

# 正则表达式

2018年5月5日 20:08

单字符 元字符	行为	示例
*	零次或多次匹配前面的字符或子表达式。 等效于 {0, }。	zo* 与“z”和“zoo”匹配。
+	一次或多次匹配前面的字符或子表达式。 等效于 {1, }。	zo+ 与“zo”和“zoo”匹配，但与“z”不匹配。
?	零次或一次匹配前面的字符或子表达式。 等效于 {0, 1}。 当 ? 紧随任何其他限定符 ( *, +, ?, {n}, {n,} 或 {n,m} ) 之后时，匹配模式是非贪婪的。非贪婪模式匹配搜索到的、尽可能少的字符串，而默认的贪婪模式匹配搜索到的、尽可能多的字符串。	zo? 与“z”和“zo”匹配，但与“zoo”不匹配。 o+? 只与“oooo”中的单个“o”匹配，而 o+ 与所有“o”匹配。 do(es)? 与“do”或“does”中的“do”匹配。
^	匹配搜索字符串开始的位置。如果标志中包括 m ( 多行搜索 ) 字符，^ 还将匹配 \n 或 \r 后面的位置。 如果将 ^ 用作括号表达式中的第一个字符，则会对字符集求反。	^d{3} 与搜索字符串开始处的 3 个数字匹配。 [ ^abc ] 与除 a、b 和 c 以外的任何字符匹配。
\$	匹配搜索字符串结尾的位置。如果标志中包括 m ( 多行搜索 ) 字符，^ 还将匹配 \n 或 \r 前面的位置。	\d{3}\$ 与搜索字符串结尾处的 3 个数字匹配。
.	匹配除换行符 \n 之外的任何单个字符。若要匹配包括 \n 在内的任意字符，请使用诸如 [ \s \S ] 之类的模式。	a.c 与“abc”、“a1c”和“a-c”匹配。
[]	标记括号表达式的开始和结尾。	[ 1-4 ] 与“1”、“2”、“3”或“4”匹配。[ ^aAeEiIoOuU ] 与任何非元音字符匹配。
{ }	标记限定符表达式的开始和结尾。	a{2,3} 与“aa”和“aaa”匹配。
()	标记子表达式的开始和结尾。可以保存子表达式以备将来之用。	A( \d ) 与“A0”至“A9”匹配。保存该数字以备将来之用。
	指示在两个或多个项之间进行选择。	z food 与“z”或“food”匹配。( z f )ood 与“zood”或“food”匹配。
/	表示 JScript 中的文本正则表达式模式的开始或结尾。在第二个“/”后添加单字符标志可以指定搜索行为。	/abc/gi 是与“abc”匹配的 JScript 文本正则表达式。g ( 全局 ) 标志指定查找模式的所有匹配项，i ( 忽略大小写 ) 标志使搜索不区分大小写。
\	将下一字符标记为特殊字符、文本、反向引用或八进制转义符。	\n 与换行符匹配。\\ ( 与“(”匹配。\\ 与“\”匹配。
多字符 元字符	行为	示例
\b	与一个字边界匹配；即字与空格间的位置。	er\b 与“never”中的“er”匹配，但与“verb”中的“er”不匹配。
\B	非边界字匹配。	er\B 与“verb”中的“er”匹配，但与“never”中的“er”不匹配。
\d	数字字符匹配。 等效于 [ 0-9 ]。	在搜索字符串“12 345”中，\d{2} 与“12”和“34”匹配。\\d 与“1”、“2”、“3”、“4”和“5”匹配。
\D	非数字字符匹配。 等效于 [ ^0-9 ]。	\D+ 与“abc123 def”中的“abc”和“def”匹配。
\w	与以下任意字符匹配：A-Z、a-z、0-9 和下划线。 等效于 [ A-Za-z0-9_ ]。	在搜索字符串“The quick brown fox...”中，\w+ 与“The”、“quick”、“brown”和“fox”匹配。
\W	与除 A-Z、a-z、0-9 和下划线以外的任意字符匹配。 等效于 [ ^A-Za-z0-9_ ]。	在搜索字符串“The quick brown fox...”中，\W+ 与“...”和所有空格匹配。
[xyz]	字符集。与任何一个指定字符匹配。	[ abc ] 与“plain”中的“a”匹配。
[^xyz]	反向字符集。与未指定的任何字符匹配。	[ ^abc ] 与“plain”中的“p”、“l”、“i”和“n”匹配。
[a-z]	字符范围。匹配指定范围内的任何字符。	[ a-z ] 与“a”到“z”范围内的任何小写字母字符匹配。
[^a-z]	反向字符范围。与不在指定范围内的任何字符匹配。	[ ^a-z ] 与不在范围“a”到“z”内的任何字符匹配。
{n}	正好匹配 n 次。n 是非负整数。	o{2} 与“Bob”中的“o”不匹配，但与“food”中的两个“o”匹配。
{n,}	至少匹配 n 次。n 是非负整数。 * 与 {0, } 相等。 + 与 {1, } 相等。	o{2,} 与“Bob”中的“o”不匹配，但与“fooooood”中的所有“o”匹配。
{n,m}	匹配至少 n 次，至多 m 次。n 和 m 是非负整数，其中 n <= m。逗号和数字之间不能有空格。 ? 与 {0, 1} 相等。	在搜索字符串“1234567”中，\d{1,3} 与“123”、“456”和“7”匹配。
(模式)	与模式 匹配并保存匹配项。您可以从 JScript 中的 exec Method 返回的数组元素中检索保存的匹配项。若要匹配括号字符 ( )，请使用“\ ( ”或者“\ ) ”。	( Chapter Section ) [ 1-9 ] 与“Chapter 5”匹配，保存“Chapter”以备将来之用。

