**【第五课】浮动布局之表现设计**

[W3C标准](http://www.w3cfuns.com/forum.php?mod=viewthread&tid=245&extra=page%3D1)建议大家结构和表现相分离，“结构”就是上节课我们搭建的HTML页面框架，“表现”则是这节课的内容，给页面穿上衣服，也就是下面要学的“CSS样式设计”，样式设计的好坏也会直接影响页面的加载速度以及改版的复杂度。我们紧接着上节课  
  
**【第六步】**  
为了页面能够具有更好的兼容性，所以在设计表现的时候，首先要对标签重置，不明白的可以去看看[[基础二] CSS选择器的通用选择器部分](http://www.w3cfuns.com/forum.php?mod=viewthread&tid=249&extra=page%3D1" \t "_blank)，在上节设计的结构里面，我们用到哪些标签就重置哪些标签，上节课用到“body,div,p,ul,li,dl,dt,dd,h1,a”这么几个标签，所以重置代码为：

1. body,div,p,ul,li,dl,dt,dd,h1,a{margin:0; padding:0;}

复制代码

打开效果图，我们先看一下主背景，是一个从上至下由蓝色渐变为白色，且带有云彩的背景，这个无规律的渐变背景我们应该怎么做？  
  
针对背景的渐变我们好处理，将渐变背景切割为一个宽度为10px的小图，起名为bg，然后水平方向平铺（repeat-x）



而对于云彩这种无规律的背景，直接切割出来作为背景，起名为clouds，如下图：



这两张背景图，在每步最后提供的源代码的images文件夹内可以找到，名子分别为bg.gif和clouds.gif。  
  
        这么处理的好处是为了提升网页的用户体验，如果我们不做任何处理直接将页面内的背景整个切割下来，当用户打开页面，会看到背景一片空白，然后瞬间显示背景，给用户的感觉很突兀，而我们的做法是，先加载前面那个10px宽度的小图片，这样背景加载速度快，用户打开网页的时候，先把这种背景图加载上，不至于展现在浏览者前面的是一片空白，然后等云彩背景下载完毕后，再加载在页面内，用户感觉会很舒服，这就是为什么Web前端开发人员还要懂一些用户体验设计方面的知识。  
  
        具体如何实现背景的显示顺序，方法有很多，针对本例，最适用的办法就是设置为不同的盒子的背景，bg.gif就当做html盒子的背景，而clouds.gif就当做body盒子的背景，因为网页会先加载html盒子，然后再加载body盒子，那么CSS代码如下：

1. html{background:url(../images/bg.gif) repeat-x;}
2. body{background:url(../images/clouds.gif) repeat-x;}

复制代码

效果是像下面这样，背景加载上了



此时内容有点乱这是肯定的，因为我们还没有对他们定义样式~  
  
**第六步源代码：** http://www.w3cfuns.com/static/image/filetype/rar.gif [浮动布局页面6.rar](http://www.w3cfuns.com/forum.php?mod=attachment&aid=MTEzMHwwNzYxZmU3ZnwxNDA5ODM0NDI2fDB8NDYx) (256.84 KB, 下载次数: 1076)   
  
  
  
  
**【第七步】**  
        页面的宽度为1000px，所以四大板块header、banner、content、footer的宽度也为1000px，且水平居中于浏览器，所以用CSS集体声明的方法，对四大板块的共同样式进行定义：

1. #header,#banner,#content,#footer{width:1000px;  margin:0 auto;}

复制代码

其中“**margin:0 auto；**”的作用就是将页面元素水平居中，这个很重要，布局页面经常用到需要记住！  
现在再看一下效果，四大板块是不是全部水平居中于浏览器了~  
  
        另外效果图中的头部是与上边沿有45px的距离，为实现这个效果，我们采用设置body上内边距的办法,将padding-top加到body的样式内：  
  
**CSS代码**

1. body{background:url(../images/clouds.gif) repeat-x; padding-top:45px;}

复制代码

离我们的最后完成页面制作又前进一步~  
  
**第七步源代码：** http://www.w3cfuns.com/static/image/filetype/rar.gif [浮动布局页面7.rar](http://www.w3cfuns.com/forum.php?mod=attachment&aid=MTEzMnw3NmYxZDU1MnwxNDA5ODM0NDI2fDB8NDYx) (256.9 KB, 下载次数: 679)   
  
