



HTML 教程

课程内容

第 1 章	HTML 简介	3
1.1	什么是HTML	3
1.2	为什么学习HTML	3
1.3	HTML的发展史	4
第 2 章	第一个HTML文档	4
2.1	HTML结构	4
2.2	我的第一个网页	5
第 3 章	HTML元素	5
3.1	HTML标签	5
3.2	标签属性	6
	练习:	6
第 4 章	使用Dreamweaver Mx开发网页	7
第 5 章	HTML基本标签	7
5.1	段落	7
5.2	标题元素	8
5.3	换行	8
5.4	水平线	8
5.5	HTML中的注释	9
5.6	嵌入元素	9
	练习:	9
第 6 章	HTML格式化标签	10
6.1	HTML的标签	10
6.2	格式化文字:	11
6.3	删除和插入文字:	12
6.4	预格式化文本	12
6.5	其他参考:	13
6.5.1	“计算机输出”标签:	13
6.5.2	地址:	14
6.5.3	缩写和首字母缩略法:	14
6.5.4	文字方向:	14
6.5.5	块引用:	15
第 7 章	HTML链接	15
7.1	连接	15
7.2	锚标签和href属性	16
7.3	邮件链接:	17
7.4	target属性	18
	练习:	18
第 8 章	HTML表格	18



8.1	表格头使用<th>标签指定	19
8.2	有标题的表格:	19
8.3	美化表格.....	20
8.4	frame属性	20
	练习:	23
第 9 章	HTML 列表	23
9.1	无序列表.....	23
9.2	有序列表.....	24
9.3	自定义列表.....	24
	练习:	25
第 10 章	HTML 图像	25
10.1	alt属性.....	25
10.2	对齐图像:	25
10.3	图像导航:	26
10.4	图像地图:	27
	练习:	27
第 11 章	颜色表示.....	27
第 12 章	HTML 特殊字符/实体	28
第 13 章	HTML 表单	29
第 14 章	HTML 框架	32
14.1	垂直分栏:	32
14.2	水平分栏:	32
14.3	混合框架:	33
14.4	导航框架.....	33
14.5	<noframes>标签:	33
	练习:	34
14.6	内嵌框架.....	34
第 15 章	HTML 样式.....	34
15.1	内部样式表.....	35
15.2	行内样式.....	35
15.3	连接外部样式表.....	35
15.4	样式标签.....	36
第 16 章	HTML 文件头.....	36
	例 1: 给所有连接添加一个目标.....	36
	例 2: 背景音乐.....	37
16.1	HTML Meta	37
16.1.1	name属性	37
16.1.2	http-equiv属性	38
第 17 章	HTML 脚本.....	40
第 18 章	HTML4.0 事件属性	41
第 19 章	HTML 属性.....	42



参考信息

<http://www.w3schools.com/>

<http://www.netvtm.com/w3/>

相关技术手册

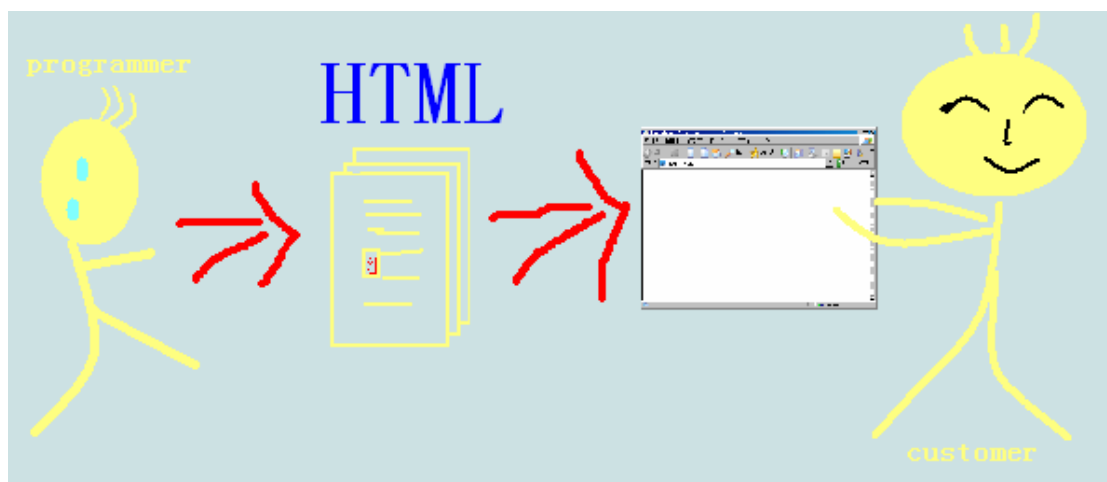
<http://www.yesky.com/imagesnew/software/html/index.html>

微软提供的在线帮助

<http://msdn.microsoft.com/library/chs/default.asp?url=/library/CHS/vbcon/html/vblrfHTMLDivControls.asp>

第1章 HTML 简介

1.1 什么是 HTML



- ✓ HTML 文件是一个包含标记的文本文件。
- ✓ HTML 语言（Hyper Text Markup Language，表示超文本标记语言）：描述信息所用的规范标签，这些标记告诉浏览器怎样显示这个页面。
- ✓ 浏览器：HTML 的翻译显示工具。
- ✓ 浏览器的显示控制：Internet 属性。

