

# 正则表达式快速参考手册

胡志飞 <WisdomFusion#gmail.com>

2012 年 6 月 2 日

# 目 录

<b>1 简介</b>	<b>1</b>
<b>2 基本语法</b>	<b>1</b>
<b>3 高级语法</b>	<b>6</b>
<b>4 举些栗子</b>	<b>9</b>
<b>5 正则表达式“流派”</b>	<b>9</b>
<b>6 应用场景</b>	<b>9</b>
6.1 正则表达式工具箱 . . . . .	9
6.2 应用案例 . . . . .	9

# 1 简介 Introduction

正则表达式, ( **Regular Expression**, 在代码中常简写为 **regex**、**regexp** 或 **RE** ), 计算机科学的一个概念。正则表达式使用字符来描述、匹配一系列符合某个句法规则的字符串。在很多文本编辑器里, 正则表达式通常被用来检索、替换那些符合某个模式的文本。许多程序设计语言都支持利用正则表达式进行字符串操作。例如, 在 **Perl**<sup>1</sup>中就内建了一个功能强大的正则表达式引擎。正则表达式这个概念最初是由 **Unix** 中的工具软件 ( 例如 **sed**<sup>2</sup>和 **grep**<sup>3</sup> ) 普及开的。

需要注意的是, 用什么工具, 用什么编辑语言, 正则表达式的语法有些差别, 特性的支持也参差不齐, 称之为正则表达式“流派”(第5部分详述 ), 所以要单独参考工具和编程语言本身的文档才行。

# 2 基本语法 Basic Syntax

特性	语法	描述	举个例子
字符	除 <code>[ \ ^ \$ .   ? * + 0</code> 以外的任意字符	除了 <code>[ \ ^ \$ .   ? * + 0</code> 以外的任意字符, {和} 也是文字文本, 除了下面说到的成对出现的量词语法, 如 <code>{n}</code> 和 <code>{m,n}</code> 等。	<code>a</code> 匹配 <code>about</code> 中的 <code>a</code>
	字符转义	<code>\t, \?, \*, \+, \., \ , \{, \}, \\, \[, \], \(. \)</code>	<code>\+</code> 匹配 <code>+</code> ; <code>\? \-</code> 匹配 <code>?-</code>
	<code>\n, \r</code> 和 <code>\t</code>	Windows 文件格式换行符是 <code>\r\n</code> , UNIX 文件格式换行符是 <code>\n</code> , <code>\t</code> 匹配水平制表符	

To be continued...

<sup>1</sup>**Perl**被称为“实用报表提取语言”( **Practical Extraction and Report Language** ), 正则表达式特性的推动者, 文本处理非常方便。  
<sup>2</sup>**sed**是一种 **UNIX/Linux** 平台下的轻量级流编辑器, 日常一般用于处理文本文件。  
<sup>3</sup>**grep**, **global search regular expression and print out the line**, 是一种强大的文本搜索工具, 它能使用正则表达式搜索文本, 并把匹配的行打印出来。

( 续表 )

特性	语法	描述	举个栗子
	<code>\cA</code> 到 <code>\cZ</code> , <code>\ca</code> 到 <code>\cz</code> <code>\a</code> , <code>\e</code> , <code>\f</code> , <code>\v</code> <code>\Q ... \E</code>	<code>Ctrl</code> + <code>A</code> 到 <code>Ctrl</code> + <code>Z</code> , 与 ASCII 字符 <code>\x01</code> 到 <code>\x1A</code> 等价  依次为警报 ( <code>\x07</code> )、Esc 字符 ( <code>\x1B</code> )、进纸符 ( <code>\x0C</code> ) 和垂直制表符 ( <code>\x0B</code> )  文字文本范围, 被包含在 <code>\Q</code> 和 <code>\E</code> 之间的文字, 都被视为普通文字, 如 <code>[^\\$. ?*\+O{}]</code> 也不再转义了, 这个最早是由 Perl 引入正则表达式的。	<code>\Q+.*\/\E</code> 匹配的就是 <code>+.*\/</code>
基本特性	<code>.</code> ( 点 )  <code> </code>	匹配除换行符之外的任意字符, 有些正则表达式“流派”还支持点是否匹配换行符的开关。  管道, 或的关系, 匹配 <code> </code> 的左侧或右侧的字符串	<code>.</code> 匹配 <code>about</code> 中的任意一个字符  <code>abc def xyz</code> 匹配 <code>abc</code> 或 <code>def</code> 或 <code>xyz</code>
字符类	<code>[...]</code>  <code>[^\]]</code>  <code>[^...]</code>	匹配字符类中列举的任意一个字符  在字符类中, 要匹配 <code>^-]</code> 这几字符, 得使用 <code>\</code> 转义  排除型字符类, <code>^</code> ( 脱字符, <code>caret</code> ) 紧跟 <code>[</code> 之后, 可以把字符类中列举的字符排除匹配范围, 也就是这个字符类将匹配任意一个不在列出字符范围内的字符	<code>[abc]</code> 匹 配 <code>a</code> 或 <code>b</code> 或 <code>c</code> ; <code>[!?!]</code> 匹 配 <code>.</code> 或 <code>!</code> 或 <code>?</code>  <code>[^\]]</code> 匹配 <code>^</code> 或 <code>]</code>  <code>[^a-d]</code> 匹配除了 <code>a,b,c,d</code> 之外的任意一个字符

To be continued...

