## 一、CSS hack

### 1. 什么是CSS hack？

由于不同厂商的流览器或某浏览器的不同版本（如IE6-IE11,Firefox/Safari/Opera/Chrome等），对CSS的支持、解析不一样，导致在不同浏览器的环境中呈现出不一致的页面展现效果。这时，我们为了获得统一的页面效果，就需要针对不同的浏览器或不同版本写特定的CSS样式，我们把这个针对不同的浏览器/不同版本写相应的CSS code的过程，叫做CSS hack!

### 2. CSS hack的原理

由于不同的浏览器和浏览器各版本对CSS的支持及解析结果不一样，以及CSS优先级对浏览器展现效果的影响，我们可以据此针对不同的浏览器情景来应用不同的CSS。

### 3. CSS hack分类

CSS Hack大致有3种表现形式，**CSS属性前缀法**、**选择器前缀法**以及**IE条件注释法**（即HTML头部引用if IE）Hack，实际项目中CSS Hack大部分是针对IE浏览器不同版本之间的表现差异而引入的。

属性前缀法（即类内部Hack）：例如 IE6能识别下划线"\_"和星号" \* "，IE7能识别星号" \* "，但不能识别下划线"\_"，IE6~IE10都认识"\9"，但firefox前述三个都不能认识。

选择器前缀法（即选择器Hack）：例如 IE6能识别\*html .class{}，IE7能识别\*+html .class{}或者\*:first-child+html .class{}。

IE条件注释法（即HTML条件注释Hack）：针对所有IE(注：IE10+已经不再支持条件注释)： <!--[if IE]>IE浏览器显示的内容 <![endif]-->，针对IE6及以下版本： <!--[if lt IE 6]>只在IE6-显示的内容 <![endif]-->。这类Hack不仅对CSS生效，对写在判断语句里面的所有代码都会生效。

注意：**CSS hack书写顺序，一般是将适用范围广、被识别能力强的CSS定义在前面。**

### 4. CSS hack的实现

#### 4.1 CSS hack方式一：条件注释法

这种方式是IE浏览器专有的Hack方式，微软官方推荐使用的hack方式。举例如下：

只在IE下生效

<!--[if IE]>

这段文字只在IE浏览器显示

<![endif]-->

只在IE6下生效

<!--[if IE 6]>

这段文字只在IE6浏览器显示

<![endif]-->

只在IE6以上版本生效

<!--[if gte IE 6]>

这段文字只在IE6以上(包括)版本IE浏览器显示

<![endif]-->

只在IE8上不生效

<!--[if ! IE 8]>

这段文字在非IE8浏览器显示

<![endif]-->

#### 4.2 CSS hack方式二：CSS属性前缀法

属性前缀法是在CSS样式属性名前加上一些只有特定浏览器才能识别的hack前缀，以达到预期的页面展现效果。

IE浏览器各版本 CSS hack 对照表

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| hack | 写法 | IE6(S) | IE6(Q) | IE7(S) | IE7(Q) | IE8(S) | IE8(Q) | IE9(S) | IE9(Q) | IE10(S) | IE10(Q) |
| \* | \*color | Y | Y | Y | Y | N | Y | N | Y | N | Y |
| + | +color | Y | Y | Y | Y | N | Y | N | Y | N | Y |
| - | -color | Y | Y | N | N | N | N | N | N | N | N |
| \_ | \_color | Y | Y | N | Y | N | Y | N | Y | N | Y |
| # | #color | Y | Y | Y | Y | N | Y | N | Y | N | Y |
| \0 | color:red\0 | N | N | N | N | Y | N | Y | N | Y | N |
| \9\0 | color:red\9\0 | N | N | N | N | N | N | Y | N | Y | N |
| !important | color:blue !important;color:green; | N | N | Y | N | Y | N | Y | N | Y | N |

