

IE中对浮动元素上clear特性的解释错误

float clear 清除浮动 位置

操作系统版本：

Windows 7 Ultimate build 7600

浏览器版本：

IE6

IE7

IE8

Firefox 3.6

Safari 4.0.4

Chrome 5.0.356.2 dev

受影响的浏览器：

IE6(S)(Q)

IE7(S)(Q)

IE8(Q)

- [一、clear特性的作用](#)
- [二、含有clear特性的浮动元素对其后浮动元素定位的影响](#)
- [三、含有clear特性的浮动元素对自身定位的影响](#)
- [四、如何避免受此问题影响](#)

一、clear特性的作用

clear特性表明元素框的哪一个边不可以和先前的浮动框相邻。

另外，如果clear属性设置在浮动元素上，结果是对浮动定位的规则做了改动，加入了额外的约束：浮动元素的顶外边必须在所有先前的左浮动框的底外边界之下（如果'clear:left'）或所有先前的右浮动框的底外边界之下（如果'clear:right'），或两个都要（'clear: both'）。

关于浮动定位的详细资料，请参考W3C CSS2.1规范 [9.5.1](#) 中的内容。
关于clear特性的详细资料，请参考W3C CSS2.1规范 [9.5.2](#) 中的内容。

二、含有clear特性的浮动元素对自身定位的影响

在**IE6(S)(Q)/IE7(S)(Q)/IE8(Q)**中，如果**clear**特性被设置在一个浮动的元素上时，该浮动元素可能会与其前边的浮动元素**放在同一行**。

测试代码如下：

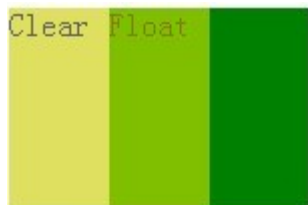
```
<div id="Container" style="background-color:silver; width:150px; float:left;">  
  <div id="Float" style="float:right; width:100px; height:100px; background-color: green;">Float</div>
```

```
<div id="Clear" style="clear:right; float:left; width:100px; height:100px; background-color:yellow;filter:alpha(opacity=50);-moz-opacity:0.5; opacity: 0.5;">Clear</div>
```

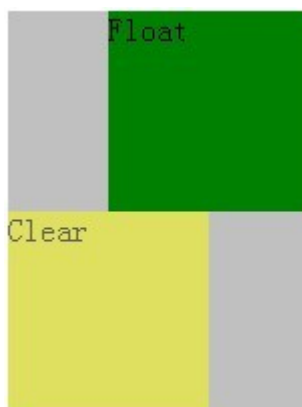
- **Container**的宽度为150px，设置"float:left"是为使容器高度自适应。
- **Clear**是含有**clear**特性的浮动元素，**Float**和**Clear**的尺寸都是100px*100px。
- 对**Clear**设置`filter:alpha(opacity=50);-moz-opacity:0.5; opacity: 0.5;`是为了让**Clear**可以透明，突出**Clear**把**Float**覆盖的事实。

根据浮动元素位置规则，**Clear**不应该与**Float**处于同一行，应该折行显示。

以上代码在不同的浏览器环境中的表现：



IE6(S)(Q)/IE7(S)(Q)/IE8(Q)



其他浏览器及IE8(S)

三、含有**clear**特性的浮动元素对其后浮动元素定位的影响

在**IE6(S)(Q)/IE7(S)(Q)/IE8(Q)**中，如果**clear**属性被设置到一个浮动的元素上，位于其后的浮动元素的外顶边可能高于该元素的外顶边。

测试代码如下：

```
<div id="Container" style="width:300px; height:200px; border:1px solid red;">
  <div id="DIV1" style="float:right; width:150px; height: 50px; background-color:darkgray;">1 float:right;</div>
  <div id="DIV2" style="clear:right; float:right; width:250px; height: 50px; background-color:silver;">2 clear:right; float:right;</div>
  <div id="DIV3" style="float:left; width:100px; height: 50px; background-color:gray;">3 float:left;</div>
</div>
```

- **Container**的宽度大于**DIV1**+**DIV3**的宽度和，**Container**的宽度小于**DIV2**+**DIV3**的宽度。
- **DIV1**是一个右浮动的元素
- **DIV2**一个右浮动元素，它含有**clear:right**特性，可以清除前面的右浮动，可以使**DIV2**位于**DIV1**的下方。
- **DIV3** 是一个普通的左浮动元素。

根据[浮动元素的位置规则第五条](#)，**DIV3**的顶外边不应该高于**DIV2**的顶外边。所以，根据标准推测，当**DIV2+DIV3**宽度大于容器宽度时，DIV3应该位于DIV2的下方；而当**DIV2+DIV3**宽度小于容器宽度时，**DIV3**应该与**DIV2**处在同一行。

这段代码在不同的浏览器环境中表现如下：

3
float:left;

1 float:right;

2 clear:right; float:right;

1 float:right;

2 clear:right; float:right;

3
float:left;

IE6(S)(Q)/IE7(S)(Q)/IE8(Q)

其他浏览器

综合第二、第三部分可见：
在IE6(S)(Q) IE7(S)(Q) IE8(Q)中，如果浮动元素的“clear”不为“none”（“clear”的默认值是“none”），那么它自身的位置和处于它之后的浮动元素的位置都有可能出现兼容性问题。这些浏览器并没有按照W3C CSS2.1规范中描述的方式去处理“clear”特性。
在其他浏览器中，则按照规范进行了正确的处理。

这个差异是由IE6(S)(Q) IE7(S)(Q) IE8(Q)中处理clear特性时的Bug造成的，它在IE8(S)中被修复。

如果一个页面在IE早期版本中被设计，并且触发了该Bug，那么这个页面在其他浏览器中的布局将与预期的不符。

四、如何避免受此问题影响

不要将“clear”特性应用在浮动元素上，以免出现上述不兼容的问题。