

toStringList()

This function **converts** a comma-separated **text** into a **text list**.

A common use case is a **text list** representing each of the selected elements of a selectable field like an issue picker.

If you need the actual text instead of a list, wrap your expression in a [toString\(\)](#) function.

Syntax

```
toStringList(text) #Output: Text list
```

Examples

Parser expression	Description
<code>%{toStringList(%{issue.components})}</code>	This example returns a text list with each of the components selected in the current issue , e.g.: ui,portal
<code>%{toString(toStringList(%{issue.labels}))}</code>	This example returns a text with each of the labels selected in the current issue , e.g.: web, customer, mobile To achieve this, the following functions are used: <ul style="list-style-type: none">• toString()

Additional information

Parameters used in this function

Parameter	Input (data type)	Description
text	<div>TEXT</div>	Any comma-separated text satisfying the format: "Element1, Element2". This function is often used in combination with selectable fields like an issue picker.

Output

The function returns a

TEXT LIST

Variant where you can additionally define custom **separators**.

Leading and trailing spaces around each list element are automatically removed.

Syntax

```
toStringList(text, separators) #Output: Text list
```

Example

Parser expression	Description
<pre>%{toStringList("red, orange, yellow? green; blue; purple", ",?;")}</pre>	This example returns the following text list: ["red", "orange", "yellow", "green", "blue", "purple"]

Additional information

Parameters used in this function

Parameter	Input (data type)	Description
text	<input type="text" value="TEXT"/>	Any text with elements separated by one of the separators defined in the second argument.
separators	<input type="text" value="TEXT"/>	Any separator characters.

Output

This function returns a



Use cases and examples

Use case	JWT feature	Workflow function	Parser functions	Label
Add watchers from another field		Update fields Transition issue	union() toStringList()	
Create a sub-task for each component		Create issue	toStringList() getMatchingValue()	