

## 第3章 CSS 技术

### 3.1 CSS 简介

单纯使用 HTML 开发的 Web 页面一经下载到客户端浏览器显示后其内容就不可变化,随着 Web 的发展,人们更多地希望能够看到动态的页面效果,因此产生了相应的 DHTML(Dynamic HTML,动态 HTML)开发技术,主要包括 CSS(Cascading Style Sheets,层叠样式表)、脚本语言、HTML 4.0 及其以上版本和支持动态效果的浏览器。

CSS 中文译为层叠样式表,它是对 Web 页面显示效果进行控制的一套标准。样式就是页面显示的文字、图片等元素的格式、风格。层叠样式也就是指当页面同一元素在显示时受到多种样式控制时,将按照一定的层次顺序处理。

我们可以使用任何的文本编辑软件来编写 CSS 文件,编写好的文件以扩展名“.css”保存,这样只需要在页面文件中调用相应的 CSS 文件即可,当然也可以将 CSS 代码直接嵌入 HTML 文件中实现对页面显示样式的控制。

CSS 扩充了 HTML 标记的属性设定,使得页面显示效果更加丰富,表现效果更加灵活,更具有动态性。同时使用 CSS 可以将页面样式定义和 HTML 文件分离,使得页面开发及维护工作更容易进行。对于使用同一种显示效果的页面,我们可以定义一个 CSS 文件,而让所有显示样式相同的页面文件都调用这个 CSS 文件,我们也可以在在一个页面文件中调用多个 CSS 样式表。当我们要调整或修改样式时,只需要修改 CSS 文件即可,不需要对页面文件进行任何修改。

CSS 的主要特点可以表现在以下几个方面。

(1) 将网页的内容结构和格式控制分开。在 CSS 出现以前,网页文件中格式控制代码和页面显示内容是交错在一起的,这样对于代码的查看和修改造成了极大的不便。现在使用 CSS 可以将两者分开,这样很大程度上方便了网页设计开发人员,网页文件只负责存放页面要显示的内容,而所有的格式控制都由 CSS 文件来处理。这样不仅缩减了网页文件的体积,加快了下载显示速度,同时提高了网站的维护工作效率和网页开发效率,大大减少了工作量。

(2) 可以精确控制页面的所有元素。CSS 弥补了 HTML 的不足,提供了大量的样式属性,同时可以通过脚本代码来实现对页面元素的控制,使得对页面元素的控制更加精确。

(3) 页面显示效果更加丰富。CSS 中提供了大量的网页视觉效果,如丰富了颜色的显示,增加了大量的图像处理特效等,使得网页的显示效果更加丰富。

(4) 支持多种浏览器。现在主流的浏览器,如 IE 4.0 以上版本和 NE 4.0 以上版本都支持 CSS。

## 3.2 CSS 基本语法

### 3.2.1 CSS 的基本格式

CSS 定义的基本语法格式如下：

选择符 {规则列表}

其中选择符是指要使用该样式的对象(将在 3.2.3 小节详细介绍),它可以是一个或多个 HTML 标记、CLASS 选择符或 ID 选择符,如果为多个则使用逗号“,”进行分隔。规则列表是由一个或多个属性定义语句组成的样式规则,各语句间使用分号“;”进行分隔。属性定义语句的语法格式如下：

属性名：属性值

如：

```
h3{font-family:隶书;color:blue}
h2,h3{ font-family:宋体;color:red }
myfont{font-size:20pt}
#myfont{font-size:20pt}
```

在页面文件中对 CSS 的定义有以下几种方式。

(1) 内联样式。内联样式就是直接在页面文件中使用 HTML 标记的 style 属性。这种方式可以直接在 HTML 标记中定义该标记的显示样式,并且该样式定义只能用于这个标记。其语法格式如下：

