



Users Guide

Version 6.0 December 02, 2019

Copyright for ThoughtSpot publications. © 2019 ThoughtSpot, Inc. All rights reserved.

ThoughtSpot, Inc. 910 Hermosa Ct
Sunnyvale, CA 94085

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

Introduction.....	6
Getting Started	
User onboarding experience	7
Finding your way around	10
ThoughtSpot browser access	15
Your user preferences	17
Understanding privileges.....	25
Use stickers to organize	30
Use Search	
What are searches and answers?.....	34
Search basics	
Start a new search	37
Choose a data source.....	40
Search bar features	41
Search suggestions	44
Search results and column types	46
Last data refresh time.....	48
Work with Answers	50
Ask an Expert	55
Manage Ask an Expert requests.....	57
Retrieve Ask an Expert answers.....	60
Answer an Ask an Expert question	61
Results that are tables	64
Use keywords in search	
Overview of keyword searchess.....	68
Time series analysis.....	71
Search using growth over time.....	76
Using the in keyword for nested searches	80
Proximity searches "near" and "farther than".....	83
Work with filters	
Understand filters.....	85
Add a filter to a table.....	88

Add a filter to a chart	91
Create a bulk filter.....	94
Delete a filter	97
Filter on null, blank, or empty values.....	98
Other search actions	
Other search actions	100
Change result display options	101
Show underlying data	105
See the search behind a result	108
Apply conditional formatting to a table.....	110
Download your search.....	115
Replay search.....	117
Work with charts	
Understand charts.....	118
Chart types	
Column charts	123
Bar charts	125
Line charts	127
Pie charts	128
Area charts	130
Scatter charts	132
Bubble charts	133
Pareto charts	135
Waterfall charts	136
Treemap charts	137
Line column charts	139
Funnel charts.....	142
Geo charts	143
Pivot table	147
Sankey charts	153
Radar Charts.....	154
Candlestick Charts.....	155
Changing a chart	
Change the chart.....	160
Change axes options	163
Reorder the labels.....	180

Set the y-axis range	182
Hide and show values.....	184
Change chart colors	186
Additional chart options	191
Zoom into a chart.....	200
 Work with formulas	
Understand formulas	202
Add a formula to a search	204
View or edit a formula in a search	208
 Aggregate formulas	
Overview of aggregate formulas	210
Cumulative functions	214
Moving functions.....	218
Flexible aggregation functions (group aggregate and filters).....	223
Grouping functions	227
Filtered aggregation functions.....	230
Conversion functions	235
Date functions	236
Percent (simple number) calculations.....	243
Formula operators.....	244
Nested formulas	246
Formulas for chasm traps	247
 Use pinboards	
Basic pinboard usage	248
Follow a pinboard.....	253
Edit a pinboard	258
Pinboard filters	265
Schedule a pinboard job.....	270
Search actions within a pinboard	274
Copy a pinboard	275
Copy a pinboard or visualization link	277
Reset a pinboard visualization	280
Present a pinboard as a slideshow.....	282
Download as PDF	287
 Use custom R scripts	
About R in ThoughtSpot	295

Create and share R scripts.....	299
Run prebuilt R scripts on answers	310
Save and share R visualizations	317
SpotIQ	
 What is SpotIQ	322
 Best SpotIQ practices.....	324
 Monitor Headlines	328
 Comparative Analysis	331
 Custom SpotIQ analysis	340
 Advanced R customizations.....	349
 Insight feedback	354
 SpotIQ preferences.....	357
Work with data	
 Understand data sources	359
 Create and load CSV Files.....	361
 Append data through the UI	365
 View a data profile.....	367
 Date and currency format settings.....	368
 Share your work	
 About sharing	370
 Share a pinboard.....	372
 Share answers.....	376
 Share uploaded data.....	379
 Revoke access (unshare)	381
Use SearchIQ	
 About SearchIQ	384
 Use SearchIQ for natural searches	387
 Teach SearchIQ your language.....	391
Slack and ThoughtSpot	
 Slack and Spot.....	395
 How to use Spot	396
 More help and support.....	400
 Keyword reference.....	405
 TQL reference	413

tsload flag reference	426
tscli command reference.....	429
Date and time formats reference.....	474
Row level security rules reference	478
Formula function reference	490
Alert codes reference	506
User action codes reference	515
Frequently asked questions	517

Introduction to the User Guide

This ThoughtSpot User Guide contains information on navigating and searching data with ThoughtSpot. It assists 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 quickly and easily, just 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 don't need to understand how ThoughtSpot stores information, attend days of training, or know SQL to do these things. Collaboration and security features make it easy to protect sensitive data, and safely share information with others.

ThoughtSpot gives administrators the ability to modify data properties to meet business needs. Administrators can specify search synonyms for common terms, boost the importance of a column in search results, or format 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 can be changed.

- **Finding your way around**

To make navigation easy, we organized ThoughtSpot into several sections. You can see them on the menu bar.

- **About the user profile**

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

- **Understanding privileges**

Your privileges determine the things you can do. ThoughtSpot sets privileges at the group level.

- **About stickers**

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

User Onboarding Experience

Summary: ThoughtSpot's onboarding is quick and intuitive; you can learn to use the application very quickly and efficiently.

When you begin using ThoughtSpot, the onboarding process starts automatically and guides you through a few basic scenarios.

Usually, you receive an email welcoming you to the onboarding process at ThoughtSpot.

To repeat user onboarding, see [Revisit onboarding](#).

How onboarding works for the user

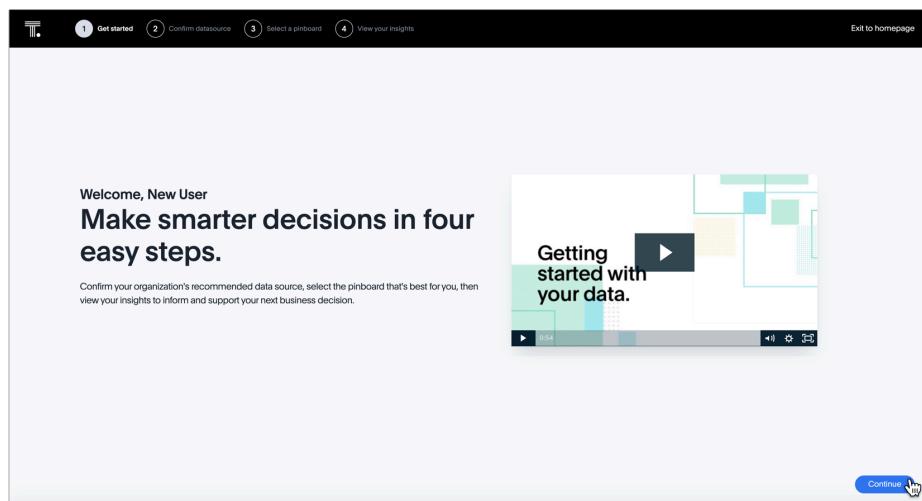
Your onboarding experience begins when you login for the first time.

1. Step 1: Get Started

The initial welcome screen provides an overview of the onboarding process, and offers a quick video.

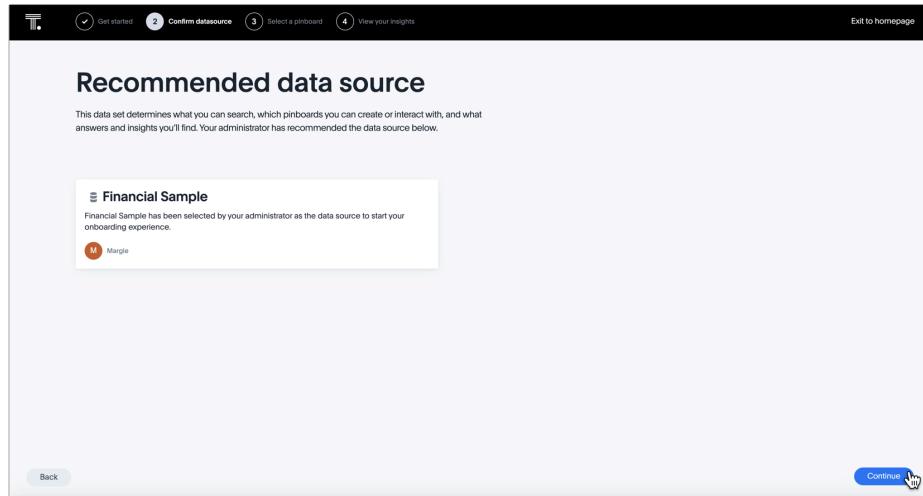
Watch the video, and click **Continue**.

Alternatively, you can click **Exit to homepage** at the top right corner of the screen, and end the onboarding process. This option appears at this and every subsequent step.



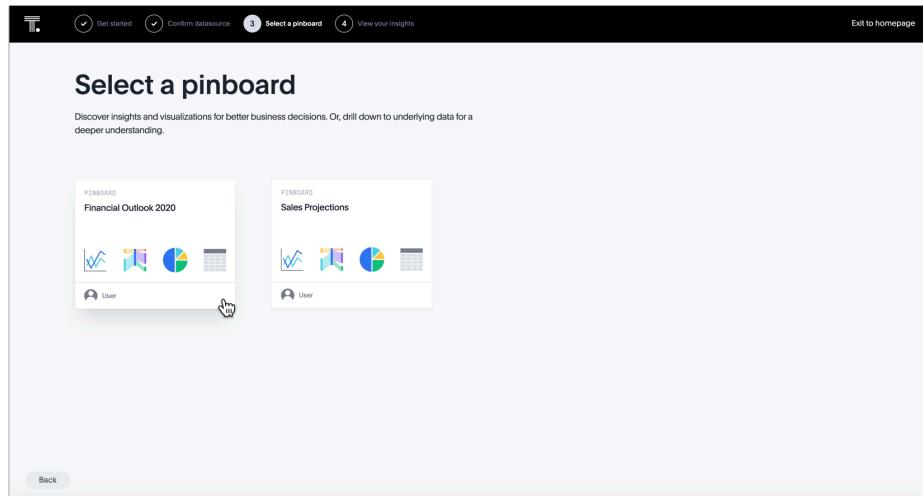
2. Step 2: Recommended data source

This screen introduces the primary data source for onboarding. The administrator selects this source before you begin. Click **Continue**.



3. Step 3: Select a pinboard

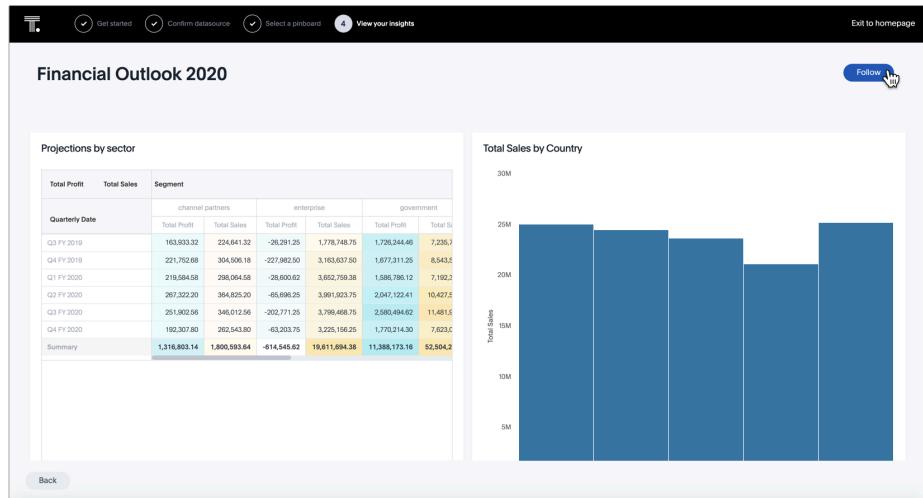
Consider which of the initial pinboards to explore first, and click on it.



4. Step 4: View your insights

Examine the pinboard you selected, and learn what insights it provides.

Click **Follow** to receive periodic emails about this pinboard.



Repeating the onboarding process

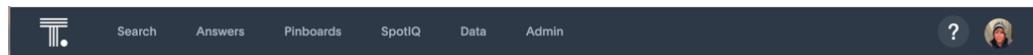
[Optional] Any user can repeat their onboarding experience at any time. Simply select **Profile** from the user icon on the top right corner of the page, and under **Preferences > New user onboarding**, click **Revisit Onboarding**. See [Revisit onboarding](#).

If you are a new user and you did not experience onboarding, please contact your administrator and request that they configure it for you and other new users.

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:



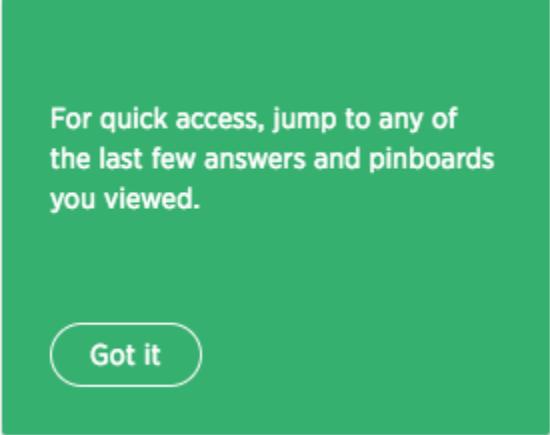
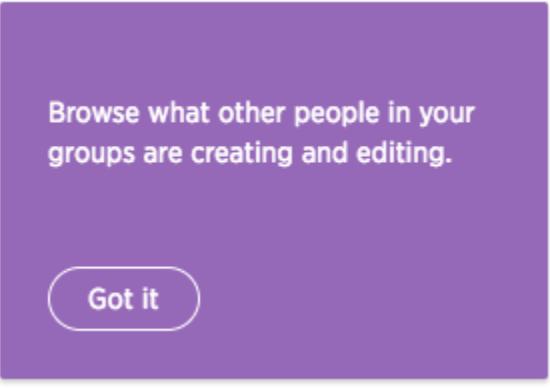
Home

Click the ThoughtSpot logo to go to the application home page. At the top of the home page you can see a search bar, and several areas that show activity in ThoughtSpot.

Label	Description
All time popular	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 can find when you click an item.
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 can find when you click an item.

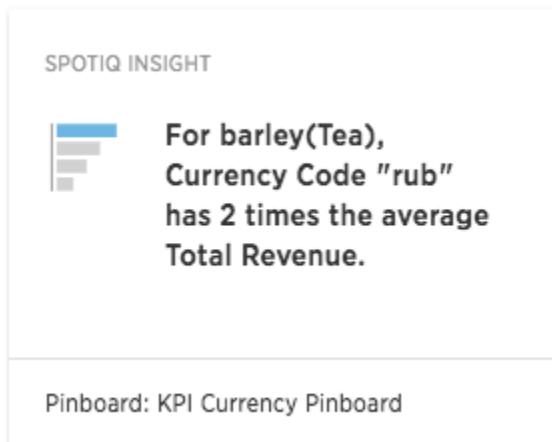
Check out all-time popular or recently trending answers and pinboards.

Got it

Label	Description
Recently viewed	Answers and pinboards you've viewed recently.  A green rectangular callout box with rounded corners and a thin white border. Inside, centered text reads: "For quick access, jump to any of the last few answers and pinboards you viewed." At the bottom center is a white button with a thin black outline and rounded corners, containing the text "Got it".
Recent team activity	Answers, pinboards, worksheets, and tables people in your company have created or edited recently.  A purple rectangular callout box with rounded corners and a thin white border. Inside, centered text reads: "Browse what other people in your groups are creating and editing." At the bottom center is a white button with a thin black outline and rounded corners, containing the text "Got it".

Label	Description
Did you know?	Contains auto analysis results from SpotIQ. Visible only to users that have the SpotIQ privilege.

SPOTIQ INSIGHT



For barley(Tea),
Currency Code "rub"
has 2 times the average
Total Revenue.

Pinboard: KPI Currency Pinboard

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.

If SearchIQ (**Beta**) is enabled by your administrator, this label is **Search+**, to differentiate it from SearchIQ. Use the search bar on the homepage to use SearchIQ.

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

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

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. After saving a pinboard, it can be shared with others or viewed as a slideshow. The **Pinboards** page shows a list of saved pinboards. Click one to view, edit, or share it.

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

SpotIQ

If SpotIQ is enabled by your administrator, you can go here to view the status of your analysis requests to see the results.

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

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 Icon

You can find information and support resources for ThoughtSpot here. There are 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.

To learn more about this section, visit [More help and support](#).

User Icon

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:

Firefox

68.x, 69.x, and later

Chrome

76.x, 77.x, and later

Internet Explorer

11.x, and later

Edge

44.x, and later

Safari

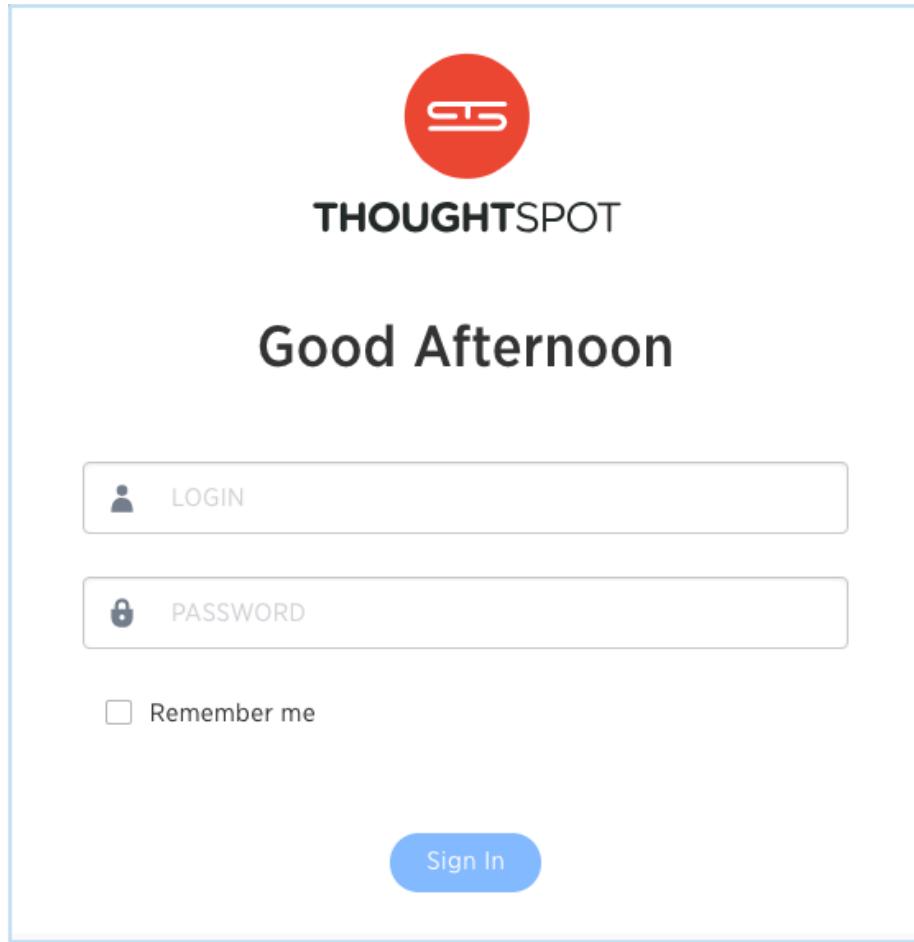
13.x, and later

☒ 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.

Sign in

To sign in to ThoughtSpot from a browser:

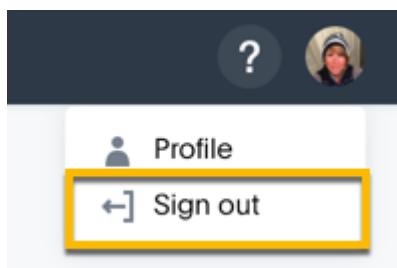
1. Open the browser and type in the Web address for ThoughtSpot: `http://HOSTNAME_OR_IP`
2. Enter your username and password and click **Sign In**.



Sign out

After you finish the search session, you can optionally sign out of ThoughtSpot. To sign out of ThoughtSpot from a browser:

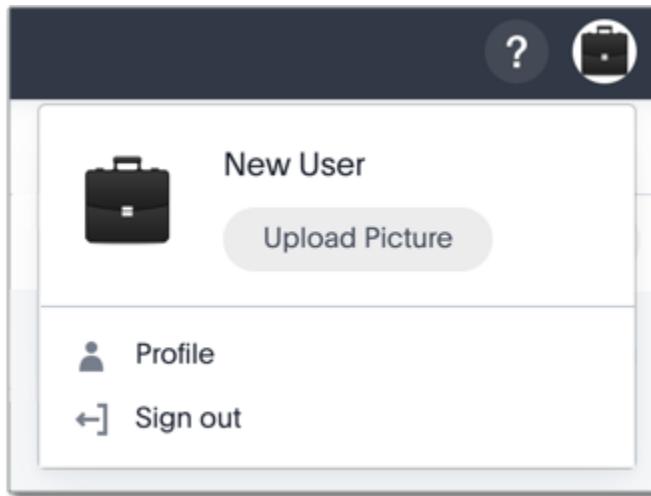
1. Click your user icon at the top right hand corner of the screen.
2. Click **Sign out**.



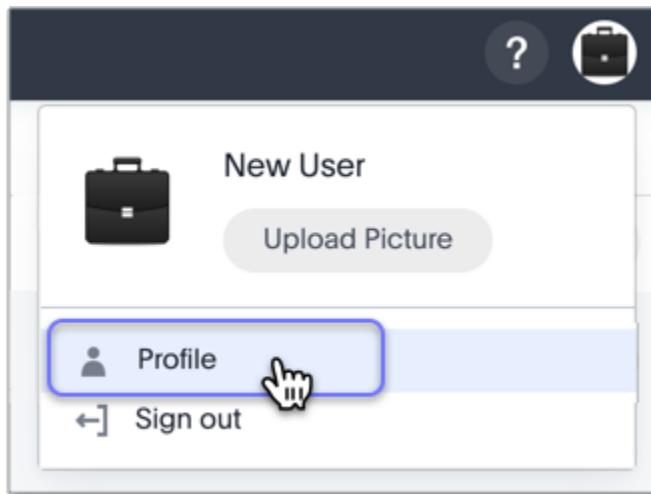
About your user profile

Summary: The user icon lets you view your profile, adjust language options, specify notification preferences, revisit onboarding, or sign out of ThoughtSpot.

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



Click **Profile** to navigate to your profile, where you can change your icon, password, email, locale, and other preferences.

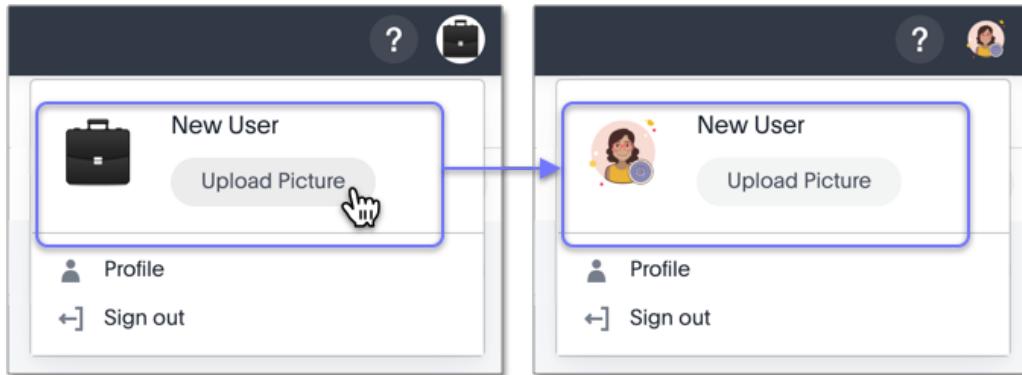


Notice that you can now see the **My Profile** interface.

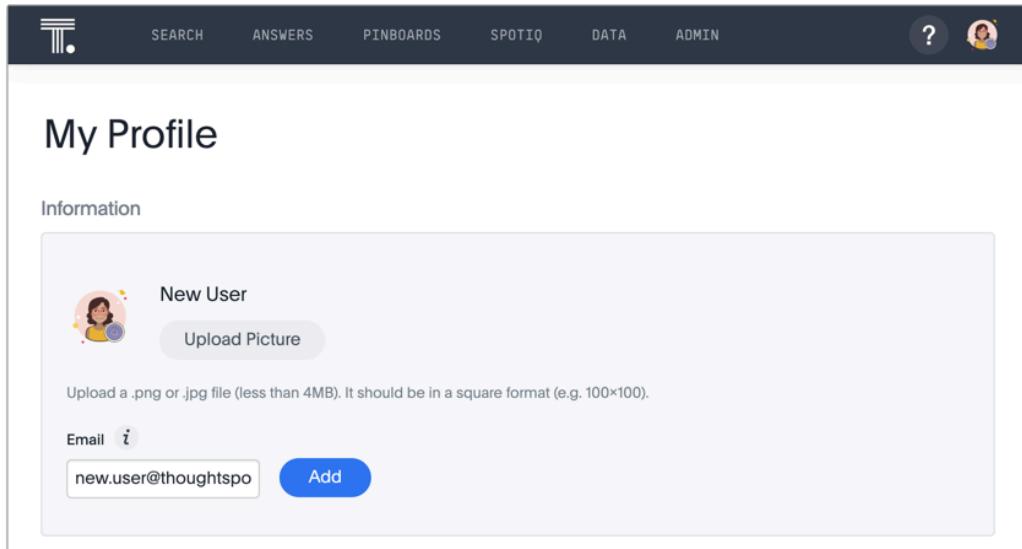
Change your profile picture

You can change your profile picture clicking the user profile in the top right of the interface, clicking **Upload Picture**, and selecting a new image from the file browser.

You can use any a `*.png` or `*.jpg` file. Image files cannot exceed 4MB.



You can also make this change within the **User Profile**, under **Information**.



SEARCH ANSWERS PINBOARDS SPOTIQ DATA ADMIN

My Profile

Information

New User

Upload Picture

Upload a .png or .jpg file (less than 4MB). It should be in a square format (e.g. 100x100).

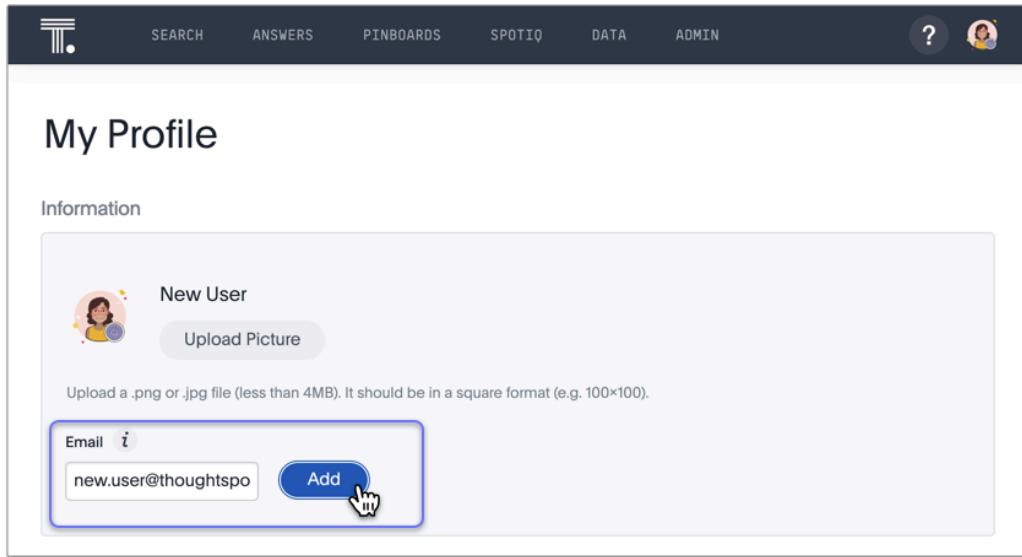
Email *i*

new.user@thoughtspo

Add

Update your email

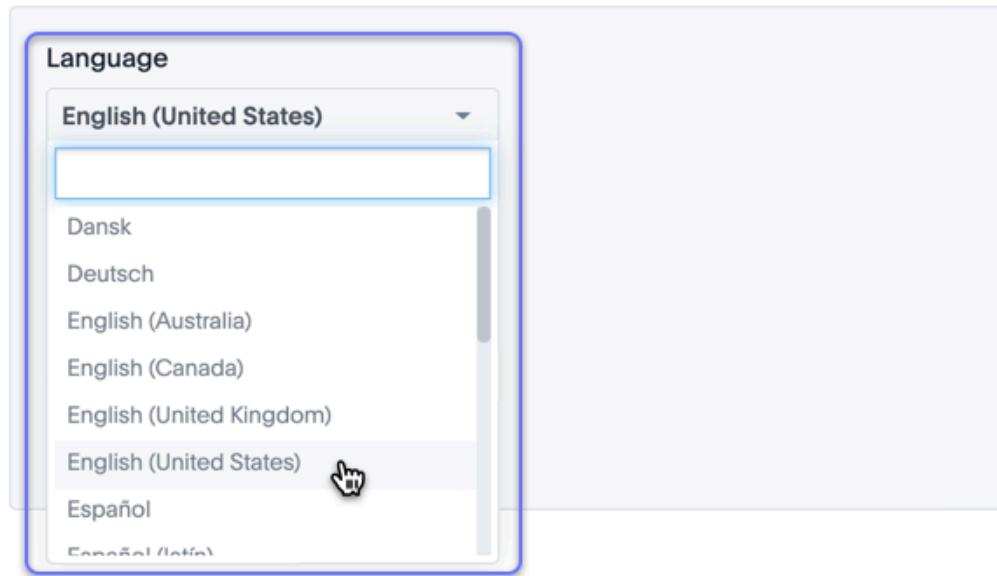
To change your email, navigate to **My Profile**. Under **Information**, enter an alternate email, and click **Add**.



Change language and data format settings

To change your language, navigate to **My Profile**. Under **Preferences**, see the **Language** option to set the language.

Preferences



By default, the language that ThoughtSpot UI displays depends on the system locale. It is simple to change it using the **Profile** interface.

The **Language** selection specifies more than just the language: it sets the locale, which controls both the language choice and standard data formats for date and number. So, if you set French as the default locale in your profile settings, the interface updates to reflect this. Be sure to refresh your browser page.

For example, in the United States the number format for large numbers uses the comma thousands separator and a period decimals separator, and looks like this: `xxx,xxx.xx`. In most European countries, they use the reverse notation, with comma decimals separator and period thousands separator, like this: `xxx.xxx,xx`.

In addition to American English (`en-US`), ThoughtSpot supports the following locales:

Locale	Language
<code>da-DK</code>	Dansk
<code>de-DE</code>	Deutsche
<code>en-AU</code>	English (Australia)
<code>en-CA</code>	English (Canada)

Locale	Language
en-GB	English (United Kingdom)
en-US	English (United States)
es-US	Español (latín)
es-ES	Español (España)
fr-CA	Français (Canada)
fr-FR	Français (France)
it-IT	Italiano
nl-NL	Nederland (beta)
nb-NO	Norsk
pt-BR	Português (Brazil)
pt-PT	Português (Portugal)
fi-FI	Suomi
sv-SE	Svenska
zh-CN	中文(简体)
ja-JP	日本語

ThoughtSpot translates keywords, operators, and error messages. See the [keyword reference for all supported languages](#).

ThoughtSpot *DOES NOT* translate formulas, or metadata entered by the user. For example, if you name a visualization ‘Quarterly Sales’ in any variant of English and subsequently change the locale to a variant of French, the visualization remains ‘Quarterly Sales’ and does not become ‘Ventes trimestrielles’.

Update sharing notification settings

Under **Preferences**, see the **Notifications** option.

Select **Email me sharing notifications** to receive emails whenever another user shares pinboards with you.

Preferences

Language

English (United States) ▾

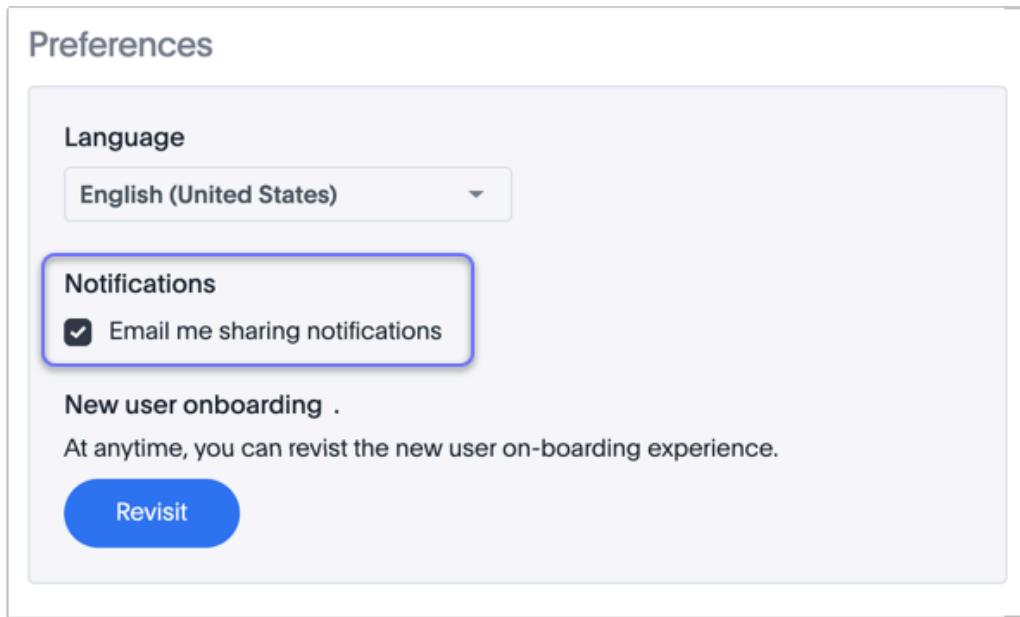
Notifications

Email me sharing notifications

New user onboarding .

At anytime, you can revisit the new user on-boarding experience.

Revisit



Revisit user onboarding

When you are relatively new to using ThoughtSpot, we help you to become productive faster.

Whenever you need a refresh, navigate to **My Profile***. Under ****Preferences**, see the **New user onboarding** option. Click **Revisit**, and ThoughtSpot guides you through onboarding again.

Preferences

Language

English (United States) ▾

Notifications

Email me sharing notifications

New user onboarding .

At anytime, you can revisit the new user on-boarding experience.

Revisit



Change Password

When you need to change your password, navigate to **My Profile**, and under **Preferences** enter the following information:

- Current Password
- New Password
- Confirm Password

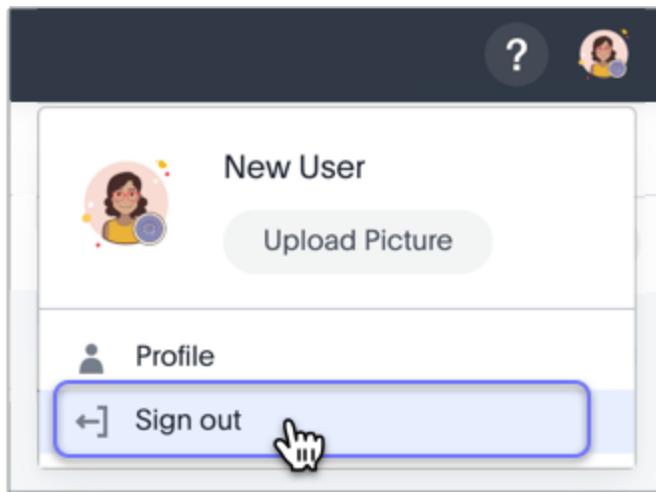
Click **Update**.

Update password

The form has a light gray background with rounded corners. It contains three text input fields, each with a placeholder line. The first field is labeled "Current Password", the second "New Password", and the third "Confirm Password". Below these fields is a blue, rounded rectangular button labeled "Update".

Signing out

To sign out of your ThoughtSpot account, click the user profile in the top right of the interface, and choose **Sign out**.



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 answers 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:

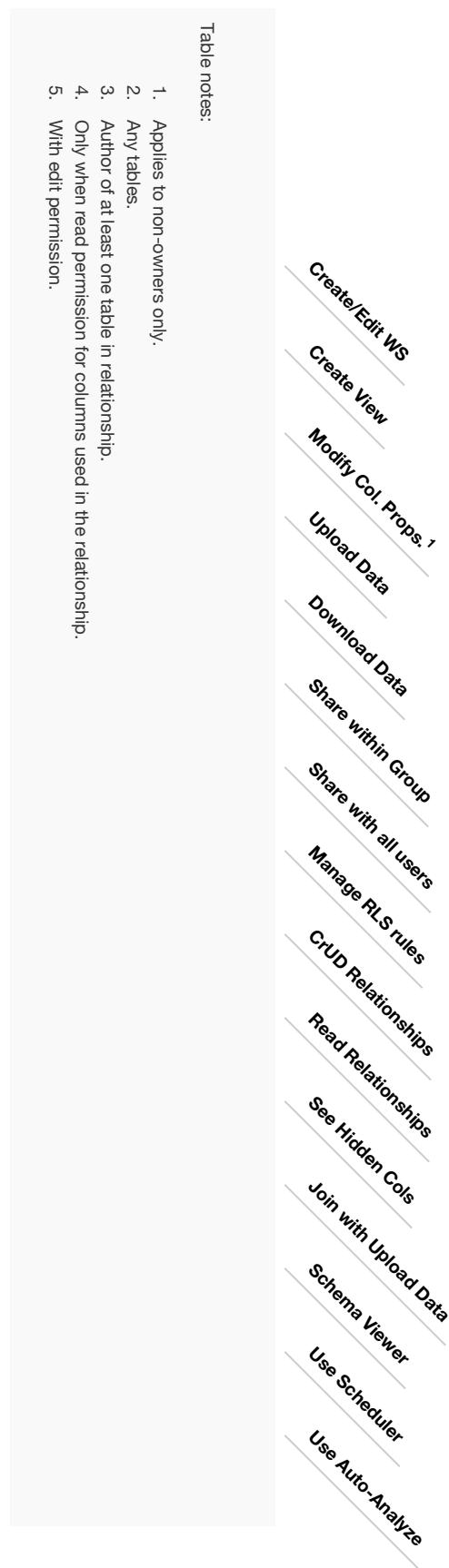
Privilege	Description
Can administer ThoughtSpot	Can manage Users and Groups and has view and edit access to all data. Users with this privilege can also download a saved answer.
Can upload user data	Can upload their own data from the application's Data page using Actions > Upload data .
Can download data	Can download data from search results and pinboards.
Can share with all users	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 NOT SHAREABLE .
Can manage data	Can create worksheets and views. Note that to edit a worksheet or a view created by another user, you must have the **Edit** permission on that object, and it must be shared with you.
Can use experimental features	Can access trial and experimental features that ThoughtSpot makes available to early adopters.
Can invoke Custom R Analysis	Can access R scripts to further explore search answers. Includes options to invoke R scripts on visualizations, create and share custom scripts, and share the results of R analysis as answers and pinboards.
Can schedule pinboards	Can create pinboard schedules and edit their own scheduled jobs.

Privilege	Description
Can administer and bypass RLS	<p>Users in groups with this privilege (directly or through group inheritance):</p> <ul style="list-style-type: none">• Are exempt from row-level security (RLS) rules.• Can add/edit/delete existing RLS rules.• Can check or uncheck Bypass RLS on a worksheet. <p>Your installation configuration may enable or disable the availability of this privilege. By default, it is enabled. Administrators or groups with the privilege Can administer ThoughtSpot can grant this privilege.</p>

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 View	Modify Col. Props. ¹	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 Scheduler	Use Auto-Analyze
Can administer ThoughtSpot	Y	Y	Y	Y	Y	Y	Y	Y ²	Y	Y	Y	Y	Y	Y	Y
Can upload user data	N	N	N	Y	N	Y	N	Y ³	Y ⁴	N	N	N	N	N	N
Can download data	N	N	N	N	Y	Y	N	N	N	Y ⁴	N	N	N	N	N
Can manage data	Y	Y	Y	Y	N	Y	N	Y ⁴	Y ⁴	Y ⁵	Y	N	N	N	N
Can share with all users	N	N	N	N	Y	Y	N	N	Y ⁴	N	N	N	N	N	N
Can Auto-Analyze (SQL privilege)	N	N	N	N	N	N	N	Y ⁴	N	N	N	N	N	N	Y
Can Administer and Bypass RLS	N	N	N	N	N	N	N	Y	N	N	N	N	N	N	N
None	N	N	N	N	Y	N	N	N	N	N	N	N	N	N	N



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 the currently selected sticker, scroll to the bottom of the list, and click **+ Add**.

The screenshot shows a list of four documents:

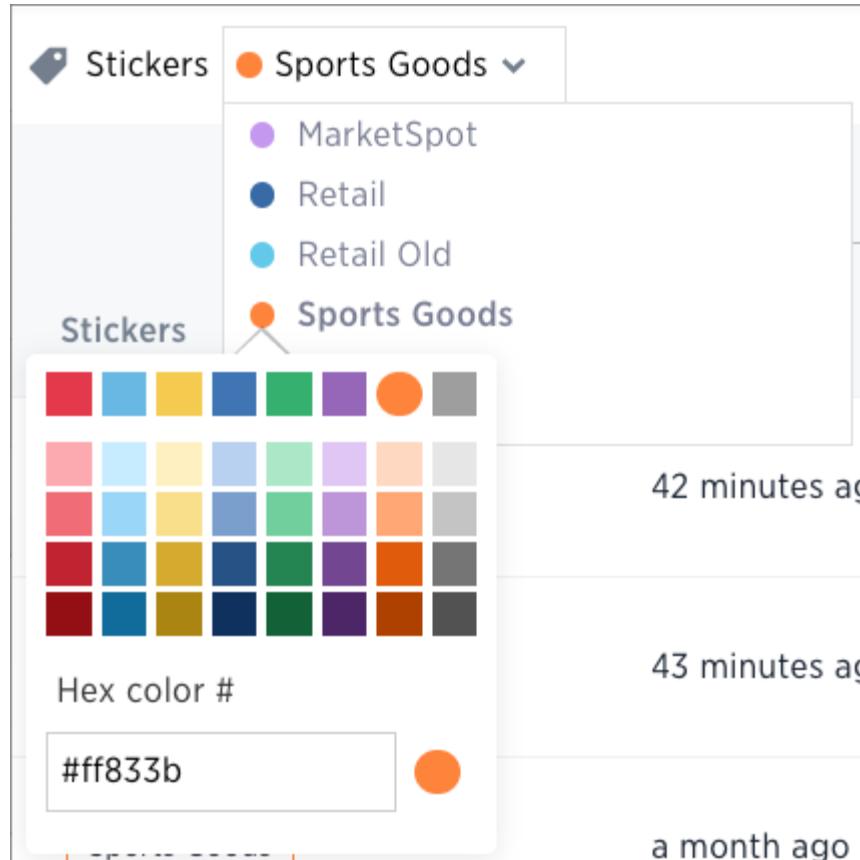
- ThoughtSPORT Overview: Sticker: Sports Goods, 9 minutes ago
- Advanced Analysis with R: Sticker: Sports Goods, 9 minutes ago
- ThoughtSPORT Analysis: Overview of Advanced Formulas in Thoughtspot, Sticker: Sports Goods, a month ago
- Comparative Analysis: Sticker: Sports Goods, a month ago

A dropdown menu for the "Sports Goods" sticker is open, showing other options: MarketSpot, Retail, Retail Old, and Sports Goods. The "+ Add" button is highlighted with a blue box.

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

A dropdown menu for a "Sports Goods" sticker is open, showing two options: "Edit name" and "Remove sticker".

5. You can change the color of a sticker by clicking the color circle next to its name.

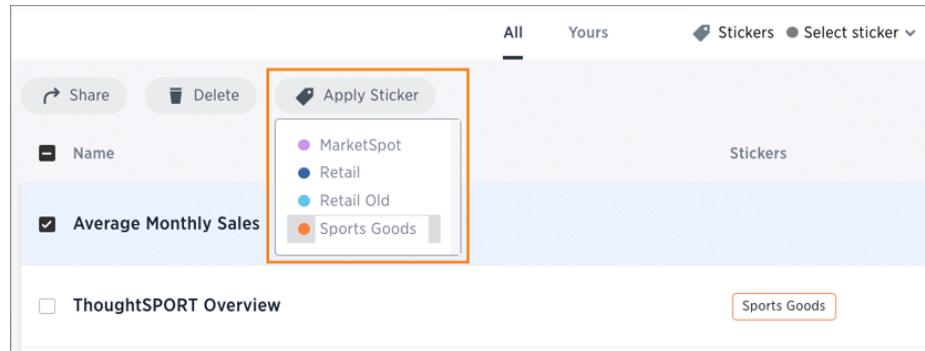


Apply a sticker

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.



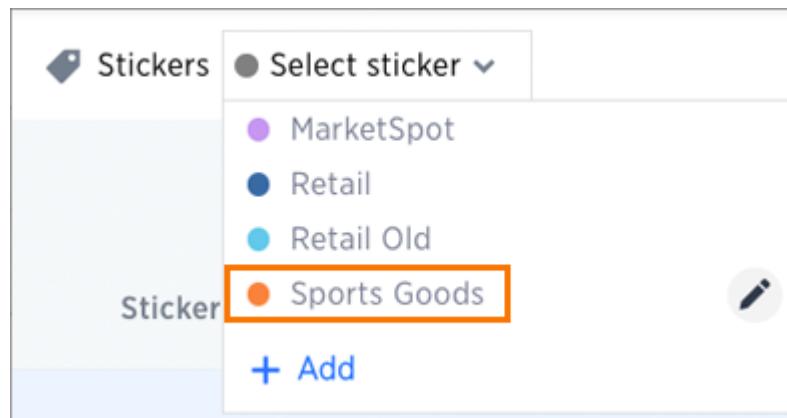
Filter by stickers

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 **Select sticker**, and select the name of the sticker you want to filter by.



What are Searches and Answers?

Summary: You use search to 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.

You can see an answer in the form of a chart or a table. If you do not see an answer, press **Enter** on your keyboard.

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

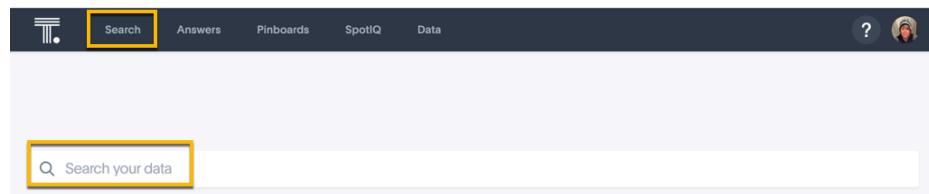
- [Choose a data source](#)
- [Search bar features](#)
- [Search suggestions](#)
- [Search results and column types](#)
- [Share an answer](#)
- **Beta** [About SearchIQ](#)

Start a new search

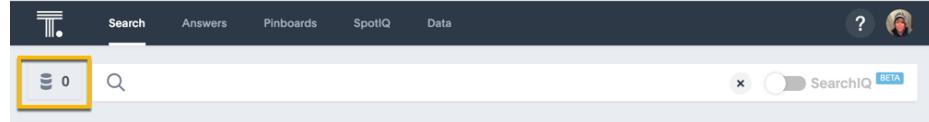
Summary: Starting a new ThoughtSpot search is simple, like starting a new Google search.

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. Click the Data button, to the left of the search field.



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

The screenshot shows the ThoughtSpot interface with the 'Data' tab selected. A modal window titled 'Choose Sources' is open. At the top left of the modal is a yellow box containing the text 'Choose Sources'. At the top right is a 'CLOSE' button. The modal is divided into sections: 'WORKSHEETS' (MarketSpot Bootcamp, MarketSpot Worksheet, Phone Sales, Phone Sales, Foot T.., Retail Old, Sales, Email Performance, Sporting Goods Retail, TS: Database, TS: Metrics, TS: Search, West Region Sales), 'VIEWS' (Top 100 Products M.., TS: Internal Capacity, TS: Internal Table W.., TS: Table Info, TS: Table Row Counts, TS: Table Row Count, TS: Table Row Size, TS: Table Shards), and 'IMPORTED DATA' (Austin_Animal_Center, credit cards, Customer_Dimension, Dates_Dimension). On the left side of the modal, there is a sidebar with a legend: MarketSpot (purple dot), Retail (dark blue dot), Retail Old (light blue dot), and Sports Goo.. (orange dot). Below the sidebar is a link 'EXPLORE ALL DATA'.

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 **CLOSE**.
-
4. 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 by clicking each column to select it, and then clicking **+ Add Columns**.

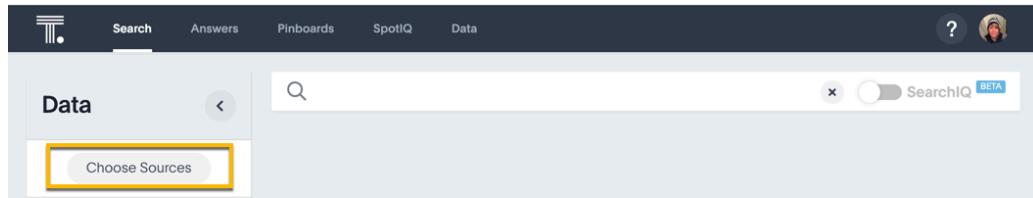
The screenshot shows the ThoughtSpot Data interface. At the top, there is a navigation bar with tabs: Search, Answers, Pinboards, SpotIQ, and Data. On the far right of the top bar are a help icon, a user profile icon, and a sign-out button. Below the navigation bar, the word "Data" is displayed in a large font. To the left, a sidebar titled "Choose Sources" is open, showing a list of columns from a dataset named "ThoughtSPORT Worksheet". The list includes: Age Group, Annual Income, Cost_Dollar_Amount, Customer Address, Customer Age, Customer Age Group, Customer City, Customer County, Customer Gender, Customer Name, Customer Region, Customer Since, Customer State, and Customer Type. A button labeled "+ Add Columns" is at the bottom of this list, with a yellow box drawn around it. To the right of the sidebar, there is a search bar with the placeholder "Search your data" and a "SearchIQ BETA" toggle switch. The main area of the interface is currently empty.

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](#).



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.

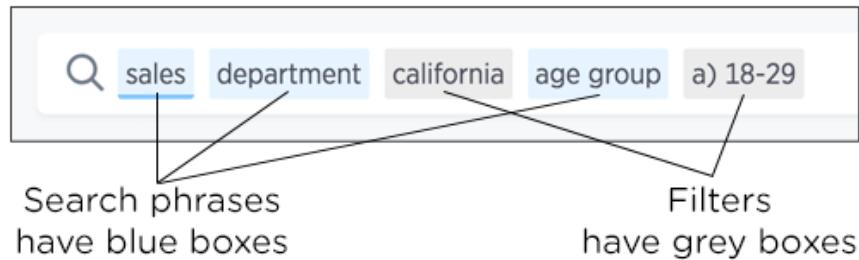
Search bar features

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.



Editing a search

You can edit a search without affecting existing search results. When you click a search phrase, it is highlighted, and ThoughtSpot displays other suggestions from which you can choose to replace the highlighted phrase. When you hover over a boxed phrase, you can see an **x**, which you can click to remove it from the search. You can insert a new phrase in the middle of a search, by clicking between phrases and entering the new phrase. You can even merge phrases without breaking the search. As you edit your search, ThoughtSpot continuously generates new results.

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.

Quick select search results

When you type search phrases in the search bar, the first suggestion is automatically highlighted and you can use tab to navigate further. In addition, after you type a phrase or a letter and press **Enter**, the highlighted suggestion is automatically selected. Suggestions can be an attribute, measures, value, keyword, historic query, or exactly what you typed as the search phrase. You can also press **Enter** or **Tab** to select the first suggestion. If there are no suggestions, the suggestion list will not be displayed. Historic suggestions always display on top with the first suggestion highlighted as default.

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.

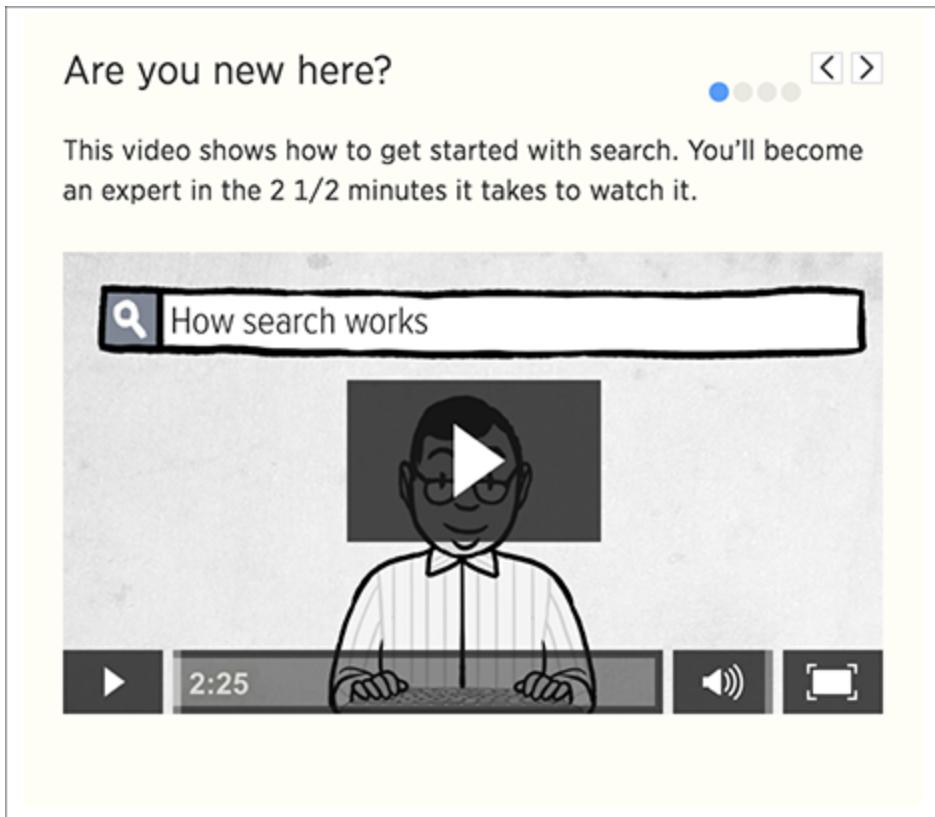
Dictionary synonyms

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.



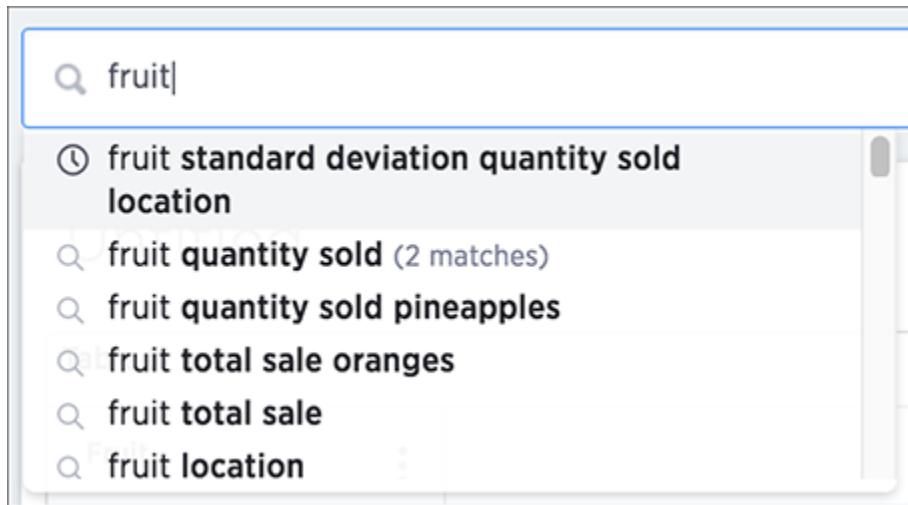
Related searches

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 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.



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.

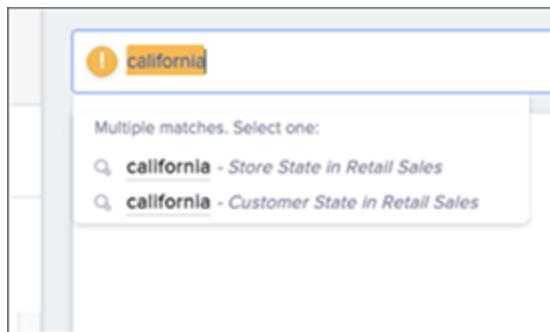
UBR based on search patterns makes ThoughtSpot more valuable over time. Search suggestions become more helpful the more searches you complete.

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 can 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

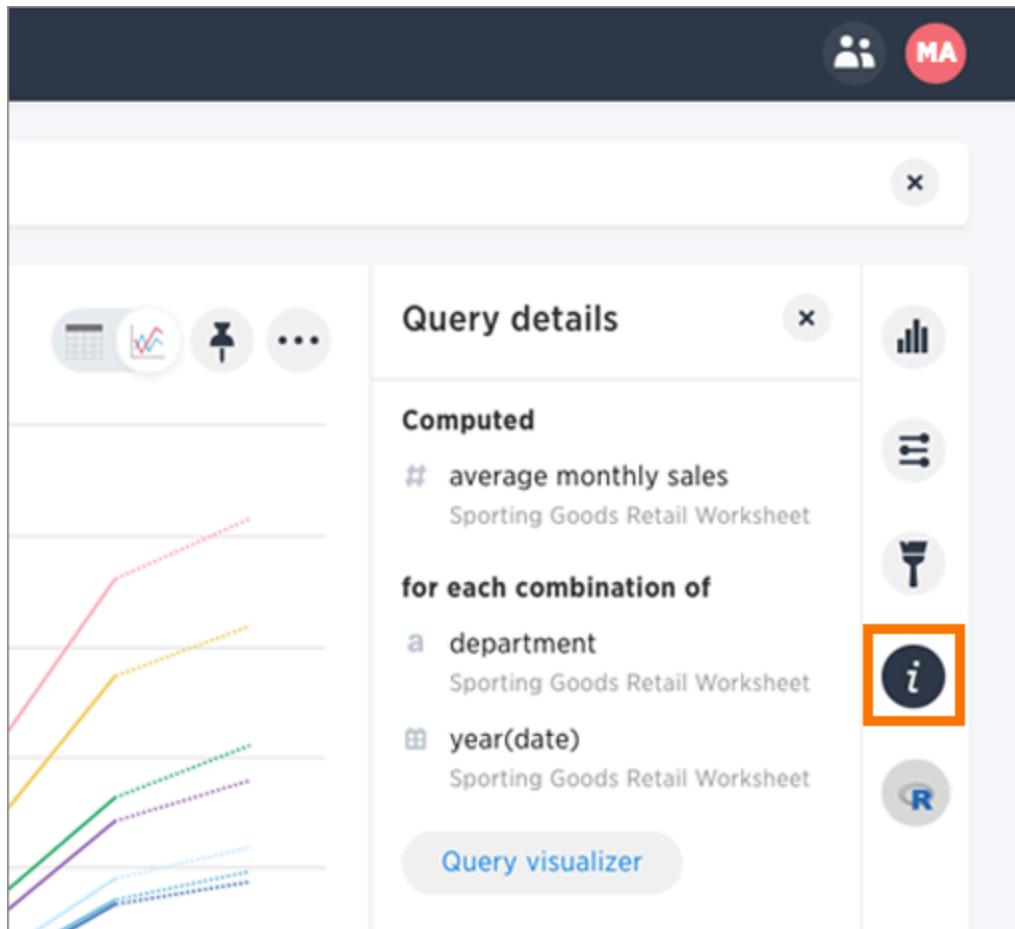
Attribute are primarily text or date values. Attributes make up the x-axis of your chart most of time. Some examples of attributes in terms of a person include name, eye color, occupation, social security number, address, employee ID, and phone number.

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 **Query details** button on the right-hand side of the screen to open the **Query details** panel.



This panel shows the measures that 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 a list of imported data sources under the 'IMPORTED DATA' section. One item, 'Customer_Dimension', has a tooltip displayed over it. The tooltip contains the following information:

- NAME: Customer_Dimension
- AUTHOR: Andrew Yeung
- CREATED: a year ago
- UPDATED: 3 months ago
- DATABASE: ImportedDatabase
- SCHEMA: Retail Data from SQL Server - AY

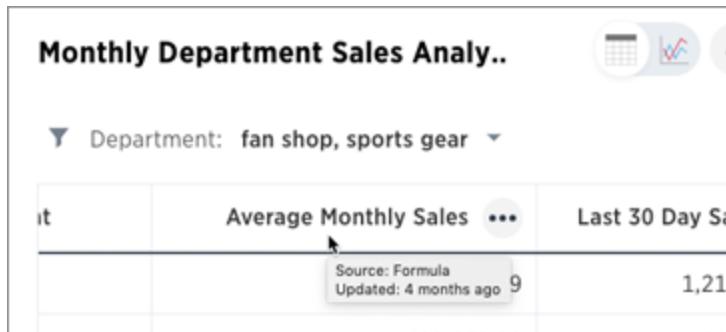
You can also hover over a column in the **Data** column to see when it was last refreshed. This popup will also show sample values from that column.

The screenshot shows a detailed view of a data source. On the left, there is a sidebar with a 'Choose Sources' button and a search bar. The main area displays a table titled 'Monthly Department Sales Analy'. A tooltip is shown for the 'Sales' column, containing the following information:

- Name: Sales
- Source: Sales_Dollar_Amount
- (ThoughtSPORT_Retail_Sales_Fact)
- Updated: 4 months ago
- Data type: DOUBLE

Below the table, there are two columns: 'Monthly Sales' and 'Last 30 Day Sales'. The 'Monthly Sales' column contains the value 944,665.99, and the 'Last 30 Day Sales' column contains the value 1,216,6.

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



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 and edit an answer

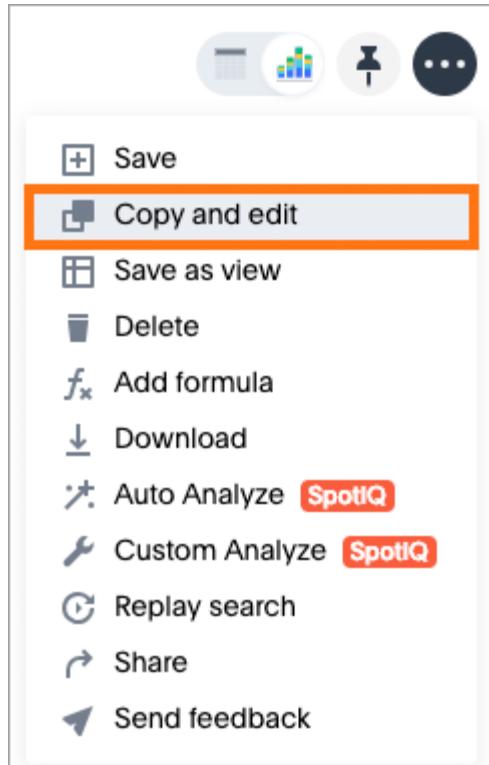
You can edit a copy of an answer if you would like to make edits without changing the original answer.

When saving, you can type in a different name for your copy.

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



2. On the answer list page, click the answer you would like to edit a copy of.
3. Click the ellipses icon and select **Copy and edit**.



A copy of the answer is displayed with “Copy of” added to the beginning of the title.

4. Make any changes you want to the answer.
5. Click the ellipses icon  and select **Save**.

The Save Answer dialog box appears.

Save Answer

Name

Description

[Cancel](#) [SAVE](#)

6. Change the name and description, as needed, and click **SAVE**.

When you navigate back to the Answers page, your edited Answer will appear in the list.

Other features

At the bottom of the **Answers** page, you can see an additional area, called **SpotIQ Insights**. This area is derived from your original answer and you can select any of the insights to deepen your search.

The screenshot shows the 'Did you know?' section of the ThoughtSpot interface. It contains eight insights arranged in two rows of four. Each insight includes a title, a detailed description, and a small icon.

- SPOTIQ ANOMALY**: For West(Store Region), Store Zip Code **98108** has 7.9 times the average % Gross Margin. (Icon: Global Sales)
- SPOTIQ ANOMALY**: For Athletics(Sport), Country Code **USA** has 91.9 times the average Total Won Gold. (Icon: Olympic Medals History)
- SPOTIQ ANOMALY**: Date **JULY 2018** has 2.7 times the average Total Sales. (Icon: Advanced Analysis with R)
- SPOTIQ ANOMALY**: For East(Store Region), Store Zip Code **10199** has 6.7 times the average % Gross Margin. (Icon: Global Sales)
- SPOTIQ ANOMALY**: For Athletics(Sport), Country Code **USA** has 53.9 times the average Total Won Silver. (Icon: Global Sales)
- SPOTIQ TREND**: **Total Sales** is overall trending upwards. (Icon: Global Sales)
- SPOTIQ ANOMALY**: For South(Store Region), Store Zip Code **38101** has 5.3 times the average % Gross Margin. (Icon: Global Sales)
- SPOTIQ ANOMALY**: For Athletics(Sport), Country Code **USA** has 38.7 times the average Total Won Bronze. (Icon: Global Sales)

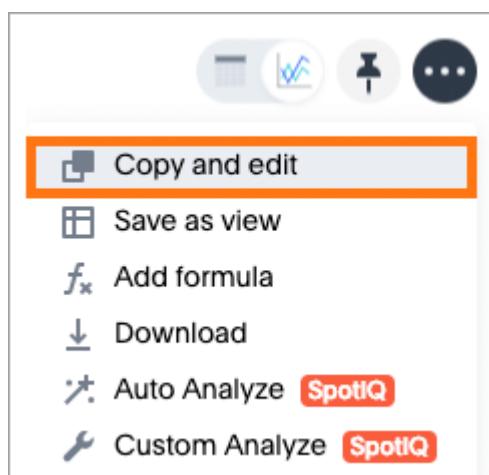
At the bottom right of the section, there is a small logo for THOUGHTSPOT.

The **SpotIQ Insights** section is only visible to users with the **Has SpotIQ** privilege. After 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.

Copy and edit a SpotIQ Insight

You can edit a copy of a SpotIQ Insight if you would like to make edits without changing the original insight. When saving, you can type in a different name for your copy.

1. Find the SpotIQ insight you want to edit and click it to open it as an answer.
2. Click the ellipses icon and select **Copy and edit**.



A copy of the answer is displayed with “Copy of” added to the beginning of the title.

3. Make any changes you want to the answer.
4. Click the ellipses icon  and select **Save**.

The Save Answer dialog box appears.

5. Change the name and description, as needed, and click **SAVE**.

Ask an expert

Summary: Do you need help finding something in your data? Ask an expert to create the search for you.

At times you may struggle to answer a specific question, build a chart you have in mind, or find something in your data. You can use **Ask an expert** to request help from someone in your organization. Your request will go to the person in your company who best know the data source you're using and how to search it in ThoughtSpot. When your request is answered, you will be able to see the result in the **Answers** listing in ThoughtSpot.

Note: If you don't see **Ask an expert** it's possible that your company has chosen to disable this capability. If you want to try it out, ask your administrator to consider enabling it .

Ask an expert can help in these ways:

1. You can get an answer to your tough question, even when you're stuck.
2. The expert gets to see what people have trouble with, so he or she can make it easier.
3. You can use the answer from the expert to understand the data better.

How do I use Ask an expert?

1. When you're doing a search, you can see a link under your search that says **Ask an expert**. Click that if you're stuck and need help.
2. The search you were working on is captured and shown back to you, with a text box for your message. Send a message describing what you were looking for, along with any information that makes it easy for the expert to understand what problem you were having.

Ask an expert



David Cohen will get right back to you.

What you searched for

sales per customer by store region date weekly last 3 months

Can you explain in a little more detail?

I couldn't get the totals to display on the chart.

Cancel Send

An example would be:

"I'm having trouble getting this search to show as a stacked column, with each department shown as a separate color."

1. Then click **Send**, so your request will go to the expert.
2. The expert will share their answer with you. You can then [Review answers from Ask an expert](#).

What kinds of things can I ask?

You can use **Ask an expert** to ask any kind of question you like. Even if your question is about some data that isn't in ThoughtSpot at all, the expert may be able to help, so it's always worth asking.

Related information

- [Review answers from Ask an expert](#)
- [Answer Ask an expert questions](#)

Manage expert requests

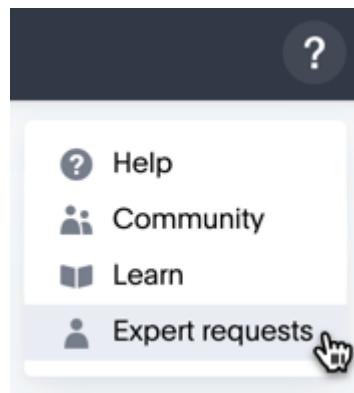
Summary: You can monitor and manage Ask and Expert requests, both as a requester and as an expert.

Within ThoughtSpot, you can keep track of your **Ask an Expert** requests.

Navigate to the Expert Requests interface

Follow these steps to the request monitoring interface:

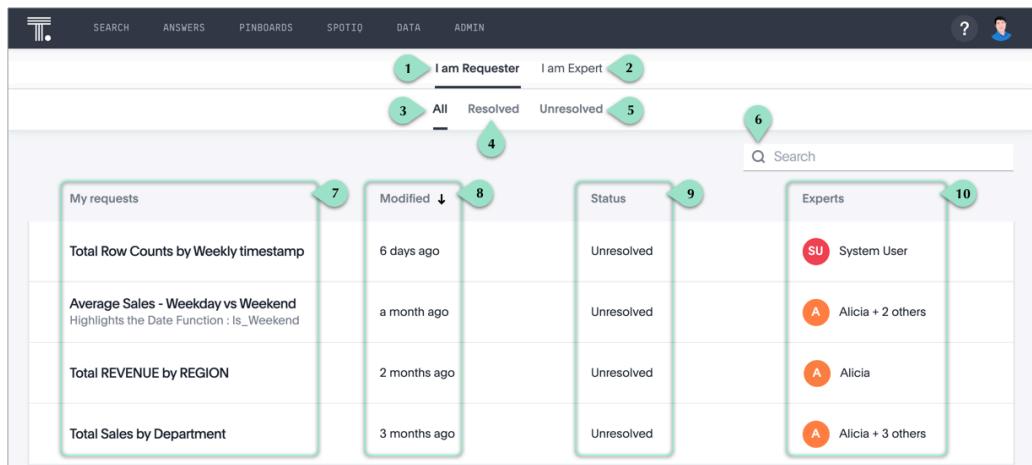
1. In the top right corner of the ThoughtSpot interface, click the  (help) icon menu.
2. In the drop-down, select **Expert Requests**.



3. The **Expert** interface appears, with information for you both as a **Requester**, and as an **Expert**.

When you are a Requester

This is what you see when you are a requester:



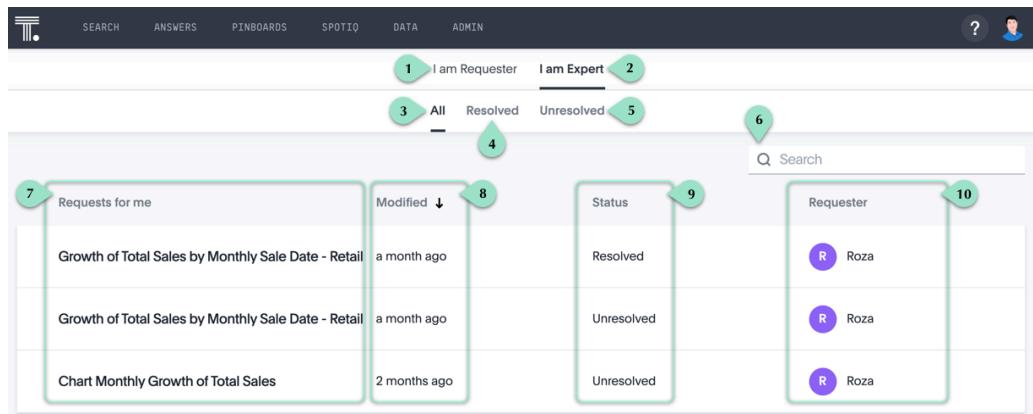
Note these features of the interface:

Legend Feature

1. **I am Requester:** for all requests where you are a requester
2. **I am Expert:** for all requests where you are an expert
3. **All:** all requests, both *Resolved* and *Unresolved*
4. **Resolved:** all resolved requests
5. **Unresolved:** for all unresolved requests
6. **Search:** find requests
7. **My Requests:** list of your requests
8. **Modified:** how long ago you or an expert modified the request
9. **Status:** *Resolved* or *Unresolved*
10. **Experts:** users who are experts for your requests

When you are an Expert

This is what you see when you are an expert:



Note these features of the interface:

Legend Feature

1. **I am Requester:** for all requests where you are a requester
2. **I am Expert:** for all requests where you are an expert
3. **All:** all requests, both *Resolved* and *Unresolved*
4. **Resolved:** all resolved requests
5. **Unresolved:** for all unresolved requests
6. **Search:** find requests
7. **Requests for me:** list requests where you are the Expert
8. **Modified:** how long ago you or an expert modified the request
9. **Status:** *Resolved* or *Unresolved*
10. **Requester:** the user who initiated the request

Related information

- Ask an Expert

View an answer from an expert

Summary: How to find an answer from an expert

When you use **Ask an expert** to get help, you can view the answer after the expert responds.

To view the answer:

Click the link in the email you receive automatically when the expert answers your question, or if you're not set up with an email in ThoughtSpot:

1. Click **Answers** in the top menu.
2. Click the name of the answer you want to review. If you can't find it, try using the **Yours** filter to narrow down the list.
3. View your answer.

The answer will include any changes the expert made, to get to the answer you were looking for. You should review the new answer, to learn from the things the expert did to answer your question.

Note that the answer is a full fledged ThoughtSpot answer. This means that you can pin it to a pinboard, edit it, or share it with others, just like the result of any ThoughtSpot search.

Related information

- [Ask an expert](#)
- [Answer Ask an expert questions](#)

Answer an expert question

Summary: You're the expert. Now what? Here's how to answer a request from Ask an Expert.

When a user struggles to answer a specific question, build a chart you have in mind, or find something in the data, they can use **Ask an expert** to request help. The request will go to the designated expert for the data source they were searching. The user will be able to see the result in the **Answers** listing in ThoughtSpot.

How do I answer a question from Ask an Expert?

1. You receive an email stating that your help is needed, with a link to the request.

If you do not have an email set up in ThoughtSpot, you must do this to check for requests:

- a. Log in to ThoughtSpot.
- b. Click **Answers** in the top menu.
- c. On the top right hand side, click the ellipses and choose **View Requested Answers**. You can see a listing of all requests for help.

2. When you choose a request to answer, you can see the answer the user was working on. You are now in "expert mode". If the user provided an explanation of what they were looking for, you can see that under the visualization.



1. Work on the answer until it looks like what the user wanted to see. Then click **Close**.
2. If the requester has an email address in ThoughtSpot, an email will be sent to them with a link to the answer. If not, you must share a link to the answer with them using email or a direct message. A link to the answer is provided, so you can easily share the answer.

Mark as answered

Share this answer with Administrator before marking it as answered:

<https://seed.corp.thoughtspot.com/#/saved-answer/dcf1f9b0-ea01-430c-be3a-8aa57f42647c>

[Copy link](#)

[Later](#) [MARK AS ANSWERED](#)

1. When you've answered the question (and copied the link, if needed), you can mark the request answered by clicking **Mark as Answered**.
2. If you want to see the answer again, you can find it listed along with the other answers. Click **Answers** in the top menu to see the list of saved answers.

Related information

- [Review answers from Ask an expert](#)
- [Answer Ask an expert questions](#)

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 can 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 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.

Monthly Department Sales Analysis			
Department: fan shop, sports gear			
Yearly (Transaction Date) ↓	Department	Average Monthly Sales	Last 30 Day Sales
2017	Fan Shop	944,665.99	1,154,842.42
2017	Sports Gear	489,659.92	594,130.43
2016	Fan Shop	760,579.94	764,598.82
2016	Sports Gear	387,817.03	400,748.20
2015	Fan Shop	440,016.37	412,678.07
2015	Sports Gear	213,930.58	199,212.95
2014	Sports Gear	201,775.80	198,997.84
2014	Fan Shop	412,574.38	415,234.76

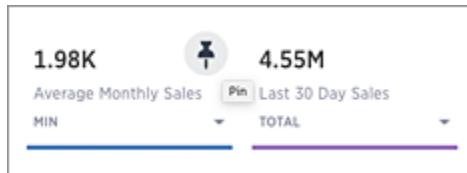
{ showing rows 1-10 of 12 }

2012 - 2017 Yearly (Transaction Date)	2 Department UNIQUE COUNT	1.98K Average Monthly Sales MIN	4.55M Last 30 Day Sales 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 the headline and click the **Pin** icon.



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

Table aggregate headline

The Table Aggregate headline option is available when an aggregate function is used either through a formula or a search bar query like average of a measure. It recalculates the function for the entire table. In such cases, Table Aggregate is shown by default in a headline under the table, instead of the "Avg" option, which does a second level of aggregation on top of the existing aggregation.

In the following example, the table aggregate is shown as a result of the Average Profit formula `sum (profit) / count (ship mode)`, which divides the total profit of each ship mode by the sum total count for that ship mode. Table Aggregate, recalculates that function for the entire table taking the sum total profits of all ship modes and dividing it by the sum total count of all ship modes, providing a table aggregate average profit of 181. This is a more useful result compared to the Avg headline option which would simply sum the average profit for all ship modes and divide it by the number of ship modes (3), providing an average of 187.

The screenshot shows a search results page with the following components:

- Search Bar:** At the top left, there is a search input field containing "ship mode" and several buttons labeled "count ship mode", "profit", and "average profit". To the right of the search bar is a "SearchIQ" toggle switch.
- Table Header:** Below the search bar, the title "Total Count Ship Mode, Total Profit, Average Profit by Ship Mode" is displayed. To its right is a toolbar with icons for chart types (line, bar, pie), export (CSV, PDF), and other actions.
- Table Data:** The main content area contains a table with four columns: "Ship Mode", "Total Profit", "Total Count Ship Mode", and "Average Profit". The data rows are:

Ship Mode	Total Profit	Total Count Ship Mode	Average Profit
regular air	1,104,692.79	6,270	176.19
delivery truck	269,644.87	1,146	235.29
express air	147,430.31	983	149.98
- Summary Bar:** At the bottom of the table area, there is a summary bar with four metrics:
 - Ship Mode Unique Count:** 3
 - Ship Mode Avg:** 2.8K
 - Profit Total:** 1.52M
 - Average Profit Table Aggregate:** 181The "Average Profit Table Aggregate" metric is highlighted with a red box.
- Right Sidebar:** On the far right, there is a vertical sidebar with several icons: a bar chart, a line graph, a funnel, a magnifying glass, an info icon, and a refresh/reload icon.

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 [reference](#) section or in the help. Open the help by selecting **Help** from the top navigation bar.

Keyword reference

Estimated reading time: 9 minutes

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.

Keywords in Other Languages

Currently, we offer the following keyword translations. We plan to add more incrementally.

日本語	中文 (简体)	Deutsche	Español (latín)	Français (Canada)	Français (France)	Português (Brasil)
-----	---------	----------	--------------------	----------------------	----------------------	-----------------------

On this page

- [Keywords in Other Languages](#)
- General
- Date
- Time
- Text
- Number
- Comparative
- Location
- Period

General

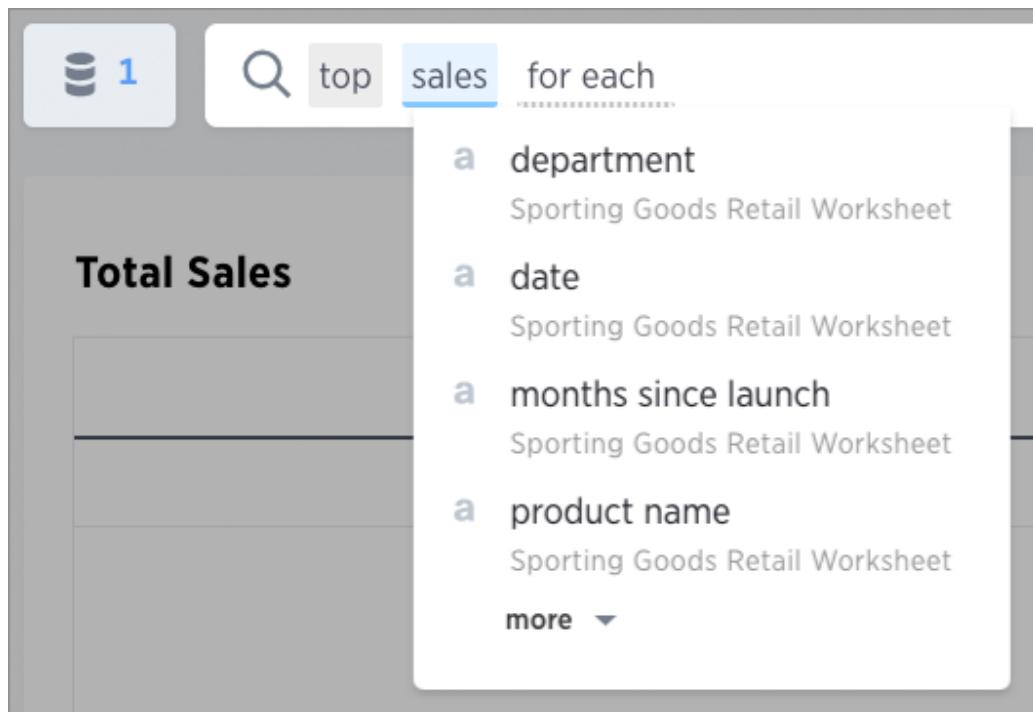
Keyword Examples

top	<ul style="list-style-type: none">• <code>top sales rep by count sales for average revenue > 10000</code>• <code>sales rep average revenue for each region top</code>
bottom	<ul style="list-style-type: none">• <code>bottom revenue average</code>

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.

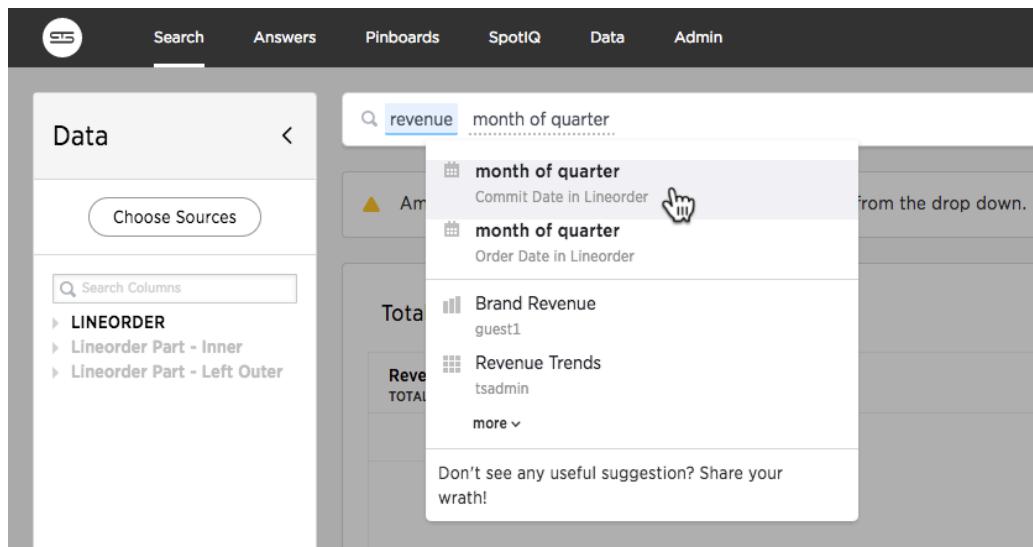
Related information

- [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)	<code>quarter</code> (purchase date)

quarter of year (date)	quarter of year (purchase date)
month of quarter (date)	month of quarter (purchase date)
week of year (date)	week of year (ship date)
week of quarter (date)	week of quarter (ship date)
week of month (date)	week of month (ship date)
day of year (date)	day of year (ship date)
day of quarter (date)	day of quarter (ship date)
day (date)	day (ship date)
day of month (date)	day of month (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.

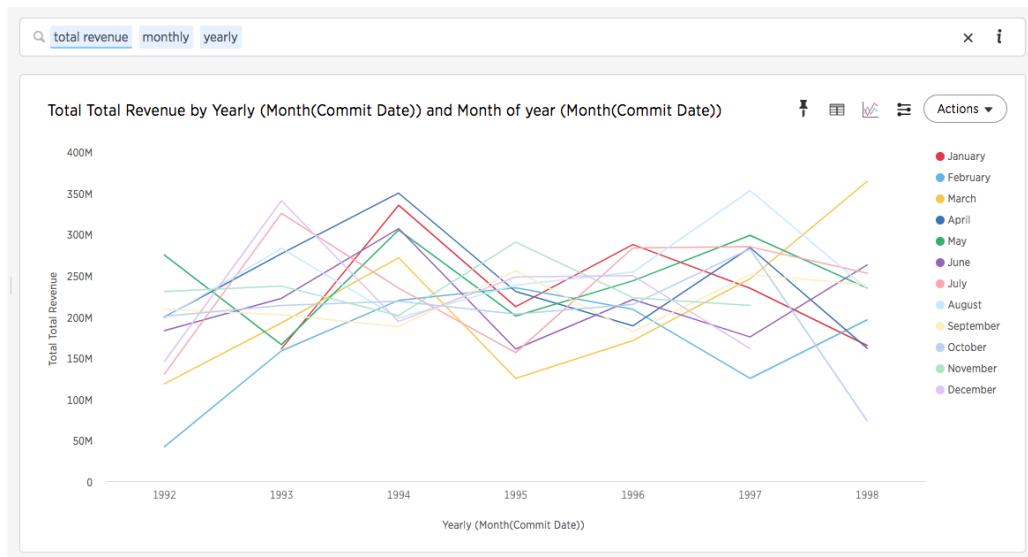
By default, the **Search** bar suggests these keywords less frequently than others.

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

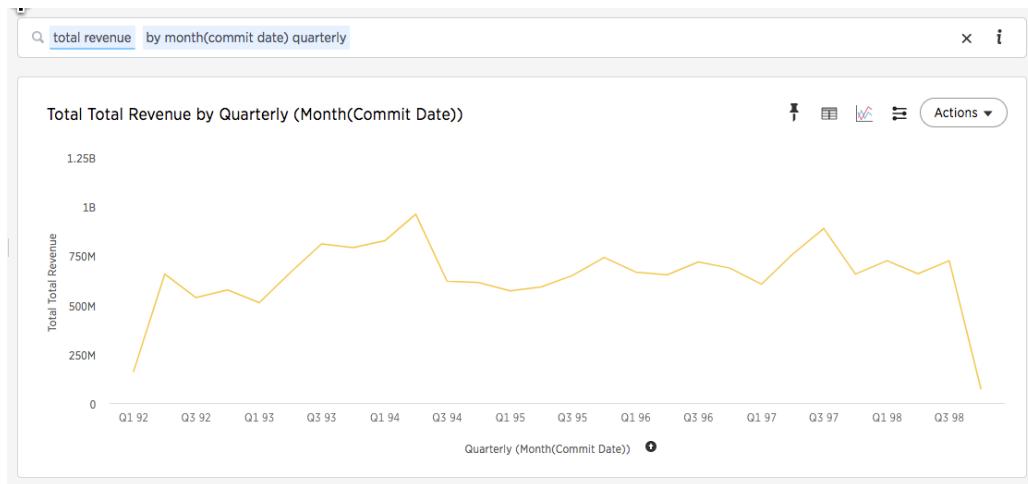
- `Detailed`
- `Hourly`
- `Daily`
- `Weekly`
- `Monthly`
- `Quarterly`
- `Yearly`

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.



