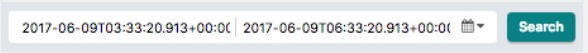
Search Cheat Sheet

This cheat sheet describes how to employ the most commonly used search methods in Loggly.

I want to...	I use...	Sample queries
Search all of my logs over a recent time period	**The calendar icon to select the time period via drop-downs or by entering a dates for a custom time period.**	
Find a single word (called a token in Loggly terms) in any field of my logs	**The search box. Type the word.** Learn how Loggly parses your logs to give your searches maximum flexibility. This can affect your search results.	`timeout`
Find all of two or more words in a single event	**space** or **AND** Strictly speaking, you don't need *AND* because we use it by default. But if you use *AND*, it must be uppercase.	`frontend apache` `frontend AND apache`
Find any of two or more words in a single event	**OR** Must be uppercase.	`apache OR nginx`

Search Cheat Sheet

LOGGLY DOCUMENTATION

I want to...	I use...	Sample queries
Find log events that do NOT contain a particular word	**NOT** or **-** *NOT* must be uppercase.	`-apache`
Find a phrase or a compound word (see below for a definition of a compound word)	**Quotations around the words** *Note*: Many hostnames, variable names, file names, etc. are compound words, so should be quoted.	`"unexpected exception"` `"apache01"`
Find events to match the start of a word	*** to match more than one character** *Note*: Wildcards can only be used standalone or when trailing a partial term.	`err*`

Search Cheat Sheet

LOGGLY DOCUMENTATION

I want to...	I use...	Sample queries
Find a word in a specific field of my logs	**Enter the field name followed by a colon, then the word.** • When you use a field name, you're restricting your search to that part of your structured data. • If the data includes nested JSON, the field name should be given with dot notation. • There can be NO spaces between the field name, colon, or word. • All of the operators described above are available with fields.	`syslog.host:frontend` `json.sport.baseball:Giants`
Find log events with fields containing numeric values	**`field_name:[low# TO high#]`** **`field_name:>low#`** **`field_name:<high#`** **`field_name:>=low#`** **`field_name:<=high#`** A field name must be supplied. There should be no spaces between the colon, the operator, and the value.	`json.responseTime:[100 TO 500]` `json.responseTime:>=500`
Do power searches with regular expressions	**Read the Loggly documentation for details.**	

FREE TRIAL

Search Cheat Sheet

LOGGLY DOCUMENTATION

About Loggly tokenization

Loggly analyzes the text of every log event as we receive it. We do this in order to create the maximum possible number of search tokens (i.e., words) for each log event, which gives you more power and flexibility when you are searching your logs.

The process

Step 1: We split the original text into words whenever we find a non-alphanumeric character

Example: "`NullPointerException thrown by validate_path for /tmp/foo/bah on apache01-dev`"

Becomes: `NullPointerException, thrown, by, validate, path, for, tmp, foo, bah, on, apache01, dev`

'_' in `validate_path`, '/' in `/tmp/foo/bar`, and '-' in `apache01-dev` are non-alphanumeric, and hence treated as separators

Step 2: We split each of these words whenever there is a change of case or change from alpha to numeric (or vice versa)

Example: `NullPointerException` Becomes: `Null, Pointer, Exception`

Example: `apache01` Becomes: `apache, 01`

Step 3: We lowercase all of the alphabetic characters, leaving us with:
`null, pointer, exception, thrown, by, validate, path, for, tmp, foo, bah, on, apache, 01, dev`

FREE TRIAL

Search Cheat Sheet

LOGGLY DOCUMENTATION

Compound words

We completely decompose what we call compound words, (e.g., “`NullPointerException`”, “`validate_path`”, “`/tmp/foo/bah`”, and “`apache01-dev`”) above and beyond what would happen if we simply “split at whitespace,” like your brain is doing as you read this.

Tips

To search for the “unsplit” compound word, just use quotes. **It is (almost) always safe to quote the words you are searching for.** Do it when you’re cutting and pasting a section of log line into the search box. The only exception is when using wildcards, which should never be quoted. *Note*: Quoting does not perform an exact search but searches for each token in the same order with special characters removed.	`"NullPointerException" "validate_path" "/tmp/foo/bah", "apache01-dev"`
Because we decompose your data, you can find related compound words.	Searching for `Exception` will find ALL exceptions you’ve logged: `IOException`, `InterruptedException`, etc. Searching for `apache` will find ALL of your apache hosts: `apache01-dev`, `apache02-prod`, `apache21-prod`
If you match too many events when searching for an individual word, you can use all of the techniques in this cheat sheet to narrow your search.	Searching for `exception -"IOException"` will exclude events containing `IOException` Searching for `syslog.hostname:apache` will limit your search to the hostname field, avoiding matches for apache in other parts of your log message

