

append()

Supported list types

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

1

2

3

Number list

This function combines two **number lists**.

The order of elements is respected and the result may contain duplicates.

If you want to avoid duplicates, use the function [union\(\)](#).

Syntax

```
append(numberList1, numberList2) #Output: Number list
```

Examples

Parser expression	Description
<pre>append([1, 2, 3], [3, 4, 5])</pre>	This example returns [1, 2, 3, 3, 4, 5]
<pre>append(fieldValue({issue.timeSpent}, linkedIssues("is blocked by")), fieldValue({issue.timeSpent}, subtasks()))</pre>	This example returns a number list with Total time Spent (in minutes) in blocking issues and sub-tasks. To achieve this, the following functions are used: • fieldValue() • linkedIssues() • subtasks() This number list can be summed using function sum().

Additional information

Parameters used in this function

Parameter	Input (data type)	Description
numberList1	NUMBER LIST	Any given number list.
numberList2	NUMBER LIST	Any given number list.

Output

This function returns a `NUMBER LIST`.

If both lists are empty, the function returns an empty `NUMBER LIST`.

Text list

Variant for **text lists**.

Syntax

```
append(textList1, textList2) #Output: Text list
```

Examples

Parser expression	Description
<pre>append(["blue", "red", "green"], ["red", "green", "yellow"])</pre>	This example returns ["blue", "red", "green", "red", "green", "yellow"]
<pre>append(fieldValue(%{issue.fixVersion}, subtasks()), fieldValue(% {issue.fixVersion}, linkedIssues("is blocked by")))</pre>	This example returns a text list with Fix Version/s of all sub-tasks and blocking issues. To achieve this, the following functions are used: • <code>fieldValue()</code> • <code>subtasks()</code> • <code>linkedIssues()</code>

Additional information

Parameters used in this function

Parameter	Input (data type)	Description
textList1	<code>TEXT LIST</code>	Any given text list.
textList2	<code>TEXT LIST</code>	Any given text list.

Output

This function returns a `TEXT LIST`.

If both lists are empty, the function returns an empty `TEXT LIST`.

Issue list

Variant for **issue lists**.

syntax

```
append(issueList1, issueList2) #Output: Issue list
```

Examples

Parser expression	Description
<pre>append(linkedIssues("is blocked by"), subtasks())</pre>	This example returns an issue list of blocking issues and sub-tasks. If a sub-task is also linked with issue link type "is blocked by", it will appear twice in the output list. To achieve this, the following functions are used: • linkedIssues() • subtasks()

Additional information

Parameters used in this function

Parameter	Input (data type)	Description
issueList1	ISSUE LIST	Any given issue list. Usually this value is retrieved from a function (e.g. linkedIssues() or subtasks()).
issueList2	ISSUE LIST	Any given issue list. Usually this value is retrieved from a function (e.g. linkedIssues() or subtasks()).

Output

This function returns an .

If both lists are empty, the function returns an empty .

This function is the **equivalent** to the list operator [APPEND](#) and can be used interchangeably.



Use cases and examples

Use case	JWT feature	Workflow function	Field type	Automated action	Parser functions
Add components of the epic to the current issue				Update field action	