1.2 为什么学习 HTML

- ✓ Web 开发的基本技术
- ✓ 是 JSP 页面的组成部分



1.3 HTML 的发展史

- ✓ 1978 到 1986 年间,ANSI 等组织制定了 SGML (Standard Generalized Markup Language) 语言标准
- ✓ 1990,Tim Berners-Lee HTML
- ✓ 1996 年,著名的 Netscape 浏览器在其 2.0 版中增加了对 JavaApplets 和 JavaScript 的支持;
 - Microsoft 为 IE 3.0 设计了另一种后来也声名显赫的脚本语言--VBScript 语言
- ✓ 1996 年底,W3C 提出了 CSS (Cascading Style Sheets) 的建议标准
- ✓ 1996 年的 Netscape 2.0 成功地引入了对 QuickTime 插件的支持
- ✓ 1996 年,IE 3.0 正式支持在 HTML 页面中插入 ActiveX 控件的功能
- ✓ 1996 年,W3C 在 SGML 语言的基础上,提出了 XML (Extensible Markup Language) 语言草案。
- ✓ 1997 年的 Netscape 4.0 不但支持 CSS,而且增加了许多 Netscape 公司自定义的动态 HTML 标记
- ✓ 1997 年,Microsoft 发布了 IE 4.0,并将动态 HTML 标记、CSS 和动态对象模型 (DHTML Object Model) 发展成了一套完整、实用、高效的客户端开发技术体系,Microsoft 称其为 DHTML。
- ✓ 1998 年,W3C 正式发布了 XML 1.0 标准
- ✓ 1999 年,W3C 制定出了 XSLT 标准。同一年,IE 5.0 增加了对 XML 和 XSLT 的支持。
- ✓ 2000 年 1 月 26 日 XHTML1.0 成为 W3C 的推荐标准之一
- ✓ 2001 年 W3C 发布的 SVG(Scalable Vector Graphics)1.0 标准就是一种用 XML 语言表达的、全新的二维矢量图形格式

参考文档:《Web 开发技术发展史话.doc》

第2章 第一个 HTML 文档

2.1 HTML 结构

- ✓ HTML 标签是大小写无关的,W3C 在他们的 HTML4 建议中提倡使用小写标签,XHTML (下一代 HTML) 要求标签用小写。
- ✓ HTML 文件的所有内容都应该包含在<html>标记中
- ✓ HTML 语言在结构上分为两部分,即头部和主体。
 - 头部描述浏览器所需要的一些信息,如文件编码、标题等。
 - 主体则包含了文件的主体内容
- ✓ HTML 文件必须有 htm 或者 html 扩展名。



- ✓ HTML 文件可以用一个简单的文本编辑器创建。

2.2 我的第一个网页

```
<html>
<head>
<title>Title of page</title>
</head>
<body>
This is my first homepage.
<b>This text is bold</b>
</body>
</html>
```

例子解释

- `<html>`这个标签告诉浏览器这是 HTML 文档的开始。HTML 文档的最后一个标签是`</html>`，这个标签告诉浏览器这是 HTML 文档的终止。
- 在`<head>`和`</head>`标签之间文本的是头信息。在浏览器窗口中，头信息是不被显示的。
- 在`<title>`和`</title>`标签之间的文本是文档标题，它被显示在浏览器窗口的标题栏。
- 在`<body>`和`</body>`标签之间的文本是正文，会被显示在浏览器中。
- 在``和``标签之间的文本会以加粗字体显示。

注意：

浏览器的缓存功能，可用使用“刷新/重载”按钮来强迫浏览器读取编辑过的页面。

第3章 HTML 元素

- ✓ HTML 文档是由 HTML 元素组成的文本文件。
- ✓ HTML 元素是预定义的正在使用的 HTML 标签。

3.1 HTML 标签

- ✓ HTML 标签两端有两个括号字符：“<”和“>”，这两个括号字符被称为角括号。
- ✓ HTML 标签通常成对出现，比如``和``。一对标签的前面一个是开始标



签，第二个是结束标签,在开始和结束标签之间的文本是元素内容。

- ✓ 单独使用的标签不需要与之配对的结束标签，又称为空标签，例如
。W3C在他们的HTML4建议中提倡在空标签后面加上反斜杠以明确表明这是空标签，例如
。

3.2 标签属性

- ✓ 标签可以拥有属性。
- ✓ 属性进一步说明了该元素的显示或使用特性。
- ✓ 属性通常由属性名和值成对出现，属性值部分可以用英文的双引号或单引号引起来，也可以不使用任何引号。例如<body text="red">。
- ✓ 有的属性只有两种含义，存在表示使用，不存在表示不使用。例如复选框的checked属性，<input type='checkbox' checked> Apple
- ✓ 属性值本身包含引号时，使用另一种引号做定界符。例如：<input type="button" value='click this,you will get "apple"! '>
- ✓ 一个标签元素需要同时设置几个属性时，属性之间用空格分开，不区分先后顺序。
- ✓ HTML 属性名和值大小写不敏感，。

