {0}.                                                                                | Please contact your administrator             |
| TS-10008 | ERROR    | The user cannot be found               | User {0} not found in store. # {0} – name of the user                                                         | Please contact your administrator             |
| TS-10009 | ERROR    | Cannot add group                       | This group already belongs to the group you are trying to add it to.                                          | None                                          |

## Falcon Errors (10600 - 10699)

| Code     | Severity | Summary                | Details | Action |
|----------|----------|------------------------|---------|--------|
| TS-10603 | ERROR    | Falcon query cancelled | None    | None   |

## Data Errors (11000 - 11099)

| Code     | Severity | Summary                          | Details                                         | Action |
|----------|----------|----------------------------------|-------------------------------------------------|--------|
| TS-11001 | ERROR    | Invalid row                      | None                                            | None   |
| TS-11002 | ERROR    | Invalid table/query resultset    | None                                            | None   |
| TS-11003 | ERROR    | Invalid column identifier        | None                                            | None   |
| TS-11004 | ERROR    | Invalid visualization identifier | None                                            | None   |
| TS-11005 | ERROR    | No data                          | Query execution resulted in no data.            | None   |
| TS-11006 | ERROR    | Query execution failed           | Error in query execution to Falcon.             | None   |
| TS-11007 | ERROR    | Answer data generation failed    | Error in Answer data generation for Sage input. | None   |
| TS-11008 | ERROR    | Data export failed               | None                                            | None   |
| TS-11009 | ERROR    | Data generation failed           | Error in data generation in Callosum.           | None   |

## Report Generation Errors (12000 - 13000)

| Code     | Severity | Summary                          | Details                               | Action |
|----------|----------|----------------------------------|---------------------------------------|--------|
| TS-12700 | ERROR    | Error while exporting data file. | None                                  | None   |
| TS-12701 | ERROR    | Invalid input.                   | The definition of the job is invalid. | None   |
| TS-12702 | ERROR    | No author provided.              | None                                  | None   |
| TS-12703 | ERROR    | No pinboard provided.            | None                                  | None   |

| <b>Code</b> | <b>Severity</b> | <b>Summary</b>                         | <b>Details</b>                        | <b>Action</b> |
|-------------|-----------------|----------------------------------------|---------------------------------------|---------------|
| TS-12704    | ERROR           | No recipients provided.                | None                                  | None          |
| TS-12705    | ERROR           | This format is not supported.          | None                                  | None          |
| TS-12706    | ERROR           | No job name provided.                  | None                                  | None          |
| TS-12707    | ERROR           | No job description provided.           | None                                  | None          |
| TS-12708    | ERROR           | Pinboard data export error.            | None                                  | None          |
| TS-12709    | ERROR           | Visualization data export error.       | None                                  | None          |
| TS-12710    | ERROR           | User data unavailable.                 | None                                  | None          |
| TS-12711    | ERROR           | Configuration information unavailable. | None                                  | None          |
| TS-12712    | ERROR           | There are too many recipients.         | The max number of recipients is 1000. | None          |
| TS-12713    | ERROR           | Attachment size limit exceeded.        | None                                  | None          |
| TS-12714    | ERROR           | Recipient domain is not whitelisted.   | None                                  | None          |

## More Metadata Errors (13000 - 13099)

| <b>Code</b> | <b>Severity</b> | <b>Summary</b>                      | <b>Details</b>                                                                                           | <b>Action</b> |
|-------------|-----------------|-------------------------------------|----------------------------------------------------------------------------------------------------------|---------------|
| TS-13001    | ERROR           | Schema creation failed              | Error creating database schema.                                                                          | None          |
| TS-13002    | ERROR           | Views creation failed               | Error creating view.                                                                                     | None          |
| TS-13003    | ERROR           | The object cannot be found in store | Object with Id: {0} of type: {1} not found. # {0} – identity of the object # {1} – type of object        | None          |
| TS-13004    | ERROR           | The object is in an invalid state   | Object with Id: {0} of type: {1} in invalid state. # {0} – identity of the object # {1} – type of object | None          |

| <b>Code</b> | <b>Severity</b> | <b>Summary</b>                          | <b>Details</b>                                                                                            | <b>Action</b> |
|-------------|-----------------|-----------------------------------------|-----------------------------------------------------------------------------------------------------------|---------------|
| TS-13005    | ERROR           | Object already exists                   | Object with Id: {0} of type: {1} already exists.<br># {0} – identity of the object # {1} – type of object | None          |
| TS-13006    | ERROR           | Invalid object type                     | Invalid type: {0} provided. # {1} – type of object                                                        | None          |
| TS-13007    | ERROR           | Invalid Sage question                   | Insufficient or invalid input from Sage: {0}. # {0} – the invalid input                                   | None          |
| TS-13008    | ERROR           | Invalid Sage question                   | Input from Sage – missing columns of type: {0}. # {0} – column type                                       | None          |
| TS-13009    | ERROR           | Invalid Sage question                   | Invalid input from Sage – invalid expression: {0}. # {0} – the invalid expression                         | None          |
| TS-13010    | ERROR           | Sending logical metadata to Sage failed | Sending logical metadata to Sage failed due to: {0}. # {0} – reason for failure                           | None          |
| TS-13011    | ERROR           | Answer generation failed                | Answer generation failed due to: {0}. # {0} – reason for failure                                          | None          |
| TS-13012    | ERROR           | Worksheet generation failed             | Worksheet generation failed due to: {0}. # {0} – reason for failure                                       | None          |
| TS-13013    | ERROR           | Service provider unavailable            | Service provider unavailable: {0}. # {0} – provider details                                               | None          |
| TS-13015    | ERROR           | Physical model not loaded               | None                                                                                                      | None          |
| TS-13016    | ERROR           | Invalid physical schema proto           | Inconsistency in physical schema from Falcon: {0}. # {0} – error details                                  | None          |
| TS-13017    | ERROR           | Invalid duplicate columns               | Duplicate columns: {0}. # {0} – List of duplicate column identities                                       | None          |

