

filterByPredicate()

Supported list types

[Number list](#) | [Text list](#) | [Issue list](#) | [Use cases and examples](#)

1

2

3

Number list

This function filters a **number list** by a given **logical expression** where **^** is used for referencing the **c** **urrent list value**.

It basically **iterates** over each list **element**, **checks** whether the logical expression returns **true**, and if it does, **includes** the element in the **output**.

Syntax

```
filterByPredicate(numberList, logicalExpression) #Output: Number list
```

Examples

Parser expression	Description
<code>filterByPredicate([1, 2, 3, 4], ^% > 2)</code>	This example returns a number list with values greater than 2: [3, 4]
<code>filterByPredicate([1, 2, 3, 4], modulus(^%, 2) = 0)</code>	This example returns a number list with even values : [2, 4] To achieve this, the following functions are used: modulus() For additional mathematical functions, see Numbers.

Additional information

Parameters used in this function

Parameter	Input (data type)	Description
numberList	NUMBER LIST	Any given number list.
logicalExpression	TEXT	A logical expression that returns true or false . ^ is used for referencing the field codes of the seed issue.

Output

This function returns a NUMBER LIST

Text list

Variant for **text lists**. The current list value is referenced by **^%**.

Syntax

```
filterByPredicate(textList, logicalExpression) #Output: Text list
```

Examples

Parser expression	Description
<pre>filterByPredicate(["book", "rose", "sword"], length(^%) > 4)</pre>	This example returns a text list with words that have more than 4 characters : ["sword"]
<pre>filterByPredicate(["book", "rose", "sword"], ^% in % {issue.summary} OR ^% in %{issue.description})</pre>	This example returns a text list with those words that also appear in the issue's summary or description .

Additional information

Parameters used in this function

Parameter	Input (data type)	Description
textList	TEXT LIST	Any given text list.
logicalExpression	TEXT	A logical expression that returns true or false . ^% is used for referencing the field codes of the seed issue.

Output

This function returns a TEXT LIST

Issue list

Variant for **issue lists**.

Syntax

```
filterByPredicate(issueList, logicalExpression) #Output: Issue list
```

Examples

Parser expression	Description
<pre>filterByPredicate(linkedIssues(), ^{issue.summary} ~ % {issue.summary})</pre>	This example returns an issue list with linked issues (see linkedIssues()) that have the same summary like the current issue .
<pre>filterByPredicate(issuesUnderEpic(), ^{issue.assignee} = null)</pre>	This example returns an issue list with all unassigned issues under the current epic (see issuesUnderEpic()).
<pre>filterByPredicate(linkedIssues("blocks"), ^{issue. resolution} = null AND ^{issue.priority} < {issue. priority})</pre>	This example returns an issue list with unresolved blocked issues with a higher priority than the current issue.

Additional information

Parameters used in this function

Parameter	Input (data type)	Description
issueList	ISSUE LIST	Any given issue list. Usually this value is retrieved from a function (e.g. linkedIssues() or subtasks()).
blogicalExpression	TEXT	A logical expression that returns true or false . ^% is used for referencing the field codes of the seed issue.

Output

This function returns an

This is one of the **most powerful** functions in JWT since it combines **filtering** with **boolean** or **logical expressions**.

To freshen up your knowledge or to get some inspiration head over to:

- [Logical mode](#)



Use cases and examples

Use case

JWT
feature

Workflow
function

Field
type

Automated
action

Parser functions

Block a transition until all sub-tasks have certain fields populated	 	Logical validator Logical condition	count() filterByPredicate() subtasks()
Add watchers ignoring inactive users		Update or copy field values	usersInGroup() isActive() toString() filterByPredicate()
Match several values of a list		Update or copy field values	toString() distinct() filterByPredicate()
Number of open subtasks		Number	count() filterByPredicate() subtasks()
Block an Epic's transition depending on linked issues status and due date	 	Logical validator Logical condition	count() filterByPredicate() linkedIssues()