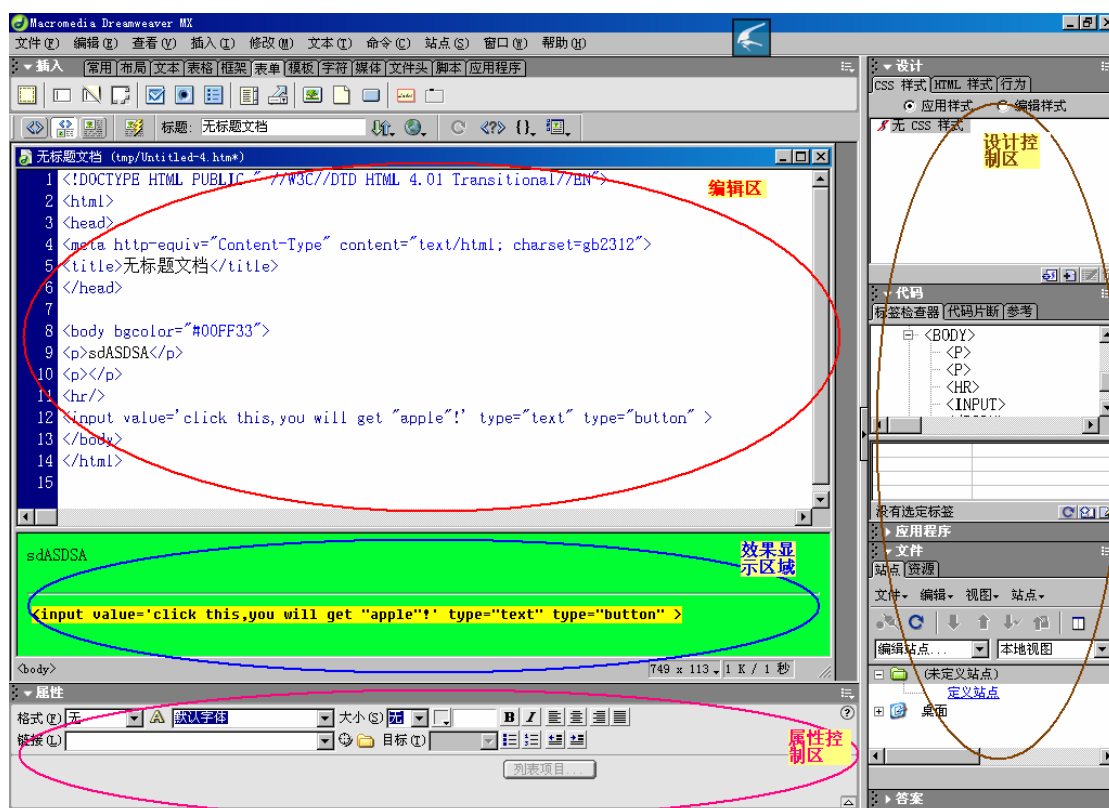
练习：

题目：设计一个网页，页面内容为 “I can design HTML page!”，背景色为黑色，字体颜色为白色。

提示：页面的背景色由 body 标签中的 bgcolor 属性控制，页面文本颜色由 text 属性控制。在 HTML 中，描述红色和蓝色的属性值为 red，blue。



第4章 使用 Dreamweaver Mx 开发网页



Dreamweaver 的优点:

- ✓ 创建页面元素
- ✓ 编码支持

第5章 HTML 基本标签

5.1 段落

段落是用<p>标签定义的。

```
<html>
<body>
<p>This is a paragraph.</p>
<p>This is a paragraph.</p>
<p>This is a paragraph.</p>
<p>Paragraph elements are defined by the p tag.</p>
```



```
</body>
</html>
```

5.2 标题元素

```
<h1>This is a heading</h1>
<h2>This is a heading</h2>
<h3>This is a heading</h3>
<h4>This is a heading</h4>
<h5>This is a heading</h5>
<h6>This is a heading</h6>
```

你认为标题是什么？

带有一定格式的段落

标题的居中属性

```
<html>
<body>
<h1 align="center">This is heading 1</h1>
<p>The heading above is aligned to the center of this page. The heading above is
aligned to the center of this page. The heading above is aligned to the center of this
page.</p>
</body>
</html>
```

5.3 换行

-
标签，能够强制换行。
-
标签是一个空标签，它没有结束标记。
- HTML 将截掉你文本中的多余空格。不管多少个空格，处理起来只当一个。一点附加信息：在 HTML 里面，一个空行也只被当作一个空格来处理。使用空段落<p>中，使用
标签来插入空白行。

```
<p>This <br> is a para<br>graph with line breaks</p>
```

5.4 水平线

<hr>标签用来分隔我们的内容。



```
<html>
<body>
<p>The hr tag defines a horizontal rule:</p>
<hr>
<p>This is a paragraph</p>
<hr>
<p>This is a paragraph</p>
<hr>
<p>This is a paragraph</p>
</body>
</html>
```

5.5 HTML 中的注释

- 注释会被浏览器忽略。
- 可以使用注释来解释代码。

```
<!-- This is a comment -->
```

注意：你需要在左括号“<”后面跟一个感叹号，右括号不用。

5.6 嵌入元素

- ✓ 音频: `<EMBED SRC="音乐文件名" AUTOSTART=true>`
- ✓ 视频: `<EMBED SRC="视频文件名" AUTOSTART=true>`

练习:

题目：编写一个页面，测试段落之间是否会用换行分割，和`
`的控制是否有显示的不同，水平线分割测试的段落。



This is the first paragraph, This is the first paragraph, This is the first paragraph, This is the first paragraph, This is the first paragraph, This is the first paragraph, This is the first paragraph, This is the first paragraph, This is the first paragraph, This is the first paragraph. |

This is second paragraph, This is second paragraph, This is second paragraph, This is second paragraph, This is second paragraph, This is second paragraph, This is second paragraph, This is second paragraph, This is second paragraph, This is second paragraph. This is second paragraph.

第6章 HTML 格式化标签

HTML 定义了很多元素用来格式化输出，比如加粗和倾斜文本。

6.1 HTML 的标签

```
<p>  
<font size="2" face="Verdana">  
This is a paragraph.
```

```
</font>
```

```
</p>
```

```
<p>
```

```
<font size="3" face="Times">
```

```
This is another paragraph.
```

```
</font>
```

```
</p>
```

字体属性

Attribute	Example	Purpose
size="number"	size="2"	定义字体大小
size="+number"	size="+ 1"	增加字体大小
size="-number"	size="- 1"	减小字体大小
face="face-name"	face="Times"	定义字体名称
color="color-value"	color="#eeff00"	定义字体颜色值
color="color-name"	color="red"	定义字体颜色名称

W3C 组织已经将标签从它的推荐中删除，将来版本的 HTML，样式表(css)将会被使用来定义布局以及显示 HTML 各元素的属性

使用正确的方法来做-使用样式

设置文字的大小、字体、颜色

```
<html>
```



```
<body>
<p style="font-family:verdana;font-size:80%;color:green">
This is a paragraph with some text in it. This is a paragraph with some text in it. This
is a paragraph with some text in it. This is a paragraph with some text in it.
</p>
</body>
</html>
```

到哪里去学习更多的样式表？

<http://www.netvtm.com/w3/css/>

6.2 格式化文字:

```
<html>
<body>
<b>This text is bold</b>
<br>
<strong>
This text is strong
</strong>
<br>
<big>
This text is big
</big>
<br>
<em>
This text is emphasized
</em>
<br>
<i>
This text is italic
</i>
<br>
This text is
<small>
small
</small>
<br>
This text contains
```



```
<sub>
subscript
</sub>
<br>
This text contains
<sup>
superscript
</sup>
</body>
</html>
```

6.3 删除和插入文字:

```
<html>
<body>
<p>
a dozen is
<del>twenty </del>
<ins>twelve </ins>
pieces
</p>
<p>
Most browsers will overstrike deleted text and underline inserted text.
</p>
<p>
Some older browsers will display deleted or inserted text as plain text.
</p>
</body>
</html>
```

6.4 预格式化文本

```
<html>
<body>
<pre>
This is
```



preformatted text.

