

Prevent issue creation with the same field value (Boolean)

On this page

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

Use case

Prevent issue creation when there is already another issue in the same project with the same value in a certain field.

In this use case let's suppose that we have a text custom field called **Invoice**, and we want to avoid this field's value to be repeated among issues of the same issue type in the same project.

This use case is based on the [Prevent issue creation with the same field value](#) with [Validation based on JQL query](#).



Configuration

1

Add the [Logical validator](#) to the "Create issue" transition of the desired workflow.

2

Expression

Add the following expression:

```
count(issuesFromJQL("project = '" + {%issue.project.key} + "' AND issuetype = '" + {%issue.issueType} + "' AND Invoice = '" + {%issue.cfnnnnn} + "'")) = 0
```

{%issue.cfnnnnn} is the field code for the custom field **Invoice**. You might need to update the expression accordingly to the custom field you want to use.



Screenshots

Update parameters of the 'Logical validator' Validator for this transition.

Update parameters of the 'Logical validator' Validator for this transition.

Basic configuration*

Define the required parameters for this validator.

Expression*

✓

1

count(issuesFromJQL("project = '" + \${issue.project.key} + "' AND issuetype = '" + \${issue.issueType} + "' AND Invoice = '" + \${issue.cf13600} + "'")) = 0

The expression must return true or false. The condition is met if the logical expression returns true.

- Validation options
- Error message

Logical validator

Documentation

Evaluate a logical expression that returns true or false - a most universal and powerful function to compare field values across any kind of issue relation, count sub-tasks or links.

Workflow

JWT 3.0.0 Use cases

TO DO

JWT (1471)

IN PROGRESS

Related use cases

Use case	JWT feature	Workflow function	Parser functions	Label
Assignee may only have a restricted number of assigned issues		Condition based on JQL query		
		Validation based on JQL query		
Block an Epic's transition depending on linked issues status and due date		Logical validator	count()	
		Logical condition	filterByPredicate() linkedIssues()	
Block an Epic's transition until all the issues under that Epic are resolved		Logical validator	count()	
		Logical condition	issuesUnderEpic() filterByResolution()	
Block a transition until all sub-tasks have certain fields populated		Logical validator	count()	STAFF PICK
		Logical condition	filterByPredicate() subtasks()	
Ensure that all issues linked with a certain issue link type have "Due Date" field set		Logical validator	count()	
		Logical condition	linkedIssues() fieldValue()	
Ensure that an issue has at least one attachment		Logical validator		STAFF PICK
		Logical condition		

Evaluate Assets objects		Logical validator Logical condition	findPattern() findReplaceAll() replaceAll() toStringList()	
Halt a transition if an element is not contained in a list		Logical validator Logical condition	toStringList()	
Make "Time spent" field required		Logical validator		
Prevent a closed issue from being reopened after resting 1 week closed		Logical validator Logical condition	isInRole()	
Prevent a transition depending of the number of components		Logical validator Logical condition	numberOfSelectedItems()	
Prevent creation of issues with a duplicate summary		Validation based on JQL query		
Prevent external users from creating issues		Logical validator	matches() toString() textOnStringList() toStringList() userDisplayName()	
Prevent issue creation if description contains less than 100 characters		Logical validator	replaceAll() length()	
Prevent issue creation with the same field value		Validation based on JQL query Logical validator		STAFF PICK
Prevent issue creation with the same field value (Boolean)		Logical validator Validation based on JQL query	count() issuesFromJQL()	
Prevent setting due dates outside business hours		Logical validator Logical condition	withinCalendar()	STAFF PICK
Prevent the creation of sub-tasks unless the parent issue is in a certain status		Logical validator		
Reject duplicated file names in attachments		Logical validator Logical condition	count() toStringList() distinct()	
Restrict issue creation per issue type and project role		Logical validator	isInRole()	
Validate a Select List (cascading) custom field		Logical validator Logical condition		

Validate only issue links created in transition screen		Logical validator	count() transitionLinkedIssues() filterByProject() filterByIssueType()
Validation based on the value of a date type project property		Logical validator Logical condition	stringToDate() projectProperty()

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