

第5章

CSS层叠样式表

内容导读

XML 文本侧重于数据内容的描述,它没有提供对数据显示的信息,但是在实际应用中,为了方便用户读取和使用 XML 文本,通常需要使用样式表技术。样式表是由一系列指令规则构成,它是专门描述结构文档表现方式的一种文本内容。W3C 针对 XML 文档推出了两种样式表语言:一种是 CSS 层叠样式表,另一种是 XSL 可扩展样式表。

本章主要围绕 CSS 层叠样式表展开详细描述,并结合相关实例介绍如何使用 CSS 对 XML 文本进行样式的设计。

本章要点

- ◇ 理解 CSS 的概念。
- ◇ 掌握 CSS 的基本语法。
- ◇ 掌握 CSS 中常用属性及用法。
- ◇ 掌握 CSS 在 XML 文本中的引用方式。

5.1 CSS 概述

XML 文本为存储结构化数据提供了存储、交流和共享的平台,它将显示内容与格式相分离,因而 XML 文本本身没有提供数据显示的信息。单纯使用 XML 文本,对于使用者而言,在阅读文档时会显得枯燥乏味,并且无法有效地获取重要数据。采用样式表技术,可以使 XML 呈现不同的样式,不但界面美观,在数据获取方面也起到重要作用。

CSS(Cascading Style Sheet,层叠样式表)也称为级联式样式表,它是 W3C 于 1996 年制定并发布的一种用于控制多重网页样式和布局的技术,它是实现了真正意义上的网页表现与内容分离的一种样式设计语言。采用 CSS 技术能够对网页中的布局、背景、颜色、字体以及其他样式效果实现更加精确的控制。使用 CSS 样式表,只需简单地改变样式,对应的元素就会自动更新,因而呈现不同的显示效果。

CSS 技术的设计目标是为 HTML 文本的显示设置的,但是 CSS 在 HTML 和 XML 中是相通的,它通过定义一组特定的属性来设置标记文本的格式,用来在页面上获得更丰富的显示效果。CSS 主要特点包括以下几个。

- ① 方便控制页面布局,并且可以同时更新多个页面的样式。
- ② 数据显示和内容相分离其数据表现力强,可读性好。

③ 减小页面文件容量,加快页面访问速度。

随着 CSS 技术的不断发展,目前 CSS 有 3 个不同级别的标准,分别为 CSS1、CSS2 和 CSS3。W3C 于 1996 年 12 月 17 日发布 CSS1 推荐标准,1999 年 1 月 11 日,此推荐标准被重新修订,作为 CSS2 发布,并增加了对媒介和可下载字体的支持,CSS2 是 CSS 中比较稳定的版本,也是目前使用最为广泛的版本。而 CSS3 是目前 CSS 技术的最高版本,W3C 仍在对其开发和完善中。

5.2 CSS 语法

1. CSS 基本语法

CSS 是由选择器、属性名和属性值等一系列元素组成,其语法格式为:

```
选择器{  
    属性名 1:属性值 1;  
    属性名 2:属性值 2;  
    ... ..  
    属性名 n:属性值 n;  
}
```

说明:选择器是 HTML 或 XML 文本对应的元素名称,它旨在告诉浏览器页面上哪个元素要受到特定规则的约束。属性名用于描述样式规则的名称,属性值描述样式规则对应的内容。其中,属性名和属性值成对出现,中间用冒号“:”隔开,它们是由 W3C 所规定的一系列特定的名称和内容。通过属性可以告诉浏览器如何显示由该规则约束的元素,不同的属性之间需要用分号“;”隔开。

2. CSS 注释

注释可增加程序的可读性,在 CSS 中也包含注释语句。包含在注释语句中的信息,浏览器在读取样式表时,将会忽略其内容。其语法格式为:

```
/* 注释内容 */
```

3. CSS 中选择器中元素的使用

CSS 技术最初的设计目标是针对 HTML 的,因而对应选择器的元素只能为英文字符,且不分字母大小写。

将 XML 应用于 CSS 中,会带来一定的问题。在 XML 中元素根据语法要求是可以自行定义的,且定义为英文字符需要区分大小写。如果使用 CSS 样式表描述 XML 文本,则 XML 文本对应的元素不能为非英文字符,且不建议使用通过区分大小写字母的方式定义元素。

使用 CSS 格式化 XML 不是常用的方法,更不能代表 XML 文本样式化的未来,W3C 推荐使用 XSL,XSL 技术将在第 6 章重点讲解。

4. CSS 中文的使用

CSS 支持多种字符集,字符集中通常包括如宋体、微软雅黑等中文字符,因此需要在 CSS 文件中设置中文字符编码,设置指令为:

```
@charset "gb2312";
```

