

# Understanding the search syntax

## In this article

- Query for values greater or less than another value
- Query for values between a range
- Query for dates
- Exclude results that match a qualifier
- Exclude results with specific keywords
- Use quotation marks for queries with whitespace
- Queries with usernames

When searching GitHub Enterprise Server, you can construct queries that match specific numbers and words.

## Query for values greater or less than another value



You can use `>`, `>=`, `<`, and `<=` to search for values that are greater than, greater than or equal to, less than, and less than or equal to another value.

Query	Example
<code>&gt;n</code>	<code>cats stars:&gt;1000</code> matches repositories with the word "cats" that have more than 1000 stars.
<code>&gt;=n</code>	<code>cats topics:&gt;=5</code> matches repositories with the word "cats" that have 5 or more topics.
<code>&lt;n</code>	<code>cats size:&lt;10000</code> matches code with the word "cats" in files that are smaller than 10 KB.
<code>&lt;=n</code>	<code>cats stars:&lt;=50</code> matches repositories with the word "cats" that have 50 or fewer stars.

You can also use [range queries](#) to search for values that are greater than or equal to, or less than or equal to, another value.

Query	Example
<code>n..*</code>	<code>cats stars:10..*</code> is equivalent to <code>stars:&gt;=10</code> and matches repositories with the word "cats" that have 10 or more stars.
<code>*..n</code>	<code>cats stars:*..10</code> is equivalent to <code>stars:&lt;=10</code> and matches repositories with the word "cats" that have 10 or fewer stars.

## Query for values between a range



You can use the range syntax `n..n` to search for values within a range, where the first number N is the lowest value and the second is the highest value.

Query	Example
<code>n..n</code>	<a href="#">cats stars:10..50</a> matches repositories with the word "cats" that have between 10 and 50 stars.

## Query for dates [🔗](#)

You can search for dates that are earlier or later than another date, or that fall within a range of dates, by using `>`, `>=`, `<`, `<=`, and [range queries](#). Date formatting must follow the [ISO8601](#) standard, which is `YYYY-MM-DD` (year-month-day).

Query	Example
<code>&gt;YYYY-MM-DD</code>	<a href="#">cats created:&gt;2016-04-29</a> matches issues with the word "cats" that were created after April 29, 2016.
<code>&gt;=YYYY-MM-DD</code>	<a href="#">cats created:&gt;=2017-04-01</a> matches issues with the word "cats" that were created on or after April 1, 2017.
<code>&lt;YYYY-MM-DD</code>	<a href="#">cats pushed:&lt;2012-07-05</a> matches repositories with the word "cats" that were pushed to before July 5, 2012.
<code>&lt;=YYYY-MM-DD</code>	<a href="#">cats created:&lt;=2012-07-04</a> matches issues with the word "cats" that were created on or before July 4, 2012.
<code>YYYY-MM-DD..YYYY-MM-DD</code>	<a href="#">cats pushed:2016-04-30..2016-07-04</a> matches repositories with the word "cats" that were pushed to between the end of April and July of 2016.
<code>YYYY-MM-DD..*</code>	<a href="#">cats created:2012-04-30..*</a> matches issues created after April 30th, 2012 containing the word "cats."
<code>*..YYYY-MM-DD</code>	<a href="#">cats created:*..2012-07-04</a> matches issues created before July 4th, 2012 containing the word "cats."

You can also add optional time information `THH:MM:SS+00:00` after the date, to search by the hour, minute, and second. That's `T`, followed by `HH:MM:SS` (hour-minutes-seconds), and a UTC offset ( `+00:00` ).

Query	Example
<code>YYYY-MM-DDTHH:MM:SS+00:00</code>	<a href="#">cats created:2017-01-01T01:00:00+07:00..2017-03-01T15:30:15+07:00</a> matches issues created between January 1, 2017 at 1 a.m. with a UTC offset of <code>07:00</code> and March 1, 2017 at 3 p.m. with a UTC offset of <code>07:00</code> .
<code>YYYY-MM-DDTHH:MM:SSZ</code>	<a href="#">cats created:2016-03-21T14:11:00Z..2016-</a>

**04-07T20:45:00Z** matches issues created between March 21, 2016 at 2:11pm and April 7, 2016 at 8:45pm.

## Exclude results that match a qualifier [↗](#)

You can narrow down search results by excluding one or more subsets. To exclude all results that are matched by a qualifier, prefix the search qualifier with a hyphen ( - ).

Query	Example
<code>- QUALIFIER</code>	<code>cats stars:&gt;10 -language:javascript</code> matches repositories with the word "cats" that have more than 10 stars but are not written in JavaScript.
<code>- QUALIFIER</code>	<code>mentions:defunkt -org:github</code> matches issues mentioning @defunkt that are not in repositories in the GitHub organization.

## Exclude results with specific keywords [↗](#)

You can exclude results containing a certain word, using the `NOT` syntax. The `NOT` operator can only be used for string keywords. It does not work for numerals or dates.

Query	Example
<code>NOT</code>	<code>hello NOT world</code> matches repositories that have the word "hello" but not the word "world."

## Use quotation marks for queries with whitespace [↗](#)

If your search query contains whitespace, you will need to surround it with quotation marks. For example:

- `cats NOT "hello world"` matches repositories with the word "cats" but not the words "hello world."
- `build label:"bug fix"` matches issues with the word "build" that have the label "bug fix."

Some non-alphanumeric symbols, such as spaces, are dropped from code search queries within quotation marks, so results can be unexpected.

## Queries with usernames [↗](#)

If your search query contains a qualifier that requires a username, such as `user` , `actor` , or `assignee` , you can use any GitHub Enterprise Server username, to specify a specific person, or `@me` , to specify the current user.

Query	Example
<code>QUALIFIER:USERNAME</code>	<code>author:nat</code> matches commits authored by @nat
<code>QUALIFIER:@me</code>	<code>is:issue assignee:@me</code> matches issues assigned to the person viewing the results

You can only use `@me` with a qualifier and not as search term, such as `@me main.workflow`.

## Legal

© 2023 GitHub, Inc. [Terms](#) [Privacy](#) [Status](#) [Pricing](#) [Expert services](#) [Blog](#)