<标记名 style="样式属性名 1: 属性值 1;样式属性名 2: 属性值 2;...">

如：

```
<p style=" font-family:宋体;color:red; font-size:10pt " >
```

下面可以通过程序清单 3-1 来详细说明,运行结果如图 3-1 所示。

**程序清单 3-1：**

```
<!--style_1.html 文件代码-->
<html>
<head>
  <title>样式表定义——直接定义标记的 style 属性</title>
</head>
<body>
  <p style=" font-family:宋体;color:red; font-size:10pt " >
    定义此处 &lt;p>标记的 style 属性: 红色宋体,字体大小为 10pt
  </p>
  <p>
    此处的 &lt;p>标记不受上述 style 属性控制。
  </p>
```

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

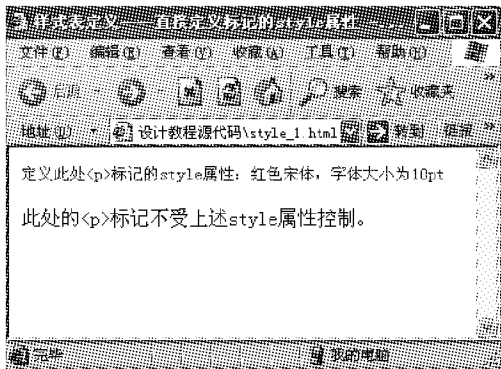


图 3-1 程序 style\_1.html 运行结果

(2) 在页面文件中定义内部样式表。这种方式通过<style>标记来定义样式,其语法格式如下:

```
<style type="text/css" >
<!--
    选择符 1, 选择符 2, ... {样式属性名 1: 属性值 1;样式属性名 2: 属性值 2;...}
    :
-->
</style>
```

其中使用了 HTML 注释标记<!--.....>,是为了当有浏览器不支持 CSS 语句时,遇到该语句段就会忽略该段内容。下面我们可以通过程序清单 3-2 来详细说明,运行结果如图 3-2 所示。

程序清单 3-2:

```
<!--style_2.html 文件代码-->
<html>
<head>
    <title>样式表定义——内部样式表</title>
</head>
<style type="text/css">
<!--
    /* HTML 标记 */
    h3{font-family:隶书;color:blue}

    /* CLASS 选择符 */
    .redfont{font-family:华文彩云;color:red}
    h3.blackfont{font-family:华文行楷;color:black}

    /* ID 选择符 */
    # myfont{font-family:黑体;color:blue;font-size:16px}
```

```

h3# hfont{font-family:华文行楷;color:red}
-->
</style>
<body>
<h3>此行文字样式为蓝色隶书</h3>
<p class=redfont>此行文字是红色华文彩云样式</p>
<p class=blackhfont>blackfont 样式仅控制 h3 标记,因此此行文字样式为默认字体</p>
<h3 class=blackfont>此行文字样式为黑色华文行楷</h3>
<p id=myfont>此行文字样式为蓝色黑体,字体大小为 16px</p>
<p id=hfont>hfont 样式仅控制 h3 标记,因此此行文字样式为默认字体</p>
<h3 id=hfont>此行文字样式为红色华文行楷</h3>
</body>
</html>