It preserves both spaces
and line breaks.

`</pre>`

`<p>The pre tag is good for displaying computer code:</p>`

`<pre>`

for i = 1 to 10

 print i

next i

`</pre>`

`</body>`

`</html>`

预格式化有什么好处？

6.5 其他参考：

6.5.1 “计算机输出” 标签：

`<html>`

`<body>`

`<code>Computer code</code>`

`
`

`<kbd>Keyboard input</kbd>`

`
`

`<tt>Teletype text</tt>`

`
`

`<samp>Sample text</samp>`

`
`

`<var>Computer variable</var>`

`
`

`<p>`

`Note:` These tags are often used to display computer/programming code.

`</p>`

`</body>`

`</html>`



6.5.2 地址:

```
<html>
<body>
<address>
Donald Duck<br>
BOX 555<br>
Disneyland<br>
USA
</address>
</body>
</html>
```

6.5.3 缩写和首字母缩略法:

```
<html>
<body>
<abbr title="United Nations">UN</abbr>
<br>
<acronym title="World Wide Web">WWW</acronym>
<p>The title attribute is used to show the spelled-out version when holding the mouse
pointer over the acronym or abbreviation.</p>
<p>This only works for the acronym element in IE 5.</p>
<p>This works for both the abbr and acronym element in Netscape 6.2.</p>
</body>
</html>
```

6.5.4 文字方向:

```
<html>
<body>
<p>
If your browser supports bi-directional override (bdo), the next line will be written
from the right to the left (rtl):
</p>
```



```
<bdo dir="rtl">
```

Here is some Hebrew text

```
</bdo>
```

```
</body>
```

```
</html>
```

6.5.5 块引用:

```
<html>
```

```
<body>
```

Here comes a long quotation:

```
<blockquote>
```

This is a long quotation. This is a long quotation. This is a long quotation. This is a long quotation. This is a long quotation.

```
</blockquote>
```

Here comes a short quotation:

```
<q>
```

This is a short quotation

```
</q>
```

```
<p>
```

With the block quote element, the browser inserts line breaks and margins, but the q element does not render as anything special.

```
</p>
```

```
</body>
```

```
</html>
```

第7章 HTML 链接

7.1 连接

- ✓ HTML 使用超级链接来连接到网络上的其他页面。
-

```
<html>
```

```
<body>
```

```
<p>
```

```
<a href="lastpage.htm">
```



This text is a link to a page on this Web site.

</p>

<p>

This text is a link to a page on the World Wide Web.

</p>

</body>

</html>

图片作为链接:

<html>

<body>

<p>

You can also use an image as a link:

</p>

</body>

</html>

7.2 锚标签和 href 属性

- ✓ 使用命名锚以后，可以让链接直接跳转到一个页面的指定位置。
- ✓ name 属性用来创建一个命名的锚。创建一个锚的语法：Text to be displayed
- ✓ 链接到锚，在 URL 地址的后面加一个“#”和这个锚的名字：
Jump to the Section

<html>

<body>

<p>

See also Chapter 4.

</p>

<p>

<h2>Chapter 1</h2>

<p>This chapter explains ba bla bla</p>



```
<h2>Chapter 2</h2>
<p>This chapter explains ba bla bla</p>
<h2>Chapter 3</h2>
<p>This chapter explains ba bla bla</p>
<a name="C4"><h2>Chapter 4</h2></a>
<p>This chapter explains ba bla bla</p>
<h2>Chapter 5</h2>
<p>This chapter explains ba bla bla</p>
<h2>Chapter 6</h2>
<p>This chapter explains ba bla bla</p>
<h2>Chapter 7</h2>
<p>This chapter explains ba bla bla</p>
<h2>Chapter 8</h2>
<p>This chapter explains ba bla bla</p>
<h2>Chapter 9</h2>
<p>This chapter explains ba bla bla</p>
<h2>Chapter 10</h2>
<p>This chapter explains ba bla bla</p>
<h2>Chapter 11</h2>
<p>This chapter explains ba bla bla</p>
<h2>Chapter 12</h2>
<p>This chapter explains ba bla bla</p>
<h2>Chapter 13</h2>
<p>This chapter explains ba bla bla</p>
<h2>Chapter 14</h2>
<p>This chapter explains ba bla bla</p>
<h2>Chapter 15</h2>
<p>This chapter explains ba bla bla</p>
<h2>Chapter 16</h2>
<p>This chapter explains ba bla bla</p>
<h2>Chapter 17</h2>
<p>This chapter explains ba bla bla</p>
</body>
</html>
```

什么情况下使用锚？

7.3 邮件链接：

```
<html>
```



```
<body>
<p>
This is a mail link:
<a href="mailto:someone@microsoft.com?subject=Hello%20again">
Send Mail</a>
</p>
<p>
<b>Note:</b> Spaces between words should be replaced by %20 to <b>ensure</b>
that the browser will display your text properly.
</p>
</body>
</html>
```

7.4 target 属性

```
<a href="http://www.w3schools.com/" target="_blank">Visit W3Schools!</a>
_self
_top
_parent
```

练习:

题目: 创建一个搜索引擎推荐页面, 推荐你所知道的好用的搜索引擎网站, 并设置到达这些引擎网站的便捷方式, 如果页面内容超出了屏幕的显示, 则设置使用户快速返回页首的功能。

第8章 HTML 表格

- ✓ 表格是用<table>标签定义的。
- ✓ 表格边框 border 属性
- ✓ 表格被划分为行 (使用<tr>标签)
- ✓ 每行又被划分为数据单元格 (使用<td>标签)

```
<table border="1">
<tr>
<td>Row 1, cell 1</td>
<td>Row 1, cell 2</td>
</tr>
```



```
<tr>
<td>Row 2, cell 1</td>
<td>Row 2, cell 2</td>
</tr>
</table>
```

