

Regular Expression

Modifiers

Modifiers are used to perform case-insensitive and global searches:

Modifier	Description
g	Perform a global match (find all matches rather than stopping after the first match)
i	Perform case-insensitive matching
m	Perform multiline matching

Brackets

Brackets are used to find a range of characters:

Expression	Description
[abc]	Find any character between the brackets
[^abc]	Find any character NOT between the brackets
[0-9]	Find any character between the brackets (any digit)
[^0-9]	Find any character NOT between the brackets (any non-digit)
(x y)	Find any of the alternatives specified

Metacharacters

Metacharacters are characters with a special meaning:

Metacharacter	Description
<code>.</code>	Find a single character, except newline or line terminator
<code>\w</code>	Find a word character
<code>\W</code>	Find a non-word character
<code>\d</code>	Find a digit
<code>\D</code>	Find a non-digit character
<code>\s</code>	Find a whitespace character
<code>\S</code>	Find a non-whitespace character
<code>\b</code>	Find a match at the beginning/end of a word, beginning like this: <code>\bHI</code> , end like this: <code>HI\b</code>
<code>\B</code>	Find a match, but not at the beginning/end of a word
<code>\0</code>	Find a NULL character
<code>\n</code>	Find a new line character
<code>\f</code>	Find a form feed character
<code>\r</code>	Find a carriage return character
<code>\t</code>	Find a tab character
<code>\v</code>	Find a vertical tab character
<code>\xxx</code>	Find the character specified by an octal number xxx
<code>\xdd</code>	Find the character specified by a hexadecimal number dd
<code>\udddd</code>	Find the Unicode character specified by a hexadecimal number dddd

Quantifiers

Quantifier	Description
<code>n+</code>	Matches any string that contains at least one <i>n</i>
<code>n*</code>	Matches any string that contains zero or more occurrences of <i>n</i>
<code>n?</code>	Matches any string that contains zero or one occurrences of <i>n</i>
<code>n{X}</code>	Matches any string that contains a sequence of <i>X</i> <i>n</i> 's
<code>n{X,Y}</code>	Matches any string that contains a sequence of <i>X</i> to <i>Y</i> <i>n</i> 's
<code>n{X,}</code>	Matches any string that contains a sequence of at least <i>X</i> <i>n</i> 's
<code>n\$</code>	Matches any string with <i>n</i> at the end of it
<code>^n</code>	Matches any string with <i>n</i> at the beginning of it
<code>?=n</code>	Matches any string that is followed by a specific string <i>n</i>
<code>?!n</code>	Matches any string that is not followed by a specific string <i>n</i>

RegExp Object Properties

Property	Description
<code>constructor</code>	Returns the function that created the RegExp object's prototype
<code>global</code>	Checks whether the "g" modifier is set
<code>ignoreCase</code>	Checks whether the "i" modifier is set
<code>lastIndex</code>	Specifies the index at which to start the next match
<code>multiline</code>	Checks whether the "m" modifier is set
<code>source</code>	Returns the text of the RegExp pattern

RegExp Object Methods

Method	Description
<code>compile()</code>	Deprecated in version 1.5. Compiles a regular expression
<code>exec()</code>	Tests for a match in a string. Returns the first match
<code>test()</code>	Tests for a match in a string. Returns true or false
<code>toString()</code>	Returns the string value of the regular expression