

sort()

Supported list types

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

1

2

3

Number list

This function sorts a given **number list** in a specified order. Available orders are **ASC** (for ascending order) and **DESC** (for descending order).

Syntax

```
sort(numberList, order) #Output: Number list
```

Examples

Parser expression	Description
<pre>%{sort([2, 4, 3, 5, 1], ASC)}</pre>	This example returns [1, 2, 3, 4, 5]
<pre>%{sort(toNumberList(%{issue.description}), DESC)}</pre>	This example returns a number list in the description sorted by descending order: [4, 3, 2, 1, -6] To achieve this, the following functions are used: <ul style="list-style-type: none">• toNumberList()

Additional information

Parameters used in this function

Parameter	Input (data type)	Description
numberList	<div>NUMBER LIST</div>	Any given number list.
order	<div>TEXT</div>	Available orders are ASC (for ascending order) and DESC (for descending order).

Output

This function returns a

NUMBER LIST

1

2

3

Text list

Variant for **text lists**.

Syntax

```
sort(textList, order) #Output: Text list
```

Examples

Parser expression	Description
<pre>%{sort(["red", "blue", "green"], ASC)}</pre>	This example returns ["blue", "green", "red"]
<pre>%{sort(toStringList(%{issue. components}), DESC)}</pre>	This example returns a text list of the components of the current issue sorted in descending order, e.g. ["UX", "UI", "Cloud"] To achieve this, the following functions are used: <ul style="list-style-type: none">• toStringList()

Additional information

Parameters used in this function

Parameter	Input (data type)	Description
textList	<input type="text" value="TEXT LIST"/>	Any given text list.
order	<input type="text" value="TEXT"/>	Available orders are ASC (for ascending order) and DESC (for descending order).

Output

This function returns a



Issue list

Variant for **issue lists**.

The list is sorted by **Issue key**. Available orders are **ASC** (for ascending order) and **DESC** (for descending order).

Syntax

```
sort(issueList, order) #Output: Issue list
```

Examples

Parser expression	Description
<pre>sort(linkedIssues("is blocked by"), ASC)</pre>	<p>This example returns an issue list of issues blocking current issue, sorted in ascending order by their Issue key.</p> <p>To achieve this, the following functions are used:</p> <ul style="list-style-type: none"> • linkedIssues()
<pre>sort(subtasks(), ASC)</pre>	<p>This example returns an issue list of sub-tasks, sorted in ascending order by their key.</p> <p>To achieve this, the following functions are used:</p> <ul style="list-style-type: none"> • subtasks()

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()).
order	TEXT	Available orders are ASC (for ascending order) and DESC (for descending order).

Output

The function returns an ISSUE LIST



Use cases and examples

Use case

No content found.