8.1 表格头使用<th>标签指定

```
<table border="1">
<tr>
<th>Heading</th>
<th>Another Heading</th>
</tr>
<tr>
<td>row 1, cell 1</td>
<td>row 1, cell 2</td>
</tr>
<tr>
<td>row 2, cell 1</td>
<td>row 2, cell 2</td>
</tr>
</table>
```

8.2 有标题的表格:

```
<html>
<body>
<h4>
This table has a caption,and a thick border:
</h4>
<table border="6">
<caption>My Caption</caption>
<tr>
<td>100</td>
<td>200</td>
<td>300</td>
</tr>
<tr>
<td>400</td>
<td>500</td>
```



```
<td>600</td>
</tr>
</table>
</body>
</html>
```

8.3 美化表格

- ✓ bgcolor: 背景色
- ✓ background: 背景图像

```
<html>
<body>
<h4>A background color:</h4>
<table border="1" bgcolor="red">
<tr>
<td>First</td>
<td>Row</td>
</tr>
<tr>
<td>Second</td>
<td>Row</td>
</tr>
</table>
<h4>A background image:</h4>
<table border="1" background="/images/bgdesert.jpg">
<tr>
<td>First</td>
<td>Row</td>
</tr>
<tr>
<td background="/images/bgdesert.jpg">Second</td>
<td>Row</td>
</tr>
</table>
</body>
</html>
```

8.4 frame 属性

```
<html>
```



```
<body>
```

```
<p>
```

If you see no frames around the tables in these examples, your browser is too old, or does not support it.

```
</p>
```

```
<h4>With frame="border":</h4>
```

```
<table frame="border">
```

```
<tr>
```

```
  <td>First</td>
```

```
  <td>Row</td>
```

```
</tr>
```

```
<tr>
```

```
  <td>Second</td>
```

```
  <td>Row</td>
```

```
</tr>
```

```
</table>
```

```
<h4>With frame="box":</h4>
```

```
<table frame="box">
```

```
<tr>
```

```
  <td>First</td>
```

```
  <td>Row</td>
```

```
</tr>
```

```
<tr>
```

```
  <td>Second</td>
```

```
  <td>Row</td>
```

```
</tr>
```

```
</table>
```

```
<h4>With frame="void":</h4>
```

```
<table frame="void">
```

```
<tr>
```

```
  <td>First</td>
```

```
  <td>Row</td>
```

```
</tr>
```

```
<tr>
```

```
  <td>Second</td>
```

```
  <td>Row</td>
```

```
</tr>
```

```
</table>
```

```
<h4>With frame="above":</h4>
```

```
<table frame="above">
```

```
<tr>
```

```
  <td>First</td>
```



```
<td>Row</td>
</tr>
<tr>
  <td>Second</td>
  <td>Row</td>
</tr>
</table>
<h4>With frame="below":</h4>
<table frame="below">
<tr>
  <td>First</td>
  <td>Row</td>
</tr>
<tr>
  <td>Second</td>
  <td>Row</td>
</tr>
</table>
<h4>With frame="hsides":</h4>
<table frame="hsides">
<tr>
  <td>First</td>
  <td>Row</td>
</tr>
<tr>
  <td>Second</td>
  <td>Row</td>
</tr>
</table>
<h4>With frame="vsides":</h4>
<table frame="vsides">
<tr>
  <td>First</td>
  <td>Row</td>
</tr>
<tr>
  <td>Second</td>
  <td>Row</td>
</tr>
</table>
<h4>With frame="lhs":</h4>
<table frame="lhs">
```



```
<tr>
  <td>First</td>
  <td>Row</td>
</tr>
<tr>
  <td>Second</td>
  <td>Row</td>
</tr>
</table>
<h4>With frame="rhs":</h4>
<table frame="rhs">
<tr>
  <td>First</td>
  <td>Row</td>
</tr>
<tr>
  <td>Second</td>
  <td>Row</td>
</tr>
</table>
</body>
</html>
```

练习:

题目：你自己的个人简介是表格形式的吗，编写一个可以发表在个人网站上的个人简介表。

第9章 HTML 列表

- HTML 支持有序、无序和自定义列表。

9.1 无序列表

- 无序列表是一个项目的序列。
- 各项目前加有标记（通常是黑色的实心小圆圈）。
- 无序列表以标签开始。
- 每个列表项目以开始。



```
<li>Coffee</li>
```

```
<li>Milk</li>
```

```
</ul>
```

在浏览器中显示如下：

- Coffee
- Milk

9.2 有序列表

- 有序列表也是一个项目的序列。
- 各项目前加有数字作标记。
- 有序列表以标签开始。
- 每个列表项目以开始。
- type 可以指定为 A、a、1、i、I

```
<ol>
```

```
<li>Coffee</li>
```

```
<li>Milk</li>
```

```
</ol>
```

在浏览器中显示如下：

1. Coffee
2. Milk

9.3 自定义列表

- 自定义列表不是一个项目的序列，它是一系列条目和它们的解释。
- 有序列表以<dl>标签开始，自定义列表条目以<dt>开始，自定义列表的定义以<dd>开始。

```
<dl>
```

```
<dt>name</dt>
```

```
<dd>zhangsan</dd>
```

```
<dt>score</dt>
```

```
<dd>100</dd>
```

```
</dl>
```




练习:

你的个人简历表中，在个人经历描述内容中使用列表是否可以使你的逻辑更清晰，更有条理。

第10章 HTML 图像

- `img` 标签和 `src` 属性
- 在 HTML 里面，图像是由 `` 标签定义的。
- `` 是空标签，它只拥有属性，而没有结束标签。
- `src` 属性的值是所要显示图像的 URL。
- 插入图像的语法: ``

如:

```

```

10.1 alt 属性

`alt` 属性用来给图像显示一个“交互文本”。`alt` 属性的值是由用户定义的。

```

```

“`alt`”属性在浏览器装载图像失败的时候告诉用户所丢失的信息，此时，浏览器显示这个“交互文本”来代替图像。给页面上的图像都加上 `alt` 属性是一个好习惯，它有助于更好地显示信息，而且，对纯文本浏览器很有用。

10.2 对齐图像:

```
<html>
<body>
<p>
An image

  in the text
</p>
<p>
An image

  in the text
</p>
<p>
```



