# 系列：HTML+CSS基础课程

目录

[系列：HTML+CSS基础课程 1](#_Toc401407524)

[1、加入强调语气，使用<strong>和<em>标签 1](#_Toc401407525)

[2、q标签，短文本引用 2](#_Toc401407526)

[3、blockquote标签，长文本引用 4](#_Toc401407527)

[4、address标签，为网页加入地址信息 4](#_Toc401407528)

[5、想加入一行代码吗？使用code标签 5](#_Toc401407529)

[6、使用pre标签为你的网页加入大段代码 6](#_Toc401407530)

[7、使用ol，添加图书销售排行榜 7](#_Toc401407531)

[8、caption标签，为表格添加标题和摘要 9](#_Toc401407532)

[9、在新建浏览器窗口中打开链接 10](#_Toc401407533)

[10、使用mailto在网页中链接Email地址 10](#_Toc401407534)

[任务 11](#_Toc401407535)

[11、使用表单标签，与用户交互 12](#_Toc401407536)

[12、文本域，支持多行文本输入 13](#_Toc401407537)

[13、使用单选框、复选框，让用户选择 15](#_Toc401407538)

[14、使用下拉列表框，节省空间 16](#_Toc401407539)

[15、使用下拉列表框进行多选 17](#_Toc401407540)

[16、使用提交按钮，提交数据 18](#_Toc401407541)

[17、使用重置按钮，重置表单信息 19](#_Toc401407542)

[任务 20](#_Toc401407543)

[18、form表单中的label标签 21](#_Toc401407544)

[任务 21](#_Toc401407545)

[19、CSS代码语法 23](#_Toc401407546)

[20、内联式css样式，直接写在现有的HTML标签中 23](#_Toc401407547)

[21、嵌入式css样式，写在当前的文件中 24](#_Toc401407548)

[22、外部式css样式，写在单独的一个文件中 24](#_Toc401407549)

[23、三种方法的优先级 25](#_Toc401407550)

[24、什么是选择器？ 25](#_Toc401407551)

[25、标签选择器 26](#_Toc401407552)

[26、类选择器 26](#_Toc401407553)

[27、ID选择器 27](#_Toc401407554)

[28、类和ID选择器的区别 27](#_Toc401407555)

[29、子选择器 28](#_Toc401407556)

[任务 29](#_Toc401407557)

[30、包含(后代)选择器 30](#_Toc401407558)

[31、通用选择器 30](#_Toc401407559)

[32、伪类选择符 30](#_Toc401407560)

[33、分组选择符 31](#_Toc401407561)

[34、继承 31](#_Toc401407562)

[35、特殊性 32](#_Toc401407563)

[36、层叠 33](#_Toc401407564)

[37、重要性 33](#_Toc401407565)

[38、重要性 34](#_Toc401407566)

[39、文字排版--字体 34](#_Toc401407567)

[40、文字排版--粗体 35](#_Toc401407568)

[41、文字排版--斜体 35](#_Toc401407569)

[42、文字排版--下划线 35](#_Toc401407570)

[43、文字排版--删除线 36](#_Toc401407571)

[44、段落排版--缩进 36](#_Toc401407572)

[45、段落排版--行间距 37](#_Toc401407573)

[46、段落排版--字间距、字母间距 37](#_Toc401407574)

[47、段落排版--对齐 37](#_Toc401407575)

[48、元素分类 38](#_Toc401407576)

[49、元素分类--行内元素 39](#_Toc401407577)

[50、元素分类--内联块状元素 39](#_Toc401407578)

[51、盒模型--边框（一） 39](#_Toc401407579)

[52、盒模型--边框（二） 40](#_Toc401407580)

[53、盒模型--宽度和高度 41](#_Toc401407581)

[54、盒模型--填充 42](#_Toc401407582)

[55、盒模型--边界 43](#_Toc401407583)

[56、css布局模型 43](#_Toc401407584)

[57、流动模型（一） 44](#_Toc401407585)

[58、流动模型（二） 44](#_Toc401407586)

[59、什么是层模型？ 44](#_Toc401407587)

[60、层模型--绝对定位 45](#_Toc401407588)

[61、层模型--相对定位 46](#_Toc401407589)

[62、层模型--固定定位 48](#_Toc401407590)

[63、Relative与Absolute组合使用 49](#_Toc401407591)

[这样box2就可以相对于父元素box1定位了（这里注意参照物就可以不是浏览器了，而可以自由设置了）。  64、盒模型代码简写 50](#_Toc401407592)

[64、颜色值缩写 51](#_Toc401407593)

[65、字体缩写 51](#_Toc401407594)

1、加入强调语气，使用<strong>和<em>标签

       有了段落又有了标题，现在如果想在一段话中特别强调某几个文字，这时候就可以用到<em>或<strong>标签。

       但两者在强调的语气上有区别:<em> 表示强调，<strong> 表示更强烈的强调。并且在浏览器中<em> 默认用**斜体**表示，<strong> 用**粗体**表示。两个标签相比，目前国内前端程序员更喜欢使用<strong>表示强调。

**语法：**

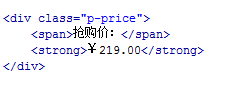
<em>需要强调的文本</em>

<strong>需要强调的文本</strong>

如，在网上商城中，某产品的打折后的价格是需要强调的。如下图。

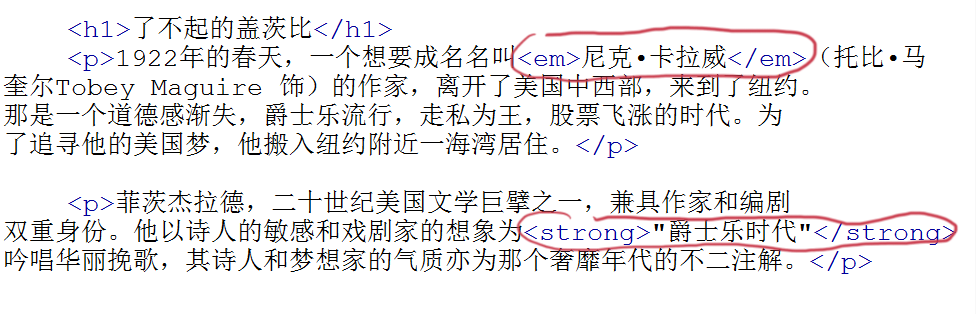
[](http://img.mukewang.com/528b406b0001f06701720271.jpg)

代码实现：

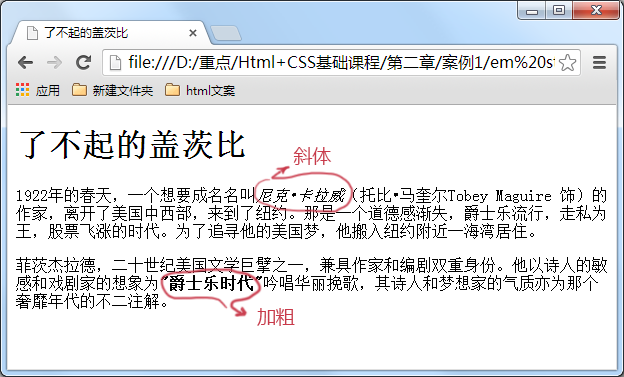
[](http://img.mukewang.com/528b409a00015f1a02300097.jpg)

**在浏览器中默认样式是有区别的：**

原代码，如下图。

[](http://img.mukewang.com/528b41a80001b81509760314.jpg)

浏览器中的样子，如下图。

[](http://img.mukewang.com/528b41cf0001e8d306240377.jpg)

<em>的内容在浏览中显示为斜体，<strong>显示为加粗。如果不喜欢这种样式，没有关系，以后可以使用css样式去改变它。

2、q标签，短文本引用

      想在你的html中加一段引用吗？比如在你的网页的文章里想引用某个作家的一句诗，这样会使你的文章更加出彩，那么<q>标签是你所需要的。

**语法：**

<q>引用文本</q>

如下面例子：

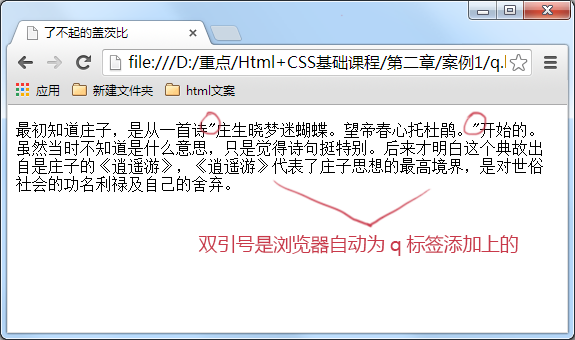
<p>最初知道庄子，是从一首诗**<q>庄生晓梦迷蝴蝶。望帝春心托杜鹃。</q>**开始的。虽然当时不知道是什么意思，只是觉得诗句挺特别。后来才明白这个典故出自是庄子的《逍遥游》，《逍遥游》代表了庄子思想的最高境界，是对世俗社会的功名利禄及自己的舍弃。</p>