```

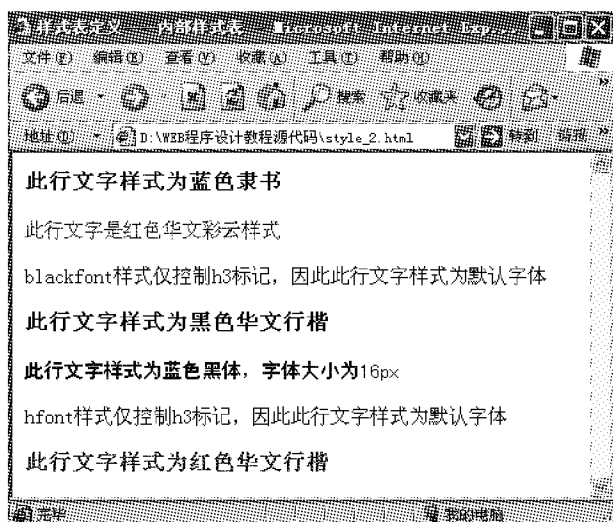


图 3-2 程序 style\_2. html 运行结果

(3) 在页面文件中嵌入外部样式表。上面两种方式都是将 CSS 语句直接嵌套在页面文件中, 这样这种 CSS 的定义语句只能使用于当前的页面文件。为了让多个页面文件可以共享 CSS 样式定义, 可以将 CSS 语句段编写为单独的一个 CSS 文件, 然后将它嵌入到页面文件中。其语法格式如下:

```

<style type="text/css">
<!--
    @import url("外部 CSS 样式表文件名");
-->
</style>

```

其中“外部 CSS 样式表文件名”是以“.css”为扩展名的 CSS 文件, 如果该文件和当前页面文件处于同一目录, 则直接给出文件名, 否则给出其相对路径。

下面可以通过程序清单 3-3 来详细说明具体使用方法, 运行结果如图 3-3 所示。

程序清单 3-3:

```
<!--style_3.html 文件代码-->
<html>
<head>
  <title>样式表定义——在页面文件中嵌入外部样式表</title>
</head>
<style type="text/css">
<!--
  @import url("mystyle.css");
-->
</style>
<body>
<h3>此行文字样式为蓝色隶书</h3>
<p class=redfont>此行文字是红色华文彩云样式</p>
<h3 class=blackfont>此行文字样式为黑色华文行楷</h3>
<p id=myfont>此行文字样式为蓝色黑体,字体大小为 16px</p>
<h3 id=hfont>此行文字样式为红色华文行楷</h3>
</body>
</html>
```

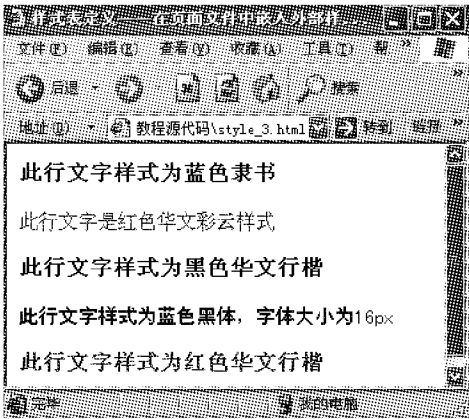


图 3-3 程序 style\_3. html 运行结果

其中嵌入的外部样式表文件为 mystyle. css,该文件和 style\_3. html 文件处于同一目录下。mystyle. css 文件的代码如下:

```
/* mystyle.css 文件代码 */
h3{font-family:隶书;color:blue}
.redfont{font-family:华文彩云;color:red}
h3.blackfont{font-family:华文行楷;color:black}
#myfont{font-family:黑体;color:blue;font-size:16px}
h3#hfont{font-family:华文行楷;color:red}
```

(4) 链接外部样式表。这种方式和嵌入外部样式表的方式相似,也要访问外部样式表,但是嵌入外部样式表时是将样式文件直接加载到 import 语句处,而链接外部样式表是直接

向样式文件索取样式。其语法格式如下：

```
<link type="text/css" rel=stylesheet href="外部 CSS 样式表文件名">
```

下面可以通过程序清单 3-4 来说明具体使用方法，所链接的外部样式表文件就是上例中的 mystyle.css 文件。

程序清单 3-4：

```
<!--style_4.html 文件代码-->
<html>
<head>
  <title>样式表定义——在页面文件中链接外部样式表</title>
</head>
<link type="text/css" rel=stylesheet href="mystyle.css">
<body>
<h3>此行文字样式为蓝色隶书</h3>
<p class=redfont>此行文字是红色华文彩云样式</p>
<h3 class=blackfont>此行文字样式为黑色华文行楷</h3>
<p id=myfont>此行文字样式为蓝色黑体,字体大小为 16px</p>
<h3 id=hfont>此行文字样式为红色华文行楷</h3>
</body>
</html>
```