其中，S表示Standards Mode，即标准模式；Q表示Quirks Mode，即兼容模式。

说明：在标准模式中

* “-″减号是IE6专有的hack
* “\9″ IE6/IE7/IE8/IE9/IE10都生效
* “\0″ IE8/IE9/IE10都生效，是IE8/9/10的hack
* “\9\0″ 只对IE9/IE10生效，是IE9/10的hack

demo如下

1. <style type="text/css">
2. body:nth-of-type(1) .iehack{
3. **color**: #F00;/\* 对Windows IE9/Firefox 7+/Opera 10+/所有Chrome/Safari的CSS hack ，选择器也适用几乎全部Mobile/Linux/Mac browser\*/
4. }
5. .demo1,.demo2,.demo3,.demo4{
6. **width**:100px;
7. **height**:100px;
8. }
9. .hack{
10. /\*demo1 注意顺序，否则IE6/7下可能无法正确显示，导致结果显示为白色背景\*/
11. **background-color**:red; /\* All browsers \*/
12. **background-color**:blue !important;/\* All browsers but IE6 \*/
13. \***background-color**:black; /\* IE6, IE7 \*/
14. +**background-color**:yellow;/\* IE6, IE7\*/
15. **background-color**:gray\9; /\* IE6, IE7, IE8, IE9, IE10 \*/
16. **background-color**:purple\0; /\* IE8, IE9, IE10 \*/
17. **background-color**:orange\9\0;/\*IE9, IE10\*/
18. **\_background-color**:green; /\* Only works in IE6 \*/
19. \*+**background-color**:pink;
20. }
22. /\*可以通过javascript检测IE10，然后给IE10的<html>标签加上class=”ie10″ 这个类 \*/
23. .ie10 #hack{
24. **color**:red; /\* Only works in IE10 \*/
25. }
27. /\*demo2\*/
28. .iehack{
29. /\*该demo实例是用于区分标准模式下ie6~ie9和Firefox/Chrome的hack，注意顺序
30. IE6显示为：绿色，
31. IE7显示为：黑色，
32. IE8显示为：红色，
33. IE9显示为：蓝色，
34. Firefox/Chrome显示为：橘色，
35. （本例IE10效果同IE9,Opera最新版效果同IE8）
36. \*/
37. **background-color**:orange; /\* all - for Firefox/Chrome \*/
38. **background-color**:red\0; /\* ie 8/9/10/Opera - for ie8/ie10/Opera \*/
39. **background-color**:blue\9\0; /\* ie 9/10 - for ie9/10 \*/
40. \***background-color**:black; /\* ie 6/7 - for ie7 \*/
41. **\_background-color**:green; /\* ie 6 - for ie6 \*/
42. }
44. /\*demo3
45. 实例是用于区分标准模式下ie6~ie9和Firefox/Chrome的hack，注意顺序
46. IE6显示为：红色，
47. IE7显示为：蓝色，
48. IE8显示为：绿色，
49. IE9显示为：粉色，
50. Firefox/Chrome显示为：橘色，
51. （本例IE10效果同IE9，Opera最新版效果也同IE9为粉色）
52. \*/
53. .element {
54. **background-color**:orange; /\* all IE/FF/CH/OP\*/
55. }
56. .element {
57. \***background-color**: blue; /\* IE6+7, doesn't work in IE8/9 as IE7 \*/
58. }
59. .element {
60. **\_background-color**: red; /\* IE6 \*/
61. }
62. .element {
63. **background-color**: green\0; /\* IE8+9+10 \*/
64. }
65. :root .element { **background-color**:pink\0; } /\* IE9+10 \*/
67. /\*demo4\*/
68. /\*
70. 该实例是用于区分标准模式下ie6~ie10和Opera/Firefox/Chrome的hack，本例特别要注意顺序
71. IE6显示为：橘色，
72. IE7显示为：粉色，
73. IE8显示为：黄色，
74. IE9显示为：紫色，
75. IE10显示为：绿色，
76. Firefox显示为：蓝色，
77. Opera显示为：黑色，
78. Safari/Chrome显示为：灰色，
80. \*/
81. .hacktest{
82. **background-color**:blue; /\* 都识别，此处针对firefox \*/
83. **background-color**:red\9; /\*all ie\*/
84. **background-color**:yellow\0; /\*for IE8/IE9/10 最新版opera也认识\*/
85. +**background-color**:pink; /\*for ie6/7\*/
86. **\_background-color**:orange; /\*for ie6\*/
87. }
89. @media screen and (**min-width**:0){
90. .hacktest {**background-color**:black\0;} /\*opera\*/
91. }
92. @media screen and (**min-width**:0) {
93. .hacktest { **background-color**:purple\9; }/\* for IE9/IE10 PS:国外有些习惯常写作\0，根本没考虑Opera也认识\0的实际 \*/
94. }
95. @media screen and (-ms-high-contrast: active), (-ms-high-contrast: none) {
96. .hacktest { **background-color**:green; } /\* for IE10+ 此写法可以适配到高对比度和默认模式，故可覆盖所有ie10的模式 \*/
97. }
98. @media screen and (-webkit-min-device-pixel-ratio:0){ .hacktest {**background-color**:gray;} } /\*for Chrome/Safari\*/
100. /\* #963棕色 :root is for IE9/IE10, 优先级高于@media, 慎用！如果二者合用，必要时在@media样式加入 !important 才能区分IE9和IE10 \*/
101. /\*
102. :root .hacktest { background-color:#963\9; }
103. \*/
104. </style>

