elastic Try Free Customers Docs **IMPORTANT**: No additional bug fixes or documentation updates will be released for this On this page version. For the latest information, see the current release documentation. Terms query Kibana Guide [7.9] » Discover » Search data » Kibana Query Language **Boolean queries** Range queries « Search data Lucene query syntax » Date range queries Exist queries Wildcard queries Kibana Query Language Nested field queries The Kibana Query Language (KQL) makes it easy to find the fields and syntax for your Elasticsearch query. If you have the Basic tier or above, simply place your cursor in the Search field. As you type, you'll get suggestions for fields, values, and operators. **Most Popular** Get Started with Elasticsearch Filter results that contain host.geo.city_name logs-* host.geo.continent_name Filter results that contain host.geo.continent_name host.geo.country_iso_code host.geo.country_name ELK for Logs & Metrics Filter results that contain host.geo.country_name ⇒ Fil host.geo.name Availabl host.geo.region_iso_code Filter results that contain host.geo.region_iso_code Popula host.geo.region_name Intro to Kibana _id ① _index _score > Sep 24, 2020 @ 10:52:30.957 { "@timestamp": "2020-09-24T17:52:30.957Z", "ecs.version": "1.5.0", "event": { "dataset": "elastic.agent.filebeat" }, "l og.origin": { "file.name": "log/input.go", "file.line": 226 }, "input": { "type": "log" }, "data_stream": { "type": "log s", "dataset": "elastic.agent.filebeat", "namespace": "default" }, "ecs": { "version": "1.5.0" }, "message": "input stat es cleaned up. Before: 1, After: 1, Pending: 0", "log": { "offset": 672585, "file": { "path": "/Users/gailchappell/Downl oads/elastic-agent-7.9.2-darwin-x86_64/data/logs/default/filebeat-json.log" } }, "log.level": "debug", "log.logger": "in put", "host": { "name": "Gails-MacBook-Pro.local", "architecture": "x86_64", "os": { "kernel": "18.7.0", "build": "18610 12". "platform": "darwin". "version": "10.14.6". "familv": "darwin". "name": "Mac OS X" }. "id": "8B282311-BDB9-5AD9-8E4 ① _type (atimestamp t agent.ephemeral_id @ agent.hostname t agent.id Recommended for you If you prefer to use Kibana's legacy query language, based on the Lucene query syntax, click Search API | Elasticsearch KQL next to the Search field, and then turn off KQL. Query string query Elasticsearch Terms query ■ Mapping | Elasticsearch A terms query matches documents that contain one or more **exact** terms in a field. Reindex API | Elasticsearch ■ Install Elasticsearch with Docker To match documents where the response field is 200: Elasticsearch response:200 + Kibana Guide: 7.9 To match documents with the phrase "quick brown fox" in the message field. What is Kibana? What's new in 7.9 message:"quick brown fox" + Get started + Set up Kibana Without the quotes, the query matches documents regardless of the order in which they appear. Documents with "quick brown fox" match, and so does "quick fox brown". - Discover Create an index pattern Terms without fields are matched against the default field in your index settings. If Set the time filter a default field is not set, terms are matched against all fields. For example, a query for response: 200 searches for the value 200 in the response field, but a query for Search data just 200 searches for 200 across all fields in your index. Kibana Query Language Lucene query syntax Save a search **Boolean queries** Save a query KQL supports or, and, and not. By default, and has a higher precedence than or. To override Filter by field the default precedence, group operators in parentheses. View document data To match documents where response is 200, extension is php, or both: View a document in context View field data statistics response:200 or extension:php + Dashboard + Canvas To match documents where response is 200 and extension is php: + Maps + Machine learning response:200 and extension:php + Graph + Visualize To match documents where response is 200 or 404. Observability Logs response:(200 or 404) Metrics + APM To match documents where response is 200 and extension is either php or css: Uptime response:200 and (extension:php or extension:css) + Elastic Security + Dev Tools To match documents where response is 200 and extension is php or extension is css, and + Stack Monitoring response is anything: + Stack Management Ingest Manager response:200 and extension:php or extension:css + Reporting + Alerting and Actions To match documents where response is not 200: + REST API Kibana plugins not response:200 Accessibility + Breaking Changes To match documents where response is 200 but extension is not php or css. + Release Notes response:200 and not (extension:php or extension:css) + Developer guide To match multi-value fields that contain a list of terms: tags:(success and info and security) Range queries KQL supports >, >=, <, and <= on numeric and date types. account_number >= 100 and items_sold <= 200</pre> Date range queries Typically, Kibana's time filter is sufficient for setting a time range, but in some cases you might need to search on dates. Include the date range in quotes. @timestamp < "2021-01-02T21:55:59" @timestamp < "2021-01" @timestamp < "2021"</pre> **Exist queries** An exist query matches documents that contain a value for a field, in this case, response: response:* Wildcard queries To match documents where machine.os starts with win, such as "windows 7" and "windows 10": machine.os:win* To match multiple fields: machine.os*:windows 10 This syntax is handy when you have text and keyword versions of a field. The query checks machine.os and machine.os.keyword for the term windows 10. Nested field queries A main consideration for querying nested fields is how to match parts of the nested query to the individual nested documents. You can: • Match parts of the query to a single nested document only. This is what most users want when querying on a nested field. • Match parts of the query to different nested documents. This is how a regular object field works. This query is generally less useful than matching to a single document. In the following document, items is a nested field. Each document in the nested field contains a name, stock, and category. "grocery_name": "Elastic Eats", "items": ["name": "banana", "stock": "12", "category": "fruit" "name": "peach", "stock": "10", "category": "fruit" "name": "carrot", "stock": "9", "category": "vegetable" "name": "broccoli", "stock": "5", "category": "vegetable" Match a single document To match stores that have more than 10 bananas in stock: items:{ name:banana and stock > 10 } items is the nested path. Everything inside the curly braces (the nested group) must match a single nested document. The following query does not return any matches because no single nested document has items:{ name:banana and stock:9 }

bananas with a stock of 9.

The following subqueries are in separate nested groups and can match different nested documents:

Match different documents

items:{ name:banana } and items:{ stock:9 }

name:banana matches the first document in the array and stock:9 matches the third document in the array. Match single and different documents

To find a store with more than 10 bananas that **also** stocks vegetables: items:{ name:banana and stock > 10 } and items:{ category:vegetable }

The first nested group (name:banana and stock > 10) must match a single document, but the

category: vegetables subquery can match a different nested document because it is in a separate group.

Nested fields inside other nested fields KQL supports nested fields inside other nested fields—you have to specify the full path. In this

"level1": ["level2": ["prop1": "foo", "prop2": "bar" "prop1": "baz", "prop2": "qux"

level1.level2:{ prop1:foo and prop2:bar }

To match on a single nested document:

« Search data

Security

Kibana

Beats

Pricing

PRODUCTS & SOLUTIONS

document, level1 and level2 are nested fields:

COMPANY

Enterprise Search Careers Sign up Email address Board of Directors Observability Contact Elastic Stack RESOURCES Elasticsearch Documentation What is the ELK Stack? Logstash What is Elasticsearch? Migrating from Splunk Subscriptions Compare AWS Elasticsearch US Public Sector Trademarks | Terms of Use | Privacy | Brand | Sitemap

Lucene query syntax »

Subscribe to our newsletter