3.2.2 CSS 注释语句

在 CSS 中也可以添加注释语句来对代码进行说明。CSS 的注释语句是位于“/ \* ”和“ \* /”标记之间的语句内容。可以在 3.2.1 小节中的程序清单 3-2 中看到注释语句的使用。

3.2.3 CSS 选择符

CSS 选择符主要有 HTML 标记、CLASS 选择符和 ID 选择符这 3 种。它们的定义和使用方法见表 3-1。

表 3-1 CSS 选字符的定义和使用

| 选择符       | 语 法 格 式                                            | 样式使用范围说明                            | 示 例                                                 |
|-----------|----------------------------------------------------|-------------------------------------|-----------------------------------------------------|
| HTML 标记   | 定义语法：标记{...}<br>使用语法：<标记>                          | 在 HTML 文件中,所有该标记包含的文本都具有定义的 CSS 样式  | h3{font-family:隶书}<br>:<br><h3>...</h3>             |
| CLASS 选择符 | 定义语法：*, 类名{...}或<br>.类名{...}<br>使用语法：<标记 class=类名> | 在 HTML 文件中的所有使用该类名的标记都具有定义的 CSS 样式  | .my{font-size:20pt}<br>:<br><h3 class=my>...</h3>   |
|           | 定义语法：标记.类名{...}<br>使用语法：<标记 class=类名>              | 在 HTML 文件中的所有指定该类名的该标记都具有定义的 CSS 样式 | h3.my{font-size:20pt}<br>:<br><h3 class=my>...</h3> |

| 选择符       | 语 法 格 式                                    | 样式使用范围说明                               | 示 例                                              |
|-----------|--------------------------------------------|----------------------------------------|--------------------------------------------------|
| ID<br>选择符 | 定义语法: # ID 名{...}<br>使用语法: <标记 id=ID 名>    | 在 HTML 文件中的所有使用该 ID 名的标记都具有定义的 CSS 样式  | #my{font-size:20pt}<br>:<br><h3 id=my>...</h3>   |
|           | 定义语法: 标记 # ID 名{...}<br>使用语法: <标记 id=ID 名> | 在 HTML 文件中的所有指定该 ID 名的该标记都具有定义的 CSS 样式 | h3#my{font-size:20pt}<br>:<br><h3 id=my>...</h3> |

可以在 3.2.1 小节中的程序清单 3-2 中看到具体的选择符的定义和使用方法。

### 3.2.4 样式表的层叠顺序

可以使用 CSS 语句对 HTML 标记设置不同的显示样式,但是由于 HTML 标记在使用中常常有嵌套情况出现,那么对于控制同一页面内容的嵌套标记,究竟哪一种样式起作用,是我们在本节中要说明的内容。下面我们先看一个简单的例子。

程序清单 3-5:

```
<!--cascade.html 文件代码-->
<html>
<head>
  <title>样式表的层叠</title>
</head>
<style type="text/css">
<!--
  h3{font-family:隶书;color:blue}
  h2{color:#FF0099;font-size:10px}
  p.font1{font-family:"华文行楷";color:red}
  center.font2{font-family:"宋体";color:green;font-style:italic}
  h2{font-family:黑体;font-size:20px}
-->
</style>
<body>
<h3 style=" font-family:宋体; font-size:10pt ">
  此行文字样式为蓝色宋体,字体大小 10pt
</h3>
<br>
<center class=font2>
<p class=font1>此行文字样式为红色华文行楷,斜体</p>
</center>
<br>
<h2>此行文字样式为黑体,颜色为# ff0099,字体大小 20px</h2>
</body>
</html>
```

