Js 正则创建

var reg=new RegExp(pattern)

Var reg=//img

使用方式

reg.test(str) 用正则检测字符串，返回的布尔值

str.macth(reg) 用字符串匹配正则，结果是数组

修饰符

i 执行对大小写不敏感的匹配

g 执行全局匹配

m 执行多行匹配

方括号

1. zA-z0-9] 匹配之间的任何字符

(red|blue|green) 查找任何指定的选项

转义字符 字符串里面也要转义

/\./ig /\[rH\]/ig /\\w/ig

元字符

. 查找单个字符，除了换行和行结束符

\w 查找单个字符

\W 非单个字符

\d 查找数字

\D 查找非数字字符

\s 查找空字符

\S 查找非空白字符

\b 匹配单词边界

\B 匹配非单词边界

\O 查找nul字符

\n 查找换行符

\uxxxx 查找以十六制数xxxx规定的Unicode字符

量词

n+ == n{1,}

n\* == n{0,}

n? == n{0,1}

^n$ 开头跟结尾

?=n 紧接着

?!n 匹配任何其后没有紧跟指定字符串n的字符串

## **史上最全正则表达式**

<https://www.cnblogs.com/fozero/p/7868687.html>

## **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))

正则的意思

(?![0-9]+$) 声明该位置后面不全是数字;

(?![a-zA-Z]+$) 声明该位置后面不全是字母;

[0-9A-Za-z] {8,30} 声明由8-30位数字或这字母组成;

$ 匹配行结尾位置;

/^(\w){6,20}$/ 只能输入6-20个字母、数字、下划线

^(?![0-9]+$)(?![a-zA-Z]+$)(?![\_~@#\$\^]+$)[0-9A-Za-z\_~@#\$\^]{8,16}$数字、字母、特殊字符~@#$^任意两种组合以上