第3章

HTML与CSS基础

本章学习内容

- (1) HTML 语言概述;
- (2) HTML 的语法结构;
- (3) HTML 标签;
- (4) CSS 的基本语法;
- (5) 伪类、伪元素以及样式表的层叠顺序;
- (6) CSS 的属性。

完成本章的学习需要大约1.5小时,相关资源获取方式见前言。

知识点

HTML语言概述和基本结构常用标签CSS 的基本语法伪类、伪元素以及样式表的层叠顺序层叠样式表CSS 预处理器在 Dreamweaver 中移动 CSS 规则CSS 中的各种属性使用 HTML标签设计网页使用 CSS Designer

本章案例介绍

范例

本章范例是一个倡导阅读的网页,分为"主页"和"书类"两部分,如图 3.1 所示。通过范例的学习,掌握在 Dreamweaver 中使用 HTML 标签和 CSS 样式设计网页的技巧和方法。

模拟案例

本章模拟案例是一个综艺类的网站,主题是向往的生活,如图 3.2 所示,通过模拟练习进一步熟悉 HTML 标签和 CSS 样式的应用。

ſ



图 3.1



图 3.2

3.1 预览完成的范例

(1) 右击"lesson03/范例/03complete"文件夹中的 index. html 文件,在弹出的菜单中 选择已安装的浏览器对 index. html 文件进行浏览。

(2) 关闭浏览器。

(3)也可以用 Dreamweaver CC 2018 打开源文件进行预览。在菜单栏中选择"文件"→ "打开"按钮。选择"lesson03/范例/03complete"文件夹中的 index. html 文件,并单击"打 开"按钮,切换到"实时"视图查看页面。

3.2 HTML 语言概述

HTML 是 Hypertext Markup Language 的字母缩写,通常称作超文本标记语言,或超 文本链接标记语言,目前的版本已发展到了 HTML5。它是基于 SGML(Standard General Markup Language,标准通用标签语言)的一种描述性语言,由 W3C(World Wide Web Consortium,万维网联盟)推出,并被国际标准 ISO 8879 所认可,是用于建立 Web 页面和其 他超级文本的语言,是 WWW 的描述语言。

HTML并不是真正的程序设计语言,它只是标签语言,扩展名通常为.htm 或.html。 了解网页的用户可能听说过许多可以编辑网页的软件,事实上,用户可以用任何文本编辑器 建立 HTML 页面,如 Windows 的"记事本"程序。

HTML 文本是由 HTML 命令组成的描述性文本,可以说明文字、图形、动画、声音、表格和链接等,它独立于各种操作系统平台(如 UNIX、Windows 等)。使用 HIML 语言描述的文件,需要通过浏览器显示效果。浏览器先读取网页中的 HTML 代码,分析其语法结构,然后根据解释的结果,将单调乏味的文字显示为丰富多彩的网页内容,而不是显示事先存储于网页中的内容。正是因为如此,网页显示的速度与网页代码的质量有很大的关系,保持精简和高效的 HTML 源代码是非常重要的。

日前较为流行的 HTML5,是针对 HTML4 而言的,是 W3C 与 WHATWG(Web Hypertext Application Technology Working Group)双方合作创建的一个新版本的 HTML,其前身名为 Web Applications 1.0。HTML5 增加了更多样的 API,提供了嵌入音频、视频、图片的函数,客户端数据存储,以及交互式文档的功能。

Dreamweaver 默认的 HTML 文档类型为 HTML5。以下是 HTML5 一些有趣的特性:

- 用于绘画的 canvas 元素。
- 用于媒介回放的 video 和 audio 元素。
- 对本地离线存储的更好的支持。
- 特殊内容元素,如 article、footer、header、nav、sectlon。
- 表单控件,如 calendar、date、time、email、url、search。

HTML5 通过制定如何处理所有 HTML 元素以及如何从错误中恢复的精确规则,改进 了互操作性,并减少了开发成本。

3.3 HTML 的语法结构

标准的 HTML 由标签和文本内容构成,并用一组"<"与">"括起来,且与字母的大小写 无关。例如,

HTML的语法结构

用浏览器显示时,标签和不会被显示,浏览器在文档中发现了 这对标签,就将其中包含的文字(本例中是"HTML的语法结构")以粗 体形式显示,如图 3.3 所示。
图 3.3

需要注意的是,标签通常是成对出现的。每当使用一个标签,如< font >,则必须用另一 个标签将它关闭。但是也有一些标签例外,例如,< input >标签就不需要另一个标 签进行关闭。

严格地说,标签和标签元素不同:标签元素是位于"<"和">"符号之间的内容,如上例中的"b";而标签则包括了标签元素和"<"和">"符号本身,如上例中的""。但是,通常将标签元素和标签当作一种东西,因为脱离了"<"和">"符号的标签元素毫无意义。在后面的介绍中,若不做特别说明,则将标签和标签元素统一称作"标签"。

一般来说,HTML的语法有以下3种表达方式:

- <标签>内容</标签>
- <标签 属性 1=参数 1 属性 2=参数 2>内容</标签>
- <标签>

下面分别对这3种形式及嵌套标签进行介绍。

3.3.1 <标签>内容</标签>

这种语法结构显示了使用封闭类型标签的形式。大多数标签是封闭类型的,也就是说, 它们成对出现。所谓成对,是指一个起始标签总是搭配一个结束标签,在起始标签的标签名 前加上符号"/"便是其终止标签,如< head >与</head >。起始标签和终止标签之间的内容 受标签的控制。

例如,<i>内容</i>,<i>和</i>之间的内容受标签的控制。标签i的作用是将所控制的文本内容显示为斜体,所以在浏览器中看到的内容将是斜体字。

如果一个应该封闭的标签没有结束标签,则可能产生意想不到的错误,根据浏览器的不同,出错结果也可能不同。例如,如果在上例中,没有以标签</i>结束对文字格式的设置,可能后面所有的文字都会以斜体字的格式出现。

注意:并非所有 HTML 标签都必须成对出现,3.3.3节会具体介绍,建议在使用 HTML 标签时,最好先弄清标签是否为封闭类型。

3.3.2 <标签 属性 1=参数 1 属性 2=参数 2>内容</标签>

这种语法结构是上一种语法结构的扩展形式,利用属性进一步设置对象的外观,而参数则是设置的结果。



视频讲解

每个 HTML 标签都可以有多个属性,属性名和属性值之间用"="连接,构成一个完整的属性,例如,< body bgcolor="#FFFFFF">表示将网页背景设置为白色。多个属性之间用空格分开。例如:

< font face = "华文行楷" size = "45" color = " # 000000">小王子

上述语句表示将"小王子"的字体设置为华文行楷,字号设置为45,颜色设置为黑色,如 图 3.4 所示。

3.3.3 <标签>

前面说过,HTML标签并非都成对出现,而这种不成对出现的标签称为非封闭类型标签。 在HTML语言中,非封闭类型的标签不多,最常见的应该是换行标签
。 例如,在Dreamweaver的"代码"视图的<body>与</body>标签之间输入如下内容:

```
< font face = "楷书" size = "5" color = "♯636363">
看东西只有用心才能看得清楚,<br>重要的东西用眼睛看不见的。
</font>
```

在浏览器中的显示效果如图 3.5 所示。



使用换行标签使一行字在中间换行,显示为两行,但结构上仍属于同一个段落。

3.3.4 标签嵌套

几乎所有的 HTML 代码都是上面 3 种形式的组合,标签之间可以相互嵌套,形成更为复杂的语法。例如,如果希望将一行文本同时设置粗体和斜体格式,则可以采用下面的语句:

<i>真实的故事</i>

在浏览器中的显示效果如图 3.6 所示。 注意,尽量不要写成如下的形式:

<i>真实的故事</i>

在上面的语句中,标签嵌套发生了错误。切换到"设计"视图,可以看到显示效果如 图 3.7 所示,状态栏上的 linting 图标显示为 ③ ,表明代码中存在错误。单击标签为黄色的 文本块,在"属性"面板中可以看到相关的错误提示,提示用户这是一个无效的标签,因为这 是一个交叠的或未关闭的标签,如图 3.8 所示。单击状态栏上的 linting 图标,弹出如图 3.9 所示的"输出"面板,其中标明了错误所在的代码行和错误可能的原因。



属性		
(?)	无效的标记	标记为非法因为这是一个交迭的或未关闭的标签。
		用文本菜单重新应用此样式,然后删除这些标签。

图 3.8

搜索	输出	Git		
	行		列	省長/警告
6	0 14		13	Tag must be paired, missing: [], start tag match failed [] on line 14.
6	0 14		17	Tag must be paired, no start tag: []

