

# 你所不知的 CSS before 和 after 伪元素用法 - Just Code - ITeye博客

CSS 有两个说不上常用的伪类 :before 和 :after, 偶尔会被人用来添加些自定义格式什么的, 但是它们的功用不仅于此。前几天发现了 [Creative Link Effects](#) 这个非常有意思的介绍创意链接特效的页面, 里面惊人的效果大量使用到的特性除了 transform 属性进行变形之外, 就是接下来要介绍的这两个伪元素了。



Creative Button Styles

## 一 基本语法

在了解进阶的应用之前, 先来了解一下语法规则。平常仅仅需要将这两个伪元素用于添加一些自定义字符时, 只需使用伪类使用的单冒号写法, 以保证浏览器的兼容性:

Html代码

```
1. p:before {}
```

不过, 在 CSS3 中为了区别伪元素和伪类为伪元素使用了双冒号, 因此如果使用了 display 或者 width 等属性时使得显示脱离了原本元素后, 建议按照标准双写。过于老的浏览器可能会存在支持问题, 不过伪元素大多是配合 CSS3 使用, 就无所谓向下兼容了:

Html代码

```
1. img::after {}
```

这两个伪类下特有的属性 content, 用于在 CSS 渲染中向元素逻辑上的头部或尾部添加内容。注意这些添加不会改变文档内容, 不会出现在 DOM 中, 不可复制, 仅仅是在 CSS 渲染层加入。比较有用的是以下几个值:

- **[String]** - 使用引号包括一段字符串, 将会向元素内容中添加字符串。示例:

Html代码

```
a. a:after { content: "↗"; }
```

- **attr()** - 调用当前元素的属性, 可以方便的比如将图片的 Alt 提示文字或者链接的 Href 地址显示出来。示例:

Html代码

```
a. a:after { content: "(" attr(href) "; " }
```

- **url() / uri()** - 用于引用媒体文件。示例:

Html代码

```
a. h1::before { content: url(logo.png); }
```

- **counter()** - 调用计数器, 可以不使用列表元素实现序号功能。具体请参见 counter-increment 和 counter-reset 属性的用法。示例:

Html代码

```
a. h2:before { counter-increment: chapter; content: "Chapter "
counter(chapter) ". " }
```

## 二 进阶技巧

清除浮动是一个时常会遇到的问题，不少人的解决办法是添加一个空的 div 应用 `clear:both;` 属性。现在，无需增加没有意义的元素，仅需要以下样式即可在元素尾部自动清除浮动：

**Html代码**

```
1. .clear-fix { *overflow: hidden; *zoom: 1; }
2. .clear-fix:after { display: table; content: ""; width: 0; clear: both; }
```

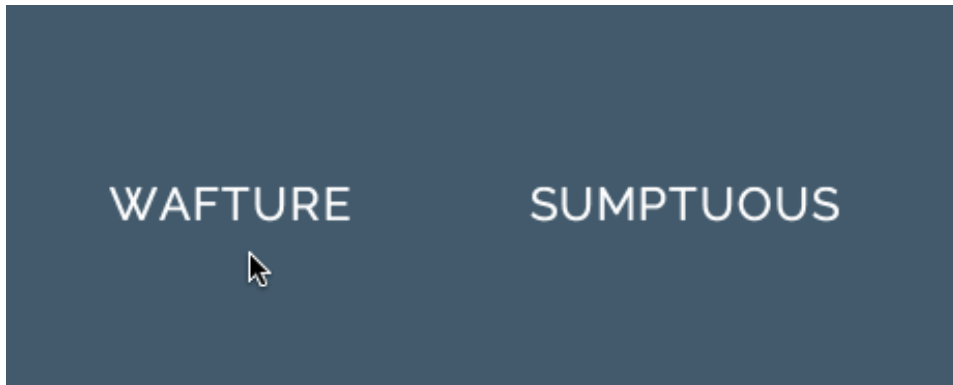
许多人喜欢给 blockquote 引用段添加巨大的引号作为背景，这种时候我们就可以用 `:before` 来代替 `background` 了，即可以给背景留下空间，还可以直接使用文字而非图片：

**Html代码**

```
1. blockquote::before {
2.     content: open-quote;
3.     position: absolute;
4.     z-index: -1;
5.     color: #DDD;
6.     font-size: 120px;
7.     font-family: serif;
8.     font-weight: bolder;
9. }
```

## 三 特效妙用

除了简单的添加字符，配合 CSS 强大的定位和特效特性，完全可以达到给简单的元素另外附加最多两个容器的效果。有一点需要注意的是，如果不需要内容仅配合样式属性做出效果，内容属性也不能为空，即 `content: ""`。否则，其他的样式属性一概不会生效。



悬浮出现方括号

鼠标移上链接，出现方括号：

**Html代码**

```
1. a {
2.     position: relative;
3.     display: inline-block;
4.     outline: none;
5.     text-decoration: none;
6.     color: #000;
7.     font-size: 32px;
8.     padding: 5px 10px;
9. }
10. a:hover::before, a:hover::after { position: absolute; }
11. a:hover::before { content: "\5B"; left: -20px; }
12. a:hover::after { content: "\5D"; right: -20px; }
```

同样，我们只需要配合 `display: block` 和 `position: absolute`，就可以将其当成两个容器，拼合成悬浮出现双边框的特效：

**Html代码**

```
1. a {
```

```
2.     position: relative;
3.     display: inline-block;
4.     outline: none;
5.     text-decoration: none;
6.     color: #000;
7.     font-size: 32px;
8.     padding: 0 10px;
9. }
10. /* 大框 */
11. a:hover::before, a:hover::after {
12.     content: "";
13.     display: block;
14.     position: absolute;
15.     top: -15%;
16.     left: -14%;
17.     width: 120%;
18.     height: 120%;
19.     border-style: solid;
20.     border-width: 4px;
21.     border-color: #DDD;
22. }
23. /* 小框 */
24. a:hover::after {
25.     top: 0%;
26.     left: 0%;
27.     width: 100%;
28.     height: 100%;
29.     border-width: 2px;
30. }
```

用 :before 和 :after 伪类结合更多 CSS3 强大的特性，还可以完成非常多有意思的特效和 Hack，这里权当抛砖引玉，诸位又能想出什么妙用呢？

原文：<http://blog.dimpurr.com/css-before-after/>