THOUGHTSPOT

Reference Guide

Version 4.4, December 06, 2017

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

Table of Contents

About this reference

This Reference section contains the commands and their syntax for all the command line tools in ThoughtSpot.

Included in this guide are:

- Keyword reference (page 3) lists the available keywords to use in your search. These are also listed in the Help Center, which is available from **Help** on the top navigation bar in ThoughtSpot.
- tscli command reference (page 24) lists the ThoughtSpot Command Line Interface commands.
- Formula reference (page 50) lists the available formula operators and functions. These are also listed in the Formula Assistant, which is available from the place in ThoughtSpot where you build formulas.
- Date and time formats reference (page 39) lists the accepted date, time, and timestamp formats that you can use when uploading data through the Web interface or using the ThoughtSpot Loader.
- Row level security rules reference (page 42) lists the operators for building row level security rules.
- TQL reference (page 14) lists the SQL commands that are supported in TQL.
- ThoughtSpot Loader flag reference (page 21) lists the options for loading data with tsload.

Keyword reference

You can use keywords when asking a question to help define your search. This reference lists the various keywords. You can also see this list of keywords and examples from within the help center.

General keywords

Keyword	Examples
top	• `top sales rep by count sales for average revenue >10000` • `sales rep average revenue for each region top`
bottom	• `bottom revenue average 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	• `revenue by state sort by average revenue` • `revenue by customer sort by region`

Date keywords

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`
day of week	• `revenue by day of week last 6 months`
week	• `revenue by week last quarter`
month	• `revenue by month last year`
daily	• `shipments by region daily`
weekly	• `revenue weekly`
monthly	• `commission > 10000 monthly`
quarterly	• `sales quarterly for each product`
yearly	• `shipments by product yearly`
day of week</var>	• `count shipments Monday`

Keyword	Examples
month</var>	• `commission January`
month year</var>	• `commission by sales rep February 2014`
year</var>	• `revenue by product 2014 product name contains snowboard`
yesterday	• `sales yesterday for pro -ski200 by store`
week to date	• `sales by order date week to date for pro-ski200`
month to date	• `sales by product month to date sales > 2400`
quarter to date	• `sales by product quarter to date for top 10 products by sales`
year to date	• `sales by product year to date`
last day	• `customers last day by referrer`
last week	• `customers last week by store`
last month	• `customers last month by day`
last quarter	• `customers last quarter sale >300`

Keyword	Examples
last year	• `top 10 customers last year by sale by store for region west`
n days	• `visitors last 7 days`
n weeks	• `visitors last 10 weeks by day`
n months	• `visitors last 6 months for homepage visits > 30 by month`
n quarters	• `visitors last 2 quarters by month by campaign`
n years	• `visitors last 5 years by revenue for sum revenue >5000`
growth of … by …	• `growth of sales by order date`
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 … weekly	• `growth of sales by receipt date weekly for proski2000`
growth of … by … quarterly	• `growth of sales by date shipped quarterly`

Keyword	Examples
growth of … by … yearly	• `growth of sales by date closed yearly`
daily year-over-year	• `growth of revenue by order date daily year-over-year`
weekly year-over-year	• `growth of revenue by date shipped weekly year-over-year`
monthly year-over-year	• `growth of revenue by receipt date monthly year-over-year`
quarterly year-over-year	• `growth of revenue by date shipped quarterly year-over-year`
n days ago	• `sales 2 days ago`
n weeks ago	• `sales 4 weeks ago by store`
n months ago	• `sales 2 months ago by region`
n quarters ago	• `sales 4 quarters ago by product name contains deluxe`
n years ago	• `sales 5 years ago by store for region west`

Keyword	Examples
today	• `sales today by store`
next day	• `shipments next day by order`
next week	• `shipments next week by store`
next month	• `appointments next month by day`
next quarter	• `opportunities next quarter amount > 30000`
next year	• `opportunities next year by sales rep`
n days	• `shipments next 7 days`
n weeks	• `shipments next 10 weeks by day`
n months	• `openings next 6 months location`
n quarters	• `opportunities next 2 quarters by campaign`
n years	• `opportunities next 5 years by revenue`

Time keywords

Keyword	Examples
detailed	• `ship time detailed`

Keyword	Examples
last minute	• `count homepage views last minute`
last hour	• `count unique visits last hour`
n minutes	• `count visitors last 30 minutes`
n hours	• `count visitors last 12 hours`
hourly	• `visitors by page name hourly`
n minutes ago	• `sum inventory by product 10 minutes ago`
n hours ago	• `sum inventory by product by store 2 hours ago`

Text keywords

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 keywords

Function	Examples
sum	• `sum revenue`
average	• `average revenue by store`
count	• `count visitors by site`
max	• `max sales by visitor by site`
min	• `min revenue by store by campaign for cost > 5000`
standard deviation	• `standard deviation revenue by product by month for date after 10/31/2010`
unique count	• `unique count visitor by product page last week`

Function	Examples
variance	• `variance sale amount by visitor by product for last year`

Filter keywords

Function	Examples
between… and	• `revenue between 0 and 1000`
>	• `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 keywords

Keyword	Examples
near	• `revenue store name county near san francisco`
near... within *n* miles|km|meters	• `revenue store name county near alameda within 50 miles`
farther than *n* miles|km|meters from	• `average hours worked branch farther than 80 km from scarborough`

Location keywords only work for searches where the data source includes latitude/longitude data.

Period keywords

Keyword	Example
quarter (*date*)	`quarter (commit date)</td> </tr>`
month of quarter (*date*)	`month of quarter (commit date)</td> </tr>`
week of year (*date*)	`week of year (commit date)</td> </tr>`
week of quarter (*date*)	`week of quarter (commit date)</td> </tr>`
week of month (*date*)	`week of month (commit date)</td> </tr>`
day of year (*date*)	`day of year (commit date)</td> </tr>`
day of quarter (*date*)	`day of quarter (commit date)</td> </tr>`

day (*date*)	`day (order date)</td>` `  </tr>`
day of week (*date*)	`day of week (order date)</td>` `  </tr>`

hour (*datetime*)	`hour (timestamp)</td>`
	`  </tr>`
	`</table>`

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.

TQL commands

You can use TQL either through the ThoughtSpot application's web interface (page 0) or the command line interface (page 14) in the Linux shell.

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.

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.

When referring to objects using fully qualified object names, the syntax is:

```
"database"."schema"."table"
```

As a best practice, you should enclose object names (database, schema, table, and column) in double quotes, and column values in single quotes.

Basic commands

Syntax	Description	Examples
`help`	Displays command help.	`TQL> help`

View schemas and data

Syntax	Description	Examples
`SHOW DATABASES`	Lists all available databases.	`TQL> 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.	`TQL> USE "fruit_database";`
`SHOW SCHEMAS`	Lists all schemas within the current database.	`TQL> SHOW SCHEMAS;`
`SHOW TABLES`	Lists all tables within the current database by schema.	`TQL> SHOW TABLES;`
`SHOW TABLE <table>`	Lists all the columns for a table.	`TQL> SHOW TABLE "locations";`

Syntax	Description	Examples
`SCRIPT SERVER`	Generates the TQL schema for all tables in all databases on the server.	`TQL> SCRIPT SERVER;`
`SCRIPT DATABASE <database>`	Generates the TQL schema for all tables in a database.	`TQL> SCRIPT DATABASE "fruit_database";`
`SCRIPT TABLE <table>`	Generates the TQL schema for a table.	`TQL> SCRIPT TABLE "vendor";`

`SELECT <cols_or_expr> FROM <table_list> [WHERE <predicates>] [GROUP BY <expr>] [ORDER BY <expr>]`	Shows specified set of table data.
	
	If you do not specify the TOP number of rows to select, the top 50 rows will be returned by default. The number of rows to return can be set using the TQL command line flag:
	
	`--query_results`
	
	`_apply_top_row_count`
	
	You can use the following aggregation functions:
	• sum
	• count
	• count distinct
	• stddev
	• avg
	• variance
	• min
	• max
	
	You can use the following date functions:
	• absyear
	• absmonth
	• absday
	• absquarter
	• date
	• time
		`TQL> SELECT TOP 10 "quantity" FROM "sales_fact";`
		
		`TQL> SELECT COUNT(*) FROM "vendor";`
		
		`TQL> SELECT "vendor", SUM("quantity") FROM "sales_fact" GROUP BY "vendor";`
		
		`TQL> SELECT "vendor", SUM("amount") FROM "vendor", "sales_fact" WHERE "sales_fact"."vendorid" = "vendor"."vendorid" AND "amount" > 100 GROUP BY "vendor" ORDER BY "amount" DESC;`
		
		`TQL> SELECT "vendor", SUM("quantity") FROM "sales_fact" GROUP BY "vendor" LIMIT 10;`

Schema creation

Syntax	Description	Examples
`CREATE DATABASE <database>`	Creates a database.	`TQL> CREATE DATABASE "fruit_database";`
`CREATE SCHEMA <schema>`	Creates a schema within the current database.	`TQL> CREATE SCHEMA "fruit_schema";`
`CREATE TABLE <table> (<column_definitions> [<constraints>])`	Creates a table with the specified column definitions and constraints. Use PARTITION BY HASH to shard a table across	`TQL> CREATE TABLE "vendor" ("vendorid" int, "name" varchar(255));`

Syntax	Description	Examples
[PARTITION BY HASH (<*number*>) [KEY ("<*column*>")]])	all nodes. If no KEY is specified, the table will be randomly sharded. Note that you can specify relationship constraints (FOREIGN KEY or RELATIONSHIP) in the CREATE TABLE statement. But it is recommended to define these using ALTER TABLE statements at the end of your TQL script, after creating your tables. This works better in scripts, because it guarantees that tables are created before they are referenced in the constraint definitions.	TQL> 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");

Schema modification

Syntax	Description	Examples
DROP DATABASE <*database*>	Drops a database and all of its schemas and tables.	TQL> DROP DATABASE "fruit_database";
DROP SCHEMA <*schema*>	Drops a schema within the current database, and drops all of the tables in the schema.	TQL> DROP SCHEMA "fruit_schema";
DROP TABLE <*table*>	Drops a table.	TQL> DROP TABLE "location";
ALTER TABLE <*table*> ADD|DROP|RENAME COLUMN <*column*>	Alters a table to add, drop, or rename a column. When you add a column to an existing table, you must provide a default value to use for existing rows.	TQL> ALTER TABLE "cart" ADD COLUMN "nickname" varchar(255) DEFAULT 'no nickname'; TQL> ALTER TABLE "cart" DROP COLUMN "nickname"; TQL> ALTER TABLE "cart" RENAME COLUMN "nickname" TO "shortname";
TRUNCATE TABLE <*table*>	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 TRUNCATE, or all dependent objects (worksheets and pinboards) in ThoughtSpot will become invalid.	TQL> TRUNCATE TABLE "location";

Syntax	Description	Examples

`ALTER TABLE <table> DROP CONSTRAINT 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.
		`TQL> ALTER TABLE "sales" DROP CONSTRAINT PRIMARY KEY;`
		
		`TQL> ALTER TABLE "sales" ADD CONSTRAINT PRIMARY KEY ("PO_number");`

`ALTER TABLE <table> DROP [FOREIGN KEY| RELATIONSHIP] <name>;`	Drops the named foreign key or relationship between two tables.	`TQL> ALTER TABLE "sales_fact" DROP FOREIGN KEY "FK_PO_number";`
		
		`TQL> ALTER TABLE "fruit_dim" DROP RELATIONSHIP "REL_dates";`

`ALTER TABLE <table> DROP [CONSTRAINT FOREIGN KEY [<table_name>] | RELATIONSHIP [WITH <table_name>];`	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.
	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:
	
	• the name of the referenced table, for a foreign key.
	• the name of the related table, for a relationship.
	