**讲解：**

1. 在上面的例子中，“**庄生晓梦迷蝴蝶。望帝春心托杜鹃。**” 这是一句诗歌，出自晚唐诗人李商隐的《锦瑟》 。因为不是作者自己的文字，所以需要使用<q></q>实现引用。

2. 注意要引用的文本不用加**双引号**，浏览器会对q标签自动添加双引号。

下图是代码显示结果：

[](http://img.mukewang.com/528c0ab90001413905750340.jpg)

注意这里用<q>标签的真正关键点不是它的默认样式**双引号**（如果这样我们不如自己在键盘上输入双引号就行了），而是它的**语义：引用别人的话**。

## 3、blockquote标签，长文本引用

      <blockquote>的作用也是引用别人的文本。但它是对**长文本**的引用，如在文章中引入大段某知名作家的文字，这时需要这个标签。

      等等，上一节<q>标签不是也是对文本的引用吗？不要忘记<q>标签是对**简短文本**的引用，比如说引用一句话就用到<q>标签。

      如想在我的文章中引用李白《关山月》中的诗句，因为引用文本比较长，所以使用**<**blockquote>。

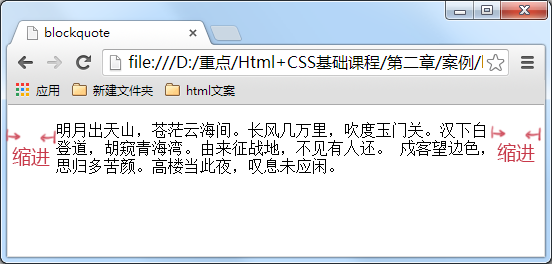
**语法:**

**<**blockquote>引用文本</blockquote>

如下面例子：

**<blockquote>**明月出天山，苍茫云海间。长风几万里，吹度玉门关。汉下白登道，胡窥青海湾。由来征战地，不见有人还。 戍客望边色，思归多苦颜。高楼当此夜，叹息未应闲。**</blockquote>**

浏览器对**<**blockquote>标签的解析是**缩进样式。如下图所示：**

**[](http://img.mukewang.com/528c50ea000146a205520264.jpg)**

## 4、address标签，为网页加入地址信息

一般网页中会有一些地址信息需要在网页中展示出来，这些地址信息如公司的地址就可以<address>标签。

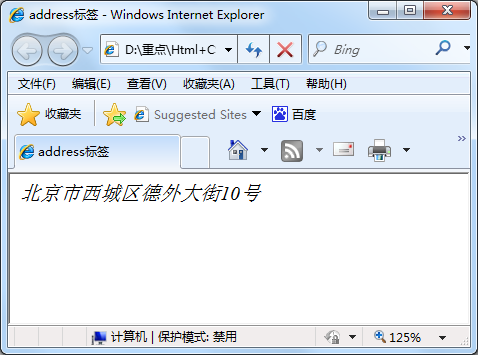
**语法：**

<address>地址信息</address>

如：

<address>北京市西城区德外大街10号</address>

      在浏览器上显示的样式为**斜体**，如果不喜欢斜体，当然可以，可以在后面的课程中使用css样式来修改它<address>标签的默认样式。

[](http://img.mukewang.com/52a917910001c16404780355.jpg)

## 5、想加入一行代码吗？使用code标签

在介绍语言技术的网站中，必免不了在网页中显示一些计算机专业的编程代码，当代码为一行代码时，你就可以使用<code>标签了，如下面例子：

<code>var i=i+300;</code>

注意：在文章中一般如果要插入多行代码时不能使用<code>标签了。

**语法：**

<code>代码语言</code>

注：如果是多行代码，可以使用<pre>标签。

## 6、使用pre标签为你的网页加入大段代码

在上节中介绍加入一行代码的标签为<code>，但是在大多数情况下是需要加入大段代码的，如下图：

[](http://img.mukewang.com/52a519bd0001441604710369.jpg)

怎么办？不会是每一代码都加入一个<code>标签吧，没有这么复杂，这时候就可以使用<pre>标签。

**语法：**

<pre>语言代码段</pre>

<pre> 标签的主要作用:预格式化的文本。被包围在 pre 元素中的文本通常会保留空格和换行符。

如下代码：

<pre>

var message="欢迎";

for(var i=1;i<=10;i++)

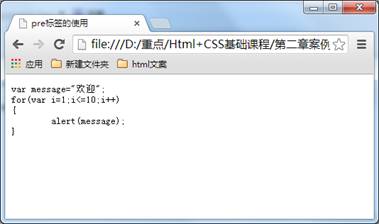
{

alert(message);

}

</pre>

在浏览器中的显示结果为：

[](http://img.mukewang.com/52a51a4f000118c403790224.jpg)

在上面的例子中可以看到代码中的空格，换行符都保留下来。如果用以前的方法，回车需要输入<br>签，空格需要输入&nbsp;

**注意：**<pre> 标签不只是为显示计算机的源代码时用的，在你需要在网页中预显示格式时都可以使用它，只是<pre>标签的一个常见应用就是用来展示计算机的源代码。

## 7、使用ol，添加图书销售排行榜

如果想在网页中展示**有前后顺序**的信息列表，怎么办呢？如，当当网上的书籍热卖排行榜，如下图所示。这类信息展示就可以使用<ol>标签来制作**有序列表**来展示。

[](http://img.mukewang.com/52d3884a00014b0702270264.jpg)

**语法：**

<ol>

<li>[信息](http://www.zuowen.com/e/20130805/51ff0eacd8d21.shtml)</li>

<li>[信息](http://www.zuowen.com/e/20130805/51ff0eacd8d21.shtml)</li>

......

</ol>

**举例：**

下面是一个热点课程下载排行榜：

<ol>

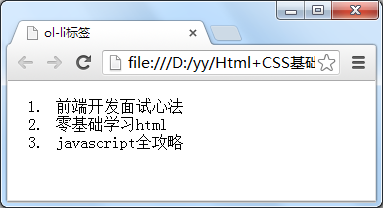
<li>前端开发面试心法 </li>

<li>[零基础学习html](http://www.zuowen.com/e/20130802/51fb0a622ef8d.shtml)</li>

<li>JavaScript全攻略</li>

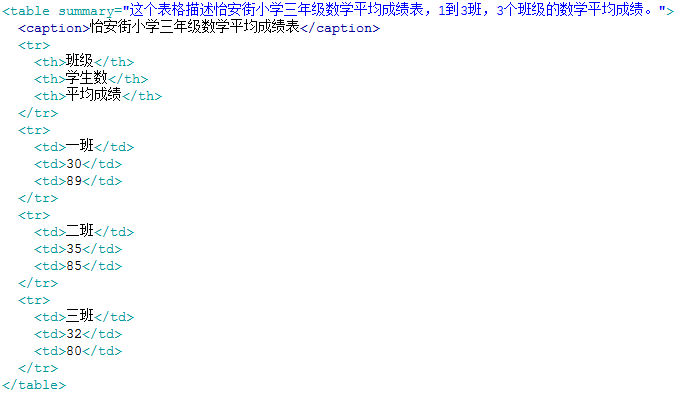
</ol>

<ol>在网页中显示的默认样式一般为：每项<li>前都自带一个序号，序号默认从1开始，如下图所示：

[](http://img.mukewang.com/52d3893400019ee003830208.jpg)

## 8、caption标签，为表格添加标题和摘要

表格还是需要添加一些标签进行优化，可以添加**标题**和**摘要**。代码如下：

[](http://img.mukewang.com/52d3a1610001a42606940398.jpg)

**摘要**

摘要的内容是不会在浏览器中显示出来的。它的作用是增加表格的可读性(语义化)，使搜索引擎更好的读懂表格内容，还可以使屏幕阅读器更好的帮助特殊用户读取表格内容。

**语法：**<table summary="表格简介文本">

**标题**

用以描述表格内容，标题的显示位置：表格上方。

**语法：**

<table>

**<caption>标题文本</caption>**

<tr>

<td>…</td>

<td>…</td>

…

</tr>

…

</table>

## 9、在新建浏览器窗口中打开链接

<a>标签在默认情况下，链接的网页是在当前浏览器窗口中打开，有时我们需要在新的浏览器窗口中打开。

如下代码：

<a href="目标网址" **target="\_blank"**>click here!</a>

## 10、使用mailto在网页中链接Email地址

<a>标签还有一个作用是可以链接Email地址，使用mailto能让访问者便捷向网站管理者发送电子邮件。我们还可以利用mailto做许多其它事情。下面一一进行讲解，请看详细图示：