需要注意的是,使用该指令只能描述中文字符集,XML 元素仍然不能使用中文。根据 CSS 基本语法格式的介绍,这里简单举例说明 CSS 样式表的用法。举例如下。

文件 5-2-1.xml

```
<?xml version = "1.0" encoding = "UTF - 8"?>
<?xml - stylesheet type = "text/css" href = "5 - 2 - 1. css"?>
< users >
  < user >
    < name >张红</name >
    < card > SY102030 </card >
    < passwd > 123456 </passwd >
  </user >
  < user >
    < name >赵楠</name >
    < card > SY102031 </card >
    < passwd > 654321 </passwd >
  </user >
</users >
```

文件 5-2-1.css

```
@charset "gb2312";
/* CSS 使用方式 */
user { display:block; }
name { font - family:微软雅黑;
       font - size:25pt; }
card { font - family:"Times New Roman";
       font - size:20pt; }
passwd { font - family:"Arial Black";
        font - size:15pt; }
```

文件 5-2-1.xml 引用文件 5-2-1.css,在浏览器显示的效果如图 5-1 所示。

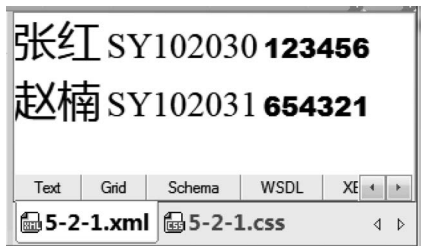


图 5-1 使用 CSS 的显示效果

5.3 CSS 选择器

选择器是 HTML 或 XML 文本对应的元素名称,它旨在告诉页面上哪个元素要按照属性的要求显示相应的信息。选择器除了使用相应元素外,根据 CSS 选择器的用途,还可以将选择器分为多元素选择器、类选择器、ID 选择器、通用选择器以及后代选择器等。

5.3.1 多元素选择器

如果需要将 XML 中的多个不同元素设置为相同样式,可以将 CSS 的选择器匹配多个元素,多个元素之间用逗号分隔。

在文件 5-2-1.css 中,如果将元素 name、card 和 passwd 设置为相同的样式,可以改为:

```
@charset "gb2312";
user { display:block; }
name,card,passwd { font-family:微软雅黑;
                  font-size:20pt; }
```

5.3.2 类选择器

在 XML 的实际应用中,同一个标记会被反复使用,如果需要为相同的标记赋予不同的 CSS,就可以使用类选择器。为了将类选择器的样式与元素关联,需要在使用类选择器前为元素设定一个 class 属性并为其指定一个相应的值,结合点符号(.)即可设置类选择器。需要注意的是,类选择器允许定义相同的 class 属性值。

文件 5-3-2-1.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/css" href="5-3-2-1.css"?>
<users>
  <user class="u01">
    <name>张红</name>
    <card>SY102030 </card>
    <passwd>123456 </passwd>
  </user>
  <user class="u02">
    <name>赵楠</name>
    <card>SY102031 </card>
    <passwd>654321 </passwd>
  </user>
</users>
```

文件 5-3-2-1.css

```
@charset "gb2312";
user.u01 { display:block;
           font-family:微软雅黑;
           font-size:20pt; }
user.u02 { display:block;
```

```
font-family:华文彩云;
font-size:20pt; }
```

文件 5-3-2-1.xml 引用文件 5-3-2-1.css,在浏览器显示的效果如图 5-2 所示。

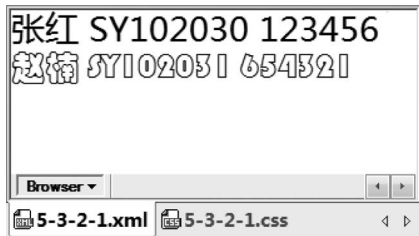


图 5-2 使用 CSS 的显示效果

5.3.3 ID 选择器

ID 选择器和类选择器相似,都可以为同一个标记赋予不同的 CSS 样式。为了将 ID 选择器的样式与元素关联,需要在使用 ID 选择器前为元素设定一个 id 属性,并为其指定一个相应的值,结合“#”符号即可设置 ID 选择器。但是不同于类选择器的是,ID 选择器只能把 CSS 指定给一个标记,即 ID 选择器只允许定义一个唯一的 id 属性值。

文件 5-3-3-1.xml

```
<?xml version = "1.0" encoding = "UTF-8"?>
<?xml-stylesheet type = "text/css" href = "5-3-3-1.css"?>
<users>
  <user id = "u01">
    <name>张红</name>
    <card>SY102030</card>
    <passwd>123456</passwd>
  </user>
  <user id = "u02">
    <name>赵楠</name>
    <card>SY102031</card>
    <passwd>654321</passwd>
  </user>
</users>
```

文件 5-3-3-1.css

