<http://www.360doc.com/content/09/1112/15/359350_8883867.shtml>

1. position : absolute / relative 时，参照物不动，元素移动

top : xx; //元素顶部相对于参照物顶部移动的距离 （正值：下移）

bottom : xx; //元素底部相对于参照物底部移动的距离 （正值：上移）

left : xx; //元素左边相对于参照物左边移动的距离 （正值：右移）

right: xx; //元素右边相对于参照物右边移动的距离 （正值：左移）

（如定义 position:absolute，则参照物为父元素；relative，则参照物为自身原位置。

left:0， 则元素左边界置于参照物左边界

top:0， 则元素上边界置于参照物上边界

right:0， 则元素右边界置于参照物右边界

bottom:0，则元素下边界置于参照物下边界

） （移动初始即越出父元素范围的方向定义为负值； %，是指相对父元素的百分比。）

2. 当一个大小确定的元素同时定义 top 和 bottom 时，按 top 定位；

同时定义 left 和 right 时，按 left 定位。

若只定义了一维，则另一维保持为原来未设定position时的情形；

若大小未定义的元素4个维度位置均设定，则由这些维度值可确定该元素的大小。

3. 若 div 没有定义 宽度/高度，而定义为 absolute，则 div 变成为适应其内容的大小。

4. IE6，若元素没有兄弟节点，margin-left 会出现双边边距，即实际效果值为设定值的2倍。

position:

1. static：默认值，即不设置 position 属性时，按正常文档流进行排列。

2. fixed ：按浏览器的窗口定位，即不随滚动条拖动而移动。

3. relative：按文档流排列

1) 定位参照物为元素原来的位置（即没有设置position属性时）。

2) 占位空间不变，元素产生偏移，即占据原来的文档流空间，但不影响其它元素的布局，

但有可能与其它元素发生层叠，可由 z-index 设置其 居上/居下。

3) 偏移是指，元素整体在原位置的基础上偏移设定值，元素整体是指：border-padding-content；

margin不跟着作相对位置变化，因为margin决定占位空间，而相对位置变化中的占位空间不变。

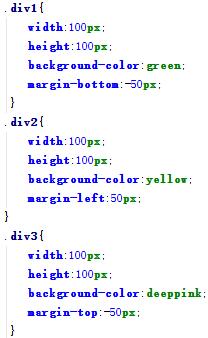
\*\* relative定位不影响相邻元素的布局，但对其子元素的影响是，子元素是在其内部布局的。

\*\* 设置margin，影响自身和相邻元素的布局；特别是margin负值时，占位空间缩减，

但是元素的物理形态不变，所以会与相邻元素发生层叠，可由 z-index 设置其居上/居下

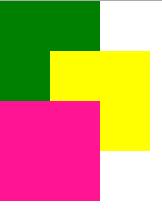
（同时要设置 position为 relative，否则 z-index 不生效）。

eg：

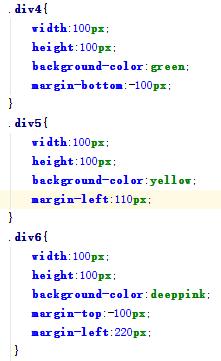


div1物理形态不变、margin-bottom为负值，则相邻元素div2的布局提前，于是产生层叠；

div3同理，margin-top为负值，则自身布局提前，层叠到相邻元素div2上。



eg：

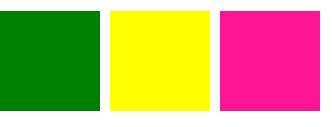


div4(margin-bottom:-100px) 使得div5提前布局，div4占位空间缩减为0，两者物理形态成一列；

注意此时div5 (margin-left) 的起点为html文档流的起点，而不是视觉上的div4，因为div4不占空间。

同理，div6(margin-top:-100px) div6提前布局，致div5占位空间缩减为0，两者物理形态成一列；

同样此时div6 (margin-left) 的起点为html文档流的起点，而不是视觉上的div5，因为div5不占空间。



\*\* margin 与 relative 混合使用：先按margin占位，再将元素按relative偏移。

margin 和 relative 都不影响元素的物理形态。

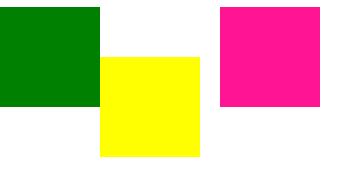
margin 影响元素自身及其相邻元素的布局位置，即占位空间。

relative 不影响元素本身的占位空间故不影响相邻元素的布局；只影响自身布局的偏移。

eg：



先按margin占位，再按relative偏移



4. absolute：按绝对位置排列，不按文档流排列，

1) 元素的祖先们没有定位(即没有设置absolute或relative)，则元素的定位参照物为最顶级元素

（依照不同浏览器，可能是body也可能是html）;

否则，该absolute元素的偏移位置，是以最近定位（含absolute和relative）的祖先元素为参照。

\*\* IE下参照物需设置宽度或高度bottom和right属性才可以正确定位。

2) 不占据原来的空间，元素从文档流中完全删除，相邻html元素补位；

相当于从文档流中抽取出来，浮动在原平面排版上，发生层叠同样可由 z-index 设置其居上/居下。

3) 元素的布局位置：

a) 在 top / bottom / left / right 设定的绝对位置上，以元素的margin开始布局：

margin-border-padding-content（margin跟着元素作绝对位置变化）。

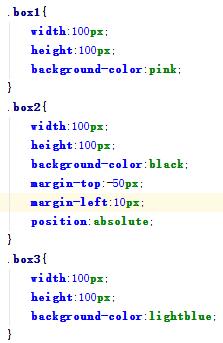
b) 若仅定义absolute而没有定义 top / bottom / left / right 中的任一个，注意它们的默认值不是0，