	If you drop a foreign key without specifying the referenced table, all foreign keys from the table you are altering will be dropped.
		`TQL> ALTER TABLE "shipments" DROP CONSTRAINT FOREIGN KEY "orders";`
		
		`TQL> ALTER TABLE "wholesale_buys" DROP RELATIONSHIP WITH "retail_sales";`
		
		`/* Drops all relationships that have wholesale_buys as a source. */`
		
		`TQL> ALTER TABLE "wholesale_buys" DROP RELATIONSHIP;`
		
		`/* Drops all foreign keys from wholesale_buys. */`
		
		`TQL> ALTER TABLE "wholesale_buys" DROP CONSTRAINT FOREIGN KEY;`

`ALTER TABLE <table> [SET DIMENSION | SET FACT [PARTITION BY HASH [(<shards>)] [KEY(<column>)]]]`	Changes the partitioning on a table by doing one of:
	
	• re-sharding a sharded table
	• changing a replicated table to a sharded table
	• changing a sharded table to a replicat-
		`TQL> ALTER TABLE "sales_fact" SET FACT PARTITION BY HASH (96) KEY ("PO_number");`
		
		`TQL> ALTER TABLE`

Syntax	Description	Examples
	ed (unsharded) table By default, ThoughtSpot does not shard dimension tables. To change the partitioning on a table, or to change a dimension table to a sharded table, use ALTER TABLE...SET FACT PARTITION BY HASH...; To make a sharded table into a dimension table (replicated on every node), use `ALTER TABLE...SET DIMENSION;` command.	`"fruit_dim" SET DIMENSION;`
`ALTER TABLE <table> MODIFY COLUMN <column> <new_data_type>;`	Changes the data type of a column. This can have implications on sharding and primary key behavior. See About data type conversion (page 0).	`TQL> ALTER TABLE fact100 MODIFY COLUMN product_id int;`

Modify Schema

Syntax	Description	Examples
`INSERT INTO <table> VALUES ...`	Inserts values into a table. Only use this for testing. Do not use INSERT on a production system.	`TQL> INSERT INTO "vendor" VALUES 'helen rose', 'jacob norse', 'eileen ruff', 'manny gates';`
`UPDATE <table> ... SET ... [WHERE ...]`	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.	`TQL> UPDATE "location" SET "borough" = 'staten island', "city" = 'new york' WHERE "borough" = 'staten isl' AND city = 'NY';`
`DELETE FROM <table> [WHERE...]`	Deletes rows from a table that match optionally provided predicates. Predicates have the form column = value connected by the AND keyword.	`TQL> DELETE FROM "vendor" WHERE "name" = 'Joey Smith' AND "vendorid" = '19463';`

Constraints and relationships

Constraints and relationships in ThoughtSpot are used to define the relationships between tables (i.e. how they can be joined). However, constraints are not enforced, as they would be in a transactional database. You can define the following constraints when creating a table with CREATE TABLE, or add them to an existing table using the ADD CONSTRAINT syntax:

Syntax	Description	Examples

PRIMARY KEY	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.
		`CREATE TABLE "schools" ( "schoolID" varchar(15), "schoolName" varchar(255), "schoolCity" varchar(55), "schoolState" varchar(55), "schoolNick" varchar(55), CONSTRAINT PRIMARY KEY ("schoolID") ) ;`
		
		`TQL> ALTER TABLE "cart" ADD CONSTRAINT PRIMARY KEY ("cart_id");`
		
		`TQL> ALTER TABLE "cart" DROP CONSTRAINT PRIMARY KEY "cart_id";`

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.
		`TQL> ALTER TABLE "batting" ADD CONSTRAINT "FK_player" FOREIGN KEY ("playerID") REFERENCES "players" ("playerID");`
		
		`TQL> ALTER TABLE "batting" ADD CONSTRAINT "FK_lg_team" FOREIGN KEY ("lgID" ,"teamID") REFERENCES "teams" ("lgID" ,"teamID");`
		
		`TQL> ALTER TABLE "shipment" ADD CONSTRAINT "FK_PO_vendor" FOREIGN KEY ("po_number", "vendor") REFERENCES "orders" ("po_number", "vendor");`
		
		`TQL> ALTER TABLE "shipment" DROP CONSTRAINT "FK_PO_vendor";`

RELATIONSHIP	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. 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.	`TQL> 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" <`

Syntax	Description	Examples
		"wholesale_buys"."expire_date"); TQL> ALTER TABLE "wholesale_buys" DROP RELATIONSHIP "REL_fruit";

Flags

The `--query_results_apply_top_row_count <number>` flag can be used with TQL to limit the number of result rows returned by a query. For example:

```
$ tql --query_results_apply_top_row_count 100
```

Data types

ThoughtSpot supports a simplified list of data types:

Syntax	Description	Examples
Character	• VARCHAR(*n*)	Specify the maximum number of characters, as in VARCHAR(255). The size limit is 1GB for VARCHAR values.
Floating point	• DOUBLE • FLOAT	DOUBLE is recommended.
Boolean	• BOOL	Can be `true` or `false`.
Integer	• INT • BIGINT	INT holds 32 bits. BIGINT holds 64 bits.
Date or time	• DATE • DATETIME • TIMESTAMP • TIME	DATETIME, TIMESTAMP, and TIME are stored at the granularity of seconds. TIMESTAMP is identical to DATETIME, but is included for syntax compatibility.

ThoughtSpot loader flag reference

For recurring data loads and for scripting loads, use the ThoughtSpot Loader (tsload). This reference section lists all the flags that can be used to modify the behavior of tsload.

General tsload flags

Flag	Description	Notes
`--target_database <database>`	Specifies the pre-existing target database into which tsload should load the data.	
`--target_schema <schema>`	Specifies the target schema.	Default is "falcon_default_schema".
`--target_table <table>`	Specifies the tables that you want to load into the database.	The tables must exist in the database specified by `--target_database`.
`--empty_target`	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 `--target_database` and `--target table` will be deleted before this data load. To perform an "upsert" on the existing data, omit this flag or specify `--noempty_target`.
`--max_ignored_rows <number>`	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.
`--bad_records_file <path_to_file>/<file_name>`	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.
`--date_format <date_formatmask>`	Specifies the format string for date values.	The default format is `yearmonthday` e.g. "Dec 30th, 2001" and is represented as `20011230`. Use the date format specifications supported in the strptime library function .
`--date_time_format <date_formatmask> <time_formatmask>`	Specifies the format string for datetime values.	The default is yearmonthday hour:minute:second e.g. Dec 30th, 2001 1:15:12 and is represented as 20011230 01:15:12. Use the datetime format specifications supported in the strptime library function .
`--time_format <time_formatmask>`	Specifies the format string for time values.	The default is hour:minute:second. Use the time format specifications supported in the strptime library function .
`--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

Flag	Description	Notes
		debugging. You should not change this value unless instructed by ThoughtSpot Support.
`--skip_second_fraction`	Skips fractional seconds when loading data.	If supplied, the upserts logic may be affected, especially if the date time being loaded is a primary key, and the data has millisecond granularity. Load the data twice, once as a string with a primary key, and again with second granularity date time. There is no support to store fractional seconds in the ThoughtSpot system.

File loading tsload flags

The following flags are used when loading data from an input file:

Flag	Description	Notes
`--source_file <path_to_file>/<file_name>`	Specifies the location of the file to be loaded.	
`--source_data_format [csv|delimited]`	Specifies the data file format.	Optional. The default is csv.
`--field_separator "<delimiter>"`	Specifies the field delimiter used in the input file.	
`--trailing_field_separator`	Specifies that the field separator appears after every field, including the last field per row.	Example row with trailing field separator: a,b,c,The default is false.
`--null_value "<null_representation>"`	Specifies how null values are represented in the input file.	These values will be converted to NULL upon loading.
`--date_converted_to_epoch [true|false]`	Specifies whether the “date” or “datetime” values in the input file are represented as epoch values.	
`--boolean_representation [true_false | 1_0 | T_F | Y_N]`	Specifies the format in which boolean values are represented in the input file.	The default is T_F. You can also use this flag to specify other values. For example, if your data used Y for true and NULL for false, you could specify:`--boolean_representation Y_NULL`
`--has_header_row`	Indicates that the input file contains a header row.	If supplied, the first row of the file is ignored. If not supplied, the first row of the file is loaded as data.
`--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:

Flag	Description	Notes
		`--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

The `tscli` command line interface is an administration interface for the ThoughtSpot instance. Use `tscli` to take snapshots (backups) of data, apply updates, stop and start the services, and view information about the system. This reference defines each subcommand and what you can accomplish with it.

The command returns 0 upon success and a non-zero exit code upon failure. Because the `tscli` command is typically running a command on multiple codes, an error may be called at different points. As much as possible, the command attempts to save errors to the `stderr` directory as configured on a node.

How to use the tscli command

The `tscli` command has the following syntax:

```
tscli [-h] [--helpfull] [--verbose] [--noautoconfig]
           [--autoconfig] [--yes] [--cluster <cluster>]
           [--zoo <zookeeper>] [--username username] [--identity_file identity_file]
           {alert,backup,backup-policy,callhome,cluster,command,dr-mirror,etl,event,
             feature,fileserver,firewall,hdfs,ldap,logs,map-tiles,monitoring,nas,node,

os,saml,scheduled-pinboards,smtp,snapshot,snapshot-policy,spot,ssl,storage,
             support}
```

The `tscli` command has several subcommands such as `alert`, `backup`, and so forth. You issue a subcommand using the following format:

```
tscli [subcommand ]
```

Subcommands have their own additional options and actions such as `tscli backup create` or `tscli backup delete` for example. To view help for a subcommand:

```
tscli [subcommand] -h
```

A subcommand itself may have several options.

tscli subcommands

This section lists each subcommand and its syntax.

alert subcommand

```
tscli alert [-h] {count,info,list,off,on,refresh,silence,status,unsilence}
```

Use this subcommand to do the following:

- `tscli alert info` Lists all alerts.
- `tscli alert list` Lists the generated alerts.
- `tscli alert off` Disables all alerts from the cluster.
- `tscli alert on` Enables alerts from the cluster.

- `tscli alert silence --name <alert_name>`

  Silences the alert with *`alert_name`*. For example, DISK_ERROR. Silenced alerts are still recorded in postgres, however emails are not sent out.

- `tscli alert status` Shows the status of cluster alerts.

- `tscli alert unsilence-name` *`alert_name`*

  Unsilences the alert with *alert_name*. For example, `DISK_ERROR`.

backup subcommand

```
tscli backup [-h] {create,delete,ls,restore}
```

Use this subcommand to do the following:

- `tscli backup create [-h] [--mode {full,light,dataless}] [--type {full,incremental}] [--base BASE] [--storage_type {local,nas}] [--remote] name out`

  Pulls a snapshot and saves it as a backup where:

  - `--mode {full,light,dataless}`

    Mode of backups. To understand these diffrent modes see Understand backup modes (page 0).

