

$str =~ /\p{Space}/ #有无空格

$str =~ /\p{Digit}/ #数字

if($str =~ /\p{Hex}/) #16进制数 【0-9A-Fa-f】

把p改为P，上面表达式表示否定意义

元字符：

. :匹配任意一个字符，换行符除外

\

\*: 匹配0或多次

.\*:匹配任意字符0-无限次

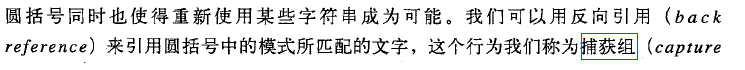
+ ：匹配1次以上

？：匹配0次或1次

（）

模式分组：()

反向引用：\1,\2



（.）\1 匹配连续出现的两个同样的字符。

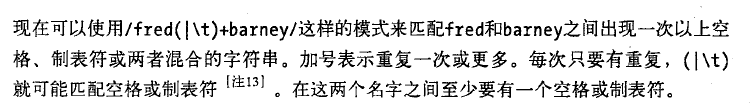
$\_ = “yabba dabba doo”;

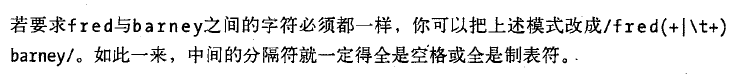
/y(....) d\1/ 或者/y(.)(.)\2\1/ #匹配abba

$\_=”aa11bb”

/(.)\g{1}11/ 或者g1 #g{N}：N为组号

|:或匹配





字符集：

放在[]中，他只匹配字符集中的单个字符，exp:[abcdefg] 只匹配一个

[a-zA-Z]

[\000-\177]:匹配任意一个7位的ASCII字符

脱字符：^

[^def]:匹配除def以外的任何字符

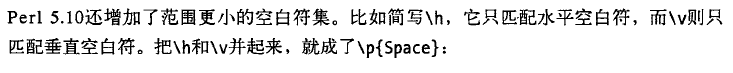
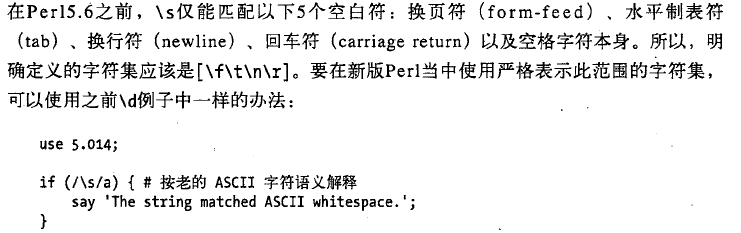
[^\n-z]:匹配除了n,-,z以外的字符

字符集括号外面的连字符-没有特殊意义

\d :数字字符集 各种语言的数字

/a: 严格按照ASCII范围匹配数字字符时，(写在末尾) /HAL-[\d]+/a

\s :匹配空白符，同\p{Spacs}



\R: 断行符 不管是\r\n或者\n都行

\w:单词字符

/s匹配任意字符，包括换行符



/x 可以在模式中添加空格，方便阅读

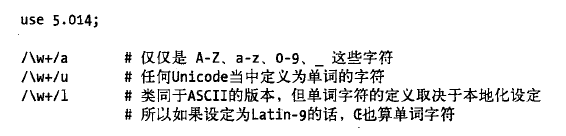
\s \s\* \s+ 匹配空白符

/i 大小写无关

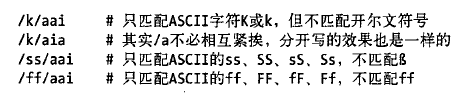
/a ascii码方式

/u Unicode方式 【更宽泛】

/l遵从本地化语言设置



若使用两个a，进一步表示仅仅采用ascii方式的大小写映射处理



锚位：

\A: 匹配字符串的绝对开头 ，/\Ahttp:/i :是否以http：开头

\z: 匹配字符串的绝对末尾。 /\.png\z/i :是否以.png结尾

\Z 行末锚位，允许后面出现换行符

**/\A\s\*\Z/** :匹配空行【允许包含若干空白符，包括制表符和空格】

^:字符串开头锚位 /^barry/m

$:字符串结尾锚位 /fred$/m

$/m :对多行内容进行匹配

若无/m,^和$的行为同\A,\z

\b单词[a-z,0-9,\_]边界锚位\B /\bfred\b/ 可匹配fred 但不能匹配frederick,afred

不捕获圆括号（?:）：Perl正则表达式允许使用圆括号分组但不进行捕获，只是分组用

?四种用法：本身，数量可有可无，非贪婪匹配，放弃捕获

#bronto只是起到匹配作用，并不想用来捕获，$1就可以捕获（|）了

If(/(?:bronto)?saurus (steak|burger)/){

Print “Fred wants a $1\n”;