  
  
**【第八步】**  
注意：第八步细节上的内容多些，有一点做的不到位，就会导致页面兼容性问题，所以大家耐下心来，仔细品味每一句话。  
  
        大家有没有发现一个问题，网页里所有能够点击的链接图片，全部都有一个宽度为2px的紫色边框，不美观并且还会导致后面做出的页面出现兼容性的问题，这个不是我们想要的，所以我们要将这个边框去掉！CSS代码如下所写：

1. img{border:none;}

复制代码

上面这段代码是对img标签边框的重置，所以我们要把它和之前标签内外边距重置的代码放一块，如下：

1. body,div,p,ul,li,dl,dt,dd,h1,a,img{margin:0; padding:0;}
2. img{border:none;}

复制代码

这样页面上的图片清爽多了，后面我们还有对其他标签的重置，用到的时候，我们再添加到第二条的后面。  
  
  
上节我们分析过了，头部header包含两部分，左侧的logo和右侧的导航nav。  
  
先说说左侧logo部分，回忆一下上节课logo部分的结构：  
**HTML代码：**

1. <div id="logo">
2. <a href="http://www.w3cstudy.com"><img src="images/logo.png" /></a>
3. </div>

复制代码

分析一下上面的结构，首先最外层由一个块状元素div构成的盒子，第二层由内联元素a构成，第三层由内联元素img构成，如果对这些常用标签分不清哪些是块状元素哪些是内联元素，就去看看[xHTML标签](http://www.w3cfuns.com/portal.php?mod=topic&topicid=21" \t "_blank)页面。  
  
        接着上面，img标签是一个很特别的标签，因为它本身是内联元素，但却体现出块状元素宽高起作用的特性，这是很矛盾的地方，这就为页面布局埋下隐患，要么为内联元素，要么为块状元素，在这里我们更需要它的块状元素的属性，所以我们将身为内联元素的img标签转化为块状元素，用“**display:block;** ”。  
  
        既然第三层的img转化为块状元素，根据W3C规范，内联元素是不能包含块状元素的，所以我们还必须把第二层的链接a，也要转化为块状元素，还是用“**display:block;** ”。  
  
        我们希望点击logo图片的任何一个地方都可以回到首页，也就是说第二层a的有效点击区域大小就为图片的大小，而外面的盒子的大小也等于图片的大小。  
  
那么相应的样式代码如下：  
**CSS代码：**

1. #logo{width:220px; height:54px; float:left; background:#991616;}
2. #logo a{display:block; width:220px; height:54px;}
3. #logo a img{display:block;}

复制代码

针对代码中的“#logo a img{}”的写法不清楚的，就去看看[[基础二] CSS选择器](http://www.w3cfuns.com/forum.php?mod=viewthread&tid=249&extra=page%3D1" \t "_blank)的选择器的嵌套  
  
代码中的“**float:left；**”的作用这里不再说了，不明白的同学去看一下[[第二课] 浮动](http://www.w3cfuns.com/forum.php?mod=viewthread&tid=248&extra=page%3D1" \t "_blank)  
  
代码中的“**background:#991616;**”是一个从logo图片上取下来的红色，目的和前面的设置页面背景一样，为了保证在网速比较慢的情况下，logo图片还未加载完成之时，先显示红色背景，增强用户体验！  
  
**此步源代码：** http://www.w3cfuns.com/static/image/filetype/rar.gif [浮动布局页面8-1.rar](http://www.w3cfuns.com/forum.php?mod=attachment&aid=MTE2MHxiYTYyNjdlMXwxNDA5ODM0NDI2fDB8NDYx) (256.98 KB, 下载次数: 531)   
  
  
        我们就可以进一步思考，既然a已经设置了宽度和高度，外面的那个id为logo的div，其实就没有存在的必要了，倒不如直接去掉外层的div，并将a的id设置为logo，这样我们的结构就少一层，代码就少一些，这正是我们期望的，所以logo部分的HTML结构就可以变为：

1. <a id="logo" href="http://www.w3cstudy.com"><img src="images/logo.png" /></a>

复制代码

CSS代码就可以变为：

1. #logo{display:block; width:220px; height:54px; float:left; background:#991616;}