图 3.9

尽管这个错误的例子在大多数浏览器中可以被正确识别,但是对于其他的一些标签,如 果嵌套发生错误的话,就不一定会被正确显示。为了保证文档有更好的兼容性,尽量不要出 现标签嵌套顺序的错误。

3.4 常用的 HTML 标签

本节将详细介绍 HTML 中常用的一些标签。掌握这些标签的用法,对今后的网页制 作可以起到事半功倍的效果。

3.4.1 文档的结构标签

在 Dreamweaver 中创建一个空白的 HTML 文档(文档类型默认为 HTML5)后,如果 切换到"代码"视图,用户会发现,尽管新建文档的"设计"视图是空白的,但是"代码"视图中 已经有了不少源代码。在默认状态下,这些源代码如下所示:



视频讲解

```
<!doctype html >
< html >
< head >
< meta charset = "utf - 8">
< title >无标题文档</title >
</head >
< body >
</body >
</html >
```

基本 HTML 页面以 doctype 开始,它声明文档的类型,主要用来说明文档使用的 XHTML 或者 HTML 的版本。浏览器根据 doctype 定义的 DTD(文档类型定义)解释页面 代码。doctype 声明必须放在每一个 XHTML 或 HTML 文档最顶部,在所有代码和标识之上,否则文档声明无效。

上面的代码包括了一个标准的 HTML 文件应该具有的 4 个组成部分。下面分别进行 简要介绍。

1. < html >标签

<html>…</html>标签是 HTML 文档的开始和结束标签,告诉浏览器这是整个

HTML 文件的范围。

HTML 文档中所有的内容都应该在这两个标签之间,一个 HTML 文档非注释代码总 是以<html>开始,以</html>结束。

2. < head >标签

位于头部的内容一般不会在网页上直接显示,而是通过另外的方式起作用。例如,在 HTML头部定义的标题不会显示在网页中,但是会出现在网页的标题栏上。

3. < title >标签

< title >…</ title >标签位于 HTML 文档的头部,即位于< head >…</ head >标签之间, 用于定义显示在浏览器窗口标题栏中的内容。

4. < body >标签

<body>…</body>标签用于定义 HTML 文档的正文部分,如文字、标题、段落和列表等, 也可以用来定义主页背景颜色。<body>…</body>定义在</head>标签之后,<html>…</html>标签之间。所有出现在网页上的正文内容都应该写在这两个标签之间。

< body >标签有 6 个常用的可选属性,主要用于控制文档的基本特征,如文本和背景颜 色等。各个属性介绍如下:

- background——为文档指定一幅图像作为背景。
- text——文档中非链接文本的默认颜色。
- link——文档中一个未被访问过的超级链接的文本颜色。
- alink——文档中一个正在打开的超级链接的文本颜色。
- vlink——文档中一个已经被访问过的超级链接的文本颜色。
- bgcolor——网页的背景颜色。

例如,如果要将文档的背景图像设置为主目录下的 1.jpg,文本颜色设置为 # BDBDBD, 未访问超级链接的文本颜色设置为 # FFFFFF,已访问超级链接的文本颜色设置为 # E5F571,正在访问的超级链接的文本颜色设置为 # F43235,则可以使用如下的< body > 标签:

< body background = "1.jpg" text = # BDBDBD link = " # FFFFFF" vlink = " # E5F571" alink = " # F43235">

在页面中输入文本,并为其创建超级链接,预览效果如图 3.10 所示。



图 3.10

3.4.2 文本格式标签

文本格式标签用于控制网页中文本的样式,如大小、字体和段落样式等。

1. < font >标签

< font >…标签用于设置文本字体格式,包括字体、字号、颜色和字形等,适当地应用可以使页面更加美观。



font 标签有 3 个属性: face、color 和 size。这 3 个属性可以自由组合,没有先后顺序。 ^{國長} 通过设置这 3 个标签属性,可以控制文字的显示效果。

• face: 用于设置文本字体名称,可以用逗号隔开多个字体名称。例如:

< font face = " Cambria, Times, serif">你好, 明天!

- size: 用于设置文本字体大小,取值范围为-7~7,数字越大字号越大。
- color:用于设置文本颜色,可以用 red、white 和 green 等助记符,也可以用十六进制 数表示,如红色为 # FF0000。

使用示例:

< font face = " 华文行楷" size = "7" color = " # DB292C">你好,明天!

上述语句在浏览器中的显示效果如图 3.11 所示。

2. 、<i>、、<h♯>标签

你好, 明天!

< b >…标签将标签之间的文本设置成粗体。
 例如:

图 3.11

文本设置为< b>粗体

• <i>····</i>标签将标签之间的文本设置成斜体。例如:

文本设置为< i>斜体</i>

• < em >…标签用于将标签之间的文字加以强调。不同的浏览器效果有所不同,通常会设置成斜体。例如:

文字加以< em >强调

上述 3 个例子在浏览器中的显示效果如图 3.12 所示。

<h♯>····</h♯>(♯=1、2、3、4、5、6)标签用于设置标题字体(Header),有 1~6 级标题,数字越大字体越小。标题将显示为黑体字。<h♯></h♯>标签自动插入一个空行,不必用标签再加空行。和<title>标签不一样,<h♯>标签中的文本将显示在浏览器中。

使用示例:

<h1>-级标题</h1>

< h2 >二级标题</h2 > < h3 >三级标题</h3 > < h4 >四级标题</h4 > < h5 >五级标题</h5 > < h6 >六级标题</h6 >

显示效果如图 3.13 所示。



初频讲解

3.4.3 排版标签

- 1. < br >、、< hr >标签
- < br>标签适用于在文本中添加一个换行符,它不需要成对使用。例如:

添加一个换行符< br>第二行

• ···标签用来分隔文档的多个段落。属性 align 有 4 个取值,分别介绍 如下:

left——段落左对齐。

center——段落居中对齐。

right——段落右对齐。

justify——段落两端对齐,即每行都有相等的宽度(就像在报纸和杂志中)。 例如:

<palign="left">左对齐</palign="left">		7
<palign="center">居中对齐</palign="center">		
<palign="right">右对齐</palign="right">		
<palign="justify">两端对齐</palign="justify">		
		_
• < br >标答用于在页面中添加一条水平约	专 例如.	内

水平线上<hr>水平线下

上述 3 个例子在浏览器中的显示效果如图 3.14 所示。

左对介	
居中对齐	
	右对齐
两端对齐	
水平线上	
水平线下	

2. < sub >和< sup >标签

• < sub >…</ sub >标签将标签之间的文本设置成下标。例如:

123 是下标< sub > 123 </ sub >

• < sup >…</ sup >标签将标签之间的文本设置成上标。例如:

456 是上标< sup > 456 </ sup >

上述两个例子在浏览器中的显示效果如图 3.15 所示。

3. < div >和< span >标签

123是下标123 456是上标456

- 图 3.15 • < div >…</div >用于块级区域的格式化显示。该标签可 以把文档划分为若干部分,并分别设置不同的属性值,使同一文字区域内的文字显 示不同的效果,常用于设置 CSS 样式。
 - < div >标签的常用格式如下:

<div align = 对齐方式 id = 名称 style = 样式 class = 类名 nowrap >...</div>

其中,对齐方式可以为 center、left 和 right, id 用于定义 div 区域的名称, style 用于定义 样式, class 用于赋予类名, nowrap 说明不能换行(默认是不加 nowrap 的,也就是可以换行)。例如:

```
<div>地球可不是普通的地球哦!</div>
<divalign="center" style="color: #F51D21" id="nav">这里有一百一十一位国王(当然包括了
非洲的国王),七千多个地理学家……</div>
```

上例在浏览器中的显示效果如图 3.16 所示。

```
地球可不是普通的地球哦!
这里有一百一十一位国王(当然包括了非洲的国王),七千多个地理学家・・・・・
```

图 3.16

• < span >…标签用于定义内嵌的文本容器或区域,主要用于一个段落、句子 甚至单词中。其格式为

< span id = 名称 style = 样式 class = 类名>.../span>

< span >标签没有 align 属性,其他属性的意义和< div >标签类似。< span >标签同样在 样式表的应用方面特别好用,它们都用于 HTML。例如:

 span 标签

< div >标签和< span >标签的区别在于,< div >标签是块级元素,可以包含段落、标题、 表格,甚至章节、摘要和备注等; 而< span >标签是行内元素,< span >的前后是不会换行的, 它纯粹地应用样式。下面以一个小实例来说明两者的区别。