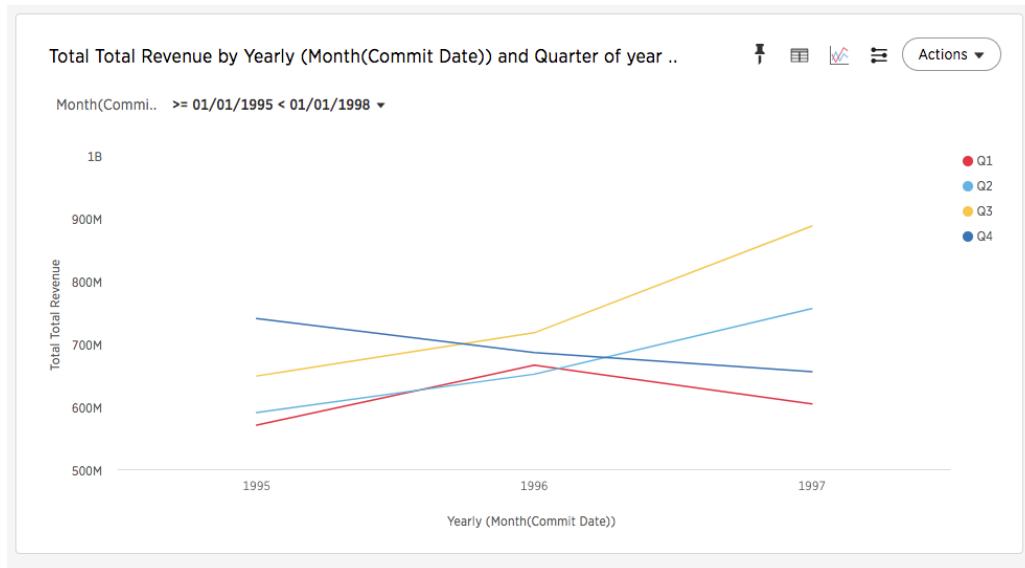
When you search for an aspect of data of time series, a typical response is a line chart showing the aspect as it rises and falls over time.



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.

Create a max(date) field and use it to filter

If you have a date field in your data set and want to return the most recent set of data based on a specific date, do the following:

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.

Search using growth over time

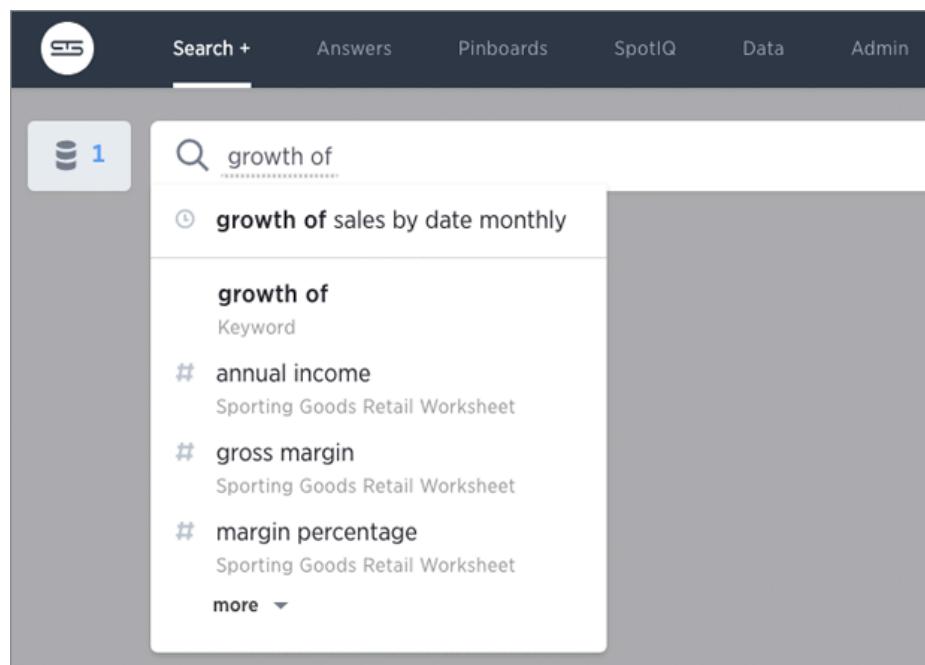
Summary: You can show growth over time by using the growth of keyword in your search.

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:

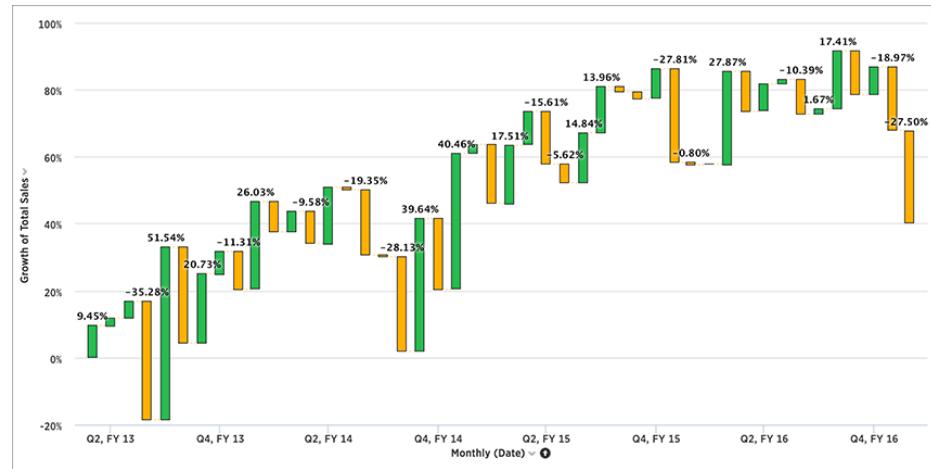
1. Type `growth of` into the search bar, and choose a measure you're interested in seeing the growth of.



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.

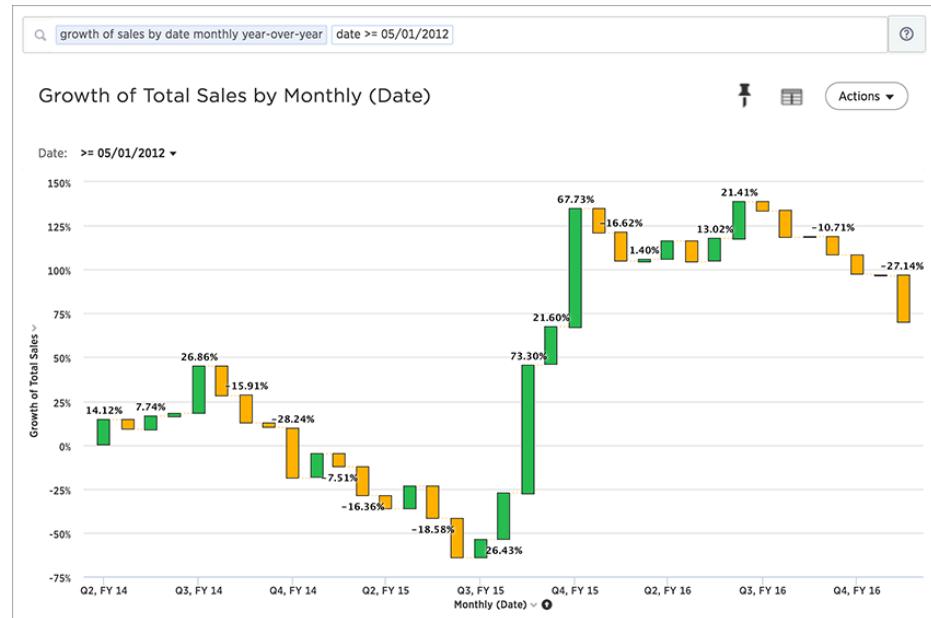
2. Type `by`, followed by a date column name.

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.

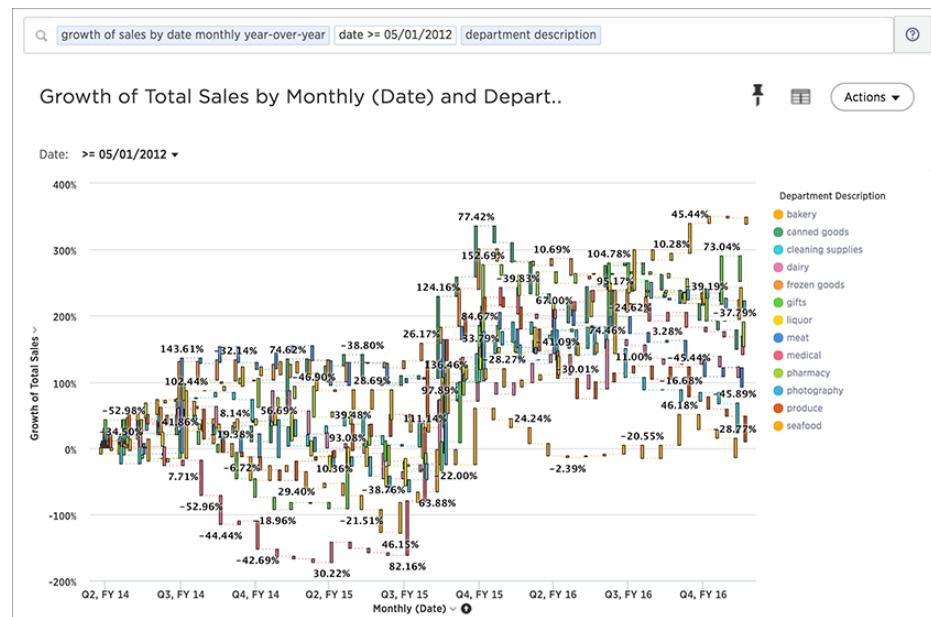
This type of growth analysis is more common in retail and other seasonal businesses. To do this, type monthly year-over-year after your growth of phrase in the search bar.



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)