复制代码

而“#logo a img{display:block;}”就要转化为

1. #logo img{display:block;}

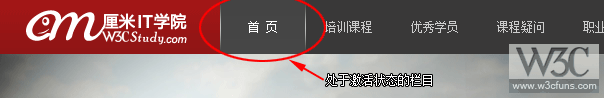
复制代码

因为现在的#logo已经为a的id了，不再是原来最外层的div的id。  
  
 **此步源代码：** http://www.w3cfuns.com/static/image/filetype/rar.gif [浮动布局页面8-2.rar](http://www.w3cfuns.com/forum.php?mod=attachment&aid=MTE2Mnw1MGViYWJmMnwxNDA5ODM0NDI2fDB8NDYx) (256.97 KB, 下载次数: 639)   
  
  
        导航nav这块的CSS代码就不说了，如果CSS代码写不出来，就去回顾一下[[第四课] 导航条(上)](http://www.w3cfuns.com/forum.php?mod=viewthread&tid=252&extra=page%3D1" \t "_blank) | [导航条(下)](http://www.w3cfuns.com/forum.php?mod=viewthread&tid=254&extra=page%3D1)  
这里直接将CSS代码列出来

1. #nav{width:780px; height:54px; float:left; background:#393838; list-style:none;}
2. #nav li{float:left;}
3. #nav li a{display:block; width:86px; height:54px;  line-height:54px; text-align:center; font-size:12px; color:#ccc; text-decoration:none; float:left;}
4. #nav li a:hover{color:#fff;}

复制代码

有一点需要补充，现在我们实现的效果是鼠标移至导航条栏目的上面，文字由灰色变为白色，鼠标移出，文字又转为灰色，而处于激活状态的栏目的背景是一个灰色且两边带有光影的图片，如下图：



激活状态的栏目是不受鼠标的移入和移出影响的，这个应该怎么做呢？我们需要定义一个类选择器，专门加在处于激活状态的栏目上：  
**CSS代码**

1. #nav li .navActive{ background:url(../images/navHoverBg.png) no-repeat; color:#fff;}

复制代码

如果激活状态的栏目是“首页”，结构代码就这么写  
**HTML代码**

1. <li><a href="http://www.w3cstudy.com" class="navActive">首 页</a></li>

复制代码

到这里header的样式定义基本完成  
  
**此步源代码**： http://www.w3cfuns.com/static/image/filetype/rar.gif [浮动布局页面8-3.rar](http://www.w3cfuns.com/forum.php?mod=attachment&aid=MTE2OHxiM2YyZTA5OXwxNDA5ODM0NDI2fDB8NDYx) (257.12 KB, 下载次数: 1026)   
  
注意：一般情况下，很多人会认为header部分的样式定义就完成了，其实没有注意到这里面隐藏着一个潜在危险，就是<div id="header">高度自适应的问题，尽管里面的logo和导航条都设置了高度，但是header的高度在IE7、IE8、IE9、FF内，高度是不能自动适应里面元素的高度，为了更清楚的说明问题，大家可以将HTML结构代码中的导航条部分删除，然后再给header设置一个背景色为黑色#000，看看header的高度是否能够自适应logo的高度。  
  
删除导航条代码后的header结构  
**HTML代码**

1. <div id="header">
2. <a  id="logo" href="http://www.w3cstudy.com"><img src="images/logo.png" /></a>
3. </div>

复制代码

**CSS代码**

1. #header{background:#000;}

复制代码

在IE6和FF内的显示效果如下：



        发现问题了没有，在FireFox内，看不到背景为黑色的header，因为他不能够自适应内部元素的高度，所以它的高度变为零，而在ie6里面却可以正常显示，这是一个很严重的问题，如果高度不能自适应的话，那内部子元素logo和nav的浮动产生的影响就会“外泄”，对header外面的版块产生影响，从而产生版块错位！  
  
        解决这个问题其实很简单，不是高度不能自适应嘛，那就给他设置一个高度，高度值为子级元素的高度，那这样header这个盒子就可以刚好装下内部的子logo和nav，而不至于浮动的影响“外泄”，但是这种办法，不够灵活，如果有一天header内部多了一些元素，那就需要再次计算一下子级元素的高度，比较麻烦，下面介绍第二种方法，在不设置header的高度情况下，无论内部元素再多，header的高度也会自动适应，直接在header的样式里写入“overflow:hidden;”就可以了

