在用CSS+DIV进行布局的时候，一直对position的四个属性值relative,absolute,static,fixed分的不是很清楚，以致经常会出现让人很郁闷的结果。今天研究了一下，总算有所了解。在此总结一下：

先看下各个属性值的定义：

1、static：默认值。没有定位，元素出现在正常的流中（忽略 top, bottom, left, right 或者 z-index 声明）。

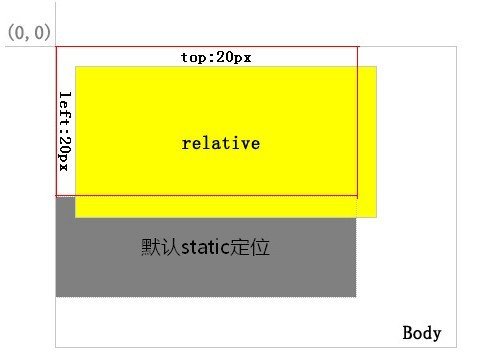
2、relative：生成相对定位的元素，通过top,bottom,left,right的设置相对于其正常位置进行定位。可通过z-index进行层次分级。

3、absolute：生成绝对定位的元素，相对于 static 定位以外的第一个父元素进行定位。元素的位置通过 "left", "top", "right" 以及 "bottom" 属性进行规定。可通过z-index进行层次分级。

4、fixed：生成绝对定位的元素，相对于浏览器窗口进行定位。元素的位置通过 "left", "top", "right" 以及 "bottom" 属性进行规定。可通过z-index进行层次分级。

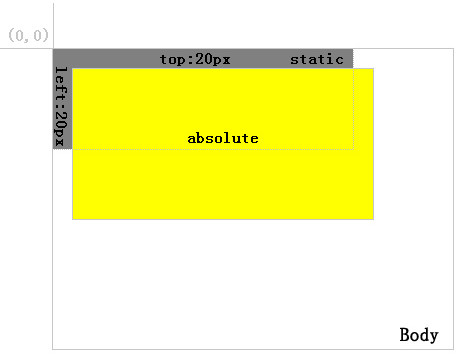
static与fixed的定位方式较好理解，在此不做分析。下面对应用的较多的relative和absolute进行分析：

1、relative。定位为relative的元素脱离正常的文本流中，但其在文本流中的位置依然存在。如图1：



黄色背景的层定位为relative，红色边框区域为其在正常流中的位置。在通过top、left对其定位后，从灰色背景层的位置可以看出其正常位置依然存在。

2、absolute。定位为absolute的层脱离正常文本流，但与relative的区别是其在正常流中的位置不在存在。如图2：

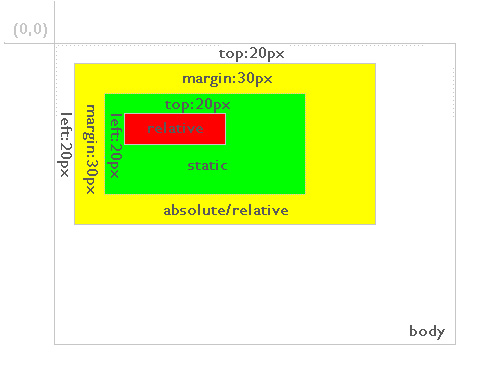


可以看到，在将黄色背景层定位为absolute后，灰色背景层自动补上。

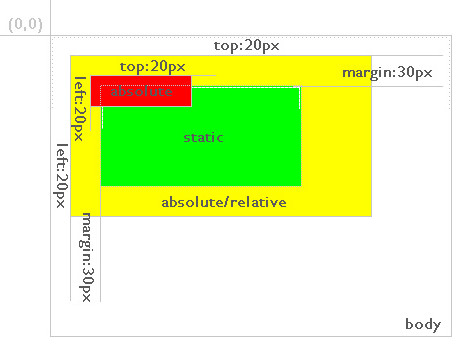
3、relative与absolute的主要区别：

首先，是上面已经提到过的在正常流中的位置存在与否。

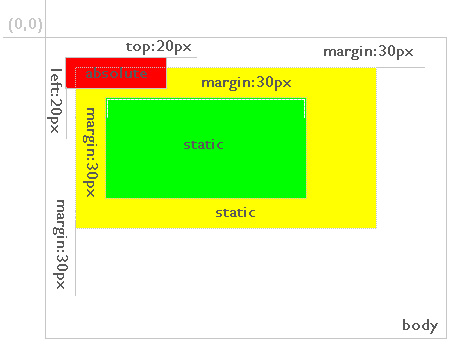
其次，relative定位的层总是相对于其最近的父元素，无论其父元素是何种定位方式（应当是相对于自己原文档流中的位置来定位）。如图3：

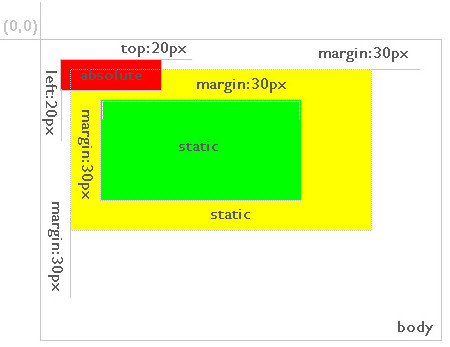


图中，红色背景层为relative定位，其直接父元素绿色背景层为默认的static定位。红色背景层的位置为相对绿色背景层top、left个20元素。而如果红色背景层定位为absolute，则情形如图4：



可以看到，红色背景层依然定义top:20px；left:20px；但其相对的元素变为定位方式为absolute或relative的黄色背景层。因此，对于absolute定位的层总是相对于其最近的定义为absolute或relative的父层，而这个父层并不一定是其直接父层。如果其父层中都未定义absolute或relative，则其将相对body进行定位，如图5：





除top、left、right、bottom定位外，margin属性值的定义也符合上述规则。