  - `--type {full,incremental}` Type of backup.(Incremental `incremental` is not implemented yet) (default: full)

  - `--base BASE`

    Based snapshot name for incremental backup. (Not Implemented yet) (default: None)

  - `--storage_type {local,nas}`

    Storage type of output directory. (default: local)

  - `--remote`
    Take backup through orion master. (default: True)

- `tscli backup delete` *name* Deletes the named backup.
- `tscli backup ls` List all backups taken by the system.
- `tscli backup restore` Restore cluster using backup.

backup-policy

```
tscli backup-policy [-h] {create,delete,disable,enable,ls,show,status,update}
```

Use this subcommand to do the following:

- `tscli backup-policy create` Prompts an editor for you to edit the parameters of the backup policy.
- `tscli backup-policy delete` *`name`* Deletes the backup policy with *`name`*.
- `tscli backup-policy disable` *`name`* Disables the policy *`name`*.
- `tscli backup-policy enable` *`name`* Enables the policy *`name`*.
- `tscli backup-policy ls` List backup policies.
- `tscli backup-policy show` *`name`* Show the policy *`name`*.
- `tscli backup-policy status` *`name`* Enables the policy *`name`*.
- `tscli backup-policy update` *name* Prompts an editor for you to edit the policy *`name`*.

callhome

```
tscli callhome [-h] {disable,enable,generate-bundle}
```

Use this subcommand to do the following:

- `tscli callhome disable` Turns off the periodic call home feature.
- `tscli callhome enable --customer_name` *`customer_name`*`

  Enables the "call home" feature, which sends usage statistics to ThoughtSpot Support every six hours via the secure file server. Before using this command for the first time, you need to set up the file server connection using `tscli fileserver configure`.

  The parameter *`customer_name`* takes the form `Shared/*`customer_name`*/stats`.
- `tscli callhome generate-bundle -d` *`directory`* `--since` *`DAYS`*
  - `--d D` Dest folder where tar file will be created. (default: None)
  - `--since` *`DAYS`*

    Grab callhome data from this time window in the past. Should be a human readable duration string, e.g. `4h` (4 hours), `30m` (30 minutes), `1d` (1 day). (default: None) Generates a tar file of the cluster metrics and writes it to the specified directory where *`DAYS`* is how far back you'd like to generate the tar file from in days. For example, `30`. If this parameter is not specified, the command will collect the stats from the last `7` days by default.

cluster

```
tscli cluster [-h]
{abort-reinstall-os,check,create,get-config,load,reinstall-os,report,restore,resume-rei
nstall-os,resume-update,set-config,set-min-resource-spec,show-resource-spec,start,statu
s,stop,update,update-hadoop}
```

Use this subcommand to do the following:

- `tscli cluster abort-reinstall-os` Abort in-progress reinstall.
- `tscli cluster check --includes {all,disk,zookeeper,hdfs,orion-cgroups,orion-oreo}` check the status nodes in the cluster.

  You must specify a component to check.
- `tscli cluster create` *`release`*

  Creates a new cluster from the release file specified by *release*. This command is used by ThoughtSpot Support when installing a new cluster, for example, `tscli cluster create 2.0.4.tar.gz`
- `tscli cluster get-config` Get current cluster network and time configuration. Prints JSON configuration to stdout. If for some reason the system cannot be connected to all interfaces, the command returns an error but continues to function.
- `tscli cluster load` Load state from given backup onto existing cluster
- `tscli cluster reinstall-os` Reinstall OS on all nodes of the cluster.
- `tscli cluster report` Generate cluster report.

- `tscli cluster restore --release release_tarball backupdir`

  Restores a cluster using the backup in the specified directory *backupdir*. If you're restoring from a dataless backup, you must supply the release tarball for the corresponding software release.

- `tscli cluster resume-reinstall-os` Resume in-progress reinstall.
- `tscli cluster resume-update` Resume in-progress updates.
- `tscli cluster set-config` Set cluster network and time configuration. Takes JSON configuration from stdin.
- `tscli cluster set-min-resource-spec` Sets min resource configuration of the cluster
- `tscli cluster show-resource-spec` Prints default or min.
- `tscli cluster start` Start cluster.
- `tscli cluster status` Gives the status of the cluster, including release number, date last updated, number of nodes, pending tables time, and services status.
- `tscli cluster stop` Pauses the cluster (but does not stop storage services).
- `tscli cluster update` Update existing cluster.
- `tscli cluster update-hadoop` Updates Hadoop/Zookeeper on the cluster.

command

```
tscli command [-h] {run}
```

Command to run a command on all nodes.

`tscli command run [-h] [--nodes NODES] --dest_dir DEST_DIR [--copyfirst COPYFIRST] [--timeout TIMEOUT] command`

- `--nodes NODES` Space separated IPs of nodes where you want to run the command. (default: `all`)
- `--dest_dir DEST_DIR` Directory to save the files containing output from each nodes. (default: None)
- `--copyfirst COPYFIRST` Copy the executable to required nodes first. (default: `False`)
- `--timeout TIMEOUT` Timeout waiting for the command to finish. (default: `60`)

dr-mirror

```
tscli dr-mirror [-h] {start,status,stop}
```

- `tscli dr-mirror start` Starts a mirror cluster which will continuously recover from a primary cluster.
- `tscli dr-mirror status` Checks whether the current cluster is running in mirror mode.
- `tscli dr-mirror stop` Stops mirroring on the local cluster.

entity

```
tscli entity [-h] {pack} ...
```

Creates a serialized, dataless object file for testing, troubleshooting, or migration. You can use this command with answer, pinboard, or aggregated/unaggregated worksheet objects.

- `tscli entity pack [-h] --id ID [--outdir FULLPATHNAME]` Packs object metadata and schema into a file. Packed filenames have the format XXX.YYY where XXX is ### ? optional arguments:

  - `--id ID` The required `ID` of the object to pack. IDs are found in thne URL of an answer, pinboard, or aggregated/unaggregated worksheet. For example, the ID for a pinboard `http://thoughtspot.com:8088/#/pinboard/`

061457a2-27bc-43a9-9754-0cd873691bf0/ is
061457a2-27bc-43a9-9754-0cd873691bf0.
- --outdir FULLPATHNAME Directory where the command places the packed object.

etl

```
tscli etl [-h] {change-password,disable-lw,download-agent,enable-lw,show-lw}
```

- tscli etl change-password --admin_username *admin_user* --username *Informatica_user*`

  Changes the Informatica Cloud account password used by ThoughtSpot Data Connect. Required parameters are:

  - --admin_username *admin_user* specifies the Administrator username for ThoughtSpot.
  - --username *Informatica_user* specifies the username for the Informatica Cloud.
- tscli etl disable-lw Disables ThoughtSpot Data Connect.
- tscli etl download-agent Downloads the ThoughtSpot Data Connect agent to the cluster.

- tscli etl enable-lw [-h] --username *USERNAME* --thoughtspot_url *THOUGHTSPOT_URL* --admin_username *ADMIN_USERNAME* [--groupname *GROUPNAME*] --org_id *ORG_ID* [--pin_to *PIN_TO*] [--proxy_host *PROXY_HOST*] [--proxy_port *PROXY_PORT*] [--proxy_username *PROXY_USERNAME*] [--max_wait *MAX_WAIT*]

  You should contact ThoughtSpot Support for assistance in setting this up. Required parameters are:

  - --username *USERNAME* Username for Informatica Cloud (default: None)
  - --thoughtspot_url *THOUGHTSPOT_URL* URL to reach thoughtspot. (default: None)
  - --admin_username *ADMIN_USERNAME* Admin username for ThoughtSpot (default: None)
  - --groupname *GROUPNAME*
  - --org_id *ORG_ID* specifies the Informatica id of the organization (company). For ThoughtSpot, this is 001ZFA. org_id shouldn't include the prefix Org. For example, if on Informatica cloud, the orgid is Org003XYZ, then use only
  - --pin_to *PIN_TO* specifies the IP address to pin to. If you specify an IP to pin to, that node becomes sticky to the Informatica agent, and will always be used. Defaults to the public IP address of the localhost where this command was run.
  - --proxy_host *PROXY_HOST* Proxy server host for network access (default:)
  - --proxy_port *PROXY_PORT* Proxy server port (default:)
  - --proxy_username *PROXY_USERNAME* Proxy server username (default:)
  - --max_wait *MAX_WAIT* Maximum time in seconds to wait for Data Connect agentto start (default: None)
- tscli etl show-lw Shows the status of ThoughtSpot Data Connect. It also returns the Informatica username and OrgId.

event

```
tscli event [-h] {list}
```

This subcommand has the following actions:

```
tscli event list [-h] [--include INCLUDE] [--since SINCE] [--from FROM] [--to TO] [--limit LIMIT] [--detail] [--summary_contains SUMMARY_CONTAINS] [--detail_contains DETAIL_CONTAINS] [--attributes ATTRIBUTES]
```

- --include *INCLUDE* Options are all, config, notification. Default config. (default: config)
- --since *SINCE* Grab events from this time window in the past. Should be a human readable duration string, e.g. 4h (4 hours), 30m (30 minutes), 1d (1 day). (default: None)

- `--from` *FROM* Begin timestamp, must be of the form: `yyyymmdd-HH:MM` (default: None)
- `--to` *TO* End timestamp, must be of the form: `yyyymmdd-HH:MM` (default: None)
- `--limit` *LIMIT* Max number of events to fetch. (default: 0)
- `--detail` Print events in detail format. This is not tabular. Default is a tabular summary. (default: False)
- `--summary_contains` *SUMMARY_CONTAINS* Summary of the event will be checked for this string. Multiple strings to check for can be specified by separating them with | (event returned if it matchesALL). Put single quotes around the param value to prevent undesired glob expansion (default: None)
- `--detail_contains` *DETAIL_CONTAINS* Details of the event will be checked for this string. Multiple strings to check for can be specified by separating them with | (event returned if it matches ALL). Put single quotes around the param value to prevent undesired glob expansion (default: None)
- `--attributes` *ATTRIBUTES* Specify attributes to match as key=value. Multiple attributes to check for can be specified by separating them with | (event returned if it matches ALL). Put single quotes around the param value to prevent undesired glob expansion (default: None)

feature

```
tscli feature [-h] {get-all-config}
```

This subcommand has the following actions:

`tscli feature get-all-config` Gets the configured features in a cluster. The command will return a list of features, such as custom branding, Data Connect, and call home, and tell you whether they are enabled or disabled.

fileserver

```
tscli fileserver [-h] {configure,download-release,purge-config,show-config,upload}
```

This subcommand has the following actions:

- `tscli fileserver configure [-h] --user` *USER* `[--password` *PASSWORD* `]` Configures the secure file server username and password for file upload/download and the call home feature. You only need to issue this command once, to set up the connection to the secure file server. You only need to reissue this command if the password changes. The parameter *PASSWORD* is optional. If a password is not specified, you will be prompted to enter it.
- `tscli fileserver download-release [-h] [--user` *USER*`] [--password` *PASSWORD*`]` *release* Downloads the specified release file and its checksum. Specify the release by number, to the second decimal point (e.g. 3.1.0, 3.0.5, etc.). You may optionally specify the `--user` and `--password` to bypass the credentials that were specified when configuring the file server connection with `tscli fileserver configure`. Before using this command for the first time, you need to set up the file server connection using `tscli fileserver configure`.
- `tscli fileserver purge-config` Removes the file server configuration.
- `tscli fileserver show-config` Shows the file server configuration.
- `tscli fileserver upload [-h] [--user` *USER*`] [--password` *PASSWORD*`] --file_name` *FILE_NAME* -server_dir_path *SERVER_DIR_PATH`

  Uploads the file specified to the directory specified on the secure file server. You may optionally specify the `--user` and `--password` to bypass the credentials that were specified when configuring the file server connection with `tscli fileserver configure`. Before using this command for the first time, you need to set up the file server connection using `tscli fileserver configure`.

