正则表达式

简介：

正则表达式是用于描述字符排列和匹配模式的一种语法规则。它主要用于字符串的模式分割、匹配、查找及替换操作。

三个作用：

匹配，也常常用于从字符串中析取信息。

用新文本代替匹配文本。

将一个字符串拆分为一组更小的信息块。

注：一个正则表达式中至少包含一个原子。

正则表达式作为一个匹配的模版，是由原子（普通字符，例如字符a到z）、特殊字符（元字符，例如\*、+和？等）、以及模式修正符三部分组成的文字模式。

在程序语言中，使用与Perl兼容的正则表达式，通常都需要将模式表达式放入定界符之间，如“/”。

作为定界符常使用反斜线“/”，如“/apple/”。用户只要把需要匹配的模式内容放入定界符之间即可。作为定界的字符也不仅仅局限于“/”。除了字母、数字和斜线“\”以外的任何字符都可以作为定界符，像 ‘#’、’|’、’!’ 等都可以的。

原子是正则表达式的最基本的组成单元，而且在每个模式中最少要少包含一个原子。原子是由所有那些未显示指定为元字符的打印和非打印字符组成，具体分为5类。

1. 普通字符作为原子： 如 a~z、A~Z、0~9 等

2. 一些特殊字符和转义后元字符作为原子：

所有标点符号，但语句特殊意义的符号需要转义后才可作为原子，如：\” \’ \\* \+ \? \. 等

3. 一些非打印字符作为原子： 如：\f \n \r \t \v \cx

4. 使用“通用字符类型”作为原子：如：\d \D \w \W \s \S。

5. 自定义原子表([])作为原子：如：’/[apj]sp/’ ’/[^apj]sp/’

原子字符

\cx 匹配由x指明的控制字符。如\cM匹配一个Control-M或回车符。X的值必须为A-Z或a-z之一。

\f 匹配一个换页符，等价于\x0c或\cL

\n 匹配一个换行符，等价于\x0a或\cJ

\r 匹配一个回车符，等价于\x0d或\cM

\t 匹配一个制表符，等价于\x09或\cI

\v 匹配一个垂直制表符，等价于\x0b或\cK

\d 匹配任意一个十进制数字，等价于[0-9]

\D 匹配任意一个除十进制以外的字符，等价于[^0-9]

\s 匹配任意一个空白符[\f\n\r\t\v]

\S 匹配除空白符以外的字符[^\f\n\r\t\v]

\w 匹配任意一个数字、字母、下划线，等价于[0-9a-zA-Z]

\W 匹配一个除数字、字母、下划线以外的任意一个字符，等价于[^0-9a-zA-Z]

元字符

字符串边界限制

1.在某些情况下，需要对匹配范围进行限定，以获得更准确的匹配结果。“^”和“$”分别指定字符串的开始和结束

2.元字符“^”或“\A” 置于字符串的开始确保模式匹配出现在字符串首端；

3.元字符“$”或“\Z” 置于字符串的结束，确保模式匹配出现字符串尾端。

注：如果不加边界限制元字符，将获得更多的匹配结果。

按单词查找

1.元字符“\b”对单词的边界进行匹配；

2.元字符“\B”对除单词边界以外的部分进行匹配。

正则表达式中有一些用于重复匹配某些原子的元字符：“?”、“\*”、“+”。他们主要的区别是重复匹配的次数不同。

1.元字符“?”:表示0次或1次匹配紧接在其前的原子。

例如：/colou?r/匹配“colour”或“color”。

2.元字符“\*”:表示0次、1次或多次匹配紧接在其前的原子。

例如：/zo\*/可以匹配z、zoo

3.元字符“+”:表示1次或多次匹配紧接在其前的原子。

例如：/go+gle/匹配“gogle”、“google”或“gooogle”等中间含有多个o的字符串。

元字符“.”匹配除换行符外任何一个字符。

1.通常可以使用“.\*”组合来匹配除换行符外的任何字符。在一些

2.书籍中也称其为“全匹配符” 或 “单含匹配符”。

元字符“{ }”准确地指定原子重复的次数，指定所匹配的原子出现的次数。

1.“{m}” 表示其前原子恰好出现m次。

2.“{m，n}”表示其前原子至少出现m次，至多出现n次。

3.“{m，}” 表示其前原子出现不少于m次。

原子表”[]”中存放一组原子，彼此地位平等，且仅匹配其中的一个原子。如果想匹配一个 ”a” 或 ”e” 使用 [ae]。

例如: Pr[ae]y 匹配 ”Pray” 或者 ”Prey ”。

原子表 ”[^]” 或者称为排除原子表，匹配除表内原子外的任意一个字符。

例如：/p[^u]/匹配“part”中的“pa”，但无法匹配“computer”中的“pu”因为“u”在匹配中被排除。

原子表“[-]”用于连接一组按ASCII码顺序排列的原子，简化书写。

例如：/x[0123456789]/可以写成x[0-9],用来匹配一个由 “x” 字母与一个数字组成的字符串。

元字符“|”又称模式选择符。在正则表达式中匹配两个或更多的选择之一

元字符“（）”将其中的正则表达式变为原子（或称模式单元）使用。与数学表达式中的括号类似，“（）”可以做一个单元被单独使用。

注：一个模式单元中的表达式将被优先匹配或运算。

（1）系统自动将模式单元“（）”中的匹配依次存储起来，在需要时可以用“\1”、“\2”、“\3”的形式进行引用。当正则表达式包含有相同的模式单元时，这种方法非常便于对其进行管理。注意使用时需要写成“\\1”、“\\2”

（2）当不需要存储匹配结果时使用非存储模式单元“（？：）”

模式修正符

i :不区分大小写

m : 将字符串视为多行

s : 将字符串视为一行

x : 忽略正则表达式中的空格

U : 取消贪婪模式 .\*?