(续表)

特性	语法	描述	举个栗子
	<code>\d</code> , <code>\w</code> , <code>\s</code>	<code>\d</code> 匹配数字, 与 <code>[0-9]</code> 等价; <code>\w</code> 匹配任意一个字母或数字或下划线或汉字; <code>\s</code> 匹配任意一个空白符	<code>[\d\s]</code> 匹配一个数字或空白符
	<code>\D</code> , <code>\W</code> , <code>\S</code>	是 <code>\d</code> , <code>\w</code> 和 <code>\s</code> 的反义字符类。 <code>\D</code> 匹配任意非数字的字符; <code>\W</code> 匹配任意不是字母、数字、下划线、汉字的字符; <code>\S</code> 匹配任意不是空白符的字符	<code>\D</code> 匹配任意非数字的字符
	<code>[\b]</code>	在字符类中, <code>[\b]</code> 为 <code>Backspace</code> 退格键字符	
POSIX	<code>[:alnum:]</code>		
	<code>[:alpha:]</code>		
	<code>[:ascii:]</code>		
	<code>[:blank:]</code>		
	<code>[:cntrl:]</code>		
	<code>[:digit:]</code>		
	<code>[:graph:]</code>		
	<code>[:lower:]</code>		

To be continued...

( 续表 )

特性	语法	描述	举个例子
	<code>[:print:]</code>		
	<code>[:punct:]</code>		
	<code>[:space:]</code>		
	<code>[:upper:]</code>		
	<code>[:word:]</code>		
	<code>[:xdigit:]</code>		
锚点	<code>^</code>		
	<code>\$</code>		
	<code>\A</code>		
	<code>\G</code>		
	<code>\Z</code>		
	<code>\b</code>		
	<code>\B</code>		
	<code>\&lt;</code>		

To be continued...

( 续表 )

特性	语法	描述	举个栗子
	\>		
	\`		
	\&		
	\'		
量词	?		
	??		
	?+		
	*		
	*?		
	*+		
	+		
	+?		
	++		

To be continued...

( 续表 )

特性	语法	描述	举个栗子
	{n}		
	{n,m}		
	{n,}		
	{,m}		
	{n,m}?		
	{n,m}+		

3 高级语法 Advanced Syntax

特性	语法	描述	举个栗子
Unicode			

To be continued...



( 续表 )

特性	语法	描述	举个栗子
分组与反向引用	(regex)		
	(?:regex)		
	\1 到 \9		
	\10 到 \99		
	\g{1} 到 \g{99}		
	\g{-1}, \g{-2}, etc.		
	(?<name>regex)		
	\k<name>, \g{name}		

To be continued...

( 续表 )

特性	语法	描述	举个栗子
模式修饰符	(?i)		
	(?-i)		
	(?s)		
	(?-s)		
	(?m)		
	(?-m)		
	(?x)		
高级分组	(?#comment)		
	(?=Regex)		
	(?!Regex)		
	(?<=regex)		
	(?<!regex)		

## 4 举些栗子 Regex Examples

一些栗子

## 5 正则表达式“流派”Regex Flavors

流派

## 6 应用场景 Application Scenarios

### 6.1 正则表达式工具箱 Regex Toolbox

总有一款适合你，Windows 下的记事本太鸡肋，Word 处理方式主要是“通配符”而不是正则表达式。

#### RegexBuddy

JGsoft 开发的一个强大的正则表达式测试工具，

#### grep

#### PowerGREP

RegexBuddy 的兄弟软件，同是 JGsoft 开发，是 grep 在 Windows

平台的实现和增强。

#### UltraEdit, Notepad++

#### Vim

#### GNU Emacs

#### sed & awk

### 6.2 应用案例 Application Cases

#### Dreamweaver 表格处理

#### VBA 中使用正则表达式

```
1 Sub IndentParaWithRegEx()  
2 ' PowerPoint VBA 批量给指定字符开头段落加动画
```

```
3 Dim oSld As Slide
4 Dim oShp As Shape
5 Dim i As Integer
6 ' 正则相关变量
7 Dim regx As Object, oMatch As Object
8
9 ' 这里写查找的正则, 参考 http://msdn.microsoft.com/en-us/library/ms974570.aspx
10 strPattern = "^开头字符串"
11
12 Set regx = CreateObject("vbscript.regexp")
13 With regx
14     .Global = True
15     .IgnoreCase = True
16     .Pattern = strPattern
17 End With
18
19 For Each oSld In ActivePresentation.Slides
20     For Each oShp In oSld.Shapes
21         If oShp.HasTextFrame Then
22             If oShp.TextFrame2.HasText Then
23                 With oShp.TextFrame2.TextRange
24                     For i = 1 To .Paragraphs.Count
25                         With .Paragraphs(i)
26                             ' 可能会出现多个匹配项的
27                             If (regx.Test(.Text) = True) Then
28                                 .ParagraphFormat.FirstLineIndent = 0
29                             End If
30                         End With
31                     Next i 'para
32                 End With
33             End If 'has text
34         End If 'has textframe
```

```
35     Next oShp
36 Next oSld
37 End Sub
```

## InDesign GREP

使用 **Perl** 正则表达式处理文件  
神的编辑器之正则