(?:模式)	与模式 匹配，但不保存匹配项；即不会存储匹配项以备将来之用。这对于用“or”字符 ( ) 组合模式部件的情况很有用。	<code>industr(?:y ies)</code> 与 <code>industry industries</code> 相等。
(?=模式)	正预测先行。 找到一个匹配项后，将在匹配文本之前开始搜索下一个匹配项。 不会保存匹配项以备将来之用。	<code>^(?=.*\d){4,8}\$</code> 对密码应用以下限制：其长度必须介于 4 到 8 个字符之间，并且必须至少包含一个数字。 在该模式中， <code>.*\d</code> 查找后跟有数字的任意多个字符。对于搜索字符串“abc3qr”，这与“abc3”匹配。 从该匹配项之前（而不是之后）开始， <code>{4,8}</code> 与包含 4-8 个字符的字符串匹配。这与“abc3qr”匹配。 <code>^</code> 和 <code>\$</code> 指定搜索字符串的开始和结束位置。这将在搜索字符串包含匹配字符之外的任何字符时阻止匹配。
(?!模式)	负预测先行。 匹配与模式 不匹配的搜索字符串。 找到一个匹配项后，将在匹配文本之前开始搜索下一个匹配项。 不会保存匹配项以备将来之用。	<code>\b(?:!th)\w+\b</code> 与不以“th”开头的单词匹配。 在该模式中， <code>\b</code> 与一个字边界匹配。对于搜索字符串“quick”，这与第一个空格匹配。 <code>(?:!th)</code> 与非“th”字符串匹配。这与“qu”匹配。 从该匹配项开始， <code>\w+</code> 与一个字匹配。这与“quick”匹配。
<code>\cx</code>	匹配 x 指示的控制字符。 x 的值必须在 A-Z 或 a-z 范围内。 如果不是这样，则假定 c 就是文本“c”字符本身。	<code>\cM</code> 与 Ctrl+M 或一个回车符匹配。
<code>\xn</code>	匹配 n，此处的 n 是一个十六进制转义码。 十六进制转义码必须正好是两位数长。允许在正则表达式中使用 ASCII 代码。	<code>\x41</code> 与“A”匹配。 <code>\x041</code> 等效于后跟有“1”的“\x04”（因为 n 必须正好是两位数）。
<code>\num</code>	匹配 num，此处的 num 是一个正整数。 这是对已保存的匹配项的引用。	<code>(.)\1</code> 与两个连续的相同字符匹配。
<code>\n</code>	标识一个八进制转义码或反向引用。 如果 <code>\n</code> 前面至少有 n 个捕获子表达式，那么 n 是反向引用。 否则，如果 n 是八进制数 (0-7)，那么 n 是八进制转义码。	<code>(\d)\1</code> 与两个连续的相同数字匹配。
<code>\nm</code>	标识一个八进制转义码或反向引用。 如果 <code>\nm</code> 前面至少有 nm 个捕获子表达式，那么 nm 是反向引用。 如果 <code>\nm</code> 前面至少有 n 个捕获子表达式，则 n 是反向引用，后面跟有文本 m。 如果上述情况都不存在，当 n 和 m 是八进制数字 (0-7) 时， <code>\nm</code> 匹配八进制转义码 nm。	<code>\11</code> 与制表符匹配。
<code>\nml</code>	当 n 是八进制数字 (0-3)，m 和 l 是八进制数字 (0-7) 时，匹配八进制转义码 nml。	<code>\011</code> 与制表符匹配。
<code>\un</code>	匹配 n，其中 n 是以四位十六进制数表示的 Unicode 字符。	<code>\u00A9</code> 与版权符号 (©) 匹配。

非打印字符	描述
<code>\cx</code>	匹配由x指明的控制字符。例如， <code>\cM</code> 匹配一个 Control-M 或回车符。x 的值必须为 A-Z 或 a-z 之一。否则，将 c 视为一个原义的 'c' 字符。
<code>\f</code>	匹配一个换页符。等价于 <code>\x0c</code> 和 <code>\cL</code> 。
<code>\n</code>	匹配一个换行符。等价于 <code>\x0a</code> 和 <code>\cJ</code> 。
<code>\r</code>	匹配一个回车符。等价于 <code>\x0d</code> 和 <code>\cM</code> 。
<code>\s</code>	匹配任何空白字符，包括空格、制表符、换页符等等。等价于 <code>[\f\n\r\t\v]</code> 。
<code>\S</code>	匹配任何非空白字符。等价于 <code>^[^\f\n\r\t\v]</code> 。
<code>\t</code>	匹配一个制表符。等价于 <code>\x09</code> 和 <code>\cI</code> 。
<code>\v</code>	匹配一个垂直制表符。等价于 <code>\x0b</code> 和 <code>\cK</code> 。