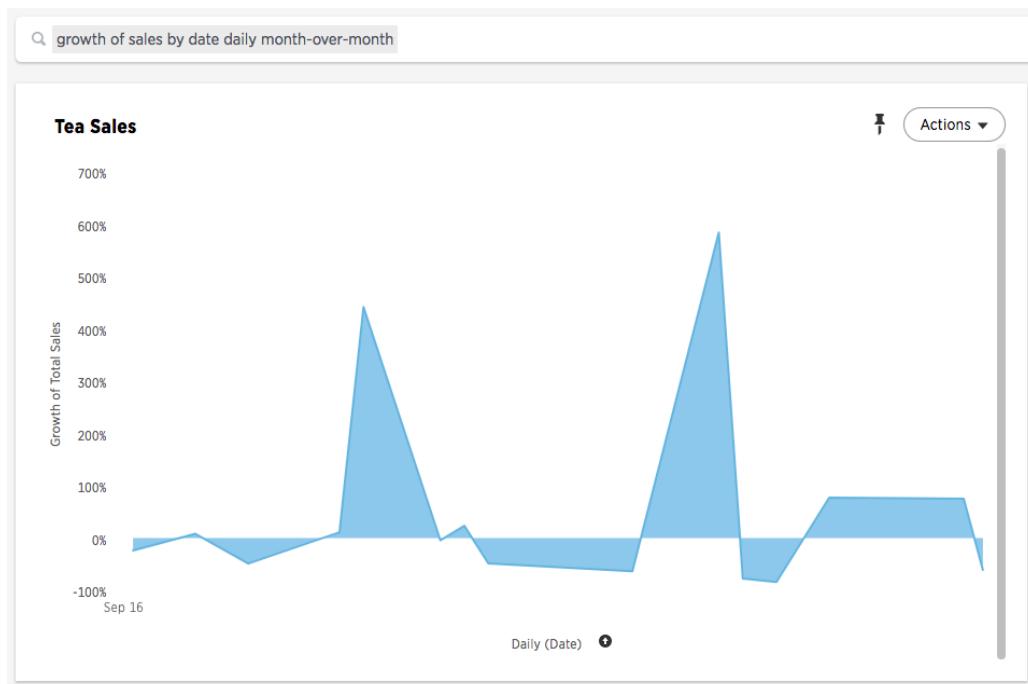
Growth query now supports more hierarchical date buckets and period calculations. The growth query syntax is as follows:

```
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:



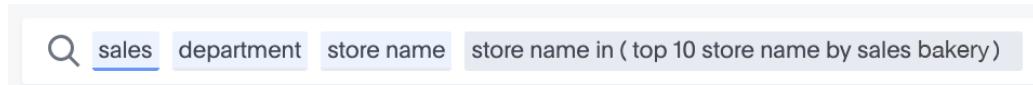
For example, the following query `growth of sales by date daily month-over-month`:



Using the *in* keyword for nested searches

Summary: You can use the *in* keyword to do a search on the result of another search.

Beginning with version 5.2 of ThoughtSpot, you no longer need a View when you want to do a search on top of another search. You can use the “*in*” keyword instead.



Suppose you wanted to narrow down a result set based on criteria defined using a search. For example:

Find the top 5 stores by sales in the bakery department

Then you want to see all sales across all departments for those five stores.

In past versions, you'd have two options:

1. Do your first search to find top 5 stores by bakery sales and save it as a View. Then join the view to your worksheet and search both together to get your answer.

This approach gives a dynamic result that's computed on the fly each time, but it requires the Can Manage Data privilege and requires multiple steps.

2. Save the names of the five stores with the most bakery sales and use them as a filter in a new search.

This approach is easy to accomplish for ad hoc searches, but doesn't compute dynamically.

Now you can do this analysis in a single search using the “*in*” keyword. This provides the benefit of computing the answer dynamically without the extra steps required when using a view.

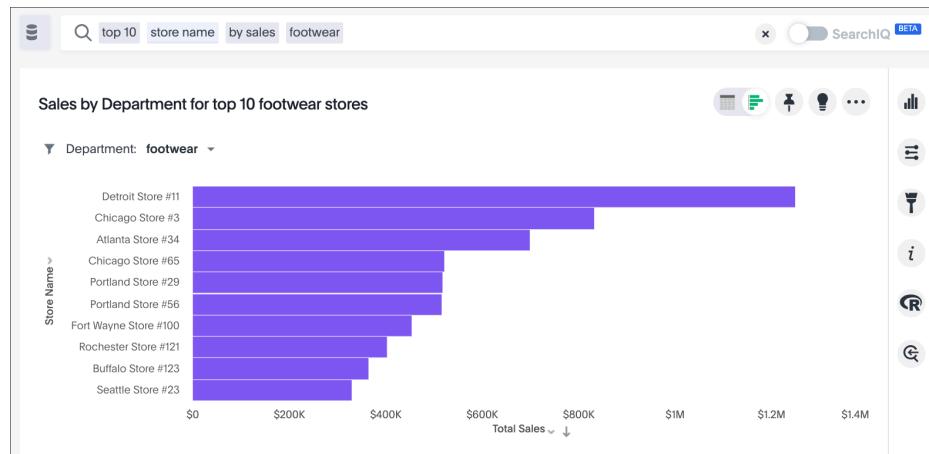
The “in” keyword

Keyword	Example
in (<i>subsearch</i>)	<code>in (top 10 store name by sales footwear)</code>

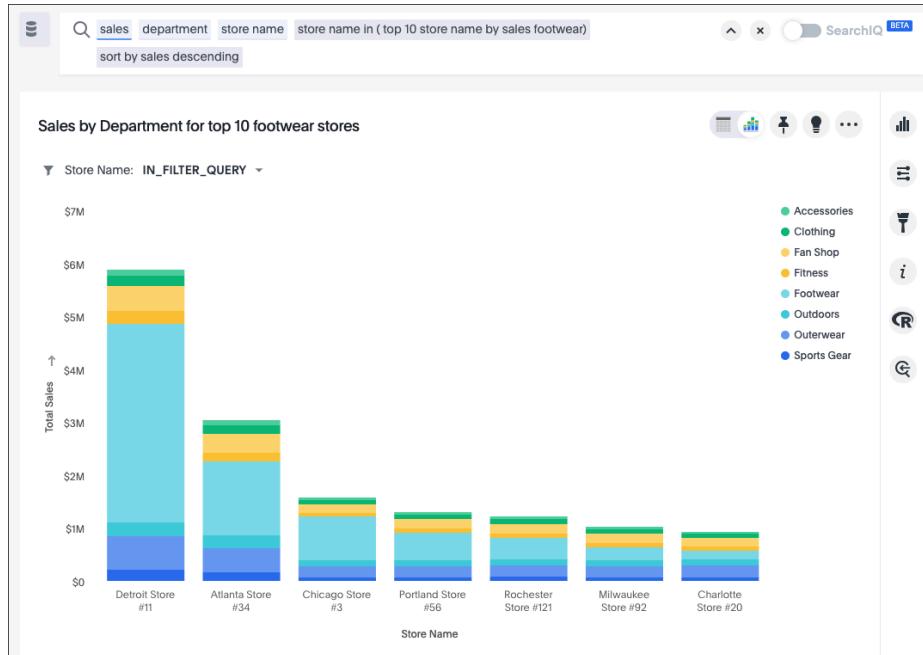
Create an in search

1. Start a new search.
2. Do the search you plan to use for the in clause. That is, the first level search.

Check that the result is what you’re looking for.



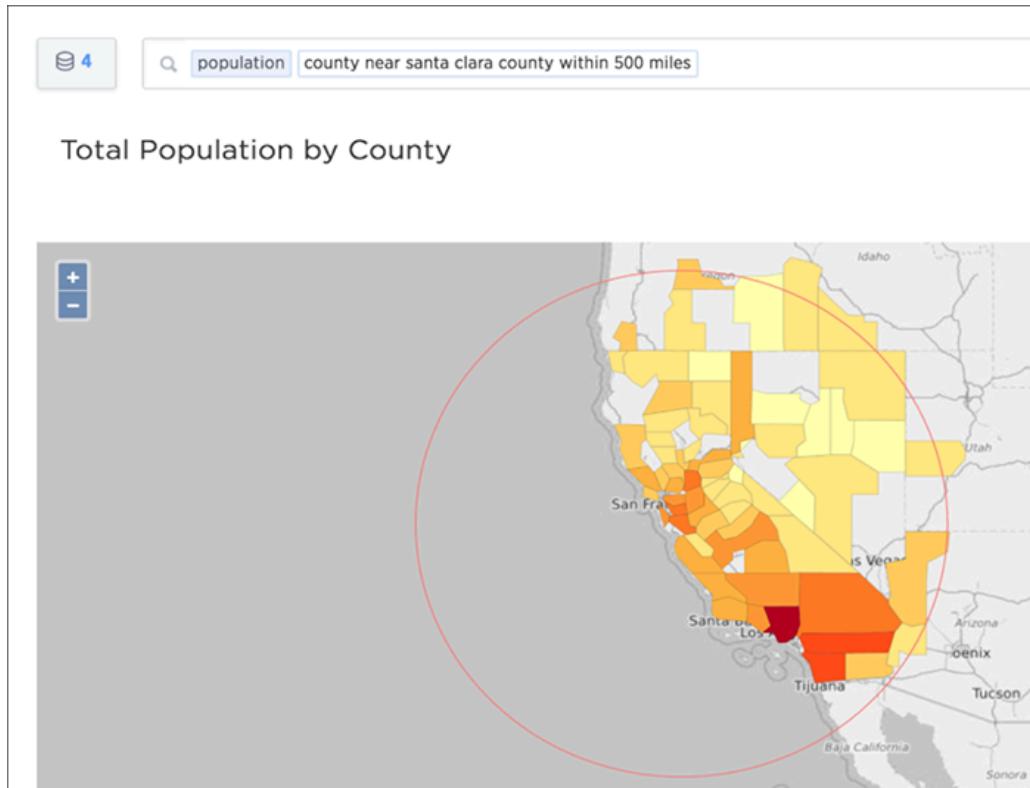
3. Copy that search, and paste it into your new search as the in clause.



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

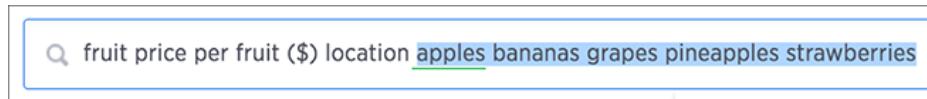
Summary: Filters narrow down the search result to only include the data you want to see.

When you add a value to your search, it becomes a filter. You can define filters on tables, views and worksheets. When you add a filter, it is applied to the table, view, or worksheet, so the result set only shows rows that satisfy a set of parameters specified in the filter. You can also set filters that are automatically used in every search you perform using a particular data source. For example, you can exclude inactive customers records from your search result set. To avoid typing `status = inactive` with every search you perform, you can use a filter. The complex the filter is, the more useful it is to set on the data sources (e.g. `status = inactive year = 2017 rating > 0`).

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.

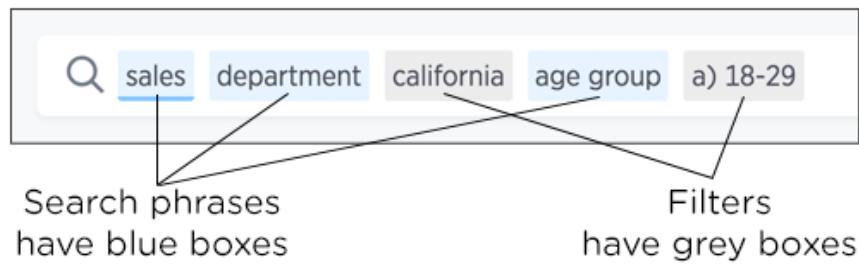
To see more keywords, refer to the [keyword reference](#).

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

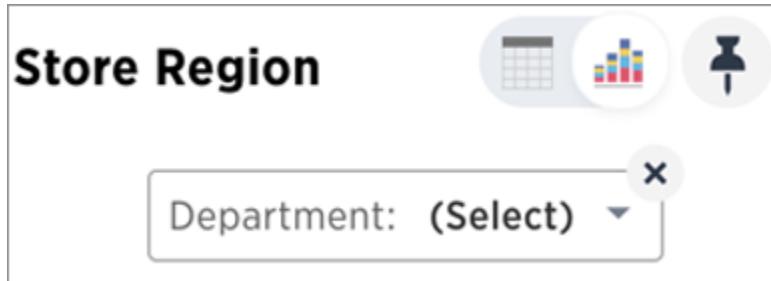
Simple filters can be applied to an answer, while pinboard filters can be applied to all visualizations of a pinboard. You can find out more about [pinboard filters in the pinboards section](#).

Where filters appear in ThoughtSpot

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



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



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 choose **Filter** from the column header or axis label. 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 ellipses icon , in the column header or by clicking on a chart axis, numeric columns and text columns provide you with the ability to include or exclude values, and a checkbox selector for the values. If the column contains a date, you can see a calendar selector when applying a filter. This is also where you can 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 to include or exclude, 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.

Cascading filters

If you want to apply a table filter whenever the table has been used (Views, Worksheets, Answers, and Pinboards), use Cascading filters. When columns from that table are applied in a search, the table filter is implicitly applied to the search. All worksheet filters are accessible from the query visualizer.

Consider a table with a filter that is used in a worksheet. When a search uses that worksheet, the filters are automatically applied as a part of the search.

Worksheet filters

A worksheet filter gets applied every time that worksheet is used. This means that for any search involving a filtered worksheet, all worksheet filters are applied before the search is submitted. So results are always filtered, even if the specific terms searched do not include the column(s) that are filtered.

Add a filter to a table

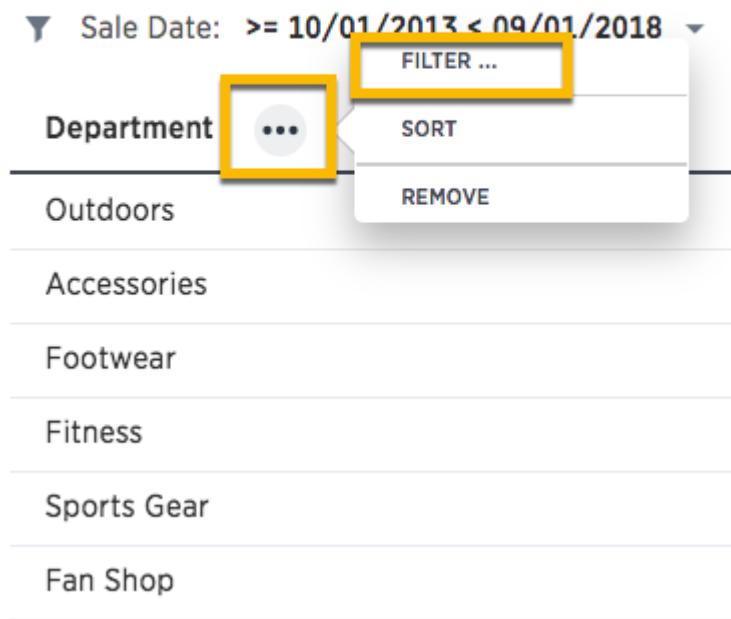
Summary: You can add a simple filter from a column header while viewing your answer as a table.

To add a filter from column headers:

1. While viewing your answer as a table, hover over the column header you want to filter, and

click the ellipses icon  .

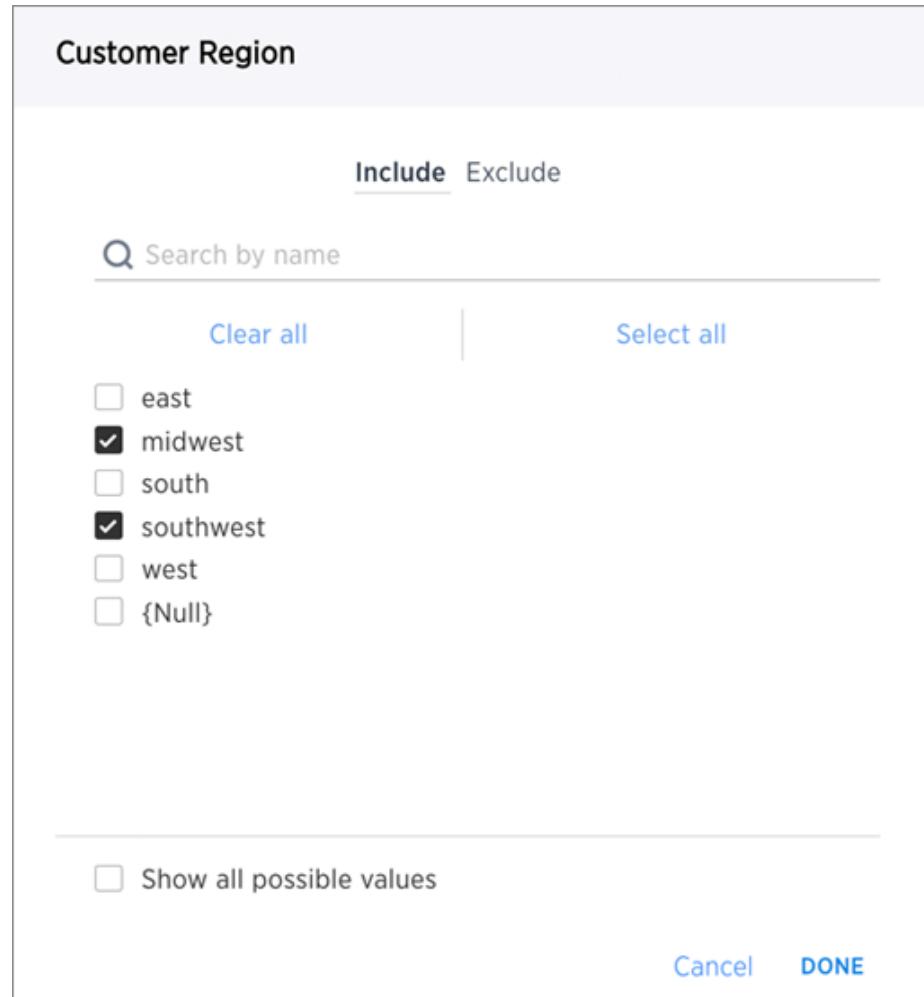
Monthly Growth of Sales



The screenshot shows a table titled "Monthly Growth of Sales". A dropdown menu is open over the "Department" column header. The menu includes options: "FILTER ...", "SORT", and "REMOVE". The "FILTER ..." option is highlighted with a yellow box. The table lists five categories under the "Department" column: "Outdoors", "Accessories", "Footwear", "Fitness", "Sports Gear", and "Fan Shop".

Monthly Growth of Sales	
	Sale Date: >= 10/01/2013 < 09/01/2018
Department	...
Outdoors	
Accessories	
Footwear	
Fitness	
Sports Gear	
Fan Shop	

2. Select **Filter....**
3. Select the values to include in your answer.



4. If you want to exclude values, click **Exclude** and choose values to exclude.

Customer Region

Include Exclude

Search by name

Clear all | Select all

- east
- midwest
- south
- southwest
- west
- {Null}

Show all possible values

Cancel **DONE**

5. 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.

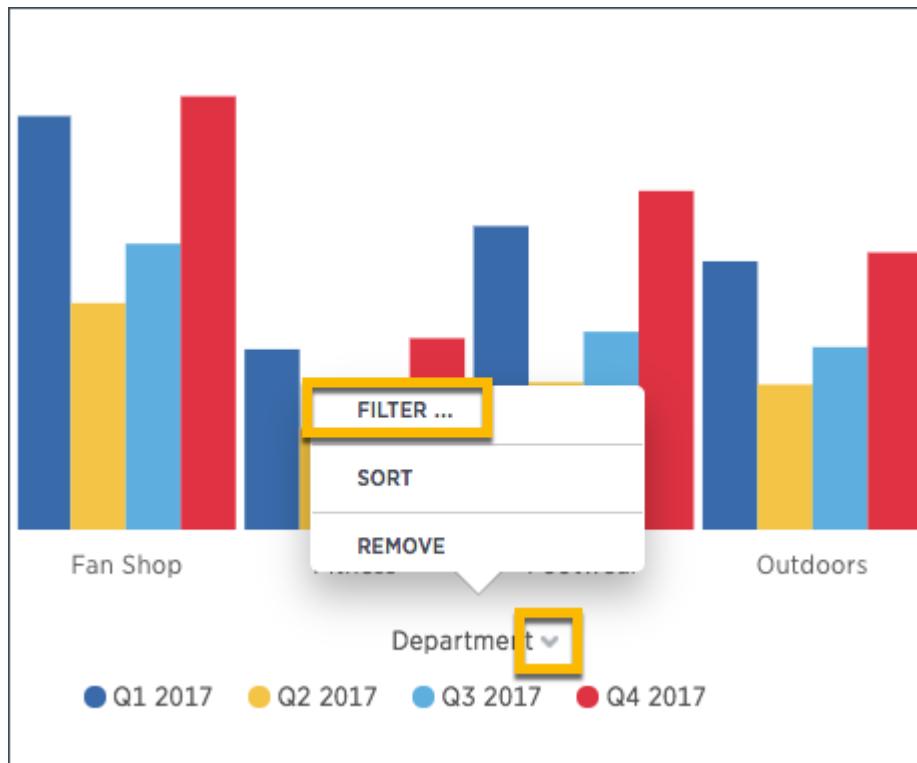
You can filter a chart in two ways:

1. Click 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.

Department

Include **Exclude**

Search by name

[Clear all](#) | [Select all](#)

fan shop
 sports gear

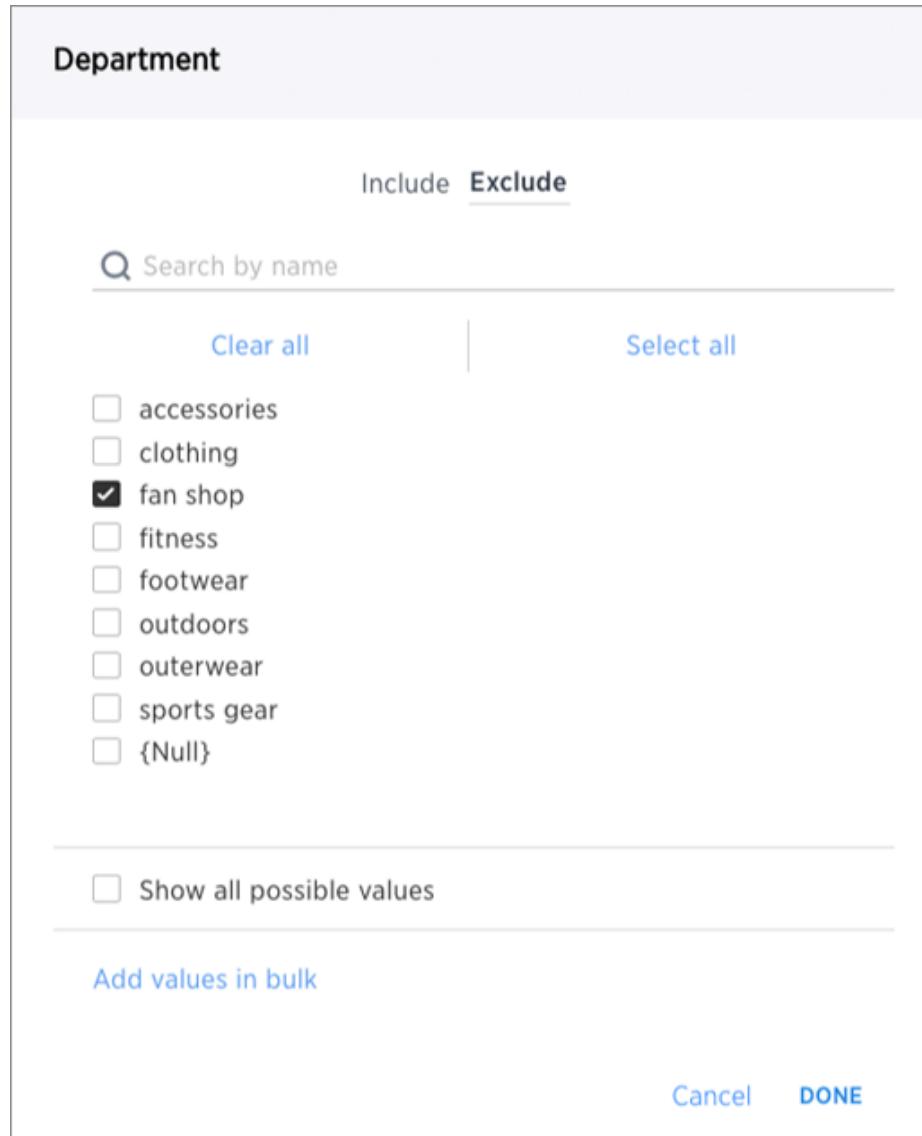
accessories
 clothing
 fitness
 footwear
 outdoors
 outerwear
 {Null}

Show all possible values

[Add values in bulk](#)

[Cancel](#) **DONE**

3. If you want to exclude values, click **Exclude** and choose values to exclude.



4. Click **DONE**.

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

Create a bulk filter

You can create a bulk filter on a saved search. 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. In table view, select multiple cells by clicking and dragging.
2. Right-click and choose **Copy to clipboard**.

The screenshot shows a table titled "Monthly Department Sales Analysis". The first column is "Yearly (Transaction Date)" and the second is "Department". A context menu is open over the "Department" header, listing options: "Copy to clipboard", "Exclude 'Fan Shop'", "Only include 'Fan Shop'", "Drill down", "Show underlying data", "Auto Analyze (SpotIQ)", and "Custom Analyze (SpotIQ)".

Yearly (Transaction Date) ↓	Department	Sales
2017	Fan Shop	1,665.99
2017	Sports Gear	1,659.92
2016	Sports Gear	1,817.03
2016	Fan Shop	760,579.94
2015	Sports Gear	213,930.58
2015	Fan Shop	440,016.37

3. Hover over the column header you want to filter, and click the **three-dot icon (ellipses)** icon.
4. Choose whether you want your bulk filter to **Include** or **Exclude** values.
5. Click **Add values in bulk**:

Department

Include Exclude

Q Search by name

Clear all | Select all

fan shop
 sports gear

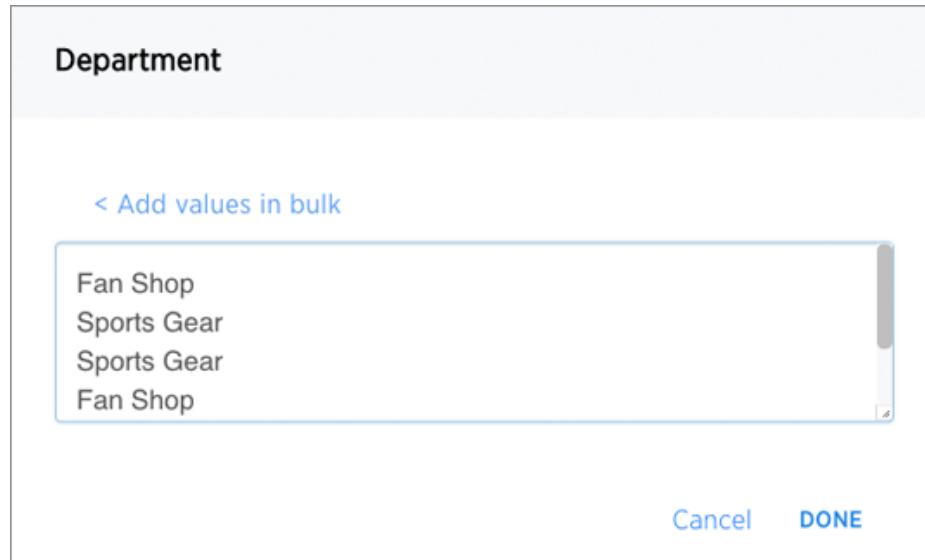
accessories
 clothing
 fitness
 footwear
 outdoors
 outerwear
 {Null}

Show all possible values

Add values in bulk

Cancel DONE

6. Paste the values into the bulk filter box.



7. 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.

Total Price per fruit (\$) by Fruit, Location

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

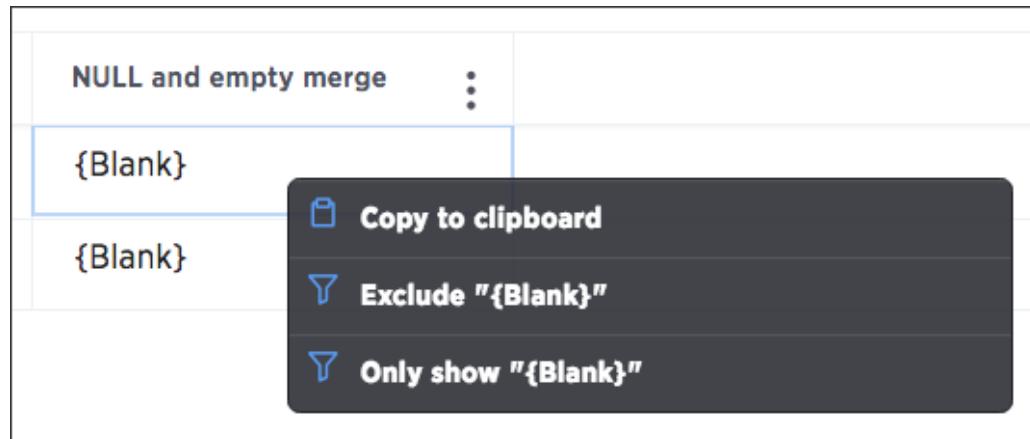
If you want to keep the same display format for NULL and blank values, but be able to filter on both using “{blank}”, your [formula](#) will be slightly different. You can use a formula like:

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

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

Filtering on your formula

After creating formula for 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 over, under, 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. 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 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**).

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**:

Yearly (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.

- In a table view, use the date bucket drop-down, under the column header.
- On a chart, you can change the change the axis label.

Table

Yearly (Order Date)

Order Date ^

NO_BUCKET

- NO_BUCKET
- HOURLY
- DAILY
- WEEKLY
- MONTHLY
- QUARTERLY
- YEARLY
- DAY_OF_WEEK
- DAY_OF_MONTH
- DAY_OF_QUARTER

Chart

Q3 94

Q1

DETAILED

HOURLY

DAY

WEEK

MONTH

QUARTER

YEARLY

FILTER ...

SORT

Quarterly (Order Date) ^

The bucket values are a subset of date and time-period keywords. To change the date bucketing, follow these steps:

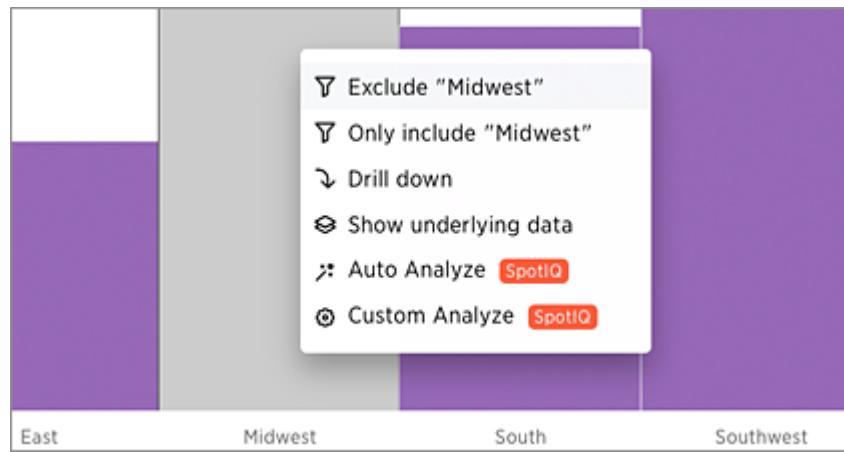
1. Go to the table or chart you want to change.
2. Locate the column or axis value with the date value you want to change.
3. Click the column's bucket menu.
4. Select a date bucket.

Exclude and include row values

You can include or exclude row values from your answer.

To exclude or include row values:

1. Right-click 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. Then, you can right-click any row and then click **Show underlying data**, to see each value which constitutes `revenue` for any given region.

Only the first 1,000 rows appear 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.

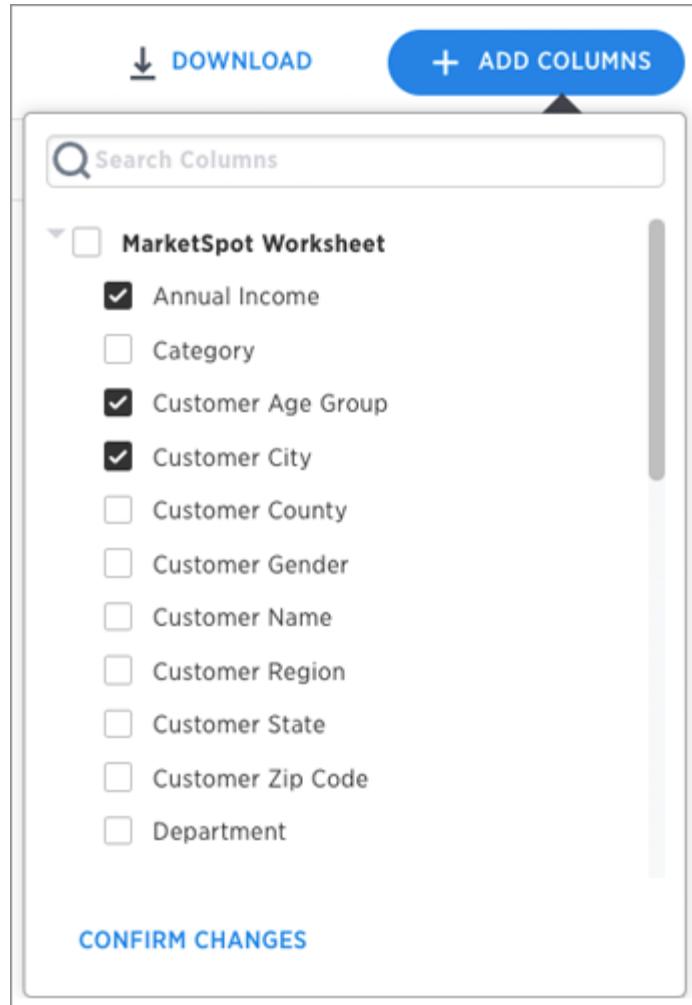
To show underlying data:

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

Store State	Total Sales	Yearly (Sale Date)	Store Region
Ohio		2016	Midwest
Indiana		2015	Midwest
Missouri		2015	Midwest
Texas		2017	Southwest
Massachusetts		2017	East

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

3. Optionally, choose to + ADD COLUMNS to add additional data.



4. Click **CONFIRM CHANGES**.
5. Click **DOWNLOAD** to download a CSV file of the data.

Showing underlying data

[CLOSE](#)

Summary

Yearly (Sale Date): 2016 Store State: Ohio Store Region: Midwest

Total Sales: \$1,145,005.69

Underlying Data

[Download](#) [+ Add Columns](#)

	Sales	Annual Income	Customer Age Group	Customer City
L6	\$5,713.81	162,483	31 to 50 Years	Ashland
L6	\$260.06	145,002	31 to 50 Years	Barnett
L6	\$44.35	868,068	19 to 30 Years	Rayle
L6	\$7.06	408,234	19 to 30 Years	Hialeah

(showing rows 1-4 of 1000)

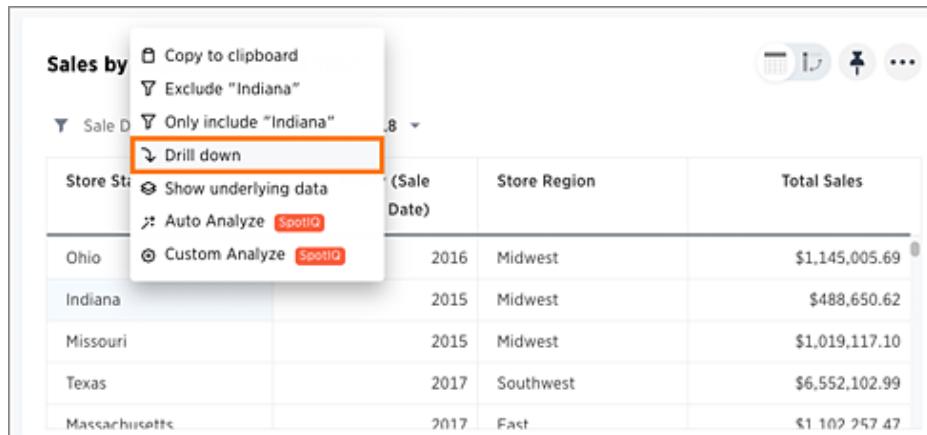
See the search behind a result

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

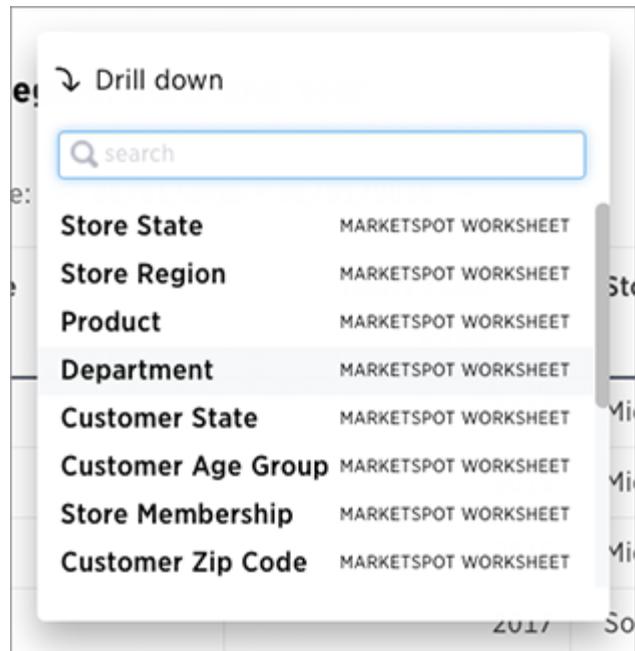
Drilling down allows you to see more information about the columns used within your search. If you own the pinboard you can drill down to the data beneath. Users you share the pinboard with can also drill down provided they *also* have access to the data on which the board was based.

To drill down:

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



2. Click 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

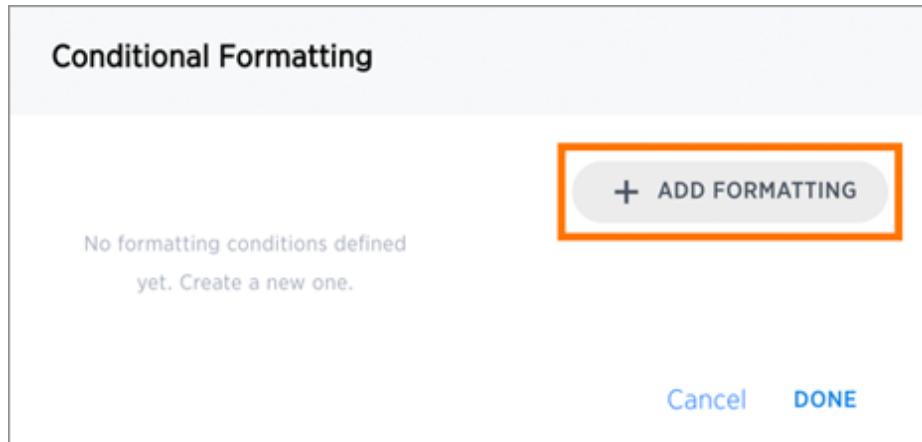
You can use conditional formatting to show table cells with a background color determined by the value they contain.

To apply conditional formatting to a table:

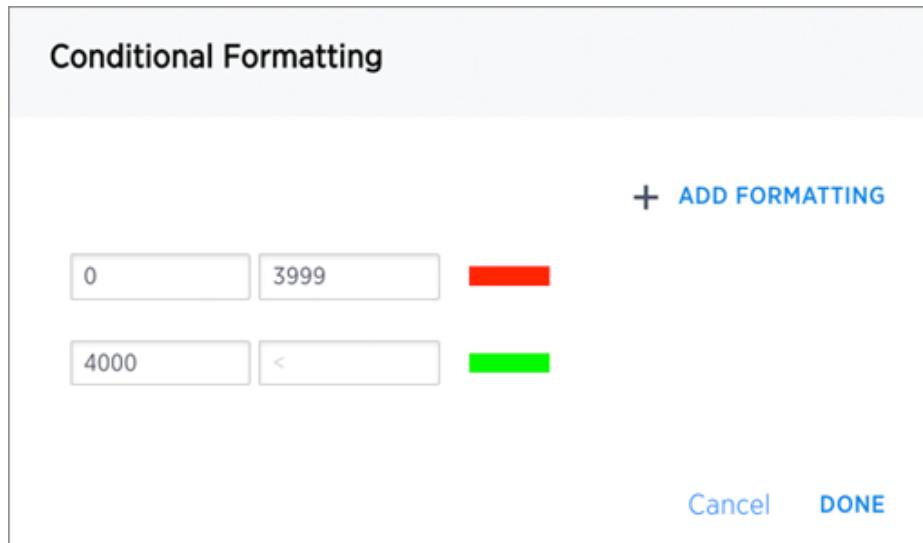
1. In the column header of your table for the column you want to apply formatting to, click the three-dot **Change Configuration** icon.

Total Sales
993.87
3,129.50
4,235.04
2,023.95
2,498.40

2. Select **Conditional Formatting**.
3. Click **+ ADD FORMATTING** in the **Conditional Formatting** window.



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.

Daily (Date)	Total Sales
12/29/2018	3,129.50
12/05/2018	4,235.04
12/01/2018	2,023.95
12/25/2018	2,498.40
12/13/2018	14,228.59

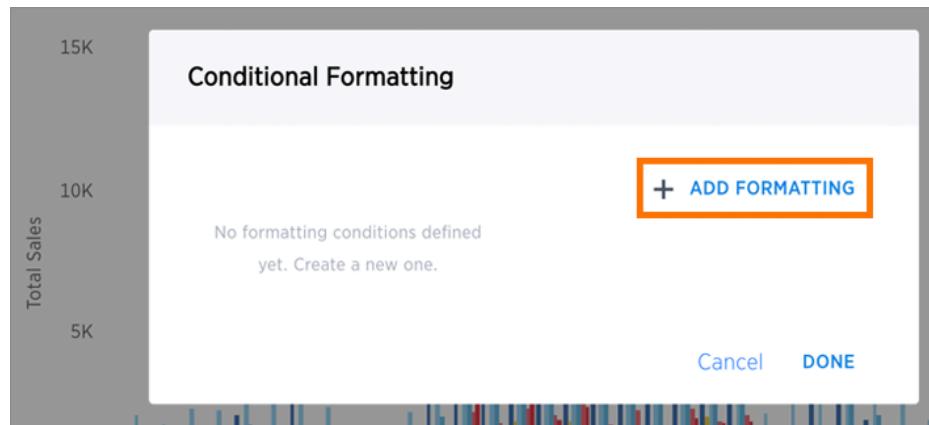
Note: If you change to a chart type, you must apply conditional formatting again. Conditional formatting is tied to the specific visualization.

Apply conditional formatting to a chart

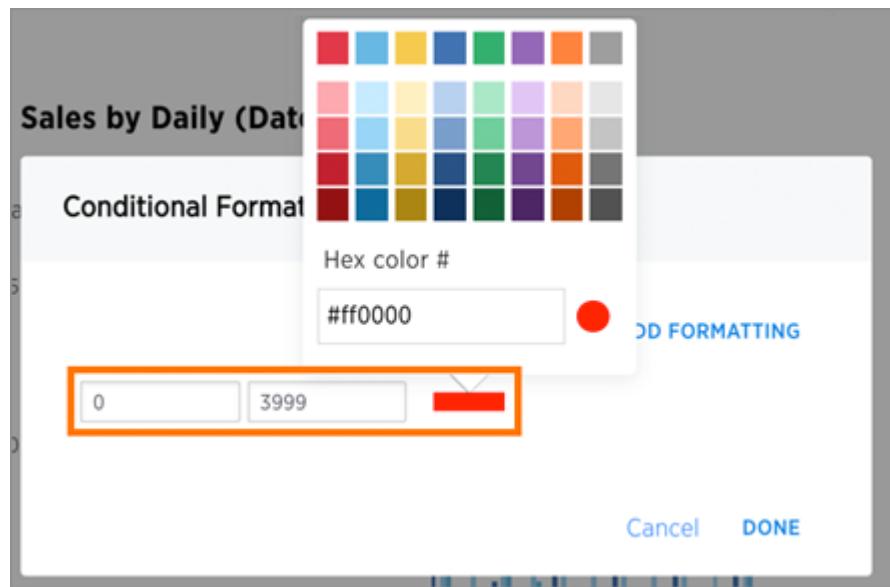
You can use conditional formatting to show charts with a target value or range drawn as a line in the chart, and the legend colors determined by where values fall relative to the target.

To apply conditional formatting to a chart:

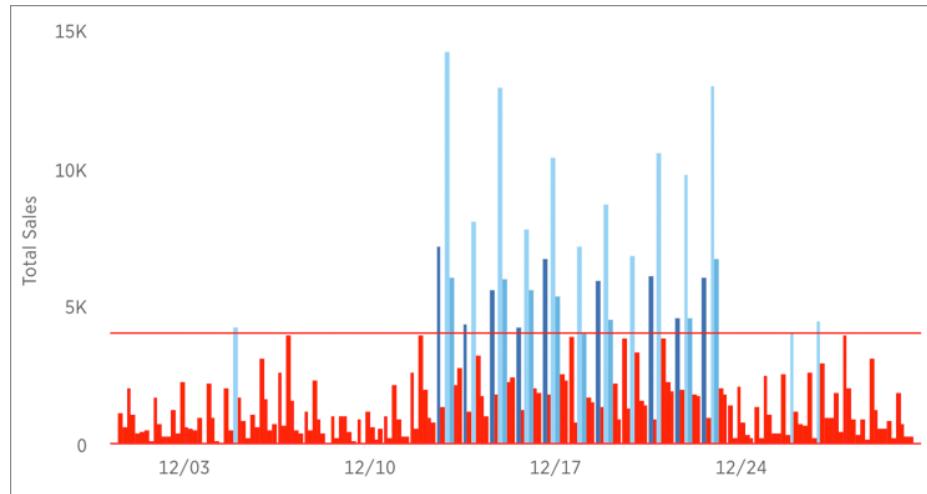
1. Click the axis label of your chart. Select **Conditional Formatting**.
2. Click **+ ADD FORMATTING** in the Conditional Formatting window.



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.



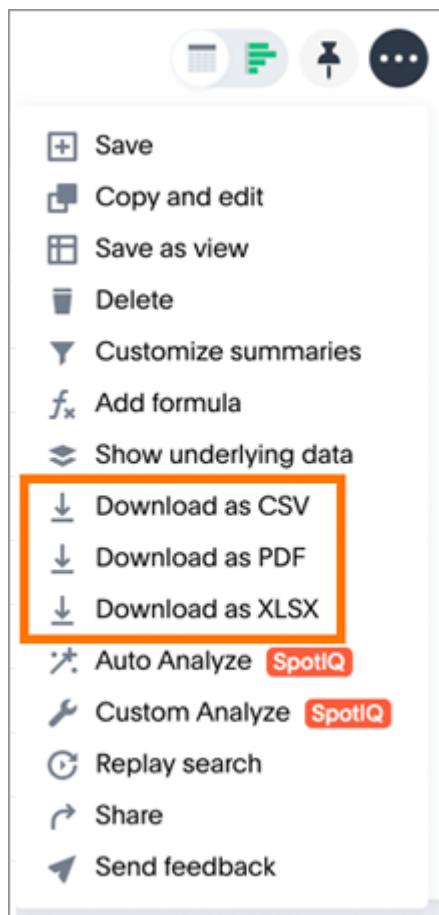
Note: If you change the chart type, you must apply conditional formatting again. It is tied to the specific visualization.

Download your search

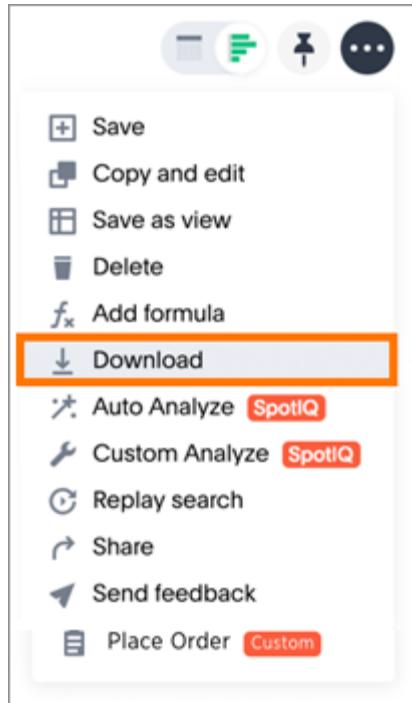
Summary: You can download your search as either a table or chart.

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 the three-dot **Actions** icon, and do any of the following:
 - For a table, choose **Download as CSV**, **Download as PDF**, or **Download as XLSX**.



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



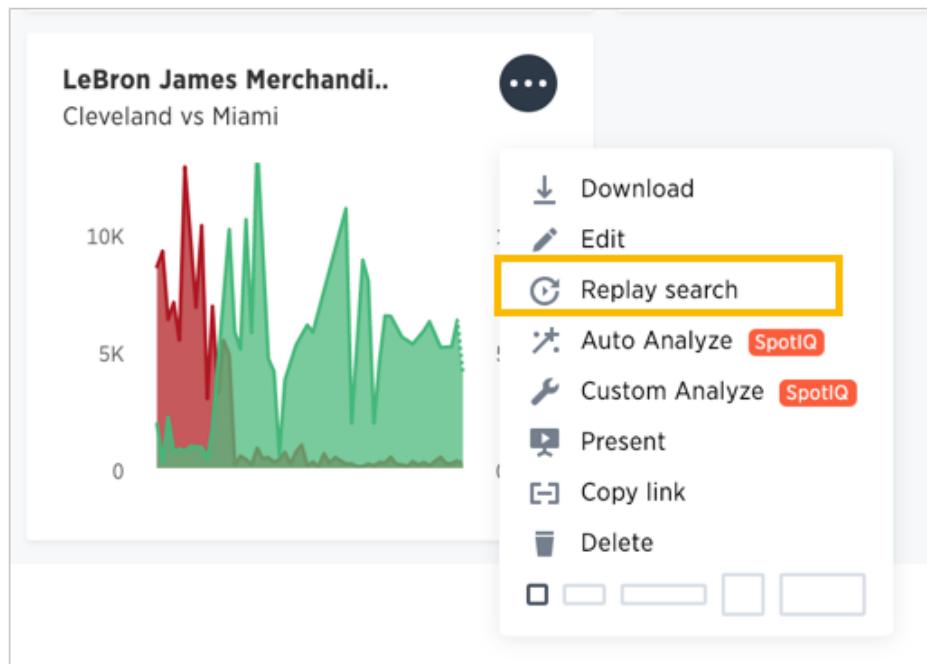
Currently, ThoughtSpot limits the maximum number of rows that can be downloaded to 10M (default is 1M). Your system limit is configured by your ThoughtSpot administrator.

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 screen recording of it and create your own ThoughtSpot training for your team.

- When viewing a chart or table, click the ellipses icon  and select **Replay search**.



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.

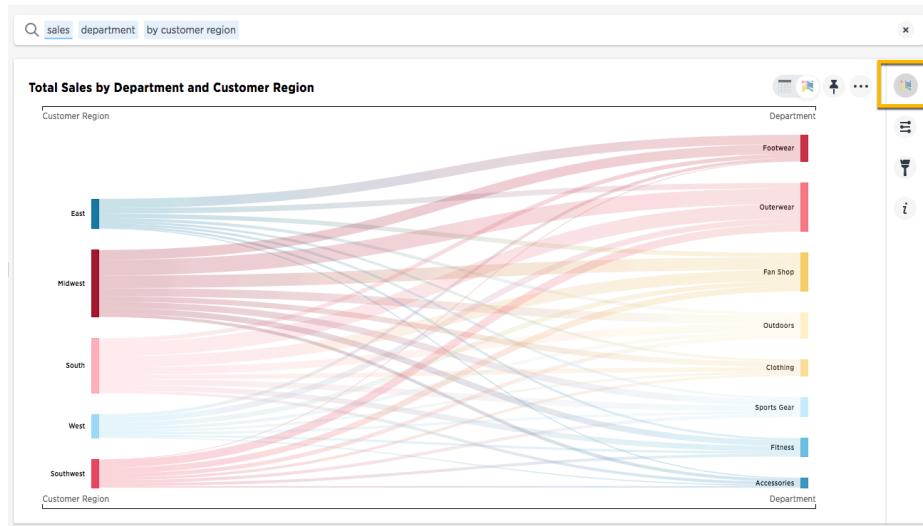
You can also [adjust the axes, labels, and view of the chart](#).

About chart types

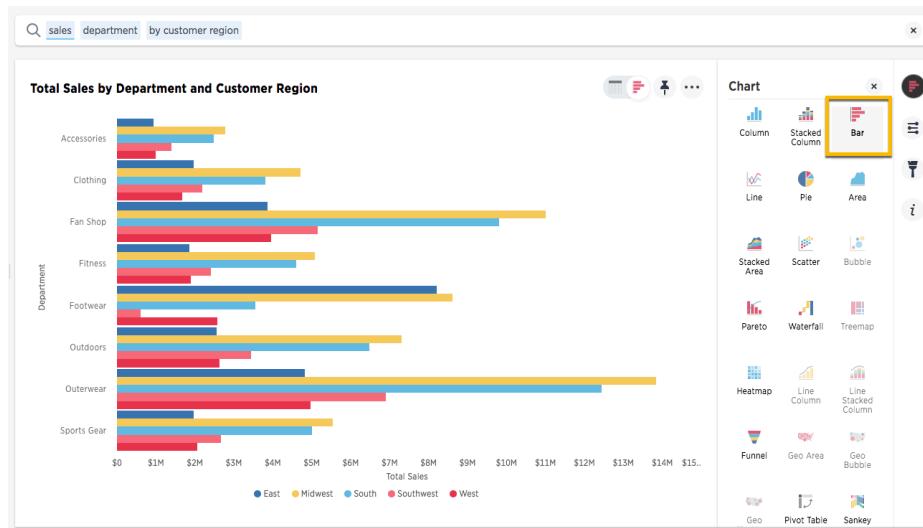
You can choose from a large number of chart types in ThoughtSpot. Each chart type provides you with a different visualization for your answer.

To change the chart type of your answer:

1. Click **Change visualization** to get a palette of charts, maps, pivot table, and so on (scroll down to see more.)



- Click a different chart or visualization type.



Choosing a chart

Note: Some chart types may not be available, depending on the columns in 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. ThoughtSpot shows unavailable chart types as grayed out.

Hovering over a chart type icon tell you what columns you require before you can use it.

- **Column charts**

The column chart is one of ThoughtSpot's simplest, yet most versatile chart type. The column

chart is often the chosen default chart type, and displays data as vertical columns.

- **Stacked columns** The stacked column combines the different secondary dimensions into a single column, stacking them. Note that the dimensions retain their relative size and color.

- **Bar charts**

The bar chart is nearly identical to the column chart. The primary difference is that it displays data as horizontal bars.

- **Stacked bar charts** Just like stacked columns, stacked bars combine the different secondary dimensions into a single stacked bar.

- **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.

- **Stacked area charts** This option stacks the values of one dimension on top of the other, enabling you to clearly see the relative volume of data under the line. Contrast this with the presentation in the area chart, where data for different overlaps.

- **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.

- **Geo charts**

There are three geo charts that let you visualize geographical data in ThoughtSpot: Area, Bubble, and Heatmap.

- **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.

- **Sankey charts**

Sankey charts show a flow from one set of values to another, usually with visual emphasis (brighter colors or larger bandwidths) for comparison of the values (e.g., top cities in terms of sales might be emphasized).

- **Radar charts** Radar charts display multivariate data on a two-dimensional chart of three or more quantitative variables that plot on axes (spokes) that originate at the same point.

- **Candlestick charts** Candlestick charts efficiently collapse financial information, such as price movements on a single day, into a unified representation. A single ‘candlestick’ shows the *open*, *high*, *low*, and *close* prices for the same day.

Charts with multiple measures on the y-axis

You can have multiple measures on the y-axis of many charts, which is a great additional way of presenting information in a chart.



Example of multiple measures on the y-axis of a stacked column chart.

The following charts support multiple measures on the y-axis:

- Column
- Stacked Column
- Bar
- Stacked Bar
- Line
- Area
- Stacked Area
- Waterfall
- Line Column
- Line Stacked Column

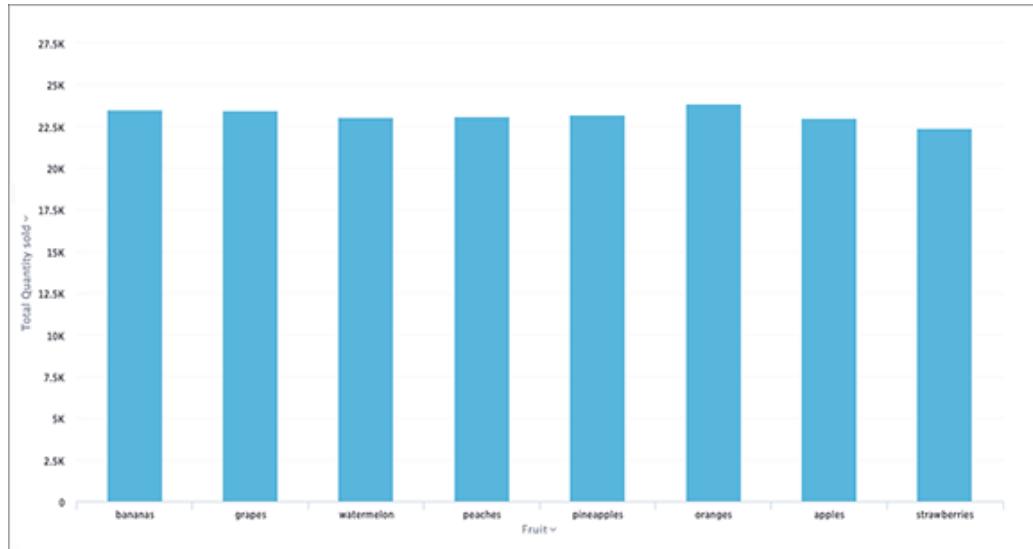
To learn more, see [Stack multiple measures on the y-axis](#).

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.



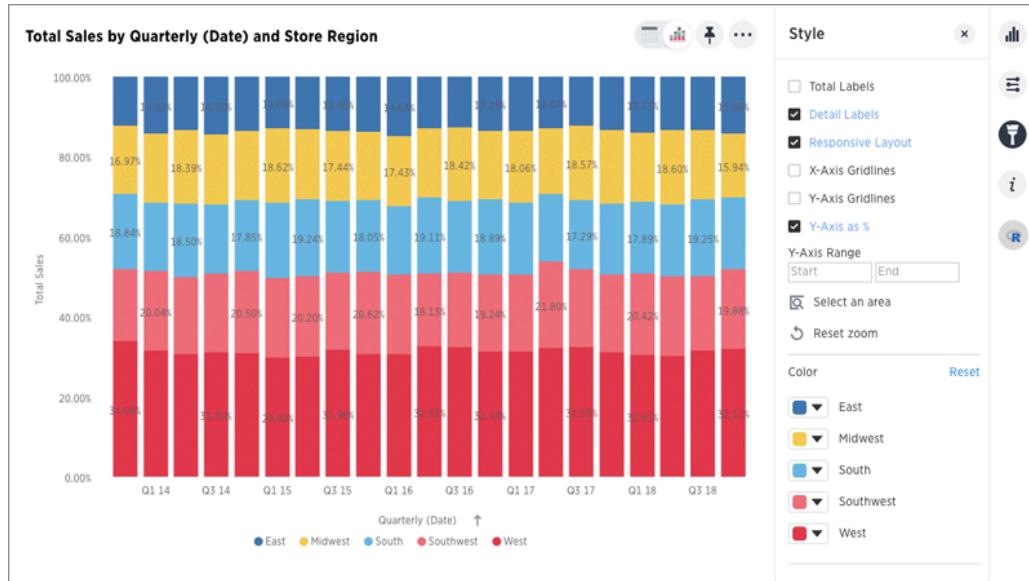
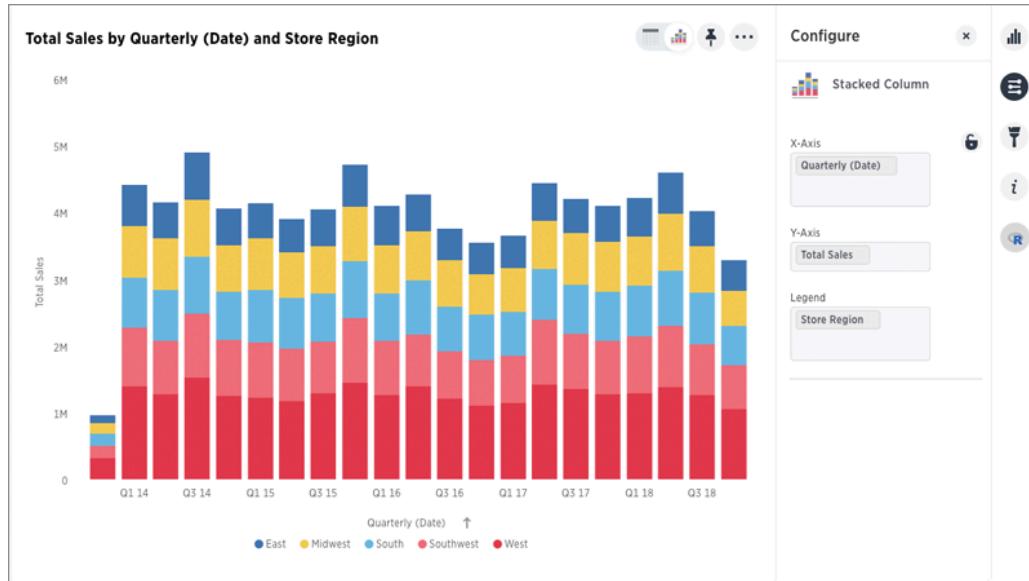
Your search needs at least one attribute and one measure to be represented as a column chart.

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. You can toggle the options in styles to show Detailed Labels (summaries for each section of each bar) and Total Labels (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.

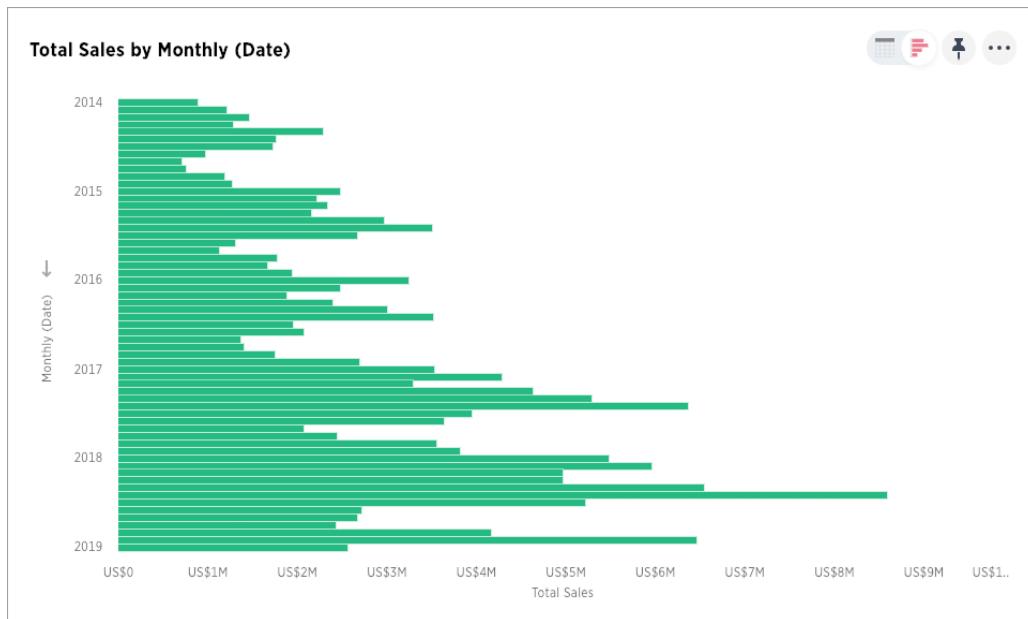


Your search needs at least two attributes and one measure to be represented as a stacked column chart.

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.

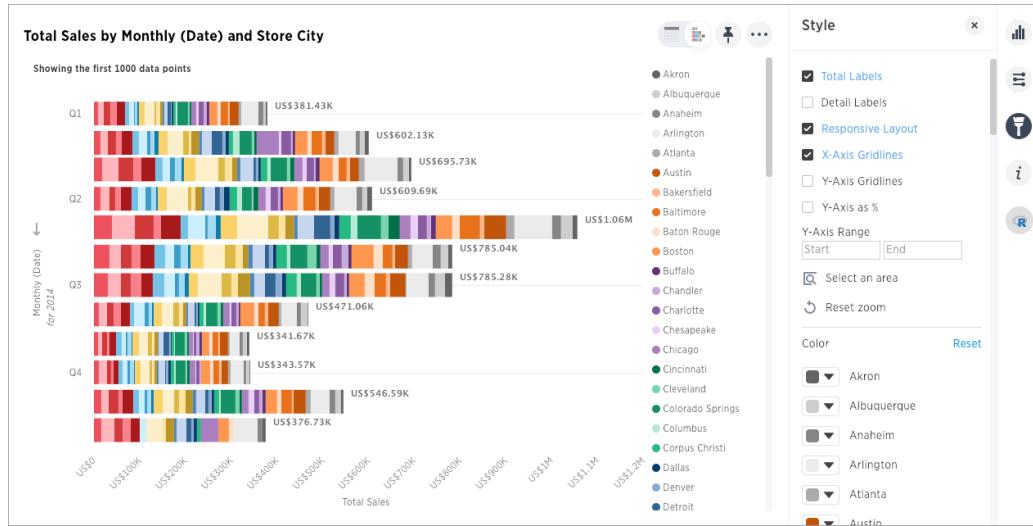


Your search needs at least one attribute and one measure to be represented as a bar chart.

Stacked bar charts

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

Stacked bar charts are typically used when you want to compare aggregated data and the data that it includes together. You can toggle the options in styles to show Detailed Labels (summaries for each section of each bar) and Total Labels (show the sum of the stacks for each bar).



It is important to note that stacked bar 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 bar chart.

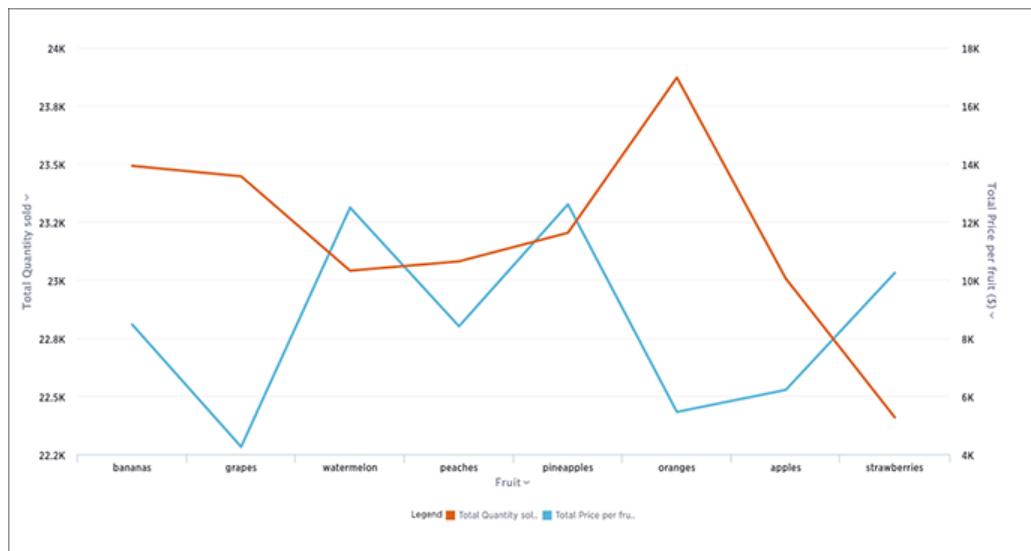
Line charts

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.

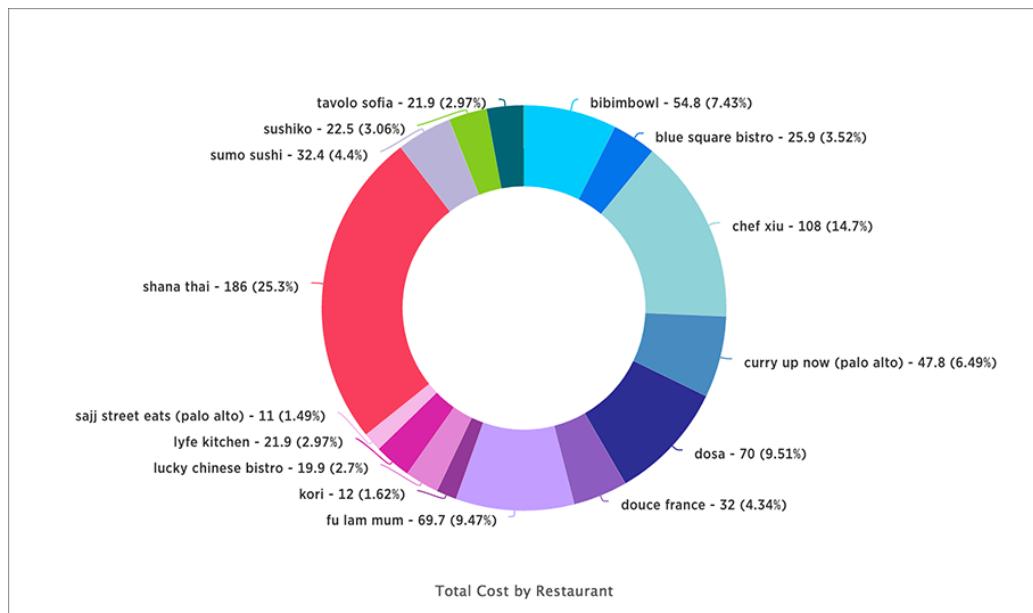
Pie charts

Summary: A pie (or a circle) chart is a statistical graphic that divides data into slices to illustrate numerical proportion. In a pie chart, the arc length of a slice is proportional to the quantity it represents.

The pie chart is a classic chart type that displays your search in a circle. The default pie chart in ThoughtSpot displays data in the shape of a doughnut, or a thick ring.

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.

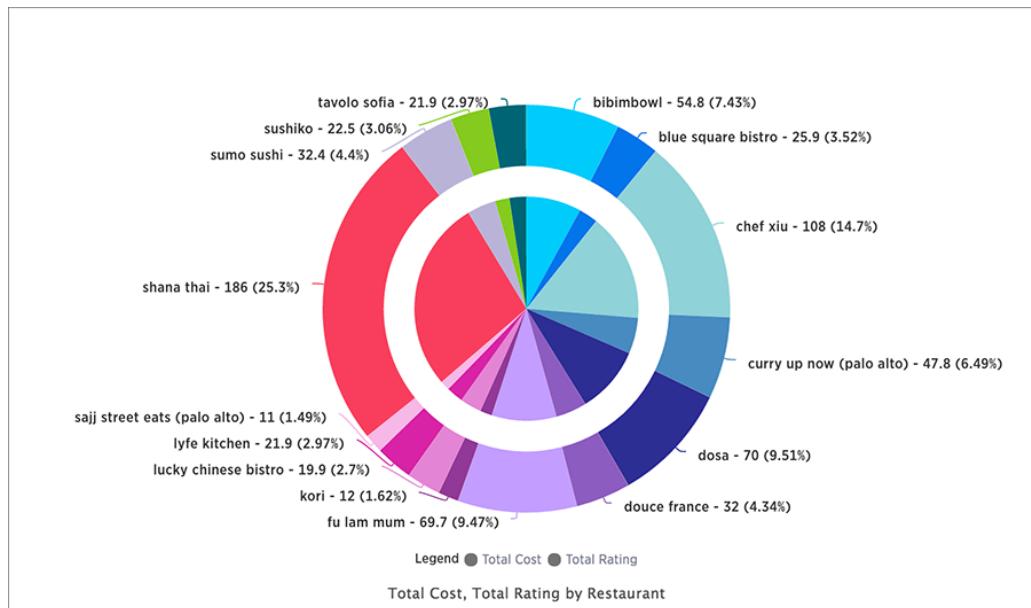


Your search needs at least one attribute and one measure to be represented as a pie chart. Also, there must be fewer than 50 values in the attribute column.

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**.



Color customization of pie in pie charts

You can customize the colors of each pie in a pie chart using the Style widget.

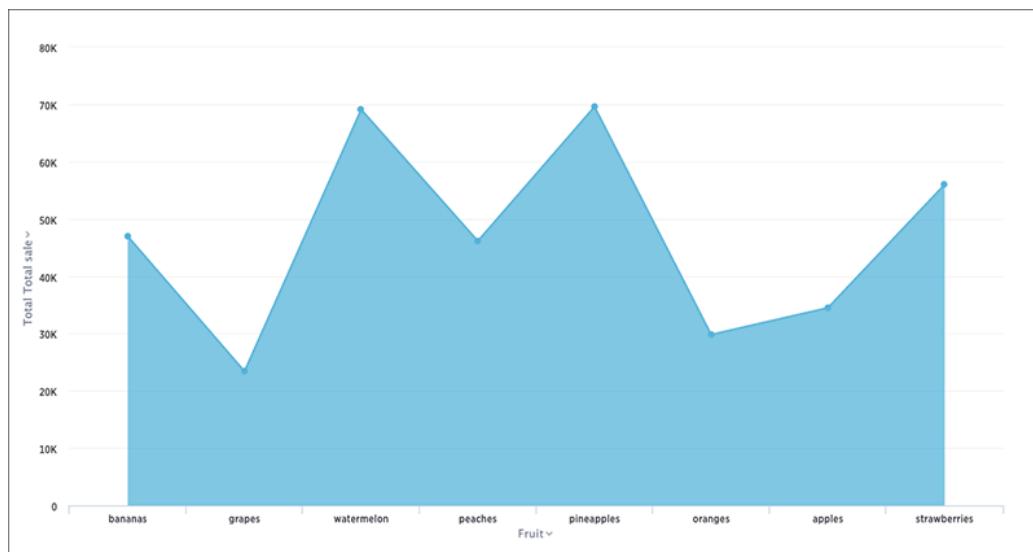
1. Choose a pinboard that has a pie chart.
2. Click the **Edit chart style** (paintbrush icon) to open the Style widget.
3. From the Style widget, assign a color of your choice to each pie as listed in the widget.
4. Click ... and choose **Update** to save the changes.
5. Click **Close**.
6. Optionally, you can click **Reset** to use the default color.

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.



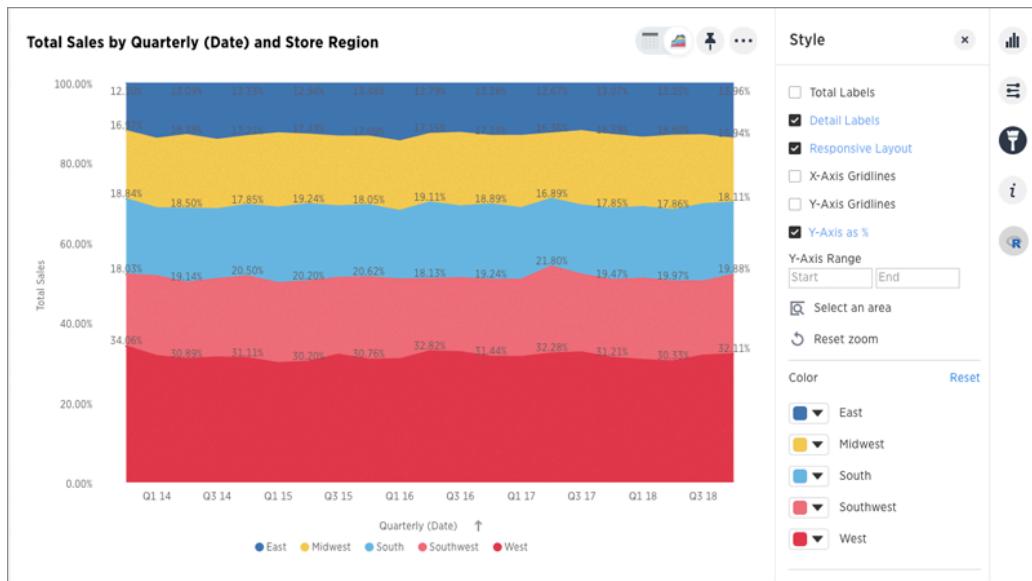
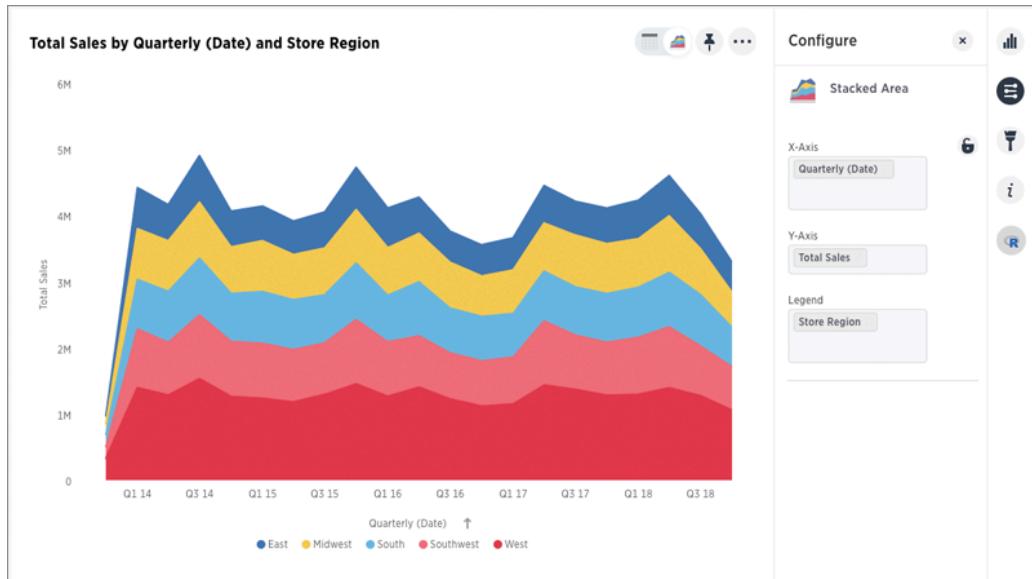
Your search needs at least one attribute and one measure to be represented as an area chart.

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.

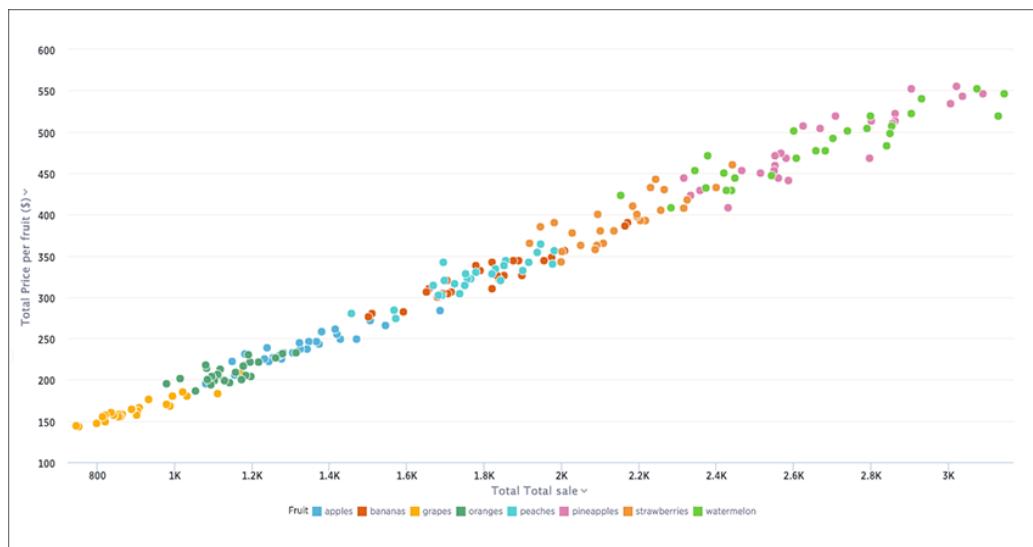


Your search needs at least two attributes and one measure to be represented as a stacked area chart.

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.

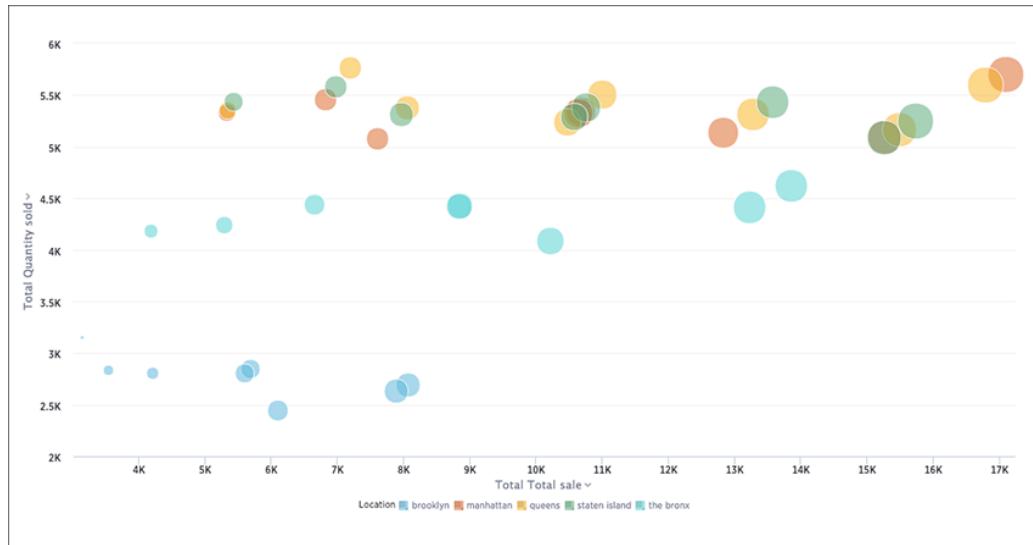


Your search needs at least one attribute and one measure to be represented as a scatter chart.

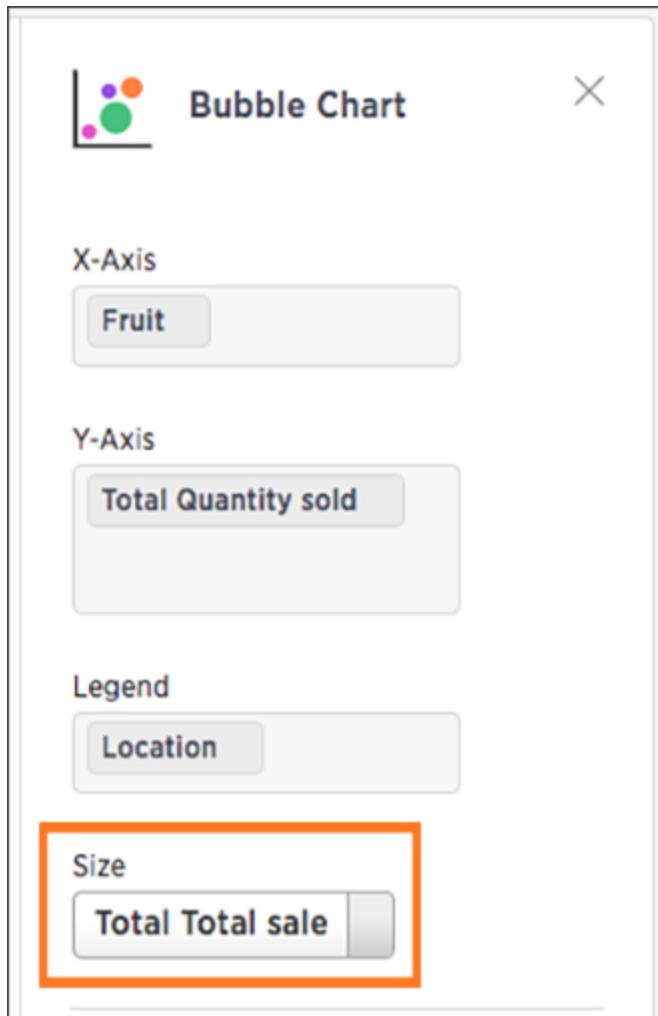
Bubble charts

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**.

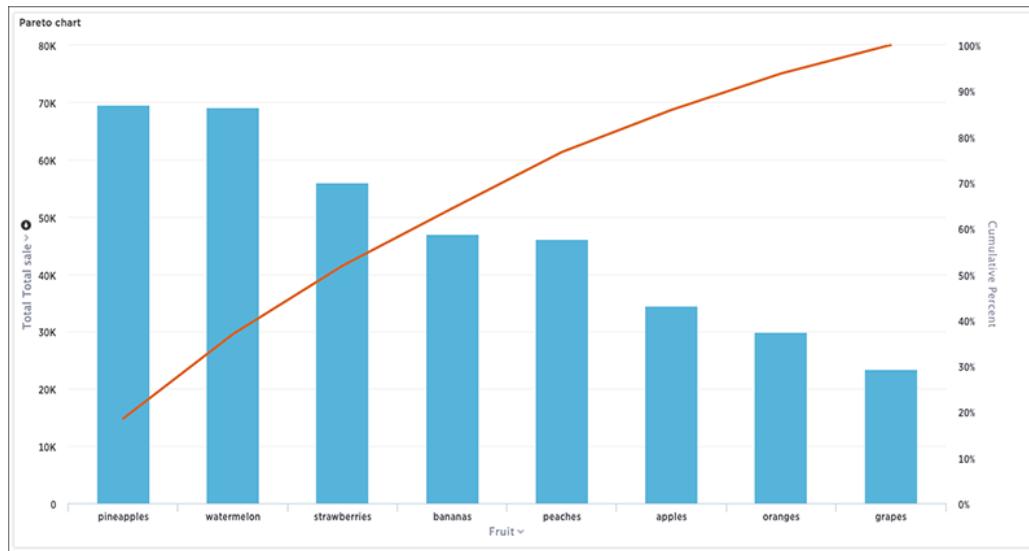


Your search needs at least one attribute and two measures to be represented as a bubble chart.

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.

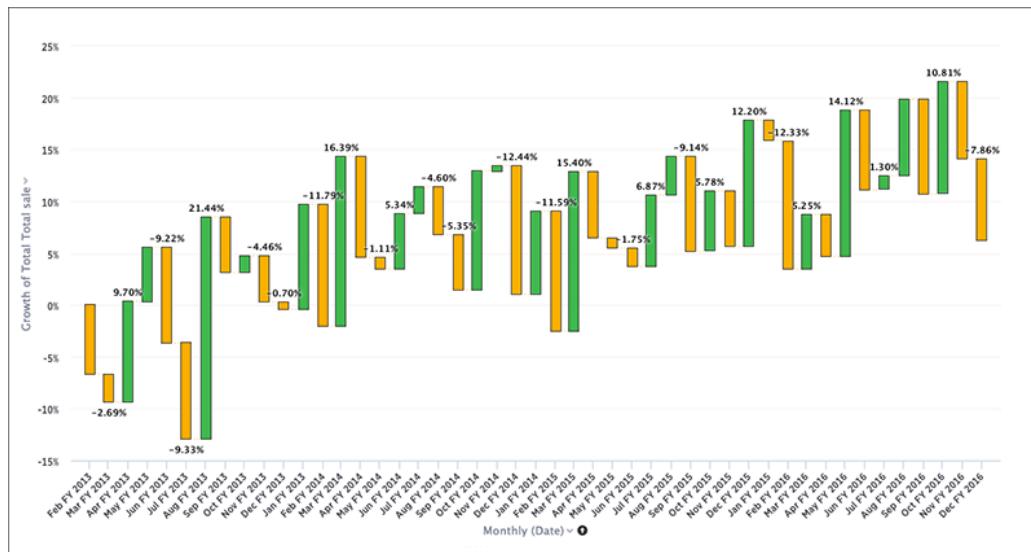


Your search needs at least one attribute and one measure to be represented as a pareto chart.

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.

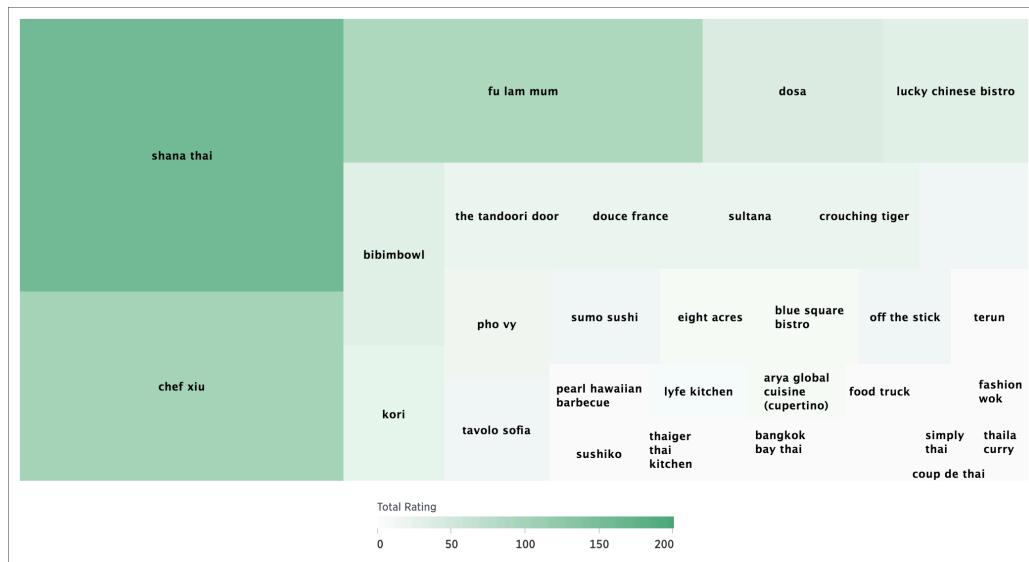


Your search needs at least one attribute and one measure to be represented as a waterfall chart.

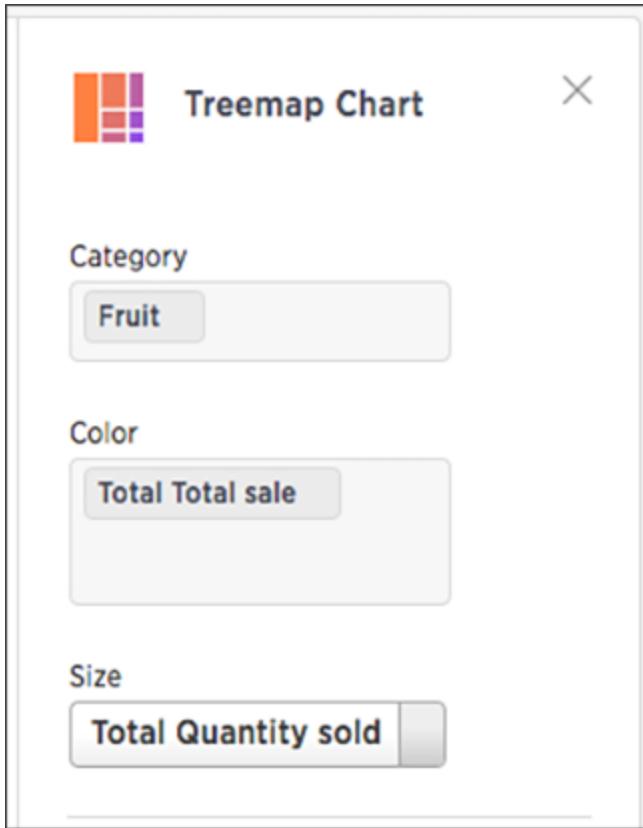
Treemap charts

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.

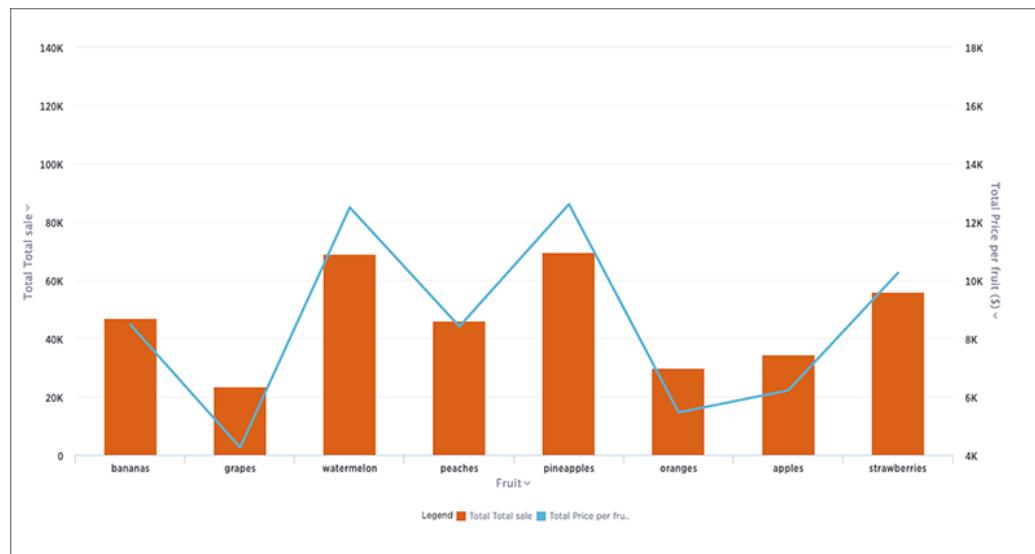
Line column charts

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

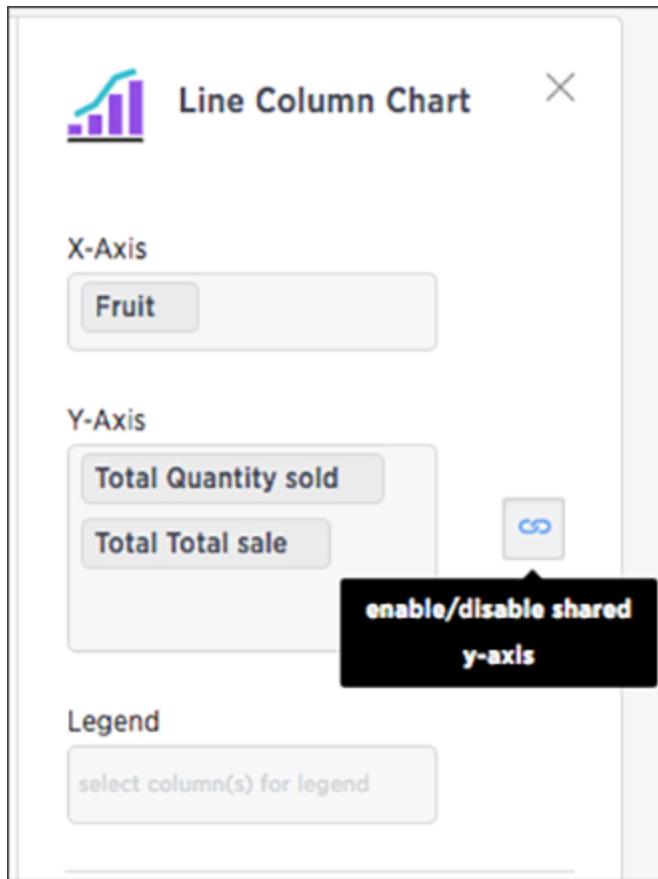
The line column chart combines the column and line charts. Your search needs at least one attribute and two measures to be represented as a line column chart.

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.

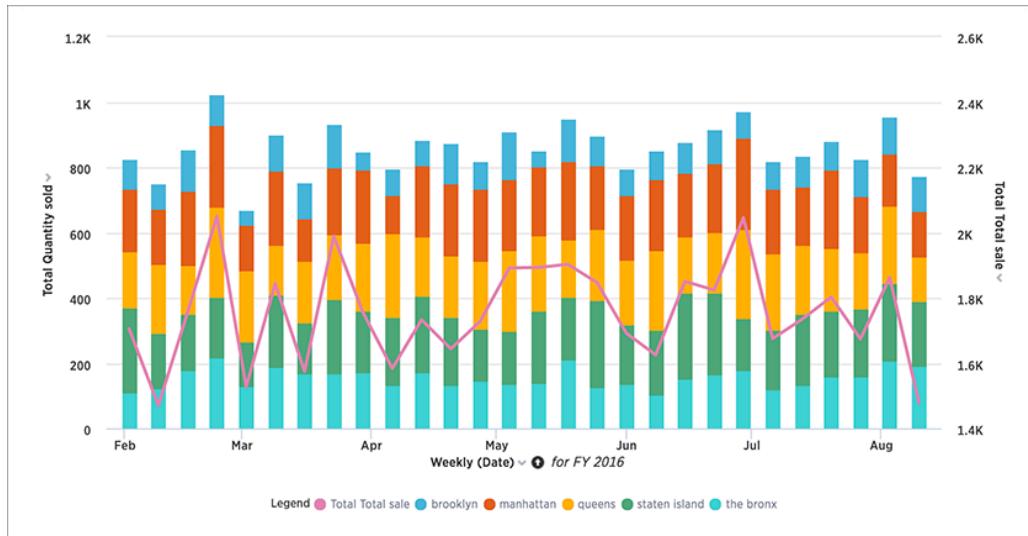


You can choose to enable shared y-axis by clicking the link icon found under **Edit chart configuration**.

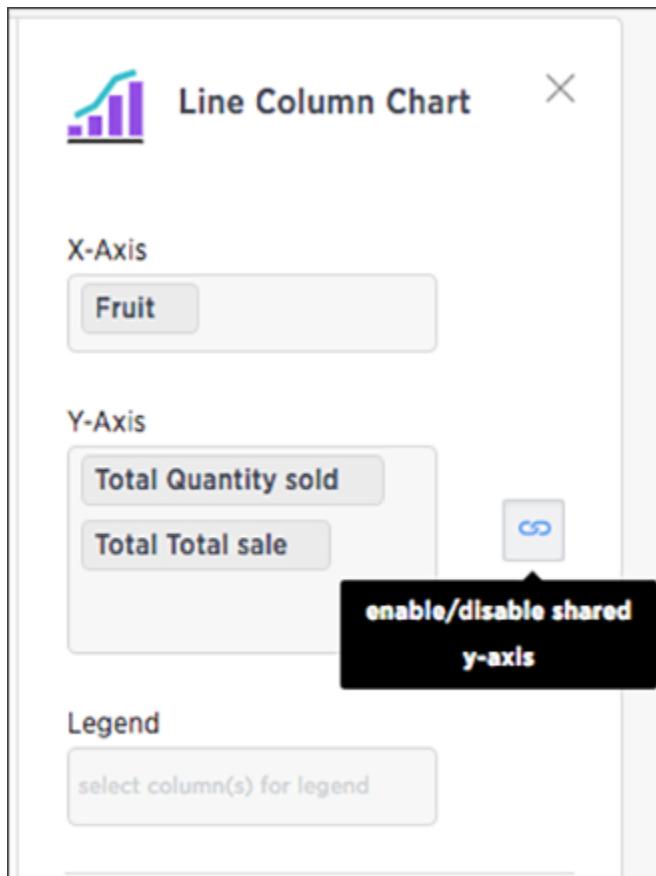


Line stacked column charts

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.



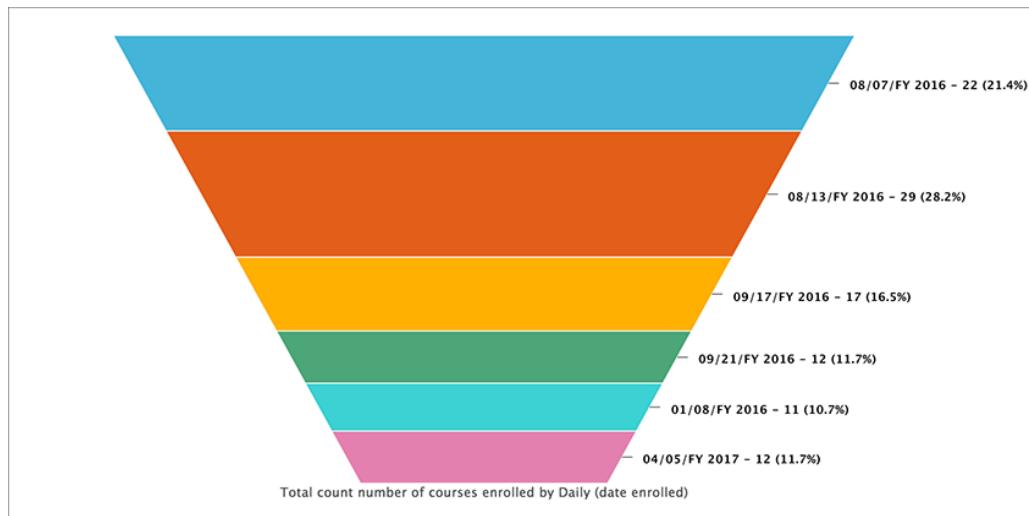
You can choose to enable shared y-axis by clicking the link icon found under **Edit chart configuration**.



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, depending on territory:

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

ThoughtSpot supports maps many countries. Please see the complete list in [Geo Map Reference](#).

Displayed Geo data

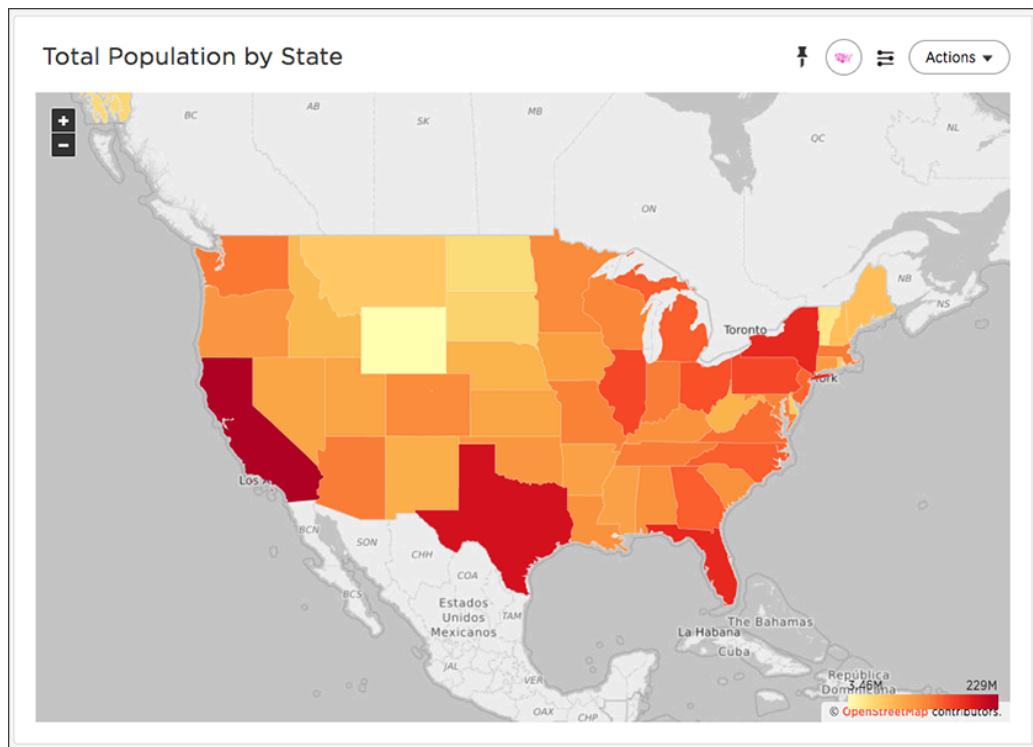
Here is a table that shows which GeoType data can be displayed using which geo chart type.

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 in the United States.
Other sub-nation regions	Geo area (default), geo bubble, geo heatmap	The display depends on the type of administrative region.

For data to display in geo charts, your administrator must configure it as geographical data. If you don't get an expected map visualization, contact your administrator so they can configure the system correctly.

Area charts

Geo area charts highlight the regions of interest, and display boundaries for various regions.

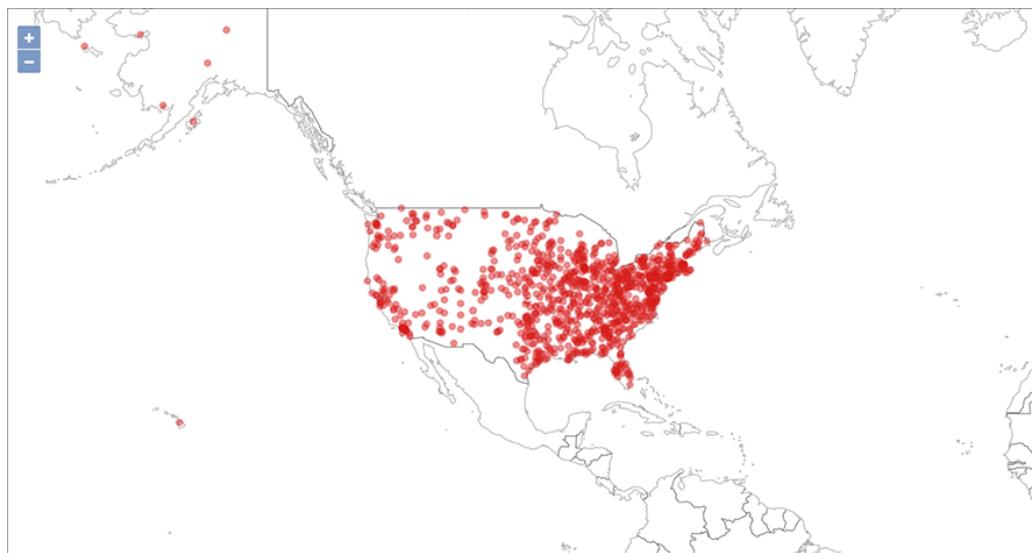


To paint a geo area chart, your search must include a geographical column with appropriate granularity.

See [Geo Map Reference](#).

Geo bubble charts

Geo bubble charts, like bubble charts, display the value of the measure through the relative size of the bubble. Zip code data is a good choice for geo bubble charts.

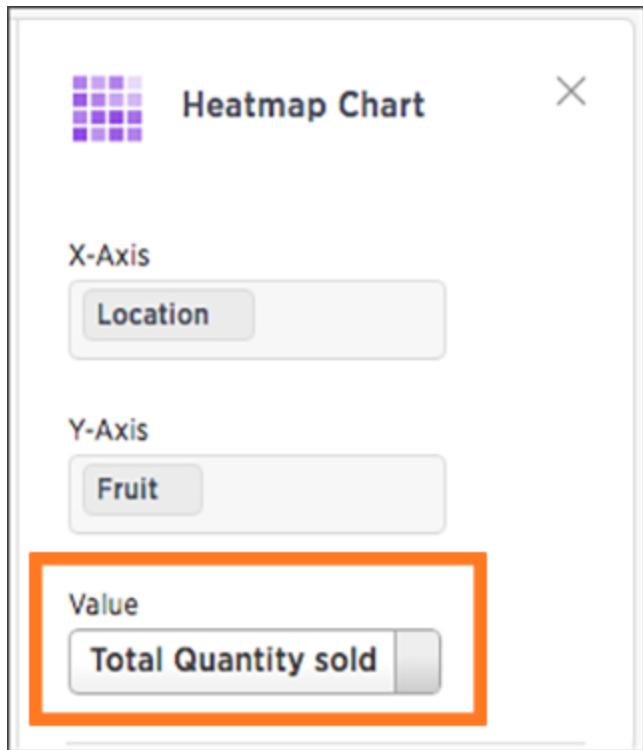


To paint a geo bubble chart, your search must include a geographical column, or a latitude and longitude pair.

Heatmap charts

Heatmap charts are similar to treemap charts: 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.





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.

The screenshot shows the ThoughtSpot interface. On the left, there is a data table with rows labeled 1 through 10. The first row has a header 'Average Monthly Sales'. Subsequent rows show numerical values: 717,162.96, 574,198.49, 326,973.47, 307,175.09, 209,847.15, 3,104.97, and 356,410.35. To the right of the table is a 'Chart' panel titled 'Chart'. This panel contains a grid of visualization icons. The icons are arranged in three rows. The first row includes 'Bubble', 'Pareto', and 'Waterfall'. The second row includes 'Treemap', 'Heatmap', and 'Line Column'. The third row includes 'Line Stacked Column', 'Funnel', and 'Geo Area'. At the bottom of the third row, the 'Pivot Table' icon is highlighted with a light gray background. The entire 'Chart' panel has a light gray background.

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

Pivot table with multiple measures				
		Market Segment	Customer Region	
		▶ automobile		▶ furniture
Quarterly (Order..)	Order Priority	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	

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 a row heading, the system displays a contextual sort menu:

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

Expanding or contract columns and rows

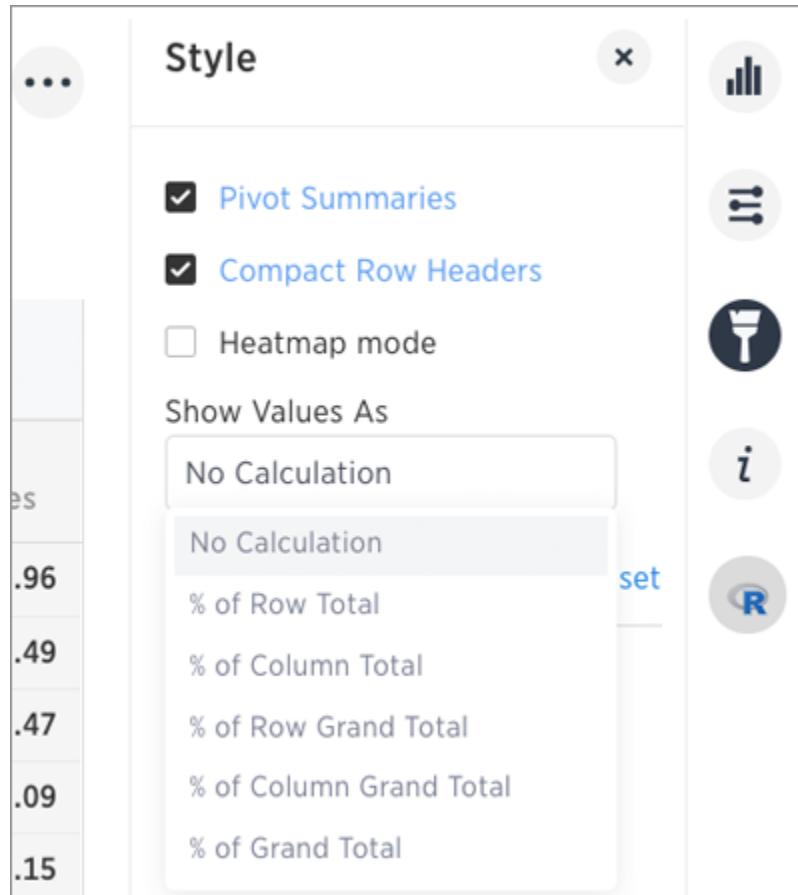
Click a column or row to expand it. Additionally, you can expand or collapse all by right clicking the arrow on the top left of a cell.



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 Sport Pivot							
Sport		United States			United States Total	United Kingdom			United Kingdom Total
		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}
Tug of War		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	
		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		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 a ThoughtSpot Pivot Table interface. On the left, there's a vertical list of metrics: TOTAL, AVG, MIN, MAX, TOTAL COUNT, UNIQUE COUNT, STD DEVIATION, VARIANCE, FILTER ..., and SORT. A context menu is open over a cell containing "hoc Impressions". The menu items shown are TOTAL, AVG, MIN, MAX, TOTAL COUNT, UNIQUE COUNT, STD DEVIATION, VARIANCE, FILTER ..., and SORT. To the right of the menu, the main table area shows a single row with columns: metric_name, user.guid.impression, and ee53-42be-a548-bd. The first column has a dropdown arrow icon.

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.

A screenshot of a ThoughtSpot table titled "Monthly Department Sales Analysis". The table has a dropdown filter "Department: fan shop, sports gear". The data is organized into three columns: "Average Monthly.." (row 1), "Department" (row 2), and "Yearly (Transacti..)" (row 3). The table contains the following data:

Average Monthly..	Department		
Yearly (Transacti..)	Fan Shop	Sports Gear	Average Monthly Sales
48972	944,665.99	489,659.92	717,162.96
47970	760,579.94	387,817.03	574,198.49
46971	440,016.37	213,930.58	326,973.47
45971	412,574.38	201,775.80	307,175.09
44972	265,370.19	154,324.10	209,847.15
43970	4,231.39	1,978.55	3,104.97
Average Monthly Sales	471,239.71	241,581.00	356,410.35

To the right of the table is a "Style" panel with checkboxes for "Pivot Summaries", "Compact Row Headers", and "Heatmap mode" (which is checked). It also includes options for "Show Values As" (set to "No Calculation"), "Color" (with a "Reset" button), and icons for "i" and "R".

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.

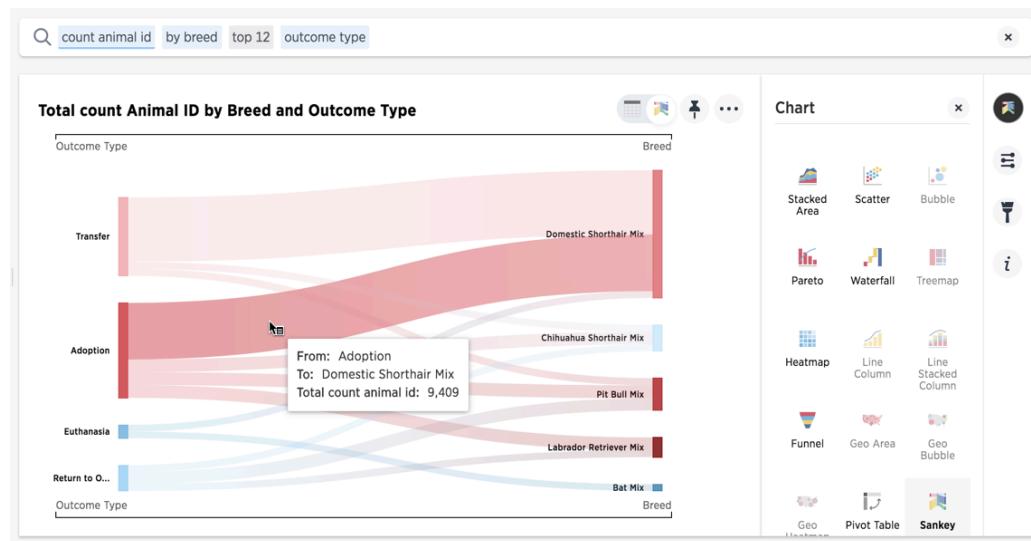
Sankey charts

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

Sankey diagrams illustrate a flow through a process or system. Transactional data is optimal for this type of diagram. For example, financial transactions that take place as money moves through accounts or as a product is processed. Many marketing users are familiar with Sankey charts from using Google Analytics to view sales conversions.

When you build a Sankey chart you need to provide at least 2 (two) attributes and one measure. Your x-axis attributes can contain at most 13 values, any more and you cannot view a SanKey chart.

ThoughtSpot Sankey charts are read from left to right. The width of the flow represents the measure, the attributes or “steps” appear as solid bars:



The [Austin Animal Center Outcomes](#) data shows the outcome of processing for different breeds of animal. The measure is the count of total animals. Looking at how an animal was classified when they entered the system, this chart shows the outcome processing that took the animal out of the system.

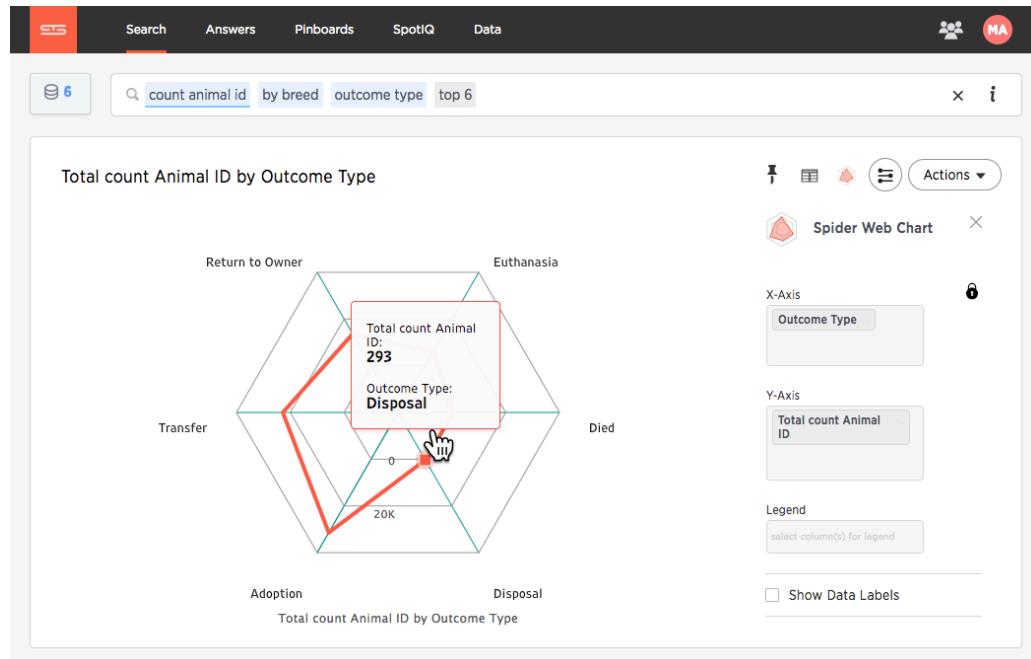
By clicking on a flow, you can see the total animals in any flow. Many domestic short haired cats (over 10k) were transferred to other agencies. While all bats that came into the system were euthanized. There were a surprising number of bats processed.

Radar charts; spiderweb charts

Summary: The Radar (or spiderweb) chart contains both columns and a special type of line chart.

A Spiderweb chart displays data in the form of a two-dimensional chart of three or more qualities, represented on axes starting from the same point. This is a good chart to use when you have asked users to rank an experience or product. You can use a spiderweb chart to examine the relative values for a single data point. You can also use it to locate outliers.

When you build a Spiderweb chart you must provide at least 3 (three) attributes and one measure. The measure values move from smallest to the outer edge of the web. Each spoke of the web is reserved for one of the variables. The points where each value lies on the web are connected.



The [Austin Animal Center Outcomes](#) data shows the outcome of processing. The measure is the count of total animals. By clicking on a spoke, you can see the total animals in any flow. The number of animals that were disposed of, meaning they were (sadly) already deceased on arrival is very small.

Candlestick charts

Summary: A candlestick chart describes price movements of financial instruments, such as stocks, derivatives, currencies, and commodities.

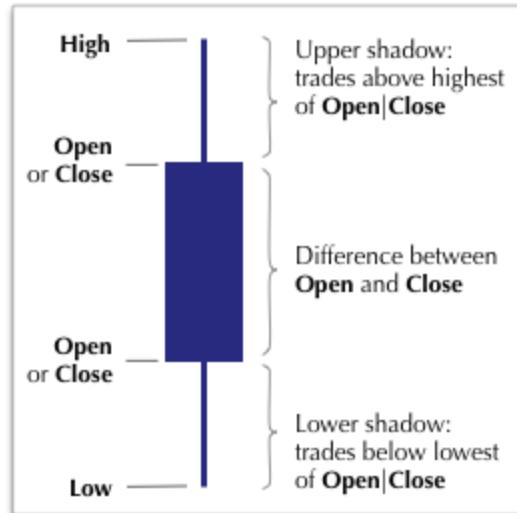
Candlestick charts enable fast decision making in stock, foreign exchange, commodity, and option trading because they describe price movements over time in terms of OHLC measurements: *open*, *high*, *low*, and *close* prices. Each “candlestick” shows data for one day, so a monthly chart shows approximately 20 candlesticks, one for each trading day. The date is on the horizontal (X) axis, and it accounts for non-trading days. This chart type can accommodate intervals shorter or longer than one day, provided each interval clearly identifies the four measurements.



How candlestick charts represent data

Candlestick charts have a very specific approach to representing data:

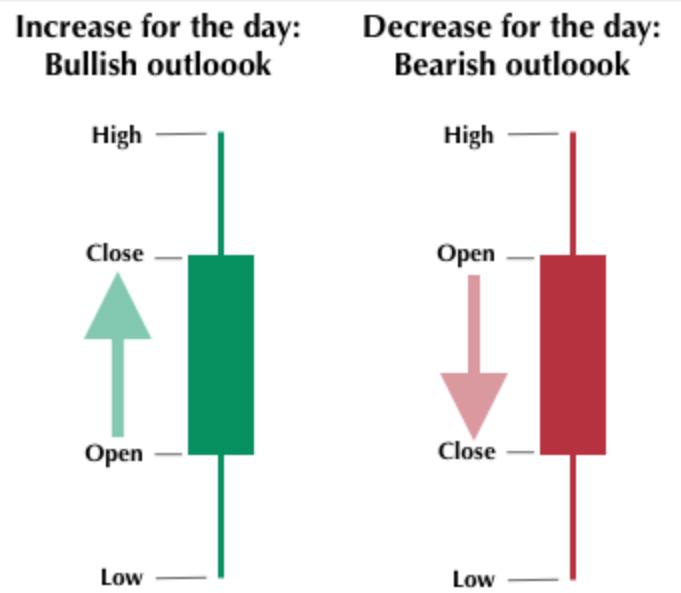
- The *High* and *Low* values plot at the top and bottom of the candlestick.
- The *Open* and *Close* positions vary, depending on the relative values of the price at opening and closing of the trading day.
- The height of the rectangle depends on the difference between the *Open* and *Close* prices, and does not indicate the volume of trades.
- The ‘wick’ that extends upwards from the rectangle indicates the difference between the highest of *Open* or *Close*, and highest traded value (*High*). Similarly for the extension downwards from the lowest of *Open* or *Close*, and the lowest traded value (*Low*).
- Your search must compute four measures. Additionally, it must have a “for each” specification for the horizontal axis. Most commonly, this is a a periodicity, such as `daily`.



The search question can be expressed in the following form:

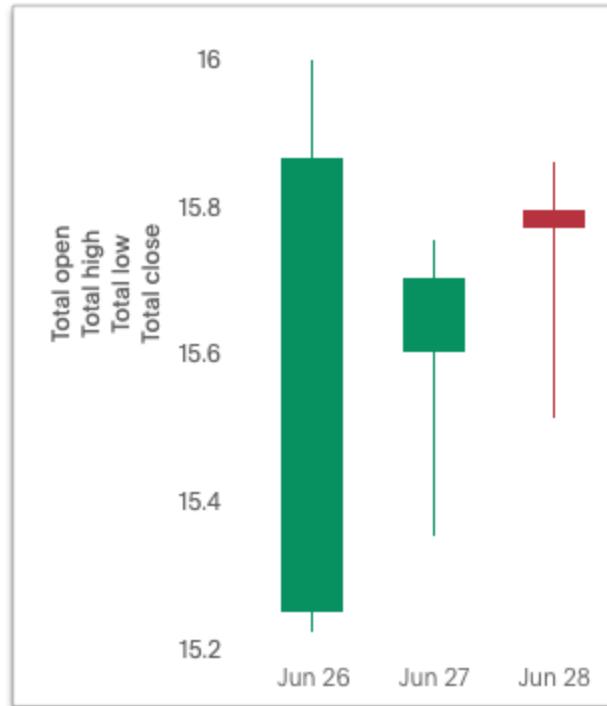
```
open high low close daily
```

- The candlestick chart does not show the volume of trades. However, the longer body of the candle generally correlates with the intensity of trading.
- When the rectangle is green and high relative to other time periods, this is an indicator that the market for the financial instrument is very bullish, with a likely "buy" recommendation.
- Alternatively, the red rectangles mean that the price is decreasing, and this pattern over time describes a bearish "sell" market for the instrument.



Interpreting candlestick charts

The following excerpt shows candlestick data for only three dates, June 26 - 28, 2019.



Date	Open	High	Low	Close
06-26-2019	\$15.30	\$16.00	\$15.20	\$15.90
06-27-2019	\$15.60	\$15.80	\$15.40	\$15.70
06-28-2019	\$15.80	\$15.90	\$15.50	\$15.80

Notice the following features of this chart:

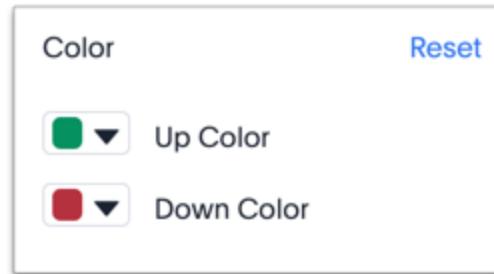
- For both June 26th and 27th, the *Close* price is higher than the *Open* price. The candlesticks for these days are rendered in green.
- The close is lower than the open on June 28th, so the candlestick is red.
- June 26th has the highest difference between its *Open* and *Close* prices. You can easily determine this based on the height of the rectangle.
- The top-to-bottom size of the wicks vary with the *High* and *Low* prices. The range of prices in individual trades is greatest on June 26th, at \$0.80. The range is identical, at \$0.40, for both June 27th and June 28th.
- The *Open* for one day almost never coincides with the *Close* of the previous day. This is due to after-hours trading.

Color customization in candlestick charts

There are two color choices for this chart type:

- Up Color
- Down Color

ThoughtSpot uses Green and Red, respectively, as default.



Each candlestick renders in one of the two options, depending on the relative values of `open` and `close` measurements.

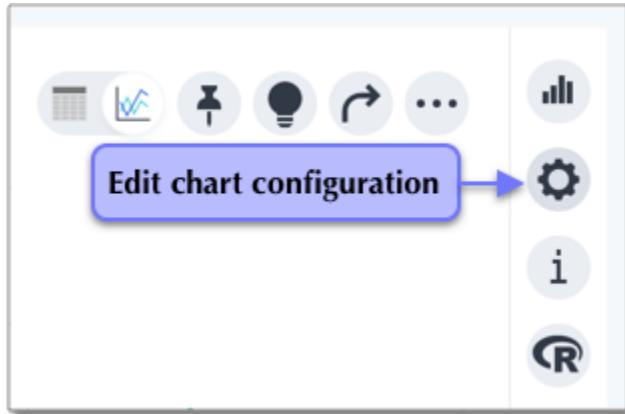
Adapting candlestick charts for other data

You can use this chart to display other statistical data, as long as you provide four measurements in addition to specifying the horizontal axis.

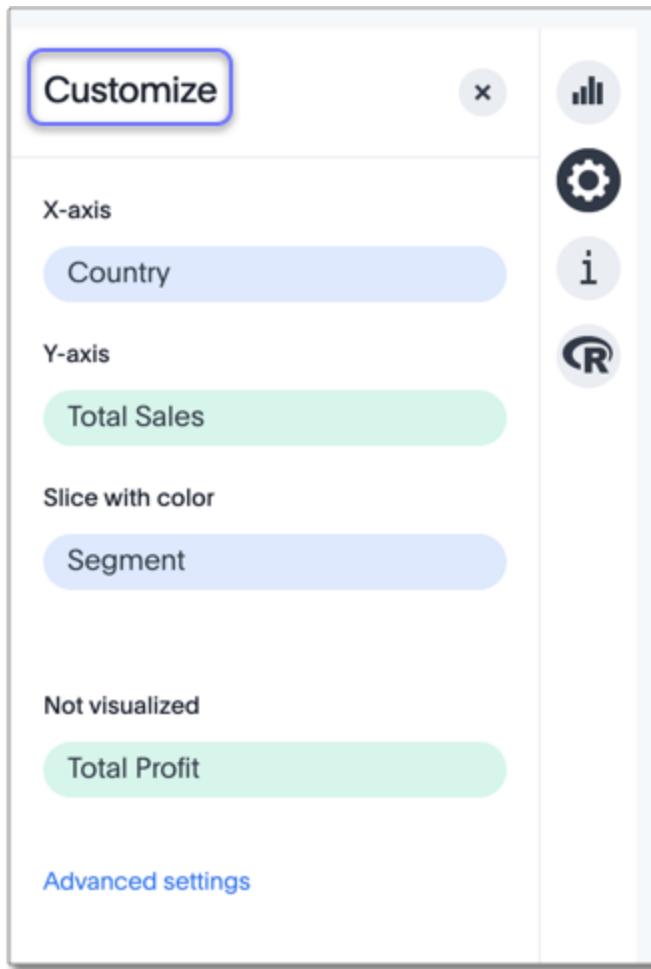
Change a chart

Summary: You can adjust all aspects of ThoughtSpot charts: color, legends, axis, number format, and many more .

To make changes to the chart configuration, click the gear icon in the top right corner of the visualization.



This opens the **Customize** menu for the chart.



You can edit some of the elements of the chart directly, by clicking on them. For example, you can change the appearance of a chart [axis](#) by clicking it and making selections in the available options.

Through this menu, you can perform a number of customizations:

1. Editable 'drag and drop' chips
 - 'Not Visualized' section
2. Number formatting for all data labels
 - Category (Number, Percentage, Currency)
 - Units (K, M, B)
 - Decimals
 - Independent control for each measure on the chart
3. [Axes](#)

- Grouping and ungrouping of axes
- Axis renaming

See [Change the options on an axis](#)

4. Color configuration
5. Tables
 - Column renaming for chart and pivot tables
 - Wrap or clip options for long text
 - No line limit for headers in wrap mode

Use the configuration options to perform any chart adjustments, such as:

- **[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](#)**

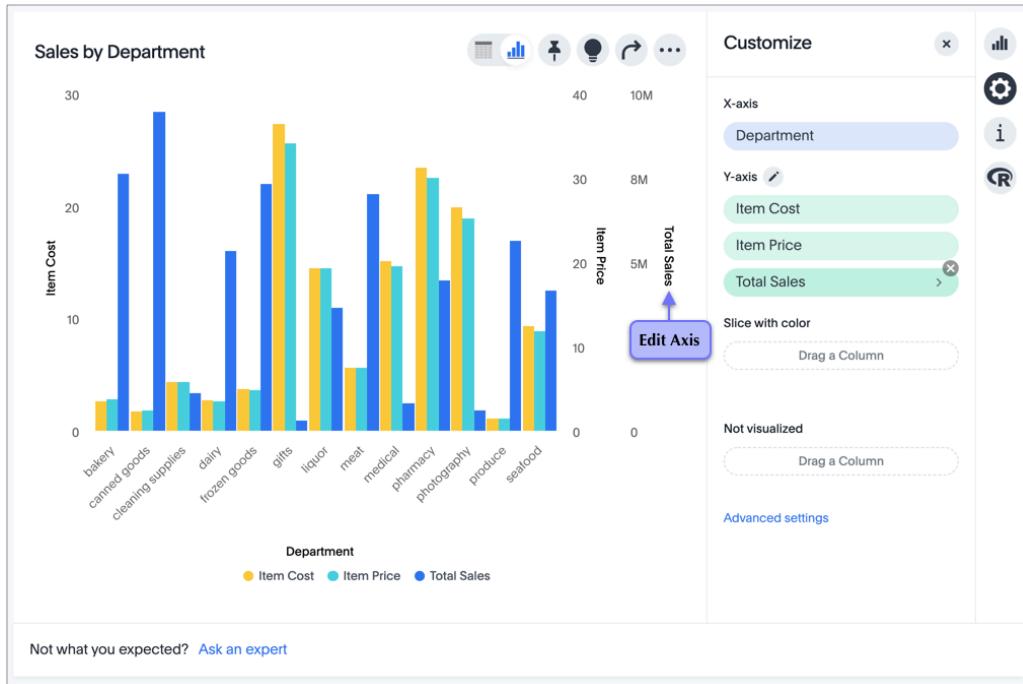
You can zoom into your chart by selecting an area with your mouse.

Change axis options

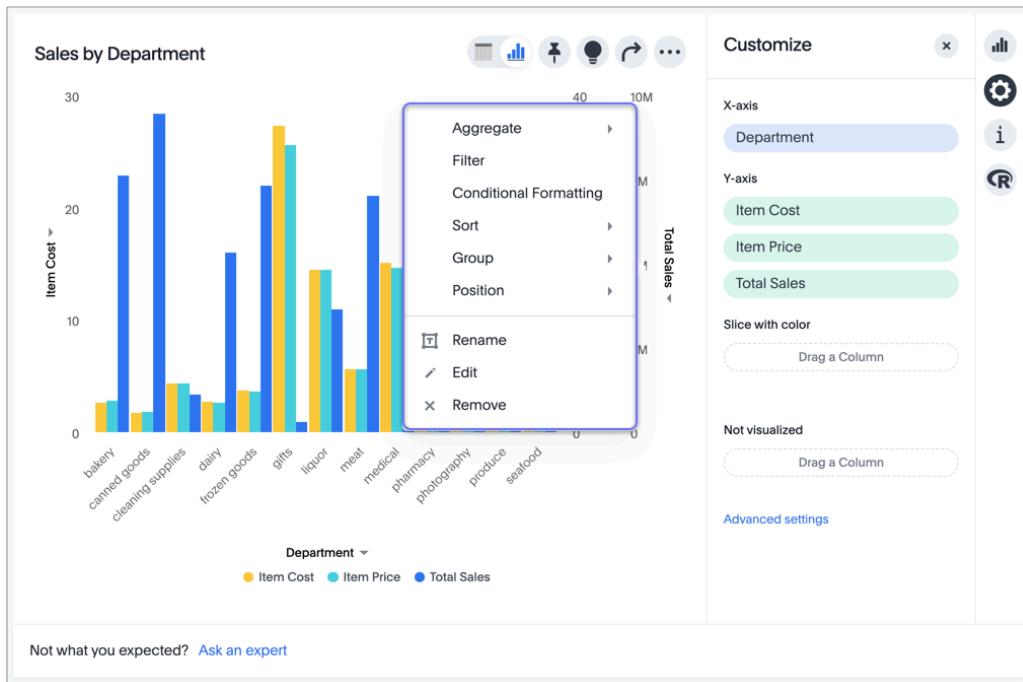
Summary: In ThoughtSpot, you can start changing all axes options by clicking on the axis, directly in the chart.

To change axis, you can click the axis itself.

Here, we demonstrate the various options for axis configuration using a visualization "Sales by Department"



This opens the detail axis customization menu in the chart.



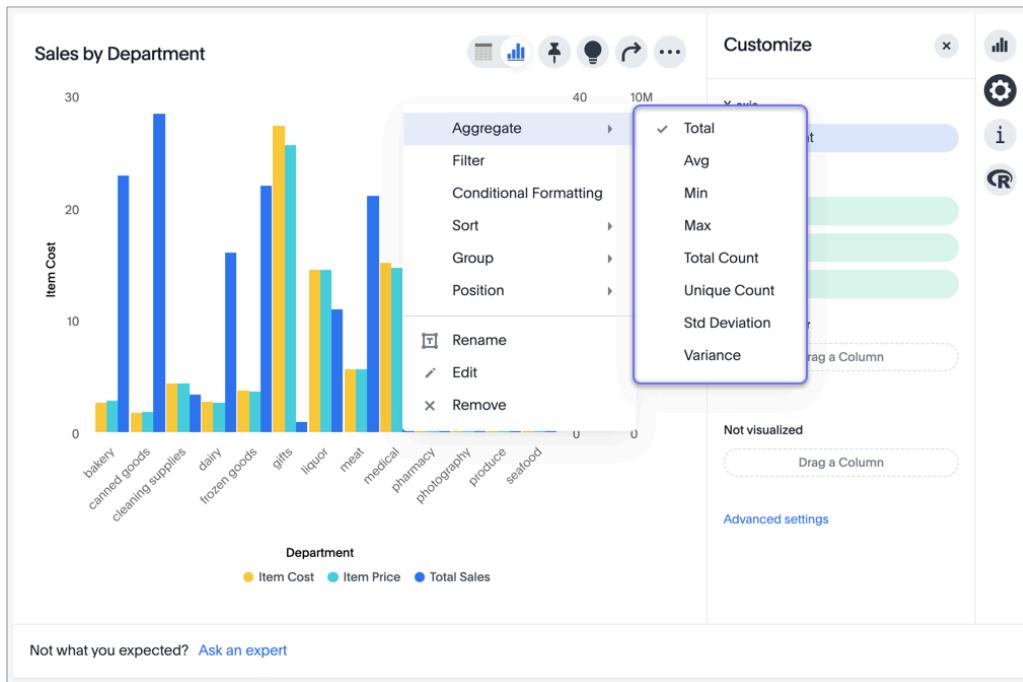
You can make the following changes to the axis:

- [Aggregate](#)
- [Filter](#)
- [Conditional formatting](#)
- [Sort](#)
- [Grouping](#)
- [Position](#)
- [Rename](#)
- [Edit](#)
- [Remove](#)

Change the axis aggregate

To change the axis aggregate, click the **Aggregate** option in the axis menu, and select an alternative from the list of options.

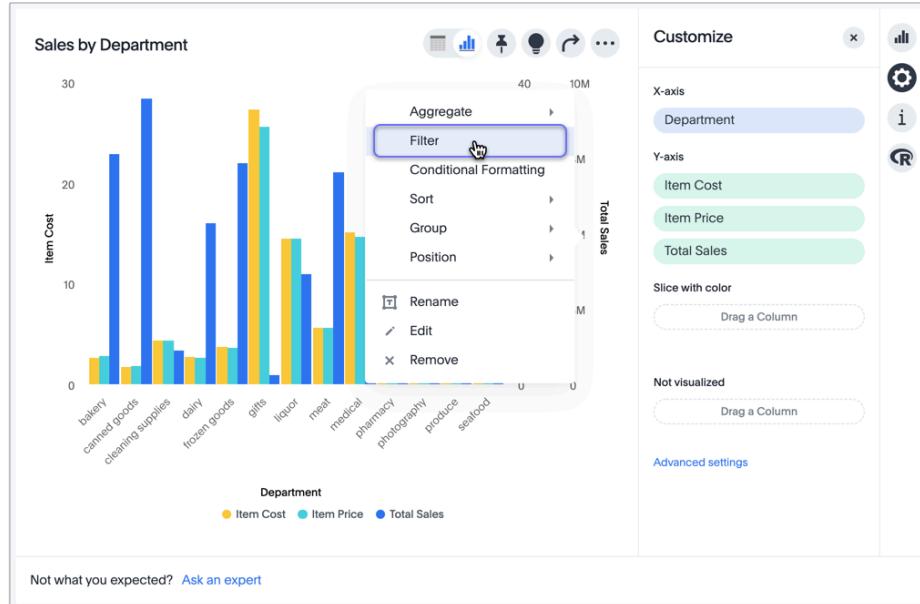
Depending on the type of data type and other settings, the aggregation options include *Total*, *Avg*, *Min*, *Max*, *Total Count*, *Unique Count*, *Std Deviation*, and *Variance*.



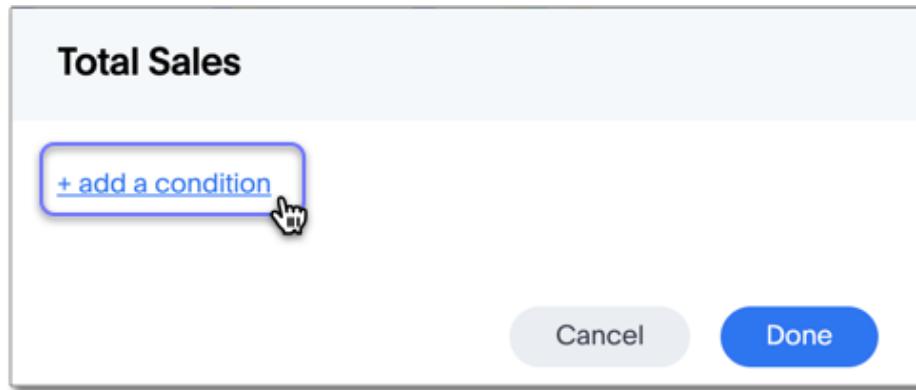
Add, Change, and Remove Filters

To manage the filters applied to the axis (in this example, *Total Sales*), follow these steps:

1. Click the **Filters** option in the axis menu.



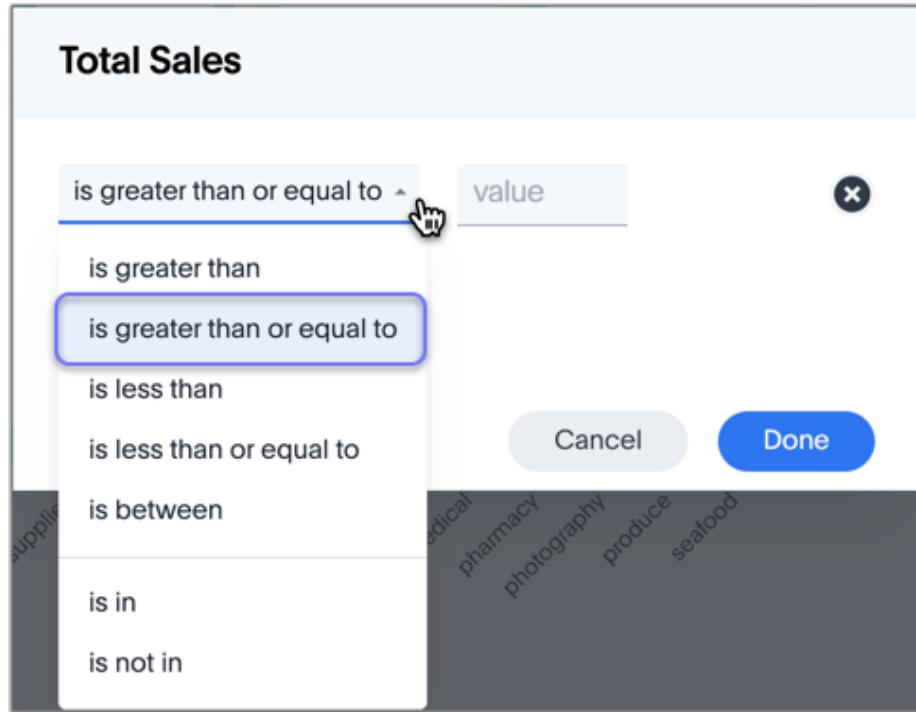
2. In the window modal, click **+ add a condition**.



3. Select the comparison option.

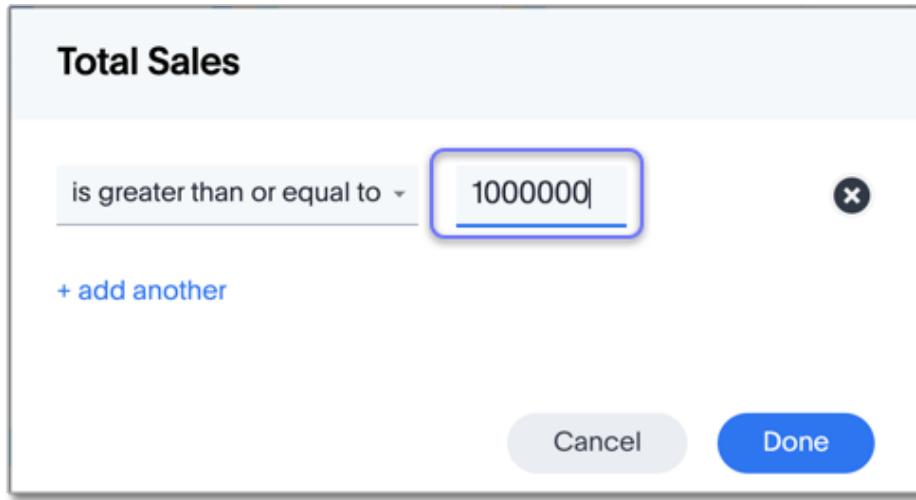
Here, the options include *is greater than*, *is greater than or equal to*, *is less than*, *is less than or equal to*, *is between*, *is in*, and *is not in*.

In this example, select *is greater than or equal to*.



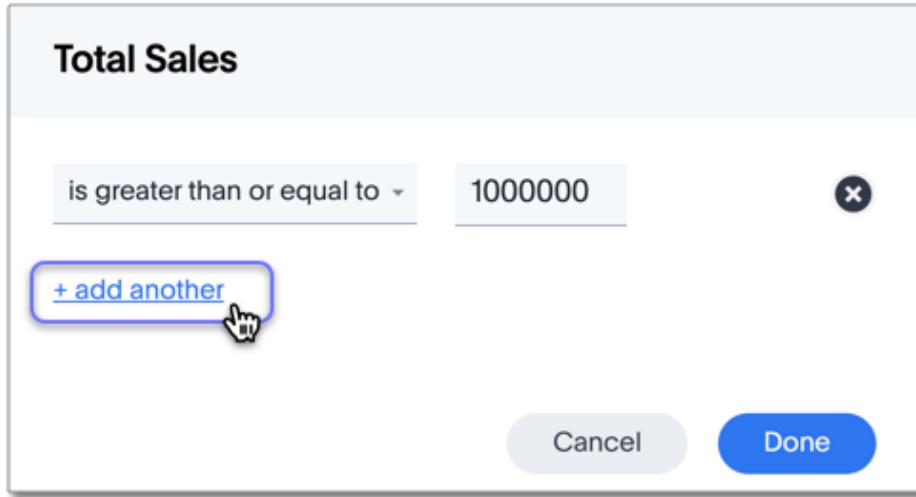
4. Set the relevant numerical limits.

Here, enter the maximum of 1000000.



Note that if you select the *is between* comparison operator, you must set two limits: minimum and maximum. Similarly, *is in* and *is not in* operators require a list of values separated by ";".

5. To add another filter condition, click **+add another**.



6. To remove a filter condition, click the delete icon (x) next to its definition.

Total Sales

is greater than or equal to ×

is greater than × ✖️

+ add another

Note: Results matching all the above conditions will be shown.

Cancel Done

7. Click **Done**.

Total Sales

is greater than or equal to ×

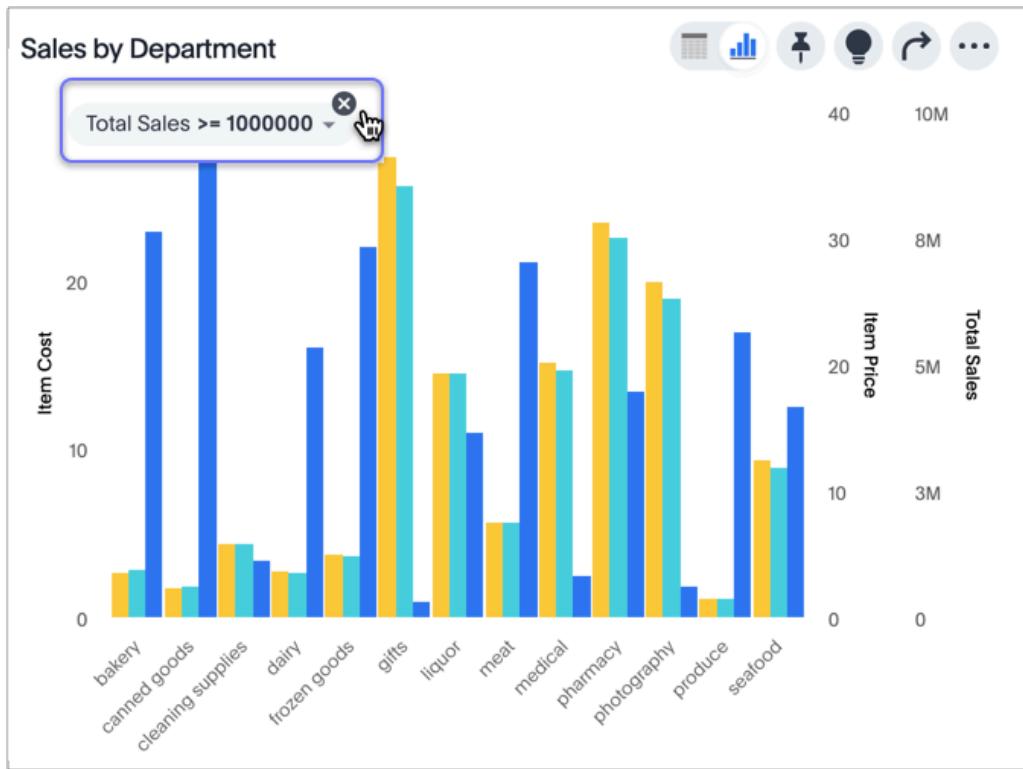
+ add another

Cancel Done ✖️

Here, you can see differences between the original chart and the one with the filter applied: Three (3) of the departments had sales under \$1,000,000, and they don't appear on the newer chart.



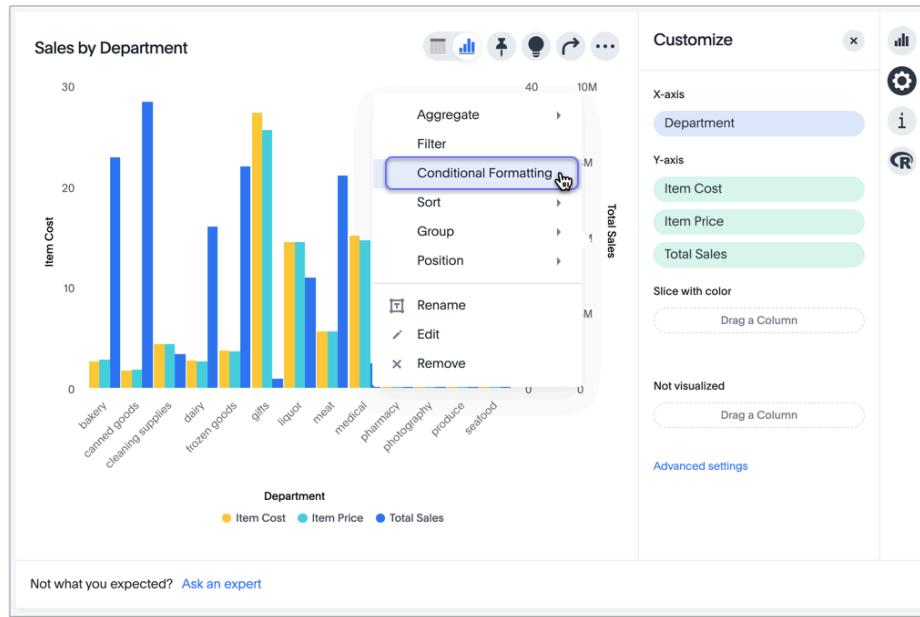
To remove a filter from the chart, click the **x** icon on the filter tile.



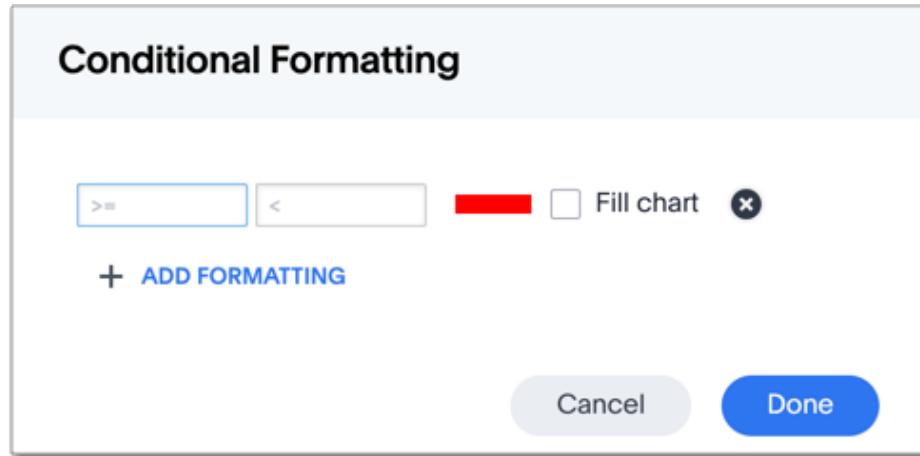
Apply conditional formatting

To apply conditional formatting to the axis (in this example, *Total Sales*), follow these steps:

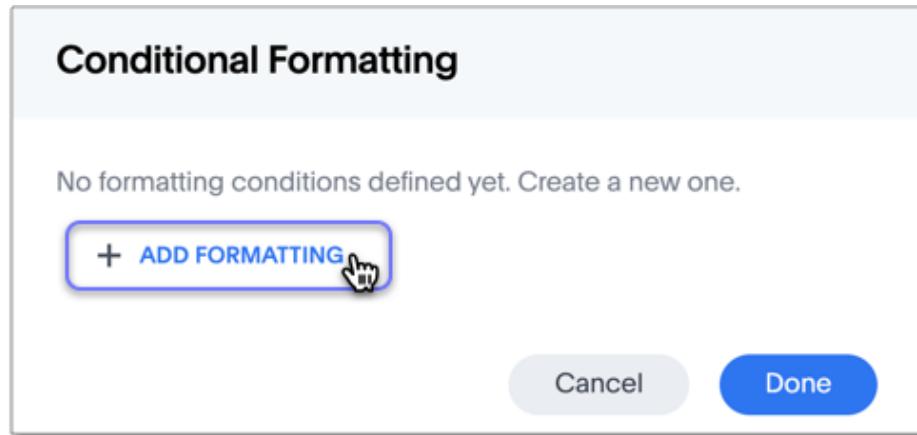
1. Click the **Conditional formatting** option in the axis menu.



2. The **Conditional formatting** window modal appears.



3. In the window modal, click **+ add formatting**.

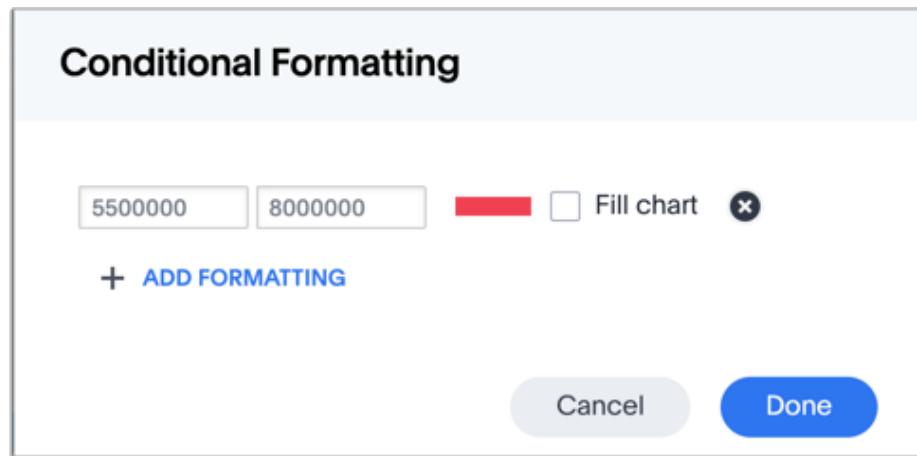


4. Select the conditional range.

The text area on the left specifies the lower limit of the condition, so the condition applies to all values that are greater or equal to it. And the text area on the right specifies the upper limit of the condition, so the condition applies to all lower values.

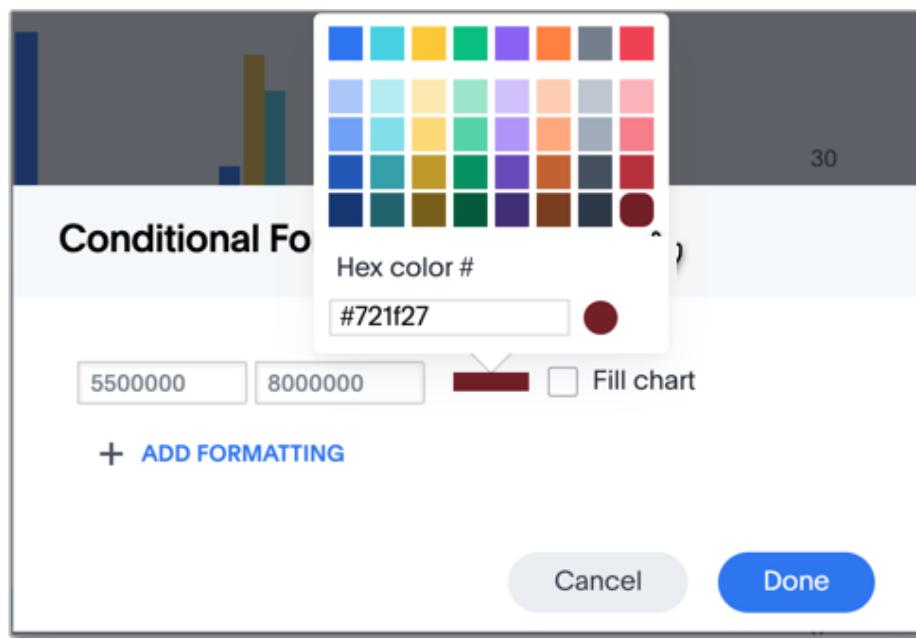
- To show values with a lower limit, enter a value in the left text box.
- To show values with an upper limit, enter a value in the right text box.
- To show values in a specified range, enter the appropriate values in both text boxes.

Here, we apply conditional formatting to values between 5,500,000 and 8,000,000.

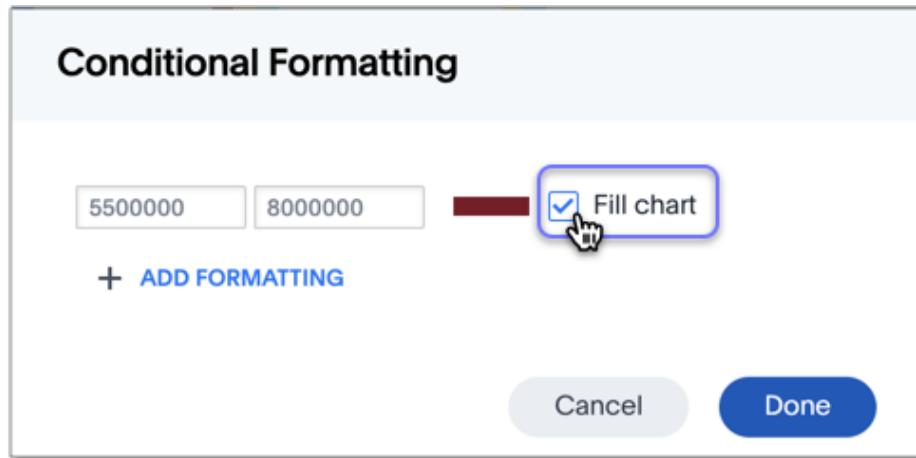


- To specify a different color of the conditional format, select the new color from the color selector.

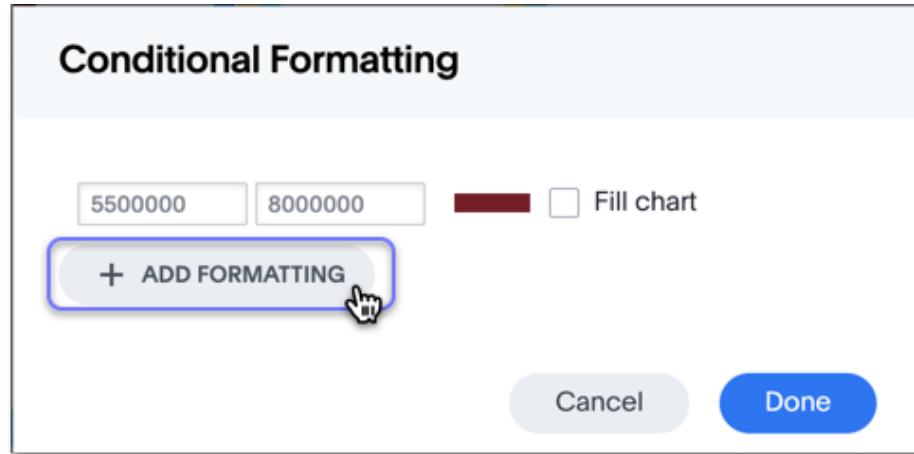
This option draws upper and lower limit lines on the chart, and colors the chart elements that meet the conditional requirements.



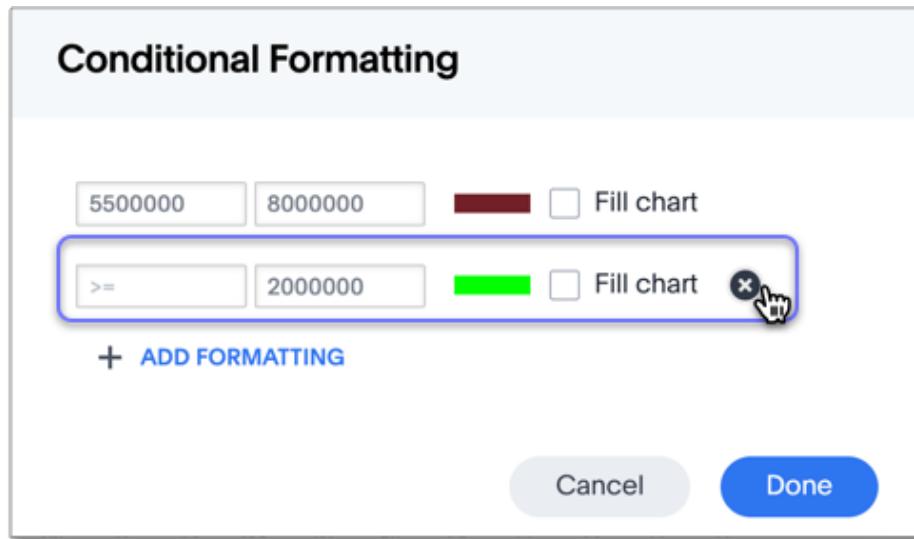
Alternatively, you can place a range band on the chart. Select the **Fill chart** option.



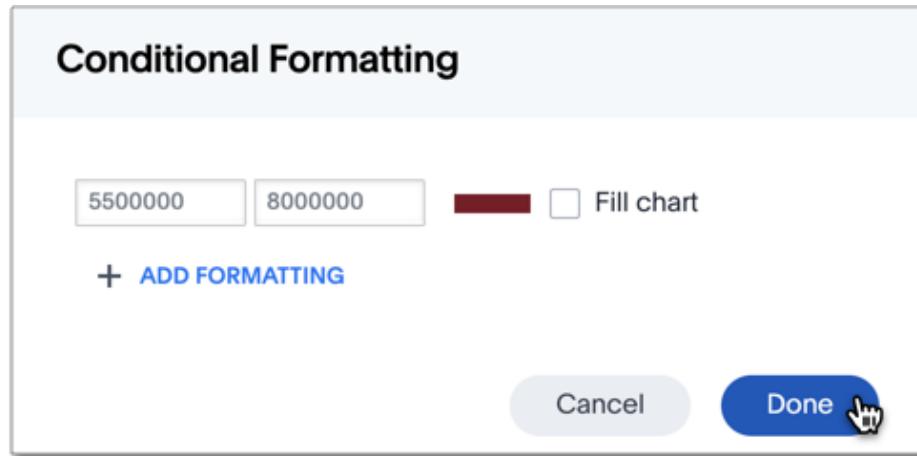
- To add another condition, click **+add formatting**.



7. To remove a defined conditional format, click the delete icon (x) next to its definition.



8. Click **Done**.

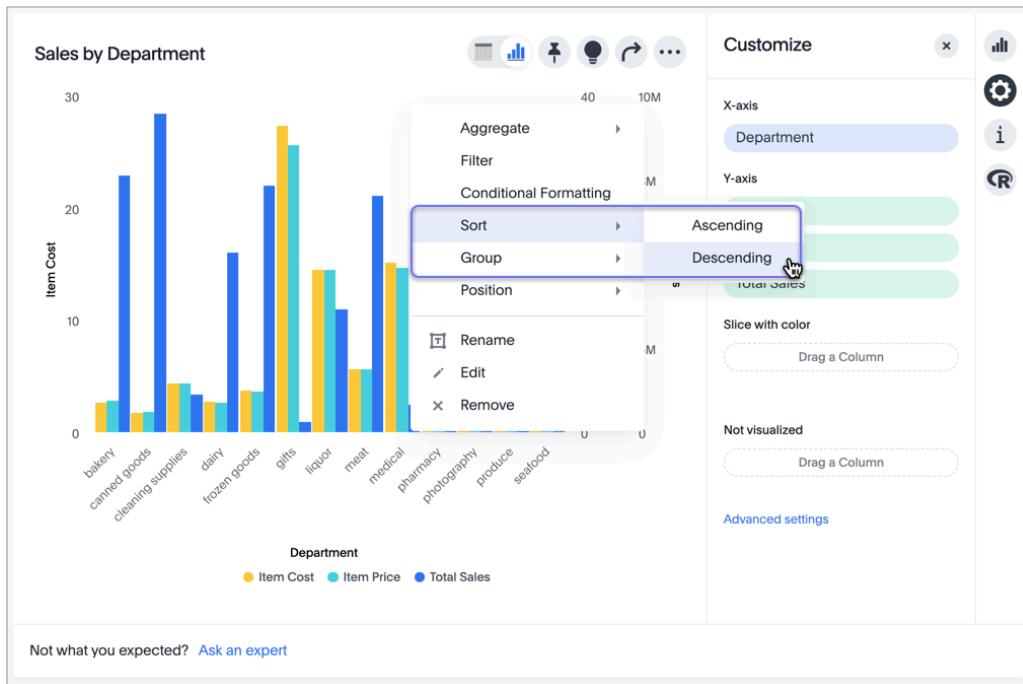


Here, you can see a chart that highlights elements with conditional formatting on some elements. You can also see how the same chart appears with a background chart band.

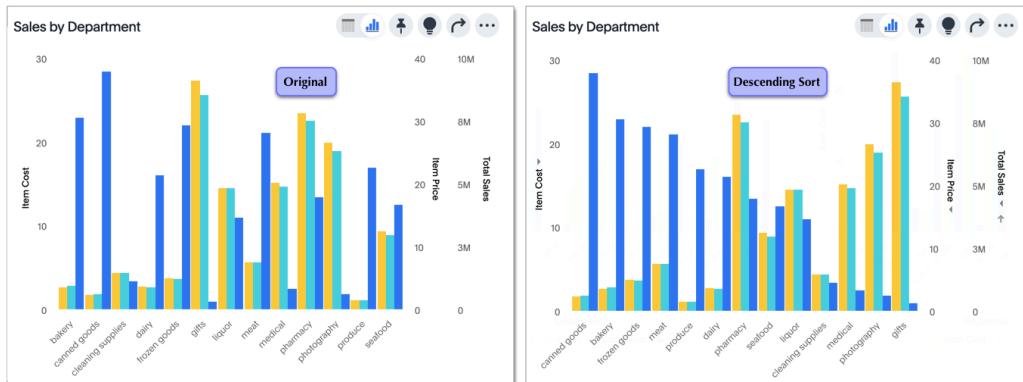


Change the Sort

To change the sorting of a measurement on an axis, click the **Sort** option in the axis menu, and select an alternative from the list of options: *Ascending* or *Descending*.



Here, you can compare the original chart that was not sorted on the *Total Sales* axis with the chart that uses descending sort.

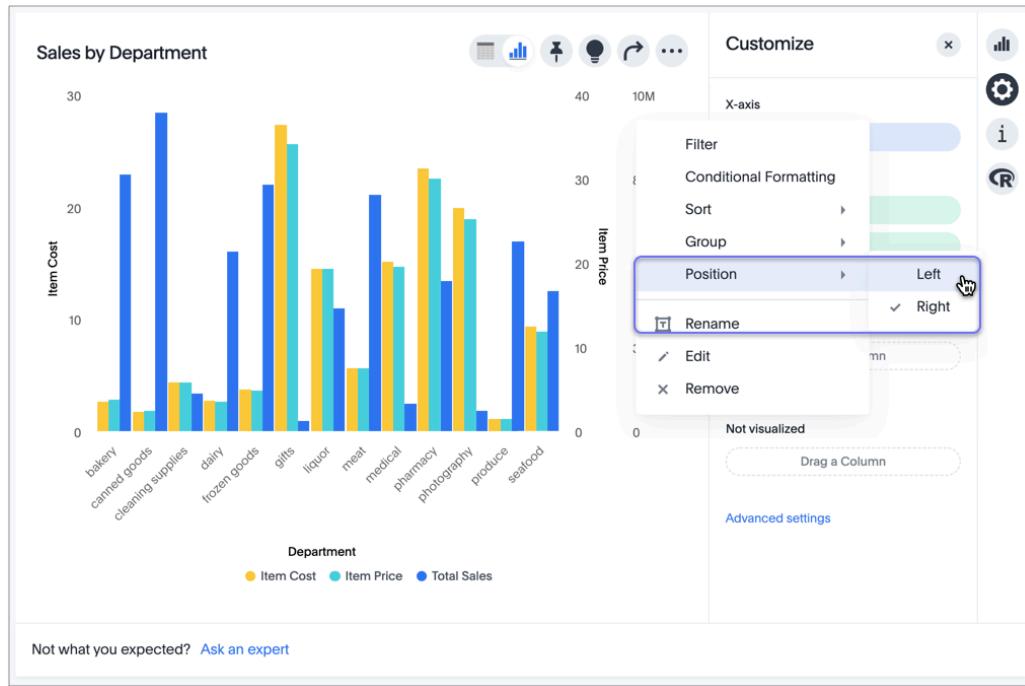


Change the Postion of the axis

It is generally easier to interpret a chart if axes that use the same units of measurement or scale appear on the same side of the chart. In our example, we can best visualize *Item Cost* and *Item Price* on the same side of chart.

To change the position of an axis, select the **Position** option in the axis menu, and then select an alternative from the list of options: *Left* or *Right*.

Here, we move the *Item Price* axis from the right side of the chart to the left side.



You can compare the original chart with the one where the *Total Sales* axis is on the right, while *Item Cost* and *Item Price* both appear on the left.

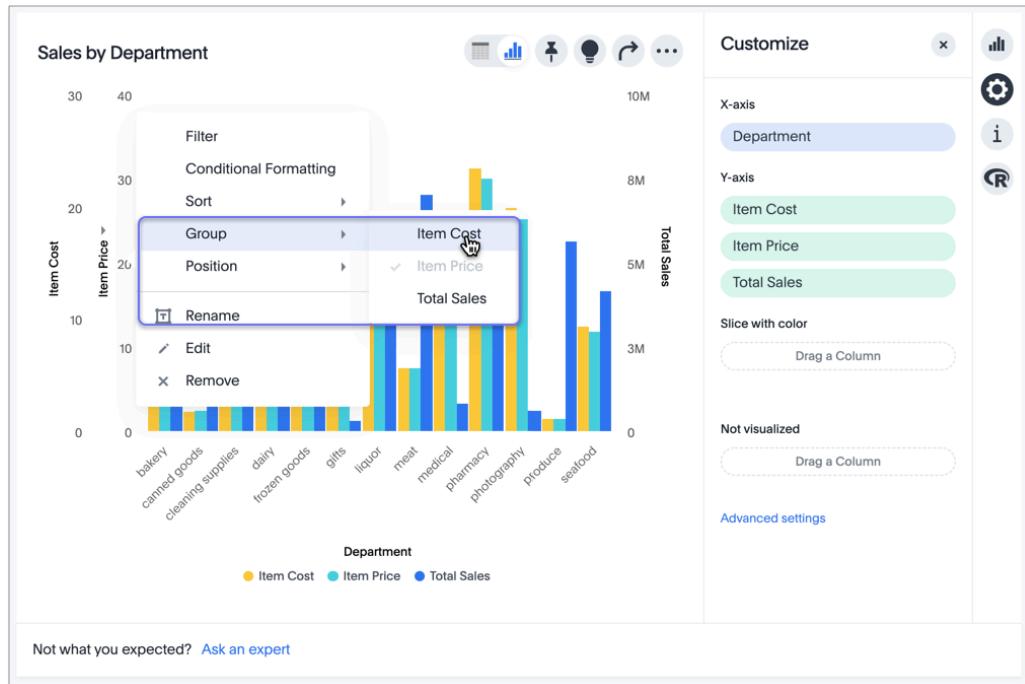


Change the Grouping

When two axis use the same unit of measure and a similar scale, we can group them together.

To change the grouping on an axis, click the **Group** option in the axis menu, and select an alternative from the list of options, which are the measurements on the other axes.

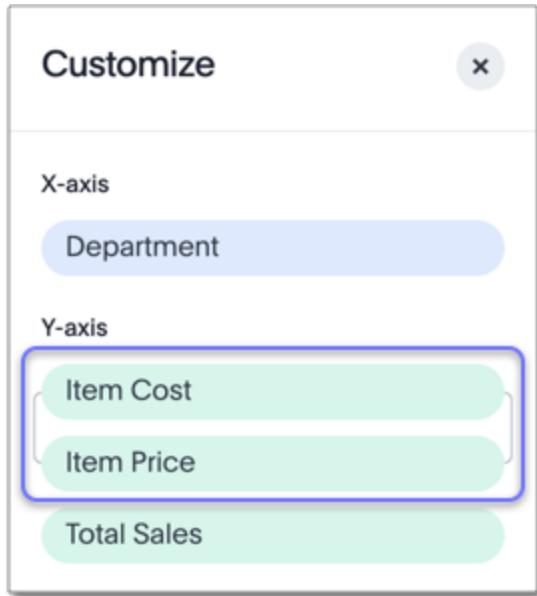
Here, we change the *Item Price* axis by grouping it with *Item Cost*.



Compare the original chart with one that groups *Item Price* and *Item Cost* as *Item Price & Item Cost*. The chart looks cleaner, and clearly communicates the distinct information on each of the two measurements.



Notice that the **Customize** menu shows a linkage between the two grouped axes.

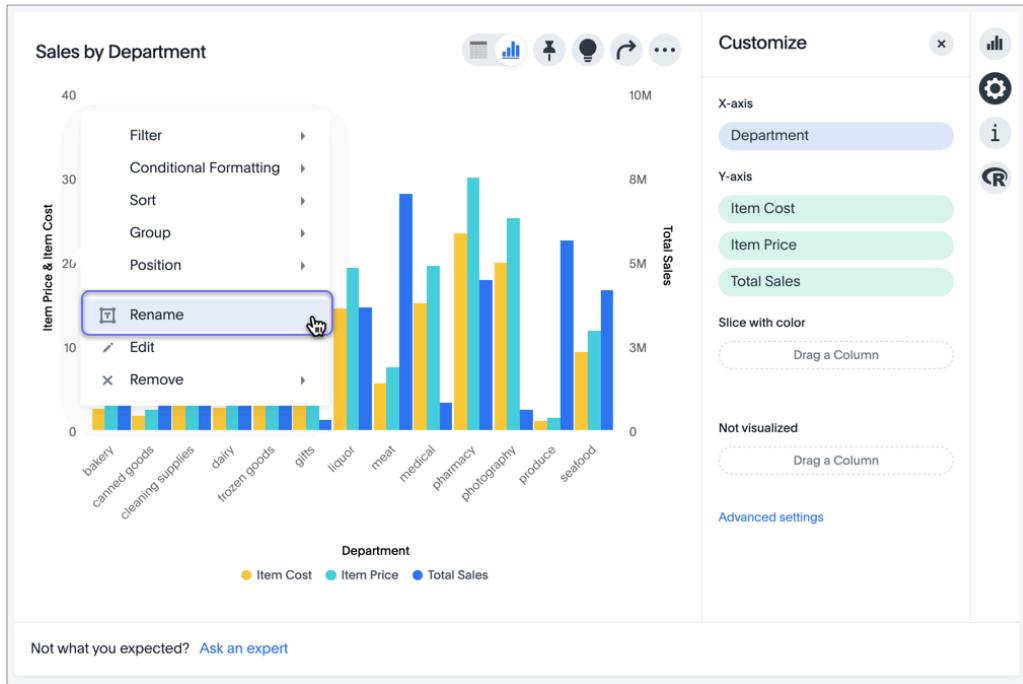


Rename an axis

You can always rename an axis for clarity, brevity, format, and so on.

In our example, it makes sense to rename the axis created from grouping as *Item Price & Item Cost* to something shorter, like *Item Price and Cost*.

To rename an axis, select the **Rename** option in the axis menu, type the new name, and either click out or hit **Enter/Return** on your keyboard.



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.

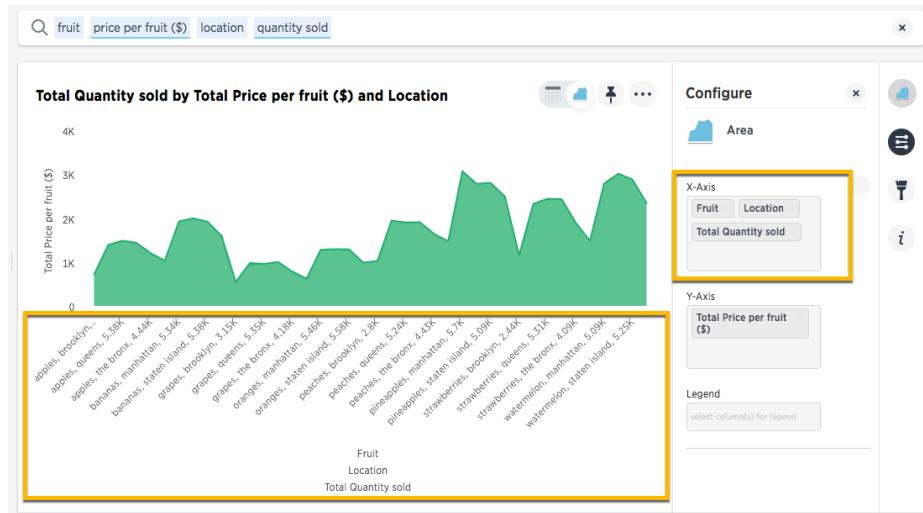
The order in which columns appear on the axis or legend is based on the sequence they are added. The first field is used as the primary sorting field for the chart. Adding another field adds it after the first one on the axis label or legend. If you want to change the order, you can remove the fields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



Your chart reorganizes itself to reflect the new label order.

Set the y-axis range

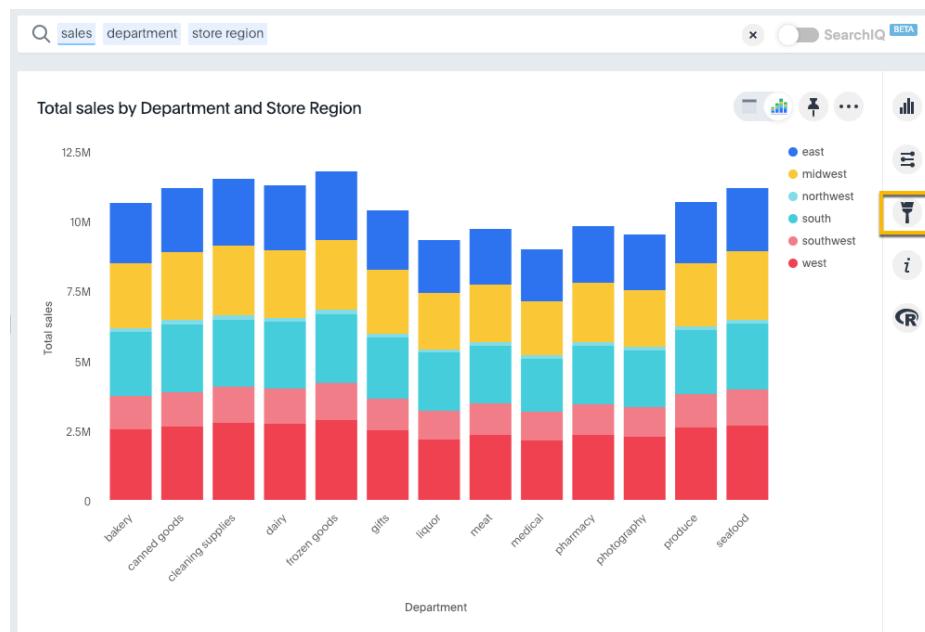
Summary: On charts, you can manually configure the y-axis range to be different from the search default. You can have multiple measures on the y-axis of many charts.

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

Setting the y-axis range

The style 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 styles**.

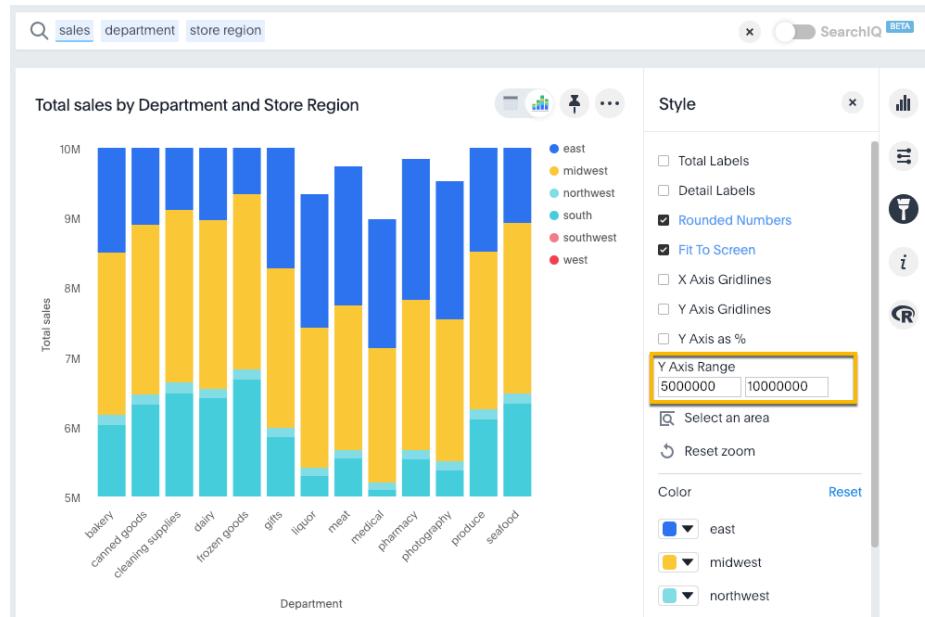


2. Under **Y-Axis Range** enter your preferred start and end values.

In this example, our original chart is showing sales values by department from 0 to over \$10M.

We can enter a y-axis range of `5000000` to `9000000` to show only sales between \$5M and \$10M.

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



Create charts with multiple measures on the y-axis

You can have multiple measures on the y-axis of most (but not all) chart types. You can configure this on the chart styles panel. To learn more, see these topics:

- [Configure columns for X and Y axis](#)
- [Charts with multiple measures on the y-axis](#)

Hide and show values

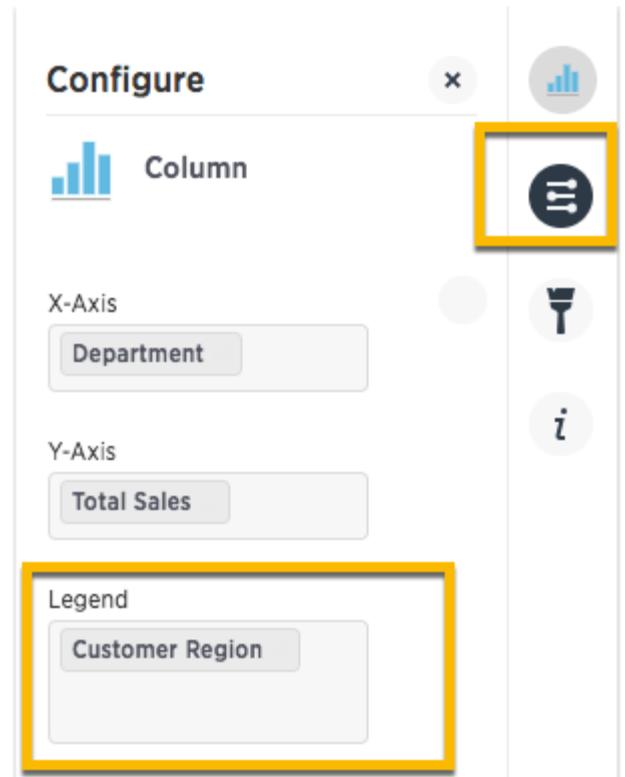
Summary: You can hide and show values on the chart using the legend.

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.

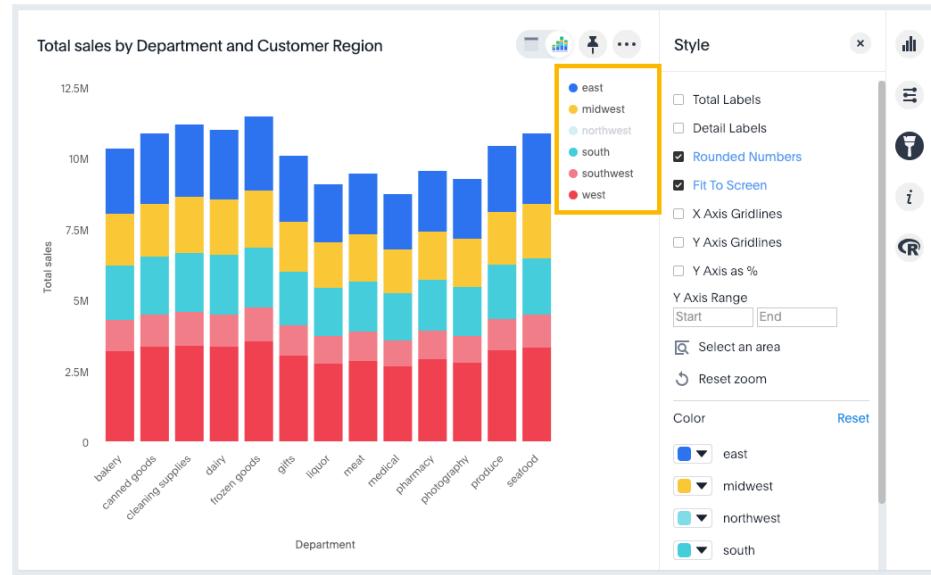


2. Make sure you have a column in the **Legend** field.



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

In the following example, we clicked `northwest` to hide it.



Change chart colors

Summary: You can change the colors used in a chart.

To change the colors used in a chart, use the color picker on the chart styles panel. You can set colors for:

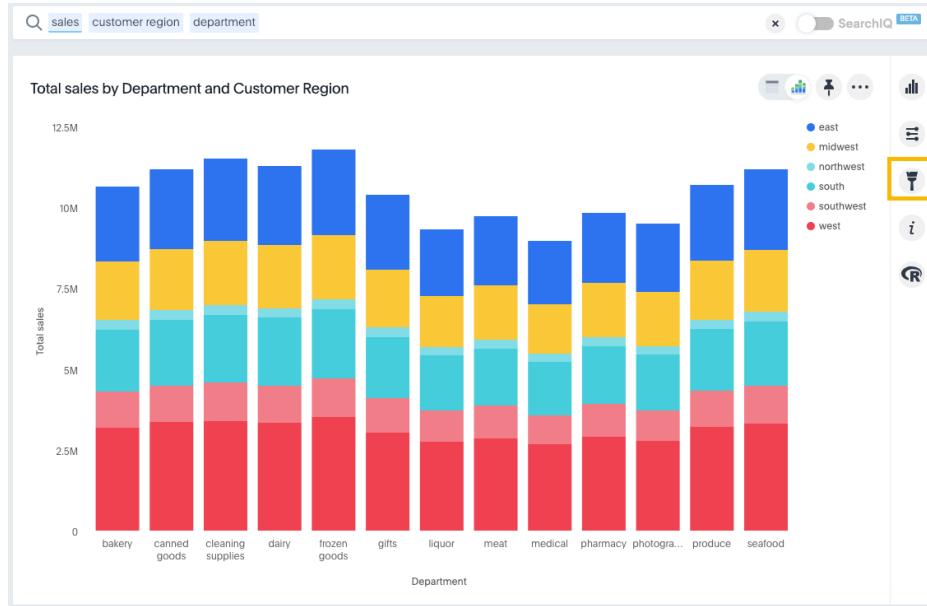
- bars, lines, areas, bubbles, maps, and other visuals on charts
- legends on charts
- tables for conditional formatting
- tags and editing the colors

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.

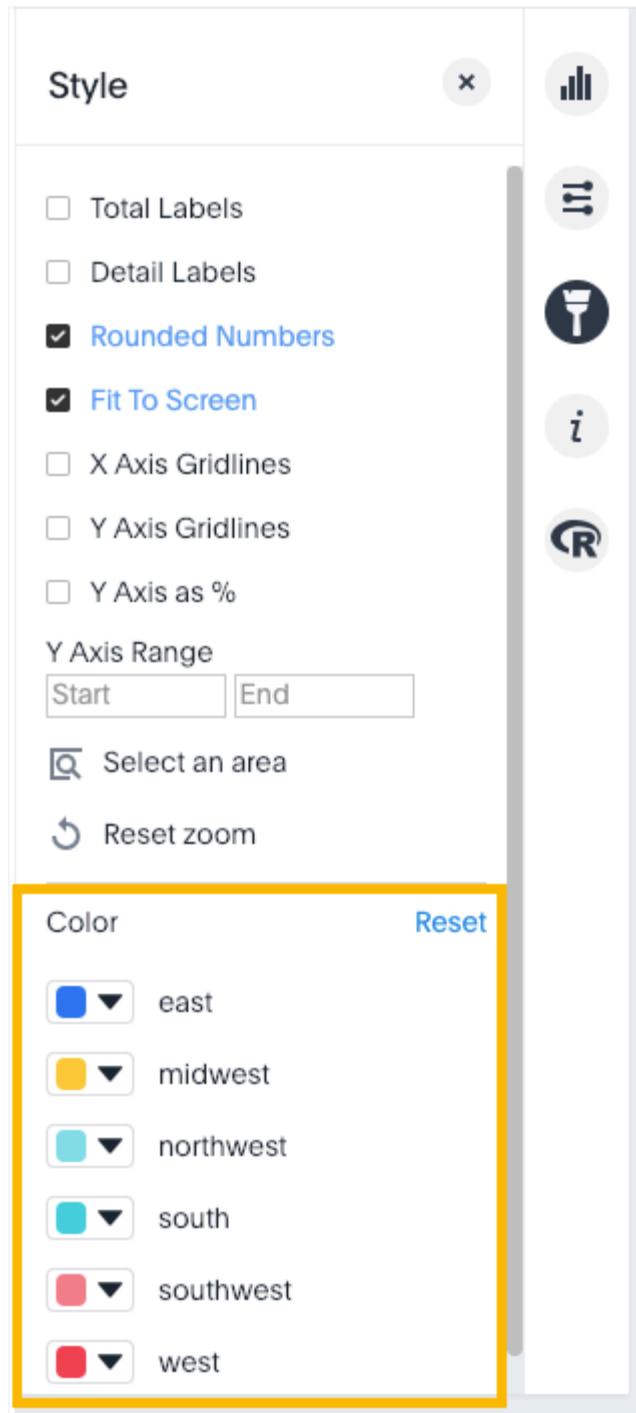
Colors are maintained across searches within a session. For example, when doing a search on revenue by state, each state will keep its color assigned to it even if you change the search or chart type.

Set colors on a chart

1. While viewing your search or answer as a chart, click **Edit chart styles** () to open the Styles panel.

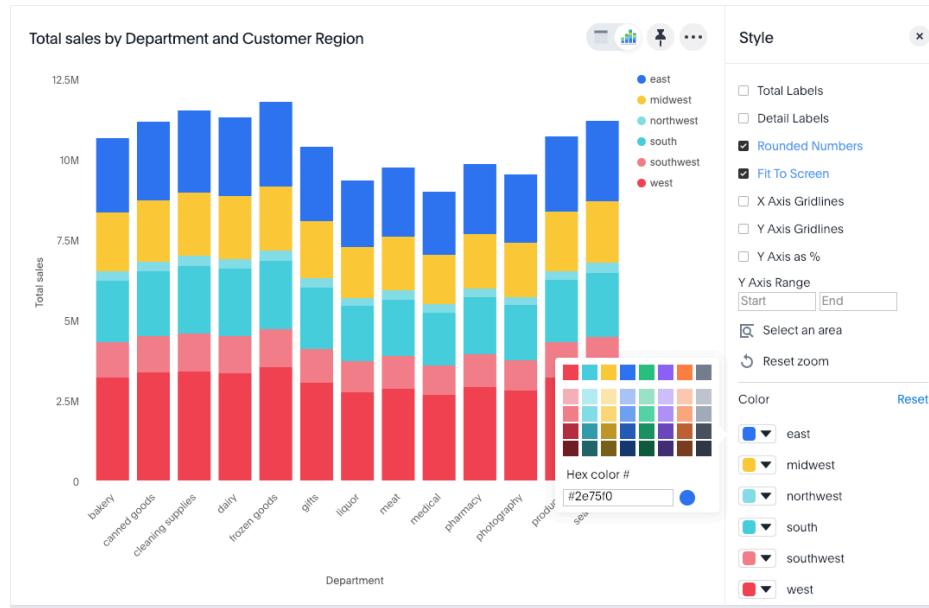


2. On the chart, click the color icon of the label or legend value you would like to change the color of.



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

You can also enter a HEX value directly.



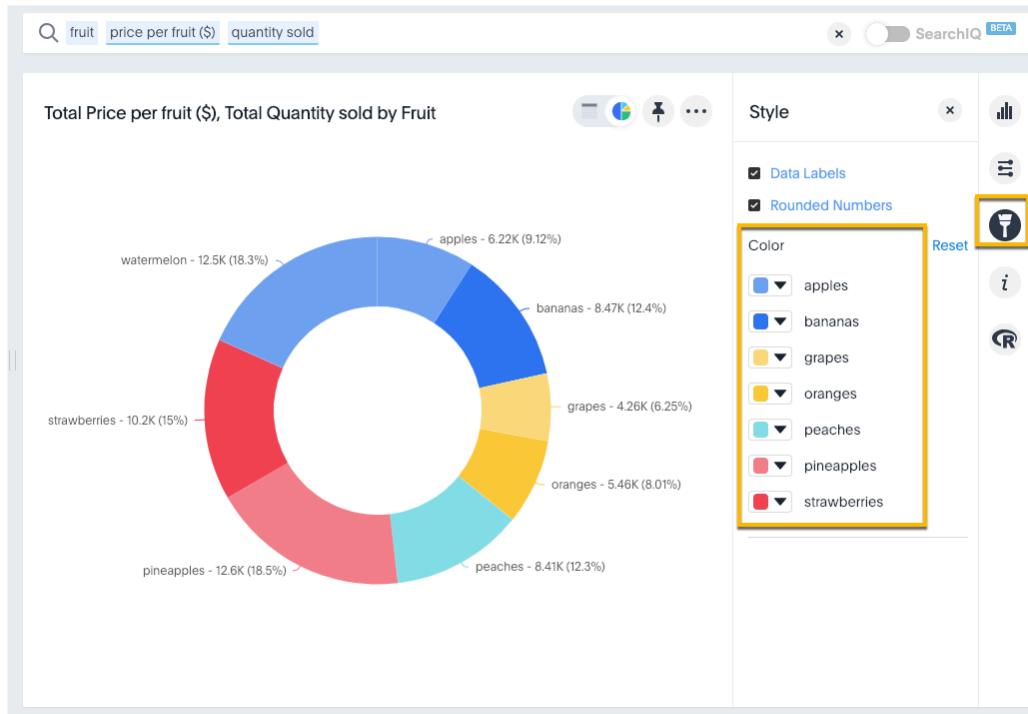
Your changes are saved immediately.

4. Click the **X** at top right to dismiss the styles panel.

Set pie chart slice colors

To customize the colors of each slice in a pie chart.

1. While viewing your search or answer as a pie chart, click **Edit chart styles** () to open the Styles panel.
2. Assign a color of your choice to each pie slice.



1. Click  and choose **Update** to save the changes.

2. Click Close.

Optionally, you can click **Reset** to use the default colors.

Additional chart options

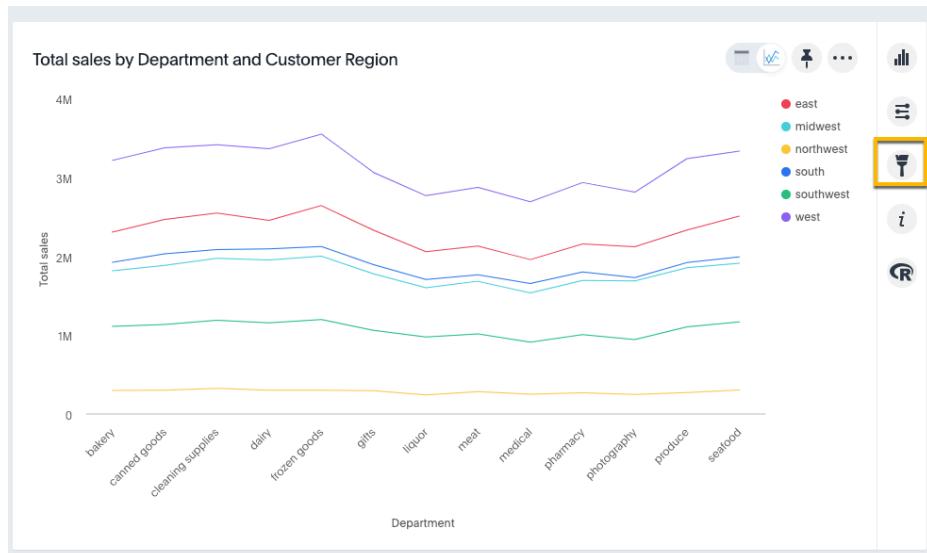
Summary: On charts, you can configure data labels, grid lines, regression lines, legends, values for x or y axis, and lock or unlock.

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

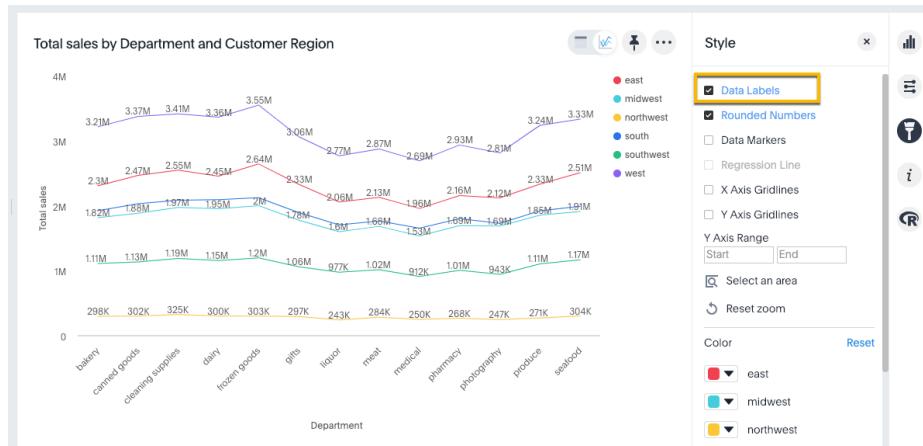
Show detail labels

To show detail labels:

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



2. Select **Detail Labels**.

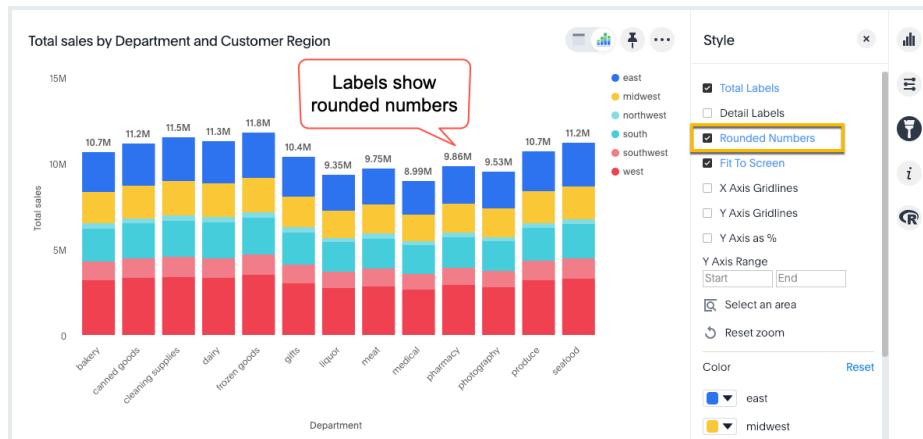


Show rounded or non-rounded numbers

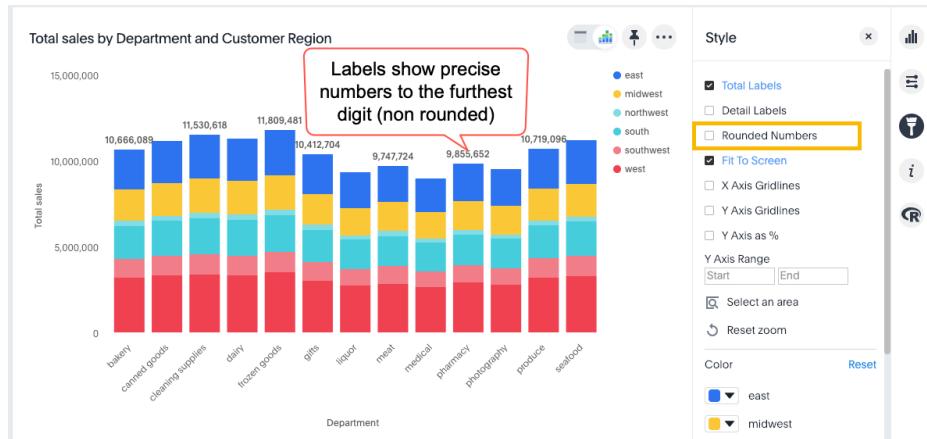
You can toggle this option off to show non-rounded numbers.

1. While viewing your search or answer as a chart, click **Edit chart styles** on the right.
2. Click **Rounded Numbers** to toggle on or off.

By default, charts with data labels enabled show rounded numbers on chart labels.



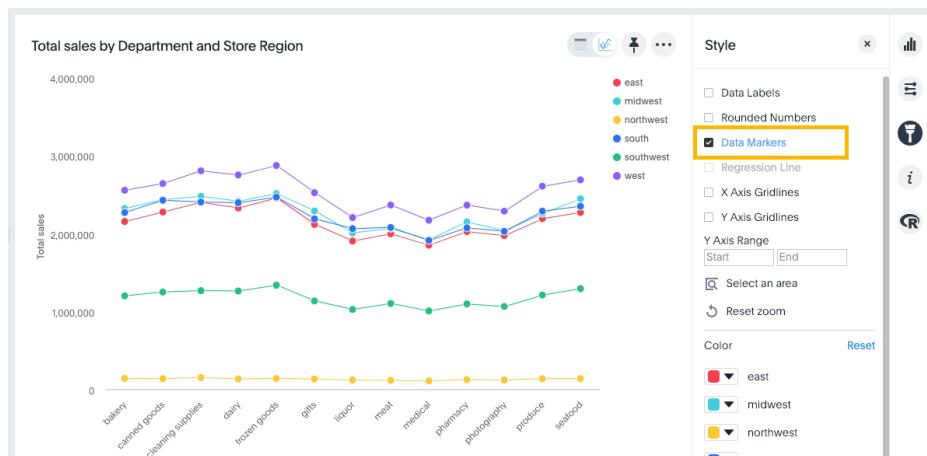
When you uncheck **Rounded Numbers**, data labels, X and Y axes, and tooltips show precise, non-rounded numbers out to the furthest decimal or integer, based on the underlying worksheet format.



Show data markers

To show data markers:

1. While viewing your search or answer as a chart, click **Edit chart styles** on the right.
2. Select **Data Markers**.



Add regression line

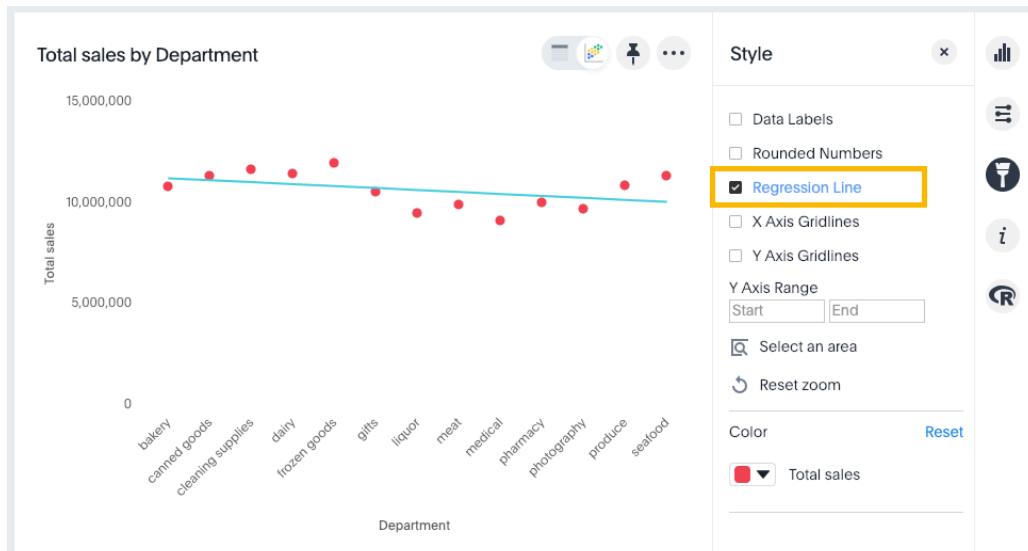
Note: To add a regression line, you need a search with only one attribute and one measure.

Otherwise, this option will show as disabled; i.e., greyed out on the styles panel. As soon as you modify the search to contain a single measure and attribute, the regression line option will be

clickable.

1. While viewing your answer as a chart, click **Edit chart styles**.
2. Select **Regression line**.

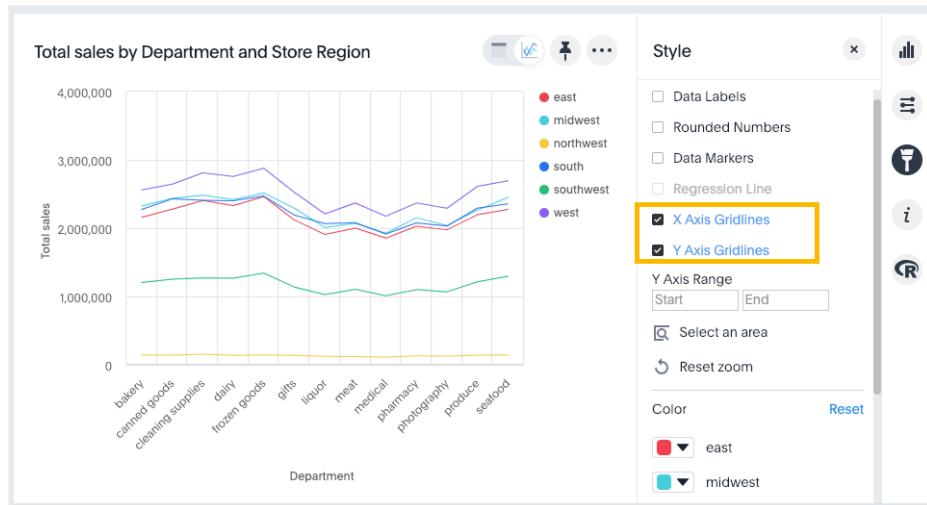
You can show regression lines on most types of charts (bar, stacked bar, line, bubble, and so on). In the following examples, `sales` is the measure and `department` is the attribute.



Display gridlines

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

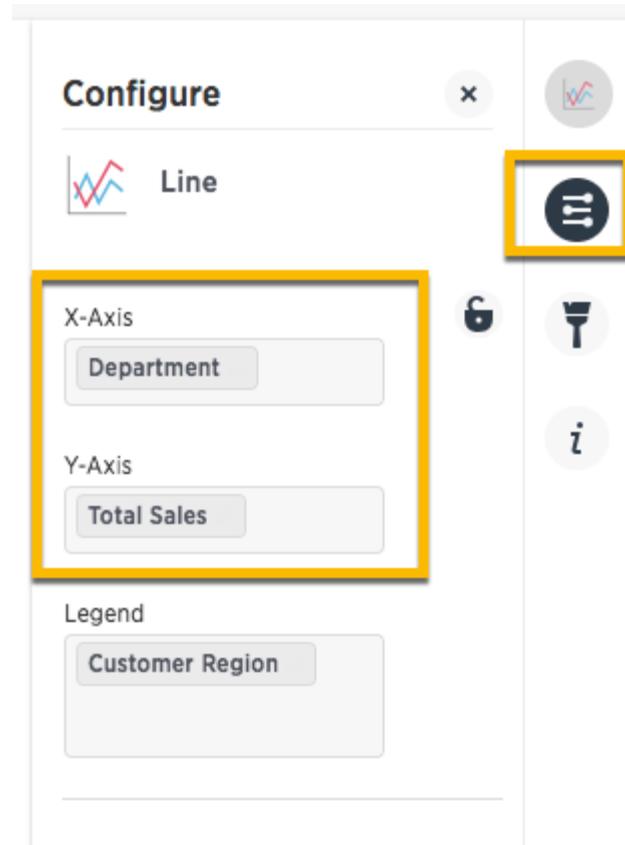
1. Choose **Edit chart styles**.
2. Select one or both of the gridline options.



Configure columns for X and Y axis

You can specify which fields to show on the X axis and Y axis of a chart.

1. Click **Edit chart configuration** on the right.
2. Click into the X or Y axis field and select the column used in the search that you want on a particular axis.

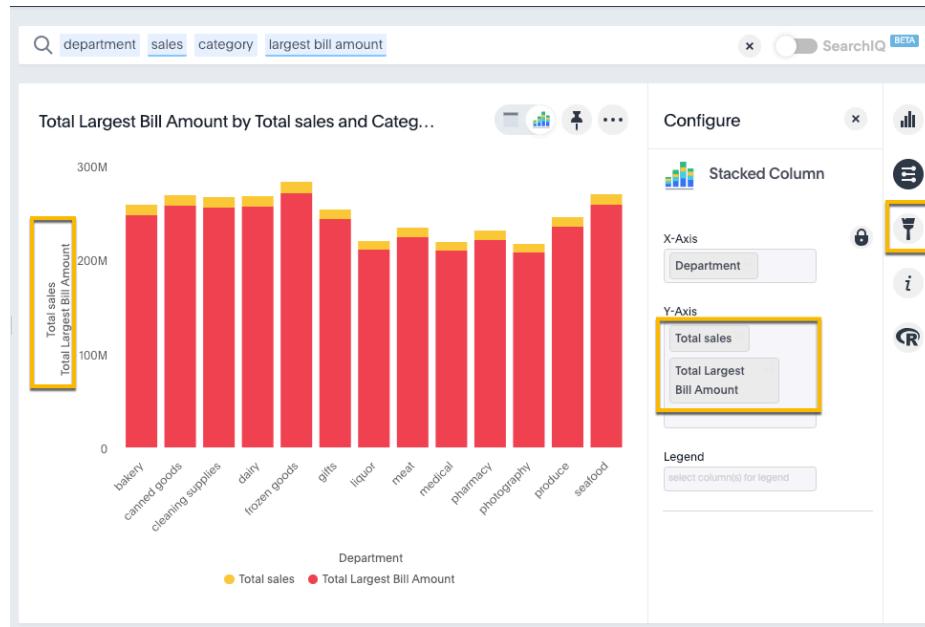


Stack multiple measures on the y-axis

You can have multiple measures on the y-axis of most (but not all) chart types, as follows:

1. Create a chart.
2. Click **Edit chart configuration** on the right.
3. Click into the **Y-Axis** field and start typing the name of a measure used in the search.
4. Select the measure you want.

The chart updates to show multiple measures on the Y-Axis.

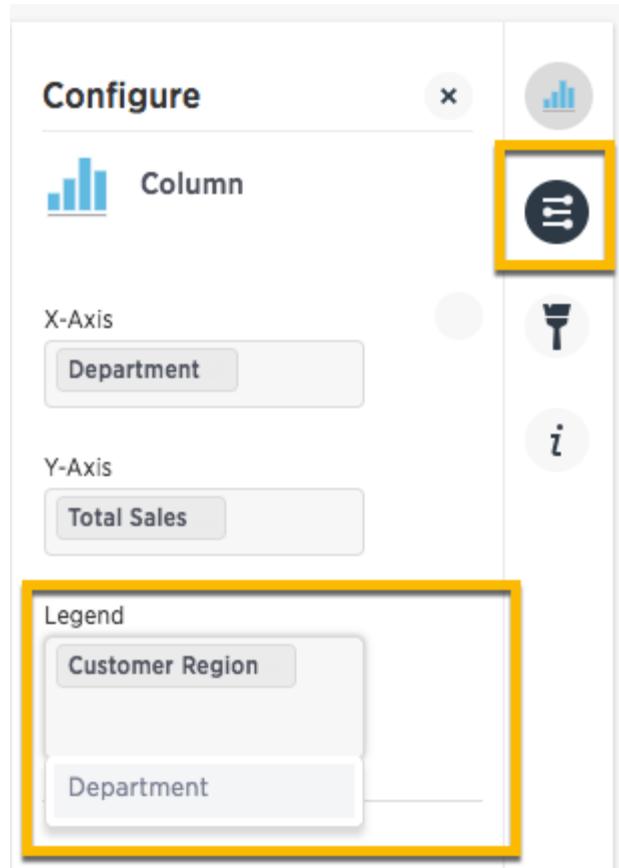


- If your original chart had a legend, you will get a message stating “*When there are multiple columns on the y-axis, you cannot add a legend. <Measure> is already in use.*” If so, remove the legend and the chart will update to show multiple measures on the y-axis.
- In some cases, the y-axis measures will show one on either side of the chart (left and right) instead of stacked on the same side, depending on the parameters of the analysis.
- For a list of chart types to which this applies, see [Charts with multiple measures on the y-axis](#).

Add columns to a chart legend

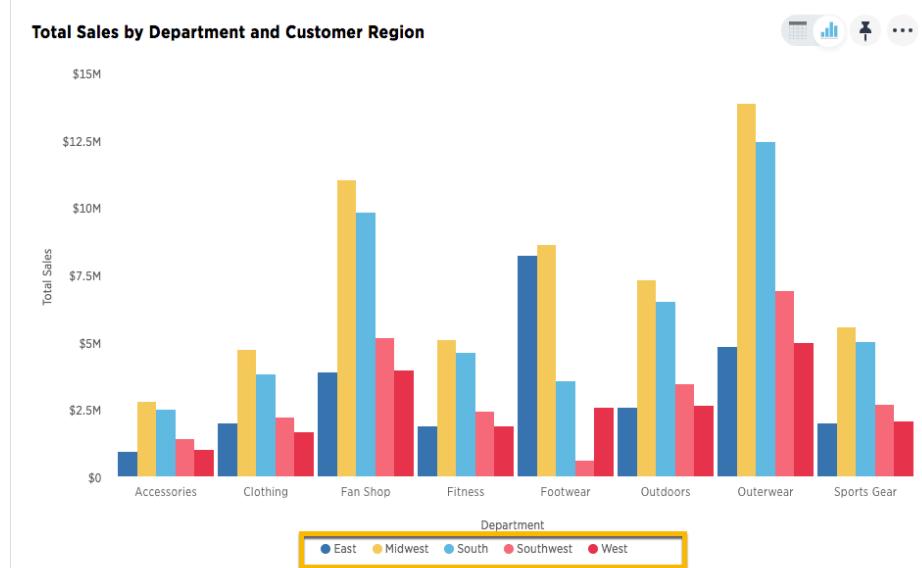
You can add to or edit the legend from the chart configuration panel.

1. Click **Edit chart configuration** on the right.
2. Click into the legend field to add one or more columns used in the search.



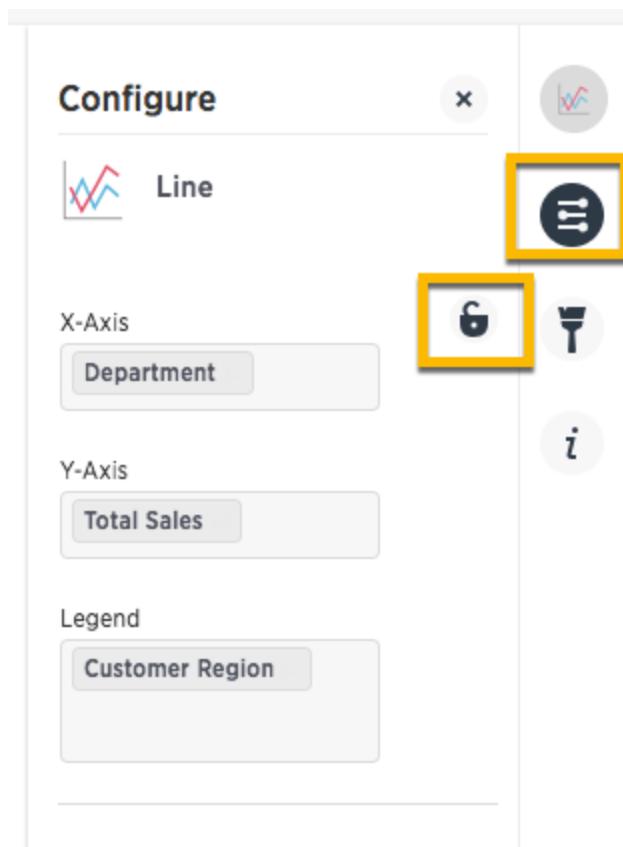
Your changes are saved automatically.

3. View the updated chart.



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.



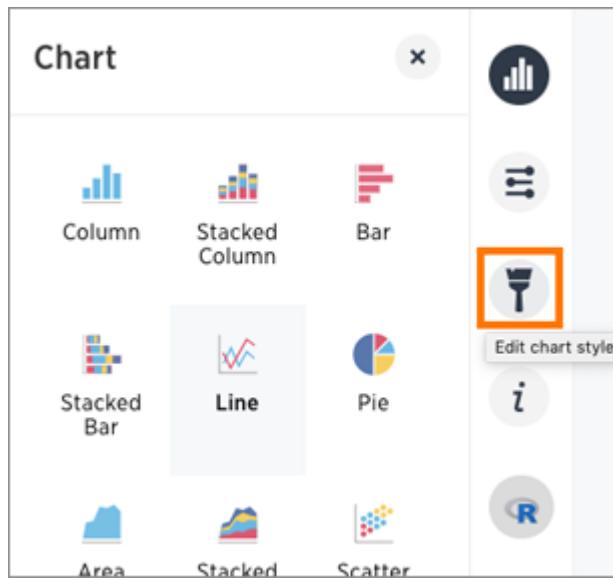
Related information

[Change chart colors](#)

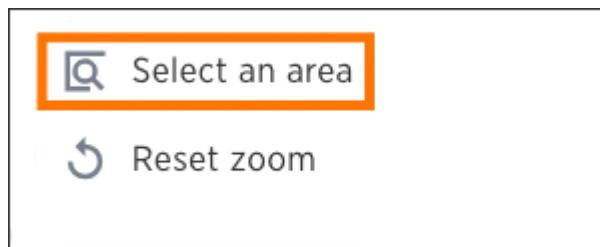
Zoom into a chart

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

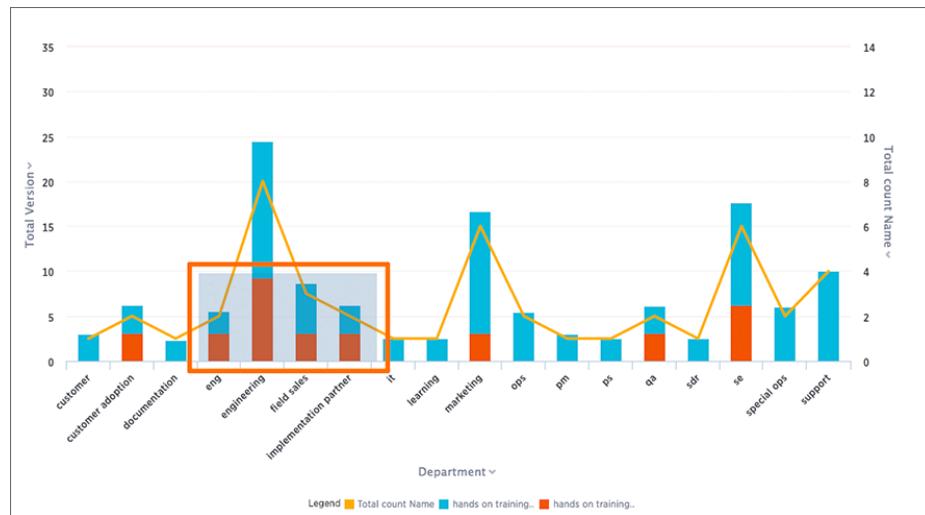
1. While viewing your answer as a chart, click **Edit chart style** on the right.



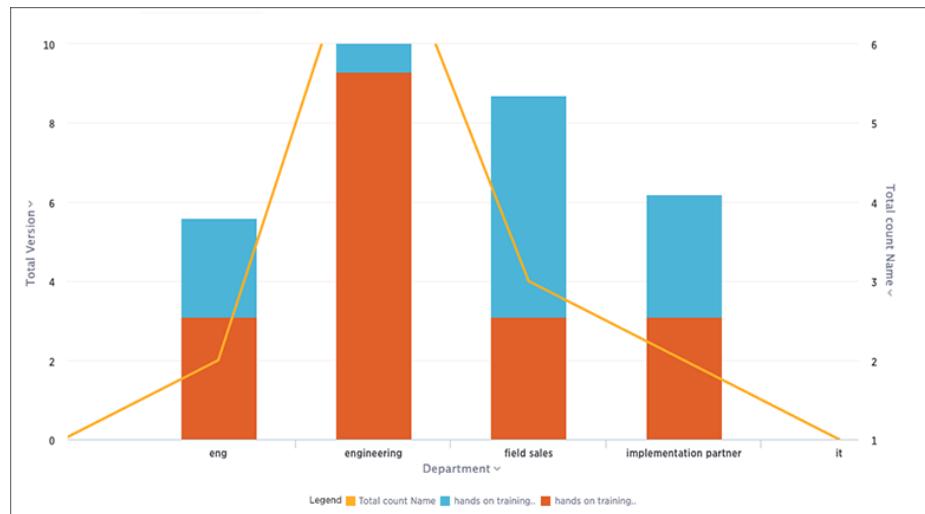
2. Click **Select an area**.



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.



- If you would like to return to the original chart view, click **Reset zoom**.



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.

Text formulas are also available. These are covered in the comprehensive [Formula function reference](#), which provides brief descriptions and examples for all types of formulas. The above topics explain concepts and give step-by-step instructions of how to work with formulas, whereas the reference is a quick cheat sheet.

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.

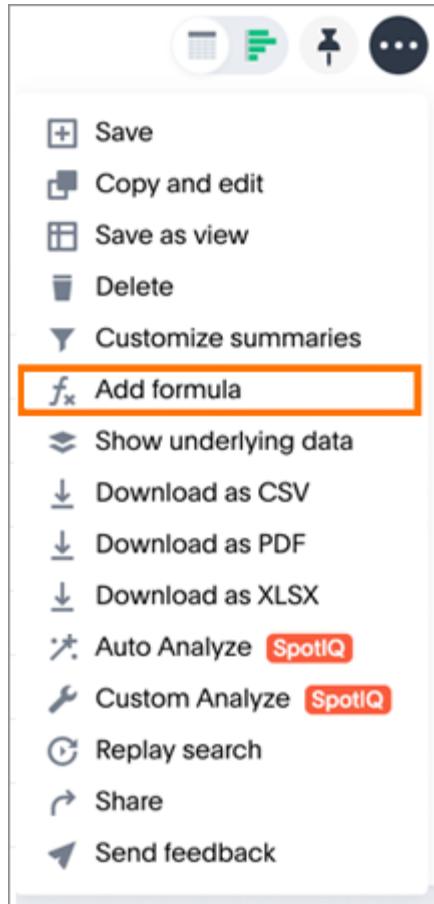
How to add a formula

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 **Table View**.



3. In the upper-right side of the table, click the three-dot **Actions** icon and select **Add formula**.



4. Name and enter your formula in the Formula Builder.

Department revenue per month

average monthly sales /departmental share |

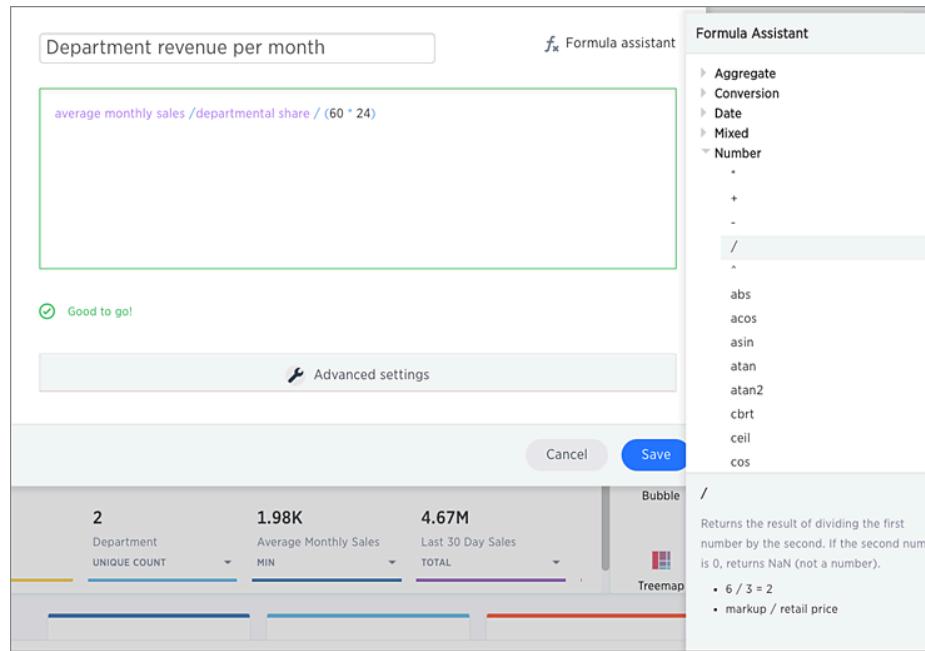
Good to go!

Advanced settings

Cancel Save

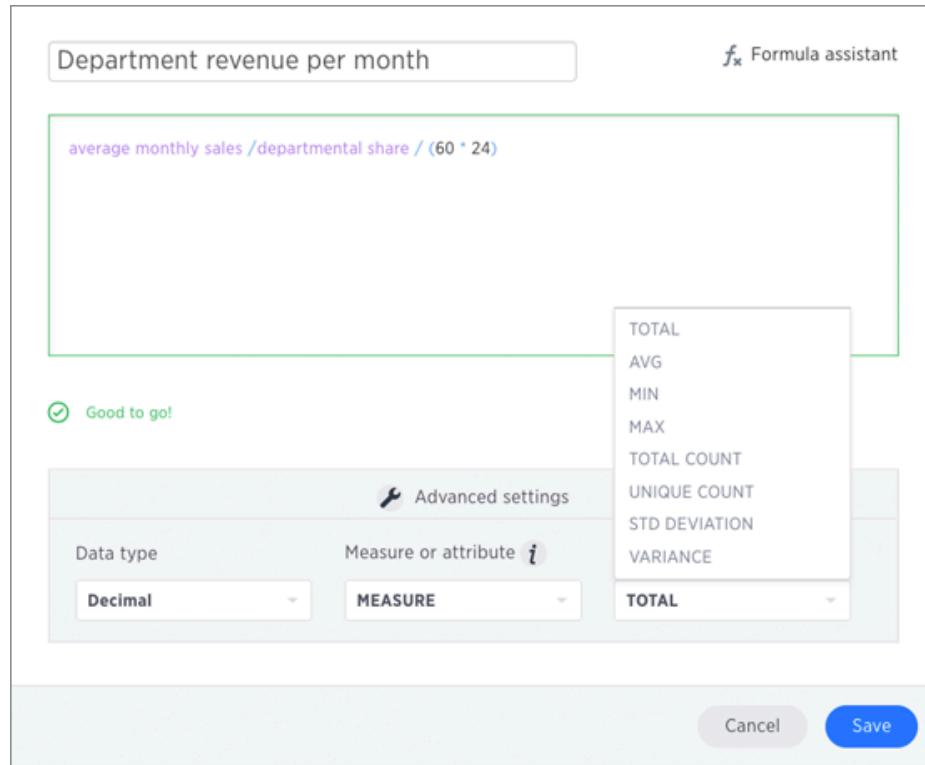
Note: Formulas elements are color coded by type and can include the formula operators and functions (blue), the names of columns (purple), and/or constants (black).

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



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
- Measure or attribute
- Aggregation



7. Name the formula by clicking 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, click the three-dot **Actions** icon and select **Edit**.
2. Scroll down to the bottom of the **Columns** listing. You will see a data source called **Formulas**.
3. Expand **Formulas**, and you can see a list of all the formulas in this answer.

Product Name	Department	Product Sales Group Sum
Under Armour Sunglasses	Accessories	17,419.94
North Face Hat	Accessories	22,247.56
Superfeet Athlete Insoles	Accessories	18,967.80
Sole Flat Athletic	Accessories	23,072.38

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

5. View or edit your formula in the Formula Builder.

Percent of Department Sales f_x Formula assistant