1. #header{overflow:hidden;}

复制代码

现在再去看看FF下的页面，是不是黑色的header显示出来了，既然高度自适应了，黑色的背景也就没用了，去掉它，然后再在HTML结构代码内恢复导航代码，这样做才是真正的完成了页面header部分的样式定义~  
  
一个Web前端开发人员是否专业，就体现在这些小细节上！  
  
**第八步源代码：** http://www.w3cfuns.com/static/image/filetype/rar.gif [浮动布局页面8-4.rar](http://www.w3cfuns.com/forum.php?mod=attachment&aid=MTE3MHw5ZDE2ODNmN3wxNDA5ODM0NDI2fDB8NDYx) (257.13 KB, 下载次数: 708)   
  
  
  
  
**【第九步】**  
        针对banner版块的样式定义，相对第八步就简单多了，结构和logo版块一样，对结构进行优化，将最外层的div去掉，然后将里面的a定义为banner，所以banner版块的结构如下：

1. <a id="banner" href="http://www.w3cstudy.com">
2. <img src="images/banner.jpg" />
3. </a>

复制代码

CSS代码：

1. #banner{display:block; width:1000px; height:292px; margin:10px auto;}
2. #banner img{display:block;}

复制代码

其中“margin:10px auto;”等价于“margin:10px auto 10px auto;”,四个数值的顺序是“**上右下左**”，根据第七步我们知道如果设置左右为auto，那么版块就会居中，但是在这里我们还想让banner距离上面的header和下面的content的距离为10px，那就设置上下为10px就可以了，于是得出了“margin:10px auto;”  
  
        到这里，效果和设计上的是一样的，但是作为Web前端开发人员，一定要思考全面，如果图片的大小不是1000px\*292px怎么办？图片大了，就会溢出，图片小了不美观，所以我们这里还要加一句，无论图片的大小如何，只要在页面上显示，宽高永远是1000px\*292px

1. #banner img{display:block; width:1000px; height:292px;}

复制代码

这句话的意思就是#banner里面的图片大小为1000px\*292px，无论图片原本大小，只要在#banner内，就是这个宽高。  
如果第九步里面有看不明白的，说明第八步没有仔细看，返回去再去学习一下。  
  
**第九步源代码：** http://www.w3cfuns.com/static/image/filetype/rar.gif [浮动布局页面9.rar](http://www.w3cfuns.com/forum.php?mod=attachment&aid=MTE2M3w2MjFmYmFiNnwxNDA5ODM0NDI2fDB8NDYx) (257.17 KB, 下载次数: 895)   
  
第九步完成后的效果



**【第十步】**  
        下面将针对content版块进行样式设计，从效果图上我们可以看出，content版块分为两块，左侧的文章（leftArticle）和右侧的资讯（rightInfo）。  
  
        首先分析下思路，content内部的leftArticle和rightInfo两个都要左侧浮动，内部元素浮动就会导致外面的content的高度不能够自适应内部元素的高度，所以我们首先要在content的CSS代码中加入“**overflow:hidden;**”,顺便将背景色、文字大小和文字行距也给定义了，如下：

1. #content{overflow:hidden; background:#eaeaea; font-size:12px; line-height:24px;}

复制代码

然后我们再针对leftArticle和rightInfo单独定义样式，仔细观察效果图能够很清楚的发现，leftArticle和rightInfo上下左右都有10像素的外边距，如果我们用左侧浮动来实现2栏效果，在存在外边距的情况下就会出现IE6的双倍边距Bug（在[[第二课] 浮动](http://www.w3cfuns.com/forum.php?mod=viewthread&tid=248&extra=page%3D1" \t "_blank)讲过），为了避免出现bug，我们就利用“**display:inline;**”,所以他们两个的CSS就这么定义

1. #leftArticle,#rightInfo{margin:10px; float:left; display:inline;}

复制代码

OK,他们两个的公共样式定义完了，下面就开始对他们单个定义，在这之前，你需要意识到，原本leftArticle和rightInfo的宽度分别为680px和320px，当存在10px的外边距后，如果再给他们用CSS定义宽度的话，那么宽度的数值就变为660px和300px，都要减去20像素（为什么是20px，因为存在10px左外边距和10px右外边距）