demo1是测试不同IE浏览器下hack 的显示效果  
IE6显示为：粉色，  
IE7显示为：粉色，  
IE8显示为：蓝色，  
IE9显示为：蓝色，  
Firefox/Chrome/Opera显示为：蓝色，  
若去掉其中的!important属性定义，则IE6/7仍然是粉色，IE8是紫色，IE9/10为橙色，Firefox/Chrome变为红色，Opera是紫色。是不是有些奇怪：除了IE6以外，其他所有的表现都符合我们的期待。那为何IE6表现的颜色不是\_background-color:green;的绿色而是\*+background-color:pink的粉色呢？其实是最后一句所谓的IE7私有hack惹的祸？不是说\*+是IE7的专有hack吗？？？错，你可能太粗心了！我们常说的IE7专有\*+hack的格式是\*+html selector，而不是上面的直接在属性上加\*+前缀。如果是为IE7定制特殊样式，应该这样使用：

\*+html #ie7test { /\* IE7 only\*/

color:green;

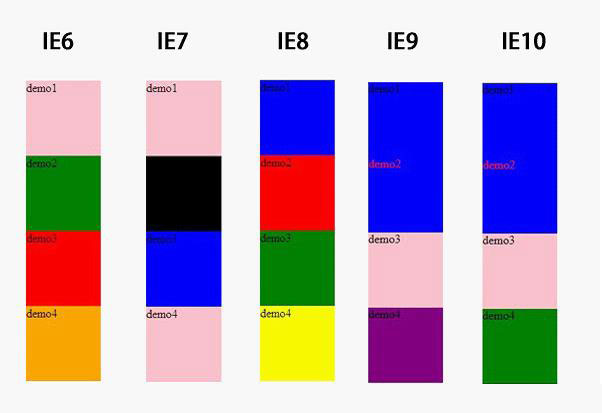
}

经过测试，发现属性前缀\*+background-color:pink;只有IE6和IE7认识。而\*+html selector只有IE7认识。所以我们在使用时候一定要特别注意。

demo2实例是用于区分标准模式下ie6~ie9和Firefox/Chrome的hack，注意顺序  
IE6显示为：绿色，  
IE7显示为：黑色，  
IE8显示为：红色，  
IE9显示为：蓝色，  
Firefox/Chrome显示为：橘色，  
（本例IE10效果同IE9,Opera最新版效果同IE8）

demo3实例也是用于区分标准模式下ie6~ie9和Firefox/Chrome的hack，注意顺序  
IE6显示为：红色，  
IE7显示为：蓝色，  
IE8显示为：绿色，  
IE9显示为：粉色，  
Firefox/Chrome显示为：橘色，  
（本例IE10效果同IE9，Opera最新版效果也同IE9为粉色）

demo4实例是用于区分标准模式下ie6~ie10和Opera/Firefox/Chrome的hack，本例特别要注意顺序  
IE6显示为：橘色，  
IE7显示为：粉色，  
IE8显示为：黄色，  
IE9显示为：紫色，  
IE10显示为：绿色，  
Firefox显示为：蓝色，  
Opera显示为：黑色，  
Safari/Chrome显示为：灰色，