```
sum(sales)/group_sum ( sales , department ) *100
sum
customer name Sporting Goods Retail Worksheet
age group Sporting Goods Retail Worksheet
income bracket Sporting Goods Retail Worksheet
customer age group Sporting Goods Retail Worksheet
```

✓ Good to go!

Advanced settings Save Cancel

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).

6. If you edited your formula, click **Save** to save your changes. Otherwise, click **Cancel**.

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

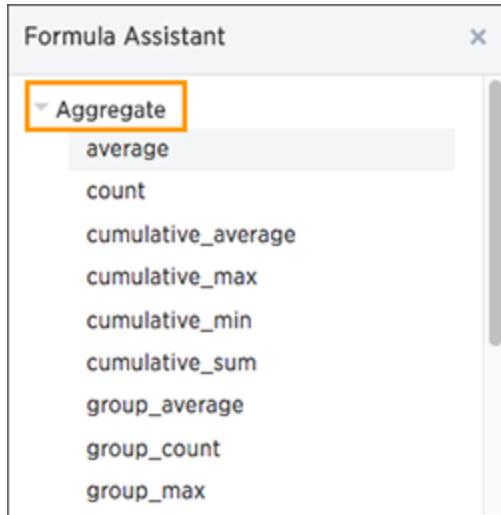
Formulas can be broken down into two types standard and aggregation formulas. Standard formulas act on individual rows and returns one result per row. Standard formulas use functions like:

- `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.

Flexible aggregation

The `group_aggregate` function makes it possible to aggregate measures at granularities that are different from the dimensions or column groupings used in the search.

See [Flexible aggregation](#) to learn more about working with this formula.

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 () ::
	TOTAL
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 () ::
	TOTAL	
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
cumulative_average	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>
cumulative_max	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>
cumulative_min	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>
cumulative_sum	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:

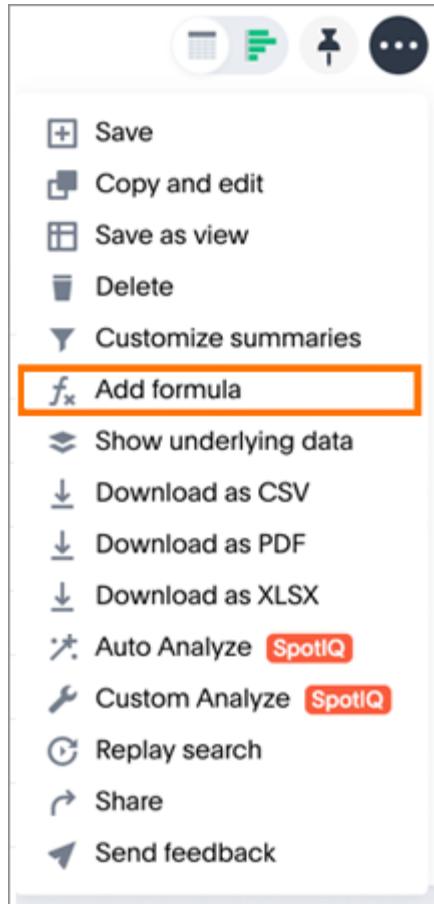
1. Start a new search.

The screenshot shows the ThoughtSpot Data interface. On the left, there's a sidebar with 'Data' at the top, followed by 'Choose Sources' and a 'Search Columns' input field containing 'Sporting Goods Retail Worksheet'. Below these are two collapsed sections: 'Sporting Goods Retail Worksheet' and 'Formulas'. The main area is titled 'Monthly Department Sales Analysis' and displays a table of sales data. At the top of the table, it says 'Department: fan shop, sports gear'. The table has four columns: 'Department', 'Average Monthly Sales', 'Last 30 Day Sales', and 'Departmental Share'. The data rows are as follows:

Department	Average Monthly Sales	Last 30 Day Sales	Departmental Share
op	944,665.99	1,168,588.73	18.73
Gear	489,659.92	615,604.43	9.71
Gear	387,817.03	401,857.21	9.66
op	760,579.94	782,611.74	18.95
Gear	213,930.58	224,205.79	9.25

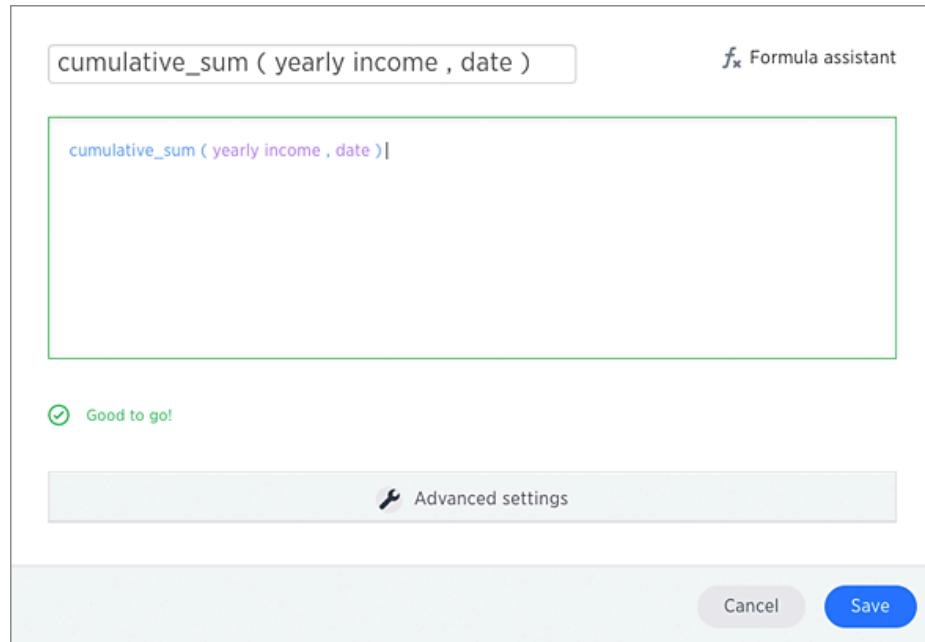
Below the table, it says '(showing rows 1-7 of 12)'. At the bottom of the interface, there are summary statistics: '2012 - 2017', '2 Department UNIQUE COUNT', '1.98K Average Monthly Sales MIN', and '4.64M Last 30 Day Sale TOTAL'. On the right side of the table, there are several icons: a magnifying glass, a line graph, a bar chart, a funnel, a person icon, and three dots. The three-dot icon is highlighted with a red box.

2. In the upper-right side of the table, click the ellipses icon 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 its title and typing a new name.

5. Click **Save**.

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

<input type="button" value="≡"/> 1 <input type="text" value="yearly average monthly sales last 30 day sales departmental share fan shop sports gear"/> <input type="text" value="cumulative_sum (yearly income , date)"/> <input type="button" value="sort by average monthly sales descending"/>				
Monthly Department Sales Analysis				
<input type="button" value="▼"/> Department: fan shop, sports gear <input type="button" value="▼"/>				
Yearly (Transaction Date)	Average Monthly Sales ↓	Last 30 Day Sales	Departmental Share	cumulative_sum (yearly income , date)
2017	1,434,325.91	6,550,600.24	0.02	51,936,503,409
2017	1,434,325.91	6,550,600.24	0.11	34,593,091,271
2017	1,434,325.91	6,550,600.24	0.10	68,997,569,131
2017	1,434,325.91	6,550,600.24	0.02	63,705,670,108
2017	1,434,325.91	6,550,600.24	0.05	58,506,339,272
2012 - 2017	6.21K	5.95B	172	16.8M
Yearly (Transaction Date)	Average Monthly Sales MIN	Last 30 Day Sales TOTAL	Departmental Share TOTAL	cumulative_sum MIN

A headline box displaying the cumulative sum within the entire table will appear at the bottom.

You can click it to toggle between different aggregations.

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:

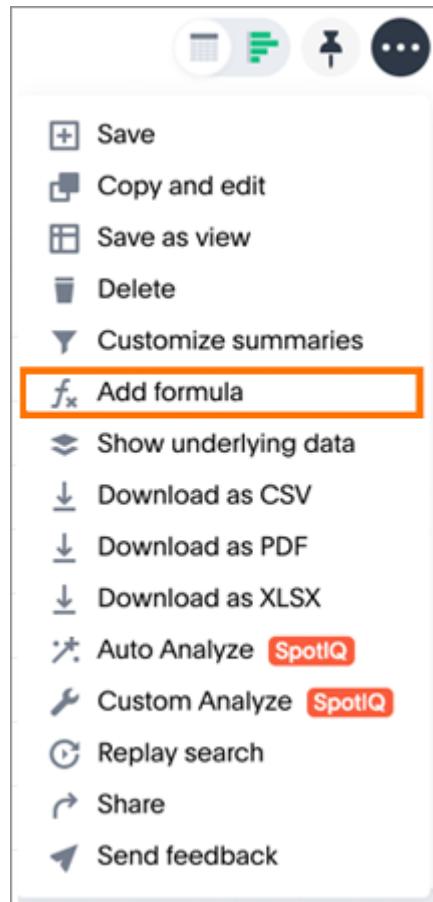
1. Start a new search.

The screenshot shows the ThoughtSpot Data interface. On the left, there's a sidebar with 'Data' at the top, followed by 'Choose Sources' and a search bar labeled 'Search Columns'. Below that are sections for 'Sporting Goods Retail Worksheet' and 'Formulas'. The main area is titled 'Monthly Department Sales Analysis' and displays a table with the following data:

Department	Average Monthly Sales	Last 30 Day Sales	Departmental Share
op	944,665.99	1,168,588.73	18.73
Gear	489,659.92	615,604.43	9.71
Gear	387,817.03	401,857.21	9.66
op	760,579.94	782,611.74	18.95
Gear	213,930.58	224,205.79	9.25

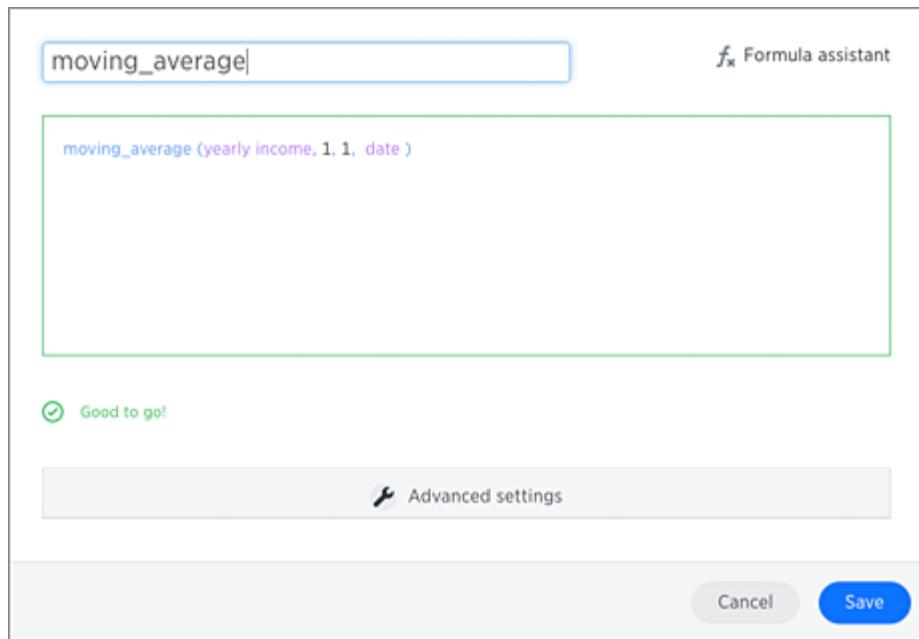
Below the table, it says '(showing rows 1-7 of 12)'. At the bottom, there are summary statistics: '2012 - 2017', '2', '1.98K', '4.64M', 'Yearly (Transaction Date)', 'Department UNIQUE COUNT', 'Average Monthly Sales MIN', 'Last 30 Day Sale TOTAL'. To the right of the table are various visualization icons.

2. In the upper-right side of the table, click the ellipses icon 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 entering a title in the top field, and then click **Save**.

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

The screenshot shows a ThoughtSpot analysis titled "Monthly Department Sales Analysis". At the top, there are search filters: "yearly", "department", "average monthly sales", "last 30 day sales", "sort by transaction date yearly descending", "departmental share", "fan shop sports gear", and "moving_average". Below the filters is a table titled "Monthly Department Sales Analysis" with the following data:

tm..	Average Monthly Sales	Last 30 Day Sales	Departmental Share	moving_average
: Gear	489,659.92	615,604.43	0.06	181,295,761.33
: Gear	489,659.92	615,604.43	0.02	151,274,473.67
: Gear	489,659.92	615,604.43	0.01	197,056,070.33
top	944,665.99	1,168,588.73	0.01	153,414,630.33
: Gear	489,659.92	615,604.43	0.01	99,732,730.00

(showing rows 1-7 of 1000+)

At the bottom, there are summary statistics and aggregation controls:

2 Department UNIQUE COUNT	1.98K Average Monthly Sales MIN	1.67B Last 30 Day Sales TOTAL	172 Departmental Share TOTAL	2M moving_average MIN
--	--	--	---	------------------------------------

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

5. To use a different aggregation type, click the current aggregation type in the bottom of the box and select another type.

Flexible aggregation functions (group aggregate and filters)

You can use the `group_aggregate` function to aggregate measures at granularities that are different from the dimensions that you have in columns used in the search.

How aggregation formulas work

Typically, the groupings and filters used in a formula will be the same as those of the columns used in the search. The concept of a grouping equates to an attribute column.

For example, in the search “revenue ship mode”, revenue is the measure, and ship mode is the attribute or *grouping*. The result of this search would show total revenue for each ship mode, e.g., a dollars for air, r dollars for rail, t dollars for truck, s dollars for sea transport.

The aggregation formulas are described in [Overview of aggregate formulas](#).

About flexible aggregation

Starting with version 5.0, ThoughtSpot provides for more flexible aggregation capability with a new function called `group_aggregate`. You can use this formula to pin columns in a query at a granularity different from the search bar query, using custom groupings and filters, rather than being bound to those of the search terms/columns.

The formula uses a sub-query to perform the custom aggregation. If the sub-query is at a coarser grain, result column is simply added to the result of original query. Roll-up or *reaggregation* is used when the sub-query is at a finer grain than the original query

This is particularly useful for comparison analysis.

You can specify to use the groups and filters from the query with `query_groups` and `query_filters`, respectively, and for `query_groups` you can also add or exclude some groups or filters.

You can use roll-up or reaggregation to fill in a column.

Examples

For a search on `revenue monthly ship mode`, you can add a formula to calculate yearly revenue by ship mode:

```
group_aggregate(sum(revenue), {ship mode, year(commit date)}, {})
```

The same formula can also be written using `query_groups()` and `query_filters()` as following:

```
group_aggregate(sum(revenue), query_groups() - {commit date} + {year(commit date)}, {})
```

This is helpful to include the main query groups that are not known at formula creation time. You can use `+/-` to modify the set of groups included from the query.

• Note: `+/-` is currently supported only for `query_groups`, not `query_filters`.

When group formula results are finer-grained than the search

With the flexibility of groupings for group formulas, the computed column created by a formula can be finer or coarser grained than the search itself.

For example, you can have a search that shows **total yearly sales** and a formula that computes total sales *for each month* (a finer-grained calculation than the search).

In such cases, if an additional aggregation is specified by the formula, the results get *reaggregated*.

Reaggregation can be applied in either of these ways:

- You can add an aggregation keyword just before a formula column in a search. For example, in this search we've added the keyword **min** just before our formula for `monthly_sales` :

```
sum revenue yearly min monthly_sales
```

where, the `monthly_sales` formula is written as:

```
group_aggregate(sum(revenue), {start_of_month(date)}, {})
```

- You can create a separate formula, such as in this search for:

```
sum revenue yearly min_monthly_sales
```

where, the `min_monthly_sales` formula is written as:

```
min(monthly_sales)
```

Alternatively, if no aggregation is specified, then the search query also inherits the formula groupings, as in this search:

```
sum revenue yearly monthly_sales
```

where, the original query is computed at a monthly grain instead of yearly.

Reaggregation scenarios

Some scenarios require aggregation on an already aggregated result.

For example, computing minimum monthly sales per ship mode, requires two aggregations:

- the first aggregation of **sum** to compute total monthly sales per ship mode.
- the second aggregation of **min** to compute minimum sale that happened for any given month for that ship mode.

An example of this is this search:

```
ship mode min monthly_sales
```

where the formula `monthly_sales` is written as:

```
group_aggregate(sum(revenue), query_groups() + {start_of_month(date)}, {})
```

Groups and filters

Flexible group aggregate formulas allow for flexibility in both [groupings](#) and [filters](#). The formulas give you the ability to specify only groupings or only filters.

Related information

- For more examples of flexible aggregation, see the `group_aggregate` function under “Aggregate functions” in the [Formula function reference](#).
- To learn about aggregation formulas in general, see [Overview of aggregate formulas](#) and [Group aggregation functions](#)

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_average (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>

Function	Description
group_max	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>
group_min	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>
group_stddev	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>
group_sum	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>
group_unique_count	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>
group_variance	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>

Flexible aggregation

The `group_aggregate` function gives you more control over aggregation and filtering.

See [Flexible aggregation](#) to learn more about specifying `query_groups` with this formula.

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 following table 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 > 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 Data interface. On the left, the 'Data' sidebar lists various columns: Net Margin, Number of Children, POS Transaction Number, Product Name, Quantity, Sales, Sales Per Customer, Share of Total Yearly Sales, Store City, Store County, Store Membership, Store Name, Store Region, Store State, Store Zip Code, Tender Type, Total Square Footage, Transaction Date, Formulas (Departmental Share, Last 30 Day Sales, Monthly Sales), by_department_filter, and moving_average. The 'by_department_filter' formula is selected, displaying the code: `sum_if (department = "sports gear", average monthly sales)`. A green box highlights the status message 'Good to go!' and the 'Save' button. The main area shows a table with two rows for 'ports Gear'. The table has columns: Department, Average Monthly Sales, Last 30 Day Sales, and Departmental Share. The first row has values: 201,775.80, 189,466.95, 9.16, and 2,421,309 respectively. The second row has values: 154,324.10, 121,524.52, 11.12, and 1,851,889 respectively. At the bottom, there's a summary row: 2012 - 2017, 2, 1.98K, 4.64M, and 172.

After 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 ThoughtSpot search interface with the query "store state vs store region sales". The results table is titled "Sales by Region, State and Year". It contains six rows of data:

Store Region	Store State	Sales(Store Region)	Sales(Store State)
West	Nevada	40,377,852.80	1,882,479.41
South	Kentucky	32,952,660.21	3,653,372.97
South	Tennessee	32,952,660.21	4,262,718.89
East	New Jersey	23,880,858.49	2,999,747.33
East	Massachusetts	23,880,858.49	3,824,734.66
Midwest	Wisconsin	36,894,048.68	3,031,827.40

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 ThoughtSpot search interface with the query "sales store state vs all yearly last 3 years". The results table is titled "Sales by Region, State and Year". It includes a filter for "Sale Date: >= 01/01/2015 < 01/01/2018". The table has four columns:

Store State	Yearly (Sale Date)	Sales(Store State)	Sales(all)
California	2015	\$4,826,688.18	27,107,540.05
Maryland	2016	\$491,129.22	28,300,547.98
North Carolina	2015	\$784,390.15	27,107,540.05
Tennessee	2015	\$650,140.37	27,107,540.05
Indiana	2015	\$487,085.59	27,107,540.05
New Mexico	2015	\$244,294.59	27,107,540.05
Florida	2016	\$1,894,384.12	28,300,547.98
Florida	2017	\$3,290,168.05	48,817,098.25

You can also provide multiple `vs` instances:

Sales by Region, State and Year

Sale Date: >= 01/01/2015 < 01/01/2018

Store State	Yearly (Sale Date)	Sales(Store State)	Sales(all)	Sales(yearly)
California	2015	18,500,241.30	104,225,186.28	27,107,540.05
Maryland	2016	1,657,786.91	104,225,186.28	28,300,547.98
North Carolina	2015	3,123,537.86	104,225,186.28	27,107,540.05
Tennessee	2015	2,525,196.99	104,225,186.28	27,107,540.05
Indiana	2015	1,905,454.05	104,225,186.28	27,107,540.05
New Mexico	2015	1,030,784.77	104,225,186.28	27,107,540.05
Florida	2016	6,945,742.24	104,225,186.28	28,300,547.98

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

Sales by Age Group, Gender and Product Category

Age Group	Customer Age Group	Customer Gender	Department	Sales(Customer Gender)	Sales(Age Group)
b) 30-49	19 to 30 Years	Female	Footwear	3,818,947.06	\$383,214.47
b) 30-49	19 to 30 Years	Female	Sports Gear	2,467,901.41	\$272,740.05
c) 50-64	31 to 50 Years	Male	Outerwear	11,532,996.16	\$613,600.36
a) 18-29	19 to 30 Years	Male	Outerwear	10,324,246.26	\$9,179,745.54
b) 30-49	19 to 30 Years	Female	Outdoors	3,052,465.24	\$324,199.39

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
to_bool	Returns the input as a boolean (true or false).	<code>to_bool (0) = false</code> <code>to_bool (married)</code>
to_date	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>
to_double	Returns the input as a double.	<code>to_double ('3.14') = 3.14</code> <code>to_double (revenue * .01)</code>
to_integer	Returns the input as an integer.	<code>to_integer ('45') + 1 = 46</code> <code>to_integer (price + tax - cost)</code>
to_string	Returns the input as a text string. To convert a date to a string, specify the date format you want to use.	<code>to_string (45 + 1) = '46'</code> <code>to_string (revenue - cost)</code> <code>to_string (date, ('%m/%d/%y'))</code>

About date formulas

Date functions 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.

Date formulas

The date formulas include:

Function	Description	Examples
add_days	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>
add_minutes	Returns the result of adding the specified number of minutes to input date/date-time/time.	<code>add_minutes (01/30/2015 00:10:20 , 5) = 01/30/2015 00:11:20</code> <code>add_minutes (invoiced , 30)</code>
add_months	Returns the result of adding the specified number of months to the given date.	<code>add_months (01/30/2015, 5) = 06/30/2015</code> <code>add_months (invoiced_date , 5)</code>
add_seconds	Returns the result of adding the specified number of seconds to the given date.	<code>add_seconds (01/30/2015 00:00:00, 5) = 06/30/2015 00:00:05</code> <code>add_seconds (invoiced_date , 5)</code>
add_weeks	Returns the result of adding the specified number of weeks to the given date.	<code>add_weeks (01/30/2015, 2) = 02/13/2015</code> <code>add_weeks (invoiced_date , 2)</code>
add_years	Returns the result of adding the specified number of years to the given date.	<code>add_years (01/30/2015, 5) = 01/30/2020</code> <code>add_years (invoiced_date , 5)</code>
date	Returns the date portion of a given date.	<code>date (home visit)</code>

Function	Description	Examples
day	Returns the number (1-31) of the day for the given date.	day (01/15/2014) = 15 day (date ordered)
day_number_of_quarter	Returns the number of the day in a quarter for a given date. Add an optional second parameter to specify whether a 'fiscal' or 'calendar' year is used to calculate the result. Default is 'calendar'. (In examples, start of fiscal year is set to May 01.)	day_number_of_quarter (01/30/2015) = 30 day_number_of_quarter (01/30/2015, 'fiscal') = 91
day_number_of_week	Returns the number (1-7) of the day in a week for a given date with 1 being Monday and 7 being Sunday.	day_number_of_week(01/15/2014) = 3 day_number_of_week (shipped)
day_number_of_year	Returns the number (1-366) of the day in a year from a given date. Add an optional second parameter to specify whether a 'fiscal' or 'calendar' year is used to calculate the result. Default is 'calendar'. (In examples, start of fiscal year is set to May 01.)	day_number_of_year (01/30/2015) = 30 day_number_of_year (01/30/2015, 'fiscal') = 275 day_number_of_year (invoiced)
day_of_week	Returns the day of the week for the given date.	day_of_week (01/30/2015) = Friday day_of_week (serviced)
diff_days	Subtracts the second date from the first date and returns the result in number of days, rounded down if not exact.	diff_days (01/15/2014, 01/17/2014) = -2 diff_days (purchased, shipped)

Function	Description	Examples
diff_time	Subtracts the second date from the first date and returns the result in number of seconds.	diff_time (01/01/2014, 01/01/2014) = -86,400 diff_time (clicked, submitted)
hour_of_day	Returns the hour of the day for the given date.	hour_of_day (received)
is_weekend	Returns true if the given date falls on a Saturday or Sunday.	is_weekend (01/31/2015) = true is_weekend (emailed)
month	Returns the month from the given date.	month (01/15/2014) = January month (date ordered)
month_number	Returns the number (1-12) of the month from a given date. Add an optional second parameter to specify whether a 'fiscal' or 'calendar' year is used to calculate the result. Default is 'calendar'. (In examples, start of fiscal year is set to May 01.)	month_number (09/20/2014) = 9 month_number (09/20/2014, 'fiscal') = 5 month_number (purchased)
month_number_of_quarter	Returns the month (1-3) number for the given date in a quarter. Add an optional second parameter to specify whether a 'fiscal' or 'calendar' year is used to calculate the result. Default is 'calendar'. (In examples, start of fiscal year is set to May 01.)	month_number_of_quarter (02/20/2018) = 2 month_number_of_quarter (02/20/2018, 'fiscal') = 1
now	Returns the current timestamp.	now ()

Function	Description	Examples
quarter_number	Returns the number (1-4) of the quarter associated with the given date. Add an optional second parameter to specify 'fiscal' or 'calendar' dates. Default is 'calendar'. (In examples, start of fiscal year is set to May 01.)	<code>quarter_number (04/14/2014) = 2</code> <code>quarter_number (04/14/2014, 'fiscal') = 4</code> <code>quarter_number (shipped)</code>
start_of_month	Returns MMM yyyy for the first day of the month. Your installation configuration can override this setting so that it returns a different format such as MM/dd/yyyy . 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>
start_of_quarter	Returns the date for the first day of the quarter for the given date. Add an optional second parameter to specify whether a 'fiscal' or 'calendar' year is used to calculate the result. Default is 'calendar'. (In examples, start of fiscal year is set to May 01.)	<code>start_of_quarter (04/01/2014) = Apr 2014</code> <code>start_of_quarter (04/01/2014, 'fiscal') = Feb 2014</code> <code>start_of_quarter (sold)</code>
start_of_week	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
start_of_year	Returns the date for the first day of the year for the given date. Add an optional second parameter to specify whether a 'fiscal' or 'calendar' year is used to calculate the result. Default is 'calendar'. (In examples, start of fiscal year is set to May 01.)	start_of_year (04/01/2014) returns Jan 2014 start_of_year (04/01/2014, 'fiscal') returns May 2013 start_of_year (joined)
time	Returns the time portion of a given date.	time (3/1/2002 10:32) = 10:32 time (call began)
week_number_of_month	Returns the week number for the given date in a month.	week_number_of_month(03/23/2017) = 3
week_number_of_quarter	Returns the week number for the given date in a quarter. Add an optional second parameter to specify whether a 'fiscal' or 'calendar' year is used to calculate the result. Default is 'calendar'. (In examples, start of fiscal year is set to May 01.)	week_number_of_quarter (04/03/2017) = 1 week_number_of_quarter (04/03/2017, 'fiscal') = 10
week_number_of_year	Returns the week number for the given date in a year. Add an optional second parameter to specify whether a 'fiscal' or 'calendar' year is used to calculate the result. Default is 'calendar'. (In examples, start of fiscal year is set to May 01.)	week_number_of_year (01/17/2014) = 3 week_number_of_year (01/17/2014, 'fiscal') = 38

Function	Description	Examples
year	Returns the year from a given date. Add an optional second parameter to specify whether a 'fiscal' or 'calendar' year is used to calculate the result. Default is 'calendar'. (In examples, start of fiscal year is set to May 01. Per standard convention, the fiscal year is defined by the year-end date.)	year (01/15/2014) = 2014 year (12/15/2013, 'fiscal') = 2014 year (date ordered)

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
```

Fiscal and Gregorian calendars

For the following date formulas, you can further specify either `fiscal` or Gregorian `calendar` on which to base date calculations. (If you do not specify a calendar type, the formula defaults to standard Gregorian, with the year starting in January.)

