

Compare two values condition

This condition evaluates the **comparison of two (field) values**.



Configuration

Field

Define the **first** of the two values that should be compared by selecting it from the list of available fields.

Depending on the type of the field you have selected only certain operators and fields in the comparison value field selector are available.

Operator

Select how to compare the two given values. Choose from one of the following options:

Option	Description
=	The first value must be equal to the second value.
!=	The first value must not be equal to the second value.
<	The first value must be less than the second value. This is typically used for comparing numeric or date-time values.
>	The first value must be greater than the second value. This is typically used for comparing numeric or date-time values.
<=	The first value must be less than or equal to the second value. This is typically used for comparing numeric or date-time values.
>=	The first value must be greater than or equal to the second value. This is typically used for comparing numeric or date-time values.
~	The first value must contain the second value. This is typically used for lists as field value to compare. The comparison value is a text or number .
!~	The first value must not contain the second value. This is typically used for lists as field value to compare. The comparison value is a text or number .

In case you have selected a field of the type or , only the **equal, not equal, contains** and **not contains** operators are available. Lists may contain duplicate values.
For fields of type , only the **equal** and **not equal** operators are available.

Comparison value

Define the **second** of the two values that should be compared either by selecting it from the list of available fields or by entering a literal value. You can switch between the two option by using the radio button which offers the options

- "Select the field to compare the value with." This is the preselected default value.
- "Enter the custom comparison value." In this case a text editor opens where you can enter the comparison value manually. Depending on the data type of the first comparison value, the following inputs are valid as custom value

Option	Description
Date	A date in the format YYYY-MM-DD, e.g. 2008-09-15
Date time	A date time value in the format YYYY-MM-DDTHH:mm:ss, e.g. 2008-09-15T15:53:00+05:00
Text	Any text, e.g. This is a text

Number	Any number, e.g. 42 or 7.75
List	List are entered as their comma separated values: This, is, a, list or 2,3,5,7
Boolean	One of the boolean values true or false
Select list (cascading)	The parent and child values are noted as text, separated by a comma, e.g. Parent, child . If the child is omitted (e.g. Parent) then the child value of the first operator has to be null as well.

The most common use case is to compare a field values provided by **field codes**.

If one of the selected fields is **hidden** or **deleted** when the condition is executed, the field value is interpreted as **null**.

If you need to compare multiple values or construct more complex comparisons make sure to have a look at the [Jira expression condition](#).

Use cases and examples

Use case	JWT feature	Workflow function	Use case description	Complexity
Due date must be in the future		Compare two values condition Compare two values validator	<p>The due date must be in the future.</p> <p>This use case is valid for both conditions and validators. The only difference is that you can specify an additional error message when using a validator.</p>	BEGINNER
Current user must be reporter		Compare two values condition Compare two values validator	<p>Current user must be reporter.</p> <p>This use case is valid for both conditions and validators. The only difference is that you can specify an additional error message when using a validator.</p>	BEGINNER
Issue must have at least two attachments		Compare two values condition Compare two values validator	<p>Require a at least two attachments.</p> <p>This use case is valid for both conditions and validators. The only difference is that you can specify an additional error message when using a validator.</p>	BEGINNER

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