因为没有给定绝对位置，所以无论父元素是否有定位，该元素都是以原来的位置为参照 ---

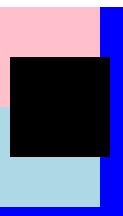
元素脱离文档流，浮动在原html文档流位置（margin跟着浮动），即自身脱离而原位置被补位。

（此处浮动不表示一定在上方，也可能在下方被补位元素覆盖，可设z-index令其显现）

eg：



box2 按 margin 布局并脱离原位置，原位置被 box3 补位。



\*\* margin 与 absolute 混合作用：先确定绝对位置，再将元素按margin布局。

margin 和 absolute 都不影响元素的物理形态。

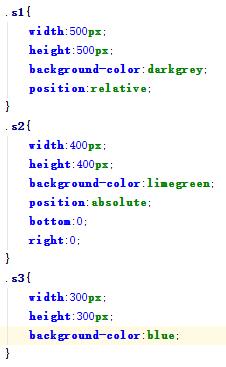
absolute 影响元素的布局位置，应找到元素的参照物（最近定位的父元素）并确定布局参考点。

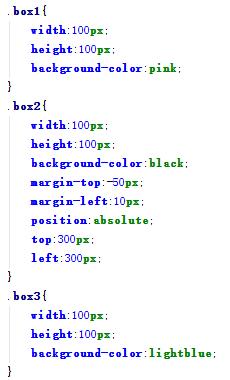
margin 影响元素的占位空间，在绝对位置上先布局 margin 再是元素内容。

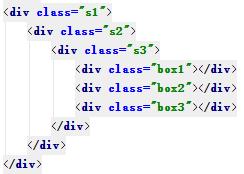
当 参考点确定，而 margin 是同一维度另一方向时，margin无法配合使用则不作布局因素考虑，如

参考点给出 bottom/right，而 margin 给出的是 top/left 时,margin没有参考点无法使用。

eg：



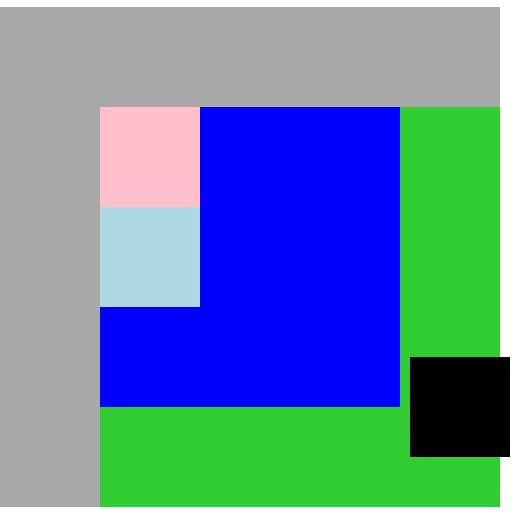




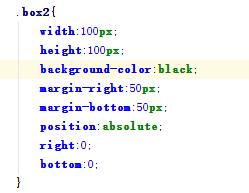
box2(黑) 为absolute，其最近的定位祖先为 s2(绿)，则 box2 以s2为参照物；

令s2的起点为坐标原点，找到位置(300,300)，即蓝块右下端点；以这个点作为参考点，

box2按 margin-top:-50px, 向上50px, 按 margin-left:10px, 向右10px放置，布局如图：

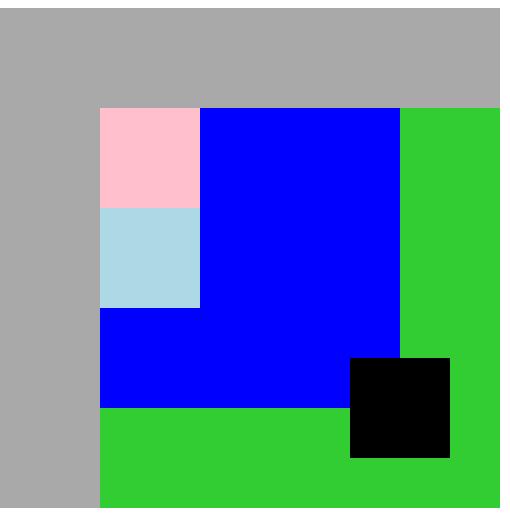


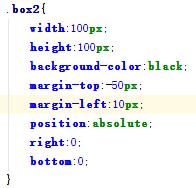
eg：



此时 box2 的参考点为父元素（绿块）的右下端点，以此开始布局，

box2按 margin-bottom:50px, 向上50px, 按 margin-right:50px, 向左50px放置，如图：





此时 box2 的参考点为父元素（绿块）的右下端点，以此开始布局，

然而 box2 已脱离文档，其 margin-top 和 margin-left 没有参考点, 无法决定布局位置；

故元素此时只按 绝对位置 点放置， 如图：

（其实仍须布局 margin-bottom 和 margin-right，只是已经被通配符设置为0）。