An image

```

```

in the text

```
</p>
```

```
<p>Note that bottom alignment is the default alignment</p>
```

```
<p>
```

An image

```

```

in the text

```
</p>
```

```
<p>
```

```

```

An image before the text

```
</p>
```

```
<p>
```

An image after the text

```

```

```
</p>
```

```
</body>
```

```
</html>
```

10.3 图像导航:

```
<html>
```

```
<body>
```

```
<p>
```

Click on one of the planets to watch it closer:

```
</p>
```

```

```

```
<map id="planetmap" name="planetmap">
```

```
<area shape="rect" coords="0,0,82,126" alt="Sun" href="sun.htm">
```

```
<area shape="circle" coords="90,58,3" alt="Mercury" href="mercury.htm">
```

```
<area shape="circle" coords="124,58,8" alt="Venus" href="venus.htm">
```

```
</map>
```

```
<p>
```

Note: We use both an **id** and a **name** attribute in the map tag because some versions of Netscape don't understand the id attribute.

```
</body>
```

```
</html>
```



这个例子说明了如何创建带有可点击区域的图像地图。每个可点击区域是一个超级链接。

10.4 图像地图:

- 将鼠标在图像上移动，状态栏将显示坐标。

```
<html>
<body>
<p>
Move the mouse over the image, and look at the status bar to see how the coordinates
change.
</p>
<p>
<a href="ismap.htm">

</a>
</p>
</body>
</html>
```

练习:

假设你有一个个人站点网站，你打算怎样设计你的首页面，采用图像导航的方式是否可以更生动些，使用图像导航技术设计你的个人站点首界面

第11章 颜色表示

- ✓ <body>标签的 bgcolor 属性将背景设置为颜色，它的值可以是一个十六进制数、RGB 值或者一种颜色名称。
 - <body bgcolor="#000000">
 - <body bgcolor="rgb(0,0,0)">
 - <body bgcolor="black">
- ✓ <body>标签的 bgcolor、background 和 text 属性在最新的 HTML 标准(HTML4 和 XHTML)中已被废弃。W3C 在他们的推荐中已删除这些属性。在 HTML 的未来版本中，层叠样式表(CSS)将被用来定义 HTML 元素的布局和显示属性。



第12章 HTML 特殊字符/实体

- ✓ 有些字符，在 HTML 中有特殊的含义，因此不能在文本中使用，需要用到字符实体，比如说“<”字符。
- ✓ 一个字符实体拥有三个部分：一个 and 符号（&），一个实体名或者一个实体号，最后是一个分号（;），如小于号，我们可以这样写：<或者<
- ✓ 使用名字相对于使用数字的优点是容易记忆，缺点是并非所有的浏览器都支持最新的实体名，但是几乎所有的浏览器都能很好地支持实体号。
- ✓ 嵌入多个空格，可用使用多个 这样不会被过滤掉。

注意：实体名是大小写敏感的。

```
<html>
<body>
<p>This is a <b>character </b>entity: &#123;</p>
<p>
</body>
</html>
```

最常用的字符实体

显示结果	描述	实体名	实体号
	不可拆分的空格	 	
<	小于	<	<
>	大于	>	>
&	and 符号	&	&
"	引号	"	"
'	单引号		'

其他一些常用的字符实体

显示结果	描述	实体名	实体号
¢	分	¢	¢
£	英镑	£	£
¥	人民币元	¥	¥



§	章节	§	§
©	版权	©	©
®	注册	®	®
×	乘号	×	×
÷	除号	÷	÷

第13章 HTML 表单

- 表单是一个能够包含表单元素的区域。
 - 表单元素是能够让用户在表单中输入信息的元素（比如文本框，密码框，下拉菜单，单选框，复选框等等）。
 - 表单是用<form>元素定义的。
 - <input>标签，输入框，其类型用 type 属性指定
- 文本框

- type="text"

- 在多数浏览器中，文本框的宽度默认是 20 个字符

<input type="text" name="firstname">

注意，表单本身并不可见。另外，在多数浏览器中，文本框的宽度默认是 20 个字符。

单选按钮

<input type="radio" name="sex" value="male" checked >Male

<input type="radio" name="sex" value="female">Female

复选框

<input type="checkbox" name="bike">

I have a bike

<input type="checkbox" name="car">

I have a car

下拉列表：

<html>

<body>

<form>

<select name="cars">

<option value="volvo">Volvo

<option value="saab">Saab



```
<option value="fiat" selected>Fiat  
<option value="audi">Audi  
</select>
```

```
</form>
```

```
</body>
```

```
</html>
```

文本域:

```
<html>
```

```
<body>
```

```
<p>
```

This example demonstrates a text-area.

```
</p>
```

```
<textarea rows="10" cols="30">
```

The cat was playing in the garden.

```
</textarea>
```

```
</body>
```

```
</html>
```

这个例子说明了如何创建文本域（多行文本），用户可以在其中输入文本。在文本域中，字符个数不受限制。

创建按钮:

```
<html>
```

```
<body>
```

```
<form>
```

```
<input type="button" value="Hello world!">
```

```
</form>
```

```
</body>
```

```
</html>
```

数据周围的标题边框:

```
<html>
```

```
<body>
```

```
<fieldset>
```

```
<legend>
```

Health information:

```
</legend>
```

```
<form>
```

```
Height<input type="text" size="3">
```

```
Weight<input type="text" size="3">
```

```
</form>
```

```
</fieldset>
```



<p>

If there is no border around the input form, your browser is too old.

</p>

</body>

</html>

含有文本框和提交按钮的表单:

<html>

<body>

<form name="input" action="action.asp" method="get">

Enter your first name:

<input type="text" name="FirstName" value="Mickey">

Enter your last name:

<input type="text" name="LastName" value="Mouse">

<input type="submit" value="Submit">

<input type="reset" value="reset" name="rs">

</form>

<p>

If you click the "Submit" button, you will send your input to a new page called action.asp.