#### 4.3 CSS hack方式三：选择器前缀法

选择器前缀法是针对一些页面表现不一致或者需要特殊对待的浏览器，在CSS选择器前加上一些只有某些特定浏览器才能识别的前缀进行hack。

目前最常见的是

\*html \*前缀只对IE6生效

\*+html \*+前缀只对IE7生效

@media screen\9{...}只对IE6/7生效

@media \0screen {body { background: red; }}只对IE8有效

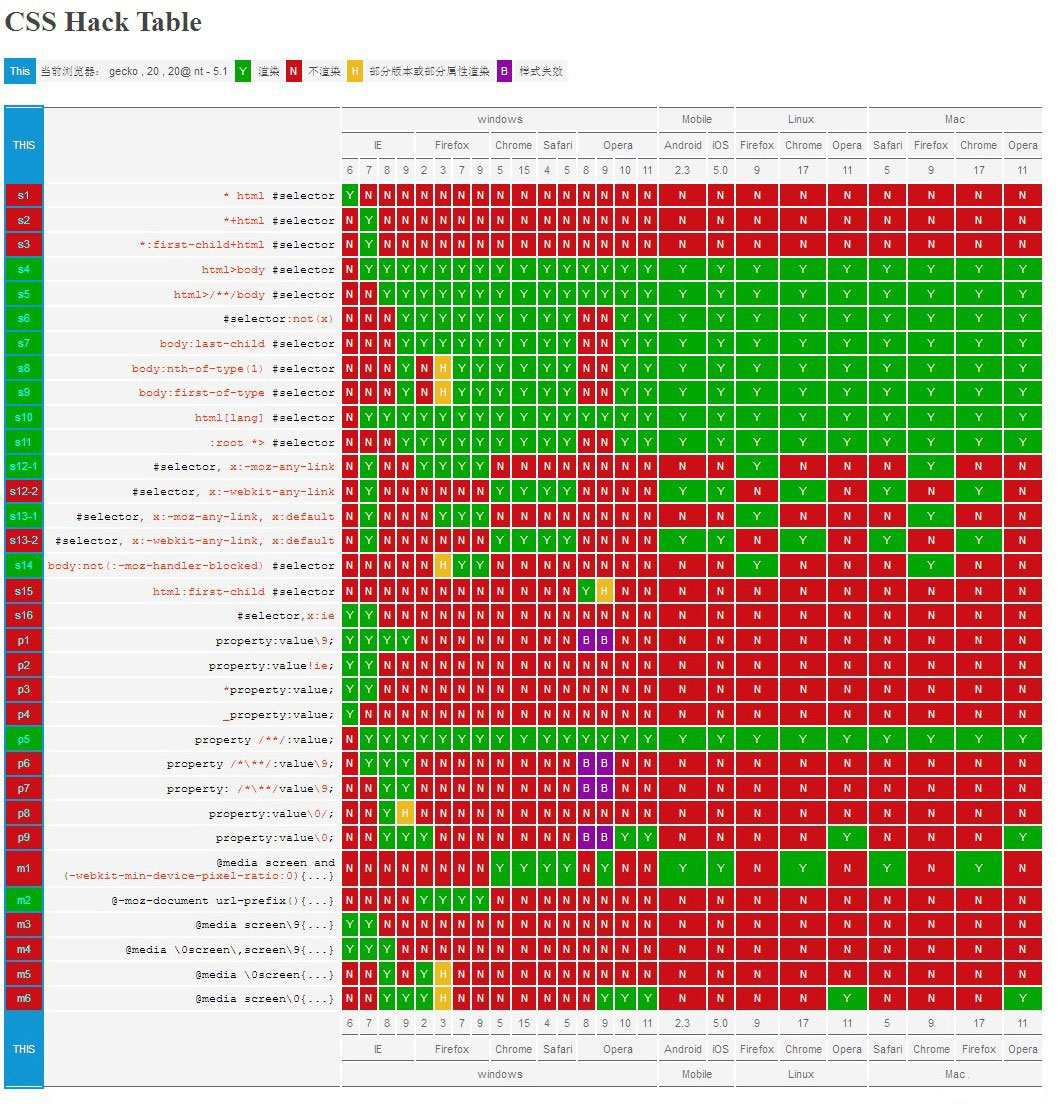
@media \0screen\,screen\9{body { background: blue; }}只对IE6/7/8有效