(1) 新建一个 HTML 文档,并设置文档的背景图像。

(2) 切换到"代码视图",在< body >和</body >标签之间输入以下代码:

```
< span >第一个 span </ span >
< span >第二个 span </ span >
< span >第三个 span </ span >
< div >第一个 div </ div >
< div >第二个 div </ div >
< div >第三个 div </ div >
```

(3)保存文件,在浏览器中查看,结果如图 3.17 所示。

第一个span 第二个span 第三个span	
第一个div	
第二个div	
第三个div	

图 3.17



3.4.4 列表标签

在 HTML 中,列表标签分为无序列表、有序列表和普通列表 3 种。下面分别对它们进行简要介绍。

1. 无序列表

所谓无序列表,是指列表项之间没有先后次序之分。…用来标记无序列表的开始和结束。列表的标签格式为…,其中的每个标签都表示一个列表项值。

例如:

```
        <1i>夜航
        人的大地
        小王子
```

上例在浏览器中的显示效果如图 3.18 所示。

2. 有序列表

有序列表与普通列表不同之处在于有序列表存在序号。···· 用于标记有序列表的开始和结束。有序列表有一个属性 type,它取不同值 时对应的功能如下:

•	夜航
٠	人的大地
•	小王子

图 3.18

- type=1 表示用数字给列表项编号,这是默认设置。
- type=a 表示用小写字母给列表项编号。
- type=A 表示用大写字母给列表项编号。
- type=i表示用小写罗马字母给列表项编号。
- type=I 表示用大写罗马字母给列表项编号。

例如:

```
夜航
人的大地
小王子
```

该示例在浏览器中的显示效果如图 3.19 所示。

3. 普通列表

普通列表通过<dl>---<dd>----</dl>>的形式实现,通常用于排版。其中,<dl>---</dl>用于创建一个普通列表;<dl>----</dl>用于创建列表中的上层项目;<dd>----</dl>用于创建列表中最下层的项目。<dt>----</dl>和<dd>----</dd>

例如:

< dl >
<dt>圣埃克苏佩里</dt>
< dd >夜航
< dd >人的大地
< dd >小王子
<dt>圣埃克苏佩里</dt>
< dd >夜航
<dd>人的大地</dd>
< dd >小王子

该示例在浏览器中的显示效果如图 3.20 所示。

1. 夜航 2. 人的大地 3. 小王子	圣埃克苏佩里 夜航 人的大地 小王子 圣埃克苏佩里 夜航 人的大地 小王子
图 3.19	图 3.20

3.4.5 表格标签

通过表格可以将数据内容分门别类地显示出来,从而使网页显得整齐美观。

1. 标签

表格由…标签构成。标签还有很多属性用于控制表格的显示效果。表格的常用属性介绍如下:



- align——设置表格与页面对齐的方式,取值有 left、center 和 right。
- cellpadding——设置表格单元格内数据和单元格边框之间的边距,以像素为单位。
- cellspacing——设单元格之间的间距,以像素为单位。
- border——设置表格的边框,如果将该属性的值设置为 0,则不显示表格的边框线。
- width——设置表格的宽度,单位默认为像素,也可以使用百分比形式。
- height——设置表格的高度,单位默认为像素,也可以使用百分比形式。

例如,下面的代码新建了一个宽为400px、边框为1像素量、边距和间距均为2px的4行4列的表格。

```
 
  & nbsp; 
 
 
>


>
  & nbsp; 

 & nbsp;
```

2. 、和标签

(1) ···标签用于标记表格一行的开始和结束。常用的属性介绍如下:

- align——设置行中文本在单元格内的对齐方式,取值有 left、center 和 right。
- bgcolor——设置行中单元格的背景颜色。

(2) …标签用于标记表格内单元格的开始和结束,应位于标签内部。 常用的属性介绍如下:

- align——设置单元格内容在单元格内的对齐方式,取值有 left、center 和 right。
- bgcolor——设置单元格的背景颜色。

- width——设置单元格的宽度,单位为像素。
- height——设置单元格的高度,单位为像素。

(3) …的作用与大致相同,主要用于标记表格内表头的开始和结束, 且其中的文本自动以粗体显示。常用的属性如下:

- colspan——设置····内的内容应该跨越几列。
- rowspan——设置…内的内容应该跨越几行。

3. < colspan >和< rowspan >标签

< colspan >和< rowspan >标签用于合并单元格,分别表示跨多列合并和跨多行合并。 例如下面的代码:

```
>
姓名
各分数
小花
80 
98 
92 
>
小米
95 
86 
69 
小红
75 
99 
 87
```

该表格包含4行4列,其中第一行设置了跨3列的合并形式,在浏览器中的效果如图3.21 所示。

姓名		各分	数
小花	80	98	92
小米	95	86	69
小红	75	99	87

图 3.21

3.4.6 表单标签

表单是 HTML 文档中用于向用户显示信息,同时获取用户输入信息的网页元素。当数据输入完毕后,单击"提交"按钮,即可将表单内的数据提交到服务器,服务器再根据输入的数据做相应的处理。表单的应用相当广泛,登录注册、网上查询等功能都离不开表单。

1. < form >标签

< form >…</form >标签用于表示一个表单的开始与结束,并且通知服务器处理表单的 内容。表单中的各种表单对象都要放在这两个标签之间。< form >标签的常用属性介绍 如下:

- name——用于指定表单的名称。
- action——指定提交表单后,将对表单进行处理的文件路径及名称(即 URL)。
- method——用于指定发送表单信息的方式,有GET方式(通过URL发送表单信息)和POST方式(通过HTTP发送表单信息)。其中POST方式适合传递大量数据,但速度较慢;GET方式适合传送少量数据,但速度快。

2. < input >标签

<input>标签用于在表单内放置表单对象,此标签不需成对使用。它有一个 type 属性,对于不同的 type 值,<input>标签有不同的属性。

(1)当 type="text"(文本域表单对象,在文本框中显示文字)或 type="password"(密码域表单对象,在文本框中显示*号代替输入的文字,起保密作用)时,<input>标签的属性如下:

- size——文本框在浏览器中的显示宽度,实际能输入的字符数由 maxlength 参数 决定。
- maxlength——在文本框中最多能输入的字符数。

(2)当 type="submit"(提交按钮,用于提交表单)或 type="reset"(重置按钮,用于清 空表单中已输入的内容)时,<input>标签的属性如下:

• value——在按钮上显示的内容。

(3) 当 type="radio"(单选按钮)或 type="checkbox"(复选框)时,<input>标签的属性如下:

- value——用于设定单选按钮或复选框的值。
- checked——可选参数,若带有该参数,则默认状态下该按钮是选中的。同一组 radio 单选按钮(name 属性相同)中最多只能有一个带 checked 属性。复选框则无此 限制。
- (4) 当 type="image"(图像)时,<input>标签的属性如下:
- src——图像文件的名称。
- alt——图像无法显示时的替代文本。
- align——图像的对齐方式,取值可以是 top、left、bottom、middle 或 right。
- 3. < select >和< option >标签

< select >····</ select >标签用于在表单中插入一个列表框对象。它与< option >····</ option > 标签一起使用,< option >标签用于为列表框添加列表项。



(1) < select >标签的常用属性简要介绍如下:

- name——指定列表框的名称。
- size——指定列表框中显示多少列表项(行),如果列表项数目大于 size 参数值,那么通过 滚动条来滚动显示。
- multiple——指定列表框是否可以选中多项,默认情况下只能选择一项。

(2) < option >标签有两个可选参数,介绍如下:

- selected——用于设定在初始时本列表项是被默认选中的。
- value——用于设定本列表项的值,如果不设此项,则默认为标签后的内容。

在 Dreamweaver 的"代码"视图的< body >...</body >标签之间输入以下代码:

```
< form method = "post">
< select name = "balls" size = "3" multiple>
< option selected >足球</option >
< option selected >羽毛球</option >
< option selected >乒乓球</option >
< option selected >羽毛球</option >
</select >
< input type = "submit">< input type = "reset">
</form >
```

保存文档,并按 F12 键在浏览器中查看显示效果,如图 3.22 所示。

4. < textarea >标签

<textarea >····<textarea >标签的作用与<input >标签的type 属性 值为text时的作用相似,二者的不同之处在于,<textarea >显示的是多 行多列的文本区域,而<input >文本框只有一行。<textarea >和 </textarea >之间的文本是文本区域的初始文本。

| 足球
羽毛球
兵乓球 | • | |
|------------------|---|----|
| 提交查 | 询 | 重置 |

图 3.22

<textarea>标签的常用属性如下:

