

# JQL Functions to Query on Project Properties

## On this page

- [projectsWhereTextProperty\( project\\_property, operator, string\\_literal | regular\\_expression \)](#)
- [projectsWhereNumberProperty\( project\\_property, operator, numeric\\_literal \)](#)
- [projectsWhereDateProperty\( project\\_property, operator, date\\_literal \)](#)

**Jira Workflow Toolbox** provides 3 JQL functions to query on values of **Project Properties**. These functions complements conditions, validators and post-functions provided by the plugin to deal with project properties.

## projectsWhereTextProperty( project\_property, operator, string\_literal | regular\_expression )

This function is used for searching **text** or **string** values in project properties.

- **project\_property**: name of a project property containing a text or string value.
- **operator**: comparison operator to be applied.
  - =: strict equal operator.
  - !=: strict not equal operator.
  - ~: approximate equal operator (case insensitive and trims blanks).
  - !~: approximate not equal operator (case insensitive and trims blanks).
  - matches: value of project property matches a regular expression.
  - doesn't match: value of project property doesn't match a regular expression.
  - >, >=, <, <=: lexicographical order operators.
- **string\_literal | regular\_expression**: a string literal used for comparison, or a regular expression in the case of matches operator.

### Examples

- `projects in projectsWhereTextProperty( country, "=", Spain )` : list of projects where property 'country' is equal to 'Spain', i.e. string {country=Spain} is present in project's description.
- `projects in projectsWhereTextProperty( country, matches, "France|Germany" )` : list of projects where property 'country' matches regular expression `France|Germany`, i.e. string {country=France} or {country=Germany} are present in project description.
- `projects in projectsWhereTextProperty( country, "~", Spain )` : list of projects where property 'country' is equal to 'Spain', 'SPAIN', ' SpAin ', etc.

## projectsWhereNumberProperty( project\_property, operator, numeric\_literal )

This function is used for searching **numeric** values in project properties.

- **project\_property**: name of a project property containing a numeric value.
- **operator**: comparison operator to be applied: =, !=, >, >=, <, <=.
- **numeric\_literal**: number to be compared with the value stored in the project property.

### Examples

- `projects in projectsWhereNumberProperty( maximumNumberOfTryOuts, ">", 3 )`
- `projects in projectsWhereNumberProperty( numberOfLevels, "=", 5 )`

## projectsWhereDateProperty( project\_property, operator, date\_literal )

This function is used for searching properties containing a date in format `yyyy/MM/dd [hh:mm]` or `yyyy-MM-dd [hh:mm]`. You can also use `w` (weeks) or `d` (days) to specify a date relative to the current date. Quote-marks " must be used.

- **project\_property**: name of a project property containing a date value.
- **operator**: comparison operator to be applied: =, !=, >, >=, <, <=.
- **date\_literal**: date to be compared with date stored in the project property.

## Examples

- `projects in projectsWhereDateProperty( birthDay, ">", 2012/01/01 )` : list of projects where property '**birthDay**' stores a date after '2012/01/01'.
- `projects in projectsWhereDateProperty( nextMeeting, ">=", -1w )` : list of projects where property '**nextMeeting**' stores a date within the last seven days.