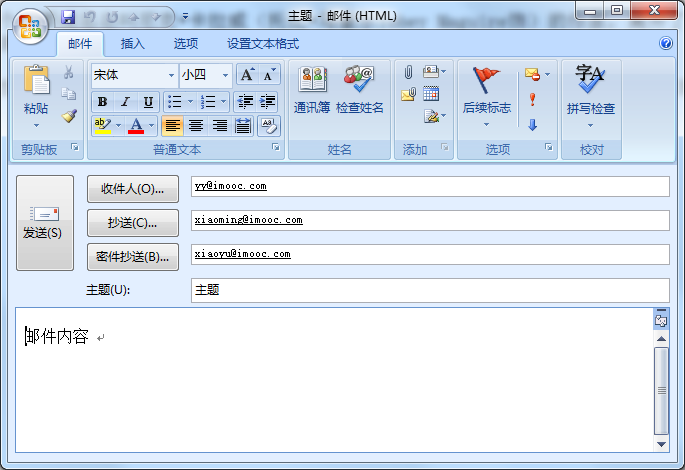
[](http://img.mukewang.com/52da4f2a000150b714280550.jpg)

**注意**：如果mailto后面同时有多个参数的话，第一个参数必须以“?”开头，后面的参数每一个都以“&”分隔。

下面是一个完整的实例:

[http://img.mukewang.com/52da52200001e00e07930061.jpg](http://img.mukewang.com/52da52200001e00e07930061.jpg)

在浏览器中显示的结果：

[](http://img.mukewang.com/52da5026000148a706850470.jpg)

任务

把右部编辑器中的第10行的“对此影评有何感想，发送邮件给我”加入链接，使其单击后可以自动发送邮件，具体要求：

1、发送人邮箱地址：yy@imooc.com。

2、邮件主题：观了不起的盖茨比有感。

3、邮件内容：你好，对此评论有些想法。

**?**不会了怎么办

1、忘记了mailto写法了？

<a href="mailto:yy@imooc.com?subject=主题名称&body=邮件内容">

**代码：**

<!DOCTYPE HTML>

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />

<title>mailto</title>

</head>

<body>

<p>1922年的春天，一个想要成名名叫卡拉威（马奎尔Tobey Maguire 饰）的作家，离开了美国中西部，来到了纽约。那是一个道德感渐失，爵士乐流行，走私为王，股票飞涨的时代。为了追寻他的美国梦，他搬入纽约附近一海湾居住。</p>

<p>菲茨杰拉德，二十世纪美国文学巨擘之一，兼具作家和编剧双重身份。他以诗人的敏感和戏剧家的想象为"爵士乐时代"吟唱华丽挽歌，其诗人和梦想家的气质亦为那个奢靡年代的不二注解。</p>

<a href="mailto:yy@imooc.com?subject=观了不起的盖茨比有感&body=你好，对此评论有些想法">对此影评有何感想，发送邮件给我</a>

</body>

</html>

## 11、使用表单标签，与用户交互

网站怎样与用户进行交互？答案是使用HTML表单(form)。表单是可以把浏览者输入的数据传送到服务器端，这样服务器端程序就可以处理表单传过来的数据。

语法：

<form method="传送方式" action="服务器文件">

**讲解：**

1.**<form> ：**<form>标签是成对出现的，以<form>开始，以</form>结束。

2.**action** **：**浏览者输入的数据被传送到的地方,比如一个PHP页面(save.php)。

3.**method** **：** 数据传送的方式（get/post）。

<form **method="post" action="save.php"**>

<label for="username">用户名:</label>

<input type="text" name="username" />

<label for="pass">密码:</label>

<input type="password" name="pass" />

</form>

**注意:**

**1、所有表单控件（文本框、文本域、按钮、单选框、复选框等）都必须放在<form></form>标签之间（否则用户输入的信息可提交不用服务器上噢！）。**

**2、method:post/get的区别这一部分内容属于后端程序员考虑的问题。感兴趣的小伙伴可以查看本小节的wiki，里面有详细介绍。**

**Wiki method="get/post",两种方式的区别**

Form中的get和post方法，在数据传输过程中分别对应了GET和POST方法。  
二者主要区别如下：

1、Get将表单中数据的按照variable=value的形式，添加到action所指向的URL后面，并且两者使用“?”连接，而各个变量之间使用“&”连接；Post是将表单中的数据放在form的数据体中，按照变量和值相对应的方式，传递到action所指向URL。  
如下形式：  
     http://www.imooc.com/test.asp?name=lilian&password=12345678  
2、Get是不安全的，因为在传输过程，数据被存放在请求的URL地址中，这样就可能会有一些隐私的信息被第三方看到。  
3、Get方式传输的数据量非常小，一般限制在 2KB 左右，但是执行效率却比 Post 方法好；而 Post 方式传递的数据量相对较大，它是等待服务器来读取数据，不过也有字节限制，这是为了避免对服务器用大量数据进行恶意攻击，根据微软方面的说法，微软对用 Request.Form()可接收的最大数据有限制，IIS4中为 80KB 字节，IIS5 中为 100KB 字节。

综上所述，请尽量用 Post 方法。

## 12、文本域，支持多行文本输入

当用户需要在表单中输入大段文字时，需要用到文本输入域。

**语法**：

<textarea **rows="行数"** **cols="列数"**>文本</textarea>

1、<textarea>标签是成对出现的，以<textarea>开始，以</textarea>结束。

2、**cols ：**多行输入域的**列数**。

3、**rows ：**多行输入域的**行数**。

4、在<textarea></textarea>标签之间可以输入**默认值**。

**举例：**

<form method="post" action="save.php">

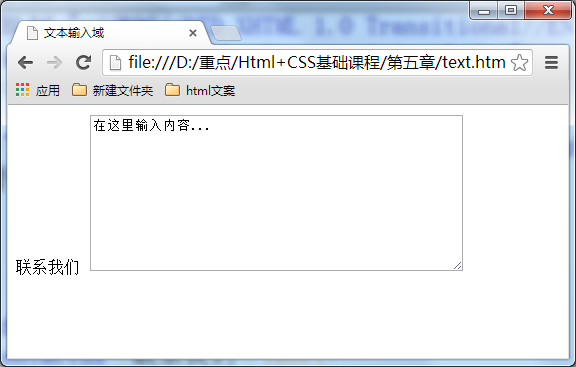
<label>联系我们</label>

<textarea cols="50" rows="10" >在这里输入内容...</textarea>

</form>

注意：代码中的<label>标签在本章5-9中讲解。

在浏览器中显示结果：

[](http://img.mukewang.com/52e5b4040001f4af05760367.jpg)

注意这两个属性可用css样式的width和height来代替：col用width、row用height来代替。（这两个css样式在以后的章节会讲解）

## 13、使用单选框、复选框，让用户选择

在使用表单设计调查表时，为了减少用户的操作，使用选择框是一个好主意，html中有两种选择框，即**单选框**和**复选框**，两者的区别是**单选框**中的选项用户只能选择一项，而**复选框**中用户可以任意选择多项，甚至全选。请看下面的例子:

语法：

<input type="radio/checkbox" value="值" name="名称" checked="checked"/>

1、**type:**

   当**type="radio"**时，控件为**单选框**

   当**type="checkbox"**时，控件为**复选框**

2、**value：**提交数据到服务器的值（后台程序PHP使用）

3、**name：**为控件命名，以备后台程序ASP、PHP使用

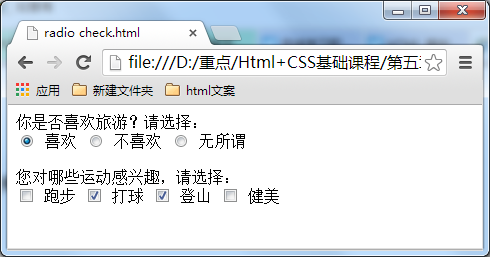
4、**checked：**当设置checked="checked"时，该选项被默认选中

如下面代码：

[](http://img.mukewang.com/52e5f7c60001a23f07360267.jpg)

注意：代码中的<label>标签在本章5-9中讲解。

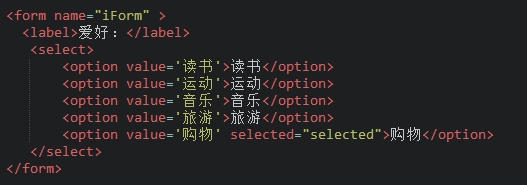
在浏览器中显示的结果：

[](http://img.mukewang.com/52e5f8010001159804900257.jpg)

注意**:同一组**的单选按钮，name取值一定要一致，比如上面例子为同一个名称“radioLove”，这样同一组的单选按钮才可以起到单选的作用。

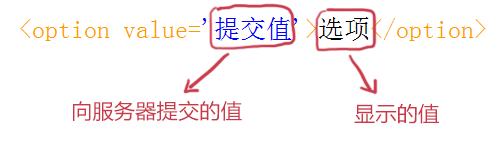
## 14、使用下拉列表框，节省空间

下拉列表在网页中也常会用到，它可以有效的节省网页空间。既可以单选、又可以多选。如下代码：

[](http://img.mukewang.com/52e604590001ae4005270185.jpg)

