Jira Search Queries

JQL examples to find issues fast.

Terms

**Query** – a question.

A query is another word for a question. Each time you use a search engine, you’re querying the internet for web pages. When you search in Jira, you’re querying the database for issues that match your specifications.

Example: How many bugs did the Marketing team report this year?

**JQL** – Jira Query Language

JQL is the way to search through thousands of issues to find the few you’re looking for. It’s also the way to control which issues appear on a board, and the data displayed on dashboards and in reports.



Examples

Use these sample queries as a starting point and further customize them to meet your needs. Try changing values and combining different clauses.

|  |  |  |
| --- | --- | --- |
| **Example** | **JQL** | **Notes** |
| Closed tasks | type = task AND status = closed | Returns all task issues in the closed status |
| Tasks or closed issues | type = task OR status = closed | Returns all task issues and issues (all types) in the closed status |
| Issues in multiple statuses | status IN (open, new) | Returns issues in the open or new statuses |
| Incomplete issues | * statusCategory != done
* status NOT IN (closed, done)
* resolution is EMPTY
 | Three different ways to find incomplete issues |
| Assigned to me | assignee = currentUser() AND statusCategory != done ORDER BY created ASC | Issues assigned to me, that need action, ordered by oldest creation date, in ascending order |
| Created by me | creator = currentUser() ORDER BY created DESC | Issues I created recently in descending order |
| Created recently | created >= startOfDay | Issues created today |
| “Stale” issues | statusCategory != done AND updated <= -30d ORDER BY updated ASC | Incomplete issues not updated in the last thirty days |
| Not resolved by issue’s due date | resolution is EMPTY AND due <= now() | Unresolved issues with a past due date |
| Status changed this week | status CHANGED TO closed AFTER startOfWeek() BEFORE endOfWeek() | Issues currently or previously in a specific status in the current week |
| Due this month or next month | due >= startOfMonth() AND due <= endOfMonth(1) | Note: Notice the “1” variable in the “endOfMonth(1)” function |
| Due soon | due > now() AND due < 6w AND statusCategory != done | Issues with due dates within the next six weeks |
| Keyword “like” search or labels search | * summary ~ reimbursement OR description ~ reimbursement
* text ~ reimbursement OR labels = reimbursement
 | Note: The “text” clause includes issues with “reimbursement” in the “Summary” (title), “Description”, or “Comment” fields |
| Without a specific label | labels != reimbursement OR labels is EMPTY | Issues not labeled “reimbursement.” Note: Don’t forget to include the “OR labels is EMPTY” clause. |
| Linked issues | issue in linkedIssues(ISSUE-123) | All issues linked to ISSUE-123 |
| Linked issues by link type | issue in linkedIssues(ISSUE-123, duplicates) | All issues that duplicate ISSUE-123 |
| Custom field has a value | * field-name is not EMPTY
* cf[12345] is not EMPTY
 | Note: Use a custom field’s name or ID the format: cf[custom-field-id]. |
| Jira project category | category = internal | Returns all issues in Jira projects categorized as internal |
| Work logged | timespent >= 1h | Issues with one or more time logs greater or equal to one hour |
| Cascading select field | field-name in cascadeOption(parent-option,child-option) | For fields where options in the second menu are determined by the selection in the first menu |