@media screen\0 {body { background: green; }} 只对IE8/9/10有效

@media screen and (min-width:0\0) {body { background: gray; }} 只对IE9/10有效

@media screen and (-ms-high-contrast: active), (-ms-high-contrast: none) {body { background: orange; }} 只对IE10有效

结合CSS3的一些选择器，如html:first-child，body:nth-of-type(1)，衍生出更多的hack方式，具体的可以参考下表：



### 5. CSS hack利弊

一般情况下，我们**尽量避免使用CSS hack**，但是有些情况为了顾及用户体验实现向下兼容，不得已才使用hack。使用hack虽然对页面表现的一致性有好处，但过多的滥用会造成html文档混乱不堪，增加管理和维护的负担。相信只要大家一起努力，少用、慎用hack，未来一定会促使浏览器厂商的标准越来越趋于统一，顺利过渡到标准浏览器的主流时代。抛弃那些陈旧的IE hack，必将减轻我们编码的复杂度，少做无用功。

### CSS3识别浏览器

按内核分类：

Trident：又称为MSHTML，代表Internet Explorer

其他：Maxthon遨游、世界之窗、腾讯TT、Netcapter、Avant等等，但Trident只能应用于Windows平台，且是不开源的

Gecko：代表作品Mozilla Firefox Netscape6至9

能在Microsoft Windows、Linux和MacOS X等主要操作系统上运行

WebKit：代表作品Safari、Chrome

Presto：代表作品Opera

有些CSS3的属性在高级浏览器里面是支持的，但是使用标准写法又不能实现，需要添加浏览器识别，也可以认为是标准化的HACK：

|  |  |
| --- | --- |
| **内核类型** | **写法** |
| **Webkit(Chrome/Safari)** | -webkit-属性 |
| **Gecko(Firefox)** | -moz-属性 |
| **Presto(Opera)** | -o-属性 |
| **Trident(IE)** | -ms-属性 |
| **W3C** | 属性 |

例如设置对象的2D属性：

|  |
| --- |
| transform:rotate(30deg);  -webkit-transform:rotate(30deg);  -moz-transform:rotate(30deg);  -o-transform:rotate(30deg);  -ms-transform:rotate(30deg); |

技巧：查看CSS属性有哪些浏览器支持： http://www.caniuse.com

## 二、常用CSS样式

### 1. CSS制作三角符号

|  |
| --- |
| <!DOCTYPE html>  <html lang="zh-CN">  <head>  <meta charset="utf-8"/>  <title>标题</title>  <style type="text/css">  h3{width:200px;background:#ccc;overflow:hidden;}  h3 span{  width:0;height:0;overflow:hidden;  border-width:8px;  border-color:#000 transparent transparent;  border-style:solid;  float:left;  margin:6px 0 0 0;  }  h3 strong{float:left;}  </style>  </head>  <body>  <h3><strong>产品分类</strong><span></span></h3>  </body>  </html> |

### 2. CSS 精灵（sprites）

|  |
| --- |
| <!DOCTYPE html>  <html lang="zh-CN">  <head>  <meta charset="utf-8"/>  <title>CSS精灵</title>  <style type="text/css">  /\*  CSS精灵：利用一张图片实现多个背景图片的显示效果  使用：减少图片数量，提高加载速度，增强服务器性能  作法：  1. 将页面上会用到的小图标放置在一个图片文件中，构造一个包含很多小图标的图片文件。  2. 利用背景图的位置控制，实现将不同的小图标显示在对应的位置上。  \*/  h3{background:url(images/icons.png) no-repeat;padding-left:20px;border-bottom:1px solid #ccc;}  .product{background-position:left -1192px;}  .share{background-position:left -68px;}  .book{background-position:left -104px;}  </style>  </head>  <body>  <h3 class="product">产品分类</h3>  <p>此处省略一万字...</p>  <h3 class="share">连接分享</h3>  <p>此处省略一万字...</p>  <h3 class="book">留言板</h3>  <p>此处省略一万字...</p>  </body>  </html> |