  Accepts these flags

  - `--user` *USER* Username of fileserver (default: None)

- `--password` *`PASSWORD`* Password of fileserver (default: None). This is required and the command prompts you for it if you do not supply it.
- `--file_name` *`FILE_NAME`* Local file that needs to be uploaded (default: None)
- `--server_dir_path` *`SERVER_DIR_PATH`* Directory path on fileserver. (default: None) The *`SERVER_DIR_PATH`* parameter specifies the directory to which you want to upload the file. It is based on your customer name, and takes the form `/Shared/support/*`customer_name`*`.

firewall

```
tscli firewall [-h] {close-ports,disable,enable,open-ports,status}
```

- `tscli firewall close-ports`

  Closes given ports through firewall on all nodes. Takes a list of ports to close, comma separated. Only closes ports which were previously opened using “open-ports”. Ignores ports which were not previously opened with “open-ports” or were already closed.

- `tscli firewall disable` Disable firewall.
- `tscli firewall enable` Enable firewall.

- `tscli firewall open-ports *ports*`

  Opens given ports through firewall on all nodes. Takes a list of ports to open, comma separated. Ignores ports which are already open. Some essential ports are always kept open (e.g. `ssh`), they are not affected by this command or by `close-ports`.

- `tscli firewall status` Shows whether firewall is currently enabled or disabled.

hdfs

```
tscli hdfs [-h] {leave-safemode}
```

This subcommand has the following actions:

`tscli hdfs leave-safemode` Command to get HDFS namenodes out of safemode.

ldap

```
tscli ldap [-h] {add-cert,configure,purge-configuration}
```

This subcommand has the following actions:

- `tscli ldap add-cert` *`name certificate`*

  Adds an SSL certificate for LDAP. Use only if LDAP has been configured without SSL and you wish to add it. Use `*`name`*` to supply an alias for the certificate you are installing.

- `tscli ldap configure`

  Configures LDAP using an interactive script. You can see detailed instructions for setting up LDAP in About LDAP integration (page 0).

- `tscli ldap purge-configuration` Purges (removes) any existing LDAP configuration.

logs

```
tscli logs [-h] {collect,runcmd}
```

This subcommand has the following actions:

- `tscli logs collect [-h] [--include INCLUDE] [--exclude EXCLUDE] [--since SINCE] [--from FROM] [--to TO] [--out OUT] [--maxsize MAXSIZE] [--sizeonly] [--nodes NODES]`

  Extracts logs from the cluster. Does not include any logs that have been deleted due to log rotation.

  These parameters have the following values:

  - `--include INCLUDE`

    Specifies a comma separated list of logs to include. Each entry is either a "selector" or a glob for matching files.Selectors must be among: `all`, `orion`, `system`, `ts`. Anything starting with / is assumed to be a glob pattern and interpreted via `find(1)`. Other entries are ignored. Put single quotes around the param value to prevent undesired glob expansion (default: `all`)

  - `--exclude EXCLUDE`

    Comma separated list of logs to exclude. Applies to the list selected by –include. Params are interpreted just like in –include (default: None)

  - `--since SINCE`

    Grab logs from this time window in the past. Should be a human readable duration string, e.g. 4h (4 hours), 30m (30 minutes), 1d (1 day). (default: None)

  - `--from FROM` Timestamp where collection begins, must be of the form: `yyyymmdd-HH:MM` (default: None)
  - `--to TO` Timestamp where collection ends, must be of the form: `yyyymmdd-HH:MM` (default: None)
  - `--out OUT` Tarball path for dumping logs from each node (default: `/tmp/logs.tar.gz`)
  - `--maxsize MAXSIZE` Only fetch logs if size is smaller that this value. Can be specified in megabytes/gigabytes, e.g. 100MB, 10GB. (default: None)
  - `--sizeonly` Do not collect logs. Just report the size. (default: False)
  - `--nodes NODES` Comma separated list of nodes from where to collect logs. Skip this to use all nodes. (default: None)

- `tscli logs runcmd [-h] --cmd CMD [--include INCLUDE] [--exclude EXCLUDE] [--since SINCE] [--from FROM] [--to TO] [--outfile OUTFILE] [--outdir OUTDIR] [--cmd_infmt CMD_INFMT] [--cmd_outfmt CMD_OUTFMT] [--nodes NODES]`

  Runs a Unix command on logs in the cluster matching the given constraints. Results are reported as text dumped to standard out, the specified output file, or as tarballs dumped into the specified directory.

  - `--cmd CMD`

    Unix-Command to be run on the selected logs. Use single quotes to escape spaces etc. Language used to specify CMDSTR has following rules.

    - A logfile and its corresponding result file can be referred by keywords `SRCFILE` & `DSTFILE`. eg. `cp SRCFILE DSTFILE`

- Without any reference to DSTFILE in CMDSTR, `> DSTFILE` will be appended to CMDSTR for output redirection. eg `du -sch SRCFILE` gets auto-transtalted to `du -sch SRCFILE > DSTFILE`
- Without any reference to SRCFILE, content of log is streamed to CMDSTR via pipe. eg. `tail -n100 | grep ERROR` gets auto-transtalted to `cat SRCFILE | tail -n100 | grep ERROR > DSTFILE` (default: None)

- `--include` *`INCLUDE`*

  Comma separated list of logs to include,each entry is either a "selector" or a glob for matching files.Selectors must be among: `all`, `orion`, `system`, `ts`. Anything starting with / is assumed to be a glob pattern and interpreted via `find(1)`. Other entries are ignored. TIP: put single quotes around the param value to prevent undesired glob expansion (default: all)

- `--exclude` *`EXCLUDE`*

  Comma separated list of logs to exclude. Applies to the list selected by `--include`. Params are interpreted just like in `--include` (default: None)

- `--since` *`SINCE`*

  Grab logs from this time window in the past. Should be a human readable duration string, e.g. `4h` (4 hours), `30m` (30 minutes), `1d` (1 day). (default: None)

- `--from` *`FROM`* Timestamp where collection begins, must be of the form: `yyyymmdd-HH:MM` (default: None)
- `--to` *`TO`* Timestamp where collection ends, must be of the form: `yyyymmdd-HH:MM` (default: None)
- `--outfile` *`OUTFILE`* File path for printing all the results. By default printed to stdout (default: None)
- `--outdir` *`OUTDIR`* Directory path for dumping results with original dir structure from each node. Used as an alternative to printing output to outfile/stdout (default: None)
- `--cmd_infmt` *`CMD_INFMT`* Specify if the inputfile should be compressed/uncompressed before running `CMD`. `C`=compressed, `U`=uncompressed. Don't use this flag if `CMD` works on both (default: None)
- `--cmd_outfmt` *`CMD_OUTFMT`* Specify if *`OUTFILE`* generated by `CMD` will be compressed/uncompressed. `C`=compressed, `U`=uncompressed. Don't use this flag if output file will be of same format as input file (default: None)
- `--nodes` *`NODES`* Comma separated list of nodes where to run command. Skip this to use all nodes. (default: None)

map-tiles

```
tscli map-tiles [-h] {disable,enable}
```

This subcommand supports the following actions:

- `tscli map-tiles enable [-h] [--online] [--offline] [--tar TAR] [--md5 MD5]`

  Enables ThoughtSpot's map tiles, which are used when constructing geomap charts. If you don't have interest access, you must download the map tiles tar and md5 files. Then you must append the following to the `tscli` command.

  - `--online` Download `maptiles` tar from internet. (default: True)
  - `--offline` Using `maptiles` tar from local disk. (default: False)
  - `--tar` *`TAR`* Specified tar file for map-tiles. (default:)

    - `--md5 MD5` Specified md5 file for map-tiles. (default:)

- `tscli map-tiles disable` Disable map-tiles functionality.

monitoring

```
tscli monitoring [-h] {set-config,show-config}
```

This subcommand has the following actions:

- `tscli monitoring set-config [-h] [--email EMAIL] [--clear_email] [--heartbeat_interval HEARTBEAT_INTERVAL] [--heartbeat_disable] [--report_interval REPORT_INTERVAL] [--report_disable]` Sets the monitoring configuration.

    - `--email EMAIL` Comma separated list (no spaces) of email addresses where the cluster will send monitoring information.
    - `--clear_email` Disable emails by clearing email configuration. (default: False)
    - `--heartbeat_interval HEARTBEAT_INTERVAL` Heartbeat email generation interval in seconds. Should be greater than 0.
    - `--heartbeat_disable` Disable heartbeat email generation. (default: False)
    - `--report_interval REPORT_INTERVAL` Cluster report email generation interval in seconds. Should be greater than 0.
    - `--report_disable` Disable cluster report email generation. (default: False)

- `tscli monitoring show-config` Shows the monitoring configuration.

nas

```
tscli nas [-h] {ls,mount-cifs,mount-nfs,unmount}
```

This subcommand has the following actions:

- `tscli nas ls [-h]` List mounts managed by NAS mounter service.

- `tscli nas mount-cifs [-h] --server SERVER [--path_on_server PATH_ON_SERVER ] --mount_point MOUNT_POINT --username USERNAME --password PASSWORD [--uid UID] [--gid GID] [--options OPTIONS ]`

  Mounts a CIFS device on all nodes.

    - `--server SERVER` IP address or DNS name of CIFS service. For example, `10.20.30.40` (default: None)

    - `--path_on_server PATH_ON_SERVER` Filesystem path on the CIFS server to mount (source). For example: `/a` (default: `/`)

    - `--mount_point MOUNT_POINT`

      Directory on all cluster nodes where the NFS filesystem should be mounted (target). This directory does not need to already exist. If this directory already exists, a new directory is not created and the existing directory is used for mounting. For example: `/mnt/external` (default: None)

    - `--username USERNAME` Username to connect to the CIFS filesystem as (default: None)
    - `--password PASSWORD` CIFS password for `--username` (default: None)

    - `--uid UID`

      UID that will own all files or directories on the mounted filesystem when the server does not provide ownership information. See `man mount.cifs` for more details. (default: `1001`)

- `--gid` *GID*

  Gid that will own all files or directories on the mounted filesystem when the server does not provide ownership information. See `man mount.cifs` for more details. (default: `1001`)

- `--options` *OPTIONS* Other command-line options to forward to `mount.cifs` command (default: `noexec`)

- `tscli nas mount-nfs [-h] --server SERVER [--protocol PROTO --path_on_server PATH_ON_SERVER] --mount_point MOUNT_POINT [--options OPTIONS]`

  Mounts a NFS device on all nodes. Parameters are:

  - `--server SERVER` IP address or DNS name of NFS service. For example, `10.20.30.40` (default: None)
  - `--path_on_server PATH_ON_SERVER` Filesystem path on the NFS server to mount (source). For example: `/a/b/c/d` (default: `/`)
  - `--mount_point MOUNT_POINT`

    Directory on all cluster nodes where the NFS filesystem should be mounted (target). This directory does not need to already exist. If this directory already exists, a new directory is not created and the existing directory is used for mounting. For example: `/mnt/external` (default: None)

  - `--options OPTIONS` Command-line options to forward to mount command (default: `noexec`).
  - `--protocol PROTO` One of `nfs` or `nfs4`. The default is `nfs`.

