

findPattern()

This function returns a **text list** with all **substrings** matching a given regular expression.

Syntax

findPattern(text, regularExpression) #Output: Text list

Examples

Parser expression	Description
<pre>findPattern("Between 1900 and 2000 world population increase from 1.5 to 6.1 billions.", "\d+(\.\d+)?")</pre>	<p>This example returns the following text list:</p> <p>["1900", "2000", "1.5", "6.1"]</p>
<pre>findPattern(%{issue.somefield}, "(?<=CN=).*?(?=?,OU=Users)")</pre>	<p>Assuming %{issue.somefield} is a multi-line text field containing a list of Active Directory user entries, e.g.</p> <pre>CN=User1,OU=Users,DC=example,DC=com CN=User2,OU=Users,DC=example,DC=com CN=User3,OU=Users,DC=example,DC=com CN=User4,OU=Users,DC=example,DC=com</pre> <p>this example returns the following text list:</p> <p>["User1", "User2", "User3", "User4"]</p>
<p>Extract all email addresses from a text field</p> <pre>findPattern(%{issue.somefield}, "[a-zA-Z0-9._-]+@[a-zA-Z0-9._-]+\.\[a-zA-Z0-9._-]+\]")</pre>	<p>Assuming %{issue.somefield} is a multi-line text field containing email addresses, e.g.</p> <pre>Hello, please use admin@example.com instead of administrator@example.com for future reference. Kind regards, jwt@example.com</pre> <p>this example returns the following text list:</p> <p>["admin@example.com", "administrator@example.com", "jwt@example.com"]</p>
<p>Processing an Elements Connect (formerly nFeed)* custom field</p> <pre>findPattern(%{issue.somefield}, "(?<=:\\[\\\" \\\",\\").*?(?=\\\" \\\")")</pre> <p><small>* Applicable for Elements Connect version 6 and above.</small></p>	<p>Assuming %{issue.somefield} is an Elements Connect (formerly nFeed) custom field containing the keys of the selected options, e.g.</p> <pre>{ "keys": ["KEY-1", "KEY-2", "KEY-3"] }</pre> <p>this example returns the following text list:</p> <p>["KEY-1", "KEY-2", "KEY-3"]</p>



Use cases and examples

Use case	JWT feature	Workflow function	Field type	Automated action	Parser functions
Display reporters of linked Cloud issues			Text		<code>executeRemoteAction()</code> <code>findPattern()</code> <code>toStringList()</code> <code>toString()</code> <code>distinct()</code>
Obtain a value from an Elements Connect custom field		Send email			<code>findPattern()</code> <code>toString()</code>
Evaluate Assets objects	 	Logical validator Logical condition			<code>findPattern()</code> <code>findReplaceAll()</code> <code>replaceAll()</code> <code>toStringList()</code>
Copy excerpted value from new comments				Update field action	<code>findPattern()</code> <code>first()</code>
Send email with the URL of the attachments included in the description		Send email			<code>replaceAll()</code> <code>toString()</code> <code>attachmentUrls()</code> <code>findPattern()</code>
Link issue to issue keys in its description		Create issue link			<code>findPattern()</code> <code>toString()</code>
Create a sub-task for every sub-task closed		Create issue			<code>findReplaceFirst()</code> <code>first()</code> <code>findPattern()</code> <code>toString()</code> <code>sum()</code> <code>toNumber()</code>
Track issues mentioned in comments				Create issue link action	<code>findPattern()</code> <code>issueKeysToIssueList()</code> <code>toString()</code>
Triage issues created by email		Move issue			<code>toUpperCase()</code> <code>toString()</code> <code>findPattern()</code> <code>isInGroup()</code>