- name——指定文本区域的名称。
- rows——文本区域的行数。
- cols——文本区域的列数。
- wrap——用于设置是否自动换行,取值有 off(不换行,是默认设置)、soft(软换行)和 hard(硬换行)。

在 Dreamweaver 的"代码"视图的< body >…</body >标签之间输入以下代码:

```
< form method = "post">
< textarea name = "content" rows = "5" cols = "30">
个人简介
</textarea >< br >
< input type = "submit">< input type = "reset">
</form >
```

保存文档,并按 F12 键在浏览器中预览显示效果,如图 3.23 所示。



图 3.23



3.4.7 其他标签

1. < img >标签

图像可以使页面更加生动美观。在 HTML 文本中可以通过< img >标签来为页面插入 图像,该标签除 src 属性不可缺少以外,其他属性均为可选项。其属性如下所示:

- src——用于指定要插入图像的地址和名称。
- alt——用于设置当图像无法显示时的替换文本。
- width 和 height——用于设置图片的宽度和高度。

2. <a>标签

HTML 最显著的优点就在于它支持文档的超级链接,可以很方便地在不同文档以及同一文档的不同位置之间跳转。HTML 是通过链接标签<a>实现超级链接的,<a>标签是封闭性标签,其起止标签之间的内容即为锚标。<a>标签有两个不能同时使用的属性 href 和 name,此外还有 target 属性等,分别介绍如下。

• href——用于指定目标文件的 URL 地址或页内锚点。

当超级链接要链接到页内锚点时,应采用如下格式:

< a href = " # ... ">...

href 后的第一个省略号为命名锚点的名字。

<a>标签使用此属性后,在浏览器中单击锚标,页面将跳转到指定的页面或本页中指定的锚点位置。例如:

< a href = "url">单击这里

表示当单击链接文本"单击这里"时,会打开 url 所指向的文件页面。

百度

表示当单击链接文本"百度"时,将会打开 http://www.baidu.com 网页。

• name——用于标识一个目标,该目标终点在一个文件中指定。例如:

< a name = "name"> text

• target——用于设定打开新页面所在的目标窗口。如果当前页面使用了框架技术, 还可以把 target 设置为框架名。例如:

< a href = "http://www.baidu.com" target = "main">锚点链接

表示当单击链接文字"锚点链接"时,将链接的页面在名为 main 的框架中打开。

3. < meta >标签

<meta>标签是实现元数据的主要标签,它能够提供文档的关键字、作者、描述等多种 信息。在 HTML 的头部可以包括任意数量的<meta>标签。<meta>标签是非成对使用的 标签,常用属性介绍如下:

- name——用于定义一个元数据属性的名称。
- content——用于定义元数据的属性值。
- http-equiv——可以用于替代 name 属性,HTTP 服务器可以使用该属性从 HTTP 响 应头部收集信息。
- charset——用于定义文档的字符解码方式。
- 4. < link >标签

< link >标签定义了文档的引用关系,通常用于链接外部样式表。它把 CSS 写到一个扩展名为.css 的文件中,主要用于多个页面排版风格的统一控制,避免单个页面重复地设置 CSS 样式。< link >是一个非封闭性标签,只能在< head >…</ head >中使用。

在 HTML 的头部可以包含任意数量的< link >标签。< link >标签有很多属性,下面介绍一些常用的属性:

- href——用于设置链接资源所在的 URL。
- title——用于描述链接关系的字符串。
- rel——用于定义文档和所链接资源的链接关系,可能的取值有 Alternate(定义交替出现的链接)、Stylesheet(定义一个外部加载的样式表)、Start(通知搜索引擎,文档的开始)、Next(记录文档的下一页,以告知浏览器下面要加载的页面)、Prev(记录文档的上一页,即定义浏览器的后退键)、Index(当前文档的索引)、Glossary(词汇)、Copyright(当前文档的版权)、Chapter(当前文档的章节)、Section(作为文档的一部分)、Subsection(作为文档的一部分)、Appendix(定义文档的附加信息)、Help(链接帮助信息)、Bookmark(书签)等。如果希望本页指定不止一个链接关系,可以在这些值之间用空格隔开。
- rev—用于定义文档和所链接资源之间的反向关系。其可能的取值与 rel 属性相同。

例如,下面的语句链接了一个外部的样式表:

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

5. < bgsound >标签

< bgsound >标签常用于在网页中添加背景音乐,常用的两个属性为 src 属性和 loop 属性。

- src 属性用于设置要加载的背景音乐的 URL 地址。
- loop 属性用于设置背景音乐播放的循环次数。当 loop 属性值设置为-1 时,表示背景音乐无限循环播放,直到页面被关闭。

例如,
bgsound src="毛不易 - 无问.mp3" loop="3"></bgsound>表示添加网页相
同目录下的"毛不易 - 无问.mp3"文件作为背景音乐,并设置循环播放次数为3次。

6. < style >标签

< style >…</ style >标签用于在网页中创建样式(也叫嵌入样式表),它把 CSS 直接写 入 HTML 的 head 部分,这是 CSS 最为典型的使用方法。在制作网站时不建议这样使用, 应将网页结构与样式分离,便于维护。

在< style >标签中可以创建多个不同的命名样式。文档内容可以直接运用这些定义好 的样式。例如:

```
< style type = "text/css">
body {
background - color: #B39797;
}
a:link{
color: #FFFFF0;
 text - decoration: none;
}
a:visited{
color: #8B8963;
 text - decoration: none;
}
a:hover{
color: #F80004;
 text - decoration: none;
}
a:active{
color: #166016;
text - decoration: none;
}
</style>
```

在上面的代码中,<stvle>标签定义了5个样式,分别用于设置页面背景颜色、未访问过 的链接颜色、已访问过的链接颜色、当前链接和活动链接颜色。如果要在文本中应用上述样 式,可以在文字修饰标签中应用 class 属性和属性值。例如:

< font class = "a:active">我的主页

3.5 CSS 的基本语法

视频讲解

CSS 语句是内嵌在 HTML 文档内的,所以编写 CSS 的方法和编写 HTML 文档的方法 是一样的,可以用任何一种文本编辑工具来编写,比如 Dreamweaver、Windows 下的记事 本和写字板以及专门的 HTML 文本编辑工具(Frontpage、UltraEdit 等)。

CSS 的代码都是由一些最基本的语句构成的,其基本语句的语法如下:

selector{property:value;}

其中, property; value 指的是样式表定义, property 表示属性, value 表示属性值,属性与属

性值之间用冒号(:)隔开,各个属性之间用分号(;)隔开,因此以上语法也可以表示如下:

选择符{属性1:属性值1;属性2:属性值2;}

selector 是选择符,一般都是定义 HTML 样式的标签,比如 table、body、p 等,代码示例 如下:

p {font - size: 48; font - style: bold; color: red}

这里 p 表示定义段落内的格式, font-size、font-style 和 color 是属性,分别用于定义 p 中字体的大小(size)、样式(style)和颜色(color),而 48、bold、red 是属性值,意思是以 48pt、 粗体、红色的字体样式显示该段落。

3.6 伪类、伪元素及样式表的层叠顺序

3.6.1 伪类和伪元素

一般来说,选择符可以和多个类采用捆绑的形式来设定,不过这样虽然能够为同一个选择符创建多个样式,但同时也限制了所设定的类不能被其他选择符所使用。伪类的产生就是为了解决这个问题,每个预声明的伪类都可以被所有的 HTML 标识符引用,当然有些块级内容的设置除外。

伪类和伪元素是 CSS 中特殊的类和元素,它们能够自动被支持 CSS 的浏览器所识别。 伪类可以用于文档状态的改变、动态的事件等,例如,visited links(已访问链接)和 active links(可激活链接)描述了两个定位锚(anchors)的类型。伪元素指元素的一部分,例如,段 落的第一个字母。

伪类或伪元素规则的形式有两种,分别如下:

选择符:伪类{属性:属性值} 选择符:伪元素{属性:属性值}

CSS 类也可以与伪类、伪元素一起使用,有两种表示方式,分别如下:

选择符.类:伪类{属性:属性值} 选择符.类:伪元素{属性:属性值}

1. 定位锚伪类

伪类可以指定以不同的方式显示链接(links)、已访问链接(visited links)和可激活链接 (active links)。

一个有趣的效果是使当前链接以不同颜色、更大的字体显示,然后当网页的已访问链接 被重选时,又以不同颜色、更小字体显示,这个样式表的示例如下:

```
a:link {color: red;}
a:active{color: blue;font - size: 200 % }
a:visited{color: black;font - size: 50 % }
```