- `tscli nas unmount [-h] --dir` *DIR*

  Unmounts all devices from the specified *DIR* (directory) location. This command returns an error if nothing is currently mounted on this directory via `tscli nas mount` (default: None)

node

```
tscli node [-h] {check,ls,reinstall-os,status}
```

This subcommand has the following actions:

- `tscli node check [-h] [--select {reinstall-preflight}] [--secondary` *SECONDARY*`]`

  Run checks per node. Takes the following parameters:

  - `--select {reinstall-preflight}` Select the type of node check (default: `reinstall-preflight`)
  - `--secondary` *SECONDARY* Secondary drive for `reinstall-preflight` (default: `sdd`)

- `tscli node ls [-h] [--type {all,healthy,not-healthy}]` Filter by node state (default: `all`)

- `tscli node reinstall-os [-h] [--secondary` *SECONDARY* `] [--cluster]` Reinstall OS on a node. This takes the following parameters:

  - `--secondary` *SECONDARY* Secondary drive to be used to carry to reinstall (default: `sdd`)
  - `--cluster` Is the node part of a cluster (default: `False`)

saml

```
tscli saml [-h] {configure,purge-configuration}
```

This subcommand has the following actions:

- `tscli saml configure [-h]` Configures SAML. To see a list of prerequisites refer to Configure SAML (page 0).
- `tscli saml purge-configuration` Purges any existing SAML configuration.

scheduled-pinboards

```
tscli scheduled-pinboards [-h] {disable,enable}
```

This subcommand has the following actions:

- `tscli scheduled-pinboards disable [-h]` Disable scheduled pinboards for this cluster.
- `tscli scheduled-pinboards enable [-h]` Enables scheduled pinboards, which is disabled in prod clusters by default.

smtp

```
tscli smtp [-h]
{reset-canonical-mapping,set-canonical-mapping,set-mailfromname,set-mailname,set-relayh
ost,show-canonical-mapping,show-mailfromname,show-mailname,show-relayhost}
```

This subcommand takes supports the following actions:

- `tscli smtp reset-canonical-mapping` Deletes the current postmap mapping.
- `tscli smtp set-canonical-mapping [-h]` *`new_key new_value`* Sets a new Postmap mapping.
- `tscli smtp set-mailfromname` *`mailfromname`* Sets the name, an email address, from which email alerts are sent, for the cluster.
- `tscli smtp set-mailname` *`mailname`* Sets the mailname, a domain, where email alerts are sent, for the cluster.

- `tscli smtp set-relayhost [-h] [--force FORCE] relayhost` Sets the Relay Host for SMTP (email) sent from the cluster.

    - `--force` *`FORCE`* Set even if relay host is not accessible. (default: `False`)
- `tscli smtp show-canonical-mapping` Shows the current postmap mapping.
- `tscli smtp show-mailfromname` Shows the mailname, from which email alerts are sent, for the cluster.
- `tscli smtp show-mailname` Shows the mailname, where email alerts are sent, for the cluster.
- `tscli smtp show-relayhost` Shows the for SMTP (email) sent from the cluster. If there is no Relay Host configured, the command returns `NOT FOUND`.

snapshot

```
tscli snapshot [-h] {backup,create,delete,ls,restore,update-ttl}
```

Learn more about snapshots and backups see the Understand the backup strategies (page 0) documentation. This subcommand supports the following actions:

- `tscli snapshot backup [-h] [--mode {full,light,dataless}] [--type {full,incremental}] [--base` *`BASE`* `] [--storage_type {local,nas}] [--remote]` *`name out`*

  Pull snapshot out as a backup. This takes the following parameters:

    - `--mode {full,light,dataless}` Mode of backups. (default: `full`)

    - *name* Name of snapshot to pull out as a backup. To list all snapshots, run `tscli snapshot ls`.
    - *out* Directory where backup will be written, must not already exist.
    - `--type {full,incremental}` Type of backup.(Incremental backup is not implemented yet) (default: `full`)
    - `--base` *BASE* Based snapshot name for incremental backup. (Not Implemented yet) (default: None)
    - `--storage_type {local,nas}` Storage type of output directory. (default: `local`)
    - `--remote` Take backup through Orion master. (default: `True`)

- `tscli snapshot create [-h]` *name reason ttl*

  Creates a new snapshot with the *name* and *reason* provided. This command does not accept `.` (periods), but does accept – (dashes). The `ttl` parameters is the number of days after which this snapshot will be automatically deleted. A value of `-1` disables automatic deletion.

- `tscli snapshot delete [-h]` *name* Deletes the named snapshot.
- `tscli snapshot ls [-h]` List available snapshots.

- `tscli snapshot restore [-h] [--allow_release_change] [--only_service_state]` *name* Restore cluster to an existing snapshot. This takes the following parameters:

    - `--allow_release_change` Allow restoration to a snapshot at a different release. (default: `False`)
    - `--only_service_state` Restore only service state. (default: `False`)

- `tscli snapshot update-ttl [-h] [--disable` *DISABLE* `]` *name ttl*

  Updates manual snapshot garbage collection policy.

    - `name` Specifies which snapshot to update.
    - `ttl` Extends the manual snapshot `ttl` (time-to-live) value. Use a positive value to increase `ttl`. Use negative value to decrease it.
    - `--disable` *DISABLE* Disable manual snapshot garbage collection. Setting this value to `True` will override any `ttl` value. (default: False)

snapshot-policy

```
tscli snapshot-policy [-h] {disable,enable,show,update}
```

This subcommand supports the following actions:

- `tscli snapshot-policy disable [-h]` Disable snapshot policy.
- `tscli snapshot-policy enable -h` Enable specified snapshot policy.
- `tscli snapshot-policy show [-h]` Show snapshot policy.

- `tscli snapshot-policy update [-h] [--config` *CONFIG*`]` Update periodic snapshot config. This takes the following parameter:

    - `--config` *CONFIG* Text format of periodic backup policy config (default: None)

spot

```
tscli spot [-h] {enable}
```

This subcommand supports the following actions:

`tscli spot [-h] {enable}` Enableds Spot integrtion.

ssl

```
tscli ssl [-h]
{add-cert,clear-min-tls-version,off,on,rm-cert,set-min-tls-version,status,tls-status}
```

status Shows whether of SSL authentication is enabled or disabled for the ThoughtSpot application. tls-status Prints the status of TLS support

This subcommand supports the following actions:

- `tscli ssl add-cert [-h]` *`key certificate`* Adds an SSL certificate, key pair.
- `tscli ssl clear-min-tls-version [-h]` Clears any customizations for the minimum TLS version to support.

- `tscli ssl off`

  Disables SSL. Disabling SSL will stop users from seeing a security warning when accessing ThoughtSpot from a browser if there is no SSL certificate installed.

- `tscli ssl on [-h]` If SSL is enabled and there is no certificate, users will see a security warning when accessing ThoughtSpot from a browser.
- `tscli ssl rm-cert` Removes the existing SSL certificate, if any.
- `tscli ssl set-min-tls-version [-h] {1.0,1.1,1.2}` Sets the minimum supported TLS version. Sets the minimum SSL version to be supported by the ThoughtSpot application. Please ensure that client browsers are enabled for this version or newer.
- `tscli ssl status` Shows whether SSL authentication is enabled or disabled.
- `tscli ssl tls-status [-h]` Prints the status of TLS support.

storage

```
tscli storage [-h] gc df
```

This subcommand supports the following actions:

- `tscli storage gc [-h] [--log_age` *`LOG_AGE`* `] [--force] [--localhost_only]`

  Garbage collect unused storage. Before issuing this command, you must stop the cluster using `tscli cluster stop`. After garbage collection has completed, you can restart the cluster with `tscli cluster start`. The command frees space in these directories:

  - `/tmp`
  - `/usr/local/scaligent/logs/`
  - `/export/logs/orion`
  - `/export/logs/oreo`
  - `/export/logs/hadoop`
  - `/export/logs/zookeeper`
  - `cores`

  Accepts these optional flags:

  - `--log_age` *`LOG_AGE`*

    Delete logs older than these many hours. Use a non-zero value ideally. A zero value will cause all temporary files to be deleted, including say those which are just temporarily closed while they are being passed from one component to the next. (default: `4`)

  - `--force` Forces deletion of all logs and temporary files regardless of age. This must

only be run on a stopped cluster. (default: False)

- `--localhost_only` If used, only the logs on the localhost will be removed. If not specified, the command acts on the entire cluster.

- `tscli storage df [--mode disk|hdfs]`

  Checks the disk usage on the relevant mounts. Returns output similar to the Linux system command `df -h <directory>`.

support

```
tscli support [-h]
{restart-remote,rm-admin-email,rm-admin-phone,set-admin-email,set-admin-phone,set-remot
e,show-admin-email,show-admin-phone,show-remote,start-remote,stop-remote}
```

This subcommand supports the following actions:

- `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-admin-phone` Removes the phone number for contacting the customer administrator. Replaces it with the default ThoughtSpot Support phone number.
- `tscli support set-admin-email` *`email`* Sets the email address for contacting the customer administrator. If you would like to display a blank email address, issue the command `tscli support set-admin-email ' '`.
- `tscli support set-admin-phone` *`phone_number`* Sets the phone number for contacting the customer administrator. Specify a phone number using any value (e.g. `+1 800-508-7008 Ext. 1`). If you would like to display a blank phone number, issue the command `tscli support set-admin-phone`.
- `tscli support set-remote [-h] [--addr` *`ADDR`*`] [--user` *`USER`*`]` Configures the cluster for remote support through SSH tunneling, where *`ADDR`* is the address of support, e.g. `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-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.

Date and time formats reference

This is a references for the date and time contexts and formats you can use with ThoughtSpot. You define data formats in specific contexts and, depending on the context, your choices in data formatting differ. You must understand date and time when you load data in these contexts:

- using data upload from the browser
- through `tsload` command
- through data connect or another extract, transform, load (ETL) tool

Data loading formats do not change how data is displayed in tables and charts.

The context where you *can control* date and time formats is data modeling. Data modeling controls how data is displayed in search and their resulting answers.

Data loading formats via tsload

When loading via the `tsload` command you must specify `date` and `timestamp` formats using the format specifications defined in the `strptime` library function. Data is imported based on the timezone of the node from which `tsload` is run.

For `date` data types, the default format is `%Y%m%d` which translates to `yearmonthday` For example, `Dec 30th, 2001` is represented as `20011230`. For `time` and `datetime` data types, the default is `%Y%m%d %H:%M:%S` which translates to `yearmonthday hour:minute:second`, for example, `Dec 30th, 2001 1:15:12` is represented as `20011230 01:15:12`.

Data modeling formats for browser data upload

These date and time formats are supported in a CSV file when uploading via the browser. You cannot specify the date format; ThoughtSpot will pick the format that fits your data best:

- 1/30/2014
- 2014-01-30
- 2014-1-30
- 30-Jan-2014
- 2014-Jan-30
- 2014-01-30 10:32 AM
- 2014-01-30 14:52
- 2014-01-30 10:32:22
- 2014-01-30 10:32:22 AM
- 2014-01-30 10:32:22.0
- 2014-01-30 10:32:22.0 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 via data connect or another ETL tool

Data that is loaded via ETL arrives through ODBC or JDBC connection. After you extract the data from the source and before you load it into ThoughtSpot, you must transform any date or timestamp into a valid format for ThoughtSpot. Once transformed, no explicit data masking is required. See the data integration guide for more details of loading data via ODBC and JDBC.