| <b>Code</b> | <b>Severity</b> | <b>Summary</b>                                                           | <b>Details</b>                                                                             | <b>Action</b>                                                                                           |
|-------------|-----------------|--------------------------------------------------------------------------|--------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------|
| TS-13018    | ERROR           | Cyclic relationship                                                      | Detected cycles: {0}. # {0} – cycle details                                                | None                                                                                                    |
| TS-13019    | WARNING         | Older physical schema version received                                   | Schema update for older version: {0} received and ignored. # {0} – received version number | None                                                                                                    |
| TS-13020    | ERROR           | Invalid relationship                                                     | Attempted to create invalid relationship: {0}. # {0} – relationship details                | None                                                                                                    |
| TS-13022    | ERROR           | Invalid filter values: {values}                                          | None                                                                                       | None                                                                                                    |
| TS-13023    | ERROR           | Creating relationship failed.                                            | None                                                                                       | None                                                                                                    |
| TS-13024    | ERROR           | Deleting schema failed.                                                  | None                                                                                       | None                                                                                                    |
| TS-13025    | ERROR           | Expression validation failed.                                            | None                                                                                       | None                                                                                                    |
| TS-13026    | INFO            | Load schedule successfully disabled.                                     | None                                                                                       | None                                                                                                    |
| TS-13027    | ERROR           | Load schedule could not be disabled.                                     | None                                                                                       | None                                                                                                    |
| TS-13028    | ERROR           | Objects fetched from the connection are invalid for editing data-source. | None                                                                                       | To proceed with editing the datasource, please edit the connection below to fetch valid source objects. |
| TS-13029    | INFO            | Successfully edited data source connection.                              | None                                                                                       | None                                                                                                    |
| TS-13030    | ERROR           | Connection test failed.                                                  | None                                                                                       | Please verify connection attributes.                                                                    |

## Loading Errors (30000 - 30099)

| Code     | Severity | Summary                                        | Details | Action |
|----------|----------|------------------------------------------------|---------|--------|
| TS-30000 | ERROR    | Table is not ready (data loading in progress). | None    | None   |

## Timely Errors (60000 - 64999)

| Code     | Severity | Summary               | Details | Action |
|----------|----------|-----------------------|---------|--------|
| TS-60000 | ERROR    | Failed to initialize. | None    | None   |

# Frequently asked questions

## Where can I find the version of ThoughtSpot I am using?

Users with administrative privileges can see this displayed on the **Admin > System Health > Overview** page.

## I'm not seeing certain columns/values in the drop-down, why?

It could be the index has not built with the latest data or something is causing the column to be dropped.

- Verify the the column is available using the **Data** page.
- View the table columns and check the **INDEX TYPE** value. If it is set to `DONT_INDEX`, change it.
- Check the column's **INDEX PRIORITY** &emdash; make sure it is `1`.

To learn more about modeling data see [modeling data](#) in this documentation.

## How do I track progress of current index build?

If you are an administrator, you can use the **Admin > System Health > Overview** page to see the number of tables currently being indexed. You can also review the

## How do I display the features used in my cluster configuration?

1. Log into the ThoughtSpot cluster as the `admin` user.
2. Use the `tscli feature` subcommand to display your current configuration.

| \$ tscli feature get-all-config |                                 |          |          |
|---------------------------------|---------------------------------|----------|----------|
| ACTION                          | NAME                            | STATUS   | CONFIGUR |
|                                 | Firewall                        | Disabled |          |
|                                 | Saml                            | Disabled |          |
|                                 | Ldap                            | Disabled |          |
|                                 | CustomBranding                  | Disabled |          |
|                                 | CustomBrandingFontCustomization | Disabled |          |
|                                 | DataConnect                     | Disabled |          |
|                                 | RLS                             | Enabled  |          |
|                                 | Callhome                        | Enabled  |          |
|                                 | SSHTunnel                       | Enabled  |          |
|                                 | Fileserver                      | Disabled |          |

### Is it possible to create a max(date) field and set it to filter?

If you have a date field in my set of data and want to return the most recent set of data based on specific date. To do this:

1. Create a formula called `Max Date`, for example:

```
date = group_max ( date_to_filter_by )
```

2. In the search bar, filter your dates by this formula for example:

```
max_date = true
```

This returns only those fields that pass the filter.