**讲解：**

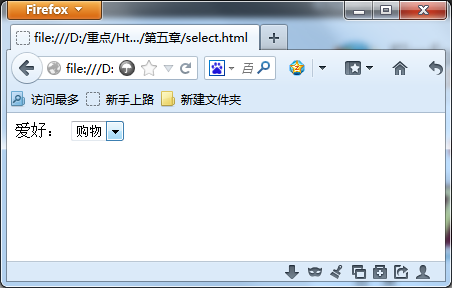
**1、value：**

[](http://img.mukewang.com/52e6037300015a9905030165.jpg)

2、**selected="selected"**：

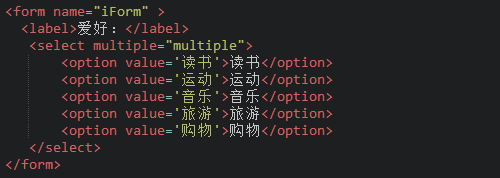
设置**selected="selected"**属性，则该选项就被默认选中。

在浏览器中显示的结果：

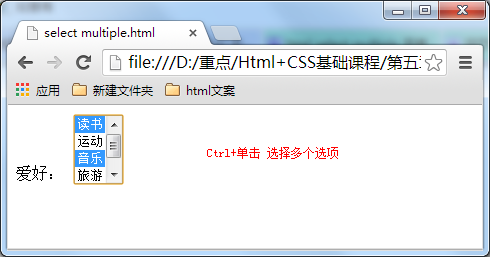
[](http://img.mukewang.com/52e605340001014804520288.jpg)

## 15、使用下拉列表框进行多选

下拉列表也可以进行多选操作，在<select>标签中设置multiple="multiple"属性，就可以实现多选功能，进行多选时按下**Ctrl**键同时进行**单击**，可以选择多个选项。如下代码：

[](http://img.mukewang.com/52e60c020001b7f805000178.jpg)

在浏览器中显示的结果：

[](http://img.mukewang.com/52e60c5d00013ced04900257.jpg)

## 16、使用提交按钮，提交数据

在表单中有两种按钮可以使用，分别为：提交按钮、重置。这一小节讲解提交按钮：当用户需要提交表单信息到服务器时，需要用到**提交按钮**。

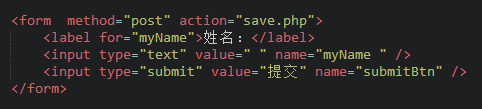
**语法**：

<input **type="submit"** value="提交">

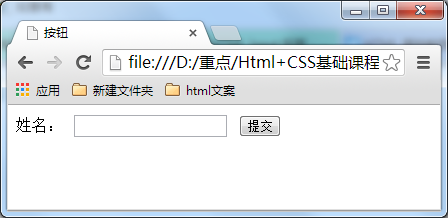
type**：只有当type值设置为submit时，按钮才有提交作用**

value**：**按钮上显示的文字

**举例：**

**[](http://img.mukewang.com/52e613350001461604820109.jpg)**

在浏览器中显示的结果：

[](http://img.mukewang.com/52e6126f0001496a04480218.jpg)

## 17、使用重置按钮，重置表单信息

当用户需要重置表单信息到初始时的状态时，比如用户输入“用户名”后，发现书写有误，可以使用重置按钮使输入框恢复到初始状态。只需要把type设置为"reset"就可以。

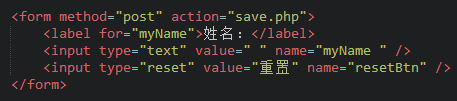
**语法**：

<input **type="reset"** value="重置">

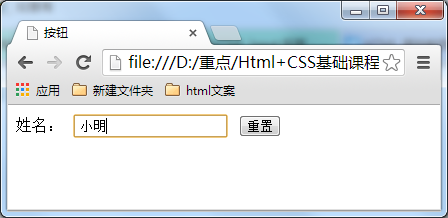
type**：只有当type值设置为reset时，按钮才有重置作用**

value**：**按钮上显示的文字

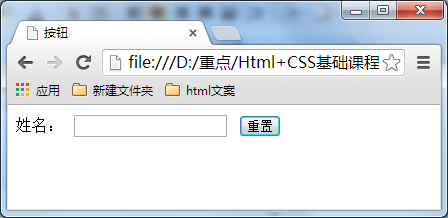
**举例：**

[](http://img.mukewang.com/52e618680001a6b204570101.jpg)

在浏览器中显示的结果：

[](http://img.mukewang.com/52e6189e000186b604480218.jpg)

输入账号

[](http://img.mukewang.com/52e618bc00015a1004480218.jpg)

单击重置按钮

### 任务

**我来试一试：把“提交按钮”改为“重置按钮”**

1、在右部编辑器中的第17行，把type="submit"改为type="reset"

2、在右部编辑器中的第17行，把value="提交"改为value="重置"

**代码：**

<!DOCTYPE HTML>

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8">

<title>重置按钮</title>

</head>

<body>

<form action="save.php" method="post" >

<label>爱好:</label>

<select>

<option value="看书">看书</option>

<option value="旅游" selected="selected">旅游</option>

<option value="运动">运动</option>

<option value="购物">购物</option>

</select>

<input type="submit" value="确定" />

<input type="reset" value="重置" />

</form>

</body>

</html>

## 18、form表单中的label标签

小伙们，你们在前面学习表单各种控件的时候有没有发现一个标签--label，这一小节就来揭晓它的作用。

label标签不会向用户呈现任何特殊效果，它的作用是为鼠标用户改进了可用性。如果你在 label 标签内点击文本，就会触发此控件。就是说，当用户单击选中该label标签时，浏览器就会自动将焦点转到和标签相关的表单控件上（就自动选中和该label标签相关连的表单控件上）。

语法：

<label for="控件id名称">

注意：标签的 for 属性中的值应当与相关控件的 id 属性值一定要相同。

例子：

<form>

<label for="male">男</label>

<input type="radio" name="sex" id="male" />

<br />

<label for="female">女</label>

<input type="radio" name="sex" id="female" />

<label for="email">输入你的邮箱地址</label>

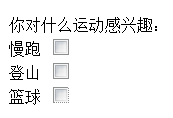
<input type="email" id="email" placeholder="Enter email">

</form>

任务

我来试试：完成下面任务

要求：慢跑、登山和篮球复选框要与各自的复选项用for属性关联。效果图如下：

[](http://img.mukewang.com/53fff8d70001fb8901730118.jpg)

备注:这一小节没有正确性验证，请查看结果窗口与任务所给的结果图片是否一致，从而判断输入代码是否正确。

**?**不会了怎么办

参考代码如下：

你对什么运动感兴趣：<br />

  <label for="jog">慢跑</label>

  <input type="checkbox" name="jog" id="jog" /><br />

  <label for="climb">登山</label>

  <input type="checkbox" name="climb" id="climb" /><br />

  <label for="basketball">篮球</label>

  <input type="checkbox" name="basketball" id="basketball" />

**代码：**

<!DOCTYPE HTML>

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8">

<title>form中的lable标签</title>

</head>

<body>

<form>

<label for="male">Male</label>

<input type="radio" name="sex" id="male" />

<br />

<label for="female">Female</label>

<input type="radio" name="sex" id="female" />

<br /><br />

你对什么运动感兴趣：

<br />

<label for=“run”>慢跑</label>

<input type="checkbox" name="run" id="run"/>

<br />

<label for="clime">登山</label>

<input type="checkbox" name="run" id=“run”/>

<br />

<label for="basketball">篮球</label>

<input type="checkbox" name="basketball" id="basketball"/>

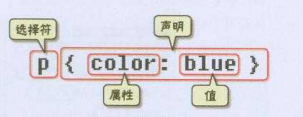
</form>

</body>

</html>

## 19、CSS代码语法

css 样式由**选择符**和**声明**组成，而**声明**又由**属性**和**值**组成，如下图所示：

[](http://img.mukewang.com/52fde5c30001b0fe03030117.jpg)

**选择符：**又称选择器，指明网页中要应用样式规则的元素，如本例中是网页中所有的段（p）的文字将变成蓝色，而其他的元素（如ol）不会受到影响。

**声明：**在英文大括号“｛｝”中的的就是声明，属性和值之间用英文冒号“：”分隔。当有多条声明时，中间可以英文分号“;”分隔，如下所示：

p{font-size:12px;color:red;}

注意：

1、最后一条声明可以没有分号，但是为了以后修改方便，一般也加上分号。

2、为了使用样式更加容易阅读，可以将每条代码写在一个新行内，如下所示：

p{

font-size:12px;

color:red;

}

## 20、内联式css样式，直接写在现有的HTML标签中

CSS样式可以写在哪些地方呢？从CSS 样式代码插入的形式来看基本可以分为以下3种：内联式、嵌入式和外部式三种。这一小节先来讲解内联式。

内联式css样式表就是把css代码直接写在现有的HTML标签中，如下面代码：

<p style="color:red">这里文字是红色。</p>

注意要写在元素的开始标签里，下面这种写法是错误的：

<p>这里文字是红色。</p style="color:red">

并且css样式代码要写在style=""双引号中，如果有多条css样式代码设置可以写在一起，中间用分号隔开。如下代码：

<p style="color:red;font-size:12px">这里文字是红色。</p>

## 21、嵌入式css样式，写在当前的文件中

现在有一任务，把右侧编辑器中的“超酷的互联网”、“服务及时贴心”、“有趣易学”这三个短词文字字号修改为18px。如果用上节课我们学习的内联式css样式的方法进行设置将是一件很头疼的事情（为每一个<span>标签加入sytle="font-size:18px"语句），本小节讲解一种新的方法嵌入式css样式来实现这个任务。