Data modeling formats

A user with administrative rights can configure data modeling for data on one or all files. You can set number, date, and currency display formats. These formats define how these value types display in tables and charts. See the Admin Guide for more information about data modeling settings. The following format strings are available for use:

Format mask	Description
`YYYY` or `yyyy`	four digit year such as `2017`
`YY` or `yy`	last two digits of year such as `17`
`M`	month with no leading zero `1-12`
`MM`	Two digit month `01-12`
`MMM`	Three letter month such as `Jan`
`D`	Day of year without a leading zero `0-365`
`DD`	Day of year with up to one leading zero `01-365`
`DDD`	Day of year with up to two leading zeroes `001-365`

Format mask	Description
`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 (page 0). 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, f 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
`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 strptime date formatting elements. Replaces all the valid strptime 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_string (45 + 1) = '46'` `to_string (revenue - cost)`

Date functions

Function	Description	Examples
`add_days`	Returns the result of adding the specified number of	`add_days (01/30/2015, 5) = 02/04/2015` `add_days (invoiced, 30)`

Function	Description	Examples
	days to the given date.	
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)
day_number_of_week	Returns the number (1-7) of the day in a week for the given date with 1 being Monday and 7 being Sunday.	day_number_of_week (01/30/2015) = 6 day_number_of_week (shipped)
day_number_of_year	Returns the number (1-366) of the day in a year for the given date.	day_number_of_year (01/30/2015) = 30 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)
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 for the given date.	month_number (09/20/2014) = 9 month_number (purchased)
now	Returns the current timestamp.	now ()
start_of_month	Returns `MMM yyyy` for	start_of_month (01/31/2015) = Jan

Function	Description	Examples
	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 if you .	`FY 2015` `start_of_month (shipped)`
`start_of_quarter`	Returns the date for the first day of the quarter for the given date.	`start_of_quarter ( 09/18/2015 ) = Q3 FY 2015` `start_of_quarter (sold)`
`start_of_week`	Returns the date for the first day of the week for the given date.	`start_of_week ( 06/01/2015 ) = 05/30/2015 Week` `start_of_week (emailed)`
`start_of_year`	Returns the date for the first day of the year for the given date.	`start_of_year ( 02/15/2015 ) = FY 2015` `start_of_year (joined)`
`time`	Returns the time portion of a given date.	`time (3/1/2002 10:32) = 10:32` `time (call began)`
`year`	Returns the year from the given date.	`year (01/15/2014) = 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	`3 >= 2 = true`

Function	Description	Examples
	than or equal to the second value.	`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)`

Function	Description	Examples
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)
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)
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)

Function	Description	Examples
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)
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'
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 the one or more values as a concatenated text string. Be sure to use single quotes instead of double quotes around each of the strings.	`concat ( 'hay' , 'stack' ) = 'haystack'` `concat (last_name , first_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)`
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	`similarity ('where is the burning man concert', 'burning man') = 46` `similarity (tweet1, tweet2)`

Function	Description	Examples
	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.	
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 the list 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`

Formula reference

ThoughtSpot allows you to create derived columns in worksheets using formulas. This reference lists the various operators and functions you can use to create formulas.

You can also see this list of operators and examples from within the Formula Builder by selecting Formula Assistant.

Aggregate functions

These functions can be used to aggregate data.

Function	Description	Examples
average	Returns the average of all the values of a column.	`average (revenue)`
count	Returns the number of rows in the table containing the column.	`count (product)`
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)`
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)`

Function	Description	Examples
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)`
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)`
min	Returns the minimum value of a column.	`min (revenue)`
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)`
moving_min	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.	`moving_min (defects, 3, 1, product)`

Function	Description	Examples
moving_sum	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.	`moving_sum (revenue, 1, 1, order date)`
stddev	Returns the standard deviation of all values of a column.	`stddev (revenue)`
sum	Returns the sum of all the values of a column.	`sum (revenue)`
unique count	Returns the number of unique values of a column.	`unique count (customer)`
variance	Returns the variance of all the values of a column.	`variance (revenue)`

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 strptime date formatting elements. Replaces all the valid strptime 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_string (45 + 1) = '46'` `to_string (revenue - cost)`

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)
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)
day_number_of_week	Returns the number (1-7) of the day in a week for the given date with 1 being Monday and 7 being Sunday.	day_number_of_week (01/30/2015) = 6 day_number_of_week (shipped)
day_number_of_year	Returns the number (1-366) of the day in a year for the given date.	day_number_of_year (01/30/2015) = 30 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)
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)

Function	Description	Examples
month_number	Returns the number (1-12) of the month for the given date.	month_number (09/20/2014) = 9 month_number (purchased)
now	Returns the current time-stamp.	now ()
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 if you .	start_of_month (01/31/2015) = Jan FY 2015 start_of_month (shipped)
start_of_quarter	Returns the date for the first day of the quarter for the given date.	start_of_quarter (09/18/2015) = Q3 FY 2015 start_of_quarter (sold)
start_of_week	Returns the date for the first day of the week for the given date.	start_of_week (06/01/2015) = 05/30/2015 Week start_of_week (emailed)
start_of_year	Returns the date for the first day of the year for the given date.	start_of_year (02/15/2015) = FY 2015 start_of_year (joined)
time	Returns the time portion of a given date.	time (3/1/2002 10:32) = 10:32 time (call began)
year	Returns the year from the given date.	year (01/15/2014) = 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

Function	Description	Examples
=	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)`

Function	Description	Examples
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)
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)
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	sign (-250) = -1

Function	Description	Examples
	than zero, -1 if less than zero, 0 if zero.	`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)`
`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'`
`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 the one or more values as a concatenated text string. Be sure to use single quotes instead of double quotes around each of the strings.	`concat ( 'hay' , 'stack' ) = 'haystack'` `concat (last_name , first_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)`
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	`similarity ('where is the burning man concert', 'burning man') = 46` `similarity (tweet1, tweet2)`

Function	Description	Examples
	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.	
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)`

Error code reference

Summary: List of error codes and messages.

This section lists error codes that can appear in ThoughtSpot, with summary information and actions to take. Error codes and messages are shown in ThoughtSpot when something goes wrong. These messages can appear in the application and in logs.

When you see an error code, you will also see a message with a brief summary of what has happened. If there is a remediation action you can take, it will be listed in this references. If there is no action listed, please contact ThoughtSpot Support.

☑ **Tip:** Only the base code number is listed for each error. So keep this in mind when searching through these codes. For example, error code `TS-00125` is simply listed as `125`.

Metadata Errors (100 - 499)|

Code	Severity	Summary	Details	Action
TS-100	INFO	Success. {1} has been added to {2}. # {1} – name of visualization # {2} – {name/link to pinboard}	None	None
TS-101	ERROR	Failure adding {1} to {2}	Visualization could not be added to {2} # {1} – name of visualization # {2} – name/link to pinboard	None
TS-102	ERROR	Failure adding {1} to {2} due to corruption	{1} could not be added to {2} as the pinboard has one or more invalid visualizations	Please try again after removing the invalid visualization(s) from {2} # {1} – name of visualization # {2} – name/link to pinboard
TS-103	INFO	Success. Visualization has been deleted from {1}. 1 – name/link to pinboard	None	None
TS-104	ERROR	Failure deleting visual from {1}	Visualization could not be deleted from the pinboard. 1 – name/link to pinboard	None
TS-105	ERROR	Failure deleting visual from {1} due to corruption	Visualization could not be deleted from {1} as the pinboard has one or more	Please try again after removing the invalid visualization(s) from the pin-

Code	Severity	Summary	Details	Action
			invalid visualizations. 1 – name/link to pinboard	board
TS-106	INFO	Success. {1} created successfully. 1 – name/link to pinboard	None	None
TS-107	ERROR	Failure creating {1}. 1 – name/link to pinboard	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-108	INFO	Sticker created successfully.	None	None
TS-109	ERROR	Failure creating the sticker.	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-110	INFO	Sticker deleted successfully.	None	None
TS-111	ERROR	Failure deleting sticker.	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-112	INFO	Pinboards deleted successfully.	None	None
TS-113	ERROR	Failure deleting pinboards	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-114	INFO	Answers deleted successfully.	None	None
TS-115	ERROR	Failure deleting answers	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-116	INFO	Tables deleted successfully.	None	None
TS-117	ERROR	Failure deleting tables	Uh oh. We're not sure what happened. Please email the trace file to {ad-	None

Code	Severity	Summary	Details	Action
			minEmail}.	
TS-118	INFO	Relationship created successfully.	None	None
TS-119	ERROR	Failure creating relationship	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-120	INFO	Relationship updated successfully.	None	None
TS-121	ERROR	Failure updating the relationship	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-122	INFO	Relationship deleted successfully.	None	None
TS-123	ERROR	Failure deleting the relationship	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-124	ERROR	Failure fetching details for table	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-125	ERROR	Failure fetching details for the tables	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-126	ERROR	Failure fetching details for datasource	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-127	ERROR	Failure fetching details for datasources	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-128	ERROR	Failure fetching details for metadata items	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None

Code	Severity	Summary	Details	Action
TS-129	ERROR	Failure opening the answer	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-130	ERROR	Failure opening the pinboard	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-131	ERROR	Failure opening the worksheet	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-132	INFO	Table saved successfully.	None	None
TS-133	ERROR	There was a problem saving the table	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-134	INFO	Visualization update successful	None	None
TS-135	ERROR	Visualization failed to update	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-136	INFO	{1} saved 1 – name of answer	None	None
TS-137	ERROR	{1} could not be saved 1 – name of answer	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-138	INFO	{1} saved 1 - name of pinboard / link	None	None
TS-139	ERROR	{1} could not be saved 1 - name of pinboard / link	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-140	INFO	{1} saved 1 – name of worksheet	None	None
TS-141	ERROR	{1} could not be	Uh oh. We're not sure	None

Code	Severity	Summary	Details	Action
		saved 1 – name of worksheet	what happened. Please email the trace file to {adminEmail}.	
TS-142	INFO	{1} saved 1 – name of answer	None	None
TS-143	ERROR	{1} could not be saved	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. 1 – name of answer	None
TS-144	INFO	{1} saved 1 – name/ link to pinboard	None	None
TS-145	ERROR	{1} could not be saved	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}. 1 – name of pinboard	None
TS-146	INFO	Worksheet saved	None	None
TS-147	ERROR	Worksheet could not be saved	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-148	INFO	Sticker updated	None	None
TS-149	ERROR	The sticker could not be updated	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-150	INFO	Successfully assigned sticker	None	None
TS-151	ERROR	The sticker could not be assigned	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-152	INFO	Successfully unassigned sticker	None	None
TS-153	ERROR	The sticker could not be unassigned	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None

Code	Severity	Summary	Details	Action
TS-154	ERROR	Failed to fetch metadata list	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-155	ERROR	Failed to fetch table list	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-156	ERROR	Failed to fetch relationship list	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-157	ERROR	Failed to fetch answer list	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-158	ERROR	Failed to fetch pinboard list	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-159	ERROR	Failed to fetch worksheet list	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-160	ERROR	Failed to fetch aggregated worksheet list	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-161	ERROR	Failed to fetch imported data list	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-162	ERROR	Failed to fetch system table list	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-163	ERROR	Failed to DB view list	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-164	ERROR	Failed to fetch data source list	Uh oh. We're not sure what happened. Please	None

Code	Severity	Summary	Details	Action
			email the trace file to {adminEmail}.	
TS-165	ERROR	Failed to fetch column list	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-166	ERROR	Failed to label list	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-167	ERROR	Failed to fetch answer	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-168	ERROR	Failed to fetch worksheet	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-169	INFO	Aggregated worksheet {1} created 1 – name of aggregated worksheet	None	None
TS-170	ERROR	Failure creating Aggregated Worksheet.	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-171	INFO	{1} updated 1 – name of aggregated worksheet	None	None
TS-172	ERROR	{1} failed to update 1 – name of aggregated worksheet	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-173	ERROR	{1} failed to update 1 – name of the formula	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-174	ERROR	Comments cannot be fetched	Failed to save client state	None
TS-175	ERROR	Comment cannot	Uh oh. We're not sure	None

