正则表达式是一种文本模式，包括普通字符（例如，a 到 z 之间的字母）和特殊字符（称为“元字符”）。模式描述在搜索文本时要匹配的一个或多个字符串。

**1.正则表达式基本语法**

两个特殊的符号'^'和'$'。他们的作用是分别指出一个字符串的开始和结束。例子如下：

"^The"：表示所有以"The"开始的字符串（"There"，"The cat"等）；  
"of despair$"：表示所以以"of despair"结尾的字符串；  
"^abc$"：表示开始和结尾都是"abc"的字符串——呵呵，只有"abc"自己了；  
"notice"：表示任何包含"notice"的字符串。

象最后那个例子，如果你不使用两个特殊字符，你就在表示要查找的串在被查找串的任意部分——你并  
不把它定位在某一个顶端。

其它还有'\*'，'+'和'?'这三个符号，表示一个或一序列字符重复出现的次数。它们分别表示“没有或  
更多”，“一次或更多”还有“没有或一次”。下面是几个例子：

"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?"：表示一个字符串有一个a后面跟着零个或者一个b；  
"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。

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"；  
"(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'\'。

请注意在方括号中，不需要转义字符。

**2.正则表达式验证控制文本框的输入字符类型**

**1.只能输入数字和英文的：**

<input onkeyup="value=value.replace(/[\W]/g,'') " onbeforepaste="clipboardData.setData('text',clipboardData.getData('text').replace(/[^\d]/g,''))" ID="Text1" NAME="Text1">

**2.只能输入数字的：**

<input onkeyup="value=value.replace(/[^\d]/g,'') " onbeforepaste="clipboardData.setData('text',clipboardData.getData('text').replace(/[^\d]/g,''))" ID="Text2" NAME="Text2">

**3.只能输入全角的：**

<input onkeyup="value=value.replace(/[^\uFF00-\uFFFF]/g,'')" onbeforepaste="clipboardData.setData('text',clipboardData.getData('text').replace(/[^\uFF00-\uFFFF]/g,''))" ID="Text3" NAME="Text3">

**4.只能输入汉字的：**

<input onkeyup="value=value.replace(/[^\u4E00-\u9FA5]/g,'')" onbeforepaste="clipboardData.setData('text',clipboardData.getData('text').replace(/[^\u4E00-\u9FA5]/g,''))" ID="Text4" NAME="Text4">

**3.正则表达式的应用实例通俗说明**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//校验是否全由数字组成

/^[0-9]{1,20}$/

^ 表示打头的字符要匹配紧跟^后面的规则

$ 表示打头的字符要匹配紧靠$前面的规则

[ ] 中的内容是可选字符集

[0-9] 表示要求字符范围在0-9之间

{1,20}表示数字字符串长度合法为1到20，即为[0-9]中的字符出现次数的范围是1到20次。  
/^ 和 $/成对使用应该是表示要求整个字符串完全匹配定义的规则，而不是只匹配字符串中的一个子串。  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//校验登录名：只能输入5-20个以字母开头、可带数字、“\_”、“.”的字串

/^[a-zA-Z]{1}([a-zA-Z0-9]|[.\_]){4,19}$/

^[a-zA-Z]{1} 表示第一个字符要求是字母。

([a-zA-Z0-9]|[.\_]){4,19} 表示从第二位开始（因为它紧跟在上个表达式后面）的一个长度为4到9位的字符串，它要求是由大小写字母、数字或者特殊字符集[.\_]组成。

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//校验用户姓名：只能输入1-30个以字母开头的字串

/^[a-zA-Z]{1,30}$/

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//校验密码：只能输入6-20个字母、数字、下划线

/^(\w){6,20}$/

\w：用于匹配字母，数字或下划线字符

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//校验普通电话、传真号码：可以“+”或数字开头，可含有“-” 和 “ ”

/^[+]{0,1}(\d){1,3}[ ]?([-]?((\d)|[ ]){1,12})+$/

\d：用于匹配从0到9的数字；

“?”元字符规定其前导对象必须在目标对象中连续出现零次或一次

可以匹配的字符串如：+123 -999 999 ； +123-999 999 ；123 999 999 ；+123 999999等

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//校验URL

/^http[s]{0,1}:\/\/.+$/ 或 /^http[s]{0,1}:\/\/.{1,n}$/ (表示url串的长度为length(“https://”) + n )