2. 首行伪元素

通常在报纸上文章的文本首行都会以粗体且全部大写展示,CSS也具有这个功能,可 将其作为一个伪元素。首行伪元素可用于任何块级元素,例如 p、hl 等,以下是一个首行伪 元素的例子:

p:first - line {font - variant: small - caps;font - weight: bold}

3. 首字母伪元素

首字母伪元素用于 drop caps(下沉行首大写字母)和其他效果。首字母伪元素可用于 任何块级元素,例如以下代码:

p:first - letter{font - size: 500 %; float: left}

以上代码表示首字母的显示效果比普通字体加大5倍。

3.6.2 样式表的层叠顺序

当使用了多个样式表时,样式表需要指定选择符的控制权。在这种情况下,总会有样式 表的规则能获得控制权,以下的特性将决定互相对立的样式表的结果。

1. ! important

可以用!important 把样式特指为重要的样式,一个重要的样式会大于其他相同权重的样式。以下是!important 声明的例子:

body {background: url(01.gif)white; background - repeat: repeat - x ! important}

2. 设计者样式和浏览者样式

网页设计者和浏览者都有能力去指定样式表,当两者的规则发生冲突时,在相同权重的 情况下,网页设计者的规则会高于浏览者的规则。但网页设计者和浏览者的样式表都高于 浏览器的内置样式表。

网页设计者应该谨慎使用!important 规则,例如,用户可能会要求以大字体显示或以 指定颜色显示,因为这些样式对于用户阅读网页是极为重要的。任何!important 规则都会 超越一般的规则,所以建议网页制作者使用一般的规则以确保有特殊样式需要的用户能阅 读网页。

3. 特性的顺序

为了方便使用,当两个规则具有同样的权重时,取后面的那个规则。

3.7 CSS 的各种属性

从 CSS 的基本语句就可以看出,属性是 CSS 非常重要的部分。熟练掌握 CSS 的各种 属性会使工作更加简单。下面就介绍 CSS 中的几种重要属性。

3.7.1 CSS 中的字体以及文本控制

1. 字体属性

字体属性是最基本的属性,网页制作中经常都会使用到,它主要包括以下这些属性。 1) font-family

font-family 是指使用的字体名称,其属性值可以选择本机上所有的字体,基本语法如下:

font-family:字体名称

代码示例如下:

<pstyle="font-family: '微软雅黑'">网页制作第三章学习

○ 这行代码定义了文本"网页制作第三章学习"将以微软雅黑的字体 显示,如图 3.24 所示。

如果在 font-family 后加上多种字体的名称,浏览器会按字体名称 的顺序逐一在用户的计算机中寻找已经安装的字体,一旦遇到与要求相

匹配的字体,就按这种字体显示网页内容并停止搜索。如果不匹配就继续搜索直到找到为止。 如果样式表中的所有字体都没有安装,浏览器就会用自己默认的字体来显示网页内容。

2) font-style

图 3.24

font-style 是指字体是否使用特殊样式,属性值为 italic(斜体)、bold(粗体)、oblique(倾 斜),其基本语法如下:

font-style:特殊样式属性值

代码示例如下:

网页制作第三章学习

这行代码定义了 font-style 属性为斜体(italic),显示效果如 图 3.25 所示。

网页制作第三章学习

图 3.25

3) text-transform

text-transform 用于控制文字的大小写。该属性可以使网页的设

计者不用在输入文字时就确定文字的大小写,而是在输入完毕后,根据需要对局部的文字设置大小写。其基本语法如下:

text - transform: 大小写属性值

控制文字大小写的属性值如下:

- uppercase 表示所有文字大写显示。
- lowercase 表示所有文字小写显示。
- capitalize 表示每个单词的首字母大写显示。
- none 表示不继承母体的文字变形参数。

4) font-size

font-size 定义字体的大小,其基本语法如下:



font-size:字号属性值

5) text-decoration

text-decoration 表示文字的修饰,文字修饰的主要用途是改变浏览器显示文字链接时 下画线的样式,其基本语法如下:

text-decoration:下画线属性值

下画线属性值的相关介绍如下:

- underline——为文字加下画线。
- reline——为文字加上画线。
- line-through——为文字加删除线。
- blink——使文字闪烁。
- none——不显示上述任何效果。

2. 文本属性

1) word-spacing

word-spacing 表示单词词间距。单词词间距指的是英文单词之间的距离,不包括中文 文字,其基本语法如下:

word-spacing: 间隔距离属性值

间隔距离的属性值为带单位(points、em、pixels、in、cm、mm、pc、ex)的值和 normal 等。

2) letter-spacing

letter-spacing 表示字母间距,字母间距是指英文字母之间的距离。该属性的功能、用 法以及参数设置和 word-spacing 很相似,其基本语法如下:

letter-spacing: 字母间距属性值

字母间距的属性值与单词间距相同,分别为带单位(points、em、pixels、in、cm、mm、pc、ex)的值和 normal 等。

3) line-height

line-height 表示行距,行距是指上下两行基准线之间的垂直距离。一般来说,英文五线 格练习本从上往下数的第三条横线就是计算机所认为的该行的基准线。line-height 基本语 法如下:

line-height: 行间距属性值

关于行距的取值,不带单位的值是以1为基数,相当于比例关系的100%;带长度单位的值是以具体的单位为准。

如果文字字号很大,而行距相对较小,那么可能会发生上下两行文字互相重叠的现象。

4) text-align

text-align 表示文本水平对齐的方式,该属性可以控制文本的水平对齐,而且并不仅仅 针对文字内容,也包括设置图片、影像资料的对齐方式,其基本语法如下:

text-align: 属性值

text-align 的属性值分别介绍如下。

- left——左对齐。
- right——右对齐。
- center——居中对齐。
- justify——两端对齐。

注意: text-align 是块级属性,只能用于、< blockquote>、、< h1>~< h6>等标识符中。

5) vertical-align

vertical-align 表示文本垂直对齐的方式。文本的垂直对齐是相对于文本母体的位置而 言的,而不是指文本在网页里垂直对齐。例如,表格的单元格里有一段文本,那么对这段文 本设置垂直居中就是针对单元格衡量的,也就是说,文本将在单元格的正中显示,而不是整 个网页的正中。vertical-align 的基本语法如下:

vertical-align: 属性值

vertical-align 的属性值分别介绍如下:

- top--顶端对齐。
- bottom——底端对齐。
- text-top——相对文本顶端对齐。
- text-bottom——相对文本底端对齐。
- baseline——基线对齐。
- middle——中心对齐。
- sub----以下标的形式显示。
- super——以上标的形式显示。
- 6) text-indent

text-indent 表示文本的缩进,主要用于中文版式的首行缩进,或是将大段的引用文本和 备注做成缩进的格式,其基本语法如下:

text-indent: 缩进距离属性值

缩进距离属性值可以是带长度单位的值或是比例关系。

注意:在使用比例关系的时候,有人会认为浏览器默认的比例是相对段落的宽度而言的,但其实并非如此,整个浏览器窗口才是浏览器所默认的参照物。另外,text-indent 是 块级属性,只能用于、<blockquote>、、<h1>~<h6>等标识符中。

3.7.2 CSS 中的颜色及背景控制

CSS 中的颜色及背景控制主要是对颜色属性、背景颜色、背景图像、背景图像的重复、 背景图像的固定和背景定位这 6 个部分的控制。



1. 对颜色属性的控制

颜色属性允许网页制作者指定一个元素的颜色,其基本语法如下:

color: 颜色参数值

颜色取值可以用 RGB 值表示,也可以使用十六进制数字色标值表示,或者以默认颜色的英文名称表示。以默认颜色的英文名称表示无疑是最为方便的,但由于预定义的颜色种 类太少,因此更多的网页设计者会用 RGB 方式或十六进制的数字色标值。RGB 方式可以 用数字的形式精确地表示颜色,也是很多图像制作软件(比如 Photoshop)默认使用的规范。

2. 对背景颜色的控制

在 HTML 中,要为某个对象加上背景色只有一种方式,即先做一个表格,在表格中设置完背景色,再把对象放进单元格中。这样做比较麻烦,不但代码较多,而且表格的大小和 定位也有些麻烦。而使用 CSS 则可以解决这些问题,且对象的范围广,可以是一段文字,也 可以只是一个单词或一个字母。背景颜色控制的基本语法如下:

background - color: 参数值

属性值同颜色属性取值相同,可以用 RGB 值表示,也可以使用十六进制数字色标值表示,或者以默认的英文名称表示,其默认值为 transparent(透明)。

3. 对背景图像的控制

对背景图像的控制的基本语法如下:

background - image:url(URL)

