



CSS3 新选择器

属性选择器 权重 0,0,1,0

直接用属性来选择

```
input[value] {} 包含value属性的input标签
```

重点：

```
input[type="password"] {} 包含type属性且属性值为password的input标签
```

其他：

选择符	简介
E[att]	选择具有 att 属性的 E 元素
E[att="val"]	选择具有 att 属性且属性值等于 val 的 E 元素
E[att^="val"]	匹配具有 att 属性且值以 val 开头的 E 元素
E[att\$="val"]	匹配具有 att 属性且值以 val 结尾的 E 元素
E[att*="val"]	匹配具有 att 属性且值中含有 val 的 E 元素

结构伪类选择器

选择符	简介
E:first-child	匹配父元素中的第一个子元素 E
E:last-child	匹配父元素中最后一个 E 元素
E:nth-child(n)	匹配父元素中的第 n 个子元素E
E:first-of-type	指定类型 E 的第一个
E:last-of-type	指定类型 E 的最后一个
E:nth-of-type(n)	指定类型 E 的第 n 个

: 就是伪类的标志

E:first-child全部排序然后选择

E:?-child (n) 要同时满足n和E

```
ul li:first-child;ul下面的第一个li
ul li:last-child;ul下面的最后一个li
ul li:nth-child(n);ul下面的第n个li
```

牛逼的是：n可以是公式或者关键字

nth-child(even)：偶数

nth-child(odd)：奇数

迭代选法（n是从零开始的）

n

```
ul li:nth-child(n);会选择所有ul里面的li
注意这个n一定只能是"n"
```

an

```
ul li:nth-child(2n);选择2的倍数个
ul li:nth-child(even);两个式子等价
```

n+a

```
ul li:nth-child(n+2);从第二个开始（包含）
```

```
ul li:nth-child(-n+2);前面两个（包含）
```

E:nth-of-type(n)指定排序然后选择

排序的是E标签的所有，选择E标签中的第n个

```
div:nth-of-type(2);排序所有的div标签然后选择第二个
```

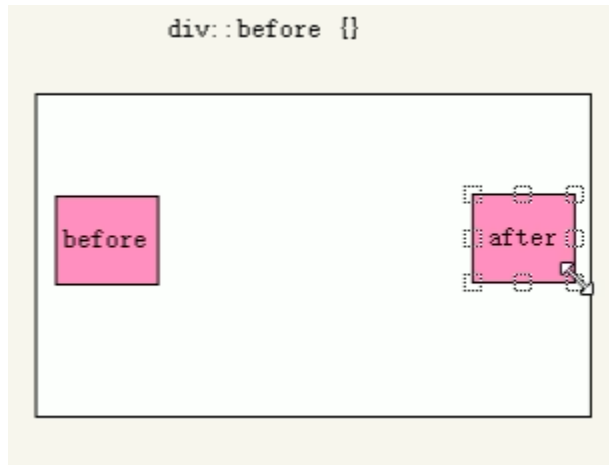
伪元素选择器 权重0,0,0,2

直接利用css创建一个标签,属于行内元素但是实际上在文档中是不存在的

```
::before  
::after
```

语法规则：

```
div::after {  
  content: "";这个是必须要有的元素  
}
```



在element父元素里面的前面添加一个新的盒子before，同理添加after

伪元素选择器使用场景

创建字体图标

```
div::after{
  position: absolute;
  top: 10px;
  right: 10px;
  font-family: 'icomoon';
  content: '复制的特殊符号 或者 \加编码';
  color: red;
  font-size: 18px;
}
```

防土豆效果添加遮罩

```
.tudou:hover::before {
  display: block;
}
```

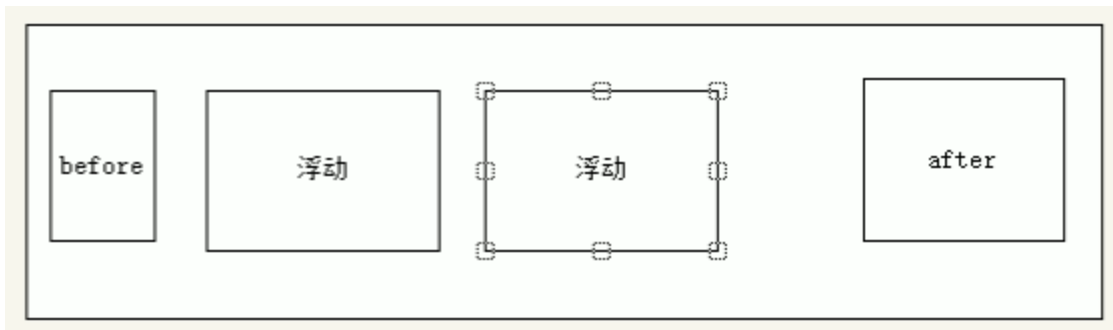
伪元素清除浮动

额外标签法，伪元素法，双伪元素法，

伪元素法

```
.clearfix:after {  
  content: "";  
  display: block;块级元素  
  height:0;  
  clear: both; 核心代码  
  visibility: hidden ; 不要显示  
}
```

双伪元素法



前后都有，单伪元素是只有后面

```
.clearfix:before, .clearfix:after {  
  content: "";  
  display: table ;转化为block块级元素并且一行显示  
}  
.clearfix:after {  
  clear: both;  
}
```