</p>

</body>

</html>

从表单发送电子邮件:

<html>

<body>

<form action="MAILTO:someone@w3schools.com" method="post"
enctype="text/plain">

<h3>This form sends an e-mail to W3Schools.</h3>

Name:

<input type="text" name="name" value="yourname" size="20">

Mail:

<input type="text" name="mail" value="yourmail" size="20">

Comment:

<input type="text" name="comment" value="yourcomment" size="40">

<input type="submit" value="Send">

<input type="reset" value="Reset">



```
</form>
</body>
</html>
```

练习:

编写一个注册网页，以收集使用你站点的用户信息，详细内容根据你站点的性质自己决定。

第14章 HTML 框架

- 在一个浏览器窗口中显示不止一个 HTML 文档。
- <frameset>标签定义了如何将窗口拆分成框架。
- 每个 frameset 标签定义了一组行和列。
- <frame>标签定义了每个框架中放入什么文件。

14.1 垂直分栏:

```
<html>
<frameset cols="25%,50%,25%">
  <frame src="frame_a.htm">
  <frame src="frame_b.htm">
  <frame src="frame_c.htm">
</frameset>
</html>
```

14.2 水平分栏:

```
<html>
<frameset rows="25%,50%,25%">
  <frame src="frame_a.htm">
  <frame src="frame_b.htm">
  <frame src="frame_c.htm">
</frameset>
</html>
```



14.3混合框架:

```
<html>
<frameset rows="50%,50%">
  <frame src="frame_a.htm">
  <frameset cols="25%,75%">
    <frame src="frame_b.htm">
    <frame src="frame_c.htm">
  </frameset>
</frameset>
</html>
```

14.4导航框架

```
<html>
<frameset cols="120,*">
  <frame src="frame_link.htm">
  <frame src="frame_a.htm" name="showframe">
</frameset>
</html>
```

在 frame_link.htm 的连接中加入 `target="showframe"`

14.5<noframes>标签:

```
<html>
<frameset cols="25%,50%,25%">
  <frame src="frame_a.htm">
  <frame src="frame_b.htm">
  <frame src="frame_c.htm">
<noframes>
<body>Your browser does not handle frames!</body>
</noframes>
</frameset>
</html>
```



练习:

分析下面页面的框架，并模拟构建该框架：



14.6 内嵌框架

- 包含在 HTML 页面里的框架

```
<html>
<body>
<iframe src="intro.htm"></iframe>
<p>Some older browsers don't support iframes.</p>
<p>If they don't, the iframe will not be visible.</p>
</body>
</html>
```

[iframe 和 frameset 的区别?](#)

第15章 HTML 样式

HTML4.0 中所有有关格式化 HTML 文档的内容都可以分离到一个样式表中
这里演示了怎样在<head>区域中添加样式信息去格式化一个 HTML 文档。



15.1 内部样式表

```
<html>
<head>
<style type="text/css">
h1 {color: red}
h3 {color: blue}
</style>
</head>
<body>
<h1>This is header 1</h1>
<h3>This is header 3</h3>
</body>
</html>
```

15.2 行内样式

无下划线的连接

```
<html>
<body>
<a href="lastpage.htm"
style="text-decoration:none">
THIS IS A LINK!
</a>
</body>
</html>
```

15.3 连接外部样式表

```
<html>
<head>
<link rel="stylesheet" type="text/css" href="styles.css" >
</head>

<body>
<h1>I am formatted with a linked style sheet</h1>
<p>Me too!</p>
</body>
</html>
```



15.4 样式标签

Tag	Description
<style>	定义样式
<link>	定义资源参考
<div>	定义文档一段区域
	同上
	[不推荐]用 style 取代
<basefont>	[不推荐]用 style 取代
<center>	[不推荐]用 style 取代

第16章 HTML 文件头

头部元素

Tag	Description
<head>	为文档定义相关信息
<title>	定义文档标题
<base>	为一个页面定义所有基础链接
<link>	定义参考源
<meta>	定义 MEAT 信息
<script>	指定页面脚本文件
<bgsound>	背景音乐

例 1：给所有连接添加一个目标

```
<html>
<head>
<base target="_blank">
</head>
<body>
<p>
<a href="http://www.w3schools.com"
target="_blank">This link</a>
will load in a new window because the target attribute is set to "_blank".
</p>
<p>
```



```
<a href="http://www.w3schools.com">
This link</a>
will also load in a new window even without a target attribute.
</p>
</body>
</html>
```

例 2：背景音乐

```
<html>
<head>
  <bgsound src="bigworld2.mp3" loop="-1">
</head>
<body>
</body>
</html>
```

Tag	Description
<!DOCTYPE>	定义文档类型.这个标签必须是在<html>之前使用

16.1HTML Meta

- HTML 在 head 元素里也包含了 meta 元素，用来给文档提供相关信息。
- meta 元素一般用来给浏览器和搜索引擎提供和你的文档内容相关的信息。

16.1.1 name 属性

16.1.1.1 这个 meta 元素定义了你的页面描述：

```
<meta name="description" content="Free Web tutorials on HTML, CSS, XML, and
XHTML">
```

16.1.1.2 定义你页面的关键字：

```
<meta name="keywords" content="HTML, DHTML, CSS, XML, XHTML,
JavaScript, VBScript">
```



16.1.1.3 记录页面作者

```
<meta name="author" content="Jan Egil Refsnes">
```

16.1.1.4 记录页面修正时间

```
<meta name="revised" content="Jan Egil Refsnes,6/10/99">
```

16.1.1.5 所用页面编辑器

```
<meta name="generator" content="Microsoft FrontPage 4.0">
```

16.1.2 http-equiv 属性

16.1.2.1 Content-Type-文档语言

```
<Meta http-equiv="Content-Type" Content="text/html; Charset=gb2312">
```

```
<Meta http-equiv="Content-Language" Content="zh-CN">
```

说明：设定页面使用的字符集，用以说明主页制作所使用的文字已经语言，浏览器会根据此来调用相应的字符集显示 page 内容。

注意：该 META 标签定义了 HTML 页面所使用的字符集为 GB2312，就是国标汉字码。如果将其中的“charset=GB2312”替换成“BIG5”，则该页面所用的字符集就是繁体中文 Big5 码。当你浏览一些国外的站点时，IE 浏览器会提示你要正确显示该页面需要下载 xx 语支持。这个功能就是通过读取 HTML 页面 META 标签的 Content-Type 属性而得知需要使用哪种字符集显示该页面的。如果系统里没有装相应的字符集，则 IE 就提示下载。其他的语言也对应不同的 charset，比如日文的字符集是“iso-2022-jp”，韩文的是“ks_c_5601”。

