**一、两栏布局（左边固定宽度，右边自动适应）**

**方法1、左浮动其中一个DIV，使其脱离文档流，另一个DIV不设置浮动即可。注意：DIV的顺序不能改变**

PS：设置浮动可以让元素脱离正常的文档流，使后面的元素占据浮动元素本身的位置。但是浮动元素只能影响后面的元素的位置，而不能够影响前面的元素，也不能叠加在前面的元素上。 （view类样式只是为了层便于识别）；

.view{ height: 200px; background: #ccc;}

#id1{float: left; width: 200px;}

#id2{background: #666; }

<body>

<div class="view" id="id1">侧边栏(float: left; width: 200px;)</div>

<div class="view" id="id2">内容栏(background: red;)</div>

</body>  
**方法2、使用绝对定位方法，是固定宽度的DIV固定在左侧，然后设置只适应的层的左边距**.view{ height: 200px; background: #ccc;}

#id2{margin-left: 200px; background: #666;}

#id1{position: absolute; left: 0;top:0; width: 200px;}

<div class="view" id="id2">内容栏要在前面利于SEO(margin-left: 200px; background: red;)</div>

<div class="view" id="id1">侧边栏:用了绝对定位(position: absolute; left: 0;top:0; width: 200px;)</div>  
**二、三栏布局（左右宽度固定，中间自适应宽度）  
方法1、设置浮动，使文档脱离文档流，注意层的顺序**.view{ height: 200px; background: #ccc;}

#id1{float:left; width:200px;}

#id2{float:right; width:200px;}

#id3{background:red;}

<body>

<div class="view" id="id1">1、{float:left; width:200px;}</div>

<div class="view" id="id2">2、{float:right; width:200px;}</div>

<div class="view" id="id3">3、{background:red;}</div>

</body>  
**方法2、同样通过负边距来实现，缺点是需要另外增加一个层**.view{ height: 200px; background: #ccc;}

.warp{float:left;width:100%;}

#id3{ background:red; margin-left:200px; margin-right:200px;}

#id1{float:left; position:relative; width:200px; margin-left:-100%;}

#id2{float:left; position:relative; width:200px; margin-left:-200px;}

<div class="warp">

<div class="view" id="id3">

外层包裹warpdiv，设置外层warp CSS{float:left;width:100%;}内容放最前面有利于SEO<br>

内层div控制左右两边的距离 #id3{ background:red; margin-left:200px; margin-right:200px;}

</div>

</div>

<div class="view" id="id1">左侧边负边距： #id1{float:left; position:relative; width:200px; margin-left:-100%;}</div>

<div class="view" id="id2">右侧边负边距：#id2{float:left; position:relative; width:200px; margin-left:-200px;}</div>  
**方法3、也可以通过绝对定位来实现**.view{ height: 200px; background: #ccc;}

#id3{margin-left:200px; margin-right:200px; background:red; min-width:200px;}

#id1{position:absolute; left:0; top:0; width:200px;}

#id2{float:right; width:200px; position:absolute; right:0; top:0;}

<div class="view" id="id3">顺序1、#id3{margin-left:200px; margin-right:200px; background:red;}</div>

<div class="view" id="id1">顺序2、使用绝对定位 #id1{position:absolute; left:0; top:0; width:200px;}</div>

<div class="view" id="id2">顺序3、使用绝对定位 #id2{float:right; width:200px; position:absolute; right:0; top:0;}</div>  
**三、三栏布局（左右自适应，中间宽度固定，这个有点变态，一般没这么布局的）  
方法1、增加了多个DIV，负外边距+绝对定位 如下：**\*{margin: 0;}

.warp1{float:left; width:50%;\_margin-right:-3px;}

.warp2{margin-left:50%; text-align:right; \_margin-left:0; \_zoom:1;}

.view{ height: 200px; background: #ccc;}

#id1{ margin-right:100px;}

#id2{margin-left:100px;}

#id3{ background:red;width:200px; position:absolute; left:50%; margin-left:-100px; top:0; z-index:2;}

<div class="warp1">

<div id="id1" class="view">.warp1{float:left; width:50%;}<br>#id1{ margin-right:100px;}</div>

</div>

<div class="warp2">

<div id="id2" class="view"> .warp2{margin-left:50%;}<br>#id2{margin-left:100px;}</div>

</div>

<div id="id3" class="view">

#id3{ background:red;width:200px; position:absolute; left:50%; margin-left:-100px; top:0; z-index:2;}

</div>