嵌入式css样式，就是可以把css样式代码写在**<style type="text/css"></style>**标签之间。如下面代码实现把三个<span>标签中的文字设置为红色：

<style type="text/css">

span{

color:red;

}

</style>

嵌入式css样式必须写在<style></style>之间，并且一般情况下嵌入式css样式写在<head></head>之间。如右边编辑器中的代码。

## 22、外部式css样式，写在单独的一个文件中

外部式css样式(也可称为外联式)就是把css代码写一个单独的外部文件中，这个css样式文件以“.css”为扩展名，在<head>内（不是在<style>标签内）使用<link>标签将css样式文件链接到HTML文件内，如下面代码：

<link href="**base.css**" rel="stylesheet" type="text/css" />

注意：

1、css样式文件名称以有意义的英文字母命名，如 main.css。

2、rel="stylesheet" type="text/css" 是固定写法不可修改。

3、<link>标签位置一般写在<head>标签之内。

## 23、三种方法的优先级

有的小伙伴问了，如果有一种情况：对于同一个元素我们同时用了三种方法设置css样式，那么哪种方法真正有效呢？在右边编辑器就出现了这种情况

1、使用内联式CSS设置“超酷的互联网”文字为粉色。

2、然后使用嵌入式CSS来设置文字为红色。

3、最后又使用外部式设置文字为蓝色（style.css文件中设置）。

但最终你可以观察到“超酷的互联网”这个短词的文本被设置为了粉色。因为这三种样式是有优先级的，记住他们的优先级：内联式 > 嵌入式 > 外部式

但是嵌入式>外部式有一个前提：嵌入式css样式的位置一定在外部式的后面。如右代码编辑器就是这样，<link href="style.css" ...>代码在<style type="text/css">...</style>代码的前面（实际开发中也是这么写的）。感兴趣的小伙伴可以试一下，把它们调换顺序，再看他们的优先级是否变化。

其实总结来说，就是--就近原则（离被设置元素越近优先级别越高）。

但注意上面所总结的优先级是有一个前提：内联式、嵌入式、外部式样式表中css样式是在的相同权值的情况下，什么是权值呢？在后面的9-2小节中会讲解到。

## 24、什么是选择器？

每一条css样式定义由两部分组成，形式如下：

选择器{

样式;

}

在{}之前的部分就是“选择器”，“选择器”指明了{}中的“样式”的作用对象，也就是“样式”作用于网页中的哪些元素。比如右侧代码编辑器中第7行代码中的“body”就是选择器。

## 25、标签选择器

标签选择器其实就是html代码中的标签。如右侧代码编辑器中的<html>、<body>、<h1>、<p>、<img>。例如下面代码：

p{font-size:12px;line-height:1.6em;}

上面的css样式代码的作用：为p标签设置12px字号，行间距设置1.6em的样式。

## 26、类选择器

类选择器在css样式编码中是最常用到的，如右侧代码编辑器中的代码:可以实现为“胆小如鼠”、“勇气”字体设置为红色。

语法：

**.**类选器名称{css样式代码;}

注意：

1、**英文圆点开头**

2、其中**类选器名称**可以任意起名（但不要起中文噢）

使用方法：

第一步：使用合适的标签把要修饰的内容标记起来，如下：

<span>胆小如鼠</span>

第二步：使用class="类选择器名称"为标签设置一个类，如下：

<span **class="stress"**>胆小如鼠</span>

第三步：设置类选器css样式，如下：

.stress{color:red;}/\***类前面要加入一个英文圆点**\*/

## 27、ID选择器

在很多方面，ID选择器都类似于类选择符，但也有一些重要的区别：

1、为标签设置id="ID名称"，而不是class="类名称"。

2、ID选择符的前面是井号**（#）**号，而不是英文圆点**（.）**。

右侧代码编辑器中就是一个ID选择符的完整实例。

## 28、类和ID选择器的区别

学习了类选择器和ID选择器，我们会发现他们之间有很多的相似处，是不是两者可以通用呢？我们不要着急先来总结一下他们的相同点和不同点：

**相同点：**可以应用于任何元素  
**不同点：**

1、**ID选择器只能在文档中使用一次**。与类选择器不同，在一个HTML文档中，ID选择器只能使用一次，而且仅一次。而类选择器可以使用多次。

下面代码是正确的：

<p>三年级时，我还是一个<span **class="stress"**>胆小如鼠</span>的小女孩，上课从来不敢回答老师提出的问题，生怕回答错了老师会批评我。就一直没有这个<span **class="stress"**>勇气</span>来回答老师提出的问题。</p>

而下面代码是错误的：

<p>三年级时，我还是一个<span **id="stress"**>胆小如鼠</span>的小女孩，上课从来不敢回答老师提出的问题，生怕回答错了老师会批评我。就一直没有这个<span **id="stress"**>勇气</span>来回答老师提出的问题。</p>

2、**可以使用类选择器词列表方法为一个元素同时设置多个样式。**我们可以为一个元素同时设多个样式，但只可以用类选择器的方法实现，ID选择器是不可以的（**不能使用 ID 词列表）。**

下面的代码是**正确**的(完整代码见右侧代码编辑器)

.stress{

    color:red;

}

.bigsize{

    font-size:25px;

}

<p>到了<span **class="stress bigsize"**>三年级</span>下学期时，我们班上了一节公开课...</p>

上面代码的作用是为“三年级”三个文字设置文本颜色为红色并且字号为25px。

下面的代码是**不正确**的(完整代码见右侧代码编辑器)

#stressid{

    color:red;

}

#bigsizeid{

    font-size:25px;

}

<p>到了<span **id="stressid bigsizeid"**>三年级</span>下学期时，我们班上了一节公开课...</p>

上面代码不可以实现为“三年级”三个文字设置文本颜色为红色并且字号为25px的作用。

## 29、子选择器

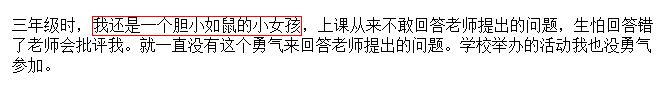
还有一个比较有用的选择器**子选择器**，即大于符号(>),用于选择指定标签元素的**子元素。**如右侧代码编辑器中的代码：

.food>li{border:1px solid red;}

这行代码会使class名为food下的子元素li加入红色实线边框。

### 任务

我也来试试，把文字内容的“我还是一个胆小如鼠的小女孩”这一句话加入红色边框。效果图如下：

[](http://img.mukewang.com/54003a0100012cc906650087.jpg)

1、在右侧代码编辑器的第8行输入代码：

.first>span{border:1px solid red;}

**代码：**

<!DOCTYPE HTML>

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8">

<title>子选择符</title>

<style type="text/css">

.food>li{border:1px solid red;}/\*添加边框样式（粗细为1px， 颜色为红色的实线）\*/

.first>span{border:1px solid red;}

</style>

</head>

<body>

<p class="first">三年级时，<span>我还是一个<span>胆小如鼠</span>的小女孩</span>，上课从来不敢回答老师提出的问题，生怕回答错了老师会批评我。就一直没有这个勇气来回答老师提出的问题。学校举办的活动我也没勇气参加。</p>

<h1>食物</h1>

<ul class="food">

<li>水果

<ul>

<li>香蕉</li>

<li>苹果</li>

<li>梨</li>

</ul>

</li>

<li>蔬菜

<ul>

<li>白菜</li>

<li>油菜</li>

<li>卷心菜</li>

</ul>

</li>

</ul>

</body>

</html>

## 30、包含(后代)选择器

**包含选择器**，即加入空格,用于选择指定标签元素下的**后辈元素。**如右侧代码编辑器中的代码：

.first span{color:red;}

这行代码会使第一段文字内容中的“胆小如鼠”字体颜色变为红色。

请注意这个选择器与子选择器的区别，子选择器（child selector）仅是指它的直接后代，或者你可以理解为作用于子元素的第一代后代。而后代选择器是作用于所有子后代元素。后代选择器通过空格来进行选择，而子选择器是通过“>”进行选择。

总结：**>**作用于元素的第一代后代，**空格**作用于元素的所有后代。

## 31、通用选择器

通用选择器是功能最强大的选择器，它使用一个（\*）号指定，它的作用是匹配html中任意标签元素，如下使用下面代码使用html中任意标签元素字体颜色全部设置为红色：