### 3. CSS的定位技巧（子绝父相）

|  |
| --- |
| <!DOCTYPE html>  <html lang="zh-CN">  <head>  <meta charset="utf-8"/>  <title>定位</title>  <style type="text/css">  html,body{margin:0;padding:0;}  .searchform{width:480px;margin:10px auto;padding:0;position:relative;}  .keywords{border:2px solid #888;height:34px;position:absolute;z-index:1;width:376px; background:url(images/input\_bg.jpg) no-repeat 350px center;font:24px/34px arial;}  .search\_btn{position:absolute;z-index:2;left:376px;}  </style>  </head>  <body>  <form action="" method="post" class="searchform">  <input type="text" name="keywords" class="keywords"/>  <input type="image" src="images/search\_btn.jpg" title="搜索" class="search\_btn"/>  </form>  </body>  </html> |

Input\_bg.jpg input_bg search\_btn.jpg search_btn

### 4. CSS布局（两列，三列）

|  |
| --- |
| <!DOCTYPE html>  <html lang="zh-CN">  <head>  <meta charset="utf-8"/>  <title>布局</title>  <style type="text/css">  html,body{margin:0;padding:0;}  /\* 左右布局 \*/  #content1{width:1000px;margin:10px auto;overflow:hidden;}  /\*  #content1\_left{width:200px;height:100px;float:left;background:#0a0;}  #content1\_right{width:790px;height:100px;float:right;background:#0a0;}  \*/  #content1\_left{width:790px;height:100px;float:left;background:#0a0;}  #content1\_right{width:200px;height:100px;float:right;background:#0a0;}  /\* 左中右布局 \*/  #content2{width:1000px;margin:10px auto;}  #content2\_left,#content2\_right{width:200px;height:200px;float:left;background:#f0f;}  #content2\_left{margin-right:10px;padding:10px 20px;width:160px;}  #content2\_right{float:right;}  #content2\_main{background:#f00;height:200px;float:left;width:580px;}  </style>  </head>  <body>  <!--左右布局-->  <div id="content1">  <h3>左右布局</h3>  <div id="content1\_left">  左边主体  </div>  <div id="content1\_right">  右边栏  </div>  </div>  <!--左中右布局-->  <div id="content2">  <h3>左中右布局</h3>  <div id="content2\_left">  左边栏  </div>  <div id="content2\_main">  中间主体  </div>  <div id="content2\_right">  右边栏  </div>  </div>  </body>  </html> |

### 5. CSS2实现圆角矩形

|  |
| --- |
| <!DOCTYPE html>  <html lang="zh-CN">  <head>  <meta charset="utf-8"/>  <title>布局</title>  <style type="text/css">  html,body{margin:0;padding:0;}  /\* CSS2实现圆角矩形 \*/  #box{background:#ee4315;width:300px;margin:20px auto;overflow:hidden;}  #box span{background:url(images/1.png) no-repeat;width:10px;height:10px;}  #box .lt{float:left;}  #box .rt{float:right;background-position:-10px 0;}  #box .lb{float:left;background-position:0 -10px;}  #box .rb{float:right;background-position:-10px -10px;}  .mainbox{clear:both;}  </style>  </head>  <body>  <div id="box">  <span class="lt">&nbsp;</span>  <span class="rt">&nbsp;</span>  <div class="mainbox"> box1box1box1box1<br/>box1box1box1box1<br/>box1box1box1box1<br/>box1box1box1box1<br/>box1box1box1box1<br/>  </div>  <span class="lb">&nbsp;</span>  <span class="rb">&nbsp;</span>  </div>  </body>  </html> |