Content-Type 的 Content 还可以是：text/xml 等文档类型；

Charset 选项：ISO-8859-1(英文)、BIG5、UTF-8、SHIFT-Jis、Euc、Koi8-2、us-ascii、x-mac-roman、iso-8859-2、x-mac-ce、iso-2022-jp、x-sjis、x-euc-jp、euc-kr、iso-2022-kr、gb2312、gb_2312-80、x-euc-tw、x-cns11643-1、x-cns11643-2 等字符集；Content-Language 的 Content 还可以是：EN、FR 等语言代码。

16.1.2.2 Content-type

Content-type 中指定的是 MIME 类型，作用是告诉浏览器以何种方式打开文件



MIME 是 Multipurpose Internet Mail Extensions 的缩写，定义一些文件的类型，常用类型如下：

MIME 最早用于电子邮件的传输，现在用来描述文件格式

Text-文本：text/html、text/plain

Multipart-多类型：multipart/form-data

Application-应用类型：application/pdf、application/msword、application/zip

Image-图象：image/gif、image/jpg

Audio-声音：audio/mpeg

Video-影像：video/mpeg、video/x-msvideo

16.1.2.3 刷新

<Meta http-equiv="Refresh" Content="30">

<Meta http-equiv="Refresh" Content="5; Url=http://www.xia8.net">

16.1.2.4 期限

指定网页在缓存中的过期时间，一旦网页过期，必须到服务器上重新调阅。

<Meta http-equiv="Expires" Content="0">

<Meta http-equiv="Expires" Content="Wed, 26 Feb 1997 08:21:57 GMT">

16.1.2.5 Pragma-cach 模式：

禁止浏览器从本地机的缓存中调阅页面内容。

<Meta http-equiv="Pragma" Content="No-cach">

16.1.2.6 Set-Cookie (cookie 设定)

说明：浏览器访问某个页面时会将它存在缓存中，下次再次访问时就可从缓存中读取，以提高速度。当你希望访问者每次都刷新你广告的图标，或每次都刷新你的计数器，就要禁用缓存了。通常 HTML 文件没有必要禁用缓存，对于 ASP 等页面，就可以使用禁用缓存，因为每次看到的页面都是在服务器动态生成的，缓存就失去意义。如果网页过期，那么存盘的 cookie 将被删除。

用 法：<Meta http-equiv="Set-Cookie" Content="cookievalue=xxx; expires=Wednesday,21-Oct-98 16:14:21 GMT; path=/">

注意：必须使用 GMT 的时间格式



第17章 HTML 脚本

- 脚本让 HTML 更具动态性和交互性

```
<html>
<body>
<script type="text/javascript">
document.write("<h1>Hello World!</h1>")
</script>
</body>
</html>
```

- 不支持脚本的浏览器的处理

```
<html>
<body>
<script type="text/javascript">
<!--
document.write("If this is displayed, your browser supports scripting!")
//-->
</script>

<noscript>No JavaScript support!</noscript>
<p>
A browser that does not support JavaScript will show the text in the noscript element.
</p>
</body>
</html>
```

- <noscript>标签

<noscript>标签用来给不能执行的脚本来定义说明文字.这个标签是为那些不能支持<script>标签的浏览器而准备的,不执行脚本而是显示<noscript> 里的文字.

```
<script type="text/javascript">
<!--
document.write("Hello World!")
//-->
</script>
<noscript>Your browser does not support JavaScript!</noscript>
VBScript:
<script type="text/vbscript">
<!--
document.write("Hello World!")
'-->
</script>
```




<noscript>Your browser does not support VBScript!</noscript>

第18章 HTML4.0 事件属性

窗口事件

Attribute	Value	Description
onload	<i>script</i>	装载时
onunload	<i>script</i>	卸载时

表单元素事件

Attribute	Value	Description
onchange	<i>script</i>	当元素有改变时脚本会执行
onsubmit	<i>script</i>	当表单提交时
onreset	<i>script</i>	当表单重置时
onselect	<i>script</i>	元素被选中时
onblur	<i>script</i>	元素失去焦点时
onfocus	<i>script</i>	元素得到焦点时

键盘事件

在 base, bdo, br, frame, frameset, head, html, iframe, meta, param, script, style,和 title 元素里都无效.

Attribute	Value	Description
onkeydown	<i>script</i>	What to do when key is pressed[当键按下时做什么]
onkeypress	<i>script</i>	What to do when key is pressed and released[当键按下然后释放时做什么]
onkeyup	<i>script</i>	What to do when key is released[当键释放时做什么]

鼠标事件

在 base, bdo, br, frame, frameset, head, html, iframe, meta, param, script, style,和 title 元素里都无效.

Attribute	Value	Description
onclick	<i>script</i>	What to do on a mouse click [单击时...]
ondblclick	<i>script</i>	What to do on a mouse double-click [双击时...]
onmousedown	<i>script</i>	What to do when mouse button is pressed [按下时...]



onmousemove	<i>script</i>	What to do when mouse pointer moves [移动时...]
onmouseout	<i>script</i>	What to do when mouse pointer moves out of an element [鼠标移开元素时]
onmouseover	<i>script</i>	What to do when mouse pointer moves over an element [鼠标在元素上方时]
onmouseup	<i>script</i>	What to do when mouse button is released [鼠标释放时]

第19章 HTML 属性

HTML 标签有属性.各标签都有各自特殊的属性.在这里罗列的属性是其中的核心并对所有标签来讲都是标准的(个别除外):

核心属性

在 base, head, html, meta, param, script, style, 和 title 元素中无效

Attribute	Value	Description
class	<i>class_rule</i> or <i>style_rule</i>	元素类别
id	<i>id_name</i>	唯一 ID
style	<i>style_definition</i>	内样式
title	<i>tooltip_text</i>	提示