- `day_number_of_quarter`
- `day_number_of_year`
- `month_number`
- `month_number_of_quarter`
- `quarter_number`
- `start_of_quarter`
- `start_of_year`
- `week_number_of_quarter`
- `week_number_of_year`
- `year`

Your ThoughtSpot administrator and ThoughtSpot Support can [set up a fiscal calendar year](#) to start on any month. If the `fiscal` year is not explicitly configured in the system, `fiscal` defaults to January, the same as the Gregorian `calendar`.

For example, the formula `month_number_of_quarter (05/01/2014)` would return `2` based on the default Gregorian calendar, whereas the formula `month_number_of_quarter (05/01/2014, 'fiscal')` would return `1` if your administrator has configured the fiscal calendar to start at May.

Related information

- [Date functions](#) in the [Formula function reference](#)
- [Set up a fiscal calendar year](#)

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

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.

Formula operators

The operators include:

Operator	Description	Examples
and	Returns true when both conditions are true, otherwise returns false.	<pre>(1 = 1) and (3 > 2) = true lastname = 'smith' and state ='texas'</pre> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;">Note: Not available for row level security (RLS) formulas.</div>
if...then...else	Conditional operator.	<pre>if (3 > 2) then 'bigger' else 'not bigger' if (cost > 500) then 'flag' else 'approve'</pre>
ifnull	Returns the first value if it is not null, otherwise returns the second.	<pre>ifnull (cost, 'unknown')</pre>
isnull	Returns true if the value is null.	<pre>isnull (phone)</pre>
not	Returns true if the condition is false, otherwise returns false.	<pre>not (3 > 2) = false not (state = 'texas')</pre>
or	Returns true when either condition is true, otherwise returns false.	<pre>(1 = 5) or (3 > 2) = true state = 'california' or state ='oregon'</pre>

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.

Related information

- [Operators](#) in the Formula function reference

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:



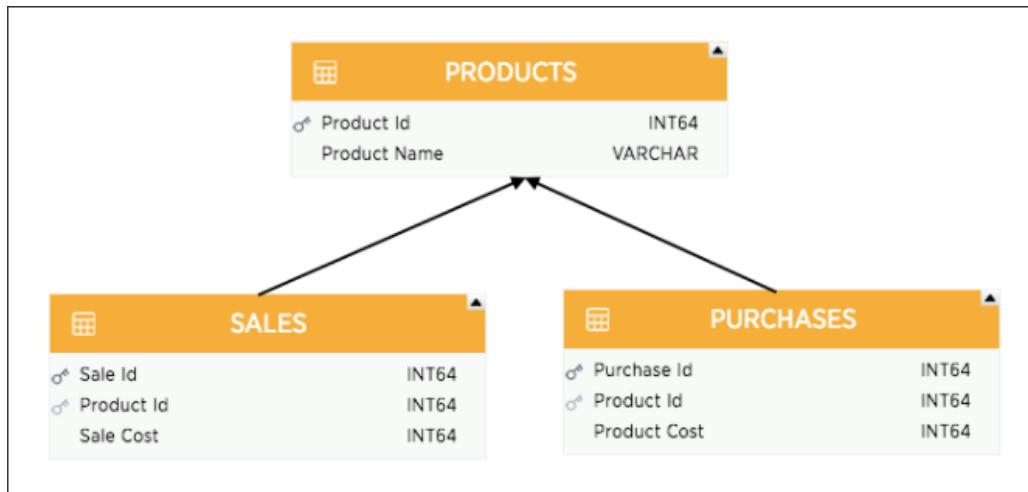
These formulas become columns, and are even suggested within the Formula Builder. There is no limit to the nesting depth you can create.

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.

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. They 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 drill down on the visualizations.

Create a pinboard

To create a pinboard:

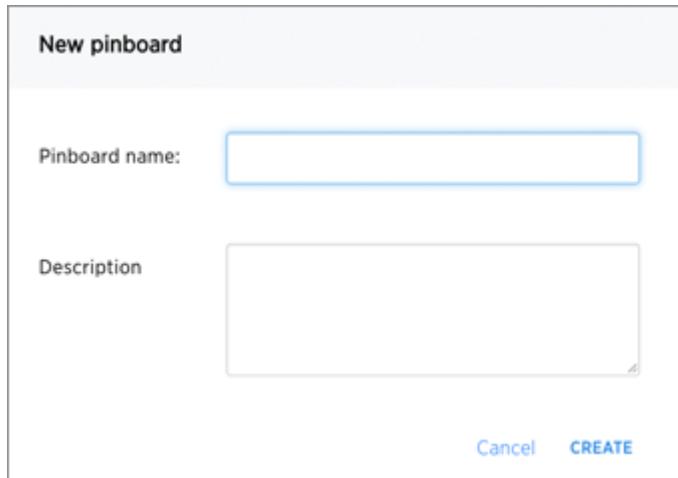
1. Click **Pinboards**, on the top navigation bar.



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

A screenshot of the Pinboards list page. The top navigation bar shows the Pinboards tab is active. Below the navigation is a toolbar with filters for All, Yours, Stickers (with a dropdown for 'Select sticker'), a search bar, and a page number '1 - 20'. A red box highlights the '+ Pinboard' button. The main area displays a table of pinboards with columns for Name, Stickers, Modified (sorted by date), and Author. Two pinboards are listed: 'Fan Shop Sales' and 'ThoughtSPORT Overview'. The 'Fan Shop Sales' pinboard was modified 1 week ago by an Administrator and has the 'Sports Goods' sticker applied.

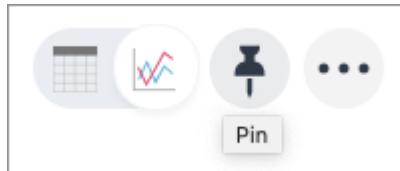
3. In the **New Pinboard** dialog box, give your pinboard a name and description. Then click **Create**.



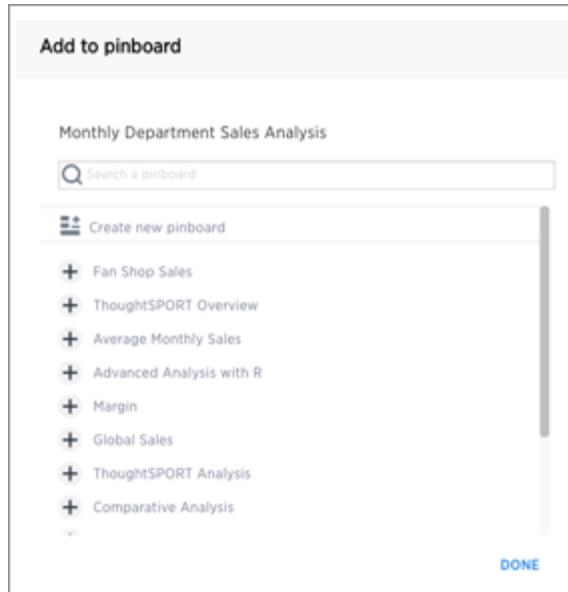
Add an answer to a pinboard

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

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



2. In the **Add to pinboard** 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: simply click the ellipses icon  and select **Download as PDF**.

```
! [](/images/pinboard-download-pdf.png "Download your pinboard")
```

For more details, see [Download a Pinboard 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

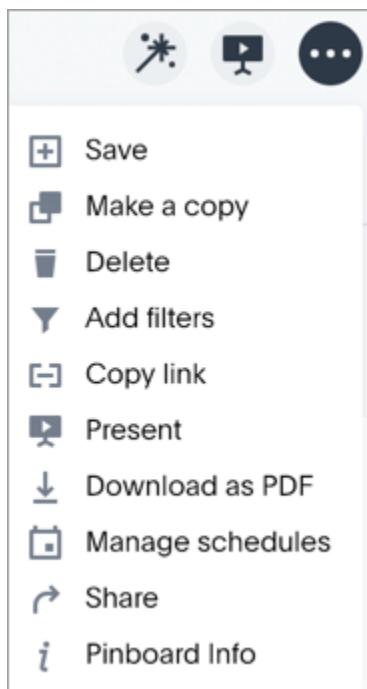
Created
Mar 5, 2018, 12:13 PM

Modified
Mar 16, 2018, 2:00 PM

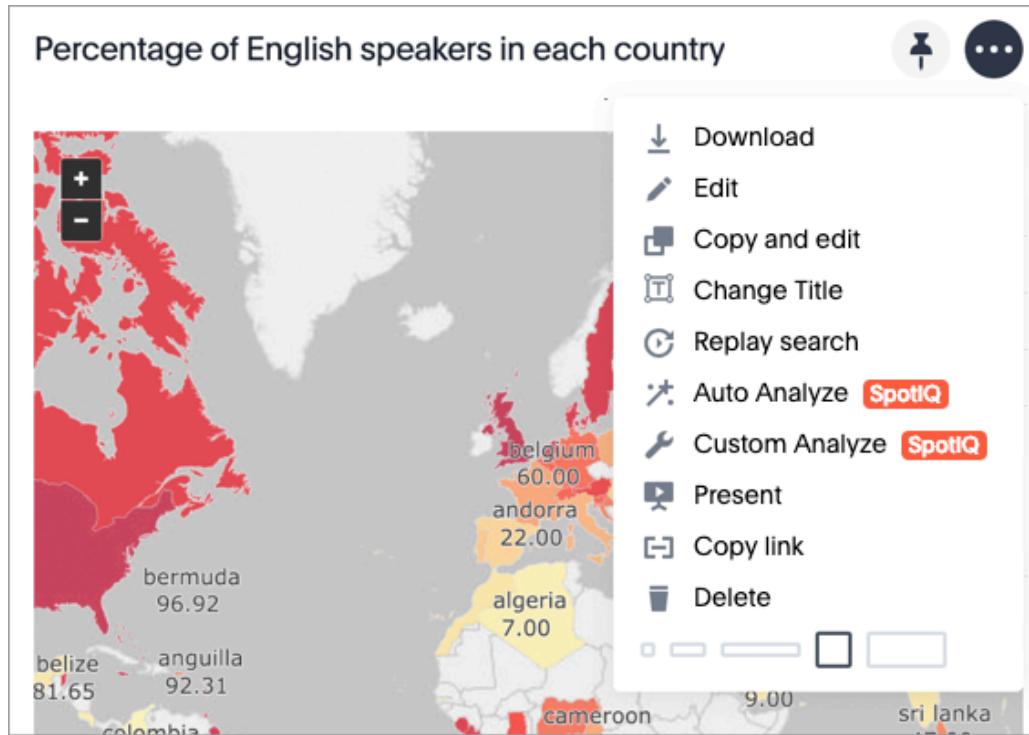
[CLOSE](#)

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 always appear, while others appear only when you hover 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 displays an alert message with a **Save changes** button at the top of the page. Click this button to make your changes permanent.

Related information

[Download a pinboard as PDF](#) [Present a pinboard](#)

Following pinboards

Summary: In ThoughtSpot, you can follow pinboards to regularly review the visuals that represent dynamic data. For example, you can follow a pinboard, and schedule weekly email notifications.

When you follow a pinboard, ThoughtSpot sends you regular emails that contain the snapshot of the pinboard. You can change the scheduling of this email to suit your business needs.

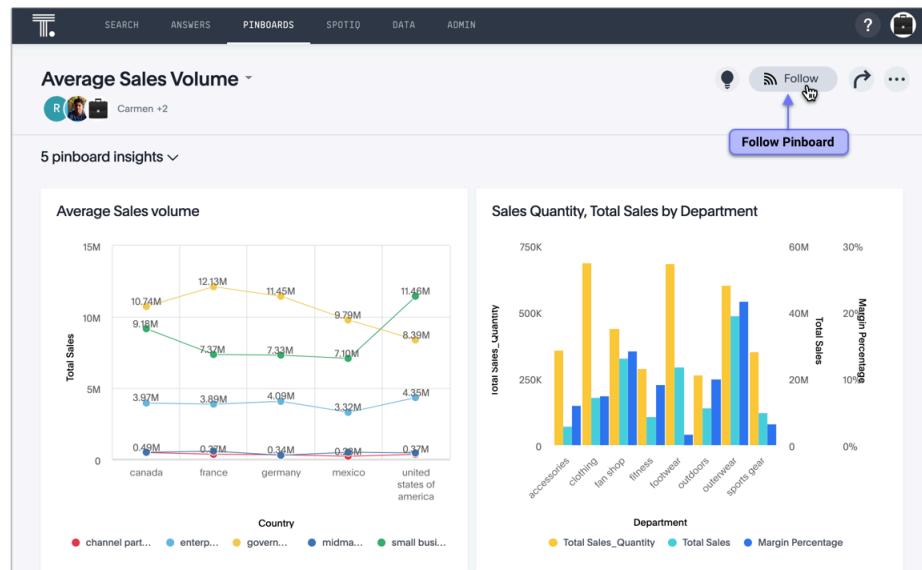
Follow Pinboards

Complete the following steps to follow a Pinboard:

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

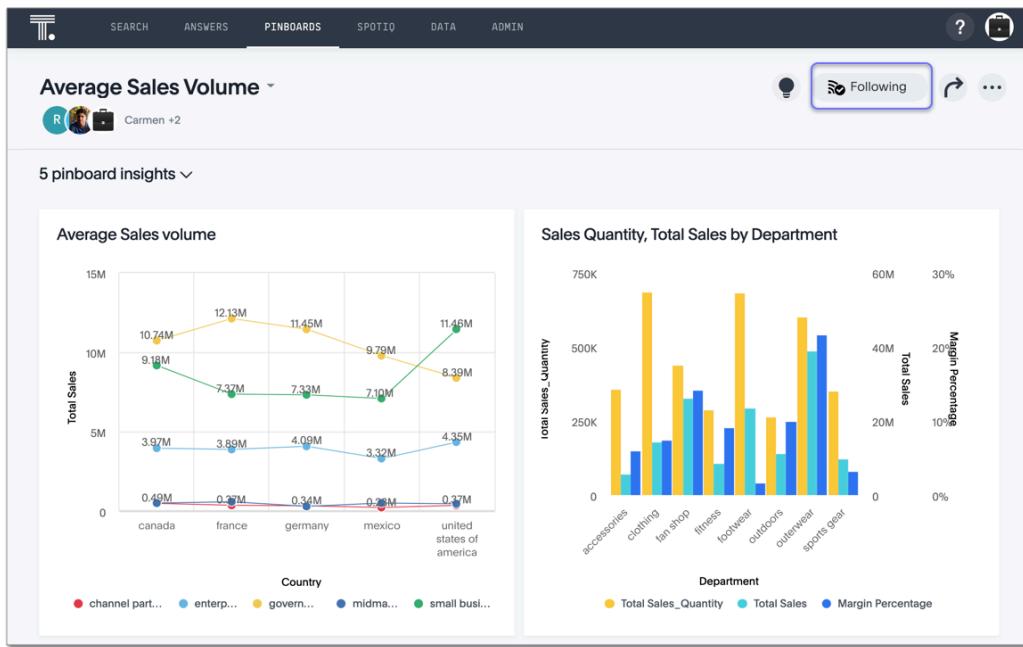


2. On the Pinboard list page, select a Pinboard by clicking on it.
3. Click **Follow** at the top right corner of the Pinboard.



This action opens the scheduler. See [Schedule Pinboard reports](#).

Notice that after you configure the schedule, the Pinboard appears as **Following**.



Schedule Pinboard reports

In the **Follow** scheduler, specify the email schedule by marking the appropriate options in the pulldown menus. The default schedule is weekly, on the day and time when you configure the schedule.

Weekly

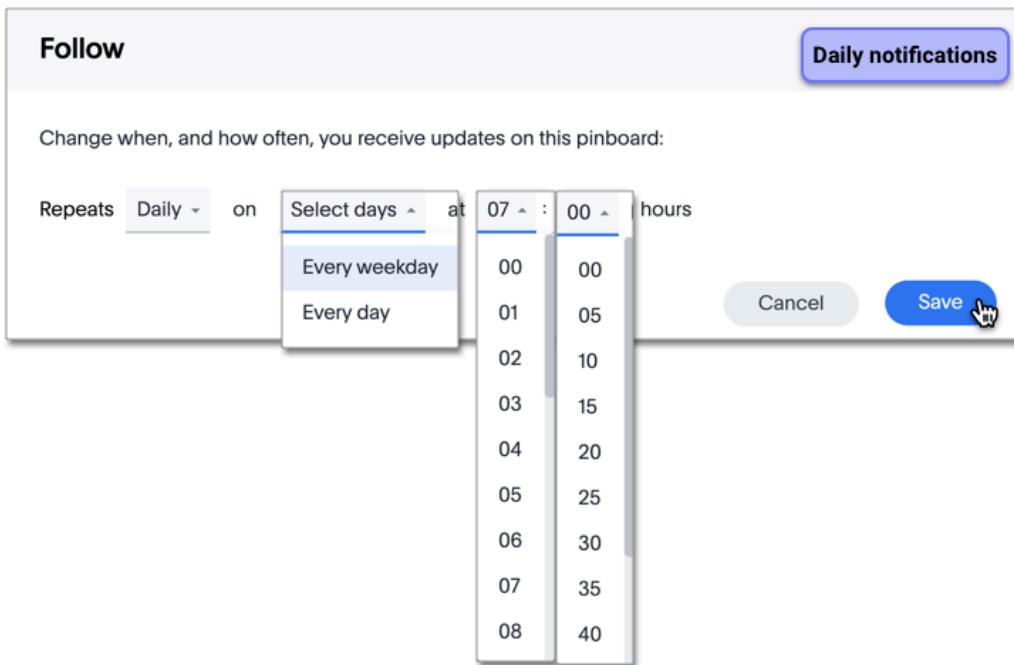
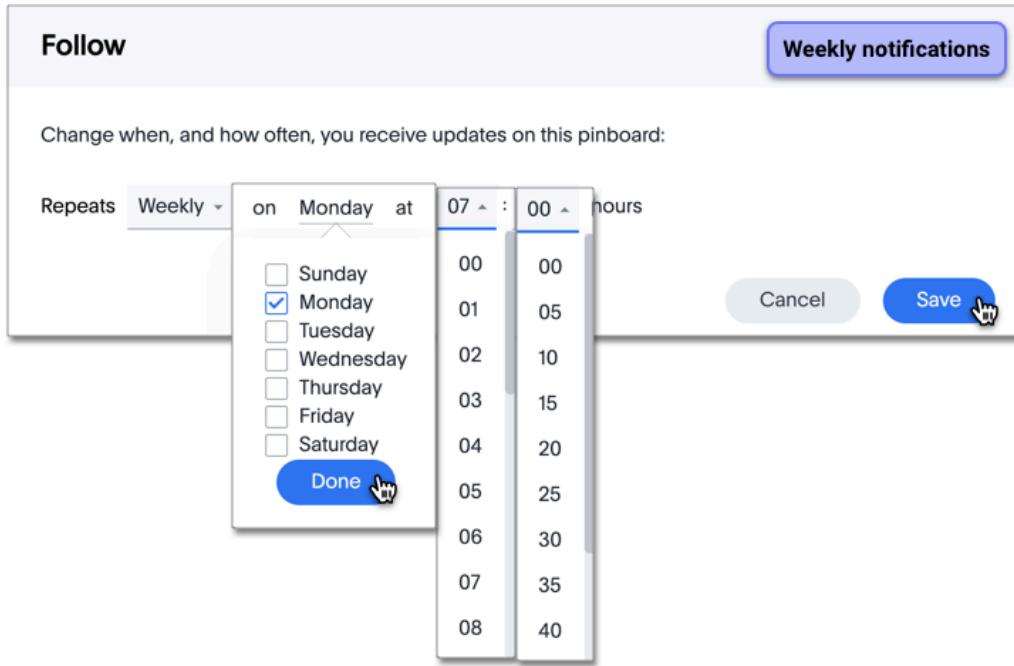
By default, the schedule is weekly, on the day of the week and at the time you create the schedule. You can select another day of the week, and then set the time.

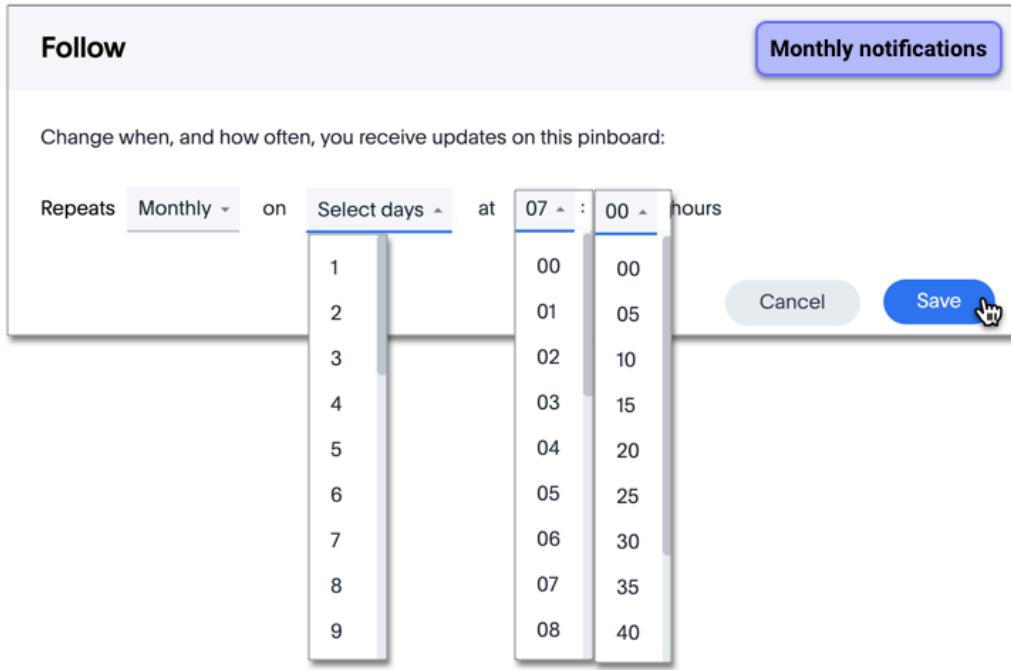
Daily

Specify *every weekday* (every working day), or *every day of the week*. Then set the time.

Monthly

Specify the day of the month, from 1 onwards. Then set the time.





Click **Save**.

If the user account is not configured with an email, you must supply one at this time.

Change the schedule

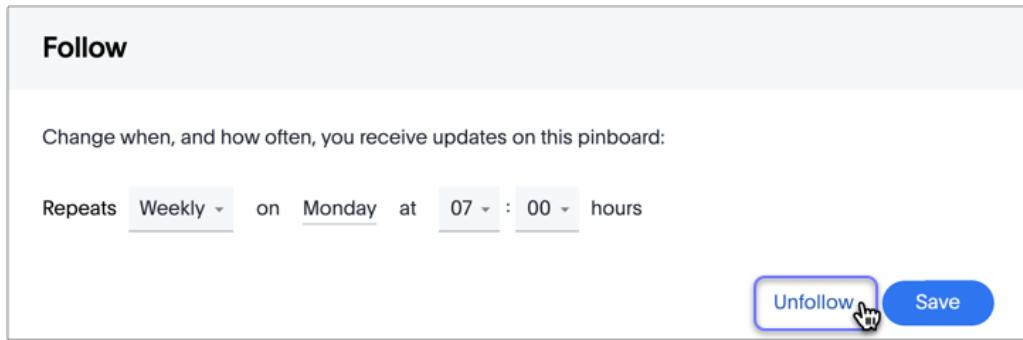
To change the existing time interval and the timing of notifications, click **Following** in the top right corner of the Pinboard.

When the scheduling interface appears, make the necessary changes, and click **Save**.

Unfollow Pinboard

To stop following a Pinboard, click **Following** in the top right corner.

When the scheduling interface appears, click **Unfollow**.



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.

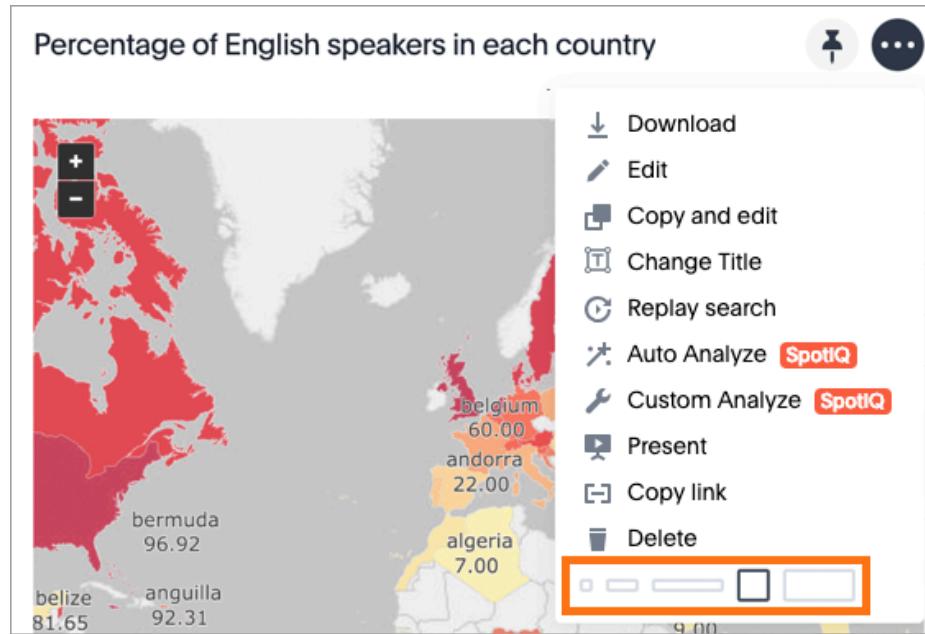
Reorder the visualization

In addition, the size picker at the bottom of a visualization's three-dot Actions menu 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.

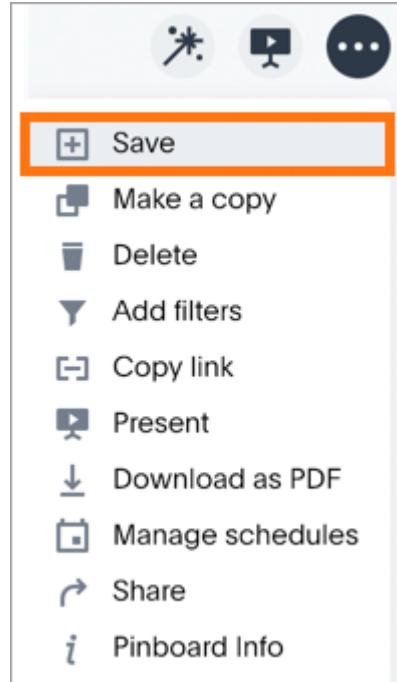
1. Click **Pinboards**, on the top navigation bar.



2. On the pinboards page, click the pinboard you would like to edit.
3. Drag and drop your visualizations on the layout grid to reorder your pinboard. Resize a visualization by choosing a predetermined size under the ellipses icon for the visualization.



4. Save your pinboard by clicking the ellipses icon  and selecting **Save**.

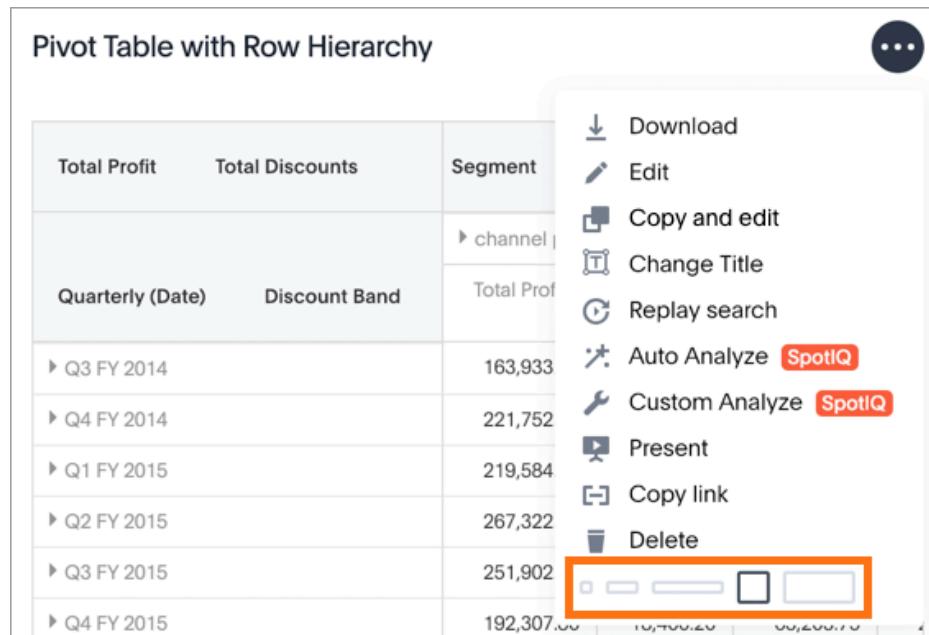


Change the orientation of a chart or table

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

1. Navigate to the chart or table.
2. Click the ellipses icon  for the chart or table.

ThoughtSpot displays the drop-down menu.



Total Profit	Total Discounts	Segment
		▶ channel
Quarterly (Date)	Discount Band	Total Prof
▶ Q3 FY 2014		163,933
▶ Q4 FY 2014		221,752
▶ Q1 FY 2015		219,584
▶ Q2 FY 2015		267,322
▶ Q3 FY 2015		251,902
▶ Q4 FY 2015		192,307...

The last item shows different possible orientations, with the current orientation selected.

3. Select a new orientation.
4. Click the ellipses icon  and select **Save**.

Delete a visualization from your pinboard

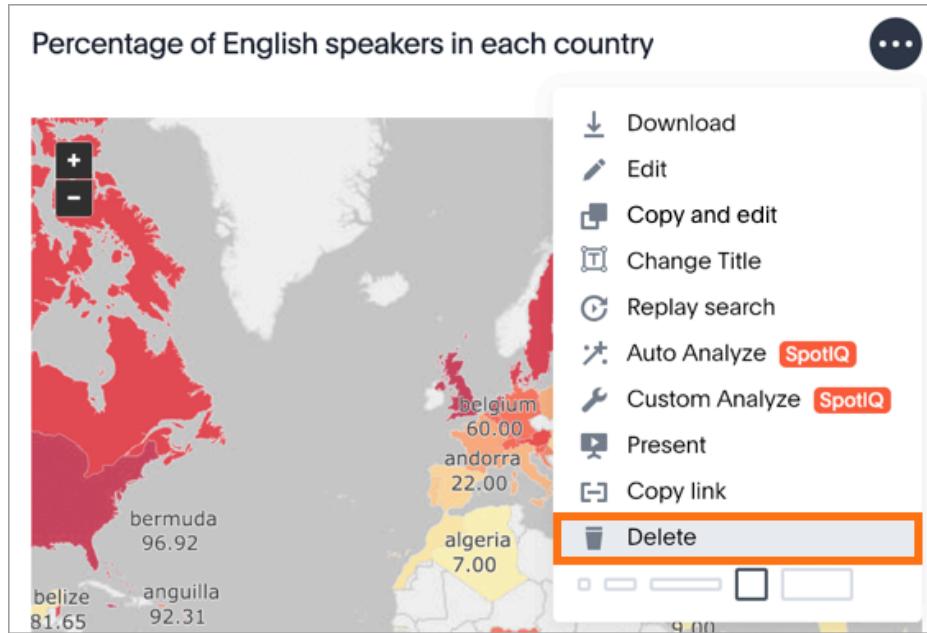
To delete a visualization:

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

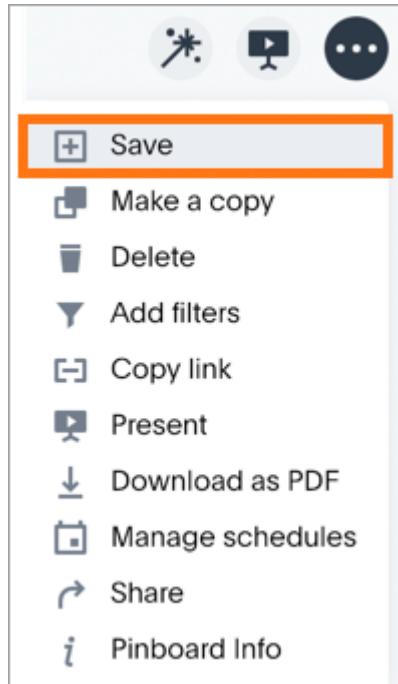


2. On the pinboard list page, click the pinboard that contains the visualization you would like to delete.

3. In the upper-right corner of the visualization, click the ellipses icon  and select **Delete**.



4. Save your pinboard by clicking the ellipses icon  and selecting **Save**.



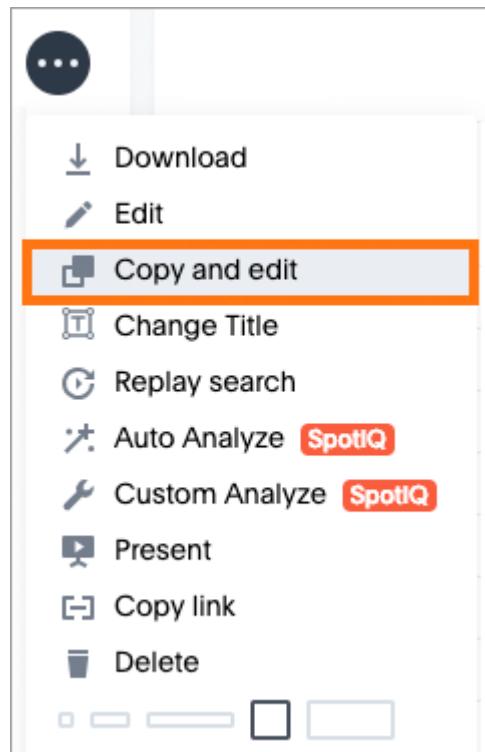
Copy and edit a visualization from your pinboard

To edit a copy of a visualization:

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



2. On the pinboard list page, click the pinboard that contains the visualization you would like to edit a copy of.
3. Find a visualization in the pinboard you want to edit.
4. In the upper-right corner of the visualization, click the ellipses icon and select **Copy and edit**.

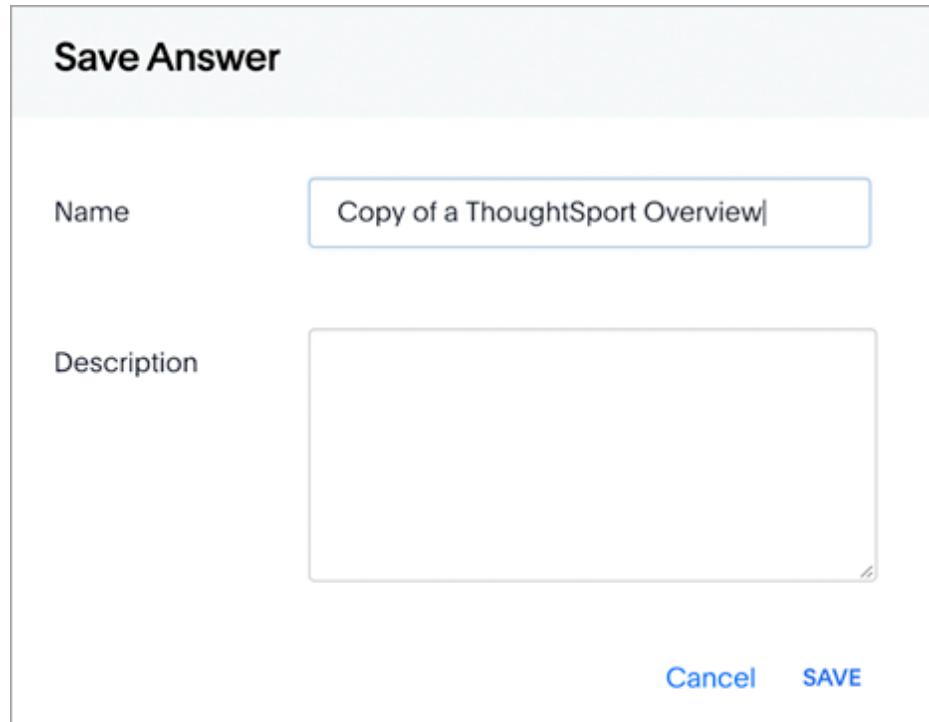


A copy of the answer that was used to create the visualization is displayed with “Copy of” added to the beginning of the title.

5. Make any changes you want to the visualization.

6. Click the ellipses icon  and select **Save**.

The Save Answer dialog box appears.

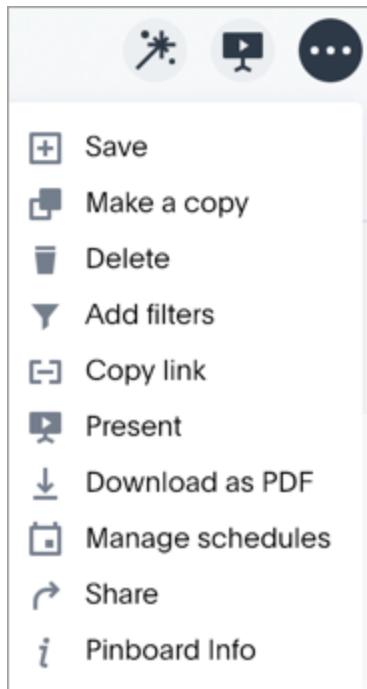


7. Change the name and description, as needed, and click **SAVE**.

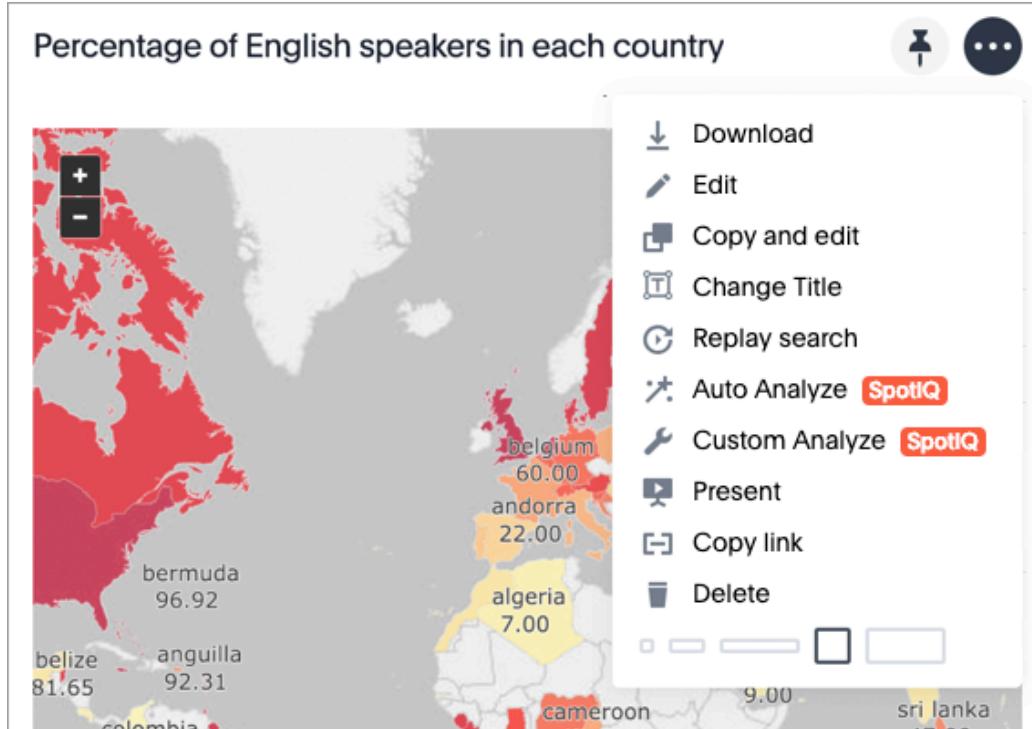
When you navigate back to the Answers page, your edited Answer will appear in the list.

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 ellipses icon .



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



Pinboard filters

Filters, including bulk filters and exclude 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.

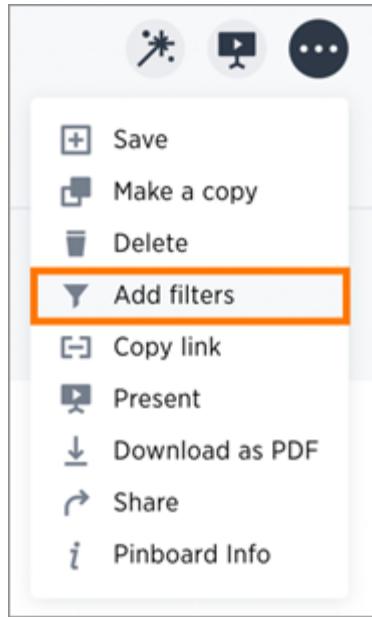
Note that pinboard filters only apply to the tables and charts that are based on worksheets. If a pinboard also includes tables and charts that were created from underlying tables or on user uploaded data, the filters don't apply to them.

Create a pinboard filter

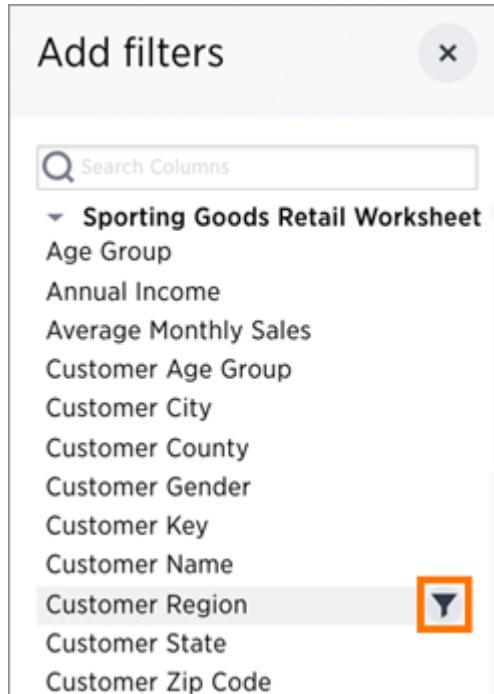
You can create a filter in the pinboard view for any pinboard. This allows you to easily manipulate the visualizations and view the modified presentation in one place.

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

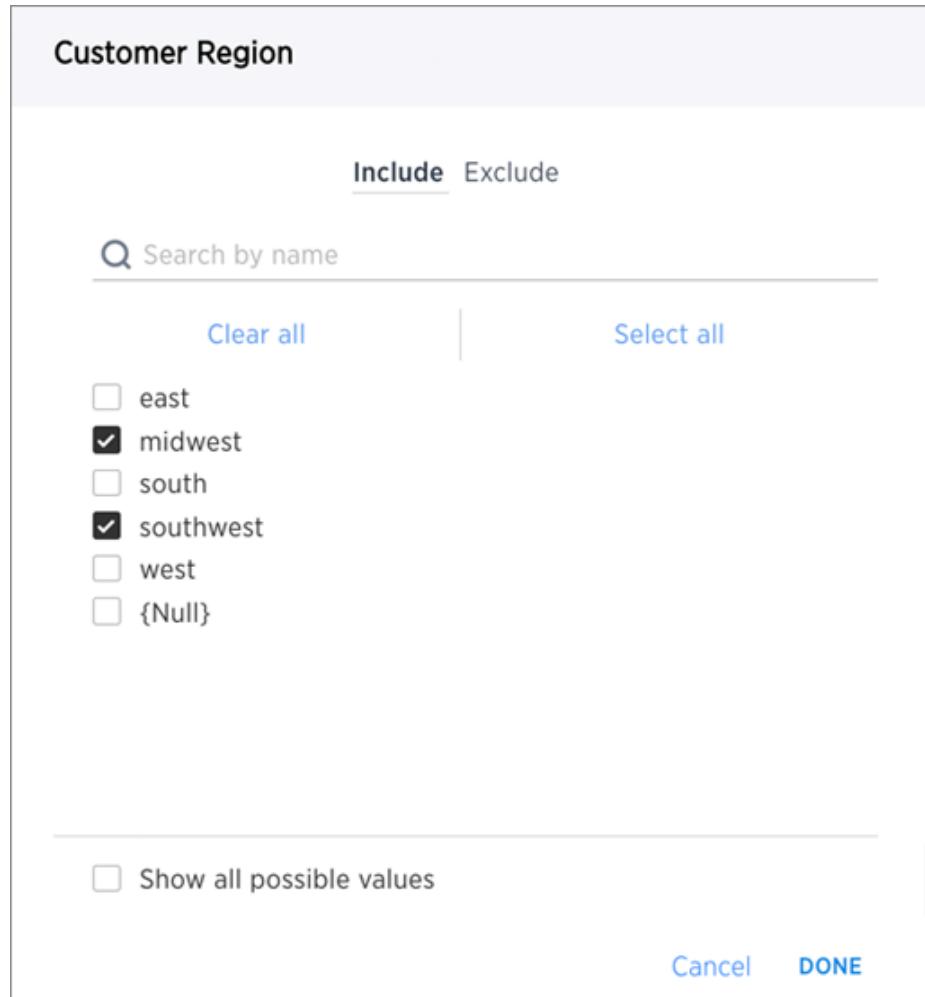
1. Click the ellipses icon  , and select **Add filters**.



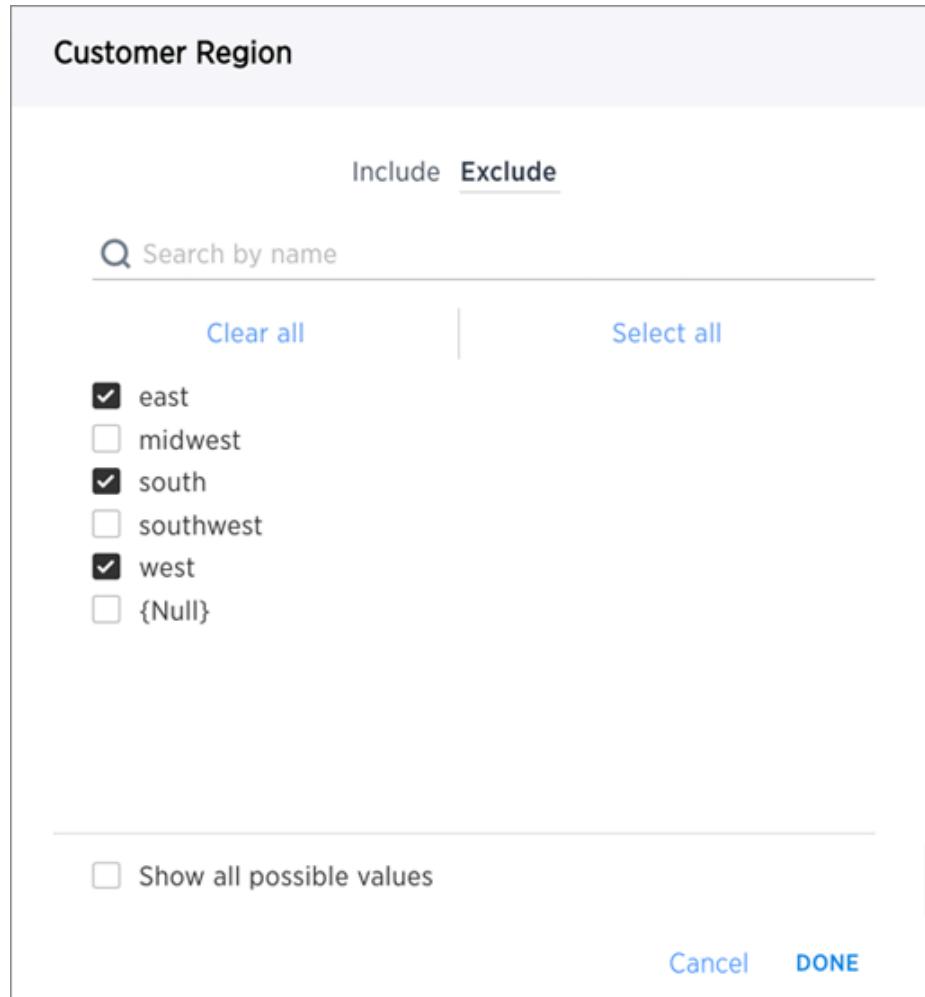
2. In the populated columns menu, click the **Add filter** icon next to the column you would like to filter on.



3. Select the values to include in your answer.



4. If you want to exclude values, click **Exclude** and choose values to exclude.

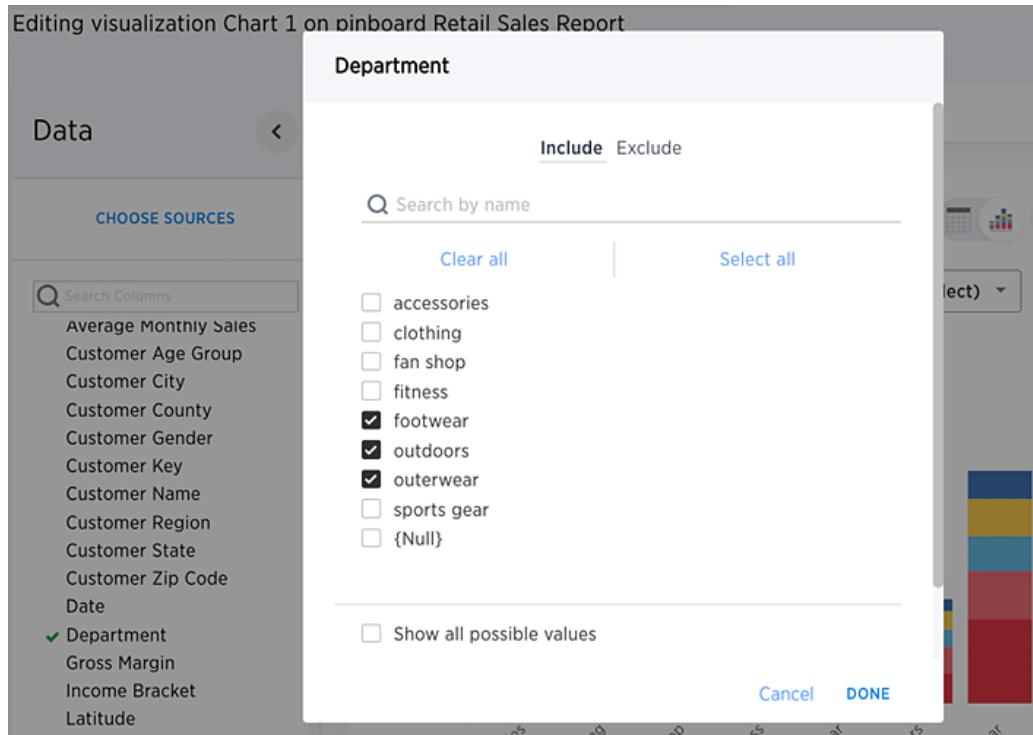


5. Click **Done**.

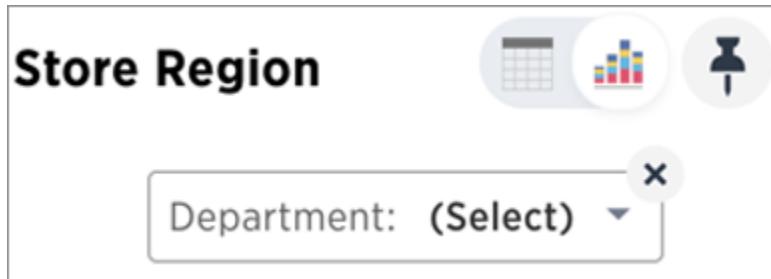
When you apply a filter, the pinboard is not automatically saved with your filter applied to every object in the pinboard that was created from a worksheet. Pinboard filters do not apply to tables and charts built directly on top of base tables.

Edit a filter from the answer

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



Or delete it:



Schedule a pinboard job

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

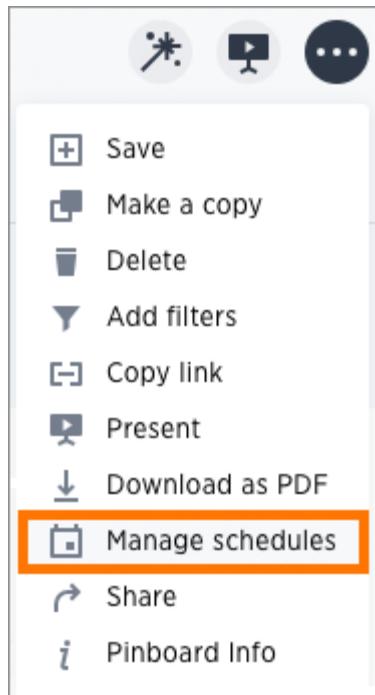
You can add multiple schedules with different configurations for a single pinboard. However, each job is limited to one pinboard schedule. To add a schedule, you must either be an administrator or have the schedule pinboard privilege, and at least edit-only and view-only access to the pinboard.

To schedule a pinboard:

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



3. Select the pinboard you want to create a schedule for.
4. Click the ellipses icon and select **Manage schedules** to view all of the schedules set for the pinboard.



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

Name	Status	Recipient	Created	Author
Finance Update Executive Team	▷ Scheduled	2 Recipients	in 21 seconds	Retail
Marketing Update Executive Team	▷ Scheduled	2 Recipients	20 seconds ago	Retail
Sales Update Company Executives	▷ Scheduled	2 Recipients	2 minutes ago	Retail

On the **Add a Schedule** page, you create a new schedule for your pinboard.

Add a schedule for [Sales Worldwide](#)

Schedule <p>Repeats <input type="button" value="Select an option"/></p> <p>Server time zone UTC</p> <p>Name * <input type="text" value=" "/></p> <p>Description <input type="text" value=" "/></p> <p>Type <input type="radio"/> CSV <input checked="" type="radio"/> PDF</p> <p>Gating condition <small>BETA</small> <input type="button" value="+ Add condition"/> <input type="button" value="Delete"/></p>	Recipients <p>Add Recipients</p> <p>Users or groups <input type="text"/> <input type="button" value="Add"/></p> <p>Emails <input type="text"/> <input type="button" value="Add"/></p>
<small>* Required field</small>	
<input type="button" value="Cancel"/> <input type="button" value="Schedule"/>	

6. Set the values for your schedule.

Field	Description
Repeats	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 Server time zone which is the timezone which will be used.
Name	Provide a short name for this schedule, <i>Monthly Report Source</i> is an example of a good name.
Description	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.
Type	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.

Gating condition

Write a statement that returns a single boolean value (true or false). For example, `sum (revenue) > 100` is a valid condition but `is_weekend (commit_date)` 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.

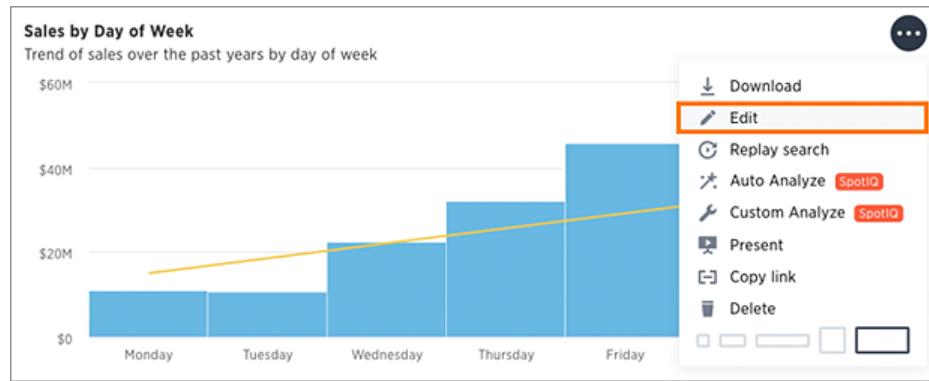
If you experience problems with sending email, your cluster may not have SMTP enabled. Contact your system administrator to resolve this issue.

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. Click the three-dot **Actions** button to the upper-right of your visualization and select **Edit**.



You will now be in 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](#).

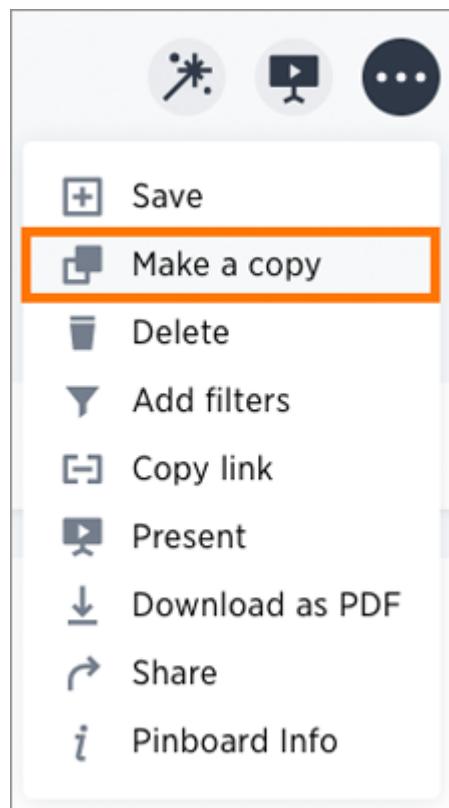
Copy a pinboard

You can make a copy of a pinboard if you would plan to make edits without overwriting the original pinboard. When you save a copy, you can use a new pinboard name.

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



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



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

Save Answer

Name

Description

[Cancel](#) [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 signing in, or through 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.

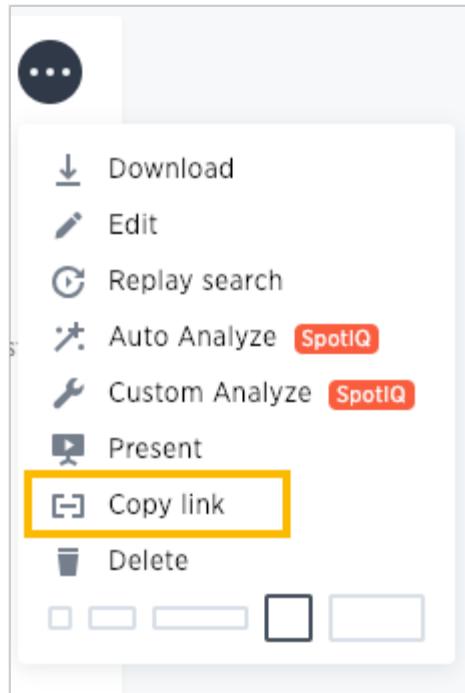
Copying a pinboard link

To copy the link for a pinboard:

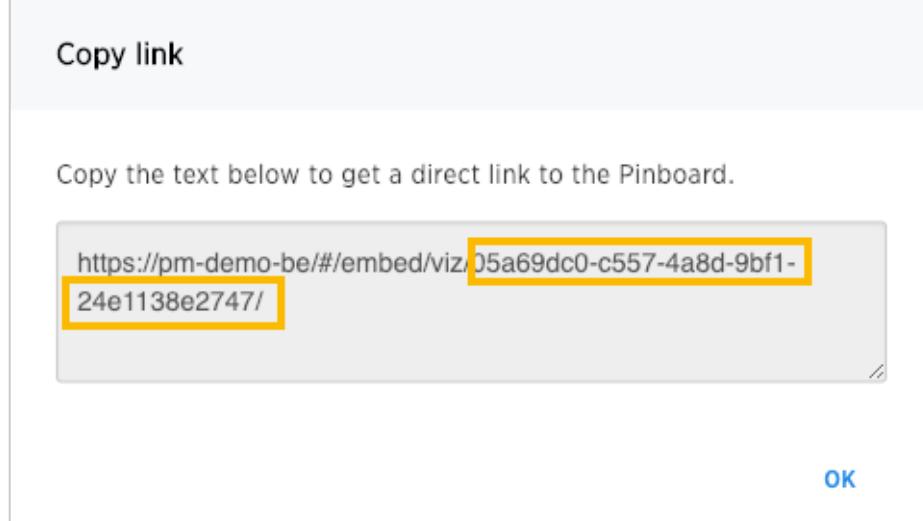
1. Click **Pinboards**, on the top navigation bar.



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

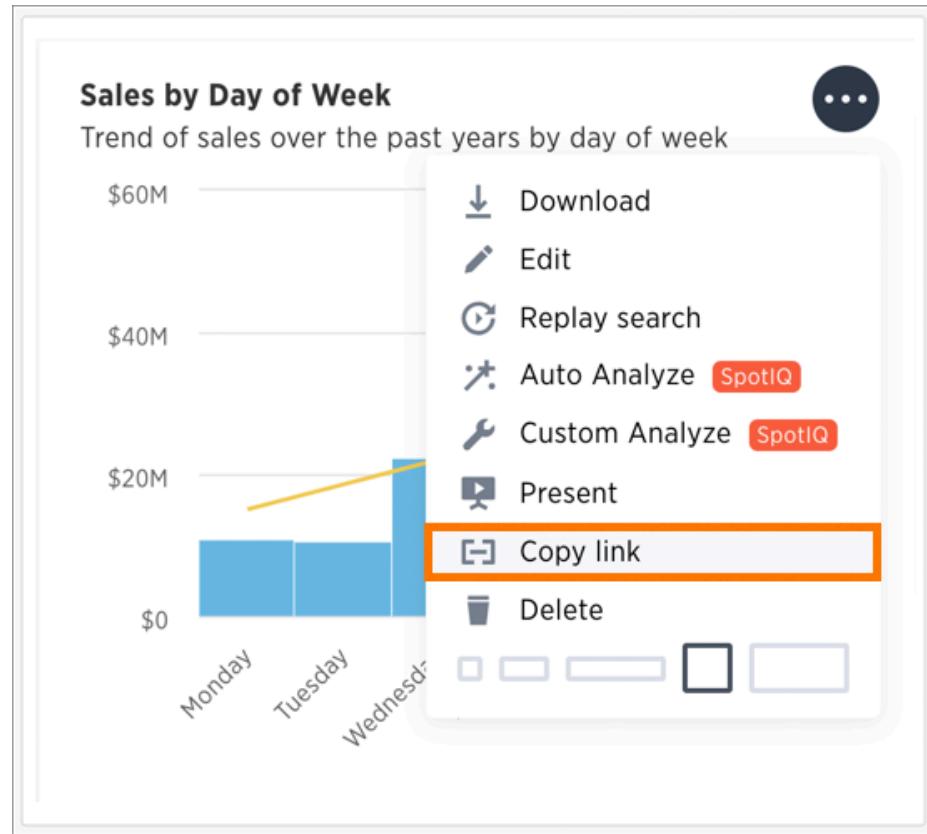


4. Copy the pinboard link. The highlighted portion is the pinboard ID.

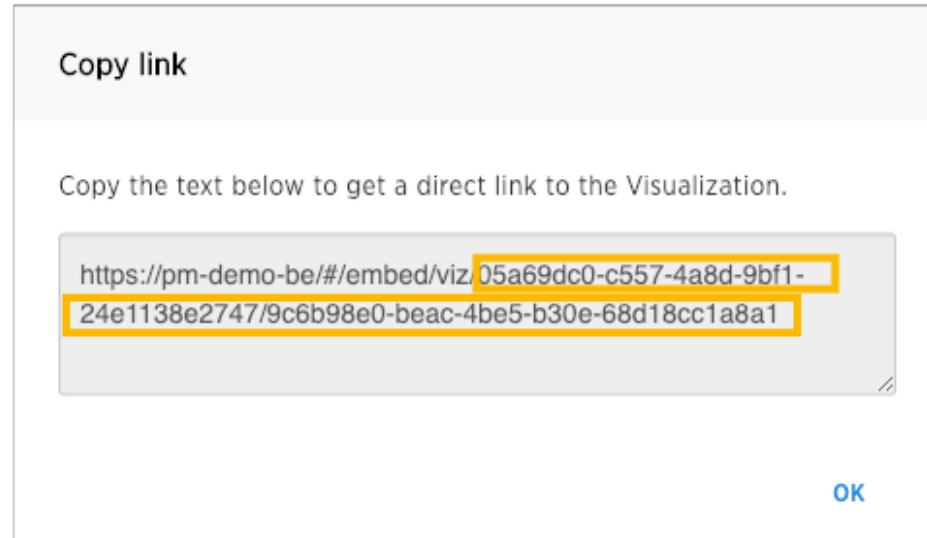


Copying a visualization link

1. To copy an individual visualization link, click the ellipses icon  and select **Copy link** of the visualization you would like to copy a link from.



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



Reset a pinboard visualization

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

Reset a pinboard visualization

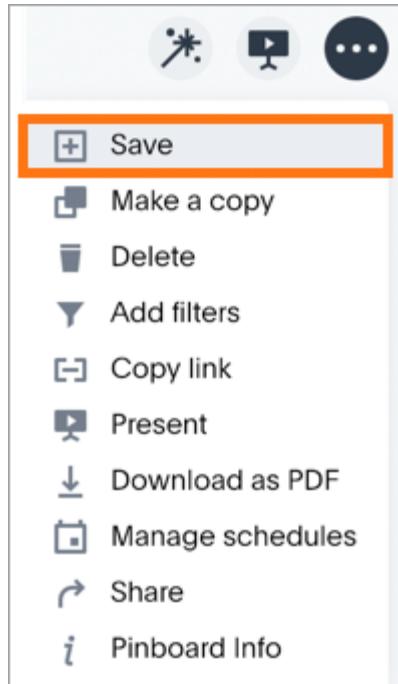
After performing ad hoc actions or edits to a visualization of your pinboard, you can reset the visualization to its original form.

To reset a visualization:

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



2. Save your pinboard by clicking the ellipses icon  and selecting **Save**.



Present a pinboard as a slideshow

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

With ThoughtSpot's interactive presentations, you can easily explore and answer questions from the audience during the presentation session. When you present your pinboard, ThoughtSpot displays the component visualizations and tables in order from left to right, and top to bottom.

Starting a presentation

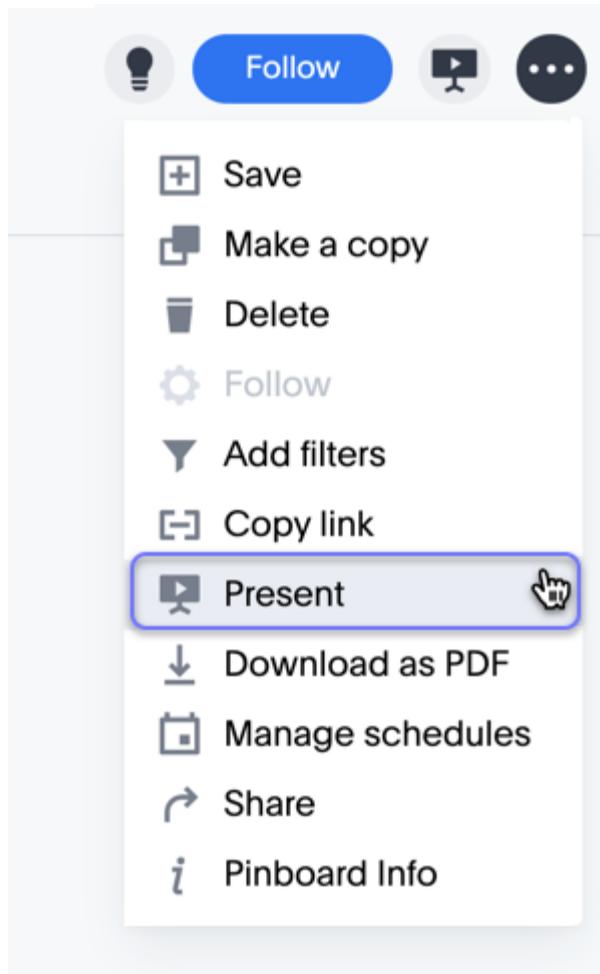
To start a presentation, follow these steps:

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



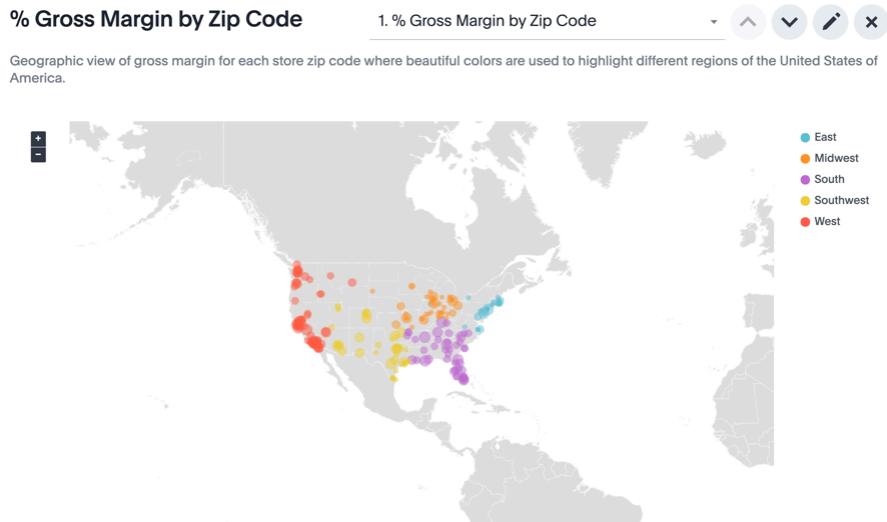
2. Select a pinboard from the list of pinboards by clicking on its title.

3. In the open pinboard, click the **More Options** menu (ellipses icon), and select **Present**.



4. ThoughtSpot displays the pinboard in full screen.

To navigate through the slides of the presentation, use the left and right arrow navigation keys on your keyboard.



Presentation navigation controls

The presentation navigation controls enable you to effectively run the presentation.

Hover over the top right corner of the presentation to see the navigation control options:

1. Click the downward arrow to expand the list of visualizations, and select it to navigate directly to this slide.
2. Click the up arrow to go to the previous slide.
3. Click the down arrow to go to the next slide.
4. Click the pencil symbol to edit the answer represented by the current slide.
5. Click the x to end the presentation.

16. Stacked charts → Horizontal legend

o. Line Chart- horizontal 900-1200 - data labels

9. Quantity and Tax by Region

10. Simple Sankey chart

11. Line chart- horizontal 900-1200 - data labels

12. Treemap, 2 categories

13. Geo Bubble - France Total population

14. Pivot table with multiple measures without summaries

15. Treemap, 2 categories

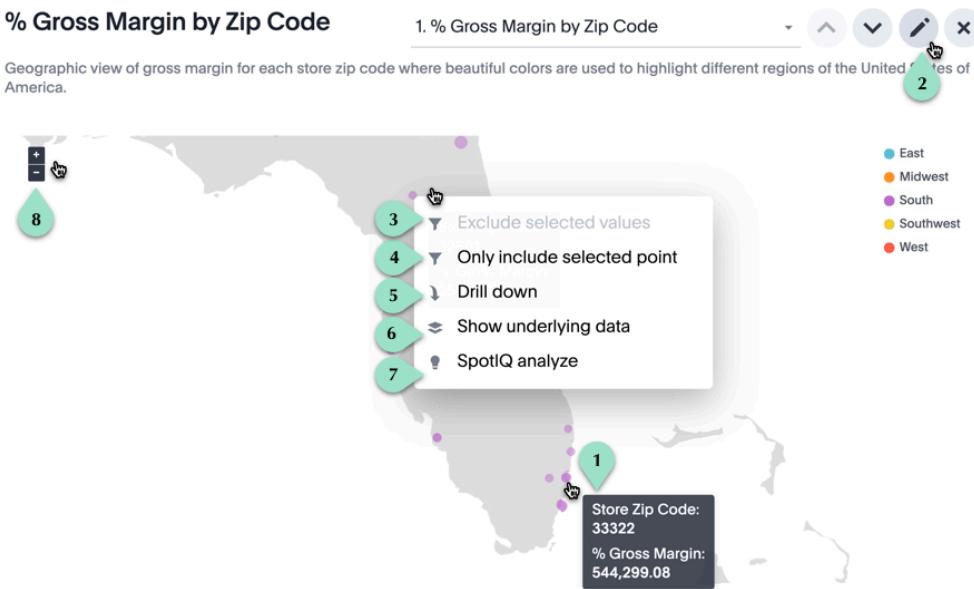
✓ 16. Stacked charts → Horizontal legend

17. Zoomed bar chart

18. Bubble chart . Excepteur sint occaecat cupidatat non proident, sunt in culpa qui offici...

Presenting pinboards

While in presentation mode, you can easily explore each visualization.



Consider the following actions on pinboards:

Legend Action

1. You can see the tooltip information about each element of the visualization as you hover the cursor over that area of the presentation.
2. To quickly edit a visualization, click the pencil icon, and proceed to make the necessary changes. This takes you out of presentation mode.
3. By left-clicking, you can open a drop menu where you can change filtering to **Exclude selected values**.
4. Similarly, you can apply a filter to **Only include selected values** or **Only include selected point**.
5. You can drill down to see more information about the columns used in search. Left-click and select **Drill down** from the menu.

Legend Action

6. To show underlying data in more detail, left-click and select **Show underlying data** from the menu.
7. To run SpotIQ, left-click and select **SpotIQ analyze** from the menu.
8. Depending on the chart type in presentation, you can perform the chart-specific actions.
In case of geographic maps, you can zoom and pan the map.

Download a pinboard as PDF

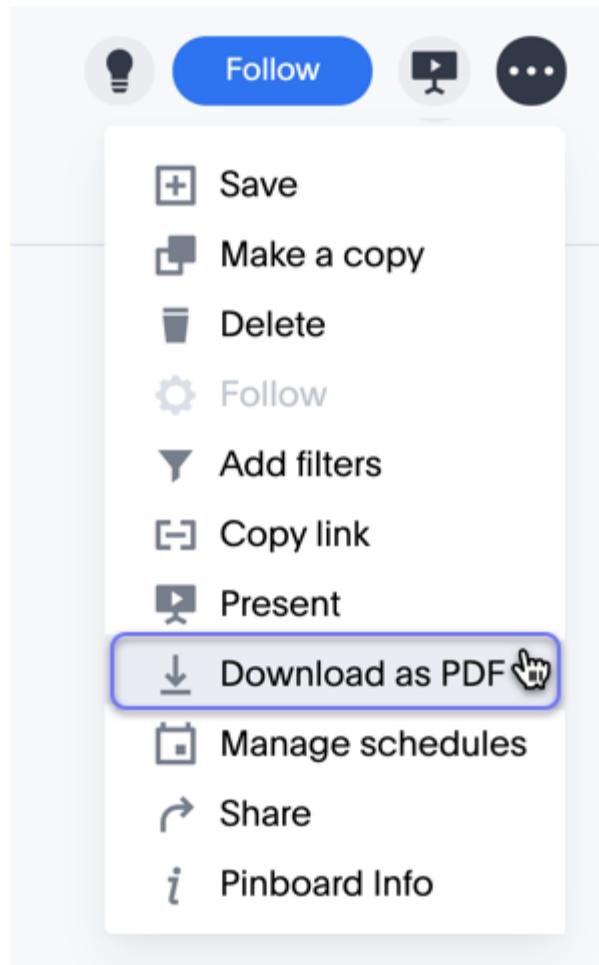
Summary: In ThoughtSpot, you can download the entire pinboard in PDF format, so you can share it with people inside and outside your organization. You have options for adding branding, composing the report of the entire Pinboard or only some of its Visualizations, truncating tables, and many others.

You can download a Pinboard in PDF format, without downloading each visualization separately. Follow these steps:

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

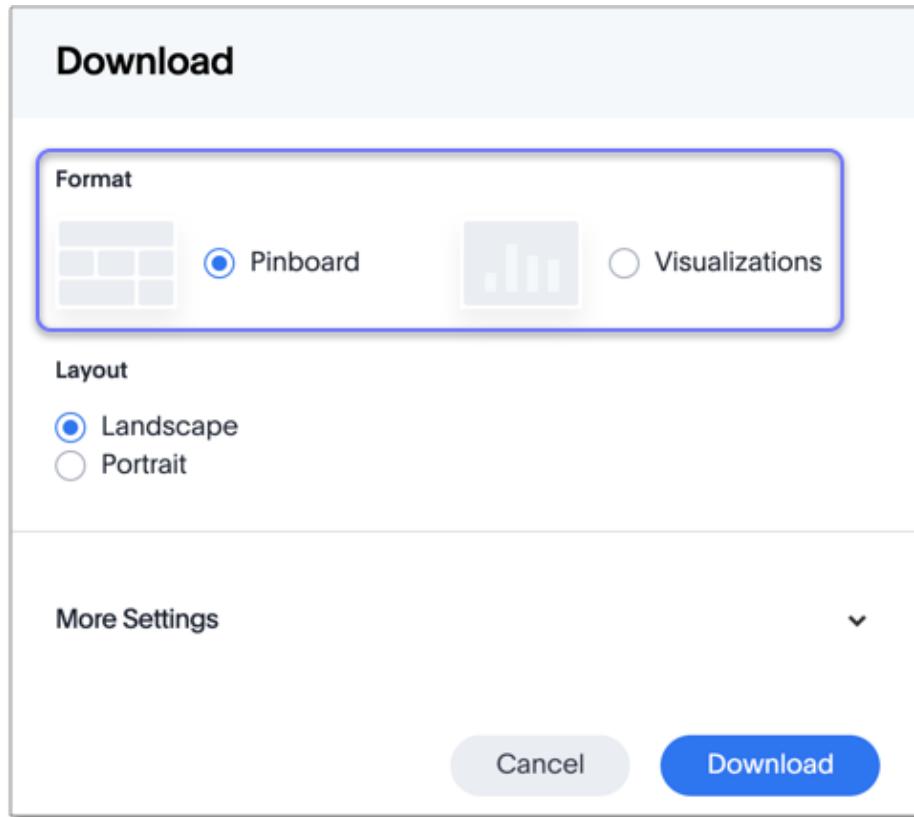


2. Select a Pinboard from the list of Pinboards by clicking on its title.
3. In the open Pinboard, click the **More Options** menu (ellipsis icon), and select **Download as PDF**.



4. In the **Download** window modal, you have two main *Format* output options:

- Select **Pinboard** to generate a PDF that combines several visual answers on the same page, just like they appear on the Pinboard.
- Select **Visualizations** to print each visualization on its own page of the PDF file.



Download PDG as Pinboard

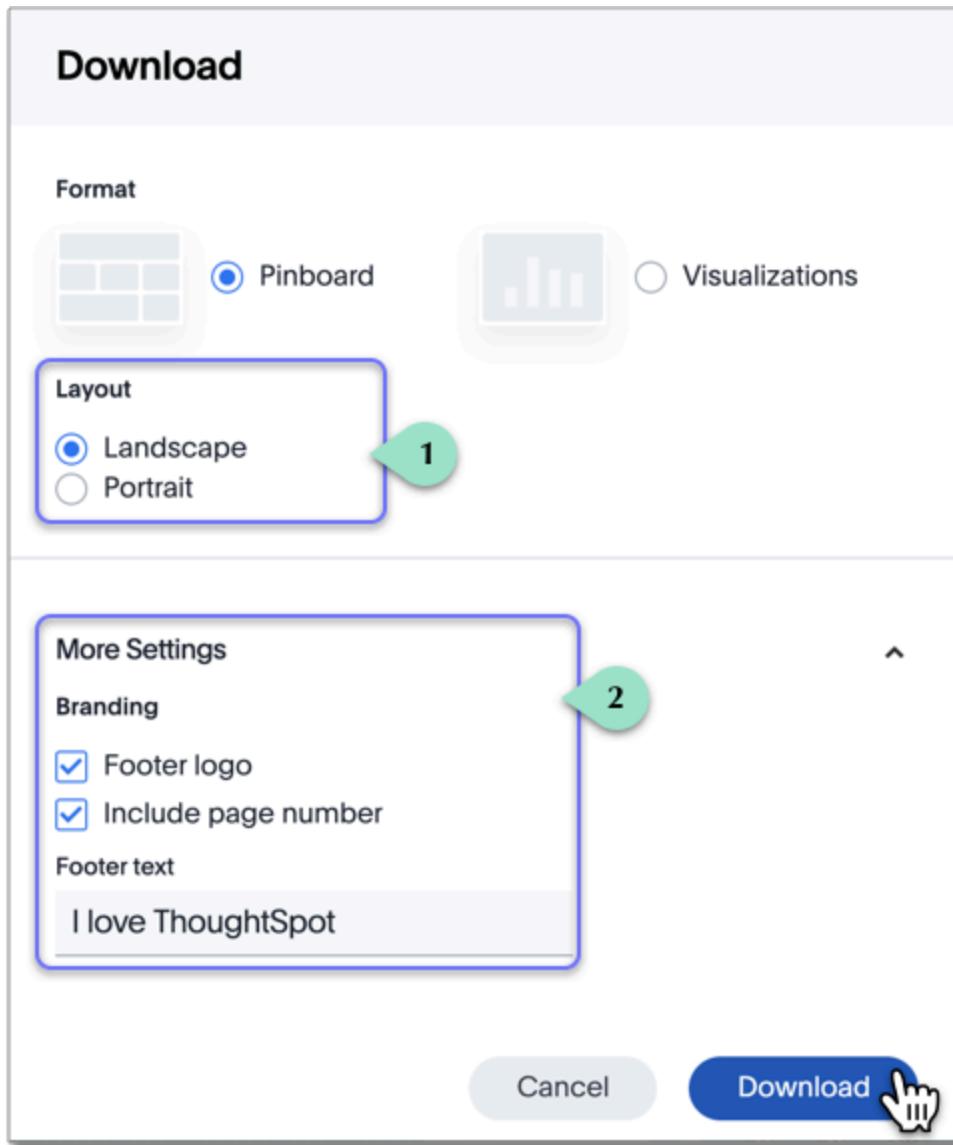
The Pinboard mode produces a PDF file that looks very similar to the Pinboard inside ThoughtSpot. The highlights and insights appear at the top, and the rest of the Pinboard is on a rectangular grid, with natural page breaks.

