



This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

# Genesys Knowledge Management User Guide

Regular Expressions

# Regular Expressions

A regular expression stands for, not one particular character string, but a class of character strings. For example, suppose that you want to find all interactions with U.S. Zip codes in them. U.S. Zip codes are five-digit numbers, so you could in theory write about 9,000 screening rules (Find("00000"), Find("00001"), Find("00002"), and so on).

Fortunately, you can use the special symbol `\d`, which stands for any digit, to write a screening rule using a regular expression: `RegExFind("\d\d\d\d\d")`. This screening rule matches any sequence of five digits.

There are often several different ways of writing the same regular expression.

For instance, two items separated by a hyphen and enclosed in square brackets denotes a range of which the two items are endpoints. So `[a-d]` matches a, b, c, or d, and `[5-8]` matches any digit between 5 and 8; hence `\d` is the same as `[0-9]`.

## Important

In general usage, apart from Genesys eServices, regular expressions are case sensitive. However, in Knowledge Manager regular expressions are not case sensitive unless you add `, false` as described above.

The table "Elements of Regular Expressions" lists some of the most commonly-used elements of regular expressions:

Elements of Regular Expressions

Symbol	Meaning	Example
.	Any character, including space	<code>b.t</code> matches <i>bat</i> , <i>bet</i> , <i>bit</i> , and <i>but</i> .
<code>\d</code>	Any digit	<code>\d\d</code> matches any pair of digits from 00 to 99.
<code>\s</code>	Space	<code>\d\s\d</code> matches 1 0, 5 9, and so on.
.	Zero or more instances of the preceding expression	<code>o*f</code> matches <i>oof</i> , <i>of</i> , and <i>f</i> . <code>me.*d</code> matches <i>med</i> , <i>mead</i> , and <i>meed</i> .
+	One or more instances of the preceding expression	<code>bre+d</code> matches <i>bred</i> , <i>breed</i> and <i>breed</i> .

Symbol	Meaning	Example
?	Zero or one instances of the preceding expression	c?rude matches <i>rude</i> and <i>crude</i> .
{x}	X instances of the preceding expression	st.{2}k matches <i>steak</i> , <i>stork</i> , and <i>stink</i> .
^	Any character except the following	s[^e]t matches <i>sat</i> , <i>sit</i> , and <i>sot</i> , but not <i>set</i> .
[ ]	Any characters or ranges within the brackets	Any characters: b[aeiou]at matches <i>boat</i> but not <i>brat</i> . Any range(s): [0-9]th matches <i>5th</i> , <i>6th</i> , <i>7th</i> [a-z] matches any lowercase letter; [A-Z] matches any uppercase letter.
\	Turns off the special meaning of the following symbol	\* matches the character * (asterisk); \. matches the character . (period or full stop).
	Or	[b p]ig matches <i>big</i> and <i>pig</i> . Do not be confused:   means <i>or</i> in regular expressions, but    means <i>or</i> as one of the <b>Operators</b> used in screening rule formulas.

Here are some other points to keep in mind:

- Space is just another character. The regular expression `savings account` contains a space, and so it does not match the string *savingsaccount*.
- Word boundaries are not considered. The regular expression `read` matches not only *read*, but also *reader*, *ready*, *spread*, *bread*, and so on.
- Use parentheses to group parts of regular expressions together. For example, `RegexFind("(\\d{3}\\.){2}")` puts `\\d{3}\\.` in parentheses so that the number-of-instances item `{2}` applies to the all of `d{3}\\.`, not just to `\\.` This expression matches any group of three digits plus period plus any three digits plus period (for example, *198.351.*). Further examples are provided in [Examples of Screening Rules](#).
- Regular expressions make use of many more special characters and operators than those listed in the table "Elements of Regular Expressions." Much documentation on regular expressions is available on the Web. Because Genesys Knowledge Management uses Java classes for regular expressions, it is best to consult documents describing the particular version of regular expressions used in Java.