1. #leftArticle{width:660px;}
2. #rightInfo{width:300px;}

复制代码

现在再看看效果，页面雏形是不是已经出来了~，开心吧~



**此步源代码：** http://www.w3cfuns.com/static/image/filetype/rar.gif [浮动布局页面10-1.rar](http://www.w3cfuns.com/forum.php?mod=attachment&aid=MTE3NXw5MWE0MjU5MnwxNDA5ODM0NDI2fDB8NDYx) (257.23 KB, 下载次数: 641)   
  
        我们继续观察左侧的leftArticle，是不是有个图片，图片是不是可以点击的，以后遇到图片链接这种情况，第一反应就是先将这两块给设置成块状元素，我们在这里用[骆驼命名法](http://www.w3cfuns.com/forum.php?mod=viewthread&tid=251&extra=page%3D1" \t "_blank)，给图片外面的链接起个id叫articlePicA，意思是文章图片外部的链接

1. <a id="articlePicA" href="http://blog.w3cstudy.com/?p=5"><img src="images/articleBanner.jpg" /></a>

复制代码

相应的CSS如下

1. #articlePicA,#articlePicA img{display:block;}

复制代码

图片链接搞定！下面轮到文章的标题，在HTML结构中用的是<h1>标签，现在我们要按照效果图上的设计，标题距离上面的图片和下方的文字的距离为20px，文字的大小为24px，首选字体为“微软雅黑”，其次为“黑体”

1. #leftArticle h1{margin:20px 0; font-size:24px; font-family:"微软雅黑", "黑体";}

复制代码

对于标题与上下的距离这里用的是外边距margin来实现，当然也可以用padding，如果这样就会把盒子给撑大，就需要对文字再次定义，让其垂直居中，反而麻烦了，方法不止一种，作为Web前端开发工程师，一定要用最简单的代码实现效果，这也是体现是否专业的一方面。  
  
        预览页面后，发现字体变了，但是大小却没有变化，原因是由于浏览器存在默认样式表产生的，所以我们就需要对h1进行重置，如下：

1. h1{font-size:100%;}

复制代码

加上这句话是不是我们前面设置的h1标签的文字大小就变为24px了  
  
        既然是重置代码，我们就需要将它和其他的重置代码放一块，好，现在我们的重置代码有三行了，分别是针对标签的内外边距的重置，图片边框的重置，以及h1文字大小的重置。

1. body,div,p,ul,li,dl,dt,dd,h1,a{margin:0; padding:0;}
2. img{border:none;}
3. h1{font-size:100%;}

复制代码

将制作出来的标题和效果图中的标题对比，发现效果图内的链接是红色的，鼠标移上去会出现下划线，为实现这个效果就需要对标题<h1>内的链接<a>的样式进行定义

1. #leftArticle h1 a{color:#900; text-decoration:none;}
2. #leftArticle h1 a:hover{text-decoration:underline;}

复制代码

标题定义完了，下面说说内容，还是要和效果图进行对比，对比的结果是效果图上的文字颜色并不是纯黑，而是数值为“#333”的黑色，并且每段文字都会缩进两个文字，每段文字距离下段文字之间的距离是30px，文字大小为14px，CSS如下定义：

1. #leftArticle p{color:#333; text-indent:2em; margin-bottom:30px; font-size:14px;}

复制代码

怎么样到这里左侧leftArticle的效果是不是和效果图里的一样了呢  
  
**此步源代码：** http://www.w3cfuns.com/static/image/filetype/rar.gif [浮动布局页面10-2.rar](http://www.w3cfuns.com/forum.php?mod=attachment&aid=MTE3Nnw3YzQ2OTlkM3wxNDA5ODM0NDI2fDB8NDYx) (257.37 KB, 下载次数: 619)   
  
  
        下面轮到定义rightInfo版块的样式了，从效果图里不难看出，“职业生涯”和“好职推荐”之间有10px的距离，实现这个效果我们就采取设置dl的下外边距的方法

1. #rightInfo dl{margin-bottom:10px;}

复制代码

针对每个版块的标题dt的定义，如下：

1. #rightInfo dl dt{
2. background:url(../images/title.jpg) no-repeat;
3. height:32px;
4. line-height:32px;
5. color:#FFF;
6. font-size:14px;
7. font-weight:bold;
8. text-indent:20px;
9. }

