

Mathematical and date-time expression calculator

This function has been **renamed** with the [JWT 3.0](#) release.

Find the new documentation at:

[Mathematical or date-time expression calculator](#)

On this page

- [Purpose](#)
- [Example 1: Set field using fields on linked issues and sub-tasks](#)
- [Example 2: Date-Time Calculation](#)
- [Usage Examples](#)
- [Related Features](#)

Purpose

This post-function allows to do almost any math and time calculation you can imagine using any system or custom field values, in current issue or any other related issue (linked issue, sub-tasks, JQL selected, etc). This feature uses the [Expression Parser](#).

Example 1: Set field using fields on linked issues and sub-tasks

In this example we set custom field "**Story Points**" in our **Epic** with the sum of values of custom field "**Story Points**" in **related stories** and **sub-tasks**:

Target field:

Story Points - [Number Field]

Field to be written with the result of evaluation of the formula.
☐ Don't overwrite target field if it's already set.

Formula:

[Line 1 / Col 92][Syntax Specification](#)[Check Syntax](#)

1
sum(fieldValue({10025}, subtasks())) + sum(fieldValue({10025}, linkedIssues("is Epic of")))

NUMERICAL AND DATE-TIME TERMS

Numeric and Date-Time field values: insert field codes with format {nnnnn}.

Original estimate (minutes) - [Number] - {00068}
Insert Numeric Value

Valid date-time literal formats: yyyy/MM/dd [hh:mm] or yyyy-MM-dd [hh:mm]. Time literals use format: hh:mm.
There is a set of [mathematical functions](#) and [time macros and functions](#) available to be used in your expression.

TEXT-STRING TERMS

Text-String field values: insert field codes with format %{nnnnn} or %{nnnnn.i} for referencing levels in cascading select fields (i = 0 for base level).

Summary - [Text] - %{00000}
Insert String Value

String literals: written in **double quotes**, e.g., "This is a string literal."
String concatenation: use operator '+' to concatenate string values, e.g., "The summary of issue with key " + %{00015} + " is \" + %{00000} + "\"."
Escape character: character \" is used with characters \"\", \"\", \"\", \"\", \"f\" and \"b\" to invoke an alternative interpretation.
There is a set of [string functions](#) available to be used in your expression.

Additional options:

☐ Update issue immediately after field writing. A specific entry will be created in issue history for this field writing.

Conditional execution:

Optional boolean expression that should be satisfied in order to actually execute the post-function.
[\(Syntax Specification\)](#)

1

Leave the field empty for executing the post-function unconditionally.
[Collection of Examples](#)
[Line 1 / Col 1]

[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 strings, multi-valued fields and lists.
[Logical literals:](#) true and false. Literal null is used with = and != to check whether a field is initialized, e.g. {00012} != null checks whether Due Date is initialized.

[String Field Code Injector:](#)

Summary - [Text] - %{00000}

[Numeric/Date Field Code Injector:](#)

Original estimate (minutes) - [Number] - {00068}

[Check Syntax](#)

Note that:

- {10025} is code numeric value of "Story Points" custom field in the particular Jira instance of this example

Once configured, post-function looks like this:

Validators 1

Post Functions 5

The following will be processed after the transition occurs

Add post function

1. Field **Story Points** will be assigned with the calculation result of the formula:
`sum(fieldValue({Story Points}, subtasks())) + sum(fieldValue({Story Points}, linkedIssues("is Epic of")))`
This feature will be run as user in field **Current user**.

Example 2: Date-Time Calculation

Time calculations are made based on the fact that **Date** and **Date-Time** system or custom fields return the number of milliseconds elapsed since January 1, 1970, 00:00:00 GMT.

Let's see how to calculate the number of whole days elapsed since issue creation:

Target field:

Days since issue creation - [Number Field]

Field to be written with the result of evaluation of the formula.
☐ Don't overwrite target field if it's already set.

Formula:

[Line 1 / Col 35] [Syntax Specification](#) [Check Syntax](#)

1 floor(({00057} - {00009}) / {DAY})

NUMERICAL AND DATE-TIME TERMS

Numeric and Date-Time field values: insert field codes with format {nnnnn}.

Original estimate (minutes) - [Number] - {00068}
Insert Numeric Value

Valid date-time literal formats: yyyy/MM/dd [hh:mm] or yyyy-MM-dd [hh:mm]. Time literals use format: hh:mm.
There is a set of [mathematical functions](#) and [time macros and functions](#) available to be used in your expression.

TEXT-STRING TERMS

Text-String field values: insert field codes with format %{nnnnn} or %{nnnnn.i} for referencing levels in cascading select fields (i = 0 for base level).

Summary - [Text] - %{00000}
Insert String Value

String literals: written in **double quotes**, e.g., "this is a string literal."
String concatenation: use operator '+' to concatenate string values, e.g., "The summary of issue with key " + %{00015} + " is \" + % {00000} + "\"."
Escape character: character '\ ' is used with characters '"', '\', 'n', 'r', 't', 'f' and 'b' to invoke an alternative interpretation.
There is a set of [string functions](#) available to be used in your expression.

Additional options:

☐ Update issue immediately after field writing. A specific entry will be created in issue history for this field writing.

Conditional execution:

Optional boolean expression that should be satisfied in order to actually execute the post-function.
[\(Syntax Specification\)](#)

1

Leave the field empty for executing the post-function unconditionally.
[Collection of Examples](#)
[Line 1 / Col 1]

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 strings, multi-valued fields and lists.

Logical literals: true and false. Literal null is used with = and != to check whether a field is initialized, e.g. {00012} != null checks whether Due Date is initialized.

String Field Code Injector:

Numeric/Date Field Code Injector:

Summary - [Text] - %{00000}
Original estimate (minutes) - [Number] - {00068}

[Check Syntax](#)

Note that:

- {00057} is code for numeric value of "Current date and time"
- {00009} if code for numeric value of "Date and time of creation"

Once configured, post-function looks like this:

Validators 1	Post Functions 5
--------------	------------------

The following will be processed after the transition occurs

Add post function

1. Field **Days since issue creation** will be assigned with the calculation result of the formula:

```
floor(({Current date and time} - {Date and time of creation}) / {DAY})
```

This feature will be run as user in field **Current user**.

Examples of Math and Time Expressions

Usage Examples

Page: Automatic work log with start and stop work transitions

Page: Automatically log work time when the user uses a "Stop Progress" transition

Page: Calculate the time elapsed between 2 transition executions

Page: Getting the number of selected values in a custom field of type Multi Select

Page: Implement a form with a series of questions and calculate a numeric value based on the answers

Page: Increment a field or set to 1 if it's not set

Page: Set "Date-Time Picker" custom field with current date-time

Page: Set "Due date" 6 natural days (or work days) earlier than a "Date Picker" custom field

Page: Set "Due date" to a specific day of next week no matter of date of creation this week

Page: Set "Due date" with certain time offset from current date

Page: Set "Total time spent" to "Current date and time - date and time of last update"

Page: Set a custom field "Urgency" depending on a combined value of issue's priority and "Impact" custom field

Page: Sum "Time Spent" in all sub-tasks of issues linked with issue link types "LinkA", "LinkB", "LinkC"

Page: Triage Jira Service Desk email requests (Move issues)

Page: Using project properties to calculate custom sequence numbers

Related Features