Code	Severity	Summary	Details	Action
		be created	what happened. Please email the trace file to {adminEmail}.	
TS-176	ERROR	Comment cannot be updated	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-177	ERROR	Comment cannot be deleted	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-178	INFO	Rule saved successfully	None	None
TS-179	ERROR	Rule could not be saved	We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-180	INFO	Rule deleted successfully	None	None
TS-181	ERROR	Rule could not be deleted	We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-182	INFO	Item deleted successfully.	None	None
TS-183	ERROR	Item could not be deleted.	We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-184	INFO	Related link created successfully.	None	None
TS-185	ERROR	Related link could not be created.	Uh oh. We're not sure what happened. Please click 'Report Problem' to email a report to your administrator, {adminEmail}.	None
TS-186	INFO	Related link updated successfully.	None	None
TS-187	ERROR	Related link could not be updated.	Uh oh. We're not sure what happened. Please click 'Report Problem' to email a report to your ad-	None

Code	Severity	Summary	Details	Action
			ministrator, {adminEmail}.	
TS-188	INFO	Related link deleted successfully.	None	None
TS-189	ERROR	Related link could not be deleted.	Uh oh. We're not sure what happened. Please click 'Report Problem' to email a report to your administrator, {adminEmail}.	None
TS-190	INFO	Related link detail fetched successfully.	None	None
TS-191	ERROR	Related link detail could not be fetched.	Uh oh. We're not sure what happened. Please click 'Report Problem' to email a report to your administrator, {adminEmail}.	None

Data Service Errors (500 - 699)

Code	Severity	Summary	Details	Action
TS-500	ERROR	Failed to fetch leaf level data	Failed to fetch leaf level data.	None
TS-501	ERROR	Failed to fetch excel data	Failed to fetch excel data.	None
TS-502	ERROR	Failed to fetch visualization data	Failed to fetch visuzliation data.	None
TS-503	ERROR	Failed to fetch visualizations data	Failed to fetch data for visualizations.	None
TS-504	ERROR	Failed to fetch chart data	Failed to fetch table data.	None
TS-505	ERROR	Failed to fetch table data	Failed to fetch table data.	None
TS-506	ERROR	Failed to fetch worksheet data	Failed to fetch worksheet data.	None
TS-507	ERROR	Failed to fetch filter data	Failed to fetch filter data.	None

Code	Severity	Summary	Details	Action
TS-508	ERROR	Failed to fetch headline data	Failed to fetch filter data.	None
TS-509	ERROR	Failed to fetch natural query	Failed to fetch natural query.	None
TS-510	INFO	File upload successful	None	None
TS-511	ERROR	Failed to upload file	Failed to upload	None
TS-512	ERROR	The pinboard data could not be exported to pdf.	Uh oh. We're not sure what happened. Please click 'Report Problem' to email a report to your administrator, {adminEmail}.	None

Dependency Errors (700 - 799)

Code	Severity	Summary	Details	Action
TS-700	ERROR	Failure fetching table dependents	Failed to fetch dependents for the table.	None
TS-701	ERROR	Failure fetching column dependents	Failed to fetch dependents for the column.	None
TS-702	ERROR	Failure fetching incomplete items	Failed to fetch incomplete items.	None

Admin Service Errors (800 - 899)

Code	Severity	Summary	Details	Action
TS-800	ERROR	Failure fetching MemCache stats	Failed to fetch MemCache stats.	None
TS-801	ERROR	Failure MemCache Clear	Failed to clear MemCache.	None
TS-802	ERROR	Failure searching from MemCache	Failed to search from MemCache.	None
TS-803	ERROR	Failure fetching Loggers	Failed to fetch Loggers.	None
TS-804	ERROR	Failure setting LogLevel	Failed to set Log Level.	None
TS-805	ERROR	Failure getting debug info	Failed to get debug info.	None

Code	Severity	Summary	Details	Action
TS-806	INFO	Memcache cleared successfully	None	None
TS-807	INFO	Log level set successfully	None	None
TS-808	ERROR	Failed to report problem	None	None
TS-809	INFO	Problem reported successfully	None	None

Permissions Errors (900 - 999)

Code	Severity	Summary	Details	Action
TS-900	ERROR	Failure fetching table permissions	Failed to fetch table permissions.	None
TS-901	ERROR	Failure fetching answer permissions	Failed to fetch answer permissions.	None
TS-902	ERROR	Failure fetching pinboard permissions	Failed to fetch pinboard permissions.	None
TS-903	ERROR	Failure getting metadata permissions	Failed to get metadata permissions.	None

Import Data Errors (1000 - 1099)

Code	Severity	Summary	Details	Action
TS-1000	ERROR	Data caching failed	Data caching failed.	None
TS-1001	ERROR	Read Columns failed.	Failed to read columns.	None
TS-1002	ERROR	Failed to read keys.	Failed to read keys.	None
TS-1003	ERROR	Failed to read relationships.	Failed to read relationships.	None
TS-1004	ERROR	Failed to load data.	Failed to load data.	None
TS-1005	ERROR	Failed to create table.	Failed to create table.	None
TS-1006	ERROR	Failed to fetch data rows.	Failed to fetch data rows.	None
TS-1007	ERROR	Failed to delete files.	Failed to fetch data rows.	None
TS-1008	ERROR	Failed to abort create table.	Failed to abort create table.	None

Code	Severity	Summary	Details	Action
TS-1009	ERROR	Failed to create schema.	Failed to create schema.	None
TS-1010	ERROR	Failed to fetch table models.	Failed to fetch table models.	None
TS-1011	ERROR	Failed to fetch sample values.	Failed to fetch sample values.	None

Scheduled Jobs Errors (1100 - 1199)

Code	Severity	Summary	Details	Action
TS-1100	INFO	The list of jobs.	None	Please click 'Report Problem' to email a report to your administrator.
TS-1110	INFO	Successfully created job.	None	None
TS-1111	ERROR	The job could not be created.	None	Please click 'Report Problem' to email a report to your administrator.
TS-1112	INFO	Successfully updated job.	None	None
TS-1113	ERROR	The job could not be updated.	None	Please click 'Report Problem' to email a report to your administrator.
TS-1114	INFO	Successfully deleted jobs.	None	None
TS-1115	ERROR	The job could not be deleted.	None	Please click 'Report Problem' to email a report to your administrator.
TS-1116	INFO	The job was paused.	None	None
TS-1117	ERROR	The job could not be paused.	None	Please click 'Report Problem' to email a report to your administrator.
TS-1118	INFO	The job was resumed	None	None
TS-1119	ERROR	The job could not be resumed.	None	Please click 'Report Problem' to email a report to your administrator.

User Admin Service Errors (1200 - 1399)

Code	Severity	Summary	Details	Action
TS-1200	ERROR	Failed to fetch users list	Failed to fetch users list	None
TS-1201	ERROR	Failed to fetch groups list	Failed to fetch groups list	None
TS-1202	ERROR	Failed to fetch users and groups list	Failed to fetch users and groups list	None
TS-1203	ERROR	Successfully created user	Successfully created user	None
TS-1204	ERROR	Failed to create user	Failed to create user	None
TS-1205	ERROR	Successfully created group	Successfully created group	None
TS-1206	ERROR	Failed to create group	Failed to create group	None
TS-1207	ERROR	Successfully updated user	Successfully updated user	None
TS-1208	ERROR	Failed to update user	Failed to update user	None
TS-1209	ERROR	Successfully updated users	Successfully updated users	None
TS-1210	ERROR	Failed to update users	Failed to update users	None
TS-1211	ERROR	Successfully updated group	Successfully updated group	None
TS-1212	ERROR	Failed to update group	Failed to update group	None
TS-1213	ERROR	Successfully updated password	Successfully updated password	None
TS-1214	ERROR	Failed to update password	Failed to update password	None
TS-1215	ERROR	Successfully deleted users	Successfully deleted users	None
TS-1216	ERROR	Failed to delete users	Failed to delete users	None
TS-1217	ERROR	Successfully deleted groups	Successfully deleted groups	None
TS-1218	ERROR	Failed to delete groups	Failed to delete groups	None
TS-1219	ERROR	Successfully assigned users to groups	Successfully assigned users to groups	None
TS-1220	ERROR	Failed to assign users to groups	Failed to assign users to groups	None

Code	Severity	Summary	Details	Action
TS-1221	ERROR	Failed to fetch profile pic	Failed to fetch profile pic	None
TS-1222	INFO	Successfully uploaded profile pic	None	None
TS-1223	ERROR	Failed to upload profile pic	Failed to upload profile pic	None
TS-1224	ERROR	Successfully assigned groups to group	Failed to assign user to group	None
TS-1228	ERROR	Successfully created role	Successfully created role	None
TS-1229	ERROR	Failed to create role	Failed to create role	None
TS-1230	ERROR	Successfully deleted role	Successfully deleted role	None
TS-1231	ERROR	Failed to delete role	Failed to delete role	None
TS-1232	ERROR	Successfully updated role	Successfully updated role	None
TS-1233	ERROR	Failed to update role	Failed to update role	None

Session Service Errors (1400 - 1599)

Code	Severity	Summary	Details	Action
TS-1400	ERROR	Failed to fetch session info	Failed to fetch session info	None
TS-1401	ERROR	Failed to login	Uh oh. We're not sure what happened. Please email the trace file to {adminEmail}.	None
TS-1402	ERROR	Failed to logout	Failed to logout	None
TS-1403	ERROR	Failed to save client state	Failed to save client state	None
TS-1404	ERROR	Failed to fetch login config	Failed to fetch login config	None
TS-1405	ERROR	Failed to fetch slack config	Failed to fetch slack config	None
TS-1406	ERROR	Health check failed	Health check failed	None
TS-1407	ERROR	Failed to fetch health portal token	Failed to fetch health portal token	None
TS-1408	ERROR	The health portal release	Uh oh. We're not sure what happened.	None

Code	Severity	Summary	Details	Action
		name could not be retrieved	Please email the trace file to {adminEmail}.	

