Texts

JWT is a text specialist!

Texts can be provided as **fixed values** - or **literals** - in the format described in the table below or they can be retrieved by using **field codes.**

Fixed values

Texts can be added to any expression as fixed values. How texts are being interpreted depend on the Parsing modes. Usually texts are enclosed by quotes **"".**

Input	Description	Example
Valid **text** values	A simple text can be added enclosed by quotes unless you are using the Basic text mode. In this case you **do not** need quotes.	`"This is a text."`

Concatenated text values	You can concatenate texts with the **+** operator. This is useful when text fragments are separated by field codes or parser functions.	`"This is" + " a text." #the result is the same as above "This is a text."`
		
		`"This must be done by" + %{issue.dueDate} + ". Thank you!"`

**Escape character **	This character can precede any of the following characters: ", \, n, r, t, f and b in order to invoke an alternative interpretation.	`"The man said: \"Hello!\"." #the result would be The man said: "Hello!"`

Variable values (field values)

Text values of ⬛ TEXT data type fields can be inserted in any expression as text field codes using the following notation `%{...somefield}` - e.g `%{issue.summary}`.

Any **field** type or data type can be transformed to **text**, so **any field** can be referenced as a **text-string value** using the following notation: `%{issue.somefield}`.

If a field has **no** value (`null`), an **empty text** will be returned.

Cascading Select fields are treated as text fields, where **i** is the index that represents the level to be accessed. (**i = 0** is used for base level) are notated as `%{issue.somefield.i}`

Available functions

Function	Short description	Output
capitalizeWords()	Capitalizes a text, only updating the **first letters**.	TEXT
capitalizeWordsFully()	**Capitalizes** an **entire** text.	TEXT
escapeHTML()	Replaces **special characters** with **HTML** entities.	TEXT
findModify()	Replaces specific **substrings** matching a given **regular expression**.	TEXT
findPattern()	Returns all **substrings** matching a given **regular expression**.	TEXT LIST
findPatternIgnoreCase()	Returns all **substrings** matching a given **regular expression**, **ignoring the case**.	TEXT LIST
findReplaceAll()	**Replaces all occurrences** of a given substring with the given replacement.	TEXT
findReplaceAllIgnoreCase ()	**Replaces all occurrences** of a given substring with a given replacement, i**gnoring the case**.	TEXT
findReplaceFirst()	**Replaces** the **first occurrence** of a given substring with the given replacement.	TEXT
findReplaceFirstIgnoreCase()	**Replaces** the **first occurrence** of a given substring with the given replacement, **ignoring the case**.	TEXT
getFromJSON()	Returns the result of a given JMESPath which is applied to a JSON	TEXT
htmlToTxt()	Removes all **HTML tags**.	TEXT
issueSecurityLevel()	Returns the **name** of the **issue security level** with a given **ID**.	TEXT
issueType()	Returns the **name** of the **issue type** with a given **ID**	TEXT
length()	Returns the **length** any given text.	NUMBER
matches()	Checks, if the given **text matches** the provided **regular expression**.	BOOLEAN
option()	Returns the **name** of a custom field option.	TEXT
priority()	Returns the **name** of the **priority** with a given **ID.**	TEXT
project()	Returns the **name** of the **project** with a given **ID.**	TEXT
replaceAll()	Replaces all substrings matching a **regular expression** with a given **replacement**.	TEXT
replaceFirst()	Replaces the **first substring** matching a given **regular expression** with a given replacement.	TEXT
resolution()	Returns the **name** of the **resolution** with a given **ID**.	TEXT
similarity()	Calculates the **similarity** between two texts.	NUMBER
status()	Returns a **status name**.	TEXT
substring()	Returns a **specific part** of a text.	TEXT
toLowerCase()	Converts a given **text** with all its characters **lower case**.	TEXT

toUpperCase()	Converts a given **text** with all characters to **upper case**.	TEXT
trim()	Removes **leading and trailing blanks** (white spaces and tabs) from a text.	TEXT
unescapeHTML()	Replaces **HTML escapes** with their corresponding **Unicode** characters.	TEXT
wikiToHTML()	Converts rich text **wiki** content to **HTML**.	TEXT

If you still have questions, feel free to refer to our support team.