

Rules

A classic query in JQL (also known as a 'clause') consists of a **field**, followed by an **operator**, followed by one or more values or **functions**.

Field - Fields are different types of information in the system. *fixVersion, issue type, date created, etc.*

Operator - Operators are the heart of the query. They relate the field to the value. *equals (=), not equals (!=), less than (<), etc.*

Function - A function performs a calculation on either specific Jira data or the function's content.

Keyword - Keywords are specific words in the language that have special meaning.

Fields

Jira Software

Assignee	Attachments	Category
Comment	Component	Component
Created	Creator	Description
Due	Environment	Epic link
Filter	Fix version	Issue key
Issue link type	Labels	Last viewed
Level	Parent	Priority
Project	Reporter	Resolution
Resolved	Sprint	Status
Summary	Text	Type
Updated	Voter	Watcher

JSD Specific

Approvals	Customer Request Type	Request channel type
Request last activity time	SLA	

Keywords

AND	Combine multiple clauses
OR	Choice between multiple clauses
NOT	Negate a clause or JQL query
EMPTY , NULL	Field without a value
ORDER BY	Specify the sorting order

Official cheat sheet

A better version of our cheat sheet is available here :
<https://bit.ly/3rFJxTI>

Operators

= , !=	equals, not equals
> , >=	(strictly) greater than
< , <=	(strictly) lesser than
~ , !~	contains, do not contain
changed	field value has been changed before
is, is not	field has no value (only with EMPTY or NULL)
was, was not	field has or had the specified value. Only works with <i>Assignee, Fix Version, Priority, Reporter, Resolution</i> and <i>Status</i> fields
was in, was not in	field has or had any of multiple specified values
in, not in	field is one of multiple specified value



By **MartechElements**

cheatography.com/martechelements/

Published 21st September, 2020.

Last updated 12th March, 2021.

Page 1 of 2.

Sponsored by **ApolloPad.com**

Everyone has a novel in them. Finish Yours!

<https://apollopad.com>

Functions

User	Issue	Time
currentUser()	linkedIssues()	now()
membersOf()	latestReleasedVersion()	remaining()
votedIssues()	releasedVersions()	futureSprints()
projectsLeadByUser()	unreleasedVersions()	issueHistory()
componentsLeadByUser()	openSprints()	startOfDay/Week/Month/Year()
updatedBy()	standardIssueTypes()	endOfDay/Week/Month/Year()
watchedIssues()	subtaskIssueTypes()	elapsed() *JSD only
projectsWhereUserHasRole()	completed() *JSD only	paused() *JSD only
pendingBy() *JSD only	myApproval() *JSD only	running() *JSD only
approver() *JSD only	breached() *JSD only	pending() *JSD only

-

Note: For time range you can either use the standard date format "-yyyy-MM-dd", for example **updated < "2020/09/16"**.

Or use "w" (weeks), "d" (days), "h" (hours) or "m" (minutes) to specify a date relative to the current time, for example: **updated < "-2w"**.

Advanced search

Wildcard searches

To do a single character wildcard search, use the "?" symbol

To do a multiple character wildcard search, use the "*" symbol.

Wildcard characters need to be enclosed in quote-marks. E.g. summary ~ "cha?k"

Fuzzy searches

To do a fuzzy search, use the "~" symbol at the end of a single word term.

Proximity searches

To do a proximity search, use the "~", symbol at the end of a phrase.



By **MartechElements**

cheatography.com/martechelements/

Published 21st September, 2020.

Last updated 12th March, 2021.

Page 2 of 2.

Sponsored by **ApolloPad.com**

Everyone has a novel in them. Finish

Yours!

<https://apollopad.com>