\* {color:red;}

## 32、伪类选择符

更有趣的是伪类选择符，为什么叫做伪类选择符，它允许给html不存在的标签（标签的某种状态）设置样式，比如说我们给html中一个标签元素的鼠标滑过的状态来设置字体颜色：

a:hover{color:red;}

上面一行代码就是为a标签鼠标滑过的状态设置字体颜色变红。这样就会使第一段文字内容中的“胆小如鼠”文字加入**鼠标滑过**字体颜色变为红色特效。

**关于伪选择符：**

    关于伪类选择符，到目前为止，可以兼容所有浏鉴器的“伪类选择符”就是a标签上使用：hover了（其实伪类选择符还有很多，尤其是css3中，但是因为不能兼容所有浏鉴器，本教程只是讲了这一种最常用的）。其实:hover可以放在任意的标签上，比如说p:hover，但是它们的兼容性也是很不好的，所以现在比较常用的还是a:hover的组合。

## 33、分组选择符

当你想为html中多个标签元素设置同一个样式时，可以使用分组选择符（，），如下代码为右侧代码编辑器中的h1、span标签同时设置字体颜色为红色：

h1,span{color:red;}

它相当于下面两行代码：

h1{color:red;}

span{color:red;}

## 34、继承

CSS的**某些样式**是具有继承性的，那么什么是继承呢？继承是一种规则，它允许样式不仅应用于某个特定html标签元素，而且应用于其后代。比如下面代码：如某种颜色应用于p标签，这个颜色设置不仅应用p标签，还应用于p标签中的所有子元素文本，这里子元素为span标签。

p{color:red;}

<p>三年级时，我还是一个<span>胆小如鼠</span>的小女孩。</p>

可见右侧结果窗口中p中的文本与span中的文本都设置为了红色。但注意有一些css样式是不具有继承性的。如border:1px solid red;

p{border:1px solid red;}

<p>三年级时，我还是一个<span>胆小如鼠</span>的小女孩。</p>

在上面例子中它代码的作用只是给p标签设置了边框为1像素、红色、实心边框线，而对于子元素span是没用起到作用的。

## 35、特殊性

有的时候我们为同一个元素设置了不同的CSS样式代码，那么元素会启用哪一个CSS样式呢?我们来看一下面的代码：

p{color:red;}

.first{color:green;}

<p class="first">三年级时，我还是一个<span>胆小如鼠</span>的小女孩。</p>

p和.first都匹配到了p这个标签上，那么会显示哪种颜色呢？green是正确的颜色，那么为什么呢？是因为浏览器是根据权值来判断使用哪种css样式的，权值高的就使用哪种css样式。

下面是权值的规则：

**标签的权值为1，类选择符的权值为10，ID选择符的权值最高为100。**例如下面的代码：

p{color:red;} **/\*权值为1\*/**

p span{color:green;} **/\*权值为1+1=2\*/**

.warning{color:white;} **/\*权值为10\*/**

p span.warning{color:purple;} **/\*权值为1+1+10=12\*/**

#footer .note p{color:yellow;} **/\*权值为100+10+1=111\*/**

**注意：还有一个权值比较特殊--继承也有权值但很低，有的文献提出它只有0.1，所以可以理解为继承的权值最低。**

## ****36、****层叠

我们来思考一个问题：如果在html文件中对于同一个元素可以有多个css样式存在并且这多个css样式具有**相同权重值**怎么办？好，这一小节中的层叠帮你解决这个问题。

**层叠**就是在html文件中对于同一个元素可以有多个css样式存在，当有相同权重的样式存在时，会根据这些css样式的前后顺序来决定，处于最后面的css样式会被应用。

如下面代码:

p{color:red;}

p{color:green;}

<p class="first">三年级时，我还是一个<span>胆小如鼠</span>的小女孩。</p>

最后 p 中的文本会设置为green，这个层叠很好理解，理解为后面的样式会覆盖前面的样式。

所以前面的css样式优先级就不难理解了：

**内联样式表（标签内部）> 嵌入样式表（当前文件中）> 外部样式表（外部文件中）**。

## 37、重要性

我们在做网页代码的时，有些特殊的情况需要为某些样式设置具有最高权值，怎么办？这时候我们可以使用**!important**来解决。

如下代码：

p{color:red**!important**;}

p{color:green;}

<p class="first">三年级时，我还是一个<span>胆小如鼠</span>的小女孩。</p>

这时 p 段落中的文本会显示的red红色。

**注意：!important要写在分号的前面**

这里注意当网页制作者不设置css样式时，浏览器会按照自己的一套样式来显示网页。并且用户也可以在浏览器中设置自己习惯的样式，比如有的用户习惯把字号设置为大一些，使其查看网页的文本更加清楚。这时注意样式优先级为：**浏览器默认的样式 < 网页制作者样式 < 用户自己设置的样式**，但记住!important优先级样式是个例外，权值高于用户自己设置的样式。

## 38、重要性

我们在做网页代码的时，有些特殊的情况需要为某些样式设置具有最高权值，怎么办？这时候我们可以使用**!important**来解决。

如下代码：

p{color:red**!important**;}

p{color:green;}

<p class="first">三年级时，我还是一个<span>胆小如鼠</span>的小女孩。</p>

这时 p 段落中的文本会显示的red红色。

**注意：!important要写在分号的前面**

这里注意当网页制作者不设置css样式时，浏览器会按照自己的一套样式来显示网页。并且用户也可以在浏览器中设置自己习惯的样式，比如有的用户习惯把字号设置为大一些，使其查看网页的文本更加清楚。这时注意样式优先级为：**浏览器默认的样式 < 网页制作者样式 < 用户自己设置的样式**，但记住!important优先级样式是个例外，权值高于用户自己设置的样式。

## 39、文字排版--字体

我们可以使用css样式为网页中的文字设置字体、字号、颜色等样式属性。下面我们来看一个例子，下面代码实现：为网页中的文字设置字体为宋体。

body{font-family:"宋体";}

这里注意不要设置不常用的字体，因为如果用户本地电脑上如果没有安装你设置的字体，就会显示浏览器默认的字体。（因为用户是否可以看到你设置的字体样式取决于用户本地电脑上是否安装你设置的字体。）  
现在一般网页喜欢设置“微软雅黑”，如下代码：

body{font-family:"Microsoft Yahei";}

或

body{font-family:"微软雅黑";}

注意：第一种方法比第二种方法兼容性更好一些。

因为这种字体即美观又可以在客户端安全的显示出来（用户本地一般都是默认安装的）。

## 40、文字排版--粗体

我们还可以使用css样式来改变文字的样式：粗体、斜体、下划线、删除线，可以使用下面代码实现设置文字以粗体样式显示出来。

p span{font-weight:bold;}

在这里大家可以看到，如果想为文字设置粗体是有单独的css样式来实现的，再不用为了实现粗体样式而使用h1-h6或strong标签了。

## 41、文字排版--斜体

以下代码可以实现文字以斜体样式在浏览器中显示：

p a{font-style:italic;}

<p>三年级时，我还是一个<a>胆小如鼠</a>的小女孩。</p>

## 42、文字排版--下划线

有些情况下想为文字设置为下划线样式，这样可以在视觉上强调文字，可以使用下面代码来实现：

p a{text-decoration:underline;}

<p>三年级时，我还是一个<a>胆小如鼠</a>的小女孩。</p>

## 43、文字排版--删除线

如果想在网页上设置删除线怎么办，这个样式在电商网站上常会见到：



上图中的原价上的删除线使用下面代码就可以实现：

.oldPrice{text-decoration:line-through;}

## 44、段落排版--缩进

中文文字中的段前习惯空两个文字的空白，这个特殊的样式可以用下面代码来实现：

p{text-indent:2em;}

<p>1922年的春天，一个想要成名名叫尼克卡拉威（托比?马奎尔Tobey Maguire 饰）的作家，离开了美国中西部，来到了纽约。那是一个道德感渐失，爵士乐流行，走私为王，股票飞涨的时代。为了追寻他的美国梦，他搬入纽约附近一海湾居住。</p>

注意：2em的意思就是文字的2倍大小。

## 45、段落排版--行间距

这一小节我们来学习一下另一个在段落排版中起重要作用的行间距属性（line-height），如下代码实现设置段落行间距为1.5倍。

p{line-height:1.5em;}

<p>菲茨杰拉德，二十世纪美国文学巨擘之一，兼具作家和编剧双重身份。他以诗人的敏感和戏剧家的想象为"爵士乐时代"吟唱华丽挽歌，其诗人和梦想家的气质亦为那个奢靡年代的不二注解。</p>

## 46、段落排版--字间距、字母间距

**文字间隔、字母间隔设置：**

如果想在网页排版中设置**文字间隔**或者**字母间隔**就可以使用    **letter-spacing**来实现，如下面代码：

h1{

letter-spacing:50px;

}

...

<h1>了不起的盖茨比</h1>

注意：这个样式使用在英文单词时，是设置字母与字母之间的间距。

**单词间距设置**：

如果我想设置英文单词之间的间距呢？可以使用**word-spacing**来实现。如下代码：

h1{

word-spacing:50px;

}

...