该程序运行结果见图 3-4 所示。

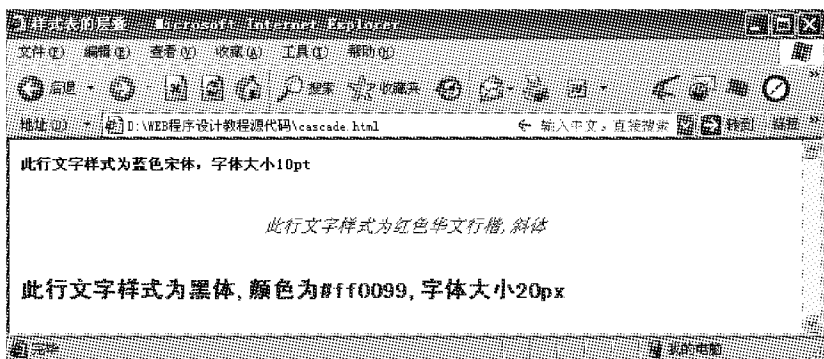


图 3-4 程序 cascade.html 运行结果

从程序的运行结果我们可以看出下面几点。

(1) 在该程序中直接对<h3>标记定义了内部样式表,同时又在页面文件中使用<h3>标记的 style 属性定义了它的样式,此时后者的优先级高于前者,所以显示的文本字体样式为“宋体,字体大小 10pt”,但由于<h3>标记的 style 属性没有定义字体的颜色,所以这行文字的颜色还是由内部样式表定义的<h3>标记的颜色样式来控制,显示为蓝色。

(2) 在内部样式表中定义了 CLASS 选择符 font1 和 font2,分别控制<p>标记和<center>标记的样式。从浏览器中的第二段文字显示效果可以看出,当<p>标记嵌套在<center>标记内标注同一段文字时,最靠近这段文字的<p>标记样式 font1 优先级要高于它外层的<center>标记样式 font2。因此当它们定义了同样的属性时,起作用的为<p>标记的 font1 样式,而在 font1 中没有定义的样式属性 font-style 由于在 font2 中定义了,所以它也被继承下来同样对该段文字起作用。

(3) 在该程序中先后两次定义了<h2>标记的样式表,从浏览器中的第三段文字显示效果可以看出,当两次都定义了相同属性时第二次定义的样式优先级更高。

因此对于不同的样式定义我们可以总结如下。

- (1) 直接在页面文件中使用 HTML 标记的 style 属性定义的样式优先级最高。
- (2) 其他的样式定义按照在页面文件中出现的顺序,越后出现的优先级越高。
- (3) CLASS 选择符的优先级高于 ID 选择符。
- (4) 没有被定义样式控制的内容将使用浏览器的默认样式。

## 3.3 CSS 基本属性

下面将介绍常用的 CSS 的基本属性的使用方法,主要包括背景属性、文本属性、字体属性、边界属性、边框属性、边距属性、列表属性和定位属性等。

### 3.3.1 CSS 背景属性

CSS 中有关背景的属性、对背景颜色的设置属性和对背景图片的设置属性。

(1) background-color。用于设置背景颜色,其属性设置格式如下:

background-color: 颜色|transparent



其中 transparent 表示透明,是该属性的默认值。颜色可以有多种表示方式。

① 使用英文名称,如 red、blue 等。

② 用 6 位十六进制数(#rrggb)表示。

③ 用 3 位十六进制数(#rgb)表示。

④ 用 rgb(r, g, b)函数表示,其中 3 个参数的取值可为 0~255 的整数或百分比值,如:rgb(0,0,0)表示黑色,rgb(100%,100%,100%)表示白色。

(2) background-image。用于设置要加载的背景图片,其属性设置格式如下:

```
background-image: url(背景图片的地址)|none
```

(3) background-repeat。用于设置背景图片的排列方式,其属性设置格式如下:

```
background-repeat: repeat|repeat-x|repeat-y|no-repeat
```

