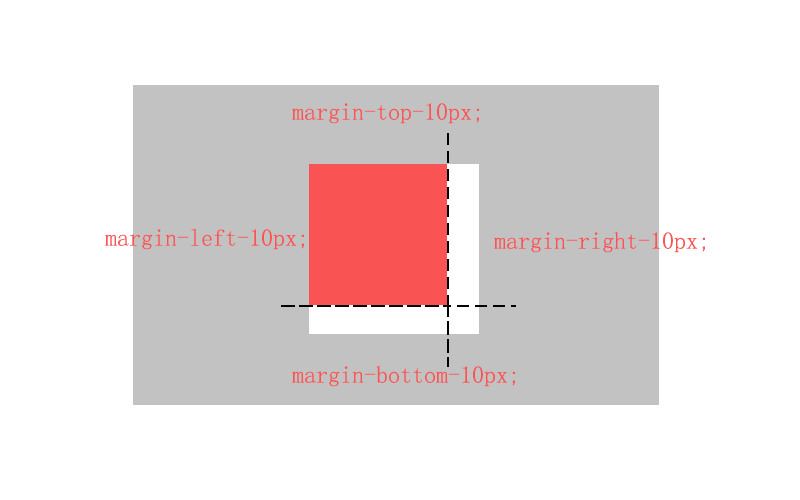
## 负margin技术

在CSS中margin属性取值可以是正数也可以是负数，无论取值为正数还是负数，都可以让当前元素或者周围元素进行移动，但是两者却有着很大的不同，我们继续听~

1. ，当元素的top或者left为负数时。当前元素会被拉到指定方向。（负数就是反着的呗）
2. 当元素的bottom或right为负数时，后续元素会被拉倒指定方向。

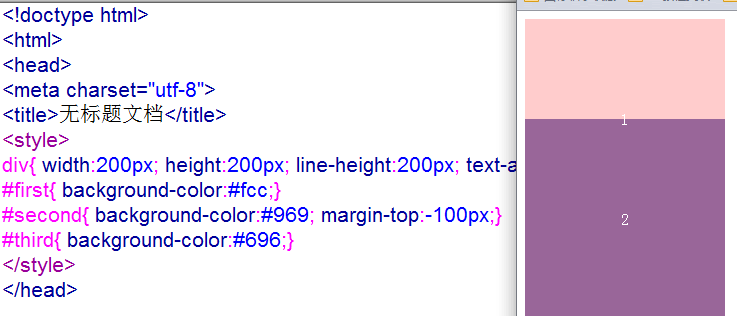
从图中我们可以简单理解一下，top和left将当前元素拉出，然后覆盖其他元素，bottom和right是将后续元素拉进，然后覆盖当前元素。



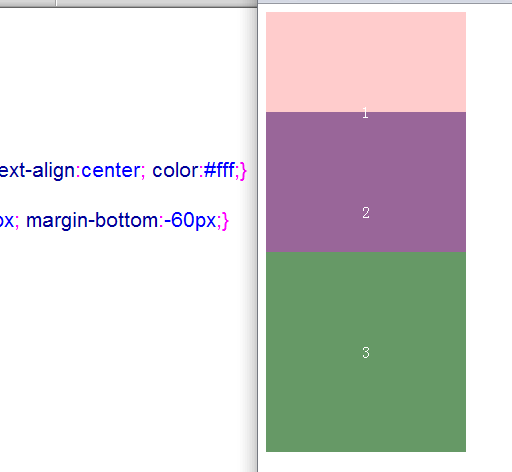
负margin对浮动元素的影响，跟负margin对普通文档流元素的影响其实是差不多的，唯一不太一样的是浮动元素可以向左也可以向右，因此对于浮动元素我们只需要比普通文档流元素多关注一点，那就是**浮动元素的流动方向**。



↑这是未设定margin-top和margin-left的显示效果，正常文档流



↑当我为第二个div添加margin-top:-100px时，第二个元素会盖住第一个元素。

当我给第三个元素设置margin-bottom：-60px的时候第三个元素会被向上拉。

负margin技巧

负margin使用很灵活，常用技巧有四个。

1. ，图片与文字对齐
2. 自适应两列布局
3. 元素垂直居中
4. Tab选项卡

1，图片与文字对齐

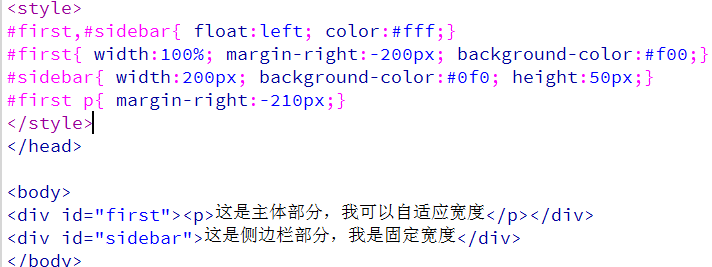
当图片与文字排在一起的时候，在底部水平方向上旺旺都是不对齐的，这是因为图片与文字默认是基线对齐（vertical-align:baseline）如果想要实现图片与文字底部水平方向对齐，除了使用vertical-align:text-bottom之外，还可以使用兼容性较好的负margin来实现。



看图，默认情况下图片与文字在底部水平方向是不对齐的，我们在CSS中添加img(margin:0 3px -70px 0;)之后，在浏览器上我们可以看到上图的这种效果。文字被挤上去啦~~~拉拉阿拉~

2，自适应两列布局

自适应两列布局，指的是在左右两列中，其中有一列的宽度为自适应，另外一列宽度是固定的，如果使用float,一般只能实现固定的左右两列布局，并不能实现其中一列的自适应布局。



我们改变浏览器的宽度，就可以很容易地看出自适应两列布局实际效果。

3，元素垂直居中

想要实现块元素的垂直居中一般来说比较麻烦，不过有一个经典的方法就是使用position结合负margin来实现。具体做法是：先给父元素写上position:relative,这样做是为了给子元素添加position:absolute;的时候不会被定位到“外太空”去，**（就是让给子元素和父级元素扯上个关系，让儿子知道谁是自己的爹，如果不知道谁是自己的爹，那子元素就会默认把body认为自己的爹）**然后给子元素添加

position:absolute,

top:50%;

left:50%;

然后再添加:

Margin-top:“height值一半的负值”

margin-left:“width值一半的负值”

父元素

{ position:relative;}

子元素{

Position:absolute;

Top:50%

Left:50%

Margin-top:height值的一半负值；

Margin-left:width值的一半负值；

}

Position这种方法是万能的，不仅可以用于块元素，还可以用于inline元素和inline块元素。就都好使就完事了