<h1>welcome to imooc!</h1>

## 47、段落排版--对齐

想为**块状元素**中的文本、图片设置居中样式吗？可以使用text-align样式代码，如下代码可实现文本居中显示。(那么什么是块状元素呢？在后面的11-1、11-2小节中会讲到。)

h1{

text-align:center;

}

<h1>了不起的盖茨比</h1>

同样可以设置居左：

h1{

text-align:left;

}

<h1>了不起的盖茨比</h1>

还可以设置居右：

h1{

text-align:right;

}

<h1>了不起的盖茨比</h1>

## 48、元素分类

在讲解CSS布局之前，我们需要提前知道一些知识，在CSS中，html中的标签元素大体被分为三种不同的类型：**块状元素**、**内联元素**和**内联块状元素**。

**常用的块状元素有：**

<div>、<p>、<h1>...<h6>、<ol>、<ul>、<dl>、<table>、<address>、<blockquote> 、<form>

**常用的内联元素有：**

<a>、<span>、<br>、<i>、<em>、<strong>、<label>、<q>、<var>、<cite>、<code>

**常用的内联块状元素有：**

<img>、<input>

## 49、元素分类--行内元素

在html中，<span>、<a>、<label>、<input>、 <img>、 <strong> 和<em>就是典型的**行内元素**（inline）元素。当然**块状元素**也可以通过代码display:inline将元素设置为**行内元素**。如下代码就是将**块状元素div**转换为**行内元素**，从页使用div元素具有**行内元素**特点。

**行内元素特点：**

1、和其他元素都在一行上；

2、元素的高度、宽度、行高及顶部和底部边距**不可**设置；

3、元素的宽度就是它包含的文字或图片的宽度，不可改变。

小伙伴们你们观查一下右侧代码段，有没有发现一个问题，行内元素之间有一个间距问题，这个问题在本小节的wiki中有介绍，感兴趣的小伙伴可以去查看。

## 50、元素分类--内联块状元素

**内联块状元素（**inline-block**）**就是同时具备内联元素、块状元素的特点，代码display:inline-block就是将元素设置为内联块状元素。(css2.1新增)，<img>、<input>标签就是这种内联块状标签。

inline-block元素特点：

1、和其他元素都在一行上；

2、元素的高度、宽度、行高以及顶和底边距都可设置。

提示：下一小节是用视频动画来讲解css中的盒模型。

## 51、盒模型--边框（一）

盒子模型的边框就是围绕着内容及补白的线，这条线你可以设置它的粗细、样式和颜色(边框三个属性)。

如下面代码为div来设置边框粗细为2px、样式为实心的、颜色为红色的边框：

div{

border:2px solid red;

}

上面是border代码的缩写形式，可以分开写：

div{

border-width:2px;

border-style:solid;

border-color:red;

}

**注意：**

1、border-style（边框样式）常见样式有：

dashed（虚线）| dotted（点线）| solid（实线）。

2、border-color（边框颜色）中的颜色可设置为十六进制颜色，如:

border-color:#888;//前面的井号不要忘掉。

3、border-width（边框宽度）中的宽度也可以设置为：

thin | medium | thick（但不是很常用），最常还是用象素（px）。

## 52、盒模型--边框（二）

现在有一个问题，如果有想为p标签单独设置下边框，而其它三边都不设置边框样式怎么办呢？css样式中允许只为一个方向的边框设置样式：

div{**border-bottom**:1px solid red;}

同样可以使用下面代码实现其它三边上、右、左边框的设置：

border-top:1px solid red;

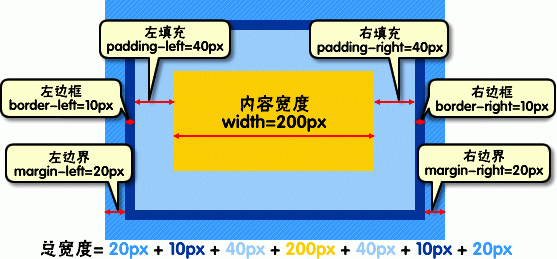
border-right:1px solid red;

border-left:1px solid red;

## 53、盒模型--宽度和高度

盒模型宽度和高度和我们平常所说的物体的宽度和高度理解是不一样的，css内定义的宽（width）和高（height），指的是填充以里的内容范围。

因此一个元素实际宽度（盒子的宽度）=左边界+左边框+左填充+内容宽度+右填充+右边框+右边界。

[](http://img.mukewang.com/539fbb3a0001304305570259.jpg)

元素的高度也是同理。

比如：

css代码：

div{

width:200px;

padding:20px;

border:1px solid red;

margin:10px;

}

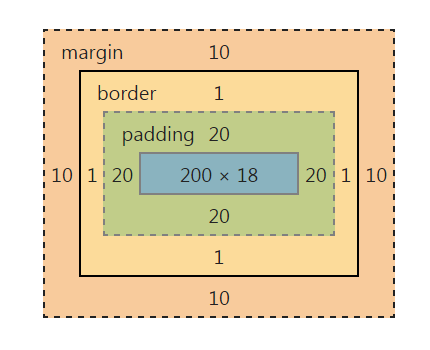
html代码：

<body>

<div>文本内容</div>

</body>

元素的实际长度为：10px+1px+20px+200px+20px+1px+10px=262px。在chrome浏览器下可查看元素盒模型，如下图：

[](http://img.mukewang.com/543b4cae0001b34304300350.jpg)

## 54、盒模型--填充

元素内容与边框之间是可以设置距离的，称之为“填充”。填充也可分为上、右、下、左(顺时针)。如下代码：

div{padding:20px 10px 15px 30px;}

顺序一定不要搞混。可以分开写上面代码：

div{

padding-top:20px;

padding-right:10px;

padding-bottom:15px;

padding-left:30px;

}

如果上、右、下、左的填充都为10px;可以这么写

div{padding:10px;}

如果上下填充一样为10px，左右一样为20px，可以这么写：

div{padding:10px 20px;}

## 55、盒模型--边界

元素与其它元素之间的距离可以使用边界（margin）来设置。边界也是可分为上、右、下、左。如下代码：

div{margin:20px 10px 15px 30px;}

也可以分开写：

div{

margin-top:20px;

margin-right:10px;

margin-bottom:15px;

margin-left:30px;

}

如果上右下左的边界都为10px;可以这么写：

div{ margin:10px;}

如果上下边界一样为10px，左右一样为20px，可以这么写：

div{ margin:10px 20px;}

总结一下：padding和margin的区别，padding在边框里，margin在边框外。

## 56、css布局模型

清楚了CSS 盒模型的基本概念、 盒模型类型， 我们就可以深入探讨网页布局的基本模型了。布局模型与盒模型一样都是 CSS 最基本、 最核心的概念。 但布局模型是建立在盒模型基础之上，又不同于我们常说的 CSS 布局样式或 CSS 布局模板。如果说布局模型是本，那么 CSS 布局模板就是末了，是外在的表现形式。   
CSS包含3种基本的布局模型，用英文概括为：Flow、Layer 和 Float。  
在网页中，元素有三种布局模型：  
1、流动模型（Flow）  
2、浮动模型 (Float)  
3、层模型（Layer）

这三个布局模型究竟是什么布局？下面几个小节会详细给大家介绍。

## 57、流动模型（一）

先来说一说**流动模型**，流动（Flow）是默认的网页布局模式。也就是说网页在默认状态下的 HTML 网页元素都是根据流动模型来分布网页内容的。

流动布局模型具有2个比较典型的特征：

第一点，**块状元素**都会在所处的**包含元素内**自上而下按顺序垂直延伸分布，因为在默认状态下，块状元素的宽度都为**100%**。实际上，块状元素都会以行的形式占据位置。如右侧代码编辑器中三个块状元素标签(div，h1，p)宽度显示为100%。

## 58、流动模型（二）

第二点，在流动模型下，**内联元素**都会在所处的包含元素内从左到右水平分布显示。（内联元素可不像块状元素这么霸道独占一行）

右侧代码编辑器中内联元素标签a、span、em、strong都是内联元素。

## 59、什么是层模型？

什么是层布局模型？层布局模型就像是图像软件PhotoShop中非常流行的图层编辑功能一样，每个图层能够精确定位操作，但在网页设计领域，由于网页大小的活动性，层布局没能受到热捧。但是在网页上局部使用层布局还是有其方便之处的。下面我们来学习一下html中的层布局。

如何让html元素在网页中精确定位，就像图像软件PhotoShop中的图层一样可以对每个图层能够精确定位操作。CSS定义了一组定位（positioning）属性来支持层布局模型。

层模型有三种形式：

1、**绝对定位**(position: absolute)

2、**相对定位**(position: relative)

3、**固定定位**(position: fixed)

## 60、层模型--绝对定位

如果想为元素设置层模型中的绝对定位，需要设置**position:absolute**(表示绝对定位)，这条语句的作用将元素从文档流中拖出来，然后使用left、right、top、bottom属性相对于其最接近的一个具有定位属性的父包含块进行绝对定位。如果不存在这样的包含块，则相对于body元素，即相对于**浏览器窗口**。

