

Set or create user property

This function has been **renamed** with the **JWT 3.0** release.

Find the new documentation at:

[Set or create JWT user property](#)

On this page

- [Purpose](#)
- [Example: Set user property to current issue's key](#)
- [Usage Examples](#)
- [Related Features](#)

Purpose

This post-function allows to set the value of a **user property**, or create it if it doesn't exist.

Example: Set user property to current issue's key

In the following screenshot we show how to set current user's property called **"LastTransitionedIssue"** with the issue key of current issue:

Name of the user property: Name of the property that will be written, or created if it doesn't exist yet for selected user.	<input type="text" value="LastTransitionedIssue"/> Field code injector: <div>Summary - [Text] - %{00000}</div> <div>Field codes with format %{nnnnn} can be inserted in the field, and they will be replaced with their corresponding field value at runtime.</div>
Value to be written: Value to be written into the user property.	<div>1 %{00015}</div> Field code injector: <div>Summary - [Text] - %{00000}</div> <div>Field codes with format %{nnnnn} can be inserted in the field, and they will be replaced with their corresponding field value at runtime.</div>
Property owner: User whose property will be written, or created if it doesn't exist.	<div>Assignee - [User]</div>
Conditional execution: Optional boolean expression that should be satisfied in order to actually execute the post-function. (Syntax Specification)	<div>1</div> <div>Leave the field empty for executing the post-function unconditionally. Collection of Examples [Line 1 / Col 1]</div> <div>Logical connectives: and, or and not. Alternatively you can also use &, and !. Comparison operators: =, !=, >, >=, < and <=. Operators in, not in, any in, none in, ~ and !~ can be used with <i>strings, multi-valued fields and lists</i>. Logical literals: true and false. Literal null is used with = and != to check whether a field is initialized, e.g. {00012} != null checks whether <i>Due Date</i> is initialized.</div> <div>String Field Code Injector: <div>Summary - [Text] - %{00000}</div>Numeric/Date Field Code Injector: <div>Original estimate (minutes) - [Number] - {00068}</div></div> <div>Check Syntax</div>

Once configured, post-function looks like this:

Validators 1	Post Functions 6
The following will be processed after the transition occurs Add post function	
1. Set/create user property LastTransitionedIssue to user in field Assignee with parsed text in field %{Issue key} .	

Usage Examples

Page: [Assign new issues to a different project role depending on field value in current issue](#)

Page: [Limit the number of hours a user can log per day](#)

Related Features

- [Read user property](#)