\d 匹配数字

(\d) matches the first digit and captures it

\1 references the captured digit

[0-9] 或者 \d

Match Non-Digits 用\D, 也可以通过

\w 来替换，words, 去除特殊字符

\w = [a-zA-Z0-9]

**\W**

This shorthand matches whitespace, punctuation, and other kinds of characters that

aren’t used in words in this example. It is the same as using the following character class:

[^a-zA-Z0-9]

*Table 2-1. Character shorthands*

Character Shorthand Description

\a Alert

\b Word boundary

[\b] Backspace character

\B Non-word boundary

\c *x* Control character

**\d Digit character**

**\D Non-digit character**

\d *xxx* Decimal value for a character

\f Form feed character

\r Carriage return

\n Newline character

pass:[<literal>\o</literal> Octal value for a character

<replaceable>\*xxx*</replaceable>]

**\s Space character**

**\S Non-space character**

\t Horizontal tab character

\v Vertical tab character

**\w Word character**

**\W Non-word character**

\0 Nul character

\ x*xx* Hexadecimal value for a character

\u *xxxx* Unicode value for a character

Character Shorthand Description

\f Form feed

\h Horizontal whitespace

\H Not horizontal whitespace

\n Newline

\r Carriage return

\t Horizontal tab

\v Vertical tab (whitespace)

\V Not vertical whitespace

. (dot)

点用来匹配所有字符

^ 匹配行首字符

$ 匹配行末字符

\b 边界字符 使用方法 \bAAA\b 定界 AAA 字符

\Be\B 不定界

\Q 是使用转换字符.^$\*+?|(){}[]\- 作为普通字符

\Q$\E

(?d) Unix lines Java

(?i) Case insensitive PCRE, Perl, Java

(?J) Allow duplicate names PCRE\*

(?m) Multiline PCRE, Perl, Java

(?s) Single line (dotall) PCRE, Perl, Java

(?u) Unicode case Java

(?U) Default match lazy PCRE

(?x) Ignore whitespace, comments PCRE, Perl, Java

(?-…) Unset or turn off options PCRE

\1

or:

$1

eG:

(It is) (an ancyent Marinere)

Scroll the subject text (third text area) down until you can see the highlighted line, and

then in the second box, enter:

$2 $1

Named Groups

*Named groups* are captured groups with names. You can access those groups by name

later, rather than by integer. I’ll show you how here in Perl:

perl -ne 'print if s/(?<one>It is) (?<two>an ancyent Marinere)/\u$+{two}

\l$+{one}/' rime.txt

(?<*name*>…) A named group

(?*name*…) Another named group

(?P<*name*>…) A named group in Python

\k<*name*> Reference by name in Perl

\k'*name*' Reference by name in Perl

\g{*name*} Reference by name in Perl

\k{*name*} Reference by name in .NET

(?P=*name*) Reference by name in Python

? Zero or one (optional)

+ One or more

\* Zero or more

{*n*} Match *n* times exactly

{*n*,} Match *n* or more times

{*m,n*} Match *m* to *n* times

{0,1} Same as ? (zero or one)

{1,0} Same as + (one or more)

{0,} Same as \* (zero or more)

*Table 7-3. Lazy quantifiers*

Syntax Description

?? Lazy zero or one (optional)

+? Lazy one or more

\*? Lazy zero or more

{n}? Lazy n

{n,}? Lazy n or more

{m,n}? Lazy m,n

?+ Possessive zero or one (optional)

++ Possessive one or more

\*+ Possessive zero or more

{n}+ Possessive n

{n,}+ Possessive n or more

{m,n}+ Possessive m,n