Follow these steps to specify additional options:

1. Under **Layout**, select either the *Landscape* or *Portrait* page orientation.
2. Click the **More Settings** option.

Under **Branding**, specify if you want to add a *Footer logo*, *Include page number*, or add *Footer text*.

3. Click **Download**.



Download PDF as Visualizations

When you select **Visualizations**, you now have a choice to print all, or some, of the visualizations in the Pinboard. The highlights and insights appear at the top, and the rest of the visualizations get their own page.

Follow these steps to specify additional options:

1. Under **Layout**, select either the *Landscape* or *Portrait* page orientation.
2. Click the **More Settings** option.

Under **Content**, specify if you want the table visuals to end after filling the whole page by selecting the *Truncate Tables* option.

3. Add **Branding** details through the *Footer logo*, *Include page number*, or *Footer text* options.
4. In the list of **Visualizations**, select the visualizations you want in the PDF. Deselect the visualization you don't want.
5. Click **Download**.

Download

Format

Pinboard Visualizations

Layout

Landscape Portrait

More Settings

Content

Truncate Tables

Branding

Footer logo
 Include page number

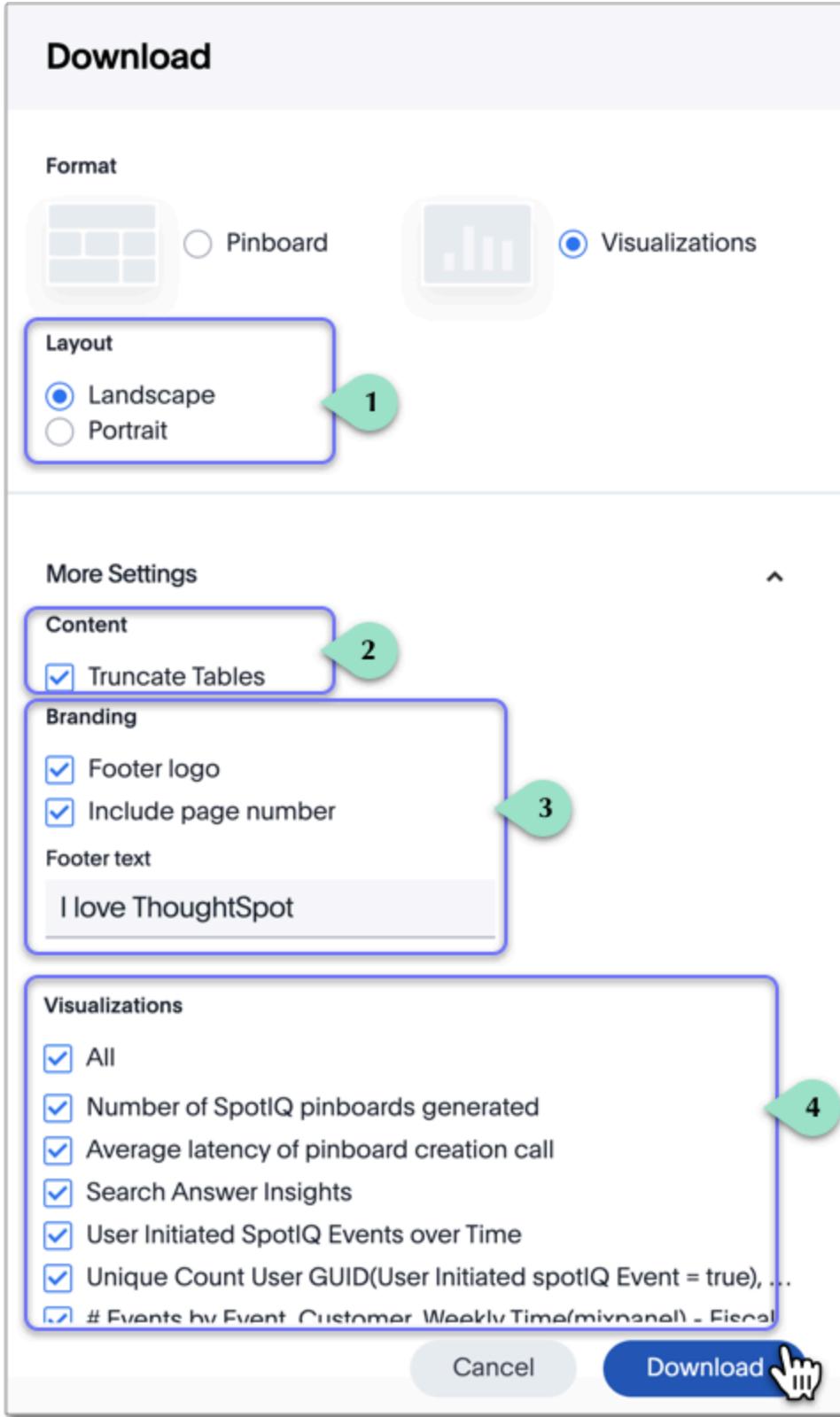
Footer text

I love ThoughtSpot

Visualizations

All
 Number of SpotIQ pinboards generated
 Average latency of pinboard creation call
 Search Answer Insights
 User Initiated SpotIQ Events over Time
 Unique Count User GUID(User Initiated spotIQ Event = true), ...
 # Events by Event Customer Weekly Time/mivnnonal - Fiecal

Cancel Download 



Examine the PDF

After the download completes, examine your PDF.

Notice that the PDF produced in **Pinboard** mode combines the visuals on the page in the same manner as the original Pinboard. Contrast this with the PDF produced in **Visualizations** mode: this PDF combines the small adjacent indicator and KPI visuals at the top, and all other visuals and tables appear on their own page.

Pinboard format

Big Pinboard

Date: 06/29/2019

Line chart- horizontal 900...
5
Treemap, 2 categories
20.2K
Line chart- horizontal 900...
5
Treemap, 2 categories
5
Simple Sankey chart
Quantity and Tax by Region
Line chart- horizontal 900-1200 - data labels
Market Segment Customer Region Total Quantity Total Tax
automobile africa 4,030
automobile america 6,405
automobile asia 5,010
automobile europe 3,198
automobile middle east 5,458
building africa 5,598
building america 5,301
building asia 5,420
building europe 4,780
building middle east 4,533
furniture africa 4,250
20 rows total

Line chart- horizontal 900...
25.3K
Line chart- horizontal 900...
20.2K
Line chart- horizontal 900...
127K
Pivot table with multiple measures without summaries
Total Revenue, Total Discount by Quarterly (Order Date), Order Priority and Market Segment, Customer Region
Quarterly Order Date Order Priority Total Revenue Total Discount Market Segment Customer Region
automobile africa 4,030
automobile america 6,405
automobile asia 5,010
Geo Bubble - France Total Population by Department INSEE code and
Bubble chart. Excepteur sint occaecat cupidatat non proid.
Stacked charts -> Horizontal
Zoomed bar chart
Treemap, 2 categories
Heatmap
Market Segment Customer Region Total Quantity Total Tax
automobile africa 4,030
automobile america 6,405
automobile asia 5,010
automobile europe 3,198
automobile middle east 5,458
building africa 5,598
building america 5,301
building asia 5,420
building europe 4,780
building middle east 4,533
furniture africa 4,250
20 rows total

2Lines-1Bar Chart
Geo Area-Total Poverty Estimate by State
with legend and data label
Pivot table with multiple measures
Total Revenue, Total Discount by Quarterly (Order Date), Order Priority and Market Segment, Customer Region
Quarterly Order Date Order Priority Total Revenue Total Discount Market Segment Customer Region
Q2 2001 12,260,686 14 54
Q2 2002 17,000,828 24 47
Q4 2002 27,653,345 43
Q1 2003 10,896,375 12 3
Q2 2003 43,647,425 37 16
Q2 2001 44
Total Quantity by Market Segment, Customer Region
Customer Region
Market Segment
automobile
building
furniture
household
Simple Sankey chart
Market Segment Customer Region
automobile
building
furniture
household
Customer Region
Market Segment
automobile
building
furniture
household

Visualizations format

Big Pinboard

Date: 06/29/2019

Line chart- horizontal 900...
5
Treemap, 2 categories
20.2K
Line chart- horizontal 900...
5
Line chart- horizontal 900...
20.2K
Line chart- horizontal 900...
127K
Line chart- horizontal 900...
25.3K
Line chart- horizontal 900...
5
Treemap, 2 categories
20.2K
Line chart- horizontal 900...
127K
Quantity and Tax by Region
Geo Area-Total Poverty Estimate by State
with legend and data label
Pivot table with multiple measures
Total Revenue, Total Discount by Quarterly (Order Date), Order Priority and Market Segment, Customer Region
Quarterly Order Date Order Priority Total Revenue Total Discount Market Segment Customer Region
Q2 2001 12,260,686 14 54
Q2 2002 17,000,828 24 47
Q4 2002 27,653,345 43
Q1 2003 10,896,375 12 3
Q2 2003 43,647,425 37 16
Q2 2001 44
Total Quantity by Market Segment, Customer Region
Customer Region
Market Segment
automobile
building
furniture
household
Simple Sankey chart
Market Segment Customer Region
automobile
building
furniture
household
Customer Region
Market Segment
automobile
building
furniture
household

About R in ThoughtSpot

Summary: Analyze your data with R scripts that ship with ThoughtSpot, and build visualizations and Pinboards based on the analytical insights you obtained. You can also create custom R scripts, and share them with your team.

R is a popular open source programming language for statistical computing, machine learning, and AI. ThoughtSpot makes R analysis available as a fully integrated feature.

Note: Custom R is disabled by default. To turn on custom R scripting, contact your site administrator.

Using R in ThoughtSpot

Users with R privileges can run R scripts directly on search results.

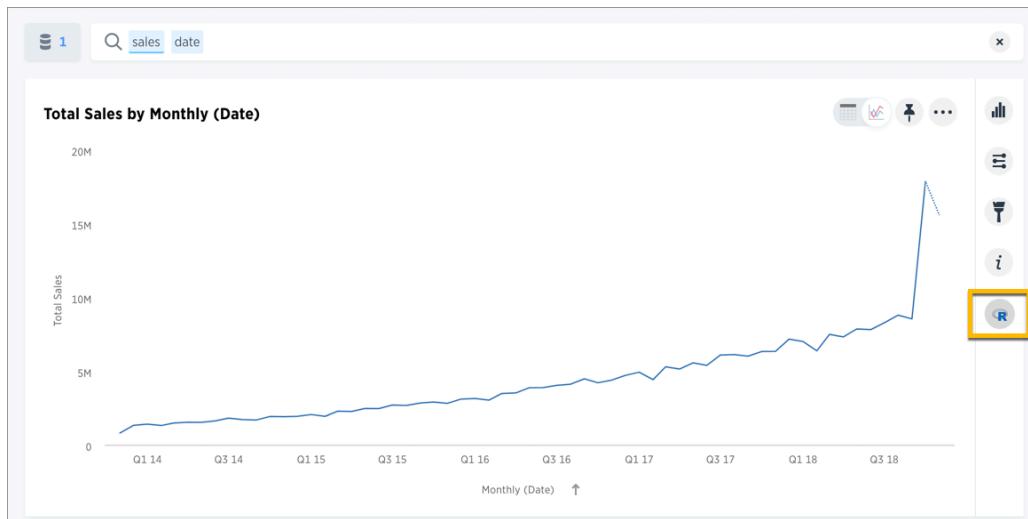
Analysts and data scientists who are proficient in R can create and share custom scripts.

Users can leverage custom scripts and ThoughtSpot provided scripts, run pre-built R scripts on top of their search results, and share R visualizations as answers and pinboards.

This topic explains how to use R in ThoughtSpot both from an end user and scripting perspective, but is not meant as an R primer.

How to access R scripts

Users with R script privileges can click the  icon on the toolbar for any search result (answer).



From here, you have options to write a custom script, or load a pre-built or ThoughtSpot provided script.

Run pre-built R scripts

You don't have to have a background in statistics or be an R programmer to run R analyses in ThoughtSpot. You can use ThoughtSpot-provided scripts and share the R visualizations with others.

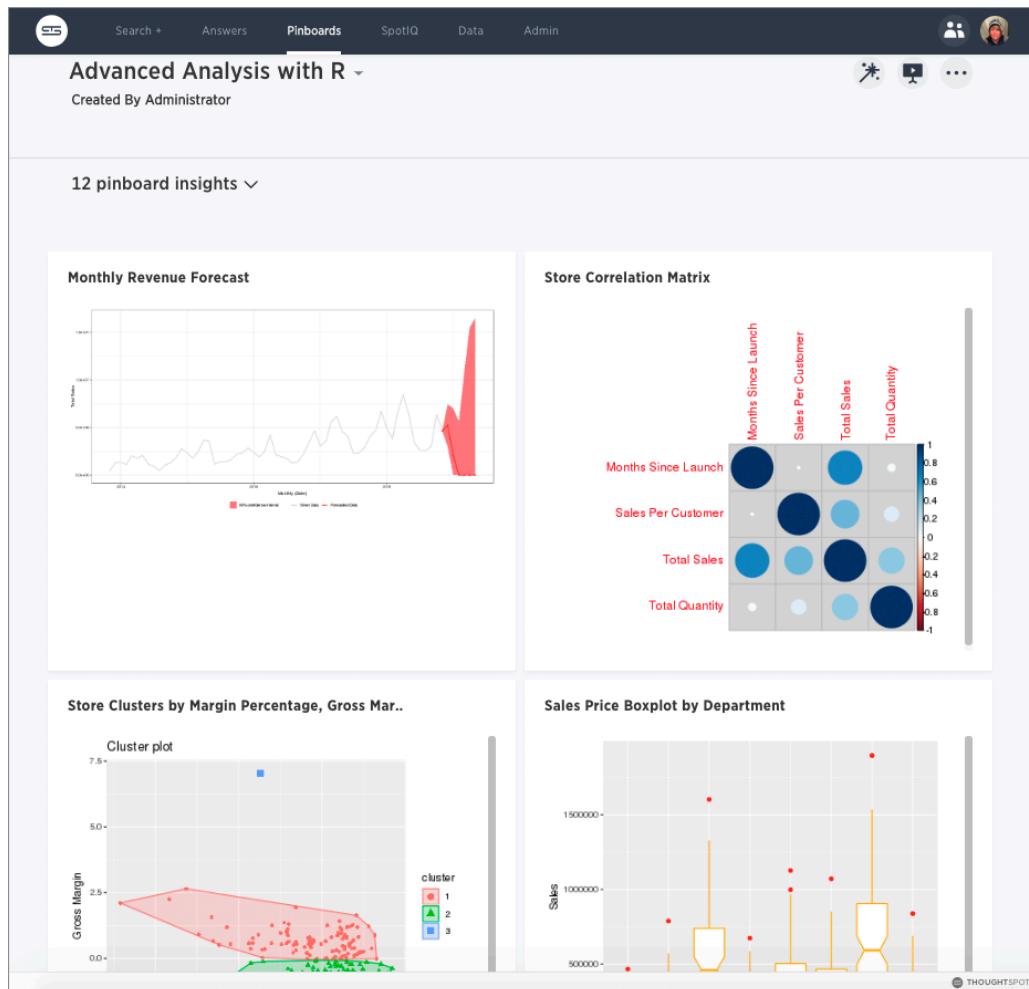
For more on how to run provided scripts, skip to [Run pre-built R scripts on answers](#)

Write your own R scripts in ThoughtSpot

If you know R, you can write your own custom scripts, share them as templates, test and run them on your data in ThoughtSpot, and build up a shared library of R analyses, scripts, visualizations, and pinboards.

Start with the topic on how to [Create and share R scripts](#) to learn more about writing R scripts in ThoughtSpot, including a few particulars on syntax and column bindings.

To learn more about R programming in general, a good place to start is [R Project for Statistical Computing](#). Also, Anthony Chen's blog post on [Using R Analysis in ThoughtSpot for Time Series Forecasting](#) is a nice introduction to writing R scripts in ThoughtSpot.



Related information

- [Create and share R scripts](#)
- [Run pre-built R scripts on answers](#)
- [SpotIQ custom analysis](#)
- [Understand groups and privileges](#)
- [tscli rpackage](#)

Create and share R scripts

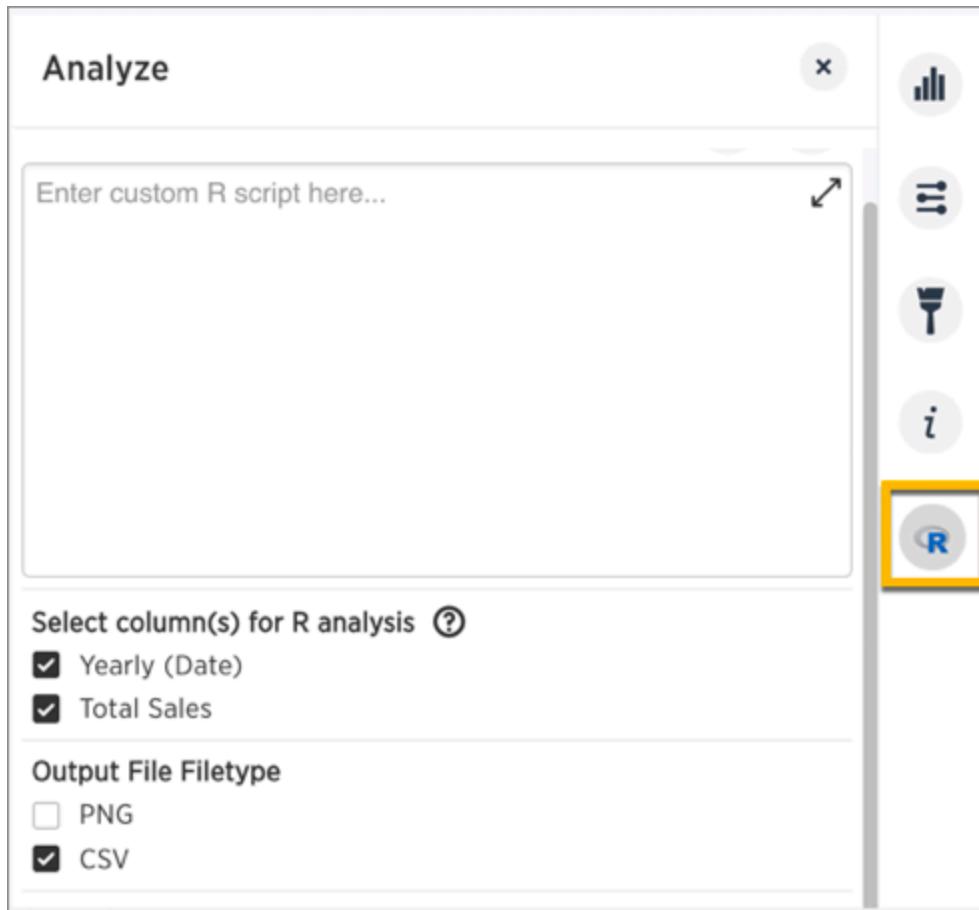
Summary: You can create and share your custom R scripts in ThoughtSpot.

Overview

Note: Custom R is disabled by default. To turn on custom R scripting, contact your site administrator.

Each ThoughtSpot cluster is capable of running an R analysis on your data. To perform your own R analysis, start with a simple search and click the R icon on the right.

Use the **Custom R Script** dialog and settings to enter your script, set which columns to include for analysis, and indicate what output data to expect from your script (PNG or CSV).



ThoughtSpot also supports the sharing of scripts among users to enable you to share your powerful R analyses across the system, and allow others to run your scripts on different search results.

How it works

An R script in ThoughtSpot is based off of your original search (both the data and the schema), using the columns you select for the analysis. You can select all columns used in the original search or a subset of those columns, depending on the script.

ThoughtSpot auto generates objects with variable names by which you can refer to data elements in your script.

You can refer to the data in the selected columns by using vectors that ThoughtSpot generates for these before the script is run. The first column you select has the variable name `.param0`, the second column you select has the variable name `.param1`, and so on. This naming scheme continues if there are more columns. (You can click the question mark icon  next to “Select columns for R analysis” to get a visual mapping of how the columns are bound to variables in R.)

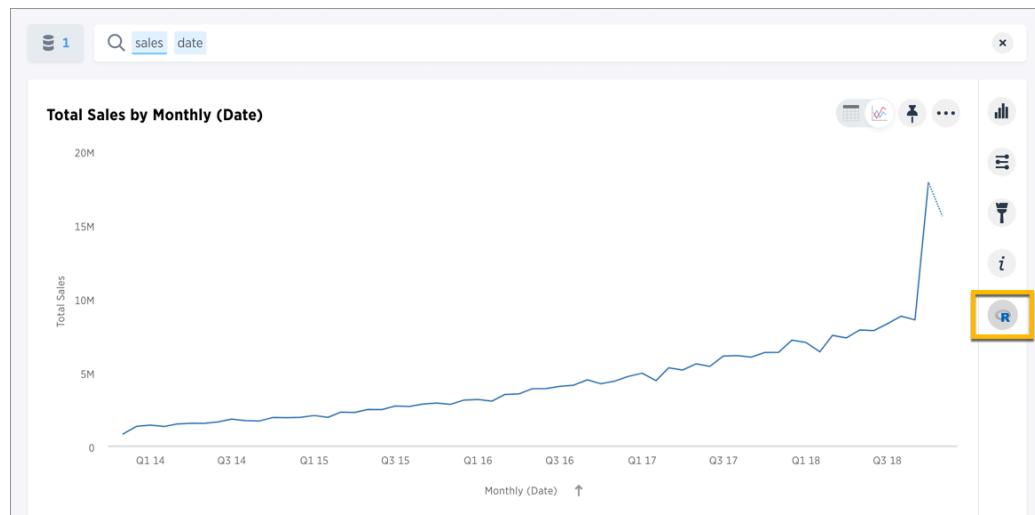
ThoughtSpot also provides an automatically-generated *data frame* object, `df`, that contains all selected column vectors. The data frame is R’s representation of a table (a 2D data structure containing rows and columns).

When the script executes, it passes the information to the server to run the analysis, and displays the answer as a visualization, in either PNG or CSV (table) format.

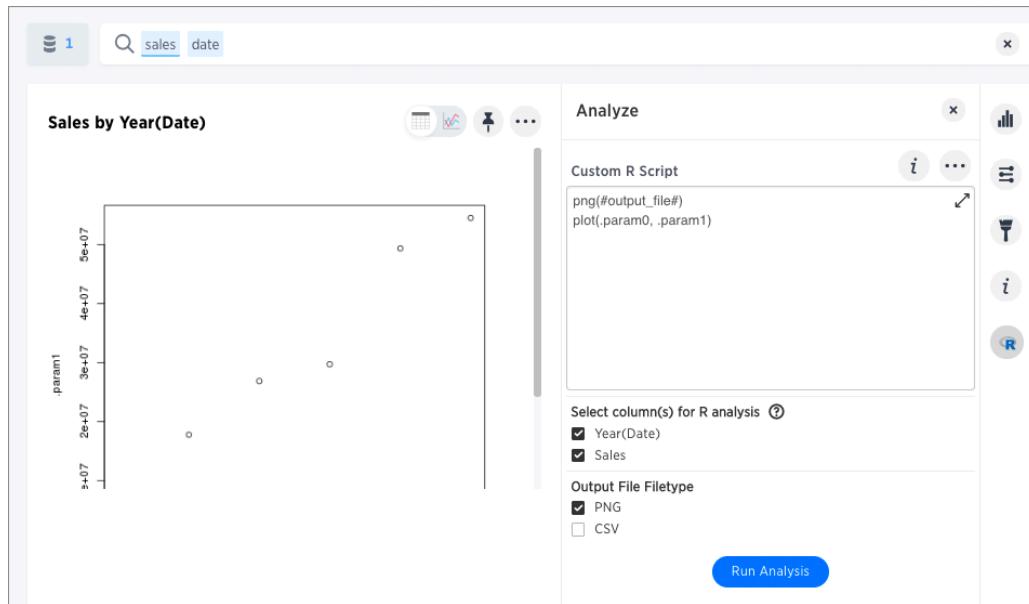
As we demonstrate in the following examples, you can chain R analyses together by running another R script on the answer to a previous script.

Write a custom script

To add a custom R script, start by running a search, then click the **R** icon  on the toolbar.



Add your script into the **Custom R Script** dialog, and set the **column bindings** and **output type** appropriately, as shown in the following examples.



For an R script to produce a meaningful output in ThoughtSpot, the script should generate a CSV or PNG file. Output should be written to the token `#output_file#`. CSV output is rendered back as a simple table in ThoughtSpot, PNG output is rendered back as a static PNG.

Basic R script to generate CSV data

This is an example of a basic R script that generates CSV data. (The `df` variable name must be lower case, as shown.)

```
write.csv(df, #output_file#)
```

The generated data is displayed back as a table when you run the analysis:

The screenshot shows a ThoughtSpot search interface with the query "sales date". On the left, a table titled "Sales by Year(Date)" displays six rows of data:

	Year(Date)	Sales
1	1357027200	1338265.38718
2	1420099200	26861846.8610099
3	1388563200	17754524.3289249
4	1514793600	54528419.326855
5	1451635200	29706221.0884754
6	1483257600	49357508.4623298

(showing rows 1-6 of 6)

On the right, the "Analyze" panel is open with the "Custom R Script" section containing the code:

```
write.csv(df, #output_file#)
```

The "Select column(s) for R analysis" section has "Year(Date)" and "Sales" checked. The "Output File filetype" section has "CSV" checked. A "Run Analysis" button is at the bottom.

This basic script returns the exact same table results as if you did the query directly in ThoughtSpot and selected the table view, barring some additional formatting you get for free on the search.

The screenshot shows a ThoughtSpot search interface with the query "sales date". On the left, a chart titled "Total Sales by Yearly (Date)" displays data for years 2016 through 2014:

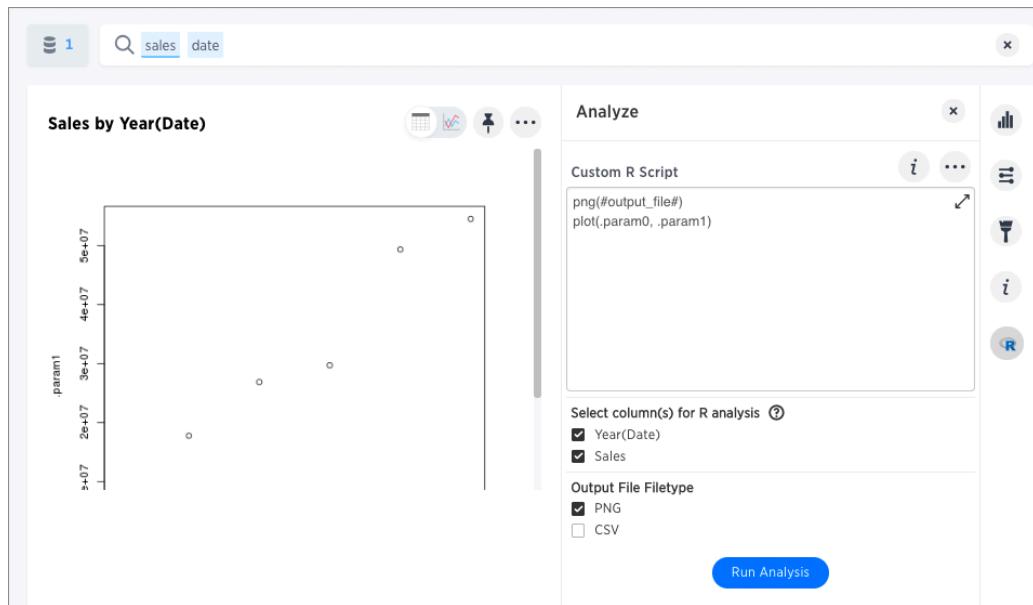
Yearly (Date)	Total Sales
2016	US\$29,706,221.09
2013	US\$1,338,265.39
2015	US\$26,861,846.86
2017	US\$49,357,508.46
2018	US\$54,528,419.33
2014	US\$17,754,524.33

Basic R script to generate a PNG graphic

This is an example of a basic R script that generates PNG data as output.

```
png(#output_file#)
plot(.param0, .param1)
```

The generated data is displayed back as a static PNG when you run the analysis:



Column bindings and output file type

- Under **Select column(s) for R analysis**, you specify the data you want to send to R and how to send it. If you do not make any choices here, all columns in the search are selected in the order they appear in the search bar.

In the examples above, `Year (Date)` is `.param0`, and `Sales` is `.param1`. To verify this, click the question mark icon next to **Select columns for R analysis** to see the “column bindings”. Reordering the columns changes the column bindings/params.

- Under **Output File Type** Select PNG or CSV, depending on the output details and in your script.

Options on scripts

You can click these icons in the R script dialog to get more options:

- The R script information icon  provides a basic reference guide for creating an R analysis in ThoughtSpot.
- The ellipses icon  provides a menu with options to save or load a previously saved R script, as well as share your R script with other users in the system.
- The arrow at the top right of the script dialog  opens a popout editor that gives you a larger space in which to view and edit your R script.
- The question mark icon  next to “**Select column(s) for R analysis**” provides a visual mapping of how the columns are bound to variables in R.

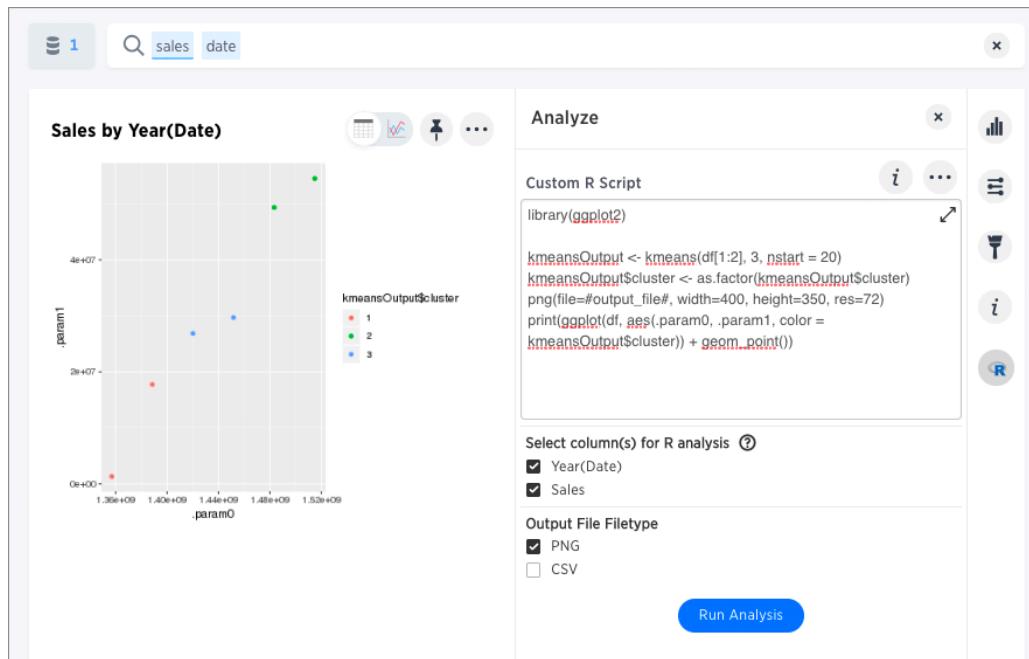
K-Means clustering example scripts

This script loads the `ggplot2` visualization package, labels the columns inside of the ThoughtSpot generated data frame object (`df`), and runs a K-Means clustering algorithm.

```
library(ggplot2)

kmeansOutput <- kmeans(df[1:2], 3, nstart = 20)
kmeansOutput$cluster <- as.factor(kmeansOutput$cluster)
png(file=#output_file#, width=400, height=350, res=72)
print(ggplot(df, aes(.param0, .param1, color = kmeansOutput$cluster)) + geom_point())
```

When you run this script on the results of the original `sales date` search, you get the following visualization as a PNG.



You can run another R script directly on this result to get CSV results:

```
kmeansOutput <- kmeans(df[1:2], 3, nstart = 20);
df$Cluster <- as.factor(kmeansOutput$cluster);
write.csv(df, file=#output_file#, row.names=FALSE);
```

The script for CSV output generates a table:

The screenshot shows the ThoughtSpot interface. On the left, there is a data table titled "Sales by Year(Date)" with columns: Yearly (Date), Total Sales, and Cluster. The table contains six rows of data. On the right, there is an "Analyze" panel with a "Custom R Script" section containing the following code:

```
kmeansOutput <- kmeans(df[1:2], 3, nstart = 20);
df$Cluster <- as.factor(kmeansOutput$cluster);
write.csv(df, file=output_file#, row.names=FALSE);
```

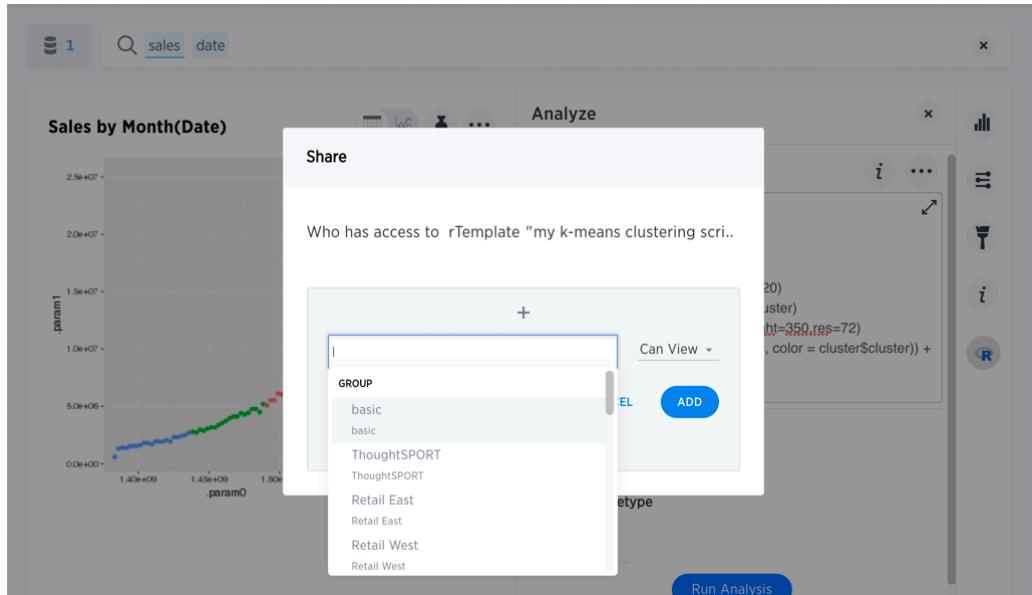
Below the script, there are sections for "Select column(s) for R analysis" (Year(Date) and Sales checked), "Output File Fieltype" (CSV checked), and a "Run Analysis" button.

Save and share your script

To save a script, choose **Save as** from the options menu of the script. After you name and save it, your script will show in the **Load scripts** dialog.

The screenshot shows the ThoughtSpot interface with a scatter plot titled "Sales by Month(Date)". A "Save As" dialog box is open in the foreground, prompting for a script name. The "Name" field contains "my k-means clustering script". In the background, the "Analyze" panel shows the same R script as before, and the "Run Analysis" button is visible.

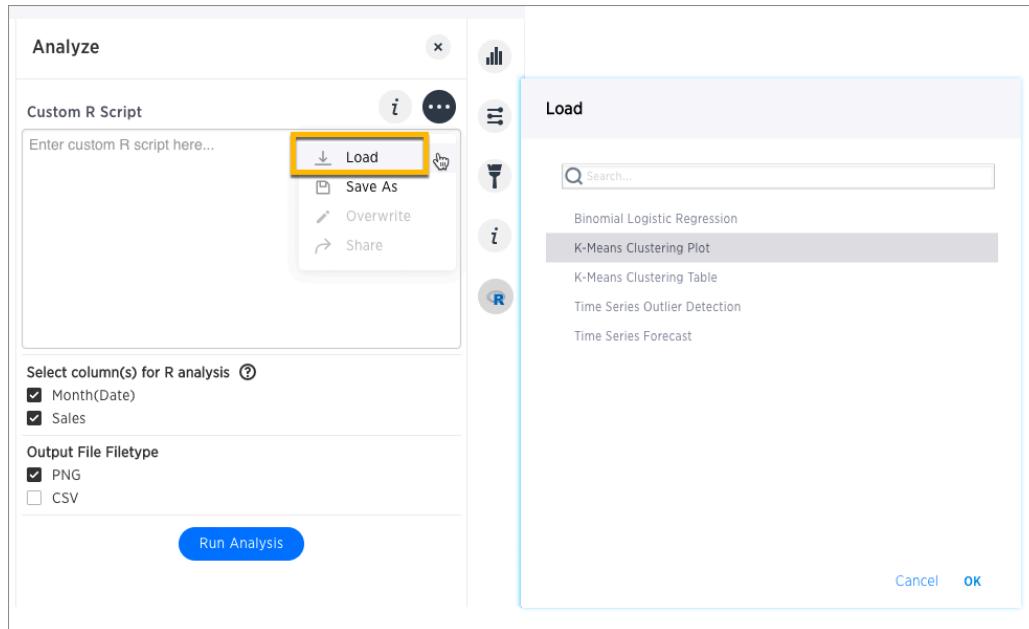
To share a script, choose **Share** from the options menu .



More script examples

For some more example scripts, check out the blog post on [Using R Analysis in ThoughtSpot for Time Series Forecasting](#), or load one of the ThoughtSpot provided scripts:

- Binomial Logistic Regression
- K-Means Clustering Plot
- K-Means Clustering Table
- Time Series Outlier Detection
- Time Series Forecast



Related information

- [Using R Analysis in ThoughtSpot for Time Series Forecasting](#) (blog post by Antony Chen of ThoughtSpot)
- [R Project for Statistical Computing](#)

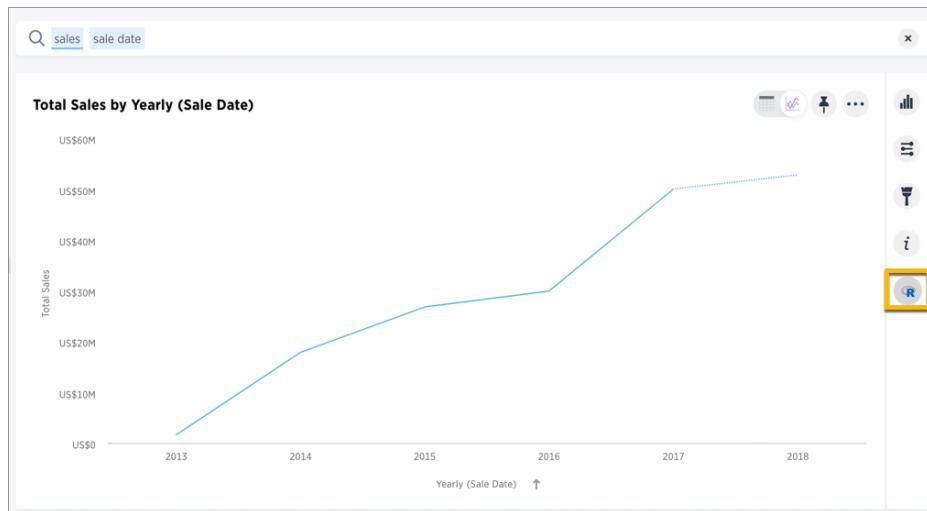
Run prebuilt R scripts on answers

Summary: Anyone with R privileges can run an R analysis ThoughtSpot using provided scripts, you don't need to be an expert.

If you have R privileges on your ThoughtSpot instance, you can run R analyses on search results, and save and share the resulting visualization with others. The users you share visualizations with do not need R privileges.

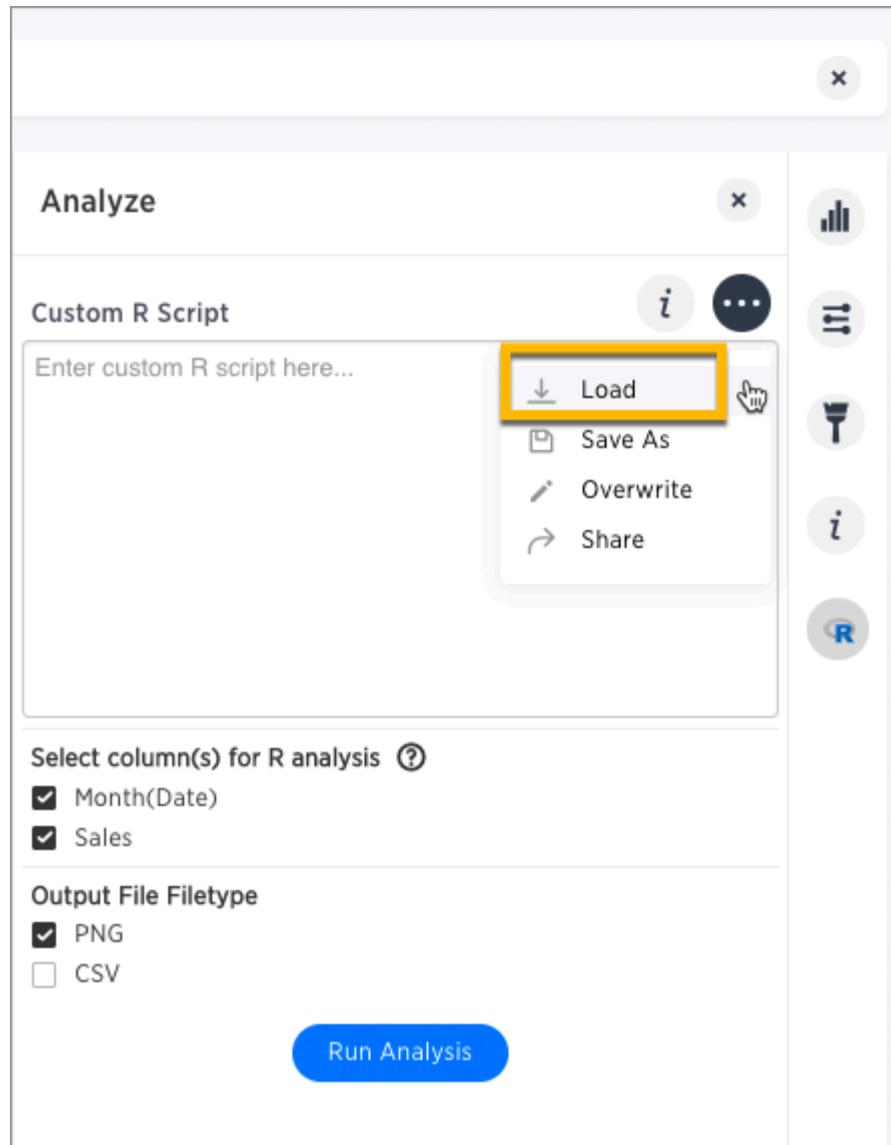
Run an R analysis

1. Click the **R** icon  on the toolbar for any search result (answer).

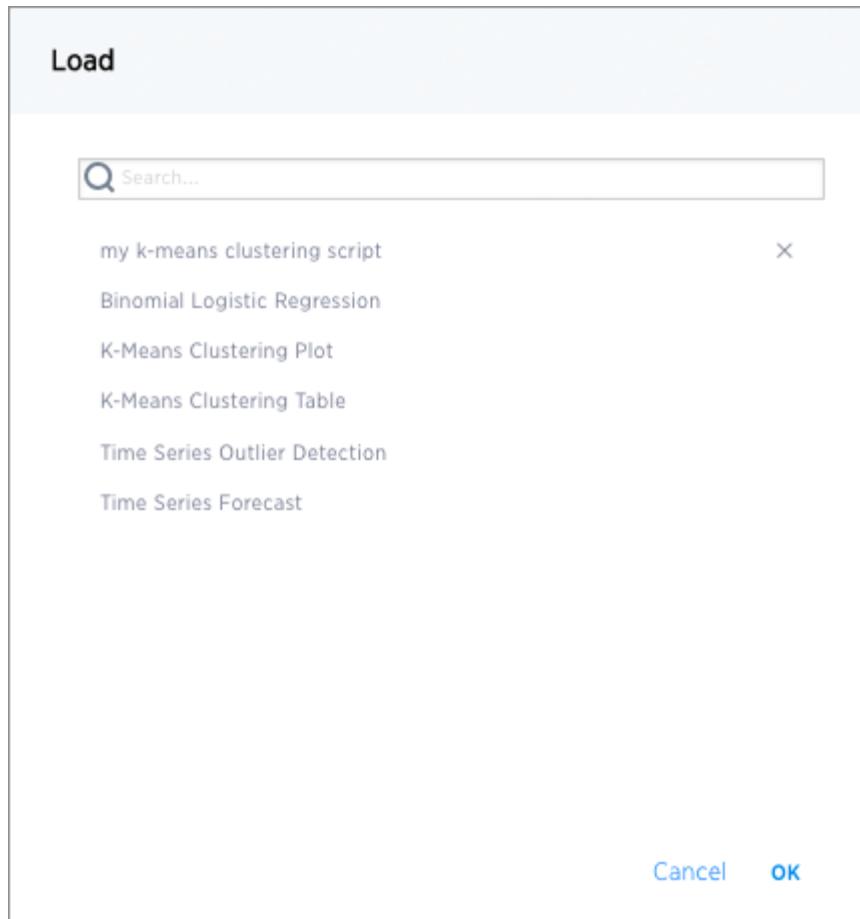


From here, you have options to write a custom script, or load a pre-built or ThoughtSpot provided script.

2. In the Analyze dialog, click the ellipses icon  next to the **Custom R Script** panel, and choose **Load**.



This brings up a list of pre-built scripts, both provided by ThoughtSpot and any created by programmers on your team.



3. Select a script, then choose the columns you want to include in the analysis and the output file type (PNG or CSV).

Note that the output file type must match the script.

For example, if you select one of the ThoughtSpot provided time series scripts, the comment at the top of the script provides guidance on what columns to select.

```
# NOTE: Only two columns are needed for this analysis.  
# The first column should represent some date column  
# and the second should be some numeric measure column.
```

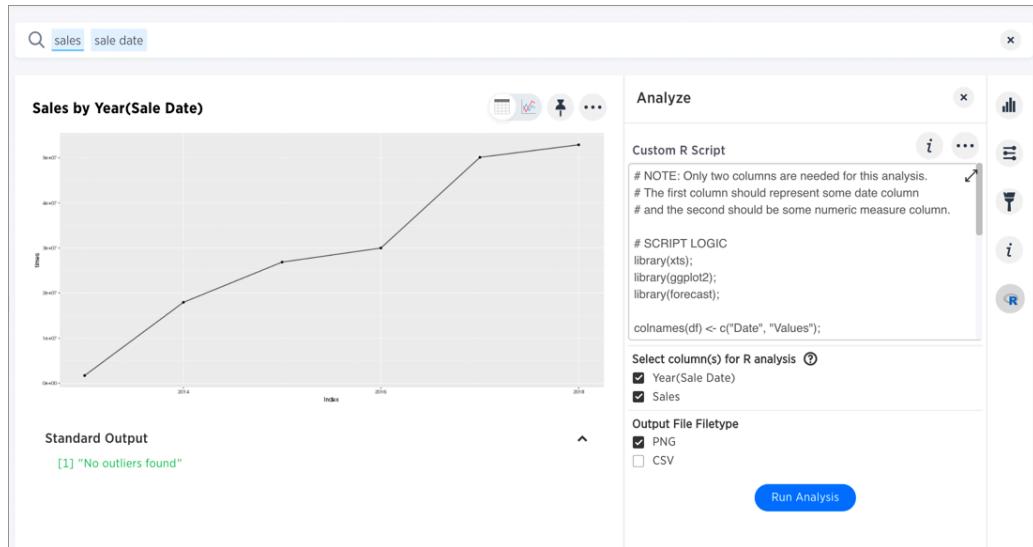
Also, scroll through the script to identify whether it's coded to produce graphical (PNG) or tabular (CSV) output. The time series scripts are both set up to produce graphical output, as indicated by a line like this at the end of the scripts.

```
png(#output_file#, width=1000);
print(img);
```

4. Select **Run Analysis** to execute the script.

Time Series Outlier Example

In this example, we ran an analysis for **Time Series Outlier Detection** on search results that show sales totals by date.

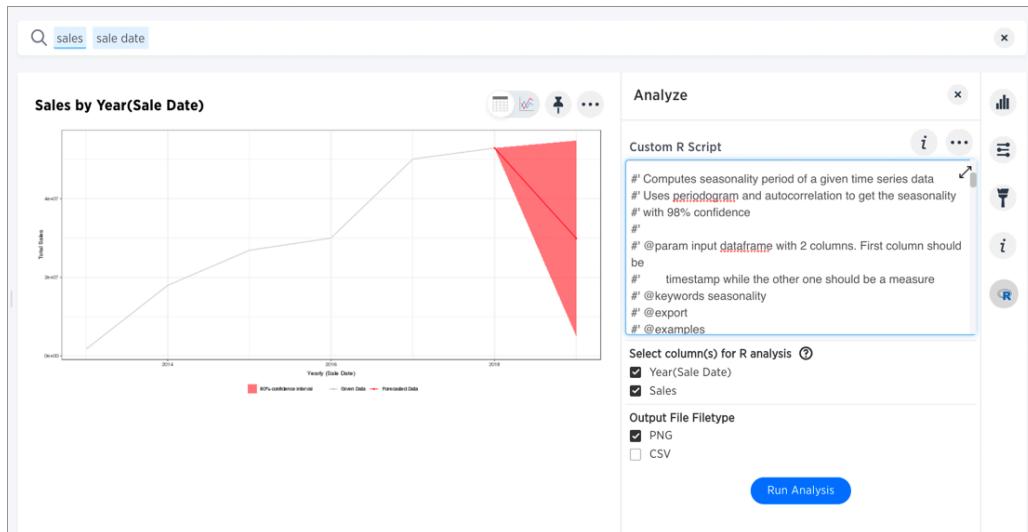


Note that we included a date column and a measure, and selected PNG as the output to match what the script requires. The original search could have had more columns than this, but you can always structure the analysis properly by selecting only the date column and measure column you want to focus on.

In this case, no outliers were found, so the R visualization matches the original search result line graph.

Time Series Forecast Example

In this example, we ran a **Time Series Forecast** on the same search result.



Diverging Bars Example

Here is an example of taking a script found online and repurposing it for a use case in ThoughtSpot.

Antony Chen demo'ded this in a SpotOn webinar. You can find his full presentation on Custom R Scripts and demo at [SpotOn Learning: ThoughtSpot 5.0 BI and Data Science Features](#) in the Community.

Consider this script, found on this website of [Top 50 ggplot2 Visualizations - The Master List \(With Full R Code\)](#). A direct link to this script is [here](#).

```
library(ggplot2)
theme_set(theme_bw())

# Data Prep
data("mtcars") # load data
mtcars$`car name` <- rownames(mtcars) # create new column for
car names
mtcars$mpg_z <- round((mtcars$mpg - mean(mtcars$mpg))/sd(mtcars$mpg), 2) # compute normalized mpg
mtcars$mpg_type <- ifelse(mtcars$mpg_z < 0, "below", "above")
# above / below avg flag
mtcars <- mtcars[order(mtcars$mpg_z), ] # sort
mtcars$`car name` <- factor(mtcars$`car name`, levels = mtcars$`car name`) # convert to factor to retain sorted order in plot.

# Diverging Barcharts
ggplot(mtcars, aes(x=`car name`, y=mpg_z, label=mpg_z)) +
  geom_bar(stat='identity', aes(fill=mpg_type), width=.5) +
  scale_fill_manual(name="Mileage",
                    labels = c("Above Average", "Below Average"),
                    values = c("above"="#00ba38", "below"#f87
66d")) +
  labs(subtitle="Normalised mileage from 'mtcars'", title= "Diverging Bars") +
  coord_flip()
```

You can modify the script above to support the phone sales use case discussed in the webinar.

In this script, `mtcars` is replaced with references to our phone sales (`df$Sales`) and `car name` is replaced with `Device Name` both from the column data in the search example used in the webinar demo. The script uses the ThoughtSpot *data frame* object (`df`), and adds two lines at the end to specify output type as a `png` image.

```
library(ggplot2)
theme_set(theme_bw())

# Data Prep
df$sales_z <- round((df$Sales - mean(df$Sales))/sd(df$Sales),
2) # compute normalized mpg
df$sales_type <- ifelse(df$sales_z < 0, "below", "above") # above / below avg flag
df <- df[order(df$sales_z), ] # sort
df`Device Name` <- factor(df`Device Name`, levels = df`Device Name`) # convert to factor to retain sorted order in plot.

# Diverging Barcharts
img <- ggplot(df, aes(x=`Device Name`, y=sales_z, label=sales_z)) +
  geom_bar(stat='identity', aes(fill=sales_type), width=.5) +
  scale_fill_manual(name="Sales",
                    labels = c("Above Average", "Below Average"),
                    values = c("above"="#00ba38", "below"#f8766d")) +
  labs(subtitle="Normalised Sales for Phones",
       title= "Diverging Bars") +
  coord_flip()
png(#output_file#, width=1000, height=1000)
print(img)
```

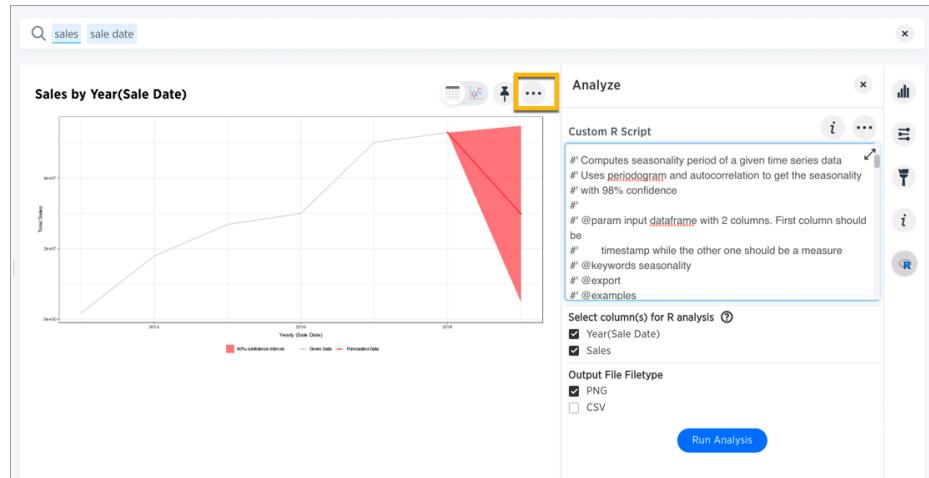
Save and share R visualizations

Summary: You can save and share R analyses as answers and pinboards.

In terms of sharing, the R visualization you get as a result of running a script is similar to any other search result in ThoughtSpot. You can save it, share it, and add it to pinboards.

Save an R visualization

1. Click the ellipses icon  above the generated graphic or table containing your analysis, and select **Save** from the menu.



2. Add a name and description for the analysis, and click **Save**.

Save answer as

Name

Description

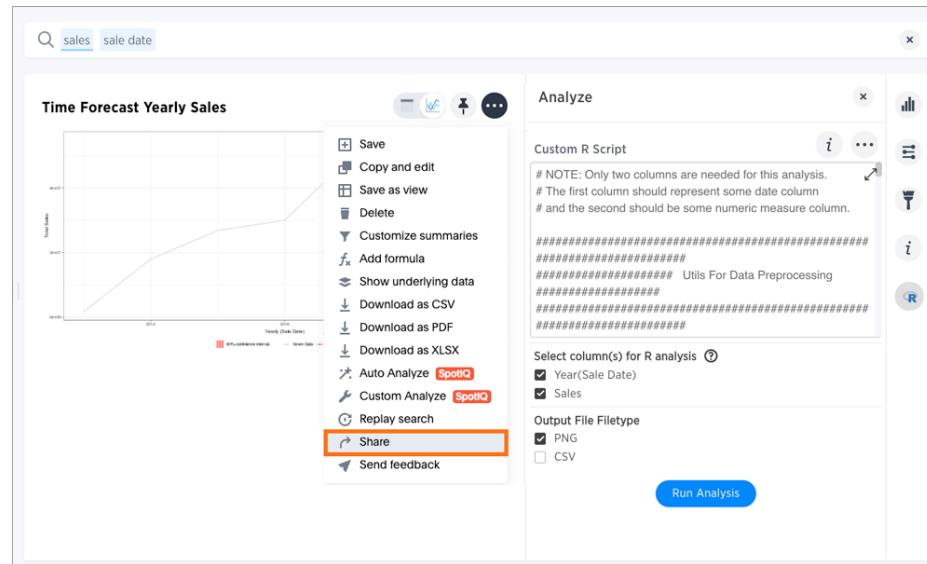
[Cancel](#) [SAVE](#)

The saved analysis is added to **Answers**.

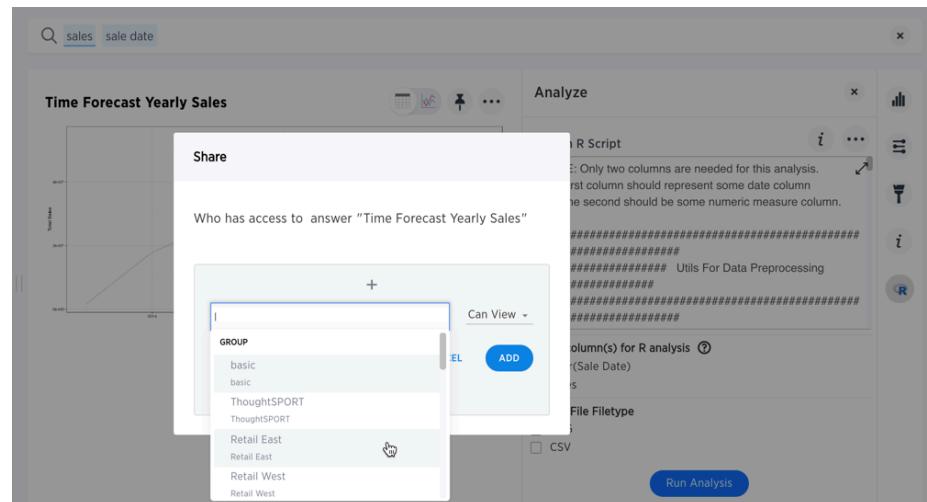
Name	Stickers	Modified	Author
Time Forecast Yearly Sales R analysis time forecast on MarketSpot data		9 seconds ago	 vicky
Monthly Department Sales Analysis		2 weeks ago	 Administrator
Low Inventory		2 weeks ago	 Administrator

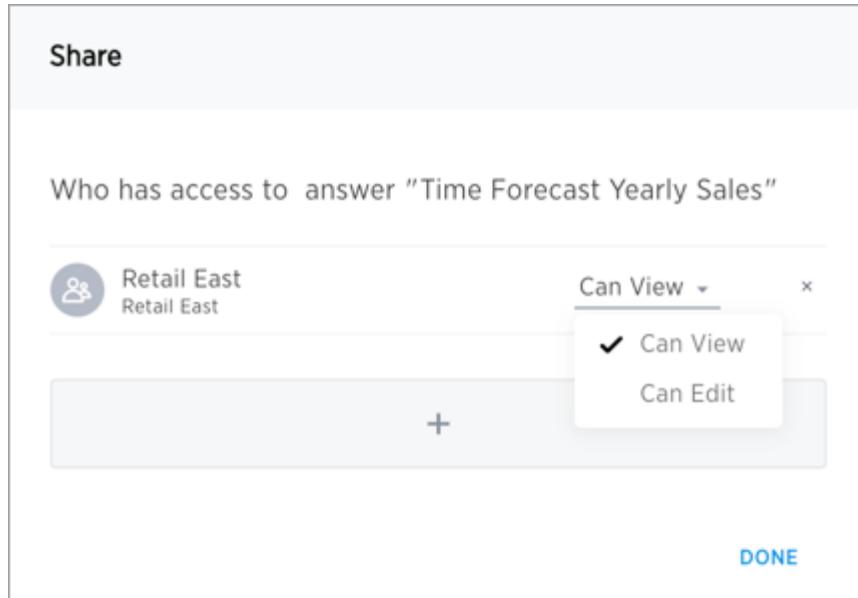
Share an R visualization

1. Click the ellipses icon  above the generated graphic or table containing your analysis, and select **Share** from the menu.



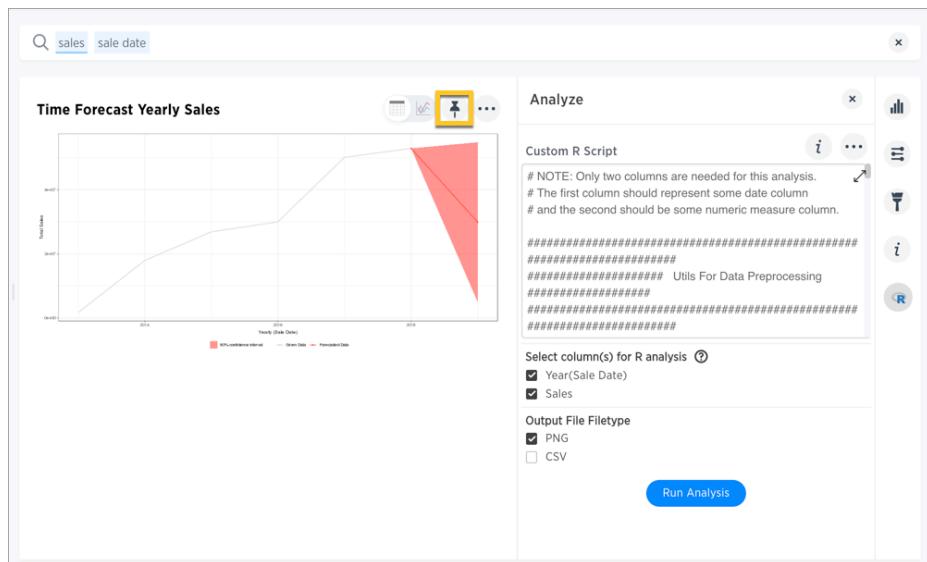
2. Select and add the user or group with which to share, along with level of access (**Can View**, **Can Edit**), and then click **Done**.



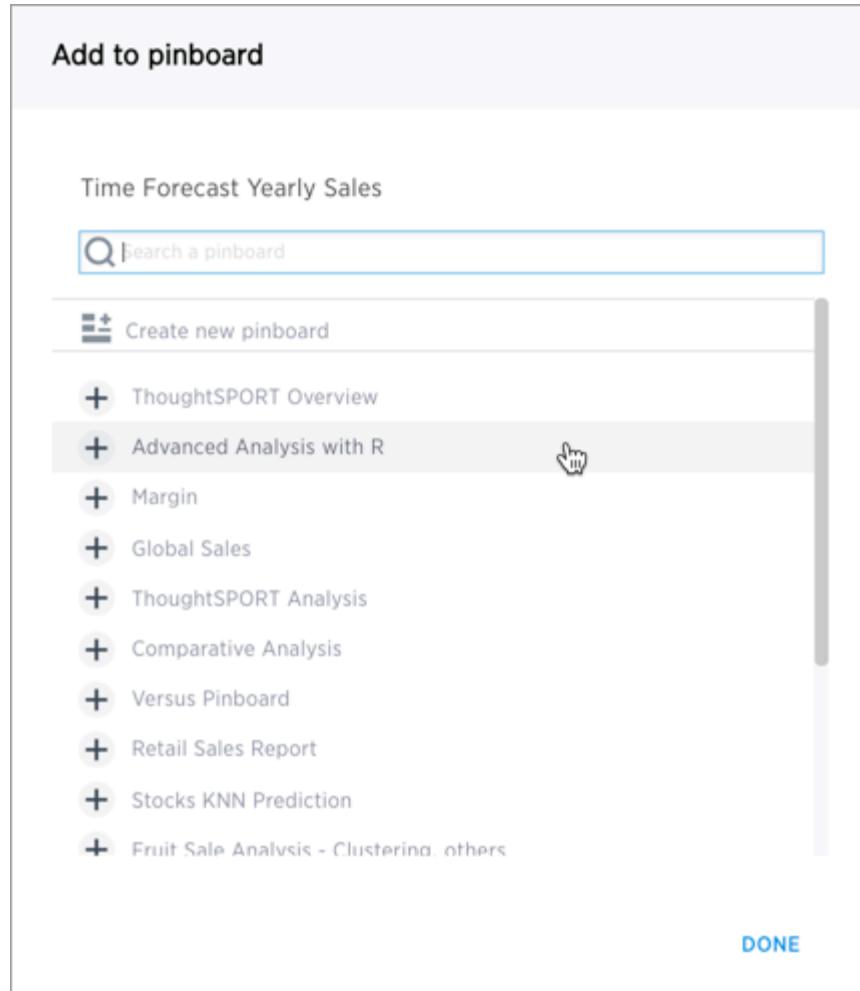


Add an R visualization to a pinboard

1. Click the pin above the generated graphic or table containing your analysis.



2. Create a new pinboard or select an existing pinboard, then click **Done**.



The visualization is available in **Pinboards**.

The screenshot shows a list of pinboards on a platform interface. The top navigation bar includes "Search", "Answers", "Pinboards" (which is the active tab), "SpotIQ", "Data", and "Admin". Below the navigation is a search bar and a filter section for "Stickers" and "Author". The main table lists four pinboards:

Name	Stickers	Modified	Author
Advanced Analysis with R	Sports Goods	39 seconds ago	Administrator
ThoughtSPORT Overview	Sports Goods	1 week ago	Administrator
Margin		2 weeks ago	Administrator
Global Sales		2 weeks ago	Administrator

What is SpotIQ?

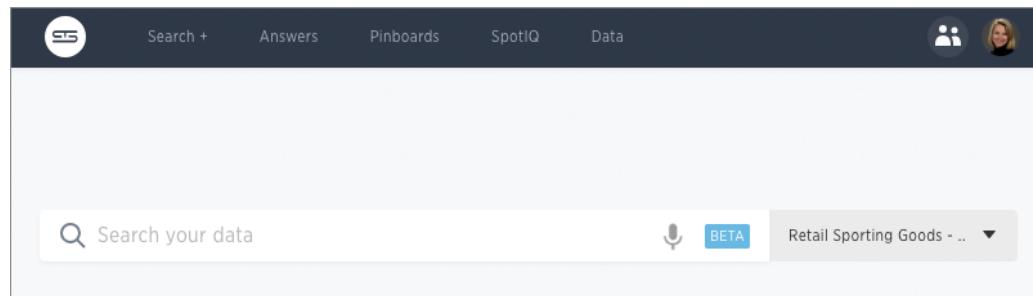
SpotIQ is a ThoughtSpot feature that helps you find insights into your data. Insights are trends, correlations, explanations of increases, explanations of decreases, and outliers (values unique from what is typical in your data).

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:



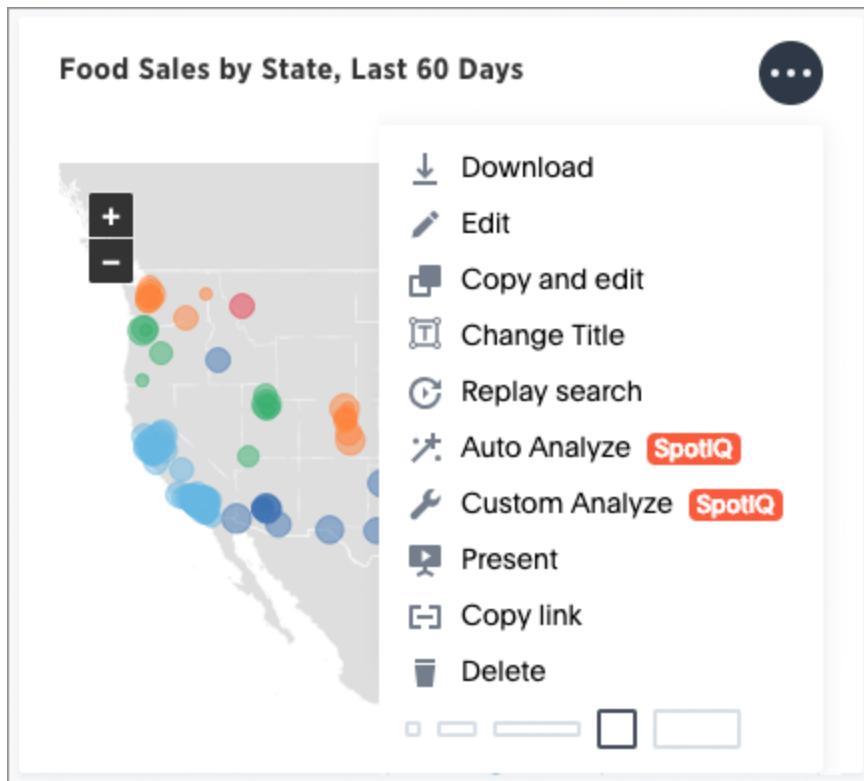
If you can't see this option, ask your ThoughtSpot Administrator to give you the **Has Spot IQ** privilege.

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 label that indicates this:



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 **Data** to get to the data management listing.
2. Click a data source you own or can edit.

This brings up the **Columns** screen, where you can 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 equip 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.

Prioritizing analyses types

You can prioritize highlighting changes in data over time instead of other changes, such as outliers or anomalies.

Columns

When you trigger a SpotIQ analysis on an answer, you can select alternate data columns. To trigger more time-related insights, pick more date-time columns.

Advanced

In the advanced tab of the SpotIQ dialog, increase the maximum number of trend and correlation insights, and reduce the number of anomaly insights.

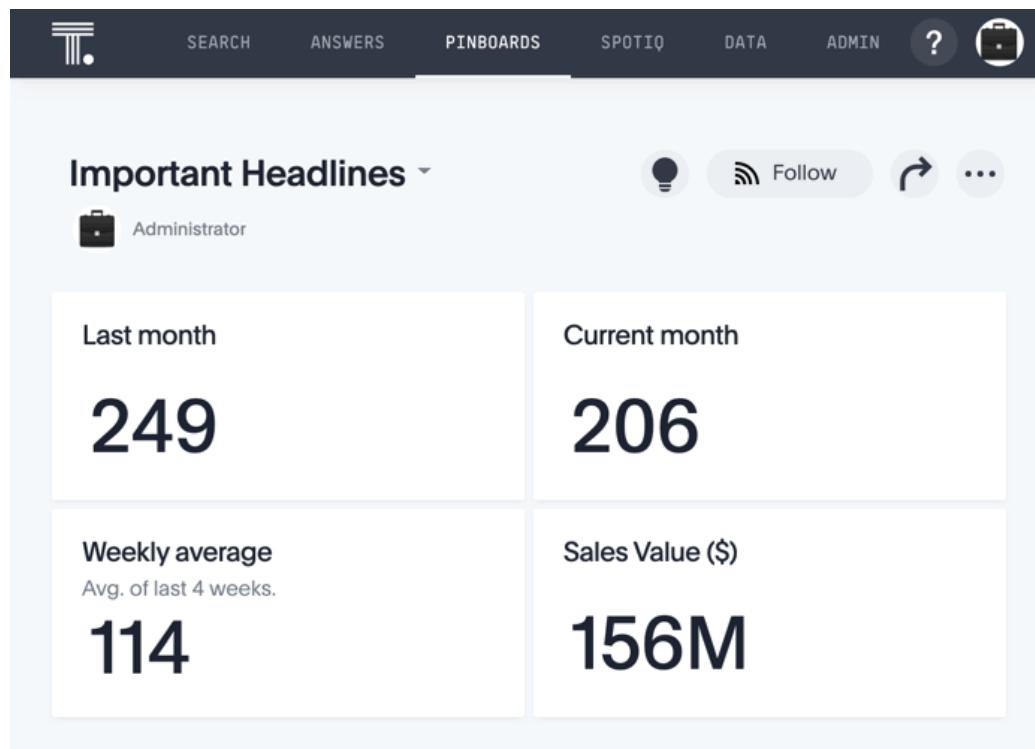
Monitor headlines with SpotIQ

Summary: You can monitor select Pinboards, and see how they change over time.

With SpotIQ Monitor, you can follow headline metrics of your favorite Pinboards, and get daily updates in your mailbox, on your ThoughtSpot home page, or in the SpotIQ Follow management interface.

Follow Headlines

To monitor a Headline, click the **Follow** icon in its top right corner.



The system confirms that you are now following that Headline, and will receive daily email updates.

Notice that the Headline's **Follow** icon has a checkmark.

If the user account is not configured with an email, you must supply it at this time.

Unfollow Headline

To stop monitoring a Headline, follow these steps:

1. Click the **SpotIQ** tab.
2. Click **Follow**.
3. Hover over the action area of the Headline you plan to unfollow.
4. Click the **Unfollow** (trash) icon.
5. Confirm by clicking **Unfollow**.

Name	Change	Last run	Author
Weekly average (Important Headlines) For your first data change, check back tomorrow...	0 %	38 seconds ago	
Sales Value (\$) (Important Headlines) For your first data change, check back tomorrow...	0 %	40 seconds ago	
Current month (Important Headlines) For your first data change, check back tomorrow...	0 %	42 seconds ago	
Last month (Important Headlines) For your first data change, check back tomorrow...	0 %	46 seconds ago	

Rename monitored Headline

To rename a followed Headline, follow these steps:

1. Click the **SpotIQ** tab.
2. Click **Follow**.
3. Hover over the action area of the Headline you plan to unfollow.
4. Click the **Rename** (gear) icon.
5. In the window modal, change the name of the headline.
6. Confirm by clicking **Save**.

Name	Change	Last run	Author
Weekly average (Important Headlines) For your first data change, check back tomorrow...	0 %	38 seconds ago	
Sales Value (\$ (Important Headlines) For your first data change, check back tomorrow...	0 %	40 seconds ago	
Current month (Important Headlines) For your first data change, check back tomorrow...	0 %	42 seconds ago	
Last month (Important Headlines) For your first data change, check back tomorrow...	0 %	46 seconds ago	

Monitoring details

Click each headline in the **SpotIQ » Follow** interface to see the detailed performance over time, examine each interval, all at the time scale you choose. You can change your visualization to table view, or to another chart type to better visualize your changing metric.



Comparative Analysis

Summary: With SpotIQ competitive analysis, you can compare two data points for complex measures.

SpotIQ competitive analysis compares data points of simple or complex measures using these aggregates:

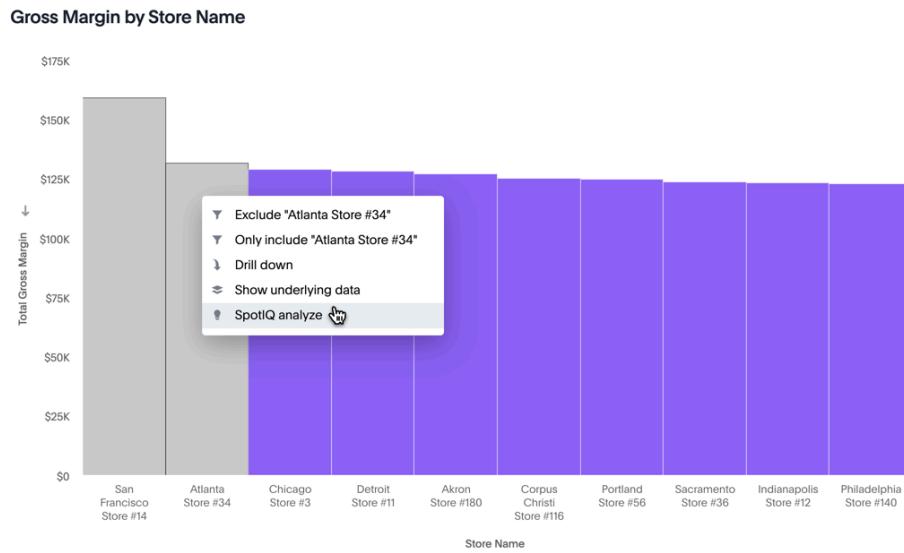
- *Sum*
- *Count*
- *Sum over sum*, which generate a Pinboard that has ‘what-if’ percentage insights
- *Average*, which generate a Pinboard that has ‘what-if’ percentage insights
- Other functions which use a ‘versus’ analysis to show the absolute change grouped by different drill attributes

Basic Comparative Analysis in SpotIQ

To perform this analysis, follow these steps:

1. On a column chart, hold down the command key (Mac OS) or control key (Linux or Windows).
2. Click the two columns you plan to compare. Here, we are comparing the results for the *San Francisco* and *Atlanta* stores.

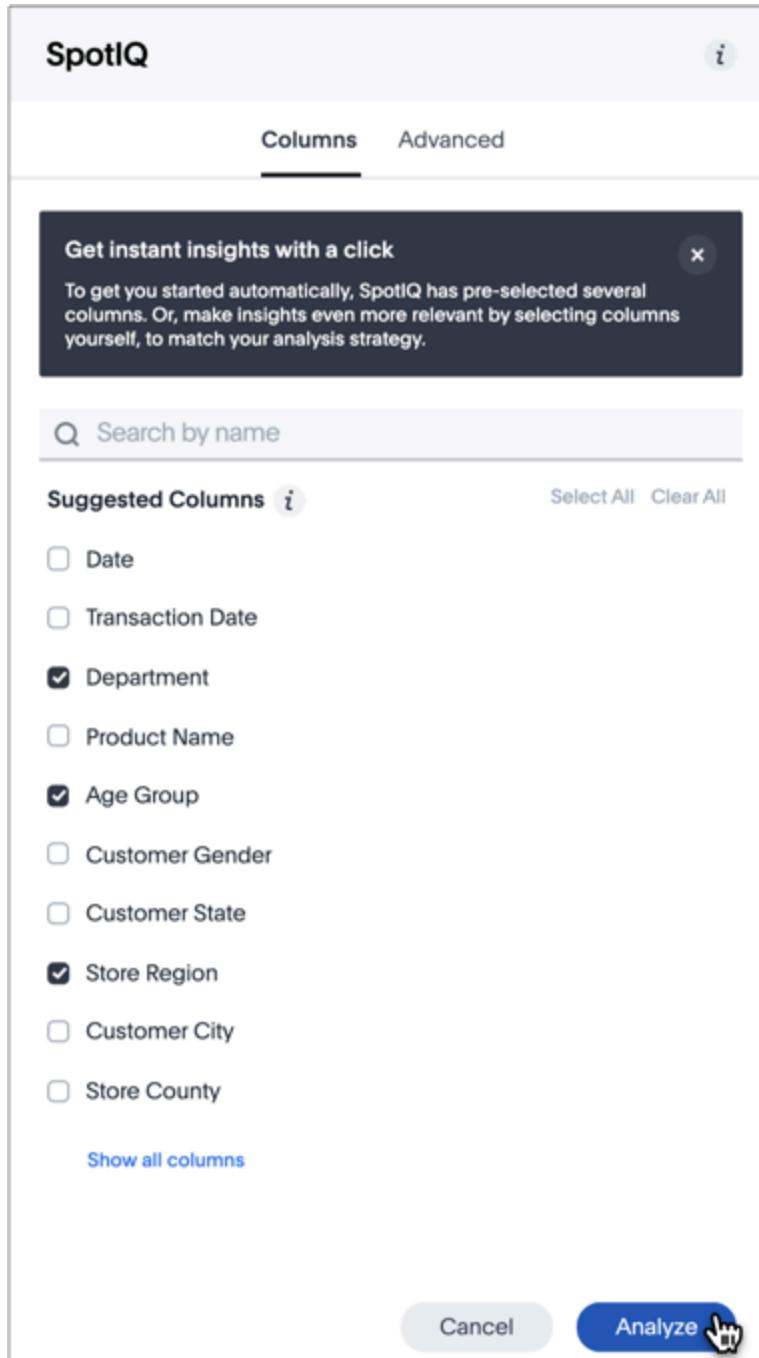
3. Right-click to see the drop-down menu, and select **SpotIQ Analyze**.



4. If you see the the SpotIQ information modal, click **Got it**.
5. In the window modal, under **Columns**, select the columns for SpotIQ to compare. By default, SpotIQ picks the most relevant columns, based on what it learned from your past activity.

Here, we selected *Department*, *Age Group*, and *Store Region*.

Click **Analyze**.



Viewing Results

Notice that ThoughtSpot saves the results of SpotIQ analysis for 24 hours.

To view the results of your analysis, follow these steps:

1. On the top navigation bar, click **SpotIQ**.



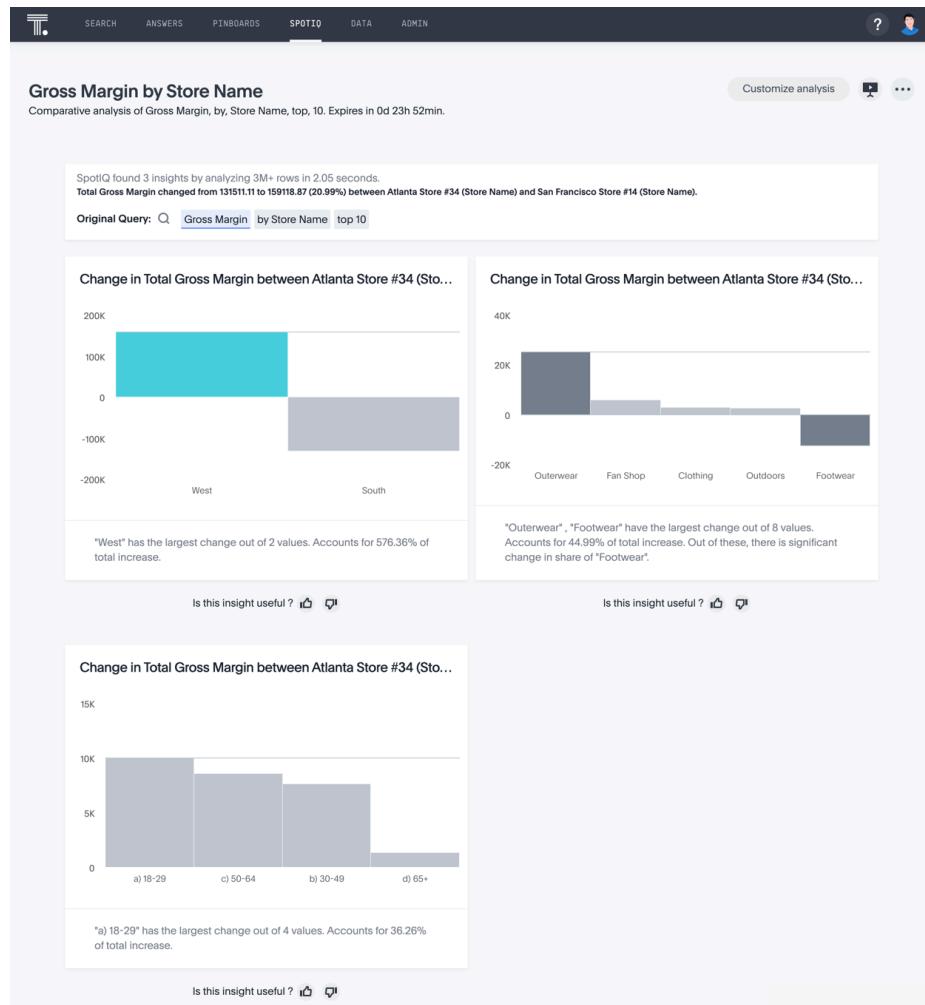
2. Select the **Analyses** tab.

Your analysis may be in progress. Wait until its **Status** changes to *Done*.

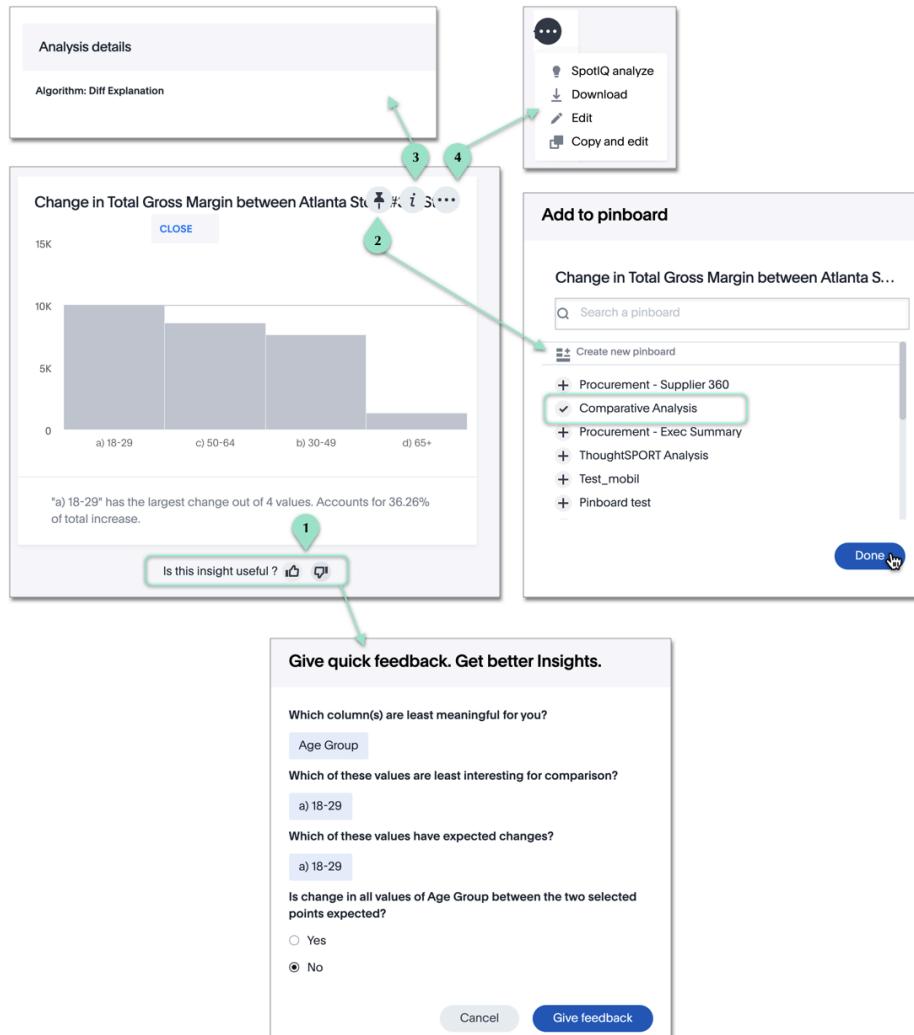
Click *View Results* to open the pinboard.