}

捕获内容直接命名： ?<LABEL>PATTERN

%+ ：保存捕获组捕获到的内容 ，键即为捕获时用的特殊标签

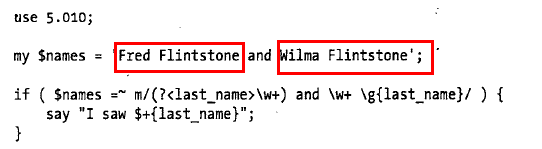
My $names = ‘fred or barney’;

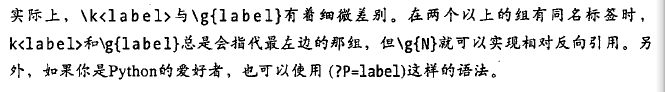
If($names =~ m/(**\w+**)(and |or)(\w+)/) #对比

If($names =~ m/(**?<name1>\w+**)(?:and | or)(?<name2>\w+)/){

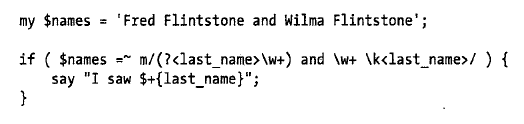
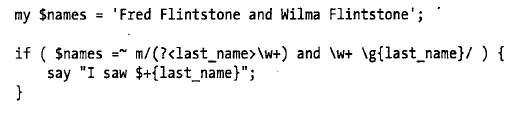
Say “I saw **$+{name1}** and $+{name2}”; #输出捕获

}





\k<label> 等效于 \g{label}



不加()也能使用的捕获变量，保存在：$&、$`、$’

If(“hello there,neighbor” =~ /\s(\w+),/){

Print “that actually matched ‘**$&**’”; #输出形式 there

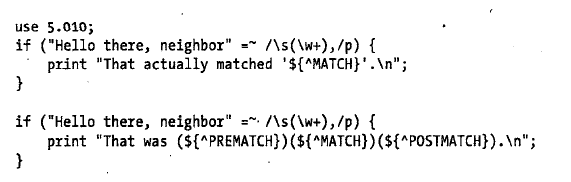
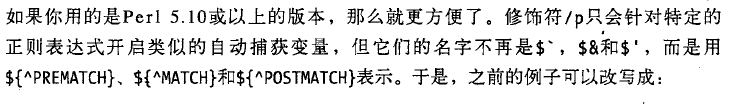
}

$&保存的是【\_there,】

目标字符串保存在$1中：【there】

目标字符串前的内容放在$`: 【\_】

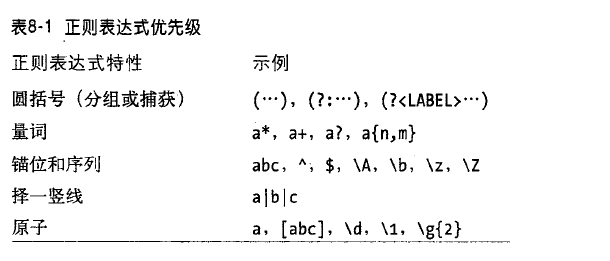
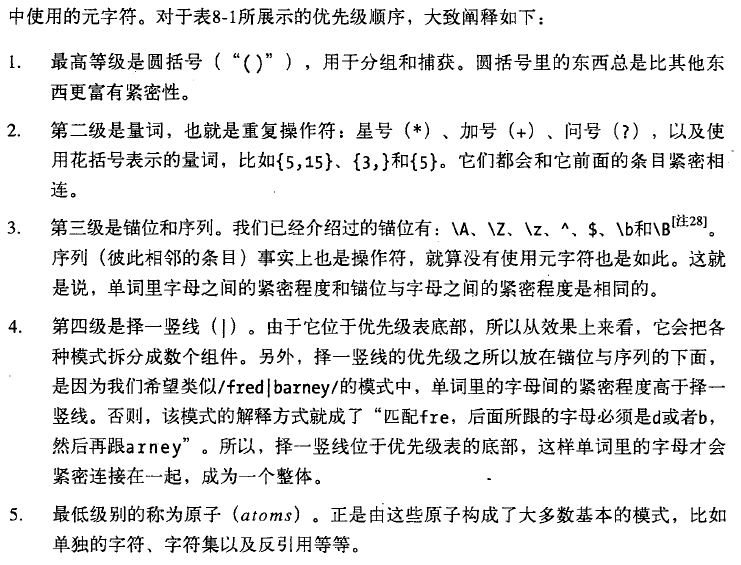
目标字符串后的内容放在$’: 【,】

