**常用元字符**

代码 说明

**.** 匹配除换行符以外的任意字符

**\w**  匹配字母或数字或下划线

**\s** 匹配任意的空白符

**\d**  匹配数字

**\b** 匹配单词的开始或结束

**^** 匹配字符串的开始

**$** 匹配字符串的结束

**常用限定符**

代码/语法 说明

**\*** 重复零次或更多次

**+** 重复一次或更多次

**?** 重复零次或一次

**{n}** 重复n次

**{n,}** 重复n次或更多次

**{n,m}** 重复n到m次

**常用反义词**

代码/语法 说明

**\W**  匹配任意不是字母，数字，下划线，汉字的字符

**\S**  匹配任意不是空白符的字符

**\D** 匹配任意非数字的字符

**\B** 匹配不是单词开头或结束的位置

**[^x]** 匹配除了x以外的任意字符

**[^aeiou]** 匹配除了aeiou这几个字母以外的任意字符

**正则表达式修饰符**

修饰符 可以在全局搜索中不区分大小写:

修饰符 描述

i 执行对大小写不敏感的匹配。

g 执行全局匹配（查找所有匹配而非在找到第一个匹配后停止）。

m 执行多行匹配。更改^ 和$的含义，使它们分别在任意一行对待行首和行尾匹配，而不仅仅在整个字符串的开头和结尾匹配

**正则表达式模式**

方括号用于查找某个范围内的字符：

表达式 描述

[abc] 查找方括号之间的任何字符。

[0-9] 查找任何从 0 至 9 的数字。

(x|y) 查找任何以 | 分隔的选项。

**元字符**是拥有特殊含义的字符：

元字符 描述

\d 查找数字。

\s 查找空白字符。

\b 匹配单词边界。

\uxxxx 查找以十六进制数 xxxx 规定的 Unicode 字符。



若要匹配任意字符[\d\D]、[\w\W]、[\s\S]和[^]中任何的一个。

\b是单词边界，具体就是\w和\W之间的位置，也包括\w和^之间的位置，也包括\w和$之间的位置。

示例：

**var** result = "[JS] Lesson\_01.mp4".replace(/\b/g, '#');

console.log(result);

// => "[#JS#] #Lesson\_01#.#mp4#"

**量词:**

量词 描述

n+ 匹配任何包含至少一个 n 的字符串。

n\* 匹配任何包含零个或多个 n 的字符串。

n? 匹配任何包含零个或一个 n 的字符串。

说明示例：

"ab\*"：表示一个字符串有一个a后面跟着零个或若干个b。（"a", "ab", "abbb"）

"ab+"：表示一个字符串有一个a后面跟着至少一个b或者更多；ab，abb，abbb

"ab?"：表示一个字符串有一个a后面跟着零个或者一个b；a，ab

"a?b+$"：表示在字符串的末尾有零个或一个a跟着一个或几个b。字符末尾的ab｛1，n｝$

你也可以使用范围，用大括号括起，用以表示重复次数的范围。

"ab{2}"：表示一个字符串有一个a跟着2个b（"abb"）；

"ab{2,}"：表示一个字符串有一个a跟着至少2个b；

"ab{3,5}"：表示一个字符串有一个a跟着3到5个b。

请注意，你必须指定范围的下限（如："{0,2}"而不是"{,2}"）。

还有，你可能注意到了，'\*'，'+'和'?'相当于"{0,}"，"{1,}"和"{0,1}"。

还有一个'|'，表示“或”操作：

"hi|hello"：表示一个字符串里有"hi"或者"hello"；分支结构也是惰性的，即当前面的匹配上了，后面的就不再尝试了，即匹配到hi后就不会去匹配hello了

"(b|cd)ef"：表示"bef"或"cdef"；

"(a|b)\*c"：表示一串"a""b"混合的字符串后面跟一个"c"；

'.'可以替代任何字符：

"a.[0-9]"：表示一个字符串有一个"a"后面跟着一个任意字符和一个数字；

"^.{3}$"：表示有任意三个字符的字符串（长度为3个字符）；

方括号表示某些字符允许在一个字符串中的某一特定位置出现：

"[ab]"：表示一个字符串有一个"a"或"b"（相当于"a|b"）；

"[a-d]"：表示一个字符串包含小写的'a'到'd'中的一个（相当于"a|b|c|d"或者"[abcd]"）；

"^[a-zA-Z]"：表示一个以字母开头的字符串；

"[0-9]%"：表示一个百分号前有一位的数字；

",[a-zA-Z0-9]$"：表示一个字符串以一个逗号后面跟着一个字母或数字结束。

你也可以在方括号里用'^'表示不希望出现的字符，'^'应在方括号里的第一位。

（如："%[^a-zA-Z]%"表示两个百分号中不应该出现字母）。

为了逐字表达，你必须在"^.$()|\*+?{\"这些字符前加上转移字符'\'。

参考网站：

http://m.blog.csdn.net/zgrkaka/article/details/75089769