Name	Status	Modified	Result	Author
Analysis for Gross Margin by Store Name for groups San Francisco Store #14 and Atlanta...	✓ Done	2 minutes ago	Last run succeeded in 10.00 seconds. View Results	Administrator
Analysis for Gross Margin by Store Name for groups San Francisco Store #14 and...	✓ Done	an hour ago	Last run succeeded in 10.00 seconds. View Results	Administrator

3. In the report, SpotIQ generates a Pinboard with insights that compare the two selected points, grouped by the columns you chose for analysis.



4. You can now examine each visualization of the analysis for details of comparison, give feedback on its usefulness, and so on.



Legend Action

- Give feedback.** SpotIQ learns how to improve its feedback based on the information you provide.
- Add to a pinboard.** You can add the specific visualization to an existing pinboard, or create a new pinboard and add it there.
- Analysis details.** Get the name of the algorithm
- More options,** including another round of SpotIQ analysis, download of data as a CSV file, editing this visualization, or editing a copy of the visualization.

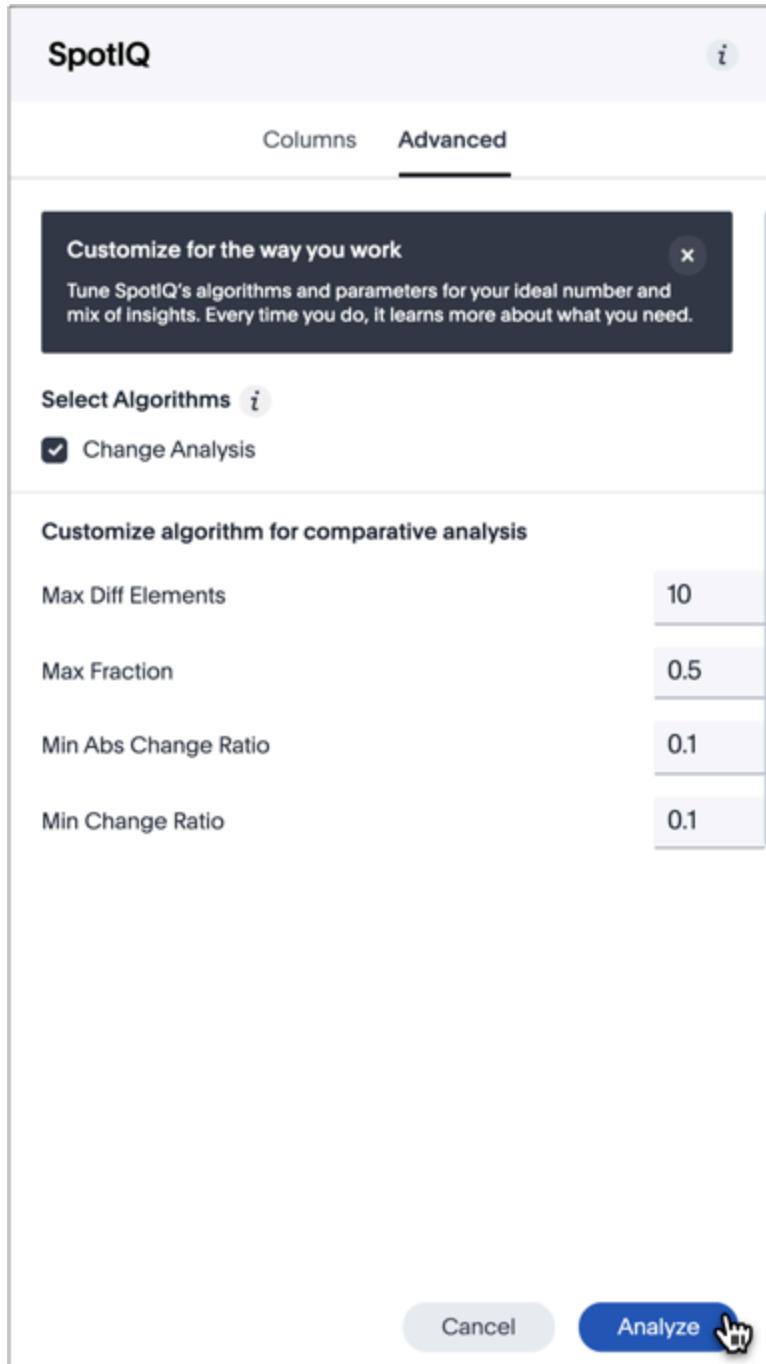
Advanced Comparative Analysis in SpotIQ

ThoughtSpot ships with a number of comparative algorithms. Using the **Advanced** option of SpotIQ, you can adjust the parameters of the analysis, or choose a different comparative algorithm for your data.

After selecting the relevant analysis columns, click the **Advanced** tab.

Under **Select Algorithms**, select the name of the algorithm. You may have several options available, or only a single one. In this example, the only valid option is **Change Analysis**.

Under **Customize algorithm for comparative analysis**, adjust the options. In this example, the algorithm parameters are **Max Diff Elements**, **Max Fraction**, **Min Abs Change Ratio**, and **Min Change Ratio**.



On the SpotIQ Feedback tab, you can see all SpotIQ feedback gave. Insights that you disliked are pushed to the bottom of the list.

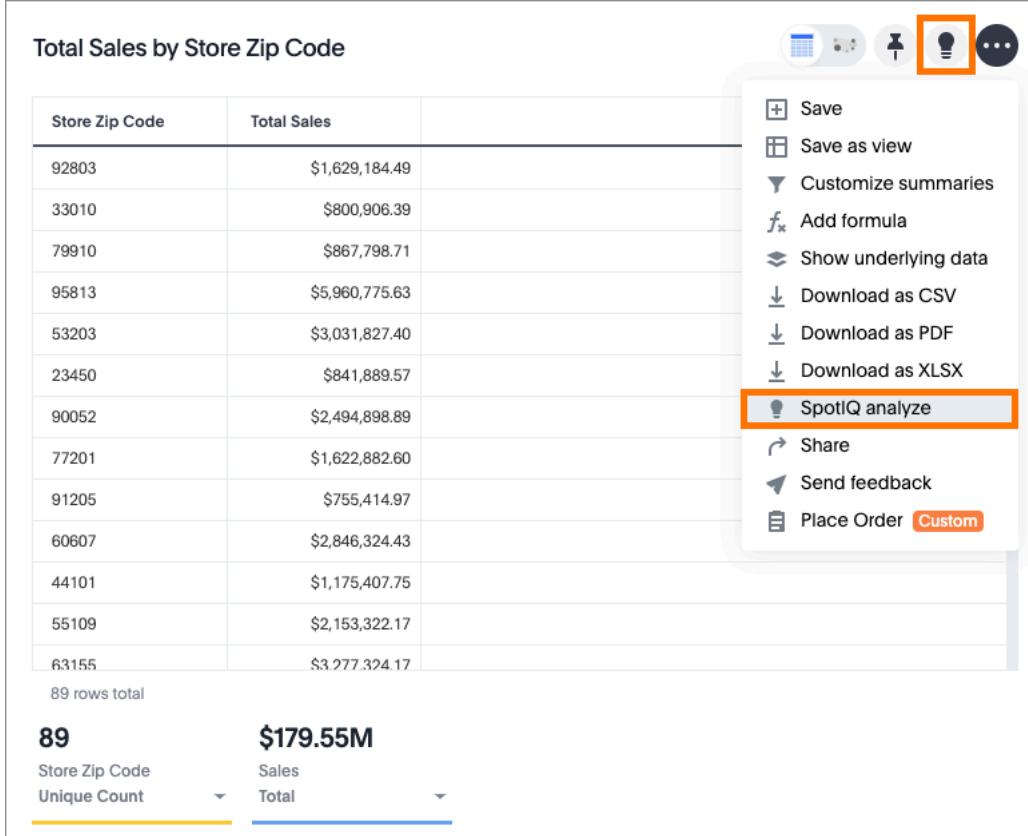
Name	Rating	Feedback	Date	Provided by
Total Sales by Sale Date	1	Trend was expected There is a better date bucket for ...	31 minutes ago	Administrator
Total Sales by Department	1		32 minutes ago	Administrator
Total Sales by Department	1	Sale Date 07/25/2016 was uninte...	32 minutes ago	Administrator
Total Sales by Department	1		32 minutes ago	Administrator
epoch, linear model by Sale Date	1	epoch was uninteresting linear model was uninteresting Correlation between epoch and l...	34 minutes ago	Administrator
Total Sales by Customer County	1		36 minutes ago	Administrator
Total Sales by Product	1		36 minutes ago	Administrator
Total Sales by Product	1		38 minutes ago	Administrator
Total Sales by Product	1	Sale Date 07/25/2016 was uninte...	38 minutes ago	Administrator

Custom SpotIQ analysis

Everywhere in ThoughtSpot where you can run **SpotIQ analyze**, you can 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 run **SpotIQ analyze**, you can customize your analysis. You access it using the lightbulb button  or the ellipses menu .



The screenshot shows a ThoughtSpot dashboard titled "Total Sales by Store Zip Code". The main area displays a table of sales data:

Store Zip Code	Total Sales
92803	\$1,629,184.49
33010	\$800,906.39
79910	\$867,798.71
95813	\$5,960,775.63
53203	\$3,031,827.40
23450	\$841,889.57
90052	\$2,494,898.89
77201	\$1,622,882.60
91205	\$755,414.97
60607	\$2,846,324.43
44101	\$1,175,407.75
55109	\$2,153,322.17
63155	\$3,277,324.17

Below the table, it says "89 rows total".

The summary section at the bottom shows:

- 89** (Store Zip Code Unique Count)
- \$179.55M** (Sales Total)

Below the summary, there are dropdown menus for "Store Zip Code" and "Sales" (with options "Unique Count" and "Total").

A context menu is open on the right side of the dashboard, listing the following options:

- Save
- Save as view
- Customize summaries
- Add formula
- Show underlying data
- Download as CSV
- Download as PDF
- Download as XLSX
- SpotIQ analyze** (highlighted with an orange border)
- Share
- Send feedback
- Place Order **Custom**

Regardless of where you run **SpotIQ analyze** in ThoughtSpot, the **SpotIQ** window has the same layout. The columns displayed, though, are specific to the data from which you run **SpotIQ analyze**.

The screenshot shows the 'Columns' tab of the SpotIQ analysis configuration interface. At the top, there are two tabs: 'Columns' (which is selected) and 'Advanced'. Below the tabs is a search bar labeled 'Search by name' with a magnifying glass icon. Underneath the search bar is a section titled 'Suggested Columns' with an information icon. To the right of this section are 'Select All' and 'Clear All' buttons. A list of columns follows, each with a checked checkbox and a descriptive label:

- Store Name - Sporting Goods Retail Worksheet
- Sales - Sporting Goods Retail Worksheet
- Transaction Date - Sporting Goods Retail Worksheet
- Customer Name - Sporting Goods Retail Worksheet
- Department - Sporting Goods Retail Worksheet
- Store City - Sporting Goods Retail Worksheet
- Net Margin - Sporting Goods Retail Worksheet
- Margin Percentage - Sporting Goods Retail Worksheet
- Customer Age Group - Sporting Goods Retail Worksheet
- Quantity - Sporting Goods Retail Worksheet

At the bottom left is a blue 'Show all columns' link. At the bottom right are 'Cancel' and 'ANALYZE' buttons.

This window has two tabs:

- **Columns**, where you can explicitly indicate which columns to include in a SpotIQ analysis.
(You can select all, some, or none.)

- **Advanced**, where you can run custom analyses and incorporate R scripts into SpotIQ.

The **Advanced** tab has many options. The panel scrolls, so more algorithms are available than appear in the following image

SpotIQ

Columns Advanced

Select Algorithms i

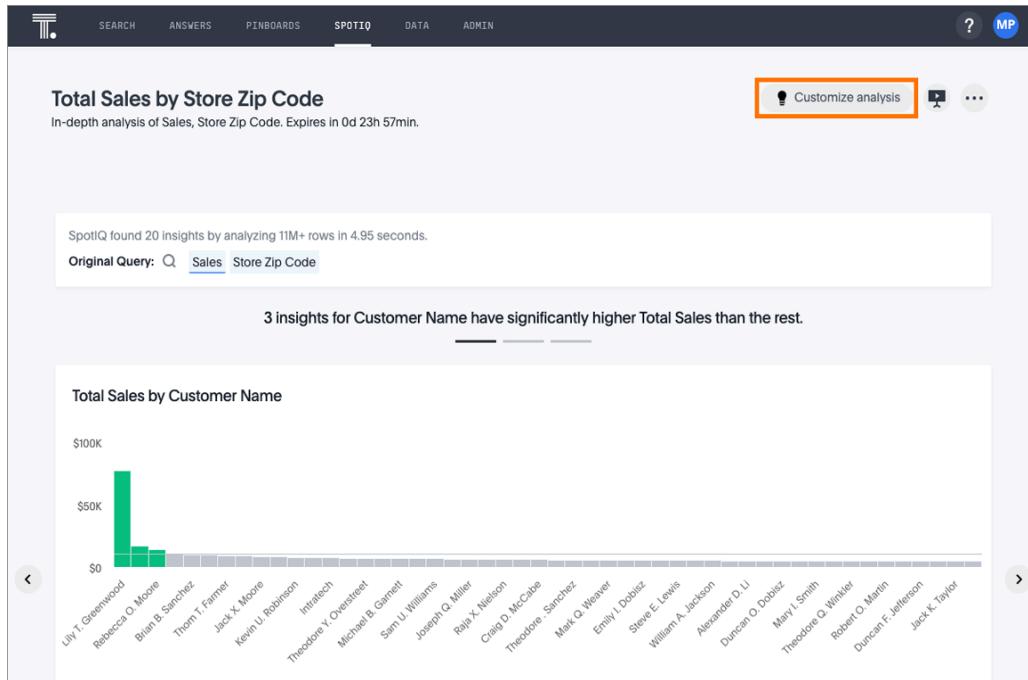
- Outlier Detection using Z-Scores
- Outlier Detection using Median Z-Scores
- Outlier Detection using Seasonal Hybrid ESD
- Outlier Detection using Linear Regression
- Trend Analysis
- Cross-Correlation Analysis
- K-Means Clustering (2-Dimensional)

Refine Parameters

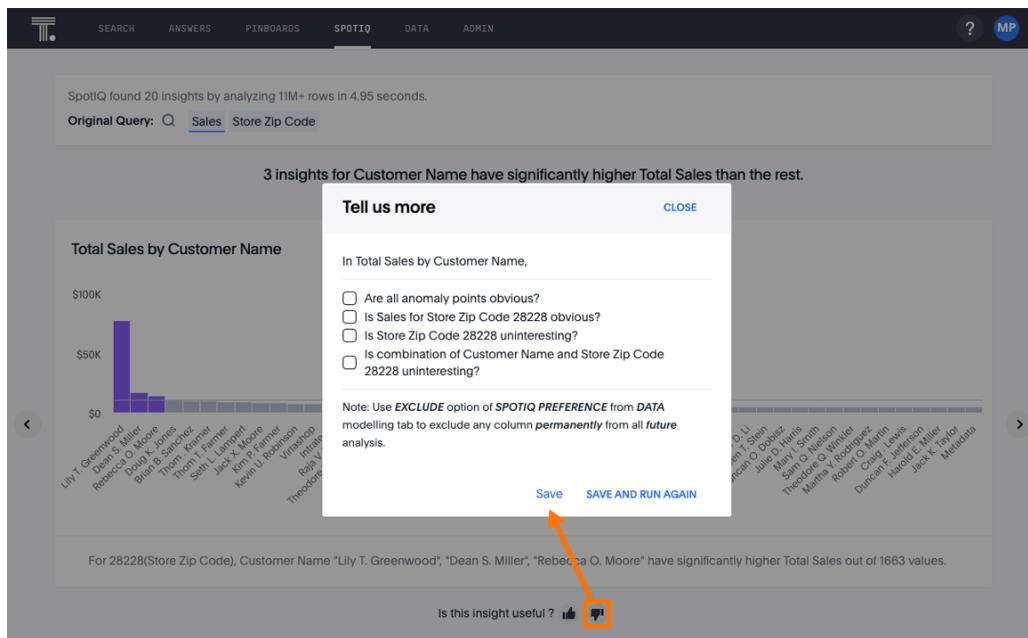
Minimum rows required for analysis	5
Multiplier for Outlier Detection	
Maximum P-Value	0.05
Minimum Correlation Coefficient	0.7
Maximum Correlation Coefficient	0.98
Maximum Correlation Lag	50
Minimum Relative Difference	5

[Cancel](#) [ANALYZE](#)

The **SpotIQ** insights page has an option to open this window to customize a single analysis.



The **Tell us more** window provides information on how to exclude a value from all future analysis.



You can set SpotIQ customization parameters in your [SpotIQ preferences](#), along with options for email notifications when analyses run.

Eliminate null values

Null values should be eliminated from your analysis.

From next analysis

In the Search bar, add a filter `column!= Null` to the search bar. Or choose **Exclude null values from current analysis**, using the **Advanced** panel.

From future analyses

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

Use the **EXCLUDE** option of **SPOTIQ PREFERENCE** under the **DATA** tab to exclude any column permanently from all future analysis.

From future analyses

Include an interesting column

You can always include columns that interest you in your analysis.

From next analysis

Choose **Customize analysis** and select columns that you want to include.

From future analyses

Ensure **Index Priority** is between 8-10 on the column under the **DATA** tab.

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
	Not applicable.

In the Search bar, add a filter `date < last time period` to the search bar.

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 Customize analysis , select the Advanced tab, and decrease the Multiplier for Outlier Detection to a value closer to zero.	Not applicable.

Explaining Anomaly Insights

When SpotIQ finds insights with anomaly points, you have the option to run the Explain Insight command to have SpotIQ do further analysis on the anomaly points.

SEARCH ANSWERS PINBOARDS SPOTIQ DATA ADMIN ? A

Sub-Category "binders" has significantly higher Total Quantity out of 17 values. Score: 0.242285

Is this insight useful?

Total Sales by State

Total Quantity by Product ID

Download Edit Copy and edit SpotIQ analyze Explain insight BETA

State "california", "newyork" have significantly higher Total Sales out of 49 values. Score: 0.161428

Is this insight useful?

Product ID "tec-ac-10003832", "off-pa-10001970", "off-bi-10001524" have significantly higher Total Quantity out of 1862 values. Score: 0.0538096

Is this insight useful?

2 insights for City have significantly higher Total Row ID than the rest. Includes similar insights for Total Quantity.

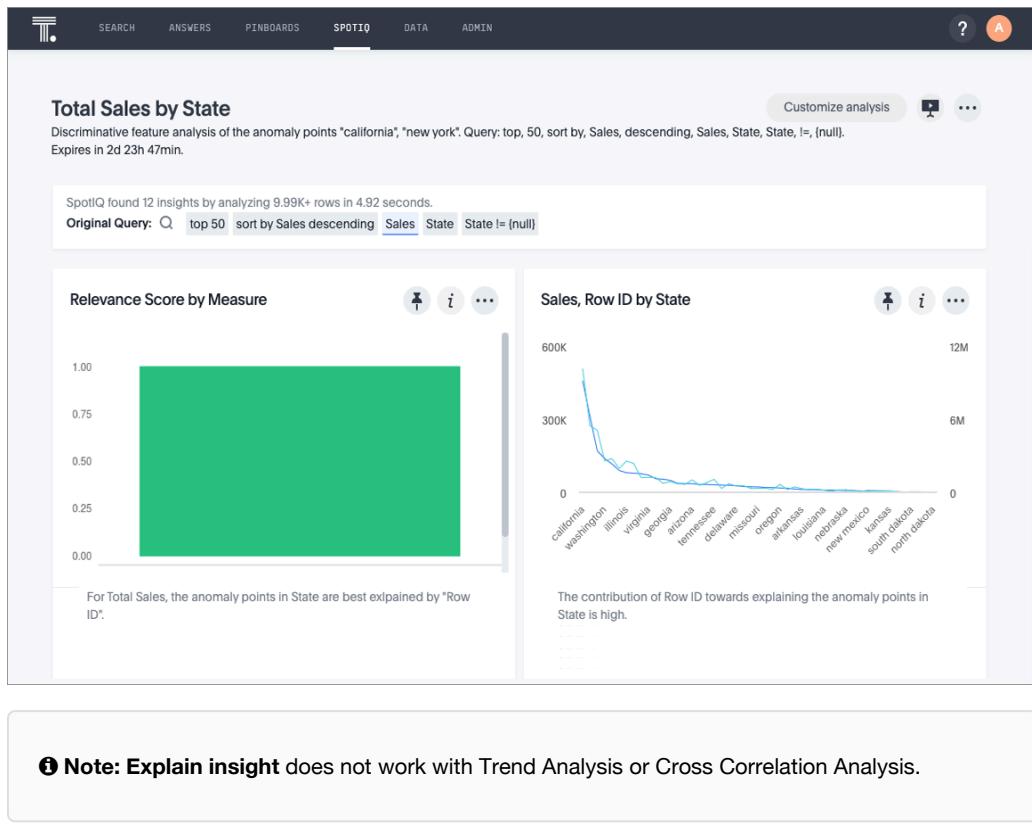
To explain an insight, do the following:

1. Click the ellipses icon  for the insight you want to explain and select **Explain insight**.
 2. In the SpotIQ window, click **Analyze**.

Analysis begins.

3. Click **SpotIQ** in the top navigation bar.
 4. Click the **Analyses** tab.
 5. Find the name of the insight that you analyzed and click **View Results**.

The results of your insight analysis explaining your anomaly points appear.



Advanced R Customizations

Admins and users with the **Has SpotIQ** privilege can run an R script.

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_file#, ...);
```

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. After 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. Sign 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 **Sporting Goods Retail Worksheet** data.
3. Enter `sales store 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 with a custom R analysis titled "Total Sales by Store Zip Code". The analysis displays a table of sales data grouped by store zip code. The table has two columns: "Store Zip Code" and "Total Sales". The data is as follows:

Store Zip Code	Total Sales
92803	\$1,629,184.49
33010	\$800,906.39
79910	\$867,798.71
95813	\$5,960,775.63
53203	\$3,031,827.40
23450	\$841,889.57

4. Click the **View R analysis** icon .

5. 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())
```

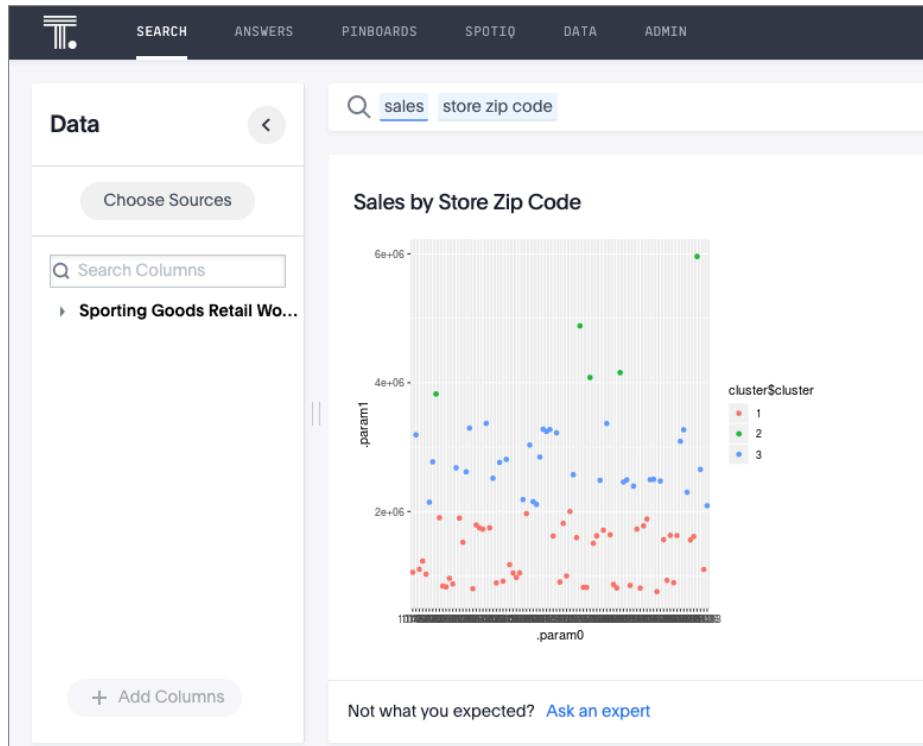
This script binds `.param0` to `Sales` and `.param1` to the `Store Zip Code` column.

You can see from the script that the output should be PNG (`#output_png#`).

6. For Select column(s) for R analysis, make sure that both **Sales** and **Store Zip Code** columns are selected.
7. For Output Filetype, make sure **PNG** is selected as the output format.
8. Click **Run Analysis**.

SpotIQ runs your analysis in the background.

9. When the analysis is completed, you should see the results in PNG format similar to the following:

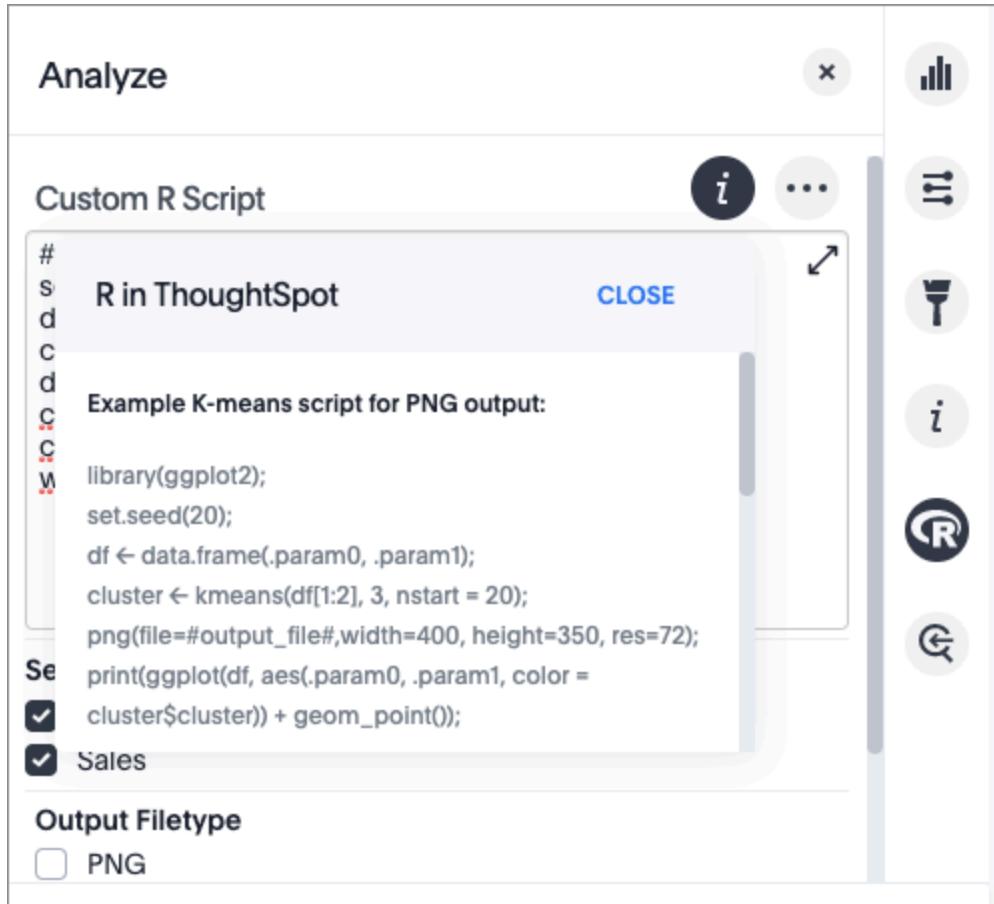


You can run another R script directly on this result to get CSV results. Try this on your own. Here is the script to give you CSV output:

```
####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_file#, row.names=FALSE);
```

Syntax help in the dialog

Use the **i** icon to see help for the R syntax.



Related information

- [tscli rpackage](#)

Insight feedback

Summary: Learn how to rate a SpotIQ insight.

With SpotIQ analysis and insight feedback, you can rate an insight by clicking the like  or dislike  icon of the insight.

When you click dislike, you can provide more details on why you disliked the insight by answering the questions based on the type of insight, decisions made by SpotIQ during analysis, aggregation, and so on. These questions help determine if the features of an insight are relevant, or too obvious.

Give quick feedback. Get better Insights.

Which column(s) are least meaningful for you?

epoch linear model Sale Date

Is the correlation between epoch and linear model obvious?

Yes No

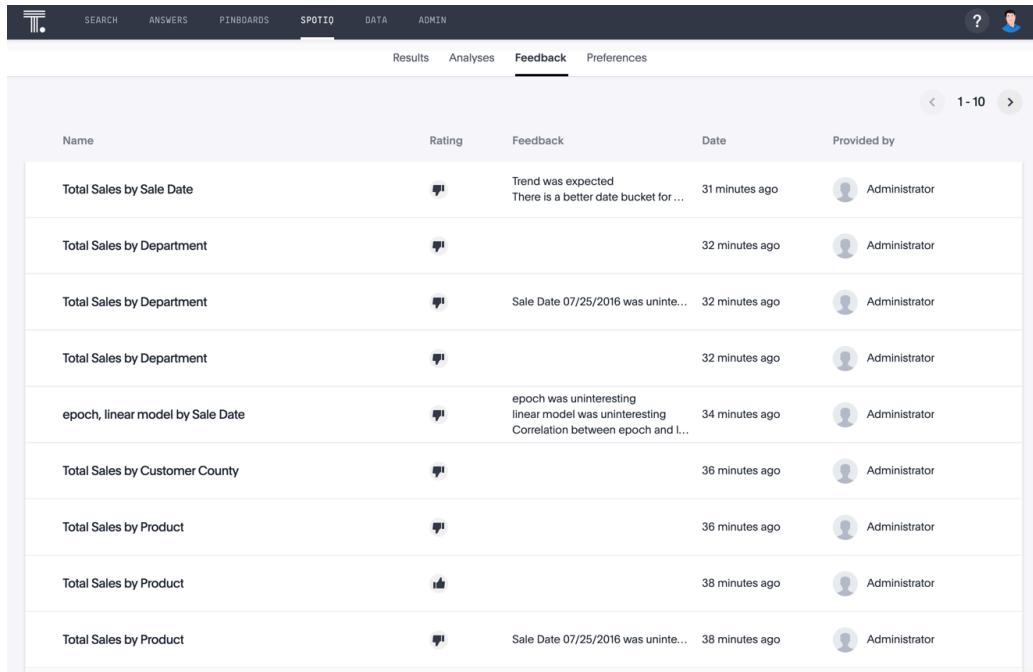
Is there a better date bucket for Sale Date?

Hourly Daily Weekly Monthly Quarterly
 Yearly

[Cancel](#) [Give feedback !\[\]\(d67fd7064711c4984c3f506fa54e7fdb_img.jpg\)](#)

Feedback tab

On the SpotIQ **Feedback** tab, you can see the SpotIQ feedback you provided.



The screenshot shows a list of feedback items on the ThoughtSpot platform. The top navigation bar includes links for SEARCH, ANSWERS, PINBOARDS, SPOTIQ, DATA, and ADMIN. Below the navigation is a secondary menu with links for Results, Analyses, Feedback (which is underlined), and Preferences. A user icon is located in the top right corner. The main content area displays a table with the following columns: Name, Rating, Feedback, Date, and Provided by. There are 10 items listed, each with a small profile icon and the word "Administrator" next to it. The feedback text for the first item is: "Trend was expected. There is a better date bucket for ...". The second item's feedback is: "Sale Date 07/25/2016 was uninte...". The fifth item's feedback is: "epoch was uninteresting linear model was uninteresting Correlation between epoch and l...". The eighth item's feedback is: "Sale Date 07/25/2016 was uninte...".

Name	Rating	Feedback	Date	Provided by
Total Sales by Sale Date	1	Trend was expected There is a better date bucket for ...	31 minutes ago	Administrator
Total Sales by Department	1		32 minutes ago	Administrator
Total Sales by Department	1	Sale Date 07/25/2016 was uninte...	32 minutes ago	Administrator
Total Sales by Department	1		32 minutes ago	Administrator
epoch, linear model by Sale Date	1	epoch was uninteresting linear model was uninteresting Correlation between epoch and l...	34 minutes ago	Administrator
Total Sales by Customer County	1		36 minutes ago	Administrator
Total Sales by Product	1		36 minutes ago	Administrator
Total Sales by Product	1		38 minutes ago	Administrator
Total Sales by Product	1	Sale Date 07/25/2016 was uninte...	38 minutes ago	Administrator

Deleting Feedback

Follow these steps to remove a specific feedback from future consideration.

1. Hover over the feedback you are removing.
2. When a checkbox appears at the beginning of the feedback line, select it.
 - You can select one or more items.
 - To choose all items, hover by the **Name** column, and select the checkbox next to it. This places a checkmark beside each feedback.
 - To unselect items, click the checkbox.
3. Select **Delete** to remove the feedback.

The screenshot shows the ThoughtSpot interface with the 'Feedback' tab selected. A modal window is open, listing three insights. Each insight has a checkbox, a name, a rating icon, a timestamp, and a provided by field.

Name	Rating	Date	Provided by
Change in Total Gross Margin between Atlanta ...	👎	34 minutes ago	Administrator
Change in Total Gross Margin between Atlanta ...	👍	2 hours ago	Administrator
Change in Total Gross Margin between Atlanta ...	👎	Change in all values of St... 2 hours ago	Administrator

SpotIQ preferences

You can specify email notifications behavior along with exclude parameters and threshold settings used for your SpotIQ analyses.

You can change your [SpotIQ](#) basic preferences by checking or unchecking the email and data preferences for analyses.

The screenshot shows the 'Email preferences' section with three checked options: 'Email me analysis notification on success', 'Email me analysis notification on failure', and 'Email me analysis pinboard as attachment'. Below this is a large empty space, likely a placeholder for a map. The 'Data preferences' section follows, containing three options: 'Exclude null values from my analysis' (unchecked), 'Exclude zero measure values from my analysis' (checked), and 'Auto-tune date boundaries for my analysis' (checked).

Email preferences

- Email me analysis notification on success
- Email me analysis notification on failure
- Email me analysis pinboard as attachment

Data preferences

- Exclude null values from my analysis
- Exclude zero measure values from my analysis
- Auto-tune date boundaries for my analysis

Under Parameters for analysis, you can further refine parameters SpotIQ will use.

Parameters for analysis

Maximum P-Value	0.05
Minimum Correlation Coefficient	0.7
Maximum Correlation Coefficient	0.98
Maximum Correlation Lag	50
Minimum Relative Difference	5

These parameters and threshold settings are a subset of those available on the [SpotIQ Customize analysis](#) dialog. If you set them here, the settings will persist for all your SpotIQ analyses including when you run custom analyses, unless you explicitly reset options for a custom analysis.

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		Raw table loaded by an Administrator	Administrators
Worksheet		Collection of related tables optimized for searching (like standard concept of a <i>view</i>)	Anyone
User Imported		Table uploaded by a user through the web browser (e.g. as a .csv file)	Anyone
View		Table saved as a materialized view by a user through the web browser through Save as view option on a search	Anyone

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. Under the 'Tables' section, the 'All types' tab is active, showing a list of data sources. The columns in the table are Name, Source, Stickers, Materialize Status, Modified, and Author. The data includes:

Name	Source	Stickers	Materialize Status	Modified	Author
FoodDollarDataReal				22 hours ago	Plummer
Total Sales by Department, Age Group				a day ago	Andrew Yeung
Top 100 Products Monthly			Materialized	3 weeks ago	Administrator
Sporting Goods Retail Worksheet	ThoughtSPORT worksheet			a month ago	Administrator
West Region Sales	Worksheet with a filter on west region states			a month ago	Administrator
ThoughtSPORT_Product_Dimension				a month ago	AS Administrator Super-User
ThoughtSPORT_Retail_Sales_Fact				a month ago	AS Administrator Super-User
MarketSpot_Vendor_Dimension				a month ago	AS Administrator Super-User

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.

- [Overview of data modeling settings](#)

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 can 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):

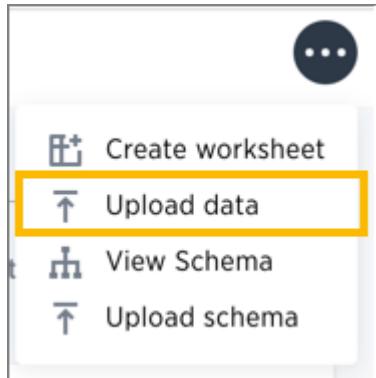
- `NULL`
- `\N`
- `NA`
- `N/A`
- [space]

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:

1. Log into ThoughtSpot from a browser.
2. Click **Data**, on the top navigation bar.
3. Click the ellipses icon  , in the upper right corner, and select **Upload Data**.

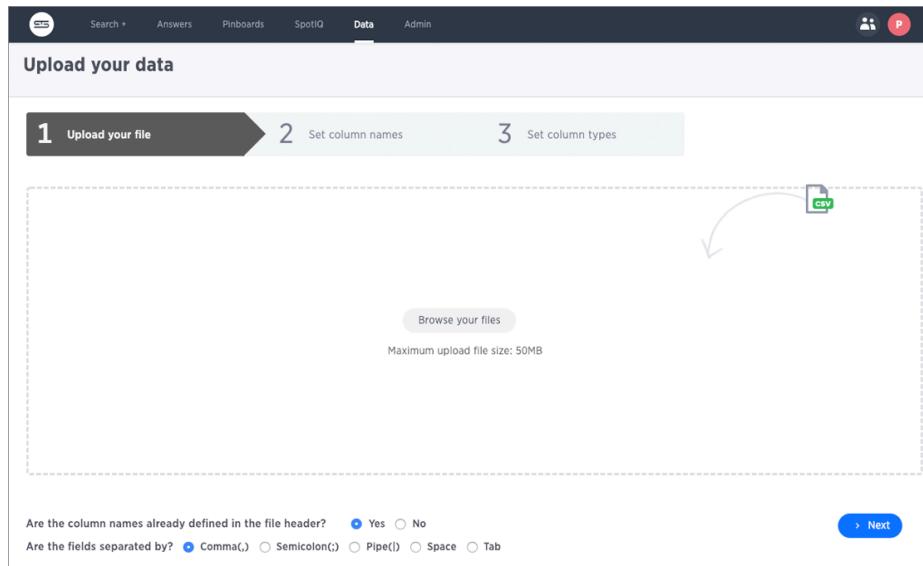


4. Upload the CSV or Excel file by doing one of these options:
 - a. Click **Browse your files** and select the file.
 - b. Drag and drop the file into the drop area.
5. Answer the question **Are the column names already defined in the file header?**
6. Answer the question **Are the fields separated by?** Click **Next**.
7. Click the column header names to change them to more useful names, if you'd like. Click **Next**.
8. 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 `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 **Data** on the top navigation bar.



3. Click the name of the table you would like to append data to.
4. Click the **Load Data** button.

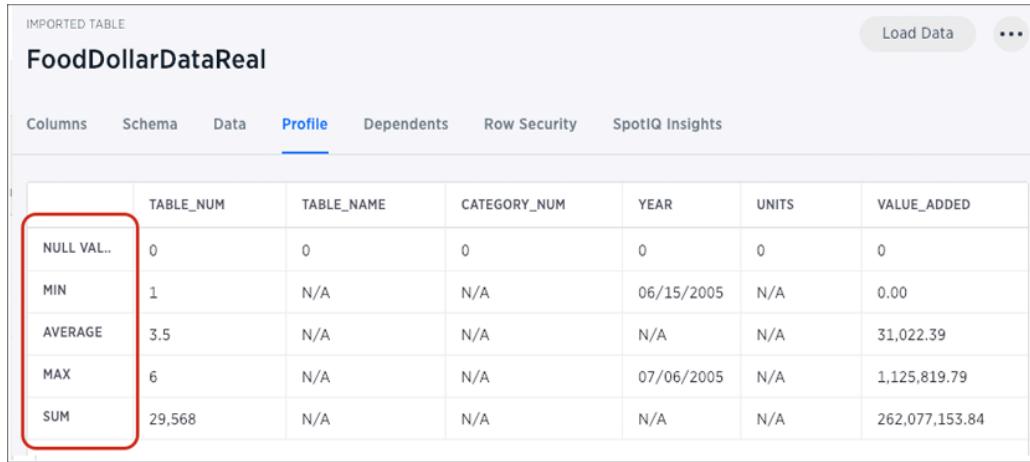
A screenshot of the ThoughtSpot interface showing the details for the 'ThoughtSPORT_Product_Dimension' table. The table has five columns: Product_Key, Product_Name, SKU_Number, Department_Desc., and Category. The 'Load Data' button is highlighted with a red box.

5. Upload the CSV or Excel file by doing one of these options:
 - Click **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. For the question **Do you want to append to the existing data or overwrite it?**, select **Append**.

8. Answer the question **Are the fields separated by?**, and click **Next**.
9. Click **Upload**.
10. Click **Link to existing data** if you want to link the data you uploaded to the data in another table or worksheet. Or click **Ask a question** 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. Click **Data** in the top navigation bar and select your table. Then click **Profile**.



	TABLE_NUM	TABLE_NAME	CATEGORY_NUM	YEAR	UNITS	VALUE_ADDED
NULL VAL..	0	0	0	0	0	0
MIN	1	N/A	N/A	06/15/2005	N/A	0.00
AVERAGE	3.5	N/A	N/A	N/A	N/A	31,022.39
MAX	6	N/A	N/A	07/06/2005	N/A	1,125,819.79
SUM	29,568	N/A	N/A	N/A	N/A	262,077,153.84

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)

By default, the language that ThoughtSpot UI displays depends on the system locale. It is simple to change it using the **Profile** interface.

The **Language** selection specifies more than just the language: it sets the locale, which controls both the language choice and standard data formats for date and number. So, if you set French as the default locale in your profile settings, the interface updates to reflect this. Be sure to refresh your browser page.

For example, in the United States the number format for large numbers uses the comma thousands separator and a period decimals separator, and looks like this: `xxx,xxx.xx`. In most European countries, they use the reverse notation, with comma decimals separator and period thousands separator, like this: `xxx.xxx,xx`.

In addition to American English (`en-US`), ThoughtSpot supports the following locales:

Locale	Language
<code>da-DK</code>	Dansk
<code>de-DE</code>	Deutsche
<code>en-AU</code>	English (Australia)
<code>en-CA</code>	English (Canada)
<code>en-GB</code>	English (United Kingdom)
<code>en-US</code>	English (United States)
<code>es-US</code>	Español (latín)
<code>es-ES</code>	Español (España)
<code>fr-CA</code>	Français (Canada)
<code>fr-FR</code>	Français (France)
<code>it-IT</code>	Italiano
<code>nl-NL</code>	Nederland (beta)
<code>nb-NO</code>	Norsk

Locale	Language
<i>pt-BR</i>	Português (Brazil)
<i>pt-PT</i>	Português (Portugal)
<i>fi-FI</i>	Suomi
<i>sv-SE</i>	Svenska
<i>zh-CN</i>	中文(简体)
<i>ja-JP</i>	日本語

ThoughtSpot translates keywords, operators, and error messages. See the [keyword reference for all supported languages](#).

ThoughtSpot *DOES NOT* translate formulas, or metadata entered by the user. For example, if you name a visualization ‘Quarterly Sales’ in any variant of English and subsequently change the locale to a variant of French, the visualization remains ‘Quarterly Sales’ and does not become ‘Ventes trimestrielles’.

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 Share uploaded data
Pinboards	A pinboard of saved search results.	Anyone who can view a pinboard can share it. See share a pinboard
Answers	The result of a single search.	Anyone who can view an answer can share it. See share answers

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

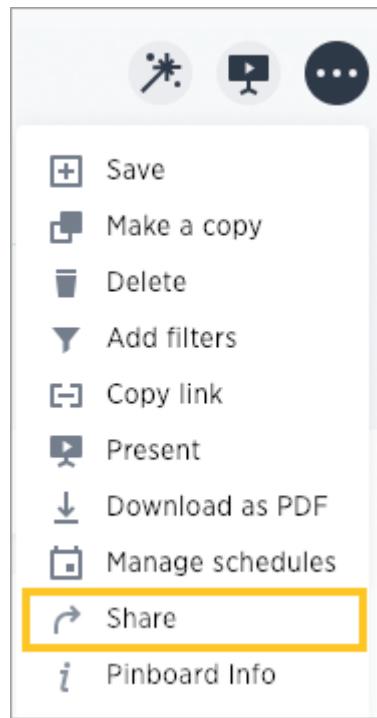
Summary: Whenever you view a pinboard you have the option of sharing it with others.

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.

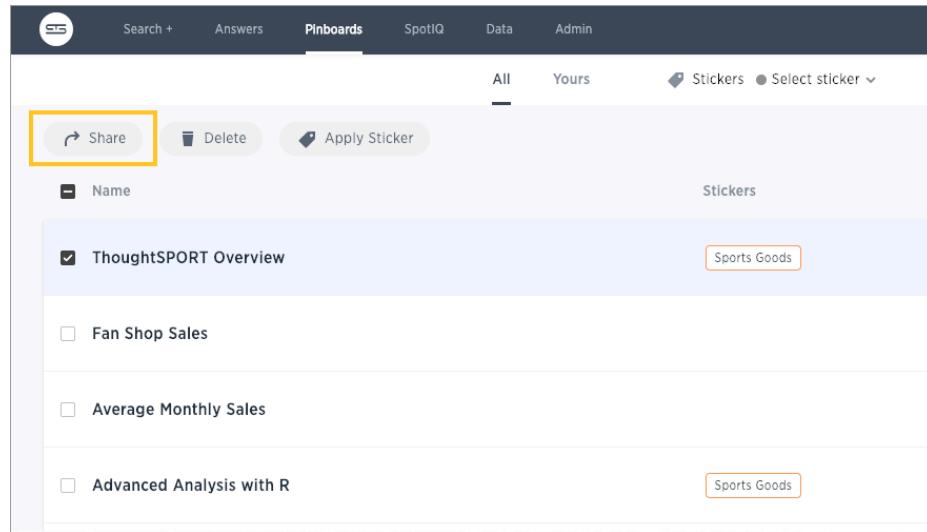
To share a pinboard:

1. Configure it to look as it must appear when shared.
2. From within a pinboard, click the ellipses icon  , and select **Share**.



Alternatively, select the pinboard you want to share from the list of pinboards and click **Share**.

(The profile picture or avatar for the owner of each pinboard is shown in the list.)



3. Click the plus (+) at the bottom of the Share dialog, and select users or groups with whom you want to share.

The screenshot shows the 'Share' interface for a pinboard titled 'ThoughtSPORT Overview'. It lists six users and their current access level:

User	Access Level
marco	Can Edit
victoria	Can View
ThoughtSPORT	Can View
basic	Can View
nathan	Can Edit
vicky	Can View

A modal window is open at the bottom, allowing a new user to be added. The input field contains 'antony' and the dropdown shows 'Can View'. The modal includes 'CANCEL' and 'ADD' buttons.

4. Configure the level of access by selecting from the dropdown next to each user or group.

Available options are based on your own access level. For example, if you have only **View** access, you will not have an option to 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** to save your changes, then click **Done**.

The screenshot shows the 'Share' interface for a pinboard titled 'ThoughtSPORT Overview'. It lists six users and their access levels:

User	Access Level
victoria	Can View
ThoughtSPORT	Can View
basic	Can View
nathan	Can Edit
vicky	Can View
antony	Can View

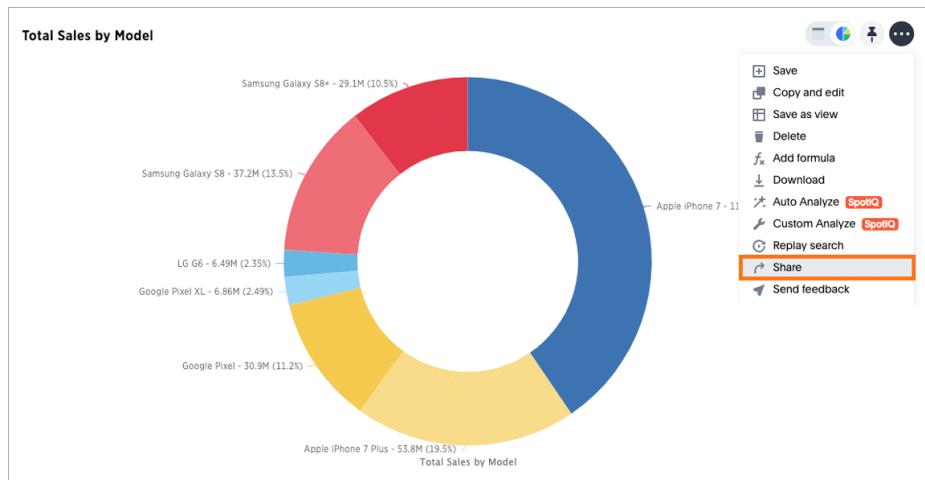
A '+' button is available to add more users, and a 'DONE' button is at the bottom right.

Share answers

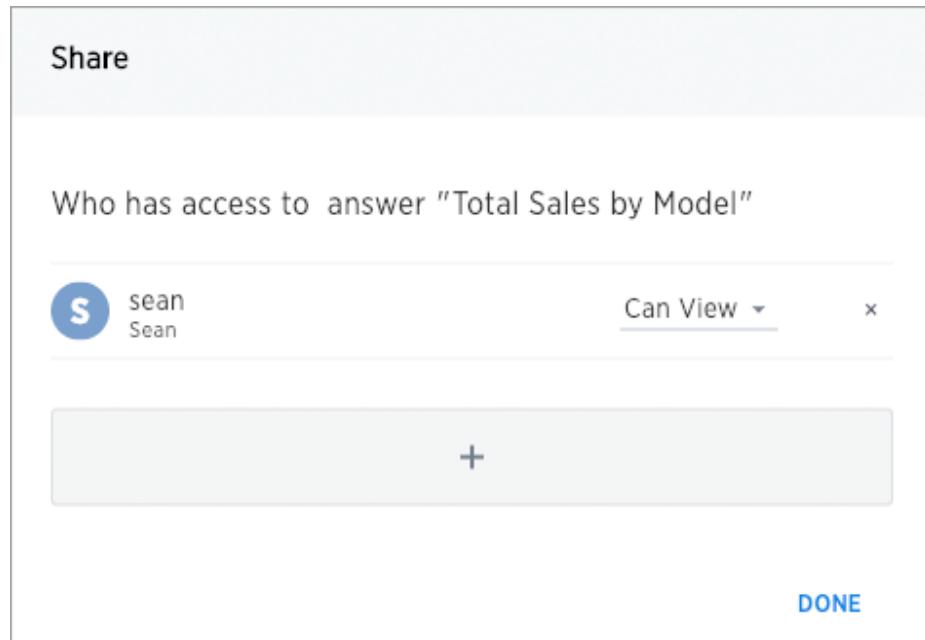
Summary: You do not have to be an administrator or the owner to share saved answers. Any user can share them, based on the access levels the user has.

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.

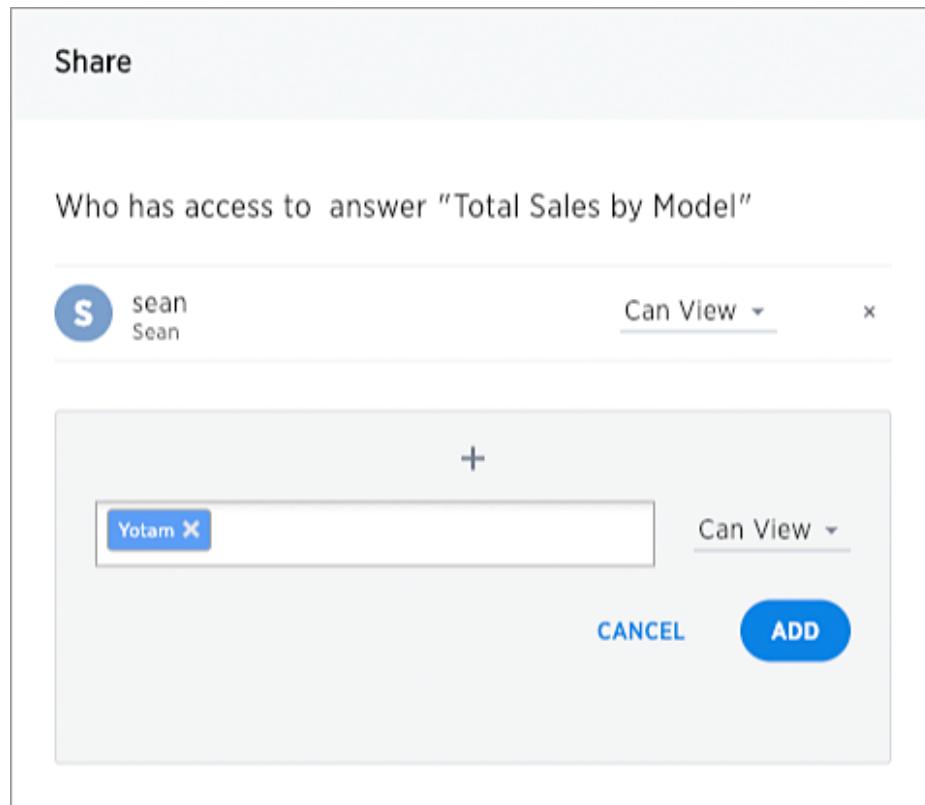
1. Configure the answer to look exactly like it must appear when shared.
2. Save the answer by clicking the ellipses icon  , and selecting **Save**.
3. Share the answer by clicking the ellipses icon  , and selecting **Share**.



4. Click the plus (+) at the bottom of the Share dialog and select users or groups with whom you want to share this answer.
5. Configure the level of access by selecting from the drop-down 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**.



7. Click **Done**.

Share

Who has access to answer "Total Sales by Model"

 sean Sean	Can View 	
 Yotam Yotam	Can View 	
		

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.

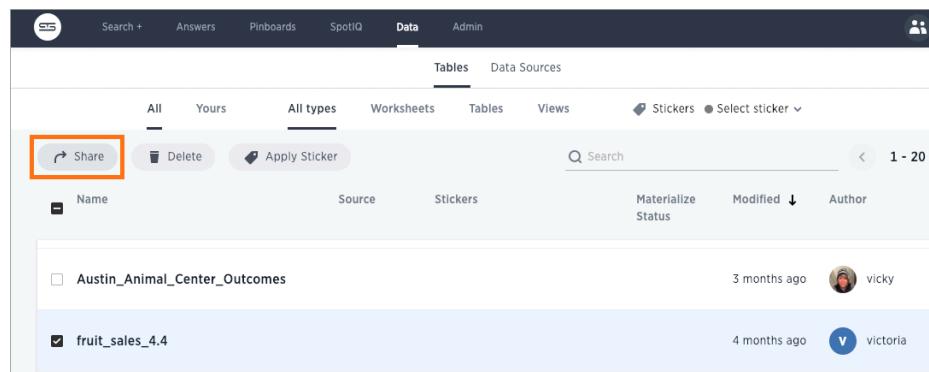
By default, if you share only some table columns, users with access to those columns cannot see the other column's data in their searches. However, the data can become visible if a worksheet or pinboard that *also contain* those columns is shared with these users.

If you want to prevent shares of these types from also revealing the columns, you can ask your administrator to enable a stricter behavior.

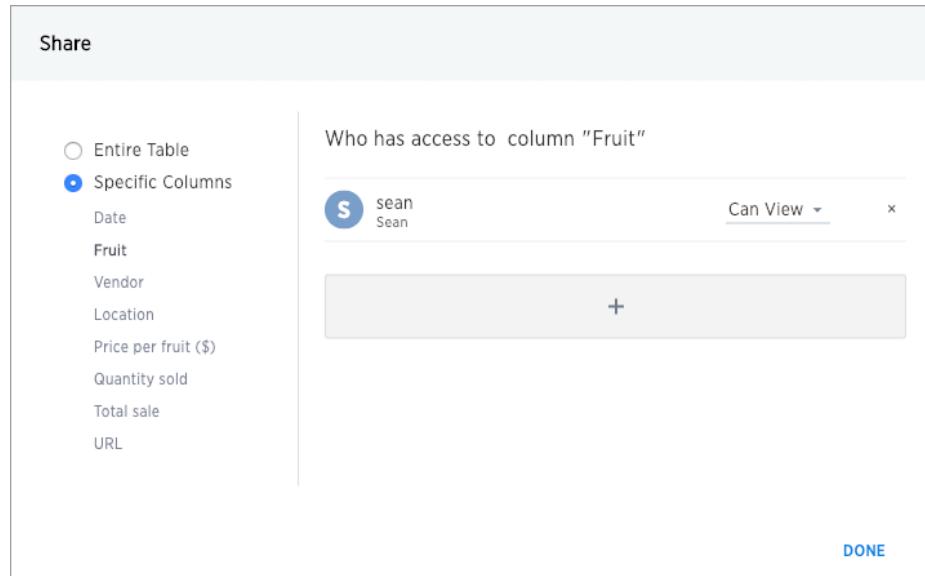
How to share data

Share uploaded data by following these steps:

1. Click **Data** on the top navigation bar.
2. Click the name of the uploaded data you want to share.
3. Click the **Share** icon.



4. Select **Entire Table or Specific Columns**.



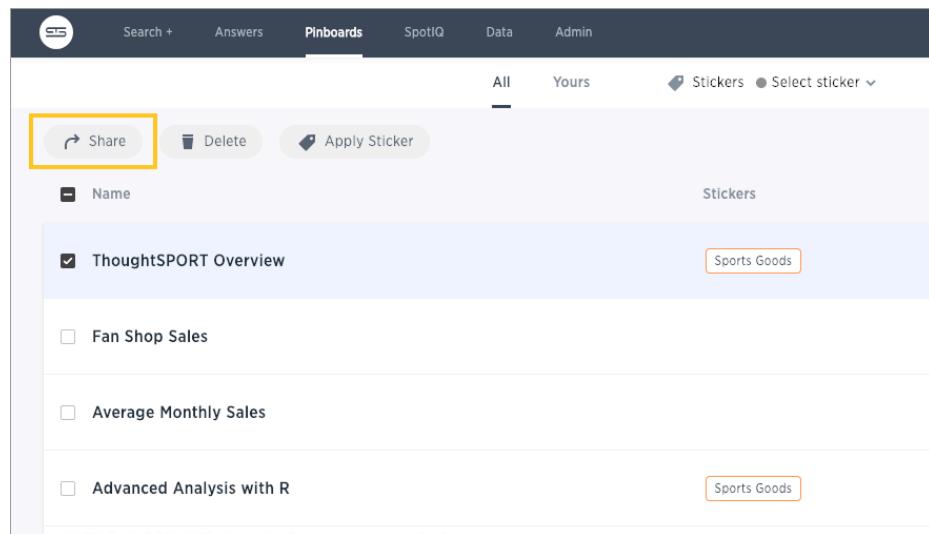
5. If you selected **Specific Columns**, select the column(s) to share.
6. Click **+** and select the users and groups with whom you want to share.
7. Configure the level of access by selecting from the dropdown list. You can select:
 - **Can View** to provide read-only access. This enables viewing the 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**.
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.



4. Click the X next to the users and groups that you want to remove from sharing.

Share

Who has access to pinboard "ThoughtSPORT Overview"

 victoria victoria	Can View ▾	x
 ThoughtSPORT ThoughtSPORT	Can View ▾	x
 basic basic	Can View ▾	x
 nathan Nathan	Can Edit ▾	x
 vicky vicky	Can View ▾	x
 marco marco	Can Edit ▾	x

+ 

5. Click **Save**, and then click **Done**.

Share

Who has access to pinboard "ThoughtSPORT Overview"

 victoria victoria	Can View ▾	x
 ThoughtSPORT ThoughtSPORT	Can View ▾	x
 basic basic	Can View ▾	x
 nathan Nathan	Can Edit ▾	x
 vicky vicky	Can View ▾	x

+

[DONE](#)

About SearchIQ

Summary: With SearchIQ, you can search your data through natural language, just like speaking.

Note: SearchIQ is in Beta

- SearchIQ is only available in English.
- The feature is off by default; to enable it, contact [ThoughtSpot Support](#).
- You must have the **Can use experimental features** permission.

To understand natural language search, SearchIQ collects data on searches and user-language preferences. Find more information about what SearchIQ collects at [SearchIQ data collection](#).

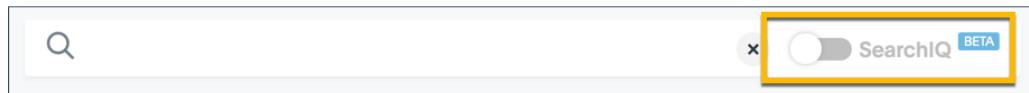
SearchIQ is a different search experience that understands more natural, speech-like search language than the [original ThoughtSpot search](#). For example, you can search for `What was my top selling product last month?`, instead of typing `top 1 product by sales last month`. You can also speak your search using the voice-to-text capability of your operating system.

How do I use SearchIQ?

Search IQ collects data to improve accuracy of natural language interpretation. For more details, see [SearchIQ data collection](#).

Clicking on the search bar on your homepage, or clicking **Search** in the top menu bar takes you to the search screen.

From here, if SearchIQ is enabled for you by your administrator, you can see a **SearchIQ** toggle button to the right of the search bar.



Switch it to the on position. This puts the search into SearchIQ mode, and you can now type in a search using plain, speech-like language.

If you leave the **SearchIQ** setting in the off position, you have the [original ThoughtSpot search experience](#), which uses keywords, column names, and data values.

What kinds of things can I type?

Like regular search, SearchIQ 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

What's different about SearchIQ is that the form of your question can be more like natural language. This means that SearchIQ ignores extraneous words like `What was the` or `Can you show me`. If you find that your natural language search doesn't return the results you expect, you can [teach SearchIQ your language](#) easily.

How do I use voice search with SearchIQ?

The ability to do voice search comes with your computer's operating system and your browser's speech-to-text capabilities. As such, it isn't provided by ThoughtSpot. But you can take advantage of these when searching with SearchIQ.

Make sure your microphone on your computer is enabled and working correctly. If you're unsure, check the documentation for your operating system.

Next, make sure your browser supports the text-to-speech capability by consulting its documentation.

If all is well, you can see a microphone icon in the search bar, when it is in SearchIQ mode. Clicking on the microphone will begin accepting voice input, which is translated to text in the search bar.

Note: If the SearchIQ text-to-speech functionality is not working, ensure that you are accessing ThoughtSpot in your browser using `HTTPS://` and not `HTTP://`.

Related information

- [SearchIQ Data Collection](#)
- [Use SearchIQ](#)
- [Teach SearchIQ your language](#)

Use SearchIQ

Summary: Use SearchIQ to do a natural language search.

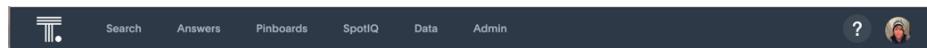
SearchIQ is in Beta.

Note: SearchIQ is turned off by default, you can have ThoughtSpot Support enable it for you.

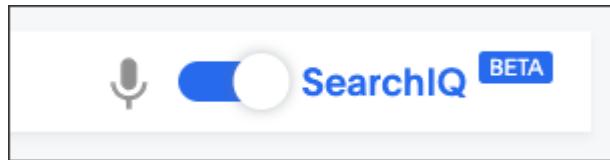
SearchIQ is only available in English.

To search using SearchIQ, follow these steps:

1. Click the ThoughtSpot search bar on the homepage or select **Search** from the top menu.

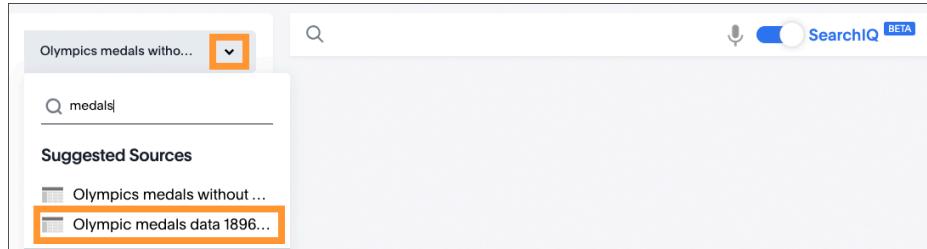


2. Click the SearchIQ toggle button to go into SearchIQ mode.



If you later decide to go back to the classic search mode, you can click the SearchIQ toggle button to turn it off again.

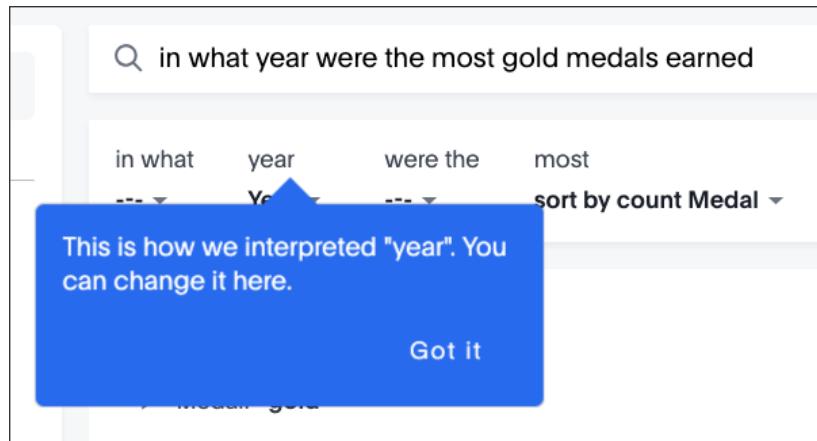
3. Choose the data you want to search using the selector in the left panel.



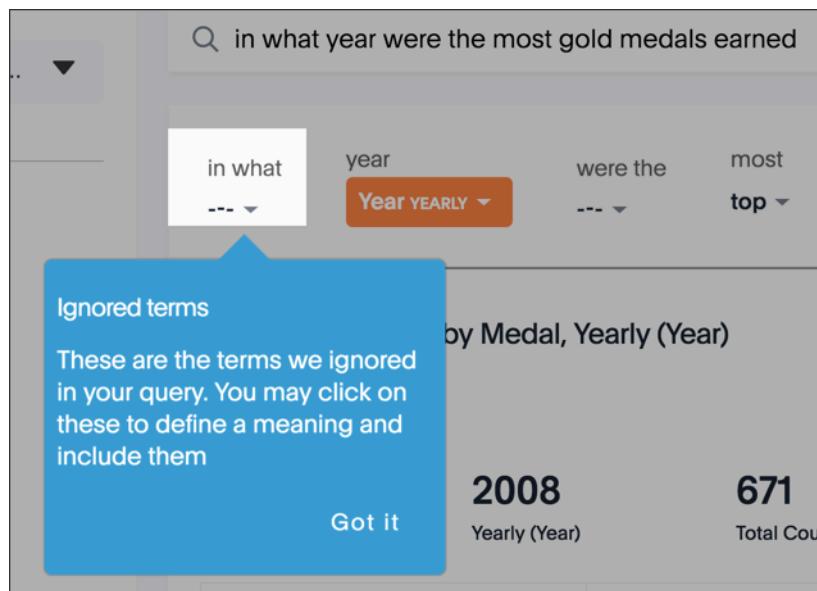
4. Type your search in the search bar using plain language.

If you see a microphone icon in the search bar, that means your operating system and browser are enabled for speech-to-text. In this case, you can click the microphone and speak your search instead of typing.

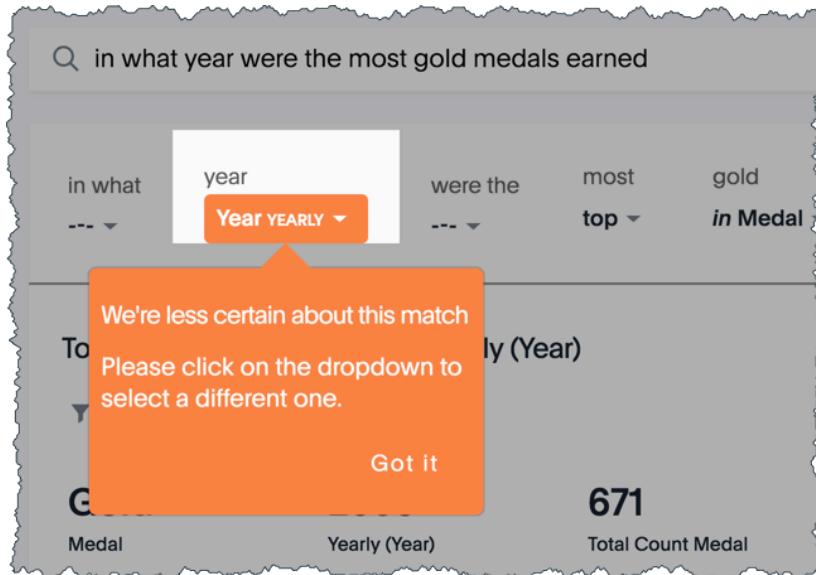
5. Press the **Enter** key when your search is complete.
6. When your answer appears, you may also see one or more of these messages:
 - Interpretation: You can see how different words you typed were understood.



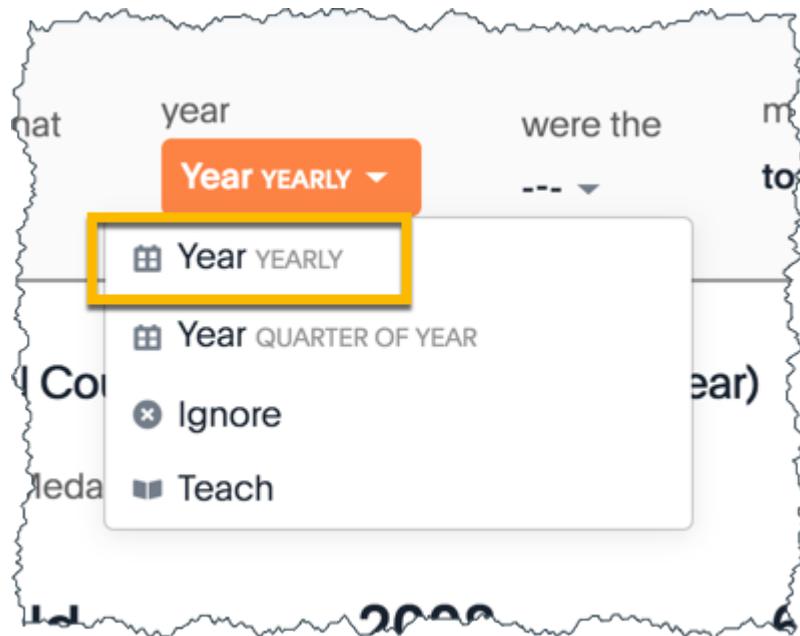
- Ignored words: These words were not needed in order to create the answer, so they are ignored.



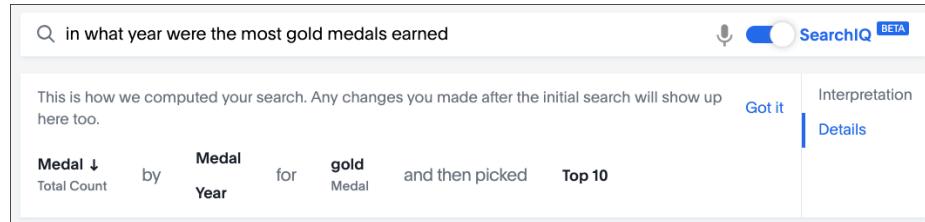
- Ambiguous words: These are words that could have more than one meaning in the context of the search.



You can clarify by choosing what you meant from the possible meanings shown.



- If you want to see details of how your search was computed, click the **Details** link to the right of the search terms.



8. If the answer isn't what you were expecting, you can refine your answer by [teaching SearchIQ your language](#).
9. If you need help, use [Ask an expert](#).

Related information

- [About SearchIQ](#)
- [Teach SearchIQ your language](#)
- [Ask an expert](#)

Teach SearchIQ your language

Summary: You can teach SearchIQ your language, to make its results more accurate.

SearchIQ is in Beta.

Note: SearchIQ is turned off by default, you can have ThoughtSpot Support enable it for you.

SearchIQ is only available in English.

Sometimes the results from SearchIQ won't be quite what you were expecting. You can easily teach SearchIQ your language as you use it. Over time, it gets smarter about how to interpret terms that don't come directly from keywords, column names, or data values.

Teach SearchIQ your Language

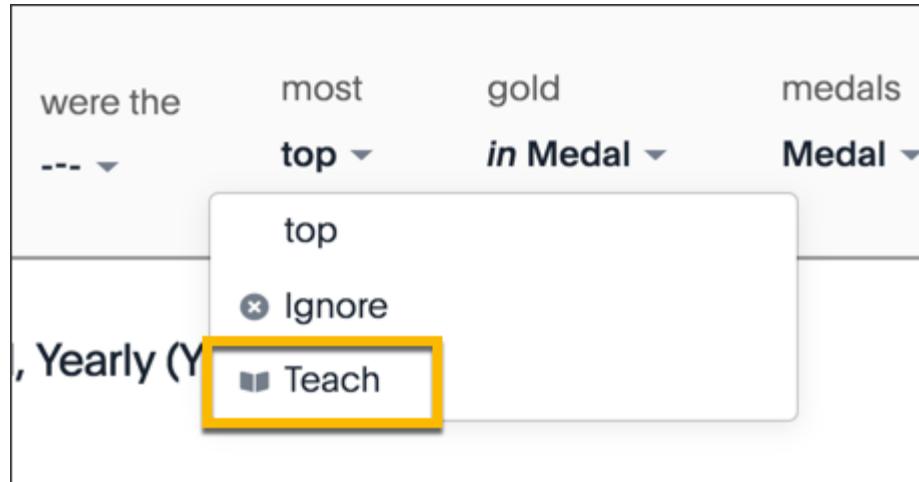
To teach SearchIQ your language:

1. After receiving a search result from SearchIQ, notice a section under the search bar that shows the SearchIQ interpretation of what you typed or spoke. If you do not see this, click **Refine your query**.

Wherever you see --, those terms were ignored. Where you see a word or words, those are the keywords, column names, or data values as understood by SearchIQ.

Clicking on any term brings up the menu where you can change how ThoughtSpot understands that term.

2. Click any term you want SearchIQ to ignore in the future, and choose **Ignore**.
3. Click any term you wish SearchIQ had interpreted differently, and choose **Teach**.



- When you see the teaching dialog box, 1) use the handles on the search diagram to select a term for which you want to change the mapping. Then 2) select a column name to map it to, or type your mapping in the search bar below the diagram. 3) You may need to select what you mean from the search suggestions.

Teach me your language 🎉

Search Columns

You can simply highlight what you wish using these handles! Got it

Use the sliders to select word(s) to teach me

in what year were the most gold medals earned

Olympic medals data 1896... Athlete, City, Country, Discipline, Event, Event_gender, Gender, Int Olympic Committee c..., Medal, Sport, Year

+ ADD COLUMNS

what "most" means below

top 25

25 Number

25 kilometres

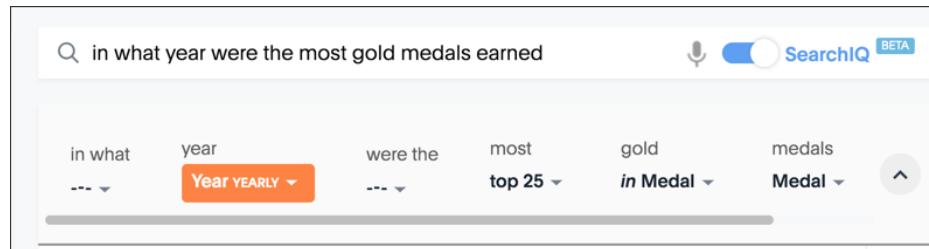
Event in Olympic medals data 1896 to 2008

25m army pistol

Event in Olympic medals data 1896 to 2008

CONFIRM

- Choose **CONFIRM** to see the new search result.



6. Repeat these steps until SearchIQ understands how to interpret your search.

SearchIQ will remember what you taught it, and use that to understand your searches in the future.

What are some best practices when teaching SearchIQ?

Searchiq learns the meanings of words and phrases as exact matches. Because of this, it is recommended to include the context in the phrase being mapped to improve the accuracy of the mappings.

For example, consider the following search:

"How many travellers are travelling from New York to San Francisco ?".

You can train SearchIQ to map this natural language search to the keyword search “unique count traveler name source city = ‘new york’ destination name = ‘san francisco’”. However, SearchIQ will not be able to understand whether “new york” is to be considered a match for “new york” as the source city or destination city.

To make the mapping more specific, you should map “from New York” to “source city = ‘New York’” and “to San Francisco” to “destination city = ‘San Francisco’”. This will help SearchIQ match a city to the source city or destination city based on the preceding preposition (to or from).

What if I teach SearchIQ a different language from somebody else?

SearchIQ learns the language you teach it, and your efforts are useful to everyone else who uses SearchIQ in the future. If SearchIQ learns different search term mappings from different users, these are applied to new searches in a hierarchical fashion, with the top learnings carrying more weight than those further down the list.

The hierarchy of SearchIQ language learning is set up like this:

1. Language mappings the user taught SearchIQ
2. Language mappings the user's **Group members** taught SearchIQ
3. [Language mappings set by the administrator](#)

In this way, SearchIQ provides the best and most customized search experience possible for all users, and your efforts to teach it are valuable to as many people as possible.

Related information

- [About SearchIQ](#)
- [Use SearchIQ](#)
- [Modify SearchIQ mappings](#)

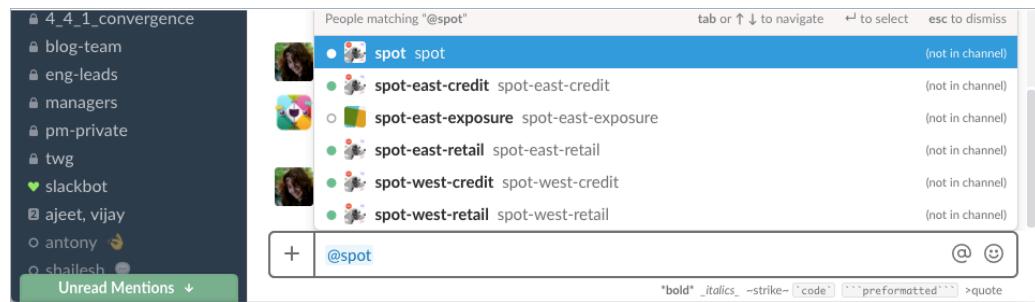
Slack and Spot

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.

If **@spot** doesn't come when you "call" you are spotless. Ask your administrator to see if you can get one.

Related Information

- Go to [How to use Spot](#) to get started using Spot.
- For information on setting up Spot, see [Slack Integration](#) in the Administration Guide.

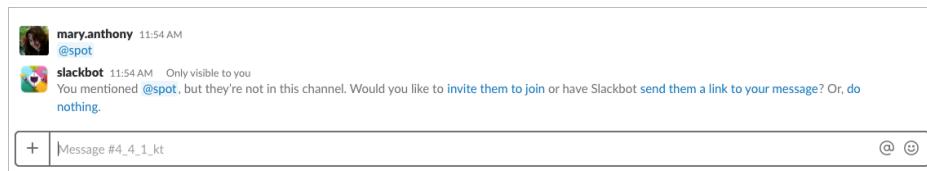
How to use Spot

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

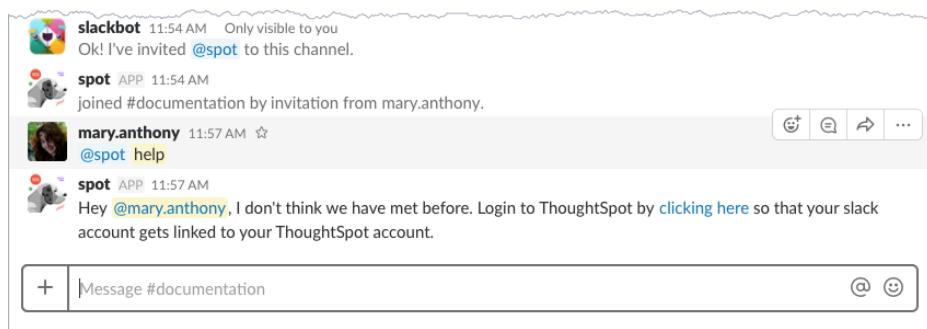
1. Launch Slack if you haven't already.
2. Go to a channel where you want to invite Spot.
3. Invite **@spot** like you would invite anyone else.

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:

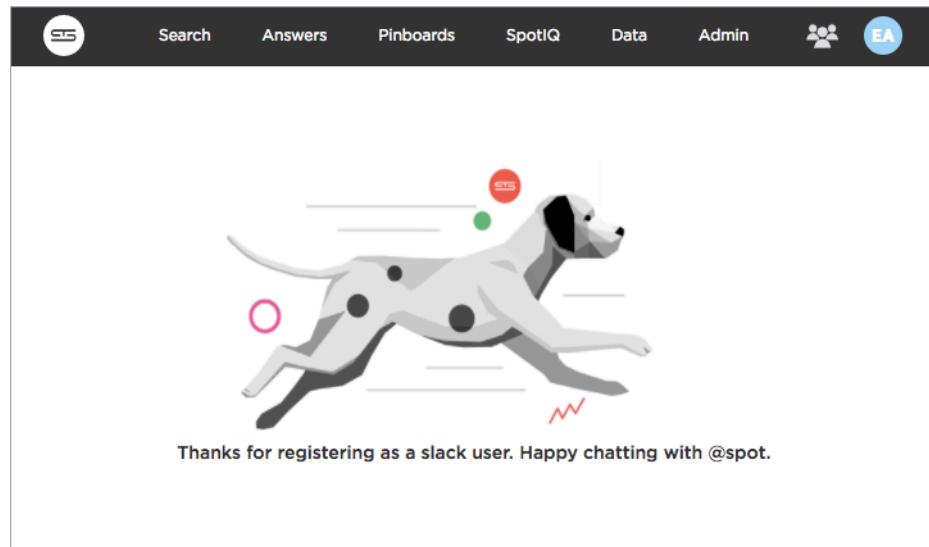


6. Take the [clicking here](#) link.

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`.

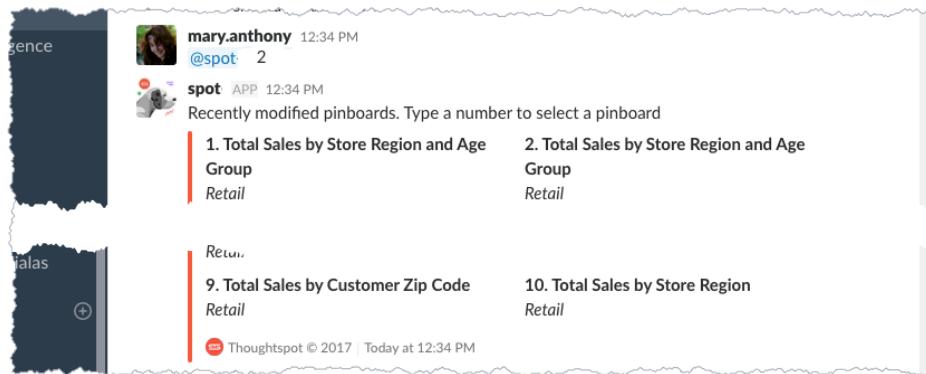
The image shows a screenshot of a Slack conversation. A user named 'mary.anthony' sends the command `@spot help`. The bot 'spot' responds with a list of commands and their descriptions. The list is as follows:

- 1. <anything>**
Will try to match a chart or pinboard with your query, Eg. *total weekly active users*
- 2. list pinboards**
Will list most recently modified pinboards
- 3. subscriptions**
Will give you a list of the items subscribed by you.
- 4. subscribe**
Search for a chart, display it. Type `subscribe` to subscribe to it.
- 5. more**
Eg: When seeing a list of objects, type `more` to see more of them.

At the bottom of the message, there's a note: 'Thoughtspot © 2017 Today at 11:58 AM'. The message input field at the bottom contains the text 'Message #documentation'.

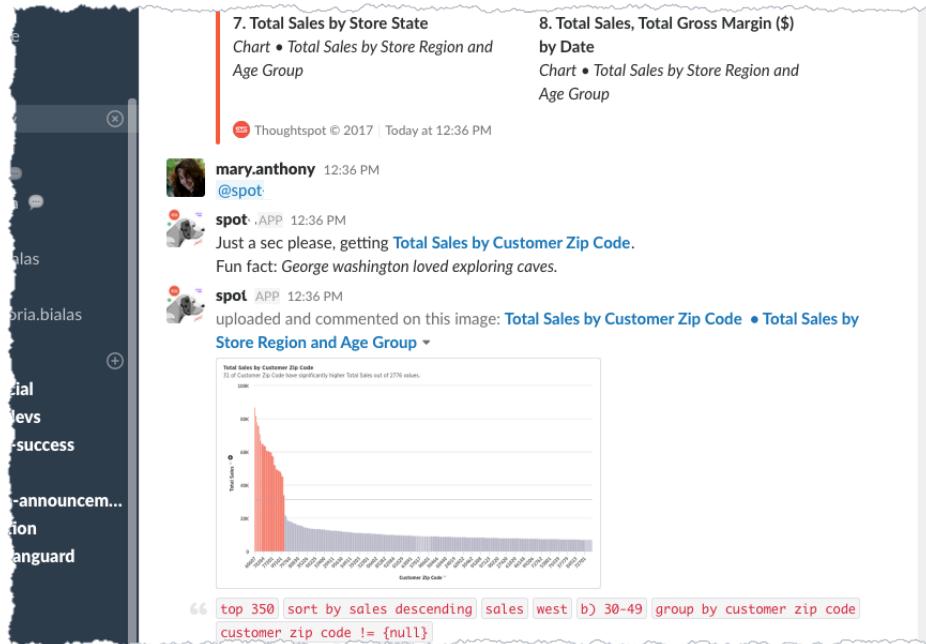
Spot can run a query directly that's the `**1. **` command.

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. After 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.

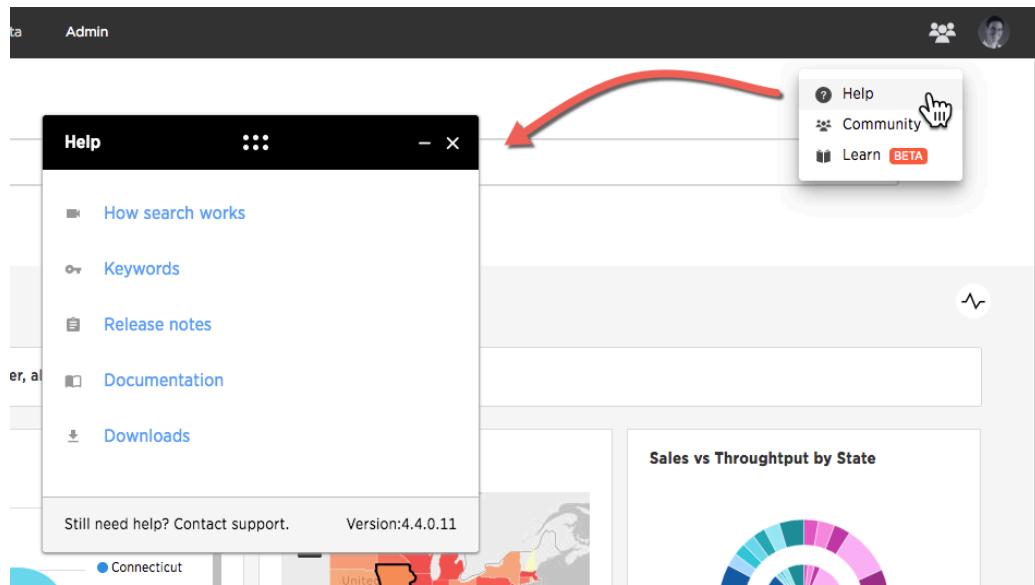
Each version of ThoughtSpot has its own documentation set. ThoughtSpot releases new documentation sets for every major release version (`x.y`) of ThoughtSpot. Documentation features added with minor versions (`x.y.z`) are added to the corresponding major release. This means, for example, that new documentation 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 in your browser's address bar. You can see the documentation for the latest version of the product.

