

Create several issues combining fields

On this page

[Prerequisites](#) | [Configuration](#) | [Related use cases](#)

Use case

This use case is useful for giving instructions on how create several issues from the combination of two fields with different options. Every newly created issue will contain a different combination of these two fields and there will be a issue for each possible combination.



Prerequisites

Set up custom fields

1

Create a **Select List (single choice)** custom field

Name: Countries

Options:

- Germany
- Spain
- US

2

Create another **Select List (single choice)** custom field

Name: Approaches

Options:

- Approach A
- Approach B



Configuration

1

Add a [Create issue](#) post function to the **workflow transition** of your choice.

2

Issues to be created

1

Expression

Write an expression using this format with parsing mode in **Number**.

```
count(toStringList(%{issue.cfCountry})) * count(toStringList(%{issue.cfApproaches}))
```

2

Issue type

Choose **Selected issue type**.

3

Specific issue type

Choose **Sub-task**.

4

Parent issue

Choose **Current issue**.

3

Summary

Enter the following expression

```
"Country: " + nthElement(toStringList(%{issue.cfCountry}),ceil(^ / count(toStringList(%{issue.cfApproaches})))) + " / Zusatzinfo: " + nthElement(toStringList(%{issue.cfApproaches}),  
(modulus(^,count(toStringList(%{issue.cfApproaches}))) = 0 ? count(toStringList(%{issue.cfApproaches})) : modulus(^,count(toStringList(%{issue.cfApproaches}))))
```

with parsing mode set to **Advanced text**.

4

Country

Enter the following expression

```
nthElement(toStringList(%{issue.cfCountry}),ceil(^ / count(toStringList(%{issue.cfApproaches}))))
```

with parsing mode set to **Advanced text**.

5

Approaches

Enter the following expression

```
nthElement(toStringList(%{issue.cfApproaches}),(modulus(^,count(toStringList(%{issue.cfApproaches}))) = 0 ? count(toStringList(%{issue.cfApproaches})) : modulus(^,count  
(toStringList(%{issue.cfApproaches}))))
```

with parsing mode set to **Advanced text**.



Related use cases

Use case	JWT feature	Workflow function	Parser functions	Label
Clone epic, tasks and sub-tasks		Create issue	toString() toStringList() issueKeysToIssueList() replaceFirst() first() fieldValue() textOnStringList() findModify() subtasks() issuesUnderEpic()	
Create an issue in the current project		Create issue		STAFF PICK
Create an issue with a custom summary		Create issue	previousValue()	
Create a sub-task for each component		Create issue	toStringList()	
Create a sub-task for every recurring deadline within a task		Create issue	addMonths() getMatchingValue() dateToString()	
Create a sub-task for every sub-task closed		Create issue	findReplaceFirst() first() findPattern() toString() sum() toNumber()	
Create a sub-task for every user in a field and assign them		Create issue	getMatchingValue() toStringList()	
Create child issue within Advanced Roadmaps hierarchy		Create issue		
Create issues randomly		Create issue	modulus() round() random()	
Create issue under epic		Create issue	toString() epic()	
Create several issues combining fields		Create issue	nthElement() toStringList() modulus() count() ceil()	
Create several sub-tasks depending on the component		Create issue		
Create sub-tasks depending on selected values in custom field		Create issue	toStringList()	STAFF PICK
Create three issues with individual summaries		Create issue	getMatchingValue()	

If you still have questions, feel free to refer to our [support team](#).