URL 就是背景图像的存放路径。如果用 none 来代替背景图像的存放路径,则不显示 图像。用该属性来设置一个元素的背景图像,其代码如下:

body{background - image:url(images/01.jpg)}
p{background - image:url(http://baidu.com/bg.png)}

4. 对背景图像重复的控制

背景图像重复控制的是背景图像是否平铺。当属性值为 no-repeat 时,不重复平铺背景 图像;当属性值为 repeat-x 时,图像只在水平方向上平铺;当属性值为 repeat-y 时,图像只 在垂直方向上平铺。也就是说,结合背景定位的控制,可以在网页上的某处单独显示一幅背 景图像。背景图像重复控制的基本语法如下:

background - repeat: 属性值

如果不指定背景图像重复的属性值,那么浏览器默认的是背景图像在水平、垂直两个方向上同时平铺。

5. 背景图像固定控制

背景图像固定控制用于指定背景图像是否随网页的滚动而滚动。如果不设置背景图像固定属性,那么浏览器默认背景图像随网页的滚动而滚动。背景图像固定控制的基本语法如下:

background - attachment: 属性值

当属性值为 fixed 时,网页滚动时背景图像相对于浏览器的窗口固定不动;当属性值为 scroll 时,背景图像随浏览器的窗口一起滚动。

视频讲的

6. 背景定位控制

背景定位用于控制背景图像在网页中的显示位置,基本语法如下:

background - position: 属性值

background-position 的属性值分别介绍如下:

- top——相对前景对象顶端对齐。
- bottom——相对前景对象底端对齐。
- left——相对前景对象左对齐。
- right——相对前景对象右对齐。
- center——相对前景对象中心对齐。

3.7.3 CSS 中边框的控制属性

CSS 样式表定义了一个容器(Box),它可以存储一个对象的所有可操作的样式,各个方面之间的关系如图 3.26 所示。

1. 边界

如图 3.26 所示,边界位于 Box 模型的最外层,包括 4 项属性,分别如下:

- margin-top——顶部空白距离。
- margin-right——右边空白距离。
- margin-bottom——底部空白距离。
- margin-left——左边空白距离。

空白距离可以用带长度单位的值表示。如果使用

上述属性的简化方式 margin,可以在其后连续加上 4

个带长度单位的值,设置元素相应边与框边缘之间的相对或绝对距离,有效单位为 mm、 cm、in、pixels、pt、pica、ex 和 em。

以父元素宽度的百分比设置边界尺寸或是 auto(自动),这个设置取浏览器的默认边界,分别表示 margin-top、margin-right、margin-bottom、margin-left,每个数字中间要用空格分隔,例如以下代码:

```
<!doctype html >
< html >
< head >
< meta charset = "utf - 8">
< title >无标题文档</title >
</head >
< body bgcolor = " # FFFFFF">
 margin 
margin 
</body >
</html >
```



图 3.26

将以上代码保存,在浏览器中打开,效果如图 3.27 所示。

margin			
margin			
2	图	3.27	

2. 边框

边框位于边框空白和对象间隙之间,包括了7项属性,分别如下:

- border-top----顶边框宽度。
- border-right——右边框宽度。
- border-bottom——底边框宽度。
- border-left——左边框宽度。
- border-width——所有边框宽度。
- border-color——边框颜色。
- border-style——边框样式参数。

其中,border-width可以一次性设置所有的边框宽度。用 border-color 同时设置 4 条边 框的颜色时,可以连续写上 4 种颜色并用空格分隔,连续设置的边框都是按 border-top、 border-right、border-bottom、border-left 的顺序。border-style 相对其他属性而言稍微复 杂,因为它还包括了多个边框样式的参数。

- none——无边框。
- dotted——边框为点线。
- dashed——边框为长短线。
- solid——边框为实线。
- double——边框为双线。
- groove——定义 3D 凹槽边框。
- ridge——定义 3D 垄状边框。
- inset——定义 3D 内嵌边框。
- outset——定义 3D 外凸边框。

3. 填充

填充即填充距,指的是文本边框与文本之间的距离,位于对象边框和对象之间,包括了 4项属性,分别如下:

- padding-top----顶部间隙。
- padding-right——右边间隙。
- padding-bottom——底部间隙。
- padding-left——左边间隙。

与 margin 类似,也可以用 padding 一次性设置所有的对象间隙,格式与 margin 相似。

3.7.4 CSS 中的分类属性

在 HTML 中,用户无须使用前面提到的一些字体、颜色、容器属性来对字体、颜色、边

距、填充距等进行初始化,因为在 CSS 中已经提供了进行分级的专用分类属性。

1. 显示控制样式

显示控制样式的基本语法如下:

display:属性值

各属性值如下:

- block(默认)——在对象前后都换行。
- inline——在对象前后都不换行。
- list-item——在对象前后都换行且增加了项目符号。
- none——无显示。

2. 空白控制样式

空白属性决定如何处理元素内的空格。空白控制样式的基本语法如下:

white-space: 属性值

各属性值如下:

- normal——把多个空格替换为一个空格。
- pre---按输入显示空格。
- nowrap——禁止换行。

注意: white-space 也是一个块级属性。

3. 列表项前的项目编号控制

控制列表项前面的项目编号的基本语法如下:

list-style-type: 属性值

各属性值如下:

- none——无强调符。
- disc----圆形强调符(实心圆)。
- circle----圆形强调符(空心圆)。
- square——方形强调符(实心)。
- decimal——十进制数强调符。
- lower-roman——小写罗马字强调符。
- upper-roman——大写罗马字强调符。
- lower-alpha——小写字母强调符。
- upper-alpha——大写字母强调符。
- 4. 在列表项前加入图像

在列表项前加入图像的基本语法如下:

list-style-image: 属性值

属性值为 url 时,指定的是所加入图像的 URL 地址;属性值为 none 时,表示不加入图像。例如以下代码:

```
ul. check list - style - image: url(/li - markers/checkmark.gif);
ul lix1 list - style - image: url(x. png);
```

5. 目录样式位置

目录样式位置的基本语法如下:

```
list-style-position: 属性值
```

目录样式用于设置强调符的缩排或伸排,这个属性可以让强调符突出于清单以外或与 清单项目对齐。

各属性值如下:

• inside——内部缩排,将强调符与清单项目内容左边界对齐。

• outside——外部伸排,强调符突出到清单项目内容左边界以外。

其中,outside 是默认值。整个属性决定关于目录项的标记应放在哪里。如果使用 inside 值,换行会移到标记下,而不是缩进。

6. 目录样式

目录样式属性是目录样式类型、目录样式位置和目录样式图像属性的缩写,它将所有目 录样式属性放在一条语句中,基本语法如下:

list-style: 属性值

其属性值为"目录样式类型""目录样式位置"或 url。 以下是一个关于分类属性的例子。

```
<! doctype html >
< html >
< head >
< meta charset = "utf - 8">
<title>css</title>
< style type = "text/css">
p {
display: block; white - space: normal
}
em {
display: inline;
}
body ul li {
 display: list - item;
list - style - type: square;
body p img {
display: block;
}
</style>
</head>
< body >
< em > sample </em > text < em > sample </em > text < em > sample </em >
```

```
text < em > sample </em > text < em > sample </em >

lis list - 1 
list - 2 
list - 3 

</mdmth{multiple{set}} multiple{set} multimultimultiple{set} multiple{set}
```

上述代码的显示效果如图 3.28 所示。

7. 控制鼠标指针属性

当把鼠标指针移动到不同的地方时,或当鼠标指 针需要执行不同的操作时,或当系统处于不同的状态 时,都会使指针的形状发生改变。也可以用 CSS 来 改变鼠标指针的属性,就是当鼠标指针移动到不同的 元素对象上面时,让指针以不同的形状、图案显示。 在 CSS 中,这种样式是通过 cursor 属性来实现的,其 基本语法如下:

sampletext samp	letextsample textsample textsample
list-1	
list-2	
list-3	
	TE .