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 can see a link to contact Support as well as the version of ThoughtSpot you are using.

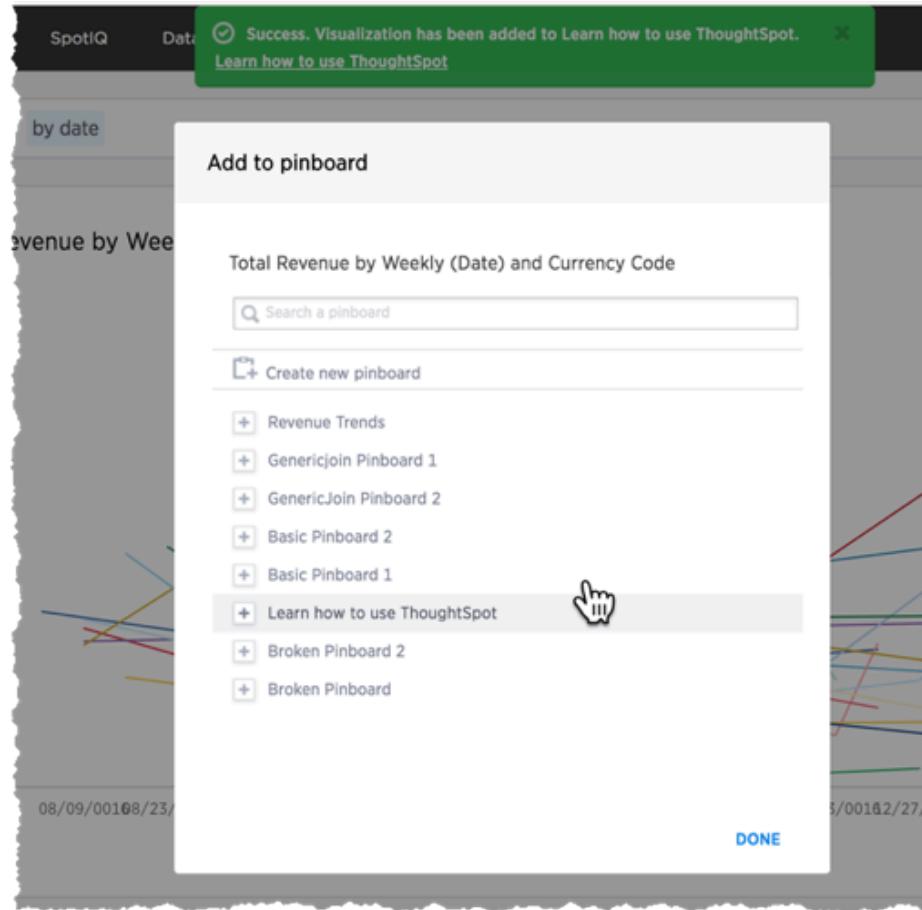
Learn

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.

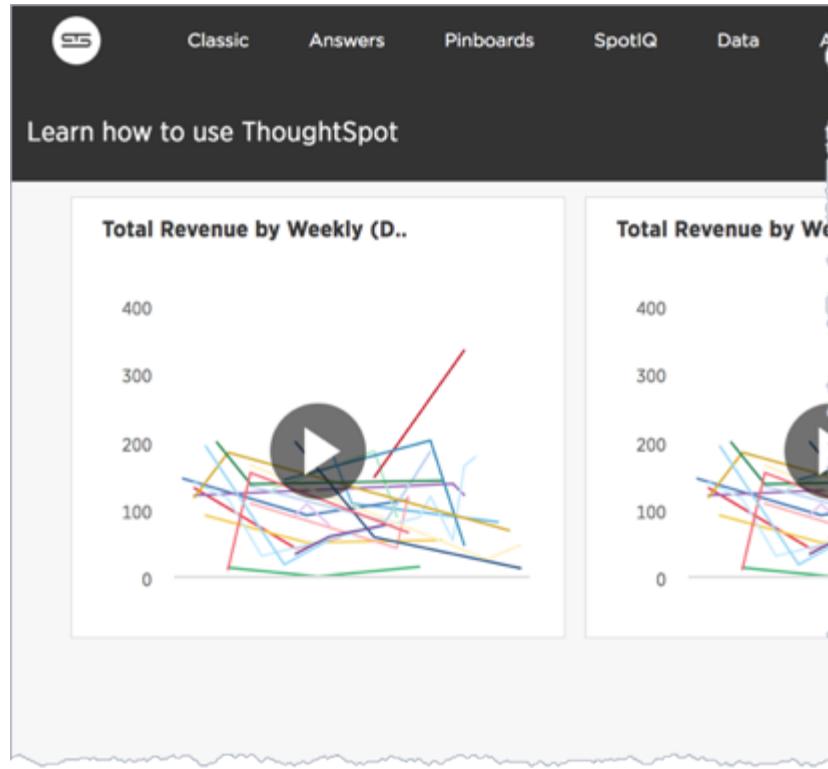
Click an item in Learn to replay it in ThoughtSpot. During a reply, you can click **Pause** or **Resume** or click **ESC** to exit the replay.

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.

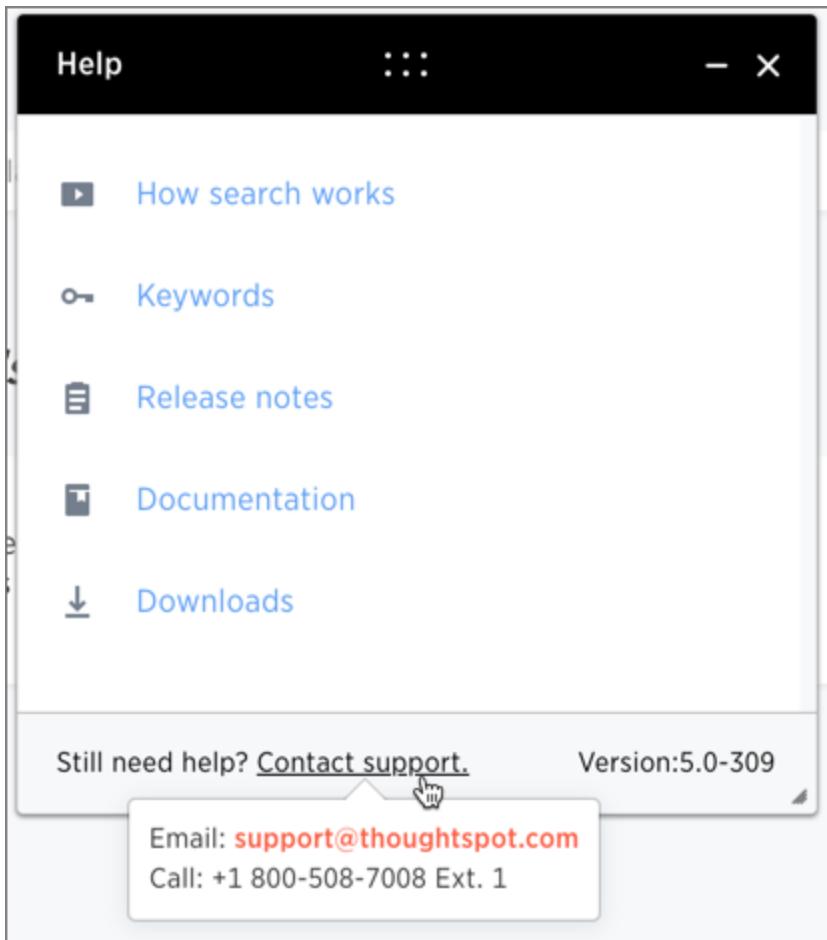


4. Navigate to the *Learn* center to replay your search.



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.

Keywords in Other Languages

Currently, we offer the following keyword translations.

日本語	中文 (简体)	Deutsche	Español (latín)	Español (España)
Français (Canada)	Français (France)	Português (Brasil)	Português (Portugal)	Italiano
Dansk	Suomi	Svenska	Norsk	Nederland

Also, see the topic on how to set [locale preferences in your user profile](#) to control language, date, and number formats on the ThoughtSpot UI.

General

Keyword	Examples
top	<ul style="list-style-type: none">• top sales rep by count sales for average revenue > 10000• sales rep average revenue for each region top
bottom	<ul style="list-style-type: none">• bottom revenue average• bottom revenue by state• customer by revenue for each sales rep bottom
n	top 10 sales rep revenue
n	bottom 25 customer by revenue for each sales rep
sort by	<ul style="list-style-type: none">• revenue by state sort by average revenue• revenue by customer sort by region

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
n days for each month	sales last 2 days for each month
n days for each quarter	revenue last 15 days for each quarter
n days for each week	total sold last 2 days for each week
n 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 pro-ski2000
growth of ... by ... yearly	growth of sales by date closed yearly
growth of ... by ...	growth of sales by order date
n hours for each day	sales last 2 hours for each day

Keyword	Examples
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
last <i>n</i> years	visitors last 5 years by revenue for sum revenue > 5000
month to date	sales by product month to date sales > 2400
<i>month year</i>	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
<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

Keyword	Examples
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
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
<i>n</i> weeks for each month	sales last 3 weeks for each month
<i>n</i> weeks for each quarter	last 2 weeks for each quarter
<i>n</i> 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

Keyword	Examples
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
<i>n</i> minutes	count visitors last 30 minutes
<i>n</i> hours	count visitors last 12 hours
hourly	visitors by page name hourly
<i>n</i> minutes ago	sum inventory by product 10 minutes ago
<i>n</i> 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'

Keyword	Examples
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
variance	variance sale amount by visitor by product for last year

Comparative

Function	Examples
all	all

Function	Examples
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 <i>n</i> miles km meters	revenue store name county near alameda within 50 miles
farther than <i>n</i> 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 (purchase date)
quarter of year (<i>date</i>)	quarter of year (purchase date)

month of quarter (<i>date</i>)	month of quarter (purchase date)
week of year (<i>date</i>)	week of year (ship date)
week of quarter (<i>date</i>)	week of quarter (ship date)
week of month (<i>date</i>)	week of month (ship date)
day of year (<i>date</i>)	day of year (ship date)
day of quarter (<i>date</i>)	day of quarter (ship date)
day (<i>date</i>)	day (ship date)
day of month (<i>date</i>)	day of month (order date)
day of week (<i>date</i>)	day of week (order date)
hour (<i>datetime</i>)	hour (timestamp)

In

Keyword	Example
in (<i>subsearch</i>)	in (top 10 store name by sales footwear)

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
SHOW DATABASES	Lists all available databases. Examples: SHOW DATABASES;
USE <database>	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: USE "fruit_database";
SHOW SCHEMAS	Lists all schemas within the current database. For example: SHOW SCHEMAS;
SHOW TABLES	Lists all tables within the current database by schema. For example: SHOW TABLES;
SHOW TABLE <table>	Lists all the columns for a table. For example: SHOW TABLE "locations";
SCRIPT SERVER	Generates the TQL schema for all tables in all databases on the server. For example: SCRIPT SERVER;

Syntax	Description
<pre>SCRIPT DATABASE <database></pre>	Generates the TQL schema for all tables in a database. For example: <pre>SCRIPT DATABASE "fruit_database";</pre>

SCRIPT TABLE <table>	Generates the TQL schema for a table. For example: <pre>SCRIPT TABLE "vendor";</pre>
-------------------------	---

Syntax	Description
<pre> SELECT <cols_or_expr> FROM <table_list> [WHERE <predicates>] [GROUP BY <expr>] [ORDER BY <expr>]</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 TSQL 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"> • sum • count • count distinct • stddev • avg • variance • min • max <p>You can use the following date functions:</p> <ul style="list-style-type: none"> • absyear • absmonth • absday • absquarter • date • time

For example:

```

SELECT TOP 10 "quantity" FROM "sales_fact";

SELECT COUNT(*) FROM "vendor";

SELECT "vendor", SUM("quantity") FROM "sales_fact" GROUP BY
"vendor";

SELECT "vendor", SUM("amount") FROM "vendor", "sales_fact"
    WHERE "sales_fact"."vendorid" = "vendor"."vendorid"
        AND "amount" > 100 GROUP BY "vendor" ORDER BY "amount"
DESC;

SELECT "vendor", SUM("quantity") FROM "sales_fact"
GROUP BY "vendor" LIMIT 10;
```

Schema creation

Syntax	Description
<pre>CREATE DATABASE <data- base></pre>	<p>Creates a database. For example:</p> <pre>CREATE DATABASE "fruit_database";</pre>
<pre>CREATE SCHEMA <schema></pre>	<p>Creates a schema within the current database. For example:</p> <pre>CREATE SCHEMA "fruit_schema";</pre>
<pre>CREATE TABLE <table> (<column_def- itions> [<con- straints>]) [PARTI- TION BY HASH (<num- ber>) [KEY ("<col- umn>")]]</pre>	<p>Creates a table with the specified column definitions and constraints.</p> <p>Use PARTITION BY HASH to shard a table across all nodes. If no KEY is specified, the table will be randomly sharded.</p> <p>Do not specify relationship constraints (FOREIGN KEY or RELATIONSHIP) in the CREATE TABLE statement. Instead, define these using ALTER TABLE 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:</p> <pre>CREATE TABLE "vendor" ("vendorid" int, "name" var- char(255)); 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");</pre>

Schema modification

Syntax	Description
<pre>DROP DATABASE <data- base></pre>	Drops a database and all of its schemas and tables. For example: <pre>DROP DATABASE "fruit_database";</pre>
<pre>DROP SCHEMA <schema></pre>	Drops a schema within the current database, and drops all of the tables in the schema. For example: <pre>DROP SCHEMA "fruit_schema";</pre>
<pre>DROP TABLE <table></pre>	Drops a table. For example: <pre>DROP TABLE "location";</pre>
<pre>TRUNCATE TABLE <table></pre>	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: <pre>TRUNCATE TABLE "location";</pre>

Syntax	Description
<pre>ALTER TABLE <table> ADD DROP RENAME COLUMN <column></pre>	<p>Alters a table to add, drop, or rename a column.</p> <p>When you add a column to an existing table, you must provide a default value to use for existing rows. For example:</p> <pre>ALTER TABLE "cart" ADD COLUMN "nickname" varchar(255) DE- FAULT 'no nickname';</pre> <pre>ALTER TABLE "cart" DROP COLUMN "nickname";</pre> <pre>ALTER TABLE "cart" RENAME COLUMN "nickname" TO "shortname";</pre>

ALTER TABLE <table> DROP CON- STRAINT PRIMARY KEY;	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:
	<pre>ALTER TABLE "sales" DROP CONSTRAINT PRIMARY KEY;</pre> <pre>ALTER TABLE "sales" ADD CONSTRAINT PRIMARY KEY ("P0_num- ber");</pre>

ALTER TABLE <table> DROP CON- STRAINT [FOREIGN KEY RELATIONSHIP] <name>;	Drops the named foreign key or relationship between two tables. For example: <pre>ALTER TABLE "sales_fact" DROP CONSTRAINT FOREIGN KEY "FK_P0_number";</pre> <pre>ALTER TABLE "fruit_dim" DROP RELATIONSHIP "REL_dates";</pre>
---	---

Syntax	Description
<pre>ALTER TABLE <table> DROP [CONSTRAINT FOREIGN KEY [<table_name>] RELATIONSHIP [WITH <table_name>];</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"> • the name of the referenced table, for a foreign key. • the name of the related table, for a relationship. <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>

Examples:

```
ALTER TABLE "shipments" DROP CONSTRAINT FOREIGN KEY "orders";
```

```
ALTER TABLE "wholesale_buys" DROP RELATIONSHIP WITH "retail_sales";
```

Drops all relationships that have wholesale_buys as a source.

```
ALTER TABLE "wholesale_buys" DROP RELATIONSHIP;
```

Drops all foreign keys from wholesale_buys.

```
ALTER TABLE "wholesale_buys" DROP CONSTRAINT FOREIGN KEY;
```

Syntax	Description
<pre>ALTER TABLE <table> [SET DI- MENSION SET FACT [PARTITION BY HASH [(<shards>)] [KEY(<column>)]]]</pre>	<p>Changes the partitioning on a table by doing one of:</p> <ul style="list-style-type: none"> • re-sharding a sharded table • changing a replicated table to a sharded table • changing a sharded table to a replicated (unsharded) table <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 <table> MODIFY COLUMN <column> <new_data_type>;</pre>	<p>Changes the data type of a column. This can have implications on sharding and primary key behavior. See About data type conversion. For example:</p> <pre>ALTER TABLE fact100 MODIFY COLUMN product_id int;</pre>
--	--

Modify data

Syntax	Description
<pre>INSERT INTO <table> 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>

Syntax	Description
<pre>ALTER TABLE <table> SET LOAD PRIORITY <value> <new_da- ta_type>;</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 1–100 . The system default for all tables is 50 . For example:</p> <pre>ALTER TABLE 'sales_facts' SET LOAD PRIORITY 1;</pre>
<pre>UPDATE <table> ... SET ... [WHERE ...]</pre>	<p>Updates rows in a table that match optionally provided predicates. Predicates have the form column = value connected by the AND keyword. 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 <table> [WHERE...]</pre>	<p>Deletes rows from a table that match optionally provided predicates. Predicates have the form column = value connected by the AND keyword. For example.</p> <pre>DELETE FROM "vendor" WHERE "name" = 'Joey Smith' AND "ven- dorid" = '19463';</pre>

Constraints and relationships

Constraints and relationships in ThoughtSpot are used to define the relationships between tables (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
PRIMARY KEY	<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")) ;</pre> <pre>ALTER TABLE "cart" ADD CONSTRAINT PRIMARY KEY ("cart_id");</pre> <pre>ALTER TABLE "cart" DROP CONSTRAINT PRIMARY KEY "cart_id";</pre>

FOREIGN KEY

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.

When creating a foreign key, give it a name. You can reference the foreign key name later, if you want to remove it.

Examples of this statement:

```
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_P0_vendor" FOREIGN KEY ("po_number",
"vendor") REFERENCES "orders" ("po_number", "vendor");
```

```
ALTER TABLE "shipment" DROP CONSTRAINT "FK_P0_vendor";
```

Syntax	Description
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" < "retail_sales"."date_sold" AND "retail_sales"."date_sold" < "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"> VARCHAR(<i>n</i>) 	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"> DOUBLE FLOAT 	DOUBLE is recommended.
Boolean	<ul style="list-style-type: none"> BOOL 	Can be true or false .

Syntax	Description	Examples
Integer	<ul style="list-style-type: none">• INT• BIGINT	<p>INT holds 32 bits.</p> <p>BIGINT holds 64 bits.</p>
Date or time	<ul style="list-style-type: none">• DATE• DATETIME• TIMESTAMP• TIME	<p>DATETIME, TIMESTAMP, and TIME are stored at the granularity of seconds</p> <p>.</p> <p>TIMESTAMP is identical to DATETIME, but is included for syntax compatibility.</p>

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 <database></code>	Specifies the pre-existing target database into which tsload should load the data.	
<code>--target_schema <schema></code>	Specifies the target schema. Default is “falcon_default_schema”.	
<code>--target_table <table></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 <number></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 <path_to_file>/<file_name></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 <date_formatmask></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 strftime library function .
<code>--date_time_format <date_formatmask>/<time_formatmask></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 strftime library function .
<code>--time_format <time_formatmask></code>	Specifies the format string for time values.	The default is <code>hour:minute:second</code> . Use the time format specifications supported in the strftime library function .

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, first time 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.	

Flag	Description	Notes
--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
--has_header_row	Indicates that the input file contains a header row.	If supplied, column names in the header row are used to match column names in the target table in ThoughtSpot. If not supplied, the first row of the file is loaded as data, the same as all subsequent rows.
--escape_character "<character>"	Specifies the escape character used in the input file.	If no value is specified, the default is "(double quotes).
--enclosing_character "<character>"	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: --enclosing_character "\""
--use_bit_boolean_values = [true false]	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 --boolean_representation [true_false 1_0 T_F Y_N] .

tscli command reference

Summary: The ThoughtSpot command line interface, or tscli, is an administration interface for the cluster. 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.

The command returns `0` on success, and a non-zero exit codes on failure. The `tscli` command logs errors to the `stderr` directory.

How to use the tscli command

The `tscli` command has the following syntax:

```
tscli [-h]
      [--helpfull]
      [--verbose]
      [--noautoconfig]
      [--autoconfig]
      [--yes]
      [--cluster ]
      [--zoo ]
      {access, alert, ansible, backup, backup-policy, calendar, callhome, cassandra,
       cluster, command, dr-mirror, etl, event, feature, fileserver,
       firewall, hdfs, ipsec, ldap, logs, map-tiles, monitoring, nas,
       node, notification, onboarding, patch, rpackage, saml, scheduled-pinboards, set, smtp, snapshot,
       snapshot-policy, socialproof spot, sssd, ssl, storage, support,
       tokenauthentication}
```

The `tscli` command has several subcommands, such as `alert`, `backup`, and so on.

Issue subcommands using the following format:

```
tscli [subcommand]
```

Subcommands have their own additional options and actions, such as `tscli backup create` or `tscli backup delete`.

Each subcommand may have several options.

To view help for a subcommand, type `-h` after the subcommand option:

```
tscli [subcommand] -h
```

tscli subcommands

This section lists each subcommand and its syntax.

access

```
tscli access [-h] {list}
```

This subcommand has the following option:

tscli access list

Lists objects by last access time, with the following parameters:

--type TYPE

Type of object, either answer or pinboard.

--limit LIMIT

The number of objects to fetch.

The default is 30.

--offset OFFSET

Offset to use to skip objects for batched results.

The default is 0.

--ascending

Sorts the answers by access time ascending.

The default is `True`.

alert

```
tscli alert [-h] {count,info,list,off,on,refresh,silence,status,unsilence}
```

This subcommand has the following options:

tscli alert count

Lists counts of generated alerts by type.

tscli alert info

Lists all alerts. Add `silenced` to list only silenced alerts, `active` to list only active alerts, or `detailed` to get detailed alert information.

tscli alert list

Lists the generated alerts, with these parameters:

--limit LIMIT

Specifies the number of recent alerts to display.

--since SINCE

Lists all alerts raised since a specified time period, in the form of a human readable duration string, such as `4h` (4 hours) or `4m` (4 minutes).

tscli alert off

Disables all alerts from the cluster in the cluster's timezone.

tscli alert on

Enables alerts from the cluster.

tscli alert refresh

Refreshes alert metadata on the cluster.

tscli alert silence --name NAME

Silences the alert with `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 NAME
```

Unsilences the alert with `NAME`. For example, `DISK_ERROR`.

ansible

```
tscli ansible [-h] {checkout,commit} [--local]
```

This subcommand has the following options:

```
tscli ansible checkout --host HOST
```

Checks out Ansible playbook with the target `HOST` that is running the `ts_ansible` service.

```
tscli ansible commit --host HOST
```

Commits Ansible playbooks with the target `HOST` that is running the `ts_ansible` service.

Use this subcommand to install and configure third-party software on the ThoughtSpot cluster.

For details, see these articles:

- [About third party security and monitoring software](#)
- [Installing third party security and monitoring software](#)

backup

```
tscli backup [-h] {create,delete,ls,restore}
```

This subcommand has the following options:

```
tscli backup create [-h] [--mode {full,light,dataless}] [--type {full,incremental}]  
[--base BASE]  
[--storage_type {local,nas}] [--remote] [--no-orion-master]
```

Pulls a snapshot and saves it as a backup, with these parameters:

```
--mode {full,light,dataless}
```

Mode of backups.

The default is `full`.

--type {full,incremental}

Type of backup.

Note: `incremental` is not implemented.

The default setting is `full`.

--base BASE

Based snapshot name for incremental backup.

Note: Because `incremental` is not implemented, neither is this option.

There is no default setting.

--storage_type {local,nas}

Storage type of output directory.

The default setting is `local`.

--remote

Takes backup through orion master.

The default setting is `True`.

--no-orion-master

Determines whether orion master is available during backup.

The default is `False`.

tscli backup delete name

Deletes the named backup.

tscli backup ls

Lists all backups taken by the system.

tscli backup restore

Restores cluster using backup, with the following parameters:

--release RELEASE

Restore the cluster on a specific release number.

--disable_rotate_keys

Disables cluster rotate key configurations.

The default is `False`.

--enable_cloud_storage

Determines whether to enable Cloud Storage setup.

--heterogeneous

Should be set for heterogeneous clusters.

The default is `False`.

backup-policy

```
tscli backup-policy [-h] {create,delete,disable,enable,ls,show,status,update}
```

Manages the backup policy.

This subcommand has the following options:

tscli backup-policy create

Prompts an editor for you to edit the parameters of a new periodic backup policy, with the following parameter:

--config CONFIG

Specifies the text format of the periodic backup policy config.

tscli backup-policy delete name

Deletes the backup policy `name`.

tscli backup-policy disable name

Disables the policy `name`.

tscli backup-policy enable name

Enables the policy `name`.

tscli backup-policy ls

Lists backup policies.

tscli backup-policy show name

Shows the backup policy `name`.

tscli backup-policy status name

Shows the status of the backup policy `name`.

tscli backup-policy update name

Prompts an editor for you to edit the backup policy `name`.

calendar

```
tscli calendar [-h] {create,delete,disable,enable,generate,ge  
t,list,update}
```

This subcommand has the following options:

tscli calendar create

Creates a new custom calendar, with the following parameters:

--file_path FILE_PATH

Path to the CSV file holding custom calendar data.

The default is `None`.

--name NAME

Custom calendar name.

The default is `None`.

--separator SEPARATOR

The separator used in the CSV file.

The default is `,`.

--no-header-row

Flag to indicate that the CSV file has no header row.

The default is `True`.

--username USERNAME

The admin username for ThoughtSpot login.

The default is `None`.

tscli calendar delete

Deletes a custom calendar table from the system, with the following parameters:

--name NAME

Deletes the custom calendar NAME.

The default is `None`.

--username USERNAME

The admin username for ThoughtSpot login.

The default is `None`.

tscli calendar disable

Disables custom calendar on the cluster.

tscli calendar enable

Enables custom calendar on the cluster.

tscli calendar generate

Creates a custom calendar table based on given specifications, with the following parameters:

--name NAME

A name to create the custom calendar CSV file with.

The default is `None`.

--start_date START_DATE

The start date to begin the custom calendar with in the form mm/dd/yyyy.

The default is `None`.

--end_date END_DATE

The end date to end the custom calendar with in the form mm/dd/yyyy.

The default is `None`.

--calendar_type {MONTH_OFFSET,4-4-5,4-5-4,5-4-4}

The type of custom calendar to create.

The default is `MONTH_OFFSET`.

--month_offset

{January,February,March,April,May,June,July,August,September,October,November,December}

The month offset to start the year from, if the calendar is the MONTH_OFFSET type.

The default is `January`.

--start_day_of_week

The day the week starts on.

The default is `Sunday`.

--quarter_name_prefix

The string to prefix a quarter name with.

--year_name_prefix YEAR_NAME_PREFIX

The string to prefix a year name with.

--username USERNAME

The admin username for ThoughtSpot login.

The default is `None`.

tscli calendar get

Procures data of a custom calendar as a CSV file, with the following parameters:

--name NAME

Procures data of custom calendar `NAME`

The default is `None`.

--username USERNAME

Admin username for ThoughtSpot login.

The default is `None`.

tscli calendar list

Procures a list of custom calendars present in the cluster, with the following parameter:

--username USERNAME

Admin username for ThoughtSpot login.

The default is `None`.

tscli calendar update

Updates a custom calendar table in the system, with the following parameters:

--file_path FILE_PATH

Path to the CSV file holding custom calendar data.

The default is `None`.

--name NAME

Custom calendar name.

The default is `None`.

--separator SEPARATOR

The separator used in the CSV file.

The default is `,`.

--no-header-row

Flag to indicate that the CSV file has no header row.

The default is `True`.

--username USERNAME

The admin username for ThoughtSpot login.

The default is `None`.

callhome

```
tscli callhome [-h] {disable,enable,generate-bundle}
```

This subcommand has the following options:

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`.

The default is `None`.

tscli callhome generate-bundle [--d D] [--since SINCE]

Generates the callhome stats tar file, with the following parameters:

--d D

Destination folder for the tar file.

There is no default setting.

--since SINCE

Grabs `callhome` data from the specified time window in the past.

This should be a human-readable duration string, such as `4h` (4 hours), `30m` (30 minutes), `1d` (1 day).

This option generates a `tar` file of the cluster metrics and writes it to the specified directory, where `SINCE` is how many days back the file must start.

There is no default setting.

cassandra

```
tscli cassandra [-h] {backup,restore}
```

Backs up cassandra.

This subcommand has the following options:

tscli cassandra backup

Takes a backup of cassandra, with the following parameters:

--keyspaces KEYSPACES

Comma separated list of keyspaces to take a backup of.

The default is `None`.

backup_dir BACKUP_DIR

The path to the backup directory to write the backup.

The default is `None`.

tscli cassandra restore

Restores cassandra from a backup, with the following parameter:

--backup_dir BACKUP_DIR

The path to the backup directory to write the backup.

The default is `None`.

cluster

```
tscli cluster [-h] abort-reinstall-os,abort-update,bucket- name,check,create,download-release,get-config,list-available-releases,list-downloaded-releases,load,reinstall-os,restore,resume-reinstall-os,resume-update,set-config,set-min-resource-spec,setup-release-host,setup-release-host-key,show-resource-spec,start,status,stop,update,update-hadoop}
```

This subcommand has the following options:

tscli cluster abort-reinstall-os

Aborts in-progress reinstall.

tscli cluster abort-update

Aborts an ongoing cluster update, if safe.

tscli cluster bucket-name

Returns the name of the s3 bucket associated with the cluster, if there is one.

tscli cluster check [--path PATH] [--includes INCLUDES] [--retry RETRY] [--localhost] [--disable-events]

Checks the status of all nodes in the cluster, with the following parameters:

--path PATH

Specifies the working directory of the diagnostic tool.

The default is `/usr/local/scaligent/release`.

--includes INCLUDES

Specifies the comma-separated component(s) to be included in the check.

The default is `all`.

--retry RETRY

The maximum number of retry times if the node is unreachable.

The default is `10`.

--localhost

Runs cluster checks only on localhost.

The default is `False`.

--disable-events

Disables raising configuration events.

The default is `False`.

tscli cluster create release

Creates a new cluster from the release file specified by adding the release number.

Used by ThoughtSpot Support when installing a new cluster. For example, `tscli cluster create 5.3.2.tar.gz`.

This command has the following parameters:

--disable_rotate_keys

Disables cluster rotate key configuration.

The default is `False`.

--enable_cloud_storage {s3a,gcs}

Determines whether to enable Cloud Storage setup, and which storage format to use.

heterogeneous

Should be set for heterogeneous clusters.

The default is `False`.

tscli cluster download-release release

Downloads the specified release to the Hadoop Distributed File System (HDFS) for later upgrading.

tscli cluster get-config

Gets current cluster network and time configuration. Prints JSON configuration to `stdout`.

If the system cannot be connected to all interfaces, the command returns an error but continues to function.

This command has the following parameters:

--local

Gets the config for the local host only.

The default is `False`.

--nodes NODES

A comma separated list of specified nodes to get the config for.

The default is `None`.

tscli cluster list-available-releases

Lists the available releases to update to on the cluster.

tscli cluster list-downloaded-releases

Lists the releases downloaded to the cluster.

tscli cluster load backupdir

Loads the state from a specified backup directory onto an existing cluster.

tscli cluster reinstall-os

Reinstalls OS on all nodes of the cluster, with the following parameters:

--secondary SECONDARY

A secondary drive for reinstall.

The default is `sdd`.

--stdin

Command to take JSON configuration from stdin.

tscli cluster restore --release RELEASE backupdir

Restores a cluster on the specified release number 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.

This command has the following parameters:

--disable_rotate_keys

Disables cluster rotate key configurations.

The default is `False`.

--enable_cloud_storage {s3a,gcs}

Determines whether to enable Cloud Storage setup.

--heterogenous

Should be set for heterogenous clusters.

The default is `False`.

tscli cluster resume-reinstall-os

Resumes in-progress reinstall.

tscli cluster resume-update

Resumes in-progress updates, with the following parameter:

--ignore_if_unhealthy

Comma separated list of node IPs on which upgrade is not attempted if they are found to be unhealthy. If a node outside of this list is found unhealthy, the upgrade is aborted.

The default is `None`.

tscli cluster set-config

Sets cluster network and time configuration. Takes JSON configuration from stdin.

This subcommand has the following parameters:

--ipv4-only

Only use ipv4 for node communication. Requires passing ipMap in config unless no-network-change is also specified.

The default is `False`.

--no-network-change

This flag ensures that a change made with set-config does not update network settings.

The default is `False`.

tscli cluster set-min-resource-spec

Sets the minimum resource configuration of the cluster.

tscli cluster show-resource-spec

Prints default or min.

tscli cluster start

Starts the 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 on all nodes.

This subcommand has the following option:

```
tscli command run [-h] [--nodes NODES] --dest_dir DEST_DIR [--copyfirst COPYFIRST]
[--timeout TIMEOUT] command
```

These are the parameters:

--nodes NODES

Space-separated IPs of nodes where to run the command.

The default setting is `all`.

--dest_dir DEST_DIR

Directory to save the files that contain the output from each node.

This is a mandatory parameter.

--copyfirst COPYFIRST

Copy the executable to required nodes first.

The default setting is `False`.

--timeout TIMEOUT

Timeout waiting for the command to finish.

The default setting is `60`.

dr-mirror**tscli dr-mirror [-h] {start,status,stop}**

This subcommand has the following options:

```
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}
```

This subcommand has the following options:

```
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]
```

Contact ThoughtSpot Support for assistance in setting this up.

Required parameters are:

--username *USERNAME*

Username for Informatica Cloud

--thoughtspot_url *THOUGHTSPOT_URL*

URL to reach thoughtspot.

--admin_username *ADMIN_USERNAME*

Admin username for ThoughtSpot

--groupname *GROUPNAME***--org_id *ORG_ID***

Specifies the Informatica `id` of the company.

--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.

--proxy_port *PROXY_PORT*

Proxy server port.

--proxy_username *PROXY_USERNAME*

Proxy server username.

--max_wait *MAX_WAIT*

Maximum time in seconds to wait for Data Connect agent to start.

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 and its options manage event notifications.

This subcommand has the following option:

```
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]
```

The `event` subcommand accepts these optional flags:

--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, such as `4h` (4 hours), `30m` (30 minutes), `1d` (1 day).

--from FROM

Begin timestamp. Must be of the form: `yyyymmdd-HH:MM`.

--to TO

End timestamp. Must be of the form: `yyyymmdd-HH:MM`.

--limit LIMIT

Maximum number of events to fetch.

The default setting is `0`.

--detail

Print events in detail format. This is not tabular. Default is a tabular summary.

The default setting is `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 `|` (pipe). The event is returned if it `matchesALL`. Put single quotes around the param value to prevent undesired glob expansion.

--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 `|` (pipe). The event is returned if it `matches ALL`. Put single quotes around the param value to prevent undesired glob expansion.

--attributes ATTRIBUTES

Specify attributes to match as key=value. Multiple strings to check for can be specified by separating them with `|` (pipe). The event is returned if it `matches ALL`. Put single quotes around the param value to prevent undesired glob expansion.

feature

```
tscli feature [-h] {get-all-config}
```

This subcommand has the following option:

tscli feature get-all-config

Gets the configured features in a cluster. The command returns a list of features, such as custom branding, Data Connect, and call home, and informs whether they are enabled or disabled.

fileserver

```
tscli fileserver [-h] {configure,download-release,purge-config,show-config,upload}
```

This subcommand has the following options:

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 have to issue this command one time, to set up the connection to the secure file server. 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, including its checksum, and verifies the integrity of release bundle.

You must specify the exact release number, such as `5.1.3`.

Before using this command for the first time, you must set up the file server connection using `tscli fileserver configure`. You can then work with a member of the ThoughtSpot Support team because a privileged user and a corresponding password must be specified to download releases.

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 specified file to the directory specified on the secure file server.

You may optionally specify the `user` and `password` to bypass the credentials specified when configuring the file server connection with `tscli fileservice configure`. Before using this command for the first time, you must set up the file server connection using `tscli fileservice configure`.

This uses the following flags:

--user *USER*

Username of the fileserver.

--password *PASSWORD*

Password of the fileserver. This is required and the command prompts you for it if you do not supply it.

--file_name *FILE_NAME*

Local file to upload.

--server_dir_path *SERVER_DIR_PATH*

Directory path on fileserver. The `SERVER_DIR_PATH` parameter specifies the directory for file upload. It is based on customer name, and takes the form `/Shared/support/customer_name`.

firewall

```
tscli firewall [-h] {close-ports,disable,enable,open-ports,stat  
us}
```

This subcommand has the following options:

tscli firewall close-ports

Closes specified ports through firewall on all nodes.

Accepts a comma-separated list of ports. Only closes ports that were previously opened using `open-ports`, and ignores ports that were not opened with `open-port`, or closed ports.

Some essential ports are always kept open, such as `ssh`; they are not affected by this command or by `open-ports`.

tscli firewall disable

Disable firewall.

tscli firewall enable

Enable firewall.

tscli firewall open-ports --ports *ports*

Opens specified ports through a firewall on all nodes.

Accepts a comma-separated list of ports.

Ignores open ports.

Some essential ports are always kept open, such as `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 option:

tscli hdfs leave-safemode

Command to get HDFS namenodes out of `safemode`.

ipsec

```
tscli ipsec [-h] {disable,enable,status}
```

This subcommand has the following options:

tscli ipsec disable

Disable IPSec

tscli ipsec enable

Enable IPSec

tscli ipsec status

Show IPSec status on all nodes

ldap

```
tscli ldap [-h] {add-cert,configure,purge-configuration}
```

This subcommand has the following options:

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.

tscli ldap purge-configuration

Purges (removes) any existing LDAP configuration.

logs

```
tscli logs [-h] {collect,runcmd}
```

Manages the logging behavior.

This subcommand has the following options:

```
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 it is interpreted through `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`.

--since SINCE

Grab logs from this time window in the past. Should be a human-readable duration string, such as `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`.

--to TO

Timestamp where collection ends; must be of the form `yyyymmdd-HH:MM`.

--out OUT

Tarball path for writing logs from each node.

The default setting is `/tmp/logs.tar.gz`.

--maxsize MAXSIZE

Only fetch logs if size is smaller than this value. Can be specified in megabytes or gigabytes, such as `100MB`, `10GB`.

--sizeonly

Do not collect logs. Just report the size.

The default setting is `False`.

--nodes NODES

Comma separated list of nodes from where to collect logs. Skip this to use all nodes.

```
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.

Accepts these optional flags:

--cmd CMD

Unix-Command to be run on the selected logs. Use single quotes to escape spaces etc. Note the language used to specify CMDSTR has following rules.

- A logfile and its corresponding result file can be referred by keywords `SRCFILE` and `DSTFILE`. For example, `cp SRCFILE DSTFILE`.
- Without any reference to `DSTFILE` in CMDSTR, `DSTFILE` will be appended to CMDSTR for output redirection. For example, `du -sch SRCFILE` gets auto-transtalted to `du -sch SRCFILE > DSTFILE`.
- Without any reference to `SRCFILE`, content of log is streamed to CMDSTR by pipe. For example, `tail -n100 | grep ERROR` gets auto-transtalted to `cat SRCFILE | tail -n100 | grep ERROR > DSTFILE`.

--include INCLUDE

Comma-separated list of logs to include,each entry is either a "selector" or a glob for matching files.

Selectors must be one of `all`, `orion`, `system`, `ts`.

Anything that starts with `/` (forward slash) is assumed to be a glob pattern and interpreted through `find(1)`. Other entries are ignored.

TIP: use single quotes around the parameter value to prevent undesired glob expansion.

The default setting is `all`.

`--exclude EXCLUDE`

Comma separated list of logs to exclude. Applies to the list selected by `--include`. Params are interpreted just like in `--include`.

`--since SINCE`

Grab logs from this time window in the past. Should be a human-readable duration string, such as `4h` (4 hours), `30m` (30 minutes), `1d` (1 day).

`--from FROM`

Timestamp where collection begins; must be of the form `yyyymmdd-HH:MM`.

`--to TO`

Timestamp where collection ends; must be of the form `yyyymmdd-HH:MM`.

`--outfile OUTFILE`

File path for printing all results. By default printed to `stdout`

`--outdir OUTDIR`

Directory path for writing results with original directory structure from each node. Used as an alternative to printing output to `outfile/stdout`.

`--cmd_infmt CMD_INFMT`

Specify if the input file should be compressed or uncompressed before running `CMD`. `C` for compressed, `U` for uncompressed. Don't use this flag if `CMD` works on both.

`--cmd_outfmt CMD_OUTFMT`

Specify if `OUTFILE` generated by `CMD` should be compressed or uncompressed. `C` for compressed, `U` for uncompressed. Don't use this flag if output file is of the same format as the input file.

`--nodes NODES`

Comma separated list of nodes where to run command. Skip this to use all nodes.

map-tiles

```
tscli map-tiles [-h] {disable,enable,status}
```

This subcommand has the following options:

```
tscli map-tiles enable [-h] [--online] [--offline] [--tar TAR] [--md5 MD5]
```

Enables ThoughtSpot's map tiles, used when constructing geomap charts.

If you don't have internet access, you must download the map tiles tar and md5 files, and append the following to the `tscli` command:

--online

Download `maptiles` tar from internet.

The default setting is `True`

--offline

Using `maptiles` tar from local disk.

The default setting is `False`

--tar TAR

Specified tar file for map-tiles.

--md5 MD5

Specified md5 file for map-tiles.

tscli map-tiles disable

Disable map-tiles functionality.

tscli map-tiles status

Check whether map-tiles are enabled.

monitoring

```
tscli monitoring [-h] {set-config,show-config}
```

This subcommand has the following options:

```
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.

The `monitoring` subcommand accepts the following optional flags:

--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.

The default setting is False .

tscli monitoring show-config

Shows the monitoring configuration.

nas

```
tscli nas [-h] {ls,mount-cifs,mount-nfs,unmount}
```

This subcommand has the following options:

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.

Accepts the following optional flags:

--server SERVER

IP address or DNS name of CIFS service.

For example, 10.20.30.40 .

--path_on_server PATH_ON_SERVER

Filesystem path on the CIFS source server to mount NAS.

For example, `/a`.

The default setting is `/` (forward slash).

--mount_point MOUNT_POINT

Directory on all cluster nodes where to mount the NFS filesystem on the target.

If this directory does not exist, the command creates it. If this directory already exists, the command uses it for mounting.

For example, `/mnt/external`.

--username USERNAME

Username to connect to the CIFS filesystem

--password PASSWORD

CIFS password for `--username`

--uid UID

The `UID` that owns all files or directories on the mounted filesystem when the server does not provide ownership information.

See `man mount.cifs` for more details.

The default setting is `1001`.

--gid GID

The `GID` that owns all files or directories on the mounted filesystem when the server does not provide ownership information.

See `man mount.cifs` for more details.

The default is `1001`.

--options OPTIONS

Other command-line options to forward to the `mount.cifs` command.

The default setting is `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.

Accepts the following optional flags:

--server SERVER

IP address or DNS name of NFS service. For example, `10.20.30.40`.

--path_on_server PATH_ON_SERVER

Filesystem path on the NFS source server.

For example, `/a/b/c/d`.

The default setting is `/`.

--mount_point MOUNT_POINT

Directory on all cluster nodes of the target system.

If this directory does not exist, the command creates it. If this directory already exists, the command uses it for mounting.

For example, `/mnt/external`.

--options OPTIONS

Command-line options to mount.

The default setting is `noexec`.

--protocol PROTO

One of `nfs` or `nfs4`.

The default is `nfs`.

tscli nas umount [-h] --dir DIR

Unmounts all devices from the specified directory, `DIR`.

This command returns an error if nothing is currently mounted on this directory through `tscli nas mount`.

node

```
tscli node [-h] {check,ls,reinstall-os,resume-reinstall-os,stat,us}
```

This subcommand has the following options:

```
tscli node check [-h] [--select {reinstall-preflight}] [--secondary SECONDARY]
```

Run checks per node.

Accepts the following flags:

```
--select {reinstall-preflight}
```

Select the type of node check

The default setting is `reinstall-preflight`.

```
--secondary SECONDARY
```

Secondary drive for `reinstall-preflight`.

The default setting is `sdd`.

```
tscli node ls [-h] [--type {all,healthy,not-healthy}]
```

Filter by node state.

The default setting is `all`.

```
tscli node reinstall-os [-h] [--secondary SECONDARY] [--cluster]
```

Reinstall OS on a node.

Accepts the following flags:

```
--secondary SECONDARY
```

Secondary drive for reinstall.

The default setting is `sdd`.

```
--cluster
```

The node part of a cluster.

The default setting is `False`.

```
tscli node resume-reinstall-os
```

Resume in-progress reinstall

onboarding

```
tscli onboarding [-h] {configure,purge-configuration}
```

Onboarding helps application administrators to bulk update user information. In particular, it configures various in-app email options.

This subcommand has the following options:

```
tscli onboarding --help
```

Prints help for the onboarding configuration

```
tscli onboarding configure
```

Configures the onboarding through series of steps.

Asks the user to provide information necessary for onboarding-related functionality, such as the following:

1. Company name
2. Product name
3. Should welcome emails be enabled?
 - Send welcome emails to new users
 - Support email
 - Custom message to include in emails
 - URL of the ThoughtSpot instance
 - URL of the ThoughtSpot documentation

```
tscli onboarding purge-configuration
```

This command removes all onboarding configuration.

patch

```
tscli patch [-h] {apply,ls,resume-apply,resume-rollback,rollbac  
k}
```

This subcommand has the following options:

```
tscli patch apply [-h] [release]
```

Apply the patch on an existing cluster.

Accepts the following flag:

```
release
```

The relative path to the patch tar ball.

```
tscli patch ls [-h] [--applied] [--rolled_back] [--service SERVICE] [--md5 MD5] [--
```

history]

Lists the patches currently applied.

Accepts the following flags:

--applied

Show only the patches applied since last full release.

The default setting is `False`.

--rolled_back

Show only the patches rolled back since last full release.

The default setting is `False`.

--service SERVICE

Show patches filtered by service.

The default setting is `None`.

--md5 MD5

Shows the details of the patch specified.

The default setting is `None`.

--history

Shows the history of all patch apply/rollback release.

The default setting is `False`.

tscli patch resume-apply [-h]

Resume patch apply

tscli patch resume-rollback [-h]

Resume patch roll-backup

tscli patch rollback [-h]

Rollback the patch from an existing cluster

rpackage

```
tscli rpackage [-h] {add,delete,list}
```

Manages R packages available to SpotIQ.

This subcommand has the following options:

```
tscli rpackage add [-h] [--repo REPO] [--timeout TIMEOUT] [--dest_dir DEST_DIR] [--nodes NODES] package_name
```

Command to add an R package, `package_name`, to the cluster.

Accepts the following flags:

--repo REPO

Specify the url of a specific repository 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

--nodes NODES

Space-separated list of IPs for nodes where to run the command.

The default setting is `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.

Accepts the following flags:

--timeout REPO

Timeout waiting before removing the R package.

The default is 60

--dest_dir REPO

Directory where to save the output of this command.

--nodes NODES

Space-separated list of node IPs where to run the command.

The default setting is `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 options:

```
tscli saml configure [-h]
```

Configures SAML.

```
tscli saml purge-configuration
```

Purges any existing SAML configuration.

To see a list of prerequisites, refer to [Configure SAML](#).

scheduled-pinboards

```
tscli scheduled-pinboards [-h] {disable,enable}
```

This subcommand has the following options:

```
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.

ⓘ Note: When you enable scheduled pinboards, you should also configure a whitelist of intended email domains. Contact ThoughtSpot Support for help on how to configure a whitelist.

smtp

```
tscli smtp [-h] {remove-mailfromname,remove-mailname,remove-relayhost,remove-saslcredentials,reset-canonical-mapping,set-canonical-mapping,set-mailfromname,set-mailname,set-relayhost,set-saslcredentials,show-canonical-mapping,show-mailfromname,show-mailname,show-relayhost}
```

This subcommand has the following options:

tscli smtp remove-mailfromname

Removes current cluster mail from name.

tscli smtp remove-mailname

Removes current cluster mail name.

tscli smtp remove-relayhost

Removes current cluster relay host.

tscli smtp remove-saslcredentials

Clears SASL credentials and disables SMTP AUTH.

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 and an email address from where email alerts are sent for the cluster.

tscli smtp set-mailname mailname

Sets the mailname and a domain from 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.

Accepts the following flag:

--force FORCE

Set even if relay host is not accessible.

The default setting is `False`.

tscli smtp set-saslcredentials

Sets SASL credentials and enables SMTP AUTH

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 a relay host is not configured, the command returns `NOT FOUND`.

snapshot

```
tscli snapshot [-h] {backup,create,delete,ls,pin,restore,unpin,update-ttl}
```

To learn more about snapshots and backups, see the [Understand the backup strategies](#) documentation.

This subcommand has the following options:

```
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.

Accepts the following flags:

--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.

The default setting is `full`.

```
--base BASE
```

Based snapshot name for incremental backup.

Incremental backup not implemented yet.

```
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 specified `name` and `reason`.

This command does not accept `.` (periods). It does accept `-` (dashes or hyphens).

The `ttl` parameter is the number of days after which this snapshot is automatically deleted. A value of `-1` disables automatic deletion.

```
tscli snapshot pin [-h] name
```

Pins a snapshot so it cannot be deleted or garbage collected.

```
tscli snapshot delete [-h] name
```

Deletes the named snapshot.

```
tscli snapshot ls [-h]
```

Lists available snapshots.

```
tscli snapshot restore [-h] [--allow_release_change] [--only_service_state] name
```

Restores cluster to an existing snapshot.

Accepts the following flags:

```
--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 unpin [-h] name
```

Unpin a snapshot so it can be deleted or garbage-collected

```
tscli snapshot update-ttl [-h] [--disable DISABLE] name ttl
```

Updates manual snapshot garbage collection policy.

Accepts the following flags:

name

Specifies which snapshot to update.

ttl

Extends the manual snapshot

ttl

This is the "time-to-live" value.

Use a positive value to increase **ttl**. Use negative values 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 has the following options:

```
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.

spot

```
tscli spot [-h] {enable}
```

Enables Spot integration.

This subcommand has the following option:

```
tscli spot enable [-h] --token TOKEN --thoughtspot_url THOUGHTSPOT_URL [--cache_timeout CACHE_TIMEOUT]
```

The `spot` subcommand accepts the following optional flags:

--token TOKEN

Slack authorization 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.

The default setting is `60000`.

ssl

```
tscli ssl [-h] {add-cert,clear-min-tls-version,off,on,rm-cert,set-min-tls-version,status,tls-status,add-valid-hosts}
```

This subcommand manages the SSL configuration.

To use SSL, the following ports must be open:

- 443

- 80

This subcommand has the following options:

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.

tscli ssl add-valid-hosts [-h] VALID_HOSTS

Enables host validation for the specified host(s). Helps improve security. This feature is for all customers that have or are planning to enable SSL. Multiple hosts must be separated by a comma (,).

Examples:

1. If you want to make sure the valid host is **cluster1.corp.example.com**, you would run the command:

tscli ssl add-valid-hosts cluster1.corp.example.com

2. If you want to allow all hosts which have the suffix **corp.example.com**, you would run the command:

tscli ssl add-valid-hosts *.corp.example.com. This wild card should be used within the hostname.

3. If you want to allow multiple valid hosts, for example both ***.corp.example.com** and **cluster1**, you would run the command:

tscli ssl add-valid-hosts *.corp.thoughtspot.com,cluster1

sssd

```
tscli sssd {enable, disable, set-sudo-group, clear-sudo-group}
```

This subcommand uses system security services daemon (SSSD), and has the following options:

```
tscli sssd enable --user USER --domain DOMAIN
```

Enables system Active Directory (AD) user access on a single node. You will be prompted for password credentials. The user must have permission to join a computer or VM to the domain.

```
tscli sssd disable
```

```
tscli sssd set-sudo-group ACTIVE_DIRECTORY_GROUP_NAME
```

Allows `sudo` permissions for AD group.

```
tscli sssd clear-sudo-group ACTIVE_DIRECTORY_GROUP_NAME
```

Clears any set AD sudo group.

For more about setting up Active Directory access, see [Enable Active Directory based access](#).

storage

```
tscli storage [-h] {gc,df}
```

This subcommand has the following options:

```
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 finishes, you can restart the cluster with `tscli cluster start`.

This command frees space in the following directories:

```
/tmp  
/usr/local/scaligent/logs/  
/export/logs/orion  
/export/logs/oreo  
/export/logs/hadoop  
/export/logs/zookeeper  
cores
```

The `storage` subcommand accepts these optional flags:

--log_age *LOG_AGE*

Deletes logs older than these many hours. Use a non-zero value, because zero deletes all temporary files, including the ones that are closed temporarily, while they are passed from one component to the next.

The default setting is `4`.

--force

Forces deletion of all logs and temporary files regardless of age. This must only be run on a stopped cluster.

The default setting is `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]  
  {bundle, restart-remote, rm-admin-email, rm-admin-phone, rm-  
feedback-email,  
   set-admin-email, set-admin-phone, set-debug-ui-password, se  
t-feedback-email,  
   set-remote, show-admin-email, show-admin-phone, show-feedba  
ck-email,  
   show-remote, start-remote, stop-remote}
```

This subcommand has the following options:

```
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 can be used for logs collection: `all`, `orion`, `system`, `ts`, or the name of a service.

Anything that starting with `/` (forward slash) is assumed to be a glob pattern, and it is interpreted through `find(1)`. Other entries are ignored.

TIP: Use single quotes around the param value to prevent undesired glob expansion. Use `all` to collect all selectors and all logs.

The default setting is `all_but_logs`.

--exclude EXCLUDE

Comma-separated list of selectors to exclude. Applies to the list selected by `--include`. Params are interpreted in the same manner as in `--include`.

Use the special keyword `logs` to exclude logs collection altogether.

There is no default setting.

--list_selectors

List the selectors available for `--include` and `--exclude`, and then exit.

The default setting is `False`.

--since SINCE

Grabs logs from this time window in the past. Should be a human-readable duration string, such as `4h` (4 hours), `30m` (30 minutes), `1d` (1 day).

There is no default setting.

--from FROM

Timestamp when collection begins. Must be of the form: `yyyymmdd-HH:MM`.

There is no default setting.

--to *T0*

Timestamp when collection ends. Must be of the form: `yyyymmdd-HH:MM`.

There is no default setting.

--out *OUT*

Tarball path for dumping the support bundle.

The default setting is `/tmp/support_bundle.tar.gz`.

--nodes *NODES*

Comma separated list of nodes from where to collect logs. Skip this to use all nodes.

There is no default setting.

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. 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. 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. 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, such as `+1 800-508-7008 Ext. 1`. 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, such as `tunnel.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

```
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}
```

This subcommand has the following options:

```
tscli cli tokenauthentication enable
```

Generates a token

```
tscli cli tokenauthentication disable
```

Purges token login configuration

Date and time formats reference

This is a reference 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 an 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 through `tsload`

When loading through 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 through 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 10:32:22.0 AM
- 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 through an ETL tool

Data loads through ETL uses ODBC or JDBC connections. After you extract the data from the source but before you load it into ThoughtSpot, you must transform any date or timestamp columns into a format that is valid for ThoughtSpot. After the data transformation completes, there is no requirement for explicit data masking. See the data integration guide for more information on loading data through 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
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

Format mask	Description
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> <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>

Function	Description	Examples
to_string	Returns the input as a text string. To convert a date to a string, specify the date format you want to use.	<code>to_string (45 + 1) = '46'</code> <code>to_string (revenue - cost)</code> <code>to_string (date, ('%m/%d/%y'))</code>

Date functions

Function	Description	Examples
add_days	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>
add_minutes	Returns the result of adding the specified number of minutes to input date/date-time/time.	<code>add_minutes (01/30/2015 00:10:20, 5) = 01/30/2015 00:11:20</code> <code>add_minutes (invoiced, 30)</code>
add_months	Returns the result of adding the specified number of months to the given date.	<code>add_months (01/30/2015, 5) = 06/30/2015</code> <code>add_months (invoiced_date, 5)</code>
add_seconds	Returns the result of adding the specified number of seconds to the given date.	<code>add_seconds (01/30/2015 00:00:00, 5) = 06/30/2015 00:00:05</code> <code>add_seconds (invoiced_date, 5)</code>
add_weeks	Returns the result of adding the specified number of weeks to the given date.	<code>add_weeks (01/30/2015, 2) = 02/13/2015</code> <code>add_weeks (invoiced_date, 2)</code>
add_years	Returns the result of adding the specified number of years to the given date.	<code>add_years (01/30/2015, 5) = 01/30/2020</code> <code>add_years (invoiced_date, 5)</code>
date	Returns the date portion of a given date.	<code>date (home_visit)</code>

Function	Description	Examples
day	Returns the number (1-31) of the day for the given date.	day (01/15/2014) = 15 day (date ordered)
day_number_of_quarter	Returns the number of the day in a quarter for a given date. Add an optional second parameter to specify whether a 'fiscal' or 'calendar' year is used to calculate the result. Default is 'calendar'. (In examples, start of fiscal year is set to May 01.)	day_number_of_quarter (01/30/2015) = 30 day_number_of_quarter (01/30/2015, 'fiscal') = 91
day_number_of_week	Returns the number (1-7) of the day in a week for a given date with 1 being Monday and 7 being Sunday.	day_number_of_week(01/15/2014) = 3 day_number_of_week (shipped)
day_number_of_year	Returns the number (1-366) of the day in a year from a given date. Add an optional second parameter to specify whether a 'fiscal' or 'calendar' year is used to calculate the result. Default is 'calendar'. (In examples, start of fiscal year is set to May 01.)	day_number_of_year (01/30/2015) = 30 day_number_of_year (01/30/2015, 'fiscal') = 275 day_number_of_year (invoiced)
day_of_week	Returns the day of the week for the given date.	day_of_week (01/30/2015) = Friday day_of_week (serviced)
diff_days	Subtracts the second date from the first date and returns the result in number of days, rounded down if not exact.	diff_days (01/15/2014, 01/17/2014) = -2 diff_days (purchased, shipped)

Function	Description	Examples
diff_time	Subtracts the second date from the first date and returns the result in number of seconds.	diff_time (01/01/2014, 01/01/2014) = -86,400 diff_time (clicked, submitted)
hour_of_day	Returns the hour of the day for the given date.	hour_of_day (received)
is_weekend	Returns true if the given date falls on a Saturday or Sunday.	is_weekend (01/31/2015) = true is_weekend (emailed)
month	Returns the month from the given date.	month (01/15/2014) = January month (date ordered)
month_number	Returns the number (1-12) of the month from a given date. Add an optional second parameter to specify whether a 'fiscal' or 'calendar' year is used to calculate the result. Default is 'calendar'. (In examples, start of fiscal year is set to May 01.)	month_number (09/20/2014) = 9 month_number (09/20/2014, 'fiscal') = 5 month_number (purchased)
month_number_of_quarter	Returns the month (1-3) number for the given date in a quarter. Add an optional second parameter to specify whether a 'fiscal' or 'calendar' year is used to calculate the result. Default is 'calendar'. (In examples, start of fiscal year is set to May 01.)	month_number_of_quarter (02/20/2018) = 2 month_number_of_quarter (02/20/2018, 'fiscal') = 1
now	Returns the current timestamp.	now ()

Function	Description	Examples
quarter_number	Returns the number (1-4) of the quarter associated with the given date. Add an optional second parameter to specify 'fiscal' or 'calendar' dates. Default is 'calendar'. (In examples, start of fiscal year is set to May 01.)	<code>quarter_number (04/14/2014) = 2</code> <code>quarter_number (04/14/2014, 'fiscal') = 4</code> <code>quarter_number (shipped)</code>
start_of_month	Returns MMM yyyy for the first day of the month. Your installation configuration can override this setting so that it returns a different format such as MM/dd/yyyy . 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>
start_of_quarter	Returns the date for the first day of the quarter for the given date. Add an optional second parameter to specify whether a 'fiscal' or 'calendar' year is used to calculate the result. Default is 'calendar'. (In examples, start of fiscal year is set to May 01.)	<code>start_of_quarter (04/01/2014) = Apr 2014</code> <code>start_of_quarter (04/01/2014, 'fiscal') = Feb 2014</code> <code>start_of_quarter (sold)</code>
start_of_week	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
start_of_year	Returns the date for the first day of the year for the given date. Add an optional second parameter to specify whether a 'fiscal' or 'calendar' year is used to calculate the result. Default is 'calendar'. (In examples, start of fiscal year is set to May 01.)	start_of_year (04/01/2014) returns Jan 2014 start_of_year (04/01/2014, 'fiscal') returns May 2013 start_of_year (joined)
time	Returns the time portion of a given date.	time (3/1/2002 10:32) = 10:32 time (call began)
week_number_of_month	Returns the week number for the given date in a month.	week_number_of_month(03/23/2017) = 3
week_number_of_quarter	Returns the week number for the given date in a quarter. Add an optional second parameter to specify whether a 'fiscal' or 'calendar' year is used to calculate the result. Default is 'calendar'. (In examples, start of fiscal year is set to May 01.)	week_number_of_quarter (04/03/2017) = 1 week_number_of_quarter (04/03/2017, 'fiscal') = 10
week_number_of_year	Returns the week number for the given date in a year. Add an optional second parameter to specify whether a 'fiscal' or 'calendar' year is used to calculate the result. Default is 'calendar'. (In examples, start of fiscal year is set to May 01.)	week_number_of_year (01/17/2014) = 3 week_number_of_year (01/17/2014, 'fiscal') = 38

Function	Description	Examples
year	Returns the year from a given date. Add an optional second parameter to specify whether a 'fiscal' or 'calendar' year is used to calculate the result. Default is 'calendar'. (In examples, start of fiscal year is set to May 01. Per standard convention, the fiscal year is defined by the year-end date.)	year (01/15/2014) = 2014 year (12/15/2013, 'fiscal') = 2014 year (date ordered)

Mixed functions

These functions can be used with text and numeric data types.

Function	Description	Examples
!=	Returns true if the first value is not equal to the second value.	3 != 2 = true revenue != 1000000
<	Returns true if the first value is less than the second value.	3 < 2 = false revenue < 1000000
<=	Returns true if the first value is less than or equal to the second value.	1 <= 2 = true revenue <= 1000000
=	Returns true if the first value is equal to the second value.	2 = 2 = true revenue = 1000000
>	Returns true if the first value is greater than the second value.	3 > 2 = true revenue > 1000000
>=	Returns true if the first value is greater than or equal to the second value.	3 >= 2 = true revenue >= 1000000
greatest	Returns the larger of the values.	greatest (20, 10) = 20 greatest (q1 revenue, q2 revenue)
least	Returns the smaller of the values.	least (20, 10) = 10 least (q1 revenue, q2 revenue)

Number functions

Function	Description	Examples
*	Returns the result of multiplying both numbers.	$3 * 2 = 6$ price * taxrate
+	Returns the result of adding both numbers.	$1 + 2 = 3$ price + shipping
-	Returns the result of subtracting the second number from the first.	$3 - 2 = 1$ revenue - tax
/	Returns the result of dividing the first number by the second.	$6 / 3 = 2$ markup / retail price
[^]	Returns the first number raised to the power of the second.	$3 ^ 2 = 9$ width ^ 2
abs	Returns the absolute value.	abs (-10) = 10 abs (profit)
acos	Returns the inverse cosine in degrees.	acos (0.5) = 60 acos (cos-satellite-angle)
asin	Returns the inverse sine (specified in degrees).	asin (0.5) = 30 asin (sin-satellite-angle)
atan	Returns the inverse tangent in degrees.	atan (1) = 45 atan (tan-satellite-angle)
atan2	Returns the inverse tangent in degrees.	atan2 (10, 10) = 45 atan2 (longitude, latitude)
cbrt	Returns the cube root of a number.	cbrt (27) = 3 cbrt (volume)
ceil	Returns the smallest following integer.	ceil (5.9) = 6 ceil (growth rate)
cos	Returns the cosine of an angle (specified in degrees).	cos (63) = 0.45 cos (beam angle)
cube	Returns the cube of a number.	cube (3) = 27 cube (length)
exp	Returns Euler's number (~2.718) raised to a power.	exp (2) = 7.38905609893 exp (growth)

Function	Description	Examples
exp2	Returns 2 raised to a power.	exp2 (3) = 8 exp2 (growth)
floor	Returns the largest previous integer.	floor (5.1) = 5 floor (growth rate)
ln	Returns the natural logarithm.	ln (7.38905609893) = 2 ln (distance)
log10	Returns the logarithm with base 10.	log10 (100) = 2 log10 (volume)
log2	Returns the logarithm with base 2 (binary logarithm).	log2 (32) = 5 log2 (volume)
mod	Returns the remainder of first number divided by the second number.	mod (8, 3) = 2 mod (revenue, quantity)
pow	Returns the first number raised to the power of the second number.	pow (5, 2) = 25 pow (width, 2)
random	Returns a random number between 0 and 1.	random () = .457718 random ()
round	Returns the first number rounded to the second number (the default is 1).	round (35.65, 10) = 40 round (battingavg, 100) round (48.67, .1) = 48.7
safe_divide	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).	safe_divide (12, 0) = 0 safe_divide (total_cost, units)
sign	Returns +1 if the number is greater than zero, -1 if less than zero, 0 if zero.	sign (-250) = -1 sign (growth rate)
sin	Returns the sine of an angle (specified in degrees).	sin (35) = 0.57 sin (beam angle)
spherical_distance	Returns the distance in km between two points on Earth.	spherical_distance (37.465191, -122.153617, 37.421962, -122.142174) = 4,961.96 spherical_distance (start_latitude, start_longitude, start_latitude, start_longitude)

Function	Description	Examples
sq	Returns the square of a numeric value.	sq (9) = 81 sq (width)
sqrt	Returns the square root.	sqrt (9) = 3 sqrt (area)
tan	Returns the tangent of an angle (specified in degrees).	tan (35) = 0.7 tan (beam angle)

Operators

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

Text functions

Function	Description	Examples
concat	Returns two or more values as a concatenated text string. Use single quotes around each literal string, not double quotes.	concat ('hay' , 'stack') = 'haystack' concat (title, ' ', first_name , ' ', last_name)
contains	Returns true if the first string contains the second string, otherwise returns false.	contains ('broomstick', 'room') = true contains (product, 'trial version')
edit_distance	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.	edit_distance ('attorney', 'atty') = 4 edit_distance (color, 'red')
edit_distance_with_cap	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.	edit_distance_with_cap ('pokemon go', 'minecraft pixelmon', 3) = 4 edit_distance_with_cap (event, 'burning man', 3)
similar_to	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.	similar_to ('hello world', 'hello swirl') = true similar_to (current team, drafted by)

Function	Description	Examples
similarity	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.	similarity ('where is the burning man concert', 'burning man') = 46 similarity (tweet1, tweet2)
spells_like	Accepts two text strings. Returns true if they are spelled similarly and false if they are not. Works with strings under 1023 characters.	spells_like ('thouhgtspot', 'thoughtspot') = true spells_like (studio, distributor)
strlen	Returns the length of the text.	strlen ('smith') = 5 strlen (lastname)
strpos	Returns the numeric position (starting from 0) of the first occurrence of the second string in the first string, or -1 if not found.	strpos ('haystack_with_needles', 'needle') = 14 strpos (complaint, 'lawyer')
substr	Returns the portion of the given string, beginning at the location specified (starting from 0), and of the given length.	substr ('persnickety', 3, 7) = snicket substr (lastname, 0, 5)

Variables

These variables can be used in your expressions.

Function	Description	Examples
ts_groups	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.	ts_groups = 'east'
ts_username	Returns the user with the matching neame.	ts_username != 'mark'

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

Aggregate functions (group aggregate)

These functions can be used to aggregate data.

Function	Description	Examples
average	Returns the average of all the values of a column.	average (revenue)
average_if	Returns the average of all the columns that meet a given criteria.	average_if(city = "San Francisco", revenue)
count	Returns the number of rows in the table containing the column.	count (product)
count_if	Returns the number of rows in the table containing the column.	count_if(region = 'west', region)
cumulative_average	Takes a measure and one or more attributes. Returns the average of the measure, accumulated by the attribute(s) in the order specified.	cumulative_average (revenue, order date, state)
cumulative_max	Takes a measure and one or more attributes. Returns the maximum of the measure, accumulated by the attribute(s) in the order specified.	cumulative_max (revenue, state)
cumulative_min	Takes a measure and one or more attributes. Returns the minimum of the measure, accumulated by the attribute(s) in the order specified.	cumulative_min (revenue, campaign)
cumulative_sum	Takes a measure and one or more attributes. Returns the sum of the measure, accumulated by the attribute(s) in the order specified.	cumulative_sum (revenue, order date)

Function	Description	Examples
group_aggregate	<p>Takes a measure and, optionally, attributes and filters. These can be used to aggregate measures with granularities and filters different from the terms/columns used in the search. Especially useful for comparison analysis.</p> <p>This formula takes the form: group_aggregate (< aggregation (measure) >, < groupings >, < filters >)</p> <p>Lists can be defined with {} and optional list functions query_groups or query_filters , which by default specify the lists or filters used in the original search. Plus (+) or (-) can be used to add or exclude specific columns for query groups.</p>	<pre>group_aggregate (sum (revenue) , {ship mode, date} , {})</pre> <pre>group_aggregate (sum (revenue) , {ship mode , date}, {day_of_week (date) = 'friday'})</pre> <pre>group_aggregate (sum (revenue) , query_groups() , query_filters())</pre> <pre>group_aggregate (sum (revenue) , query_groups() + {date} , query_filters())</pre>
group_average	Takes a measure and one or more attributes. Returns the average of the measure grouped by the attribute(s).	group_average (revenue, customer region, state)
group_count	Takes a measure and one or more attributes. Returns the count of the measure grouped by the attribute(s).	group_count (revenue, customer region)
group_max	Takes a measure and one or more attributes. Returns the maximum of the measure grouped by the attribute(s).	group_max (revenue, customer region)
group_min	Takes a measure and one or more attributes. Returns the minimum of the measure grouped by the attribute(s).	group_min (revenue, customer region)
group_stddev	Takes a measure and one or more attributes. Returns the standard deviation of the measure grouped by the attribute(s).	group_stddev (revenue, customer region)
group_sum	Takes a measure and one or more attributes. Returns the sum of the measure grouped by the attribute(s).	group_sum (revenue, customer region)
group_unique_count	Takes a measure and one or more attributes. Returns the unique count of the measure grouped by the attribute(s).	group_unique_count (product , supplier)

Function	Description	Examples
group_variance	Takes a measure and one or more attributes. Returns the variance of the measure grouped by the attribute(s).	group_variance (revenue, customer region)
max	Returns the maximum value of a column.	max (sales)
max_if	Returns the maximum value among columns that meet a criteria.	max_if((revenue > 10) , customer region)
min	Returns the minimum value of a column.	min (revenue)
min_if	Returns the minimum value among columns that meet a criteria.	min_if((revenue < 10) , customer region)
moving_average	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.	moving_average (revenue, 2, 1, customer region)
moving_max	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.	moving_max (complaints, 1, 2, store name)

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>rank</code>	Returns the rank for the current row. Identical values receive an identical rank. Takes an aggregate input for the first argument. The second argument specifies the order, 'asc' 'desc' .	<code>rank (sum (revenue) , 'asc')</code> <code>rank (sum (revenue) , 'desc')</code>
<code>rank_percentile</code>	Returns the percentile rank for the current row. Identical values are assigned an identical percentile rank. Takes an aggregate input for the first argument. The second argument specifies the order, 'asc' 'desc' .	<code>rank_percentile (sum (revenue) , 'asc')</code> <code>rank_percentile (sum (revenue) , 'desc')</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 > 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>

Function	Description	Examples
unique_count	Returns the number of unique values of a column.	unique_count (customer)
unique_count_if	Returns the number of unique values of a column provided it meets a criteria.	unique_count_if((revenue > 10) , order date)
variance	Returns the variance of all the values of a column.	variance (revenue)
variance_if	Returns the variance of all the values of a column provided it meets a criteria..	variance_if((revenue > 10) , (revenue/10.0))

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
to_bool	Returns the input as a boolean (true or false).	to_bool (0) = false to_bool (married)
to_date	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.	to_date (date_sold, '%Y-%m-%d')
to_double	Returns the input as a double.	to_double ('3.14') = 3.14 to_double (revenue * .01)
to_integer	Returns the input as an integer.	to_integer ('45') + 1 = 46 to_integer (price + tax - cost)
to_string	Returns the input as a text string. To convert a date to a string, specify the date format you want to use.	to_string (45 + 1) = '46' to_string (revenue - cost) to_string (date, ('%m/%d/%y'))

Date functions

Function	Description	Examples
add_days	Returns the result of adding the specified number of days to the given date.	add_days (01/30/2015, 5) = 02/04/2015 add_days (invoiced, 30)
add_minutes	Returns the result of adding the specified number of minutes to input date/date-time/time.	add_minutes (01/30/2015 00:10:20, 5) = 01/30/2015 00:11:20 add_minutes (invoiced, 30)
add_months	Returns the result of adding the specified number of months to the given date.	add_months (01/30/2015, 5) = 06/30/2015 add_months (invoiced_date, 5)
add_seconds	Returns the result of adding the specified number of seconds to the given date.	add_seconds (01/30/2015 00:00:00, 5) = 06/30/2015 00:00:05 add_seconds (invoiced_date, 5)
add_weeks	Returns the result of adding the specified number of weeks to the given date.	add_weeks (01/30/2015, 2) = 02/13/2015 add_weeks (invoiced_date, 2)
add_years	Returns the result of adding the specified number of years to the given date.	add_years (01/30/2015, 5) = 01/30/2020 add_years (invoiced_date, 5)
date	Returns the date portion of a given date.	date (home visit)
day	Returns the number (1-31) of the day for the given date.	day (01/15/2014) = 15 day (date ordered)

Function	Description	Examples
day_number_of_quarter	Returns the number of the day in a quarter for a given date. Add an optional second parameter to specify whether a 'fiscal' or 'calendar' year is used to calculate the result. Default is 'calendar'. (In examples, start of fiscal year is set to May 01.)	day_number_of_quarter (01/30/2015) = 30 day_number_of_quarter (01/30/2015, 'fiscal') = 91
day_number_of_week	Returns the number (1-7) of the day in a week for a given date with 1 being Monday and 7 being Sunday.	day_number_of_week(01/15/2014) = 3 day_number_of_week (shipped)
day_number_of_year	Returns the number (1-366) of the day in a year from a given date. Add an optional second parameter to specify whether a 'fiscal' or 'calendar' year is used to calculate the result. Default is 'calendar'. (In examples, start of fiscal year is set to May 01.)	day_number_of_year (01/30/2015) = 30 day_number_of_year (01/30/2015, 'fiscal') = 275 day_number_of_year (invoiced)
day_of_week	Returns the day of the week for the given date.	day_of_week (01/30/2015) = Friday day_of_week (serviced)
diff_days	Subtracts the second date from the first date and returns the result in number of days, rounded down if not exact.	diff_days (01/15/2014, 01/17/2014) = -2 diff_days (purchased, shipped)
diff_time	Subtracts the second date from the first date and returns the result in number of seconds.	diff_time (01/01/2014, 01/01/2014) = -86,400 diff_time (clicked, submitted)

Function	Description	Examples
hour_of_day	Returns the hour of the day for the given date.	hour_of_day (received)
is_weekend	Returns true if the given date falls on a Saturday or Sunday.	is_weekend (01/31/2015) = true is_weekend (emailed)
month	Returns the month from the given date.	month (01/15/2014) = January month (date ordered)
month_number	Returns the number (1-12) of the month from a given date. Add an optional second parameter to specify whether a 'fiscal' or 'calendar' year is used to calculate the result. Default is 'calendar'. (In examples, start of fiscal year is set to May 01.)	month_number (09/20/2014) = 9 month_number (09/20/2014, 'fiscal') = 5 month_number (purchased)
month_number_of_quarter	Returns the month (1-3) number for the given date in a quarter. Add an optional second parameter to specify whether a 'fiscal' or 'calendar' year is used to calculate the result. Default is 'calendar'. (In examples, start of fiscal year is set to May 01.)	month_number_of_quarter (02/20/2018) = 2 month_number_of_quarter (02/20/2018, 'fiscal') = 1
now	Returns the current timestamp.	now ()

Function	Description	Examples
quarter_number	Returns the number (1-4) of the quarter associated with the given date. Add an optional second parameter to specify 'fiscal' or 'calendar' dates. Default is 'calendar'. (In examples, start of fiscal year is set to May 01.)	<code>quarter_number (04/14/2014) = 2</code> <code>quarter_number (04/14/2014, 'fiscal') = 4</code> <code>quarter_number (shipped)</code>
start_of_month	Returns MMM yyyy for the first day of the month. Your installation configuration can override this setting so that it returns a different format such as MM/dd/yyyy . 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>
start_of_quarter	Returns the date for the first day of the quarter for the given date. Add an optional second parameter to specify whether a 'fiscal' or 'calendar' year is used to calculate the result. Default is 'calendar'. (In examples, start of fiscal year is set to May 01.)	<code>start_of_quarter (04/01/2014) = Apr 2014</code> <code>start_of_quarter (04/01/2014, 'fiscal') = Feb 2014</code> <code>start_of_quarter (sold)</code>
start_of_week	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
start_of_year	Returns the date for the first day of the year for the given date. Add an optional second parameter to specify whether a 'fiscal' or 'calendar' year is used to calculate the result. Default is 'calendar'. (In examples, start of fiscal year is set to May 01.)	start_of_year (04/01/2014) returns Jan 2014 start_of_year (04/01/2014, 'fiscal') returns May 2013 start_of_year (joined)
time	Returns the time portion of a given date.	time (3/1/2002 10:32) = 10:32 time (call began)
week_number_of_month	Returns the week number for the given date in a month.	week_number_of_month(03/23/2017) = 3
week_number_of_quarter	Returns the week number for the given date in a quarter. Add an optional second parameter to specify whether a 'fiscal' or 'calendar' year is used to calculate the result. Default is 'calendar'. (In examples, start of fiscal year is set to May 01.)	week_number_of_quarter (04/03/2017) = 1 week_number_of_quarter (04/03/2017, 'fiscal') = 10
week_number_of_year	Returns the week number for the given date in a year. Add an optional second parameter to specify whether a 'fiscal' or 'calendar' year is used to calculate the result. Default is 'calendar'. (In examples, start of fiscal year is set to May 01.)	week_number_of_year (01/17/2014) = 3 week_number_of_year (01/17/2014, 'fiscal') = 38

Function	Description	Examples
year	Returns the year from a given date. Add an optional second parameter to specify whether a 'fiscal' or 'calendar' year is used to calculate the result. Default is 'calendar'. (In examples, start of fiscal year is set to May 01. Per standard convention, the fiscal year is defined by the year-end date.)	year (01/15/2014) = 2014 year (12/15/2013, 'fiscal') = 2014 year (date ordered)

Mixed functions

These functions can be used with text and numeric data types.

Function	Description	Examples
!=	Returns true if the first value is not equal to the second value.	3 != 2 = true revenue != 1000000
<	Returns true if the first value is less than the second value.	3 < 2 = false revenue < 1000000
<=	Returns true if the first value is less than or equal to the second value.	1 <= 2 = true revenue <= 1000000
=	Returns true if the first value is equal to the second value.	2 = 2 = true revenue = 1000000
>	Returns true if the first value is greater than the second value.	3 > 2 = true revenue > 1000000
>=	Returns true if the first value is greater than or equal to the second value.	3 >= 2 = true revenue >= 1000000
greatest	Returns the larger of the values.	greatest (20, 10) = 20 greatest (q1 revenue, q2 revenue)
least	Returns the smaller of the values.	least (20, 10) = 10 least (q1 revenue, q2 revenue)

Number functions

Function	Description	Examples
*	Returns the result of multiplying both numbers.	$3 * 2 = 6$ price * taxrate
+	Returns the result of adding both numbers.	$1 + 2 = 3$ price + shipping
-	Returns the result of subtracting the second number from the first.	$3 - 2 = 1$ revenue - tax
/	Returns the result of dividing the first number by the second.	$6 / 3 = 2$ markup / retail price
[^]	Returns the first number raised to the power of the second.	$3 ^ 2 = 9$ width ^ 2
abs	Returns the absolute value.	abs (-10) = 10 abs (profit)
acos	Returns the inverse cosine in degrees.	acos (0.5) = 60 acos (cos-satellite-angle)
asin	Returns the inverse sine (specified in degrees).	asin (0.5) = 30 asin (sin-satellite-angle)
atan	Returns the inverse tangent in degrees.	atan (1) = 45 atan (tan-satellite-angle)
atan2	Returns the inverse tangent in degrees.	atan2 (10, 10) = 45 atan2 (longitude, latitude)
cbrt	Returns the cube root of a number.	cbrt (27) = 3 cbrt (volume)
ceil	Returns the smallest following integer.	ceil (5.9) = 6 ceil (growth rate)
cos	Returns the cosine of an angle (specified in degrees).	cos (63) = 0.45 cos (beam angle)
cube	Returns the cube of a number.	cube (3) = 27 cube (length)
exp	Returns Euler's number (~2.718) raised to a power.	exp (2) = 7.38905609893 exp (growth)

Function	Description	Examples
exp2	Returns 2 raised to a power.	exp2 (3) = 8 exp2 (growth)
floor	Returns the largest previous integer.	floor (5.1) = 5 floor (growth rate)
ln	Returns the natural logarithm.	ln (7.38905609893) = 2 ln (distance)
log10	Returns the logarithm with base 10.	log10 (100) = 2 log10 (volume)
log2	Returns the logarithm with base 2 (binary logarithm).	log2 (32) = 5 log2 (volume)
mod	Returns the remainder of first number divided by the second number.	mod (8, 3) = 2 mod (revenue, quantity)
pow	Returns the first number raised to the power of the second number.	pow (5, 2) = 25 pow (width, 2)
random	Returns a random number between 0 and 1.	random () = .457718 random ()
round	Returns the first number rounded to the second number (the default is 1).	round (35.65, 10) = 40 round (battingavg, 100) round (48.67, .1) = 48.7
safe_divide	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).	safe_divide (12, 0) = 0 safe_divide (total_cost, units)
sign	Returns +1 if the number is greater than zero, -1 if less than zero, 0 if zero.	sign (-250) = -1 sign (growth rate)
sin	Returns the sine of an angle (specified in degrees).	sin (35) = 0.57 sin (beam angle)
spherical_distance	Returns the distance in km between two points on Earth.	spherical_distance (37.465191, -122.153617, 37.421962, -122.142174) = 4,961.96 spherical_distance (start_latitude, start_longitude, start_latitude, start_longitude)

Function	Description	Examples
sq	Returns the square of a numeric value.	sq (9) = 81 sq (width)
sqrt	Returns the square root.	sqrt (9) = 3 sqrt (area)
tan	Returns the tangent of an angle (specified in degrees).	tan (35) = 0.7 tan (beam angle)

Text functions

Function	Description	Examples
concat	Returns two or more values as a concatenated text string. Use single quotes around each literal string, not double quotes.	concat ('hay', 'stack') = 'haystack' concat (title, ' ', first_name, ' ', last_name)
contains	Returns true if the first string contains the second string, otherwise returns false.	contains ('broomstick', 'room') = true contains (product, 'trial version')
edit_distance	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.	edit_distance ('attorney', 'atty') = 4 edit_distance (color, 'red')
edit_distance_with_cap	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.	edit_distance_with_cap ('pokemon go', 'minecraft pixelmon', 3) = 4 edit_distance_with_cap (event, 'burning man', 3)

Function	Description	Examples
similar_to	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.	similar_to ('hello world', 'hello swirl') = true similar_to (current team, drafted by)
similarity	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.	similarity ('where is the burning man concert', 'burning man') = 46 similarity (tweet1, tweet2)
spells_like	Accepts two text strings. Returns true if they are spelled similarly and false if they are not. Works with strings under 1023 characters.	spells_like ('thouhgtspot', 'thoughtspot') = true spells_like (studio, distributor)
strlen	Returns the length of the text.	strlen ('smith') = 5 strlen (lastname)
strpos	Returns the numeric position (starting from 0) of the first occurrence of the second string in the first string, or -1 if not found.	strpos ('haystack_with_needles', 'needle') = 14 strpos (complaint, 'lawyer')
substr	Returns the portion of the given string, beginning at the location specified (starting from 0), and of the given length.	substr ('persnickety', 3, 7) = snicket substr (lastname, 0, 5)

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.

answer_unsaved	User makes a change to tokens in the search bar.
answer_saved	User opens an existing saved answer and makes changes to tokens in the search bar.
answer_pinboard_context	User opens an existing saved pinboard, edits a context viz and makes a change to tokens in the search bar.
answer_aggregated_worksheet	User opens an existing saved aggregated worksheet and makes changes to tokens in the search bar.
answer_upgrade	Requests made for the sole purpose of upgrade.
pinboard_view	User opens an existing saved pinboard.
pinboard_filter	User adds, removes or applies values to a pinboard filter.
pinboard_ad_hoc	User drills down in a pinboard viz.
data_chart_config	Request for new data being generated following a chart config change.
data_show_underlying_row	Request to show underlying data for a data row(s).
data_export	Request to export data.
pinboard_tspublic_runtime_filter	Request to TSPublic/pinboarddata with runtime filters.
answer_aggregated_worksheet_save	User updates aggregated worksheet.
answer_add_new_filter	User adds a filter using the UI.
data_show_underlying_viz	Request to show underlying data for a data row(s).
answer_view	User opens an existing, saved answer.

answer_viz_context_view	User opens an existing saved pinboard, edits a context viz.
pinboard_insight_view	User opens SpotIQ tab pinboards.
pinboard_admin_view	User opens admin tab pinboards.
pinboard_embed_view	User opens embed pinboard from a URL.
pinboard_homepage_view	On loading of homepage pinboard.
pinboard_learn_view	On loading learn pinboard.
pinboard_tspublic_no_runtime_filter	Request to TSPublic/pinboard data without run-time filters.

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** — make sure it is `1`.

To learn more about modeling data see [modeling data](#) in this documentation.