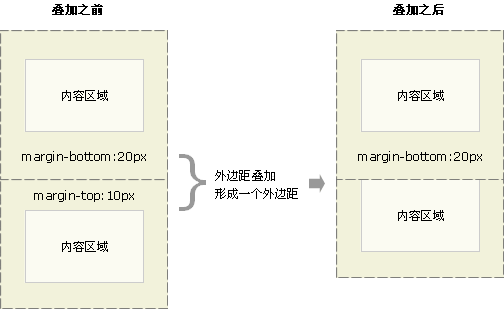
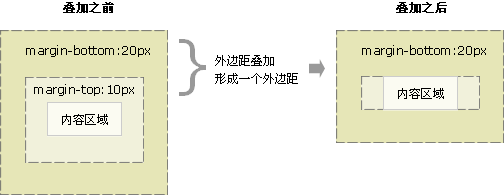
**CSS 外边距(margin)重叠及防止方法**

*边界重叠是指两个或多个盒子(可能相邻也可能嵌套)的相邻边界(其间没有任何非空内容、补白、边框)重合在一起而形成一个单一边界。*

两个或多个块级盒子的垂直相邻边界会重合。结果的边界宽度是相邻边界宽度中最大的值。如果出现负边界，则在最大的正边界中减去绝对值最大的负边界。如果没有正边界，则从零中减去绝对值最大的负边界。注意：相邻的盒子可能并非是由父子关系或同胞关系的元素生成。  
但是边界的重叠也有例外情况：

1、水平边距永远不会重合。  
2、在规范文档中，2个或以上的块级盒模型相邻的垂直margin会重叠。最终的margin值计算方法如下：  
a、全部都为正值，取最大者；  
b、不全是正值，则都取绝对值，然后用正值减去最大值；  
c、没有正值，则都取绝对值，然后用0减去最大值。  
注意：相邻的盒模型可能由DOM元素动态产生并没有相邻或继承关系。  
3、相邻的盒模型中，如果其中的一个是浮动的（float），垂直margin不会重叠，并且浮动的盒模型和它的子元素之间也是这样。  
4、设置了overflow属性的元素和它的子元素之间的margin不被重叠（overflow取值为visible除外）。  
5、设置了绝对定位（position:absolute）的盒模型，垂直margin不会被重叠，并且和他们的子元素之间也是一样。  
6、设置了display:inline-block的元素，垂直margin不会重叠，甚至和他们的子元素之间也是一样。  
7、如果一个盒模型的上下margin相邻，这时它的margin可能重叠覆盖（collapse through）它。在这种情况下，元素的位置（position）取决于它的相邻元素的margin是否重叠。  
a、如果元素的margin和它的父元素的margin-top重叠在一起，盒模型border-top的边界定义和它的父元素相同。  
b、另外，任意元素的父元素不参与margin的重叠，或者说只有父元素的margin-bottom是参与计算的。如果元素的border-top非零，那么元素的border-top边界位置和原来一样。  
一个应用了清除操作的元素的margin-top绝不会和它的块级父元素的margin-bottom重叠。  
注意，那些已经被重叠覆盖的元素的位置对其他已经重叠的元素的位置没有任何影响；只有在对这些元素的子元素定位时，border-top边界位置才是必需的。  
8、根元素的垂直margin不会被重叠。

**外边距(margin)重叠示例**  
外边距重叠是指两个垂直相邻的块级元素，当上下两个边距相遇时，起外边距会产生重叠现象，且重叠后的外边距，等于其中较大者。  
图示：

另一个重叠现象是当一个元素包含在另一个元素之中时，子元素与父元素之间也会产生重叠现象，重叠后的外边距，等于其中最大者：  
  
同理，如果一个无内容的空元素，其自身上下边距也会产生重叠。  
  
**外边距重叠的意义**

外边距的重叠只产生在普通流文档的上下外边距之间，这个看起来有点奇怪的规则，其实有其现实意义。设想，当我们上下排列一系列规则的块级元素（如段落P）时，那么块元素之间因为外边距重叠的存在，段落之间就不会产生双倍的距离。

**防止外边距重叠解决方案：**  
虽然外边距的重叠有其一定的意义，但有时候我们在设计上却不想让元素之间产生重叠，那么可以有如下几个建议可供参考：

1. 外层元素padding代替
2. 内层元素透明边框 border:1px solid transparent;
3. 内层元素绝对定位 postion:absolute:
4. 外层元素 overflow:hidden;
5. 内层元素 加float:left;或display:inline-block;
6. 内层元素padding:1px;

以上建议可根据实际情况来采取。

PS:曾经一度认为是浏览器BUG，汗颜啊！