其中 repeat 表示以并排方式排列图片,是该属性的默认值;repeat-x 表示以 x 轴方向并排排列图片;repeat-y 表示以 y 轴方向并排排列图片;no-repeat 表示不以并排方式排列图片。

(4) background-attachment。用于设置移动滚动条时背景图片位置是否固定,其属性设置格式如下:

```
background-attachment: scroll|fixed
```

其中 scroll 表示移动滚动条时背景图片随之移动,是该属性的默认值;fixed 表示移动滚动条时背景图片不动。

(5) background-position。用于设置背景图片的插入位置,其属性设置格式如下:

```
background-position: x 位置值 y 位置值
```

其属性值可以有 3 种取法。

① 百分比,用于描述图片距区域元素边框的百分比值。

② 数值,用具体数值描述,单位可为任意长度单位,如 em、px、in、pt、pc、cm、mm 等。

③ 关键字,包括 left、right、center、top 和 bottom。

### 程序清单 3-6:

```
<!--background.html-->
<html>
<head>
  <title>背景属性设置示例</title>
</head>
<style type="text/css">
<!--
    .bg1{background-image:url(./image/img1.jpg);background-repeat:no-repeat;
        background-attachment:fixed}
    .bg2{background-color:# 99CC33}
    .bg3{background-image:url(./image/img2.gif);background-repeat:repeat-x}
    .bg4{background-image:url(./image/img2.gif);background-repeat:no-repeat;
        background-position:50% 20% }
```

```

-->
</style>
<body class=bgl>
<table border=1 width=50% align=right>
  <tr height=100 class=bq2>
    <td>背景颜色</td>
  </tr>
  <tr height=200 class=bq3>
    <td>以水平方向平铺图片</td>
  </tr>
  <tr height=200 class=bq4>
    <td>固定位置设置背景图片</td>
  </tr>
</table>
</body>
</html>

```

程序清单 3-6 运行结果如图 3-5 所示。

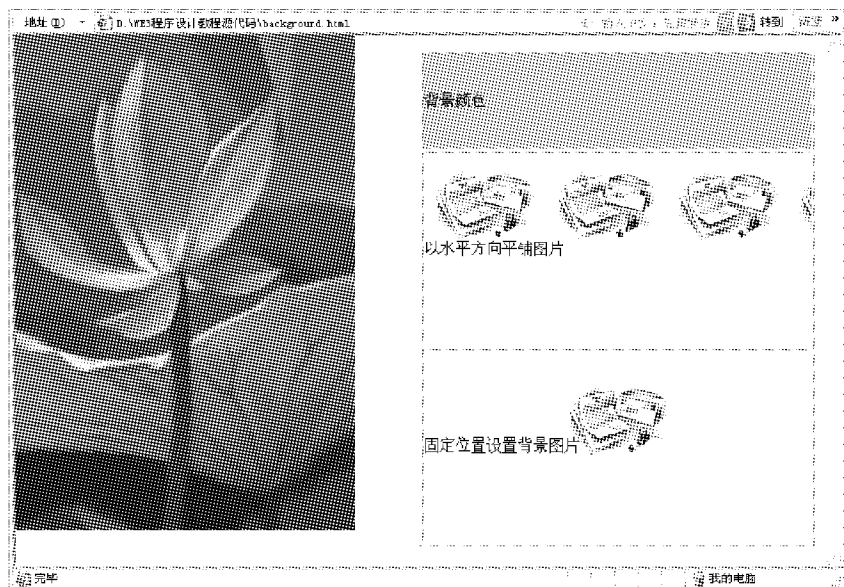


图 3-5 程序 background.html 运行结果

### 3.3.2 CSS 文本属性

CSS 中有关文本的属性主要包括以下几个。

(1) text-indent。用于设置文本的首行缩进值，默认属性值为 0(即不缩进)。其属性设置语法如下：

text-indent: 长度

其属性值用具体数值描述，单位可为任意长度单位。