## 三、CSS常见事项

1. ul 标签在 Mozilla 中默认是有 padding 值的，而在 IE 中只有 margin 值。
2. 同一个的 class 选择符可以在一个文档中重复出现，而 id 选择符却只能出现一次；对一个标签同时使用 class和 id 进行 CSS 定义，如果定义有重复，id 选择符做的定义有效。
3. 如果一组要嵌套的标签之间需要些间距的话，那就外面标签的 padding 属性，而不要去定义位于里面的标签的 margin
4. li 标签前面的图标推荐使用 background-image 而不是 list-style-image
5. 在给你的标签疯狂加选择符的时候，别忘了在 CSS 里给选择符加上注释。 等你以后修改你的 CSS 的时候才知道每个选择符代表的意思。
6. 如果你给一个标签设置了一个深色调的背景图片和亮色调的文字效果。建议这个时候给你的标签再设置一个深色调的背景颜色。
7. 定义颜色可以缩写 #8899FF = #89F
8. 完美的单象素外框线表格 table {border-collapse:collapse;} td {border:#000 solid 1px;}
9. 固定宽度汉字截断：overflow:hidden;text-overflow:ellipsis;white-space:nowrap;（不过只能处理文字在一行上的截断，不能处理多行。）
10. 固定宽度汉字（词）折行：table-layout:fixed; word-break:break-all;
11. <div style="cursor: help" title="输入要提示的文字">文字</div> 用鼠标放在前面的文字上看效果。这个效果在国外的很多网站都可以看到，而国内的少又少。
12. 图片设为半透明：opacity:0.5;}
13. 我们在做网页时常用到把鼠标放在图片上会出现图片变亮的效果，可以用图片替换的技巧，也可以用如下的滤镜，代码如下：

.pictures img { filter: alpha(opacity=50); opacity:0.5;}

.pictures img:hover { filter: alpha(opacity=100); opacity:1;}

## 四、CSS常用命名

头部：header

内容：content/container

尾：footer

导航：nav

侧栏：sidebar

栏目：column

页面外围控制整体布局宽度：wrapper

左右中：left right center

登录条：loginbar

标志：logo

广告：banner

页面主体：main

热点：hot

新闻：news

下载：download

子导航：subnav

菜单：menu

子菜单：submenu

搜索：search

友情链接：friendlink

页脚：footer

版权：copyright

滚动：scroll

标签页：tab

文章列表：list

提示信息：msg

小技巧：tips

栏目标题：title

加入：joinus

指南：guild

服务：service

注册：regsiter

状态态：status

投票：vote

合作伙伴：partner

### 1. 页面结构常用命名

容器: container

页头：header

内容：content/container

页面主体：main

页尾：footer

导航：nav

侧栏：sidebar

栏目：column

页面外围控制整体布局宽度：wrapper

左右中：left right center

### 2. 导航命名

导航：nav

主导航：mainbav

子导航：subnav

顶导航：topnav

边导航：sidebar

左导航：leftsidebar

右导航：rightsidebar

菜单：menu

子菜单：submenu

标题: title

摘要: summary

### 3. 功能命名

标志：logo

广告：banner

登陆：login

登录条：loginbar

注册：regsiter

搜索：search

功能区：shop

标题：title

加入：joinus

关于我们：aboutus

状态：status

按钮：btn

滚动：scroll

标签页：tab

文章列表：list

提示信息：msg

当前的: current

小技巧：tips

图标: icon

注释：note

指南：guild

服务：service

热点：hot

新闻：news

下载：download

投票：vote

合作伙伴：partner

友情链接：link

版权：copyright

### 4. class的常用命名

1）颜色:使用颜色的名称或者16进制代码,如

.red { color: red; }

.f60 { color: #f60; }

.ff8600 { color: #ff8600; }

2）字体大小,直接使用"font+字体大小"作为名称,如

.font12px { font-size: 12px; }

.font9pt {font-size: 9pt; }

3）对齐样式,使用对齐目标的英文名称,如

.left { float:left; }

.bottom { float:bottom; }

4）标题栏样式,使用"类别+功能"的方式命名,如

.barnews { }

.barproduct { }

### 5. CSS文件常用命名

首页： index.css

主要的： master.css main.css

模块： module.css

基本共用： base.css

布局，版面： layout.css

主题： themes.css

专栏： columns.css

文字： font.css

表单： forms.css

补丁： mend.css

打印： print.css

### 6. 命名注意事项

1）一律小写;

2）尽量用英文;

3）不加中杠和下划线等一些特殊的符号;

4）尽量不缩写，除非一看就明白的单词。