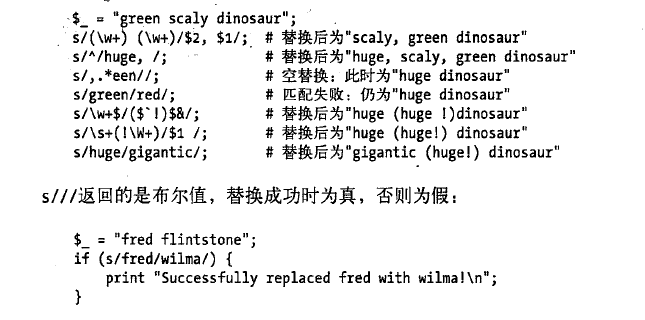


模式中的量词：

\* 【{0，}】，+【{1，}】，？【{0,1}】，{}

/a{5,15}/ 若a出现20次只会匹配前15次





/g可以让s///进行全局替换

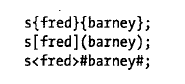
删除开头结尾的空白

S/^\s+// 开头

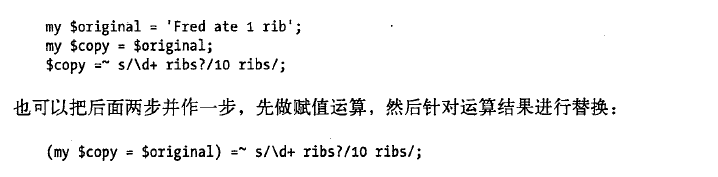
S/\s+$// 结尾

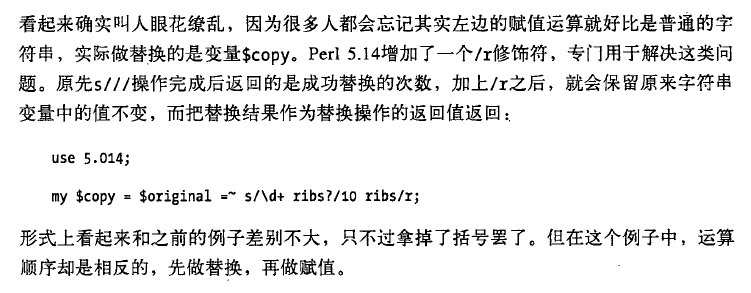
或者 s/^\s+|\s+$//g 效率慢

不同定界符的写法：







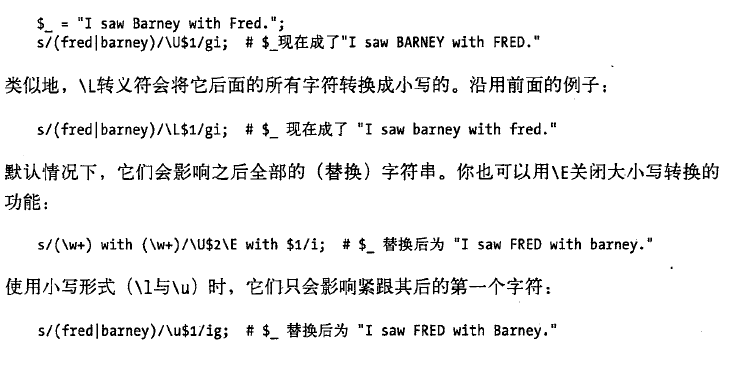


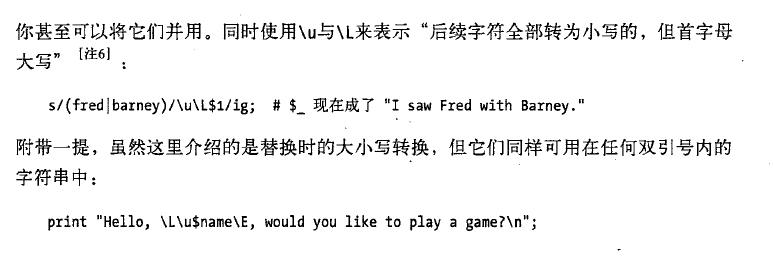
\U将它后面的字符全部转义成大写

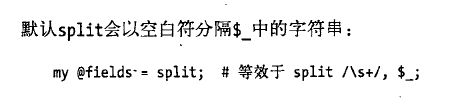
\L转成小写

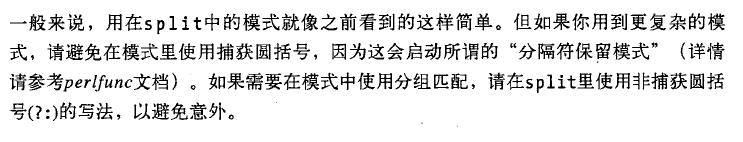
\E关闭大小写转换

使用小写（\l,\u）只影响紧跟其后的第一个字符









非贪婪量词：+？，\*？，{5,10}？，{8，}？，？？

/fred.+?barney/ 匹配最少的字符串

$^I: 内置控制编辑器的值