图 3.28

cursor: 属性值

属性值为 auto、crosshair、default、hand、move、help、wait、text、w-resize、s-resize、 n-resize、e-resize、sw-resize、nw-resize、pointer 和 url,它们所代表的含义如下。

- style="cursor:hand"----手形。
- style="cursor:crosshair"——十字形。
- style="cursor:text"----文本形。
- style="cursor:wait"---沙漏形。
- style="cursor:move"——十字箭头形。
- style="cursor:help"——问号形。
- style="cursor:e-resize"——右箭头形。
- style="cursor:n-resize"-----上箭头形。
- style="cursor:nw-resize"——左上箭头形。
- style="cursor:w-resize"——左箭头形。
- style="cursor:s-resize"---下箭头形。
- style="cursor:se-resize"----右下箭头形。
- style="cursor:sw-resize"——左下箭头形。

示例代码如下:

```
<! doctype html >
```

```
< html >
```

< head >

```
< meta charset = "utf - 8">
```

```
<title>鼠标效果</title>
```

```
</body>
</head>
</body>
</htl>

< body>
</htl>

< h1 style = "font - family: '宋体'">鼠标效果</h1>

把鼠标移到相应的位置
观看效果。
</div style = "font - family: '微软雅黑';font - size: 24px;color: #252525">

< span style = "cursor: hand">手的形状</span >

< span style = "cursor: move">移动</span >

< span style = "cursor: ne - resize">反方向</span >

< span style = "cursor: wait">等待</span >

< span style = "cursor: help">求助</span >

< span style = "cursor: help">求助</span >

< span style = "cursor: help">求助</span >

</div>

</div>

</div>

</div>
```

将代码保存并用浏览器打开,效果如图 3.29 所示。

鼠杉	示效果
把鼠材	示移到相应的位置观看效果。
手的	形状
移动	
反方	向
等待	
求助	έφ.

图 3.29



3.8 本章范例

3.8.1 index 网页区域的制作

该部分由 3 个 div 和 1 个 table 组成, 3 个 div 分别为 jsdk、content 和 btn, 如图 3.30 所示。



图 3.30

(1) 新建一个文件夹,命名为 03start。将"lesson03/范例/03complete"文件夹中的 images 文件夹复制到该文件夹内。

(2) 打开 Dreamweaver CC 2018,新建站点,站点名称为 03start,站点文件夹为刚才新 建的 03start 文件夹,如图 3.31 所示。

送可以往此处为 Dr	eamweaver站只达排本地又什类和名称。	
站点名称:	03start	
本地站点文件夹:	C:\Users\Administrator\Desktop\03start\	

图 3.31

(3) 在 Dreamweaver CC 2018 中新建一个 HTML 文件, 然后执行"文件"→"保存"命 令, 将文件保存为 start. html。

(4) 切换到"拆分"视图,将 title 标签间的文字更改为"阅读",如图 3.32 所示。

(5)执行"窗口"→"属性"命令,出现"属性"窗口。选中 body标签,在属性窗口中单击"页面属性",弹出"页面属性"对 话框,设置背景图像如图 3.33 所示。

1	html
2 🔻	<html></html>
3 🔻	<head></head>
4	<meta charset="utf-8"/>
5	<title>阅读</title>
6	

图 3.32

分类	外观 (CSS)
外观 (CSS) 外观 (HTML)	页面字体 (F): 🖵 默认字体 🛛 🗸 🗸 🗸
链接 (CSS) 标题 (CSS) 标题 (GSS)	大小(S): v px v 文本颜色(T):
親踪图像	背景颜色(B):
	背景图像 (I): images/2.jpg 浏览 (W)
	重夏(E): no-repeat ~

图 3.33

(6) 切换到"设计"视图,将光标置于页面中,执行"插入"→Div 命令,打开"插入 Div" 对话框,在 ID 文本框中输入 jsdk,如图 3.34 所示。

				×
插入: Class:	在插入点	~	~	· 取消 · · · · ·
ID:	jsdk	~		

图 3.34

(7)设置完成后单击"确定"按钮,即可在页面中插入名称为jsdk的div。将多余的文本 内容删除,输入"精神的壳"文字,然后逐字添加链接,如图 3.35 所示。最终代码效果如 图 3.36 所示。

(8)为链接效果应用 class 属性和属性值,最后代码如下:



(9)制作字体放大镜效果,为其添加相应的样式。选择"CSS设计器"→"+"(添加 CSS 源)→"在页面中定义",然后添加"选择器",输入.bian,在"属性"面板中设置如图 3.37 所示的属性值。

			bian		
			+ - 属性		
					☑ 显示算
.bian			王 文本		
+ - 属性	1		color	: 📕 #FFFFFF	
		☑ 显示集	text-align text-decoration	: = = = = =	
■ 布局 width height margin	: <u>66</u> px : <u>44</u> px : 5px 75px 15px 55	DX.	■ 边框 border	: 1px solid #FF8002	
<u>55</u> px	<u>5</u> px <u>75</u> px		width style color	: <u>1</u> px : solid : 1 px : solid	
float	<u>15 px</u>	1	── 更多 font	· bald40mr 搭体	

图 3.37

(10)设置鼠标指针移动到字体上之后的效果。添加"选择器",输入.bian:hover,在"属性"面板中设置如图 3.38 所示的属性值。切换到"实时"视图,预览效果如图 3.39 所示。

(11)制作文本效果。在名为 jsdk 的 div 之后,插入一个名为 content 的 div,然后在其内部再插入名为 content-1 和 content-2 的 div,并添加相应文本,代码如下。

```
<div id = "content">
<div id = "content - 1">孤独者的<br>温柔书单</div>
<div id = "content - 2">用文字,温柔地推翻这个世界</div>
</div>
```

			.bian:hover	
.bian:hover			+ - 属性	
+ - 風	ŧ			☑ 显示集
		☑ 显示集	■ 文本	
■ 布局			color : 💽 #FFFFFF	
width height	: <u>100</u> px : <u>66</u> px		□ 边框	
margin	: 0px 9px		border : 1px solid #000000	
00000	<u>0</u> px			
<u>9</u> px	9 px		width : 1 px style : solid color :	
	<u>0</u> px			-
padding	: Opx 5px		◎ 背景	
100000	<u>0</u> px		background-image	
<u>5</u> px	2 5 px		gradient none	
- Sanna	<u>0</u> px			
position	: relat		font · bold 55mx 守体	

图 3.38



图 3.39

(12) 为文本添加效果样式。在"CSS 设计器"中添加"选择器",输入 # content,将其 float 属性设置为左浮动。

(13) 将光标置于"代码"视图中的名为 content-1 的 div 上,在"CSS 设计器"中添加"选择器",自动显示 # content # content-1,如图 3.40 所示,设置属性值如图 3.41 所示。

+ - 属性		
T		☑ 显示算
扁 布局		
margin	: 说置速记	
	100 px	
<u>0</u> or	300 px	
	<u>0</u> pr	
float	:	
王 文本		
color	: r gba(0,0,0,1.4	00)
font-family	: 华文楷体	
font-size	: 50 px	

+ - 选择器	
₽ 觸透 CSS 規則	
body table	
#content	
#content #content-1	

图 3.40



(14) 用与上一步相同的方法,添加"选择器" ♯ content ♯ content-2,设置属性值如图 3.42 所示。最后的文字效果如图 3.43 所示。

+ - 属性		
T		☑ 显示集
王 文本		
color	: 💽 rgba(0,0,0,1.00)	
font-family	: 华文楷体	
font-size	: 30 px	

图 3.42



图 3.43

(15)制作表格区域。将光标置于名为 content 的 div 之后,执行"插入"→Table 命令, 插入一个 4 行 4 列、宽度为 800 像素的表格,如图 3.44 所示。

長格大小				-
行数:	4	列:	4	
表格宽度:	800	像素	~	
边框粗细:	0	像素		
单元格边距:	0	8		
单元格间距:	0			
无 左	-	顶部	两者	_
乐题:				
有要:			-	
101		-	cute a	7

图 3.44

(16) 将光标置于标签上,在"属性"面板中设置单元格水平"居中对齐"、垂直 "居中","宽"为 800,"高"为 200,如图 3.45 所示。

<tab <tb< th=""><th>le wid ody></th><th>th=</th><th>"800 b</th><th>ord</th><th>ler="0"</th><th>cell</th><th>lspacing="</th><th>0" cel</th><th>lpa</th><th>dding="0"></th><th></th></tb<></tab 	le wid ody>	th=	"800 b	ord	ler="0"	cell	lspacing="	0" cel	lpa	dding="0">	
	<td< td=""><td< td=""><td< td=""><td< td=""><td< tr=""></td<></td<></td<></td<></td<>	wid wid wid	th="80 th="80 th="80 th="80)0")0")0"	height height height height	="200 ="200 ="200 ="200	0" align=" 0" align=" 0" align=" 0" align="	center center center center	" v " v " v	align="mido align="mido align="mido align="mido	dle"> dle"> dle"> dle">
table	tbody										
	Mate 格式 I	ሮ (F) D(I)	无 无		~	; 锁接 (类 (无 L)	⇒ B	1 :	- 0 6	标题 (T) 目标 (G)
₩元格 □ 芷	水平 (Z) 垂直(T)	居中	中对齐 P	÷	宽(W) 高(H)	800 200	不换行 (0) □ 标題 (E) □	背景颜色	<u> (</u> G)		