\ / ：表示字符“/”。

. 表示所有字符的集

+ 等同于{1,}，就是1到正无穷吧。

**一、校验数字的表达式**

1 数字：^[0-9]\*$

2 n位的数字：^\d{n}$

3 至少n位的数字：^\d{n,}$

4 m-n位的数字：^\d{m,n}$

5 零和非零开头的数字：^(0|[1-9][0-9]\*)$

6 非零开头的最多带两位小数的数字：^([1-9][0-9]\*)+(.[0-9]{1,2})?$

7 带1-2位小数的正数或负数：^(\-)?\d+(\.\d{1,2})?$

8 正数、负数、和小数：^(\-|\+)?\d+(\.\d+)?$

9 有两位小数的正实数：^[0-9]+(.[0-9]{2})?$

10 有1~3位小数的正实数：^[0-9]+(.[0-9]{1,3})?$

11 非零的正整数：^[1-9]\d\*$ 或 ^([1-9][0-9]\*){1,3}$ 或 ^\+?[1-9][0-9]\*$

12 非零的负整数：^\-[1-9][]0-9"\*$ 或 ^-[1-9]\d\*$

13 非负整数：^\d+$ 或 ^[1-9]\d\*|0$

14 非正整数：^-[1-9]\d\*|0$ 或 ^((-\d+)|(0+))$

15 非负浮点数：^\d+(\.\d+)?$ 或 ^[1-9]\d\*\.\d\*|0\.\d\*[1-9]\d\*|0?\.0+|0$

16 非正浮点数：^((-\d+(\.\d+)?)|(0+(\.0+)?))$ 或 ^(-([1-9]\d\*\.\d\*|0\.\d\*[1-9]\d\*))|0?\.0+|0$

17 正浮点数：^[1-9]\d\*\.\d\*|0\.\d\*[1-9]\d\*$ 或 ^(([0-9]+\.[0-9]\*[1-9][0-9]\*)|([0-9]\*[1-9][0-9]\*\.[0-9]+)|([0-9]\*[1-9][0-9]\*))$

18 负浮点数：^-([1-9]\d\*\.\d\*|0\.\d\*[1-9]\d\*)$ 或 ^(-(([0-9]+\.[0-9]\*[1-9][0-9]\*)|([0-9]\*[1-9][0-9]\*\.[0-9]+)|([0-9]\*[1-9][0-9]\*)))$

19 浮点数：^(-?\d+)(\.\d+)?$ 或 ^-?([1-9]\d\*\.\d\*|0\.\d\*[1-9]\d\*|0?\.0+|0)$

**二、校验字符的表达式**

1 汉字：^[\u4e00-\u9fa5]{0,}$

2 英文和数字：^[A-Za-z0-9]+$ 或 ^[A-Za-z0-9]{4,40}$

3 长度为3-20的所有字符：^.{3,20}$

4 由26个英文字母组成的字符串：^[A-Za-z]+$

5 由26个大写英文字母组成的字符串：^[A-Z]+$

6 由26个小写英文字母组成的字符串：^[a-z]+$

7 由数字和26个英文字母组成的字符串：^[A-Za-z0-9]+$

8 由数字、26个英文字母或者下划线组成的字符串：^\w+$ 或 ^\w{3,20}$

9 中文、英文、数字包括下划线：^[\u4E00-\u9FA5A-Za-z0-9\_]+$

10 中文、英文、数字但不包括下划线等符号：^[\u4E00-\u9FA5A-Za-z0-9]+$ 或 ^[\u4E00-\u9FA5A-Za-z0-9]{2,20}$

11 可以输入含有^%&',;=?$\"等字符：[^%&',;=?$\x22]+

12 禁止输入含有~的字符：[^~\x22]+

**三、特殊需求表达式**

1 Email地址：^\w+([-+.]\w+)\*@\w+([-.]\w+)\*\.\w+([-.]\w+)\*$

2 域名：[a-zA-Z0-9][-a-zA-Z0-9]{0,62}(/.[a-zA-Z0-9][-a-zA-Z0-9]{0,62})+/.?