如下面代码可以实现div元素相对于浏览器窗口向右移动100px，向下移动50px。

div{

width:200px;

height:200px;

border:2px red solid;

position:absolute;

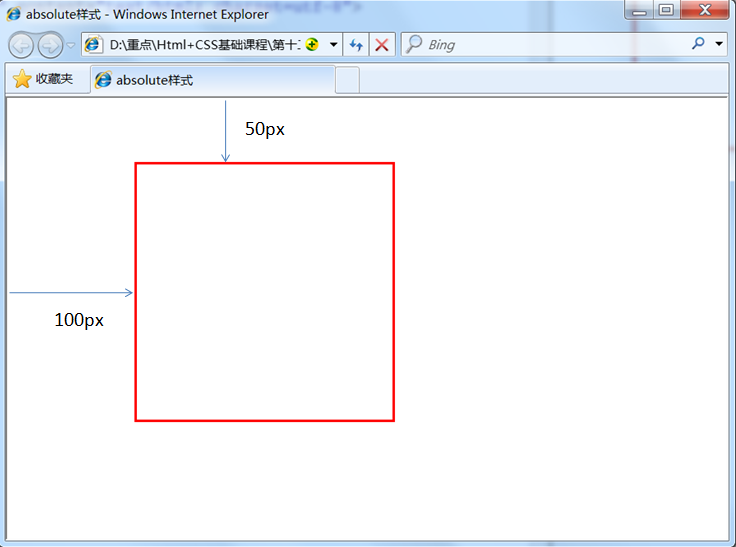
left:100px;

top:50px;

}

<div id="div1"></div>

效果如下：



## 61、层模型--相对定位

如果想为元素设置层模型中的相对定位，需要设置position:relative（表示相对定位），它通过left、right、top、bottom属性确定元素在**正常文档流中**的偏移位置。相对定位完成的过程是首先按static(float)方式生成一个元素(并且元素像层一样浮动了起来)，然后相对于**以前的位置移动，**移动的方向和幅度由left、right、top、bottom属性确定，偏移前的位置保留不动。

如下代码实现相对于以前位置向下移动50px，向右移动100px;

#div1{

width:200px;

height:200px;

border:2px red solid;

position:relative;

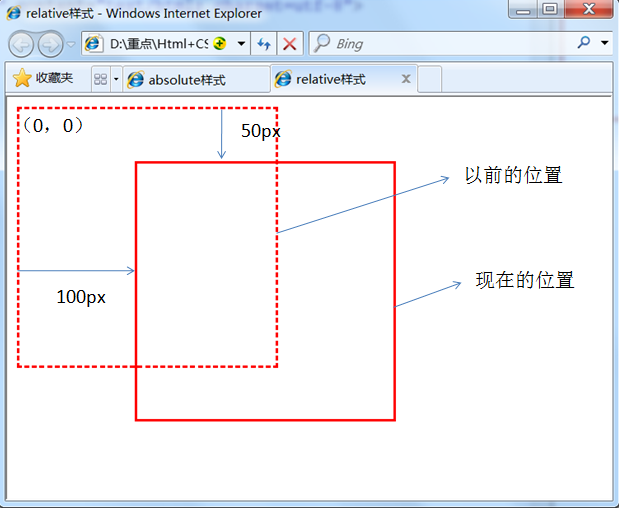
left:100px;

top:50px;

}

<div id="div1"></div>

效果图：



什么叫做“偏移前的位置保留不动”呢？

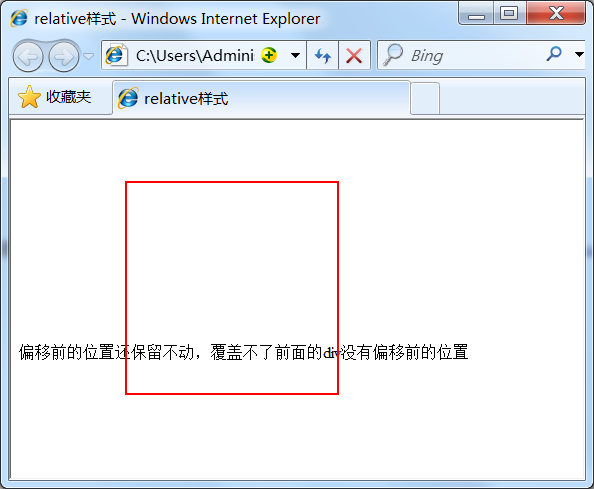
大家可以做一个实验，在右侧代码编辑器的19行div标签的后面加入一个span标签，在标并在span标签中写入一些文字。如下代码：

<body>

<div id="div1"></div><span>偏移前的位置还保留不动，覆盖不了前面的div没有偏移前的位置</span>

</body>

效果图：

[](http://img.mukewang.com/541a4bfc0001abef05940489.jpg)

从效果图中可以明显的看出，虽然div元素相对于以前的位置产生了偏移，但是div元素以前的位置还是保留着，所以后面的span元素是显示在了div元素以前位置的后面。

## 62、层模型--固定定位

fixed：表示固定定位，与absolute定位类型类似，但它的相对移动的坐标是视图（**屏幕内的网页窗口**）本身。由于视图本身是固定的，它不会随浏览器窗口的滚动条滚动而变化，除非你在屏幕中移动浏览器窗口的屏幕位置，或改变浏览器窗口的显示大小，因此固定定位的元素会始终位于浏览器窗口内视图的某个位置，不会受文档流动影响，这与background-attachment:fixed;属性功能相同。以下代码可以实现相对于**浏览器视图**向右移动100px，向下移动50px。并且拖动滚动条时位置固定不变。

#div1{

width:200px;

height:200px;

border:2px red solid;

position:fixed;

left:100px;

top:50px;

}

<p>文本文本文本文本文本文本文本文本文本文本文本文本文本文本文本文本文本文本文本文本文本文本文本文本文本文本文本文本文本文本文本文本文本文本。</p>

....

## 63、Relative与Absolute组合使用

小伙伴们学习了12-6小节的相对定位的方法：使用position:absolute可以实现被设置元素相对于浏览器（body）设置定位以后，大家有没有想过可不可以相对于其它元素进行定位呢？答案是肯定的，当然可以。使用position:relative来帮忙，但是必须遵守下面规范：

1、参照定位的元素必须是相对定位元素的前辈元素：

<div id="box1"><!--参照定位的元素-->

<div id="box2">相对参照元素进行定位</div><!--相对定位元素-->

</div>

从上面代码可以看出box1是box2的父元素（父元素当然也是前辈元素了）。

2、参照定位的元素必须加入position:relative;

#box1{

width:200px;

height:200px;

position:relative;

}

3、定位元素加入position:absolute，便可以使用top、bottom、left、right来进行偏移定位了。

#box2{

position:absolute;

top:20px;

left:30px;

}

## 这样box2就可以相对于父元素box1定位了（这里注意参照物就可以不是浏览器了，而可以自由设置了）。  64、盒模型代码简写

还记得在讲盒模型时外边距(margin)、内边距(padding)和边框(border)设置上下左右四个方向的边距是按照顺时针方向设置的：上右下左。具体应用在margin和padding的例子如下：

margin:10px 15px 12px 14px;/\*上设置为10px、右设置为15px、下设置为12px、左设置为14px\*/

通常有下面三种缩写方法:

1、如果top、right、bottom、left的值相同，如下面代码：

margin:10px 10px 10px 10px;

可缩写为：

margin:10px;

2、如果top和bottom值相同、left和 right的值相同，如下面代码：

margin:10px 20px 10px 20px;

可缩写为：

margin:10px 20px;

3、如果left和right的值相同，如下面代码：

margin:10px 20px 30px 20px;

可缩写为：

margin:10px 20px 30px;

注意：padding、border的缩写方法和margin是一致的。

## 64、颜色值缩写

关于颜色的css样式也是可以缩写的，当你设置的颜色是16进制的色彩值时，如果每两位的值相同，可以缩写一半。

例子1：

p{color:#000000;}

可以缩写为：

p{color: #000;}

例子2：

p{color: #336699;}

可以缩写为：

p{color: #369;}

## 65、字体缩写

网页中的字体css样式代码也有他自己的缩写方式，下面是给网页设置字体的代码：

body{

font-style:italic;

font-variant:small-caps;

font-weight:bold;

font-size:12px;

line-height:1.5em;

font-family:"宋体",sans-serif;

}

这么多行的代码其实可以缩写为一句：

body{

font:italic small-caps bold 12px/1.5em "宋体",sans-serif;

}

注意：

1、使用这一简写方式你至少要指定 font-size 和 font-family 属性，其他的属性(如 font-weight、font-style、font-varient、line-height)如未指定将自动使用默认值。

2、在缩写时 font-size 与 line-height 中间要加入“/”斜扛。

一般情况下因为对于中文网站，英文还是比较少的，所以下面缩写代码比较常用：

body{

font:12px/1.5em "宋体",sans-serif;

}

只是有字号、行间距、中文字体、英文字体设置。