Data Management Service Errors (1600 - 1799)

Code	Severity	Summary	Details	Action
TS-1600	ERROR	Failed to fetch data source types	Failed to fetch data source types	None
TS-1601	ERROR	Failed to fetch data source sample values	Failed to fetch data source sample values	None
TS-1602	ERROR	Failed to delete data source	Failed to delete data source	None
TS-1603	ERROR	Failed to execute DDL	Failed to execute DDL	None
TS-1604	ERROR	Failed to update schedule	Failed to update schedule	None
TS-1605	ERROR	Failed to reload tasks	Failed to reload tasks	None
TS-1606	ERROR	Failed to stop tasks	Failed to stop tasks	None
TS-1607	ERROR	Failed to get creation DDL	Failed to get creation DDL	None
TS-1608	ERROR	Failed to load from data source	Failed to load from data source	None
TS-1609	ERROR	Failed to create connection to data source	Failed to create connection to data source	None
TS-1610	ERROR	Failed to create data source	Failed to create data source	None
TS-1611	ERROR	Failed to connect to data source	Failed to connect to data source	None
TS-1612	ERROR	Failed to get data source connection field info	Failed to get data source connection field info	None
TS-1613	ERROR	Failed to get connection list for data source	Failed to get connection list for data source	None
TS-1614	ERROR	Failed to get connection attributes for data source	Failed to get connection attributes for data source	None
TS-1615	ERROR	Failed to get connections	Failed to get connections to data	None

Code	Severity	Summary	Details	Action
		to data source	source	
TS-1616	ERROR	Failed to fetch data source config	Failed to fetch data source config	None
TS-1617	ERROR	Failed to parse sql.	Failed to parse sql.	None
TS-1618	ERROR	Failed to execute sql.	Failed to execute sql.	None
TS-1619	INFO	Successfully created connection to data source	None	None
TS-1620	INFO	Successfully updated data upload schedule	None	None
TS-1621	ERROR	Failed to execute sql.	Please check the failing command, executed {1} statements successfully.	None
TS-1622	ERROR	Lightweight data-cache disabled	Lightweight data-cache disabled	None
TS-1623	INFO	Selected tables were queued for loading.	Selected tables were queued for loading.	None
TS-1624	ERROR	DataType conversion error.	No mapping found for source datatype to ThoughtSpot datatype.	None
TS-1625	INFO	Successfully reload task started.	None	None
TS-1626	INFO	Successfully connected to data source.	None	None
TS-1627	INFO	Successfully created data source.	None	None
TS-1628	INFO	Successfully stopped the tasks.	None	None
TS-1629	INFO	Successfully deleted the connection.	None	None
TS-1630	ERROR	There was an error deleting this connection.	None	None
TS-1631	INFO	Successfully executed the DDL.	None	None

Cluster Status Service Errors (1800 - 1899)

Code	Severity	Summary	Details	Action
TS-1800	WARNING	Failed to fetch cluster information from search service.	None	None
TS-1801	WARNING	Failed to fetch table detail information from search service.	None	None
TS-1802	WARNING	Failed to fetch cluster information from database service.	None	None
TS-1803	WARNING	Failed to fetch table detail information from databse service.	None	None
TS-1804	WARNING	Failed to fetch cluster information from cluster management service.	None	None
TS-1805	WARNING	Failed to fetch detail information from cluster management service.	None	None
TS-1806	WARNING	Failed to fetch log from cluster management service.	None	None
TS-1807	WARNING	Failed to fetch snapshot list from cluster management service.	None	None
TS-1808	WARNING	Failed to fetch cluster information from alert management service.	None	None
TS-1809	WARNING	Failed to fetch cluster information from event service.	None	None
TS-1810	WARNING	Failed to fetch alerts information from alert management service.	None	None
TS-1811	WARNING	Failed to fetch events information from alert management service.	None	None
TS-1812	INFO	Thanks for your feedback!	None	None
TS-1813	WARNING	Sorry! Unable to submit the feedback at this moment!	None	None
TS-1814	INFO	Successfully exported objects. File can be found at {1}.	None	None

Code	Severity	Summary	Details	Action
TS-1815	ERROR	Sorry! Unable to export objects at this moment!	What happened? {1}.	None
TS-1816	INFO	Successfully imported objects	None	None
TS-1817	ERROR	Sorry! Unable to import objects at this moment!	What happened? {1}.	None
TS-1818	INFO	Successfully deleted data source object(s).	None	None

Callosum API Errors (9000 - 9199)

Code	Severity	Summary	Details	Action
TS-9000	ERROR	The data you are trying to delete has some dependencies	Some objects depend on the data you are trying to delete	delete the dependencies before deleting this data.
TS-9001	ERROR	Uh oh. We're not sure what happened.	Please email the trace file to {adminEmail}.	None
TS-9002	ERROR	Could not authorize user	Try logging in again	None
TS-9003	ERROR	Uh oh. We're not sure what happened.	Please email the trace file to {adminEmail}.	None
TS-9004	WARNING	Still loading data, come back soon	None	None
TS-9005	ERROR	Uh oh. We're having trouble getting data for this request.	Please email the trace file to {adminEmail}.	None
TS-9006	ERROR	Uh oh. We're having trouble getting data for this request.	Please email the trace file to {adminEmail}.	None
TS-9007	ERROR	Uh oh. We're having trouble getting data for this request.	Please email the trace file to {adminEmail}.	None
TS-9008	ERROR	Something went	Uh oh. We're not sure what hap-	None

Code	Severity	Summary	Details	Action
		wrong with your search	pened. Please email the trace file to {adminEmail}.	
TS-9009	ERROR	The calculation engine has timed out. Please try again.	Please email the trace file to {adminEmail}.	None
TS-9010	ERROR	Cannot open Object	Object cannot be opened due to errors in some of its dependencies	None
TS-Blink Generated Errors (9500 - 9599)				
TS-9500	WARNING	Cannot connect to the calculation engine. Please try again soon.	None	None
TS-9501	WARNING	The calculation engine has timed out. Please try again.	None	None
TS-9502	WARNING	Cannot connect to the search engine. Please try again soon.	None	None
TS-9503	WARNING	The search engine has timed out. Please try again.	None	None
TS-9504	ERROR	Cannot open {1}	{1} cannot be opened due to errors in the following dependencies 1 - Type of the object Table/Answer/Pinboard etc.	None
TS-9505	WARNING	We're still indexing this data, try again soon	None	None
TS-9506	ERROR	Object is not present in the system	{1} is not present in the system 1 - Type of the object Table/Answer/Pinboard etc.	None
TS-9507	ERROR	ThoughtSpot is unreachable.	None	None

Code	Severity	Summary	Details	Action
		Please try again soon		

Common Errors (10000 - 10099)

Code	Severity	Summary	Details	Action
TS-10000	ERROR	A system error has occurred	Uh oh. We're not sure what happened. Please contact your administrator.	None
TS-10001	ERROR	Connection failed	The metadata store is not reachable.	Please contact your administrator
TS-10002	ERROR	The input is invalid	Input from the client to the server is invalid.	Please contact your administrator
TS-10003	ERROR	Unfortunately, you can't do that	You are not authorized to perform {1}. # {1} – action user is not authorized for	Please request access from your administrator
TS-10004	ERROR	The user could not be authorized	User {0} is not authorized to perform {1}. # {0} – name of the user # {1} – action user is not authorized for	Please request access from your administrator
TS-10005	ERROR	The base object is missing	An underlying object referenced by this object is missing in store.	Please contact your administrator
TS-10006	ERROR	The connection to Zookeeper has failed	Zookeeper is not reachable.	Please contact your administrator
TS-10007	ERROR	There's invalid parameter(s)	Invalid parameter values: {0}.	Please contact your administrator
TS-10008	ERROR	The user cannot be found	User {0} not found in store. # {0} – name of the user	Please contact your administrator
TS-10009	ERROR	Cannot add group	This group already belongs to the group you are trying to add it to.	None

Falcon Errors (10600 - 10699)

Code	Severity	Summary	Details	Action
TS-10603	ERROR	Falcon query cancelled	None	None

Data Errors (11000 - 11099)

Code	Severity	Summary	Details	Action
TS-11001	ERROR	Invalid row	None	None
TS-11002	ERROR	Invalid table/query resultset	None	None
TS-11003	ERROR	Invalid column identifier	None	None
TS-11004	ERROR	Invalid visualization identifier	None	None
TS-11005	ERROR	No data	Query execution resulted in no data.	None
TS-11006	ERROR	Query execution failed	Error in query execution to Falcon.	None
TS-11007	ERROR	Answer data generation failed	Error in Answer data generation for Sage input.	None
TS-11008	ERROR	Data export failed	None	None
TS-11009	ERROR	Data generation failed	Error in data generation in Callosum.	None

Report Generation Errors (12000 - 13000)

Code	Severity	Summary	Details	Action
TS-12700	ERROR	Error while exporting data file.	None	None
TS-12701	ERROR	Invalid input.	The definition of the job is invalid.	None
TS-12702	ERROR	No author provided.	None	None
TS-12703	ERROR	No pinboard provided.	None	None

Code	Severity	Summary	Details	Action
TS-12704	ERROR	No recipients provided.	None	None
TS-12705	ERROR	This format is not supported.	None	None
TS-12706	ERROR	No job name provided.	None	None
TS-12707	ERROR	No job description provided.	None	None
TS-12708	ERROR	Pinboard data export error.	None	None
TS-12709	ERROR	Visualization data export error.	None	None
TS-12710	ERROR	User data unavailable.	None	None
TS-12711	ERROR	Configuration information un-available.	None	None
TS-12712	ERROR	There are too many recipients.	The max number of recipients is 1000.	None
TS-12713	ERROR	Attachment size limit exceeded.	None	None
TS-12714	ERROR	Recipient domain is not whitelisted.	None	None

More Metadata Errors (13000 - 13099)

Code	Severity	Summary	Details	Action
TS-13001	ERROR	Schema creation failed	Error creating database schema.	None
TS-13002	ERROR	Views creation failed	Error creating view.	None
TS-13003	ERROR	The object can-not be found in store	Object with Id: {0} of type: {1} not found. # {0} – identity of the ob-ject # {1} – type of ob-ject	None
TS-13004	ERROR	The object is in an invalid state	Oject with Id: {0} of type: {1} in invalid state. # {0} – identity of the object # {1} – type of object	None
TS-13005	ERROR	Object already	Object with Id: {0} of	None

Code	Severity	Summary	Details	Action
		exists	type: {1} already exists. # {0} – identity of the object # {1} – type of object	
TS-13006	ERROR	Invalid object type	Invalid type: {0} provided. # {1} – type of object	None
TS-13007	ERROR	Invalid Sage question	Insufficient or invalid input from Sage: {0}. # {0} – the invalid input	None
TS-13008	ERROR	Invalid Sage question	Input from from Sage – missing columns of type: {0}. # {0} – column type	None
TS-13009	ERROR	Invalid Sage question	Invalid input from Sage – invalid expression: {0}. # {0} – the invalid expression	None
TS-13010	ERROR	Sending logical metadata to Sage failed	Sending logical metadata to Sage failed due to: {0}. # {0} – reason for failure	None
TS-13011	ERROR	Answer generation failed	Answer generation failed due to: {0}. # {0} – reason for failure	None
TS-13012	ERROR	Worksheet generation failed	Worksheet generation failed due to: {0}. # {0} – reason for failure	None
TS-13013	ERROR	Service provider unavailable	Service provider unavailable: {0}. # {0} – provider details	None
TS-13015	ERROR	Physical model not loaded	None	None
TS-13016	ERROR	Invalid physical schema proto	Inconsistency in physical schema from Falcon: {0}. # {0} – error details	None
TS-13017	ERROR	Invalid duplicate columns	Duplicate columns: {0}. # {0} – List of duplicate column identities	None
TS-13018	ERROR	Cyclic relationship	Detected cycles: {0}. # {0} – cycle details	None

Code	Severity	Summary	Details	Action
TS-13019	WARNING	Older physical schema version received	Schema update for older version: {0} received and ignored. # {0} – received version number	None
TS-13020	ERROR	Invalid relationship	Attempted to create invalid relationship: {0}. # {0} – relationship details	None
TS-13022	ERROR	Invalid filter values: {values}	None	None
TS-13023	ERROR	Creating relationship failed.	None	None
TS-13024	ERROR	Deleting schema failed.	None	None
TS-13025	ERROR	Expression validation failed.	None	None
TS-13026	INFO	Load schedule successfully disabled.	None	None
TS-13027	ERROR	Load schedule could not be disabled.	None	None
TS-13028	ERROR	Objects fetched from the connection are invalid for editing datasource.	None	To proceed with editing the datasource, please edit the connection below to fetch valid source objects.
TS-13029	INFO	Successfully edited data source connection.	None	None
TS-13030	ERROR	Connection test failed.	None	Please verify connection attributes.

Loading Errors (30000 - 30099)

Code	Severity	Summary	Details	Action
TS-30000	ERROR	Table is not ready (data loading in progress).	None	None

Timely Errors (60000 - 64999)

Code	Severity	Summary	Details	Action
TS-60000	ERROR	Failed to initialize.	None	None