3 InternetURL：[a-zA-z]+://[^\s]\* 或 ^http://([\w-]+\.)+[\w-]+(/[\w-./?%&=]\*)?$

4 手机号码：^(13[0-9]|14[5|7]|15[0|1|2|3|5|6|7|8|9]|18[0|1|2|3|5|6|7|8|9])\d{8}$

5 电话号码("XXX-XXXXXXX"、"XXXX-XXXXXXXX"、"XXX-XXXXXXX"、"XXX-XXXXXXXX"、"XXXXXXX"和"XXXXXXXX)：^(\(\d{3,4}-)|\d{3.4}-)?\d{7,8}$

6 国内电话号码(0511-4405222、021-87888822)：\d{3}-\d{8}|\d{4}-\d{7}

7 身份证号(15位、18位数字)：^\d{15}|\d{18}$

8 短身份证号码(数字、字母x结尾)：^([0-9]){7,18}(x|X)?$ 或 ^\d{8,18}|[0-9x]{8,18}|[0-9X]{8,18}?$

9 帐号是否合法(字母开头，允许5-16字节，允许字母数字下划线)：^[a-zA-Z][a-zA-Z0-9\_]{4,15}$

10 密码(以字母开头，长度在6~18之间，只能包含字母、数字和下划线)：^[a-zA-Z]\w{5,17}$

11 强密码(必须包含大小写字母和数字的组合，不能使用特殊字符，长度在8-10之间)：^(?=.\*\d)(?=.\*[a-z])(?=.\*[A-Z]).{8,10}$

12 日期格式：^\d{4}-\d{1,2}-\d{1,2}

13 一年的12个月(01～09和1～12)：^(0?[1-9]|1[0-2])$

14 一个月的31天(01～09和1～31)：^((0?[1-9])|((1|2)[0-9])|30|31)$

15 钱的输入格式：

16 1.有四种钱的表示形式我们可以接受:"10000.00" 和 "10,000.00", 和没有 "分" 的 "10000" 和 "10,000"：^[1-9][0-9]\*$

17 2.这表示任意一个不以0开头的数字,但是,这也意味着一个字符"0"不通过,所以我们采用下面的形式：^(0|[1-9][0-9]\*)$

18 3.一个0或者一个不以0开头的数字.我们还可以允许开头有一个负号：^(0|-?[1-9][0-9]\*)$

19 4.这表示一个0或者一个可能为负的开头不为0的数字.让用户以0开头好了.把负号的也去掉,因为钱总不能是负的吧.下面我们要加的是说明可能的小数部分：^[0-9]+(.[0-9]+)?$

20 5.必须说明的是,小数点后面至少应该有1位数,所以"10."是不通过的,但是 "10" 和 "10.2" 是通过的：^[0-9]+(.[0-9]{2})?$

21 6.这样我们规定小数点后面必须有两位,如果你认为太苛刻了,可以这样：^[0-9]+(.[0-9]{1,2})?$

22 7.这样就允许用户只写一位小数.下面我们该考虑数字中的逗号了,我们可以这样：^[0-9]{1,3}(,[0-9]{3})\*(.[0-9]{1,2})?$

23 8.1到3个数字,后面跟着任意个 逗号+3个数字,逗号成为可选,而不是必须：^([0-9]+|[0-9]{1,3}(,[0-9]{3})\*)(.[0-9]{1,2})?$

24 备注：这就是最终结果了,别忘了"+"可以用"\*"替代如果你觉得空字符串也可以接受的话(奇怪,为什么?)最后,别忘了在用函数时去掉去掉那个反斜杠,一般的错误都在这里

25 xml文件：^([a-zA-Z]+-?)+[a-zA-Z0-9]+\\.[x|X][m|M][l|L]$

26 中文字符的正则表达式：[\u4e00-\u9fa5]

27 双字节字符：[^\x00-\xff] (包括汉字在内，可以用来计算字符串的长度(一个双字节字符长度计2，ASCII字符计1))

28 空白行的正则表达式：\n\s\*\r (可以用来删除空白行)

29 HTML标记的正则表达式：<(\S\*?)[^>]\*>.\*?</\1>|<.\*? /> (网上流传的版本太糟糕，上面这个也仅仅能部分，对于复杂的嵌套标记依旧无能为力)

30 首尾空白字符的正则表达式：^\s\*|\s\*$或(^\s\*)|(\s\*$) (可以用来删除行首行尾的空白字符(包括空格、制表符、换页符等等)，非常有用的表达式)

31 腾讯QQ号：[1-9][0-9]{4,} (腾讯QQ号从10000开始)

32 中国邮政编码：[1-9]\d{5}(?!\d) (中国邮政编码为6位数字)

33 IP地址：\d+\.\d+\.\d+\.\d+ (提取IP地址时有用)

34 IP地址：((?:(?:25[0-5]|2[0-4]\\d|[01]?\\d?\\d)\\.){3}(?:25[0-5]|2[0-4]\\d|[01]?\\d?\\d))