



链滴

知道这 20 个正则表达式，能让你少写 1,000 行代码

作者：[gentoo666](#)

原文链接：<https://ld246.com/article/1496842495552>

来源网站：[链滴](#)

许可协议：[署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

正则表达式，一个十分古老而又强大的文本处理工具，仅仅用一段非常简短的表达式语句，便能够快速实现一个非常复杂的业务逻辑。熟练地掌握正则表达式的话，能够使你的开发效率得到极大的提升。

正则表达式经常被用于字段或任意字符串的校验，如下面这段校验基本日期格式的JavaScript代码：

```
var reg = /^(\d{4})(-|\/)(\d{1,2})\2(\d{1,2})$/;
var r = fieldValue.match(reg);
if(r==null)alert('Date format error!');
```

下面是**技匠**整理的，在**前端开发**中经常使用到的**20**个正则表达式。

1. 校验密码强度

密码的强度必须是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}$
```

2. 校验中文

字符串仅能是中文。

```
^[\u4e00-\u9fa5]{0,}$
```

3. 由数字、26个英文字母或下划线组成的字符串

```
^\w+$
```

4. 校验E-Mail 地址

同密码一样，下面是E-mail地址合规性的正则检查语句。

```
[\w!#$%&'*/+/?^_{}~]+(?:\.[\w!#$%&'*/+/?^_{}~]+)*@(?:[\w](?:[\w-]*[\w])?\.)+[\w](?:[\w-]*[\w])?
```

5. 校验身份证号码

下面是身份证号码的正则校验。15 或 18位。

15位：

```
^[1-9]\d{7}((0\d)|(1[0-2]))((0|1|2)\d)|3[0-1]\d{3}$
```

18位：

```
^[1-9]\d{5}[1-9]\d{3}((0\d)|(1[0-2]))((0|1|2)\d)|3[0-1]\d{3}([0-9]|X)$
```

6. 校验日期

“yyyy-mm-dd ” 格式的日期校验，已考虑平闰年。

```
^(?!0000)[0-9]{4}-(?!0[1-9]|1[0-2])-(?!0[1-9]|1[0-9]|2[0-8])((?!0[13-9]|1[0-2])-(?!29|30)|(?!0[1578]|1[02])-(?!31)|(?!0[9]{2}(?!0[48]|2[468]|1[3579]|2[6])|(?!0[48]|2[468]|1[3579]|2[6])00)-(?!2-29))$
```

7. 校验金额

金额校验，精确到2位小数。

```
^[0-9]+(\.[0-9]{2})?$$
```

8. 校验手机号

下面是国内 13、15、18开头的手机号正则表达式。（可根据目前国内收集号扩展前两位开头号码）

```
^(13[0-9]|14[57]|15[01|2|3|5|6|7|8|9]|18[01|2|3|5|6|7|8|9])\d{8}$
```

9. 判断IE的版本

IE目前还没被完全取代，很多页面还是需要做版本兼容，下面是IE版本检查的表达式。

```
^.*MSIE [5-8](?!\.[0-9]+)?(?!.*Trident\\\[5-9]\\.)*$
```

10. 校验IP-v4地址

IP4 正则语句。

```
\\b(?:25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)\\.){3}(?:25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)\\b
```

11. 校验IP-v6地址

IP6 正则语句。

```
(([0-9a-fA-F]{1,4}:){7,7}[0-9a-fA-F]{1,4}|([0-9a-fA-F]{1,4}:){1,7}:|([0-9a-fA-F]{1,4}:){1,6}:[0-9a-fA-F]{1,4}|([0-9a-fA-F]{1,4}:){1,5}(:[0-9a-fA-F]{1,4}){1,2}|([0-9a-fA-F]{1,4}:){1,4}(:[0-9a-fA-F]{1,4}){1,3}|([0-9a-fA-F]{1,4}:){1,3}(:[0-9a-fA-F]{1,4}){1,4}|([0-9a-fA-F]{1,4}:){1,2}(:[0-9a-fA-F]{1,4}){1,5}|[0-9a-fA-F]{1,4}(:[0-9a-fA-F]{1,4}){1,6}):(:[0-9a-fA-F]{1,4}){1,7}|fe80:(:[0-9a-fA-F]{0,4}){0,4}%[0-9a-zA-Z]{1,}:|(ffff:0{1,4}){0,1}:(25[0-5]|(2[0-4]|1{0,1}[0-9])?{0,1}[0-9])\\.){3,3}(25[0-5]|(2[0-4]|1{0,1}[0-9])?{0,1}[0-9])\\.){3,3}(25[0-5]|(2[0-4]|1{0,1}[0-9])?{0,1}[0-9])\\.){3,3}(25[0-5]|(2[0-4]|1{0,1}[0-9])?{0,1}[0-9])
```

12. 检查URL的前缀

应用开发中很多时候需要区分请求是HTTPS还是HTTP，通过下面的表达式可以取出一个url的前缀然后再逻辑判断。

```
if (!s.match(/^([a-zA-Z]+:|\\|\\/)/))  
{  
    s = 'http://' + s;  
}
```

13. 提取URL链接

下面的这个表达式可以筛选出一段文本中的URL。

```
^(f|ht){1}(tp|tps):\\V\\([\\w-]+\\.)+[\\w-]+(\\V[\\w- ./?%&=]*)?
```

14. 文件路径及扩展名校验

验证windows下文件路径和扩展名（下面的例子中为.txt文件）

```
^[a-zA-Z\\:\\|\\\\\\\\\\\\\\\\([^\\\\]+\\\\\\\\)*[^\V:*?"<>]]+\\.txt(!)?$
```

15. 提取Color Hex Codes

有时需要抽取网页中的颜色代码，可以使用下面的表达式。

```
^#[A-Fa-f0-9]{6}[A-Fa-f0-9]{3}$
```

16. 提取网页图片

假若你想提取网页中所有图片信息，可以利用下面的表达式。

```
\\< *img][^\\\\>]*src *= *["\\']{0,1}([^\\\\"\\' >]*
```

17. 提取页面超链接

提取html中的超链接。

```
(<a\\s*(?!.*\\brel=)[^>]*)(href="https?:\\V\\)((?!(:?(:www\\.)?)'.implode('(?:www\\.)?', $follow_ist.)))[^"]+)((?!.*\\brel=)[^>]*)(?:[>]*>
```

18. 查找CSS属性

通过下面的表达式，可以搜索到相匹配的CSS属性。

```
^\\s*[a-zA-Z\\-]+\\s*[:]{1}\\s[a-zA-Z0-9\\s.#]+[:]{1}
```

19. 抽取注释

如果你需要移除HTML中的注释，可以使用如下的表达式。

```
<!--(.*?)-->
```

20. 匹配HTML标签

通过下面的表达式可以匹配出HTML中的标签属性。

```
<\\V?\\w+(\\s+\\w+(\\s*=\\s*(?:'.'?'"|'.'?'"|\\^">\\s+)))+\\s*|\\s*\\V?>
```