[CSS-清除浮动](https://segmentfault.com/a/1190000004865198)

在非IE浏览器（如Firefox）下，当容器的高度为auto，且容器的内容中有浮动（float为left或right）的元素，在这种情况下，容器的高度不能自动伸长以适应内容的高度，使得内容溢出到容器外面而影响（甚至破坏）布局的现象。这个现象叫浮动溢出，为了防止这个现象的出现而进行的CSS处理，就叫CSS清除浮动。

**引用W3C的例子，news容器没有包围浮动的元素。**

.news {

background-color: gray;

border: solid 1px black;

}

.news **img** {

float: left;

}

.news **p** {

float: right;

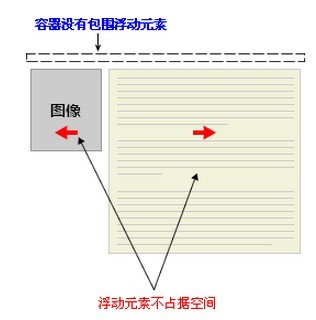
}

<**div** class="news">

<**img** src="news-pic.jpg" />

<p>some text</p>

</div>

;

## 清除浮动方法

**方法一：使用带clear属性的空元素**

在浮动元素后使用一个空元素如<div class="clear"></div>，并在CSS中赋予.clear{clear:both;}属性即可清理浮动。亦可使用<br class="clear" />或<hr class="clear" />来进行清理。

.news {

background-color: gray;

border: solid 1px black;

}

.news **img** {

float: left;

}

.news **p** {

float: right;

}

.clear {

clear: both;

}

<**div** class="news">

<**img** src="news-pic.jpg" />

<p>some text</p>

<**div** class="clear"></div>

</div>

**优点：简单，代码少，浏览器兼容性好。**  
**缺点：需要添加大量无语义的html元素，代码不够优雅，后期不容易维护。**

**方法二：使用CSS的overflow属性**

给浮动元素的容器添加overflow:hidden;或overflow:auto;可以清除浮动，另外在 IE6 中还需要触发 hasLayout ，例如为父元素设置容器宽高或设置 zoom:1。在添加overflow属性后，浮动元素又回到了容器层，把容器高度撑起，达到了清理浮动的效果。

.news {

background-color: gray;

border: solid 1px black;

overflow: hidden;

\*zoom: 1;

}

.news img {

float: left;

}

.news p {

float: right;

}

<div class="news">

<img src="news-pic.jpg" />

<p>some text</p>

</div>

**方法三：给浮动的元素的容器添加浮动**

给浮动元素的容器也添加上浮动属性即可清除内部浮动，但是这样会使其整体浮动，影响布局，不推荐使用。

**方法四：使用邻接元素处理**

什么都不做，给浮动元素后面的元素添加clear属性。

.news {

background-color: gray;

border: solid 1px black;

}

.news **img** {

float: left;

}

.news **p** {

float: right;

}

.content{

clear:both;

}

<**div** class="news">

<**img** src="news-pic.jpg" />

<p>some text</p>

<**div** class="content">\*\*\*</div>

</div>

注意这里的div.content有内容。

**方法五：使用CSS的:after伪元素**

结合 :after 伪元素（注意这不是伪类，而是伪元素，代表一个元素之后最近的元素）和 IEhack ，可以完美兼容当前主流的各大浏览器，这里的 IEhack 指的是触发 hasLayout。  
给浮动元素的容器添加一个clearfix的class，然后给这个class添加一个:after伪元素实现元素末尾添加一个看不见的块元素（Block element）清理浮动。

.news {

background-color: gray;

border: solid 1px black;

}

.news img {

float: left;

}

.news p {

float: right;

}

.clearfix:after{

content: "020";

display: block;

height: 0;

clear: both;

visibility: hidden;

}

.clearfix {

/\* 触发 hasLayout \*/

zoom: 1;

}

<div class="news clearfix">

<img src="news-pic.jpg" />

<p>some text</p>

</div>

通过CSS伪元素在容器的内部元素最后添加了一个看不见的空格"020"或点"."，并且赋予clear属性来清除浮动。需要注意的是为了IE6和IE7浏览器，要给clearfix这个class添加一条zoom:1;触发haslayout。

## 总结

通过上面的例子，我们不难发现清除浮动的方法可以分成两类：

一是利用 clear 属性，包括在浮动元素末尾添加一个带有 clear: both 属性的空 div 来闭合元素，其实利用 :after 伪元素的方法也是在元素末尾添加一个内容为一个点并带有 clear: both 属性的元素实现的。

二是触发浮动元素父元素的 BFC (Block Formatting Contexts, 块级格式化上下文)，使到该父元素可以包含浮动元素，关于这一点。

在网页主要布局时使用:after伪元素方法并作为主要清理浮动方式；在小模块如ul里使用overflow:hidden;（留意可能产生的隐藏溢出元素问题）；如果本身就是浮动元素则可自动清除内部浮动，无需格外处理；正文中使用邻接元素清理之前的浮动。

最后可以使用相对完美的:after伪元素方法清理浮动，文档结构更加清晰。