复制代码

背景我们就直接采用已经切好的title.jpg且不平铺，dt高度就是title.jpg的高度32px，为了使文字垂直居中，采取设置行高的方法，不会此方法的去看文章《[两种方法实现CSS垂直居中](http://www.w3cfuns.com/forum.php?mod=viewthread&tid=453" \t "_blank)》，然后设置文字大小为14px、粗细为加粗、以及文字缩进20px。  
  
下面是针对dd的定义

1. #rightInfo dl dd{
2. height:24px;
3. line-height:24px;
4. background:url(../images/dot.gif) no-repeat 7px 10px;
5. text-indent:16px;
6. }

复制代码

重点解释一下第三句，第三句的目的是将做好的小红点图片dot.gif设置为dd的背景且不平铺，距离dd左侧的距离为7px，上面的距离为10px；  
  
dd里面的链接颜色还不是我们效果图上的效果，定义如下：

1. #rightInfo dl dd a{color:#333; text-decoration:none;}
2. #rightInfo dl dd a:hover{color:#900; text-decoration:underline;}

复制代码

OK！右侧版块样式定义完成！页面效果如下：



**第十步源代码：** http://www.w3cfuns.com/static/image/filetype/rar.gif [浮动布局页面10-3.rar](http://www.w3cfuns.com/forum.php?mod=attachment&aid=MTE3N3w1ODdjNzVlZHwxNDA5ODM0NDI2fDB8NDYx) (257.47 KB, 下载次数: 598)   
  
  
  
  
**【第十一步】**  
        如果前面几步掌握的不错，那么这一步就是最简单的一步，学起来会轻松很多，还是要看效果图，footer是一个颜色为#393838灰色块，与上部的content距离是10px，自身高度是70px，文字颜色为灰白色(#ccc)且水平居中,文字与footer顶部的距离为18px，行距也是18px，样式定义如下：

1. #footer{
2. background:#393838;
3. height:52px;
4. line-height:18px;
5. margin-top:10px;
6. padding-top:18px;
7. text-align:center;
8. color:#ccc;
9. font-size:12px;
10. }

复制代码

因为设置了padding-top:18px,这个上内边距，所以定义footer的高度就要由原来的高度，减去内边距，即70px-18px=52px，这样footer在视觉上的高度才是70px。  
  
这时的效果与效果图上的很接近了，唯独就是链接的颜色和样式，加上下面的代码瞅瞅

1. #footer a{color:#ccc; text-decoration:none;}
2. #footer a:hover{text-decoration:underline;}

复制代码

怎么样页面和效果图一样了吧~！  
  
        细心的同学也许会发现，页面内的英文的字体全部是“宋体”，不是十分的美观，不如英文常用的字体verdana，那我们就给整个页面内的文字字体设置首选字体为“verdana”，只需要在body的样式里，加入“font-family”就可以了

1. body{background:url(../images/clouds.gif) repeat-x; padding-top:45px; font-family:Verdana, Geneva, sans-serif;}

复制代码

**第十一步源代码：** http://www.w3cfuns.com/static/image/filetype/rar.gif [浮动布局页面11.rar](http://www.w3cfuns.com/forum.php?mod=attachment&aid=MTE3OXwwMjAxNGY2ZXwxNDA5ODM0NDI2fDB8NDYx) (257.51 KB, 下载次数: 2465)   
  
  
  
**【结束】**  
此时此刻，我们将一个成品的网页设计成功的制作成网页！恭喜大家！  
  
如果您学习的目的仅仅是将页面布局出来，那么你毕业了，如果你是Web前端开发工程师或者打算从事Web前端开发，那就不能将自己局限于“能把页面制作出来就行”，网页不只是给人看的，做出来摆在那里就行，我们要实现网页的价值！  
  
实现网页价值，就必须先学会，如何提升页面对搜索引擎的亲和度，如何将代码高度压缩，如何提升页面加载速度，如何为企业节省互联网成本，如何提升网页的整体价值，将页面优化至极致！这些内容将在[中高级课程](http://www.w3cfuns.com/commodity-MQ==.html)中告诉大家。  
  
好！下课！