图 3.45

(17) 在表格中输入如图 3.46 所示的文字,然后分别选中每个表格的文字,为其添加链接并应用 class 属性,如图 3.47 所示。

上海译文	上海译文<,
理相国(/a)(/td)	理想国
译称出版(/a)(/+d)	译林出版<
果麦	果麦 <
在44~/~~/**	磨铁 <
古氏(/a)(/td)	读客 <
	小说 <
青春	青春 <
偏面给 <u>本</u> 、/_>>、/+d>	漫画绘本<
支西云本(/a)(/td/	文艺 <
又云、/ a/、/ tu/ 夕萎/ / a/、/ + d>	名著 <
内有(/a)(/td) 外文原版	外文原版<,
心理 / / _ / / + / + / >	心理 <
5座(/a/(/td/	历史 <
計判(/a)(/td)	社科 <
非虚构	非虚构
图 3.46	图 3.47

(18)设置表格内的字体样式。在 CSS 设计器中添加选择器,输入 body table,设置如 图 3.48 所示的属性值,效果如图 3.49 所示。



(19) 为每个小表格添加边框,代码如下:

tbody tr td { border: 1px dotted rgba(180,180,180,1.00); }

(20)为链接添加样式。在 CSS 设计器中添加选择器,分别输入. shu 和. shu: hover,属

.shu						
+ - 属性						
T		显示集	-	2	6	1 2
王 文本 color : [text-decoration : [. rgba(216,216,216,1.00) 国 工 〒 平		上海译文	理想的	译林出版	**
E	图 3.50		唐秋	iţ\$	小說	青春
.shu:hover			() :浸過換本	żΞ	名著	外文原版
+ - 属性			Reciple	ALC: NO	STREET,	
T		显示集		10		
T 文本 color : [, rgba(255,198,0,1.00)		wit	历史	i±41	非虚构
R	图 3.51			图	3.52	

性设置如图 3.50 和图 3.51 所示,预览效果如图 3.52 所示。

(21) 最后制作导航到 books. html 网页的导航按钮。在 table 标签之后,执行"插入"→ Div 命令,在"设计"视图中输入"了解更多",并为其加上链接,代码如下:

了解更多

然后为链接添加样式。在 CSS 设计器中添加选择器,分别输入.btn、btn:hover, .btn:focus、btn-large,属性设置如图 3.53~图 3.56 所示。



.btn 属性 + -= T 🗆 🖾 \cdots ☑ 显示集 **T** 文本 : 📮 #fff color font-family : 华文棔体 font-weight : 500px font-size : <u>40</u> px text-decoration : $\Box \underline{T} \overline{T} \mp$ text-transform : 🖸 Ab AB ab letter-spacing : 1 px □ 边框 border-radius : 0px 4r 8r 0 px 0 px 0 0 px 0 px ◎ 背景 background-color : rgba(0,0,0,0.50)

图 3.53

图 3.54

.btn:hover, .btn	:focus
+ - 属性	
T 🖾 …	☑ 显示;
王 义平	
color	: 📑 #fff
◎ 背景	
background- color	: 💽 rgba(0,0,0,0.80)

图 3.55

.btn-large		
+ - 属物		
		☑ 显示集
🖬 布局		
padding	: 15px 40px	
	<u>15</u> px	
<u>40</u> px	2 <u>40</u> px	
	<u>15 px</u>	

图 3.56

3.8.2 books 网页区域的制作

该部分由一个 table 和一个 div 组成,如图 3.57 所示。

(1)新建一个 HTML 文件,然后执行"文件"→"保存"命令,将文件保存为 books. html。切换到"拆分"视图,将 title 标签间的文字更改为"书类"。

(2)执行"窗口"→"属性"命令,出现"属性"窗口。
 选中 body 标签,在"属性"窗口中单击"页面属性",弹出"页面属性"对话框,设置背景图像如图 3.58 所示。

(3)新建制作表格区域。将光标置于页面中,执行 "插入"→Table 命令,插入一个 13 行 2 列、宽度为 800 像素的表格,如图 3.59 所示。

(4) 在"CSS 设计器"中添加"选择器", 输入 body table, 设置如图 3.60 所示的属性值。

(5) 在"属性"窗口中设置表格第一列单元格的属性 如图 3.61 所示,第二列单元格的属性如图 3.62 所示。



图 3.57

页面属性	
分类	外现 (CSS)
外观 (CSS) 外观 (HTML) 钱接 (CSS) 标题 (CSS) 标题 /解码 跟踪图像	页面字体 (F): □ 默认字体

图 3.58

(6) 在表格的第一列第一行单元格中设置背景图片为"image/3.jpg",并输入文字"上海译文"; 在表格的第二列第一行中输入文字"以译介和传播世界各民族优秀文化为己任,专注出版国际文坛举足轻重的名家名作,享有良好的声誉。"表格的第一行代码如下,效果如图 3.63 所示。

	body table	
	.btn	
	.btn:hover, .btn:focus	
	.btn-large	
	+ — 属性	
	= I 🗆 🖾 \cdots	S
	■ 布局	
	margin : 50px 0px	
	<u>50 px</u>	
	S PA	
炼大小 ····································	50 px	
行数: 13 列: 2		
表格宽度: 800 像素 ~	T 文本	
边框粗细: 0 像素	color : rgba(216,216,216,1.00)	
单 二 陈 边 部· 0 ■	font-family : 华文楷体	
	font-size : <u>30</u> px	
早元格问犯: 0 [] 题	text-align : ≡ ≣ ≡ ≡	
	□ 边框	
元 左 顶部 两者	border : 1px dotted rgba(228,228,228	,1.00)
助功能		
用题:	width : 1 px	
āğ.	style : dotted	
A.	color :rgba(228,228,228,1.00)	
Frith Think Think	◎ 背景	
	background-color : rgba(77,77,77,0.60)	

图 3.59

Ta



1

搜索	输出 盾	11日 日本 11日 11日								
<> HT	ML	格式	Ĉ(F)	无		~	1	类 (无	→ B I	:≣ 1≣ ±≣ ±≣
L CS	is	I	D(I)	无		~	链接(L)		~ ⊕ 🖻
	单元格	水平 (Z)	友	挤	~	宽(W)	300	不换行(0) 🗌	背景颜色 (G)	
	□ jí □	垂直(T)	顶	K	~	高(H)	200	标题(E) 🗌		

图 3.61

搜索	輸出 履	胜								
<> HT	ID	格式	(F)	无		~	1	类 (无	→ B I	:5 18 ± ±
th cs	s	I	D(I)	无		~	链接(L)		~ 🕀 🖿
	单元格	水平 (Z)	默认	(~	宽(W)		不换行(0) 🗌	背景颜色 (G	;)
	三派	垂直(T)	默り	L.	~	高(H)	202	标题(E) 🗌		

图 3.62



图 3.63

 上海译文

 上海译文

 以译介和传播世界各民族优秀文化为已任,专注出版国际文坛举足轻重的名家名作,享有良好的声誉.

(7) 接下来的 12 行与第 1 行插入方法相同,所以按照第(7)步的方法添加内容即可,代码参照范例文件。

(8) 最后制作导航到 index. html 网页的导航按钮。在 table 标签之后,执行"插入"→ Div 命令,在"设计"视图中输入"返回",并为其加上链接,代码如下:

```
<a class = "btn" href = "index.html">返回</a>
```

然后为链接添加样式。在"CSS设计器"中添加选择器,分别输入.btn、btn:hover, .btn:focus、btn-large,属性设置如图 3.64~图 3.66 所示。最终效果如图 3.1 所示。





一、模拟练习

打开 "lesson03/模拟/03complete/index.html"文件进行预览,根据本章所述知识做一 个类似的作品。作品资料已完整提供,获取方式见前言。

二、自主创意

自主设计制作一个网页,应用本章所学习知识,熟练使用 HTML 标签和 CSS 样式进行 网页设计。

1

三、理论题

1. HTML 的基本结构是什么?

2. 什么是 CSS 的 id 属性?

3. 什么是 CSS 的 class 属性?

4. 什么是 span 元素?