```
@charset "gb2312";
user # u01 { display:block;
              font-family:微软雅黑;
              font-size:20pt; }
user # u02 { display:block;
              font-family:华文彩云;
              font-size:20pt; }
```

文件 5-3-3-1.xml 引用文件 5-3-3-1.css,在浏览器显示的效果如图 5-3 所示。

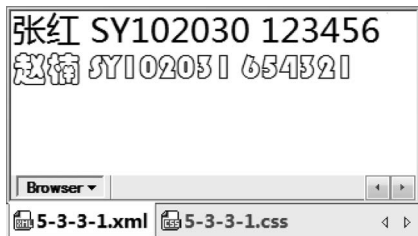


图 5-3 使用 CSS 的显示效果

5.3.4 通用选择器

在 CSS 中可以使用“*”标识选择器,它表示匹配任意元素。它可以为 XML 中的每个元素设置相同的样式。

文件 5-3-4-1. css

```
@charset "gb2312";
* { display:block;
    font-family:微软雅黑;
    font-size:20pt; }
```

文件 5-2-1. xml 引用文件 5-3-4-1. css,在浏览器显示的效果如图 5-4 所示。



图 5-4 使用 CSS 的显示效果

5.3.5 后代选择器

后代选择器是一种基于传承关系应用样式的选择器,它可以选择某个元素的后代元素设置样式。后代选择器要求该元素与后代元素间使用空格隔开即可。例如,对 5-2-1. xml 文档中< users>元素的后代元素< name>设置样式,其设置格式如文本 5-3-5-1. css 所示。

文件 5-3-5-1. css

```
@charset "gb2312";
users name { display:block;
```

```
font-family:微软雅黑;  
font-size:20pt; }
```

文件 5-2-1.xml 引用文件 5-3-5-1.css,在浏览器显示的效果如图 5-5 所示。



图 5-5 使用 CSS 的显示效果

5.4 CSS 属性设置

CSS 通过声明设置页面的样式,声明主要包括属性名和属性值的组合。本书根据 CSS 的用途重点讲解常用的属性。

5.4.1 颜色属性值

在页面设计中,颜色的应用为页面增添了多彩的效果。如何在页面中设置合理的布局,并运用精美的色彩搭配网页是设计者的任务之一。页面中颜色的运用是必不可少的一个元素,常用颜色值的地方包括字体颜色、背景颜色、边框颜色等。

一般情况下,颜色属性值可以通过颜色英文名称、RGB 颜色等方式设置。其中,RGB 色彩模式是通过对红、绿、蓝 3 个颜色通道的变化以及它们相互之间的叠加来得到各种颜色的,它是目前运用最广泛的颜色系统之一。RGB 有以下 3 种常用表现方式。

- (1) 红、绿、蓝 3 种颜色分量分别使用十六进制数表示,如 #00FF00。
- (2) 使用十进制格式为 rgb(x,y,z) 的方式表示,其中 x、y、z 的值位于 0~255,如 rgb(147,255,45)。
- (3) 使用百分数格式为 rgb(x%,y%,z%) 的方式表示颜色,其中数值为百分数,如 rgb(20%,40%,80%)。

常用颜色属性值如表 5-1 所示。

表 5-1 常用颜色属性值

颜色名称	十六进制 RGB	十进制 RGB	百分数 RGB	备 注
black	# 000000	rgb(0,0,0)	rgb(0%,0%,0%)	黑色
blue	# 0000FF	rgb(0,0,255)	rgb(0%,0%,100%)	蓝色
lime	# 00FF00	rgb(0,255,0)	rgb(0%,100%,0%)	灰色
red	# FF0000	rgb(255,0,0)	rgb(100%,0%,0%)	红色
aqua	# 00FFFF	rgb(0,255,255)	rgb(0%,100%,100%)	浅绿色

续表

颜色名称	十六进制 RGB	十进制 RGB	百分数 RGB	备 注
magenta	#FF00FF	rgb(255, 0, 255)	rgb(100%, 0%, 100%)	品红色
white	#FFFFFF	rgb(255, 255, 255)	rgb(100%, 100%, 100%)	白色

5.4.2 长度属性值

使用 CSS 设置的页面中,通常会使用长度属性值描述文字间距、页边距、行高等信息,长度的设定可以使用绝对长度、相对长度和百分数长度等。

1. 绝对长度

绝对长度是通过使用精确的单位来设定的值,主要包括 in(英寸)、cm(厘米)、mm(毫米)、pt(点)、pc(派卡),它们之间的度量关系为

$$1 \text{ in} = 2.54 \text{ cm} = 25.4 \text{ mm} = 72 \text{ pt} = 6 \text{ pc}$$

2. 相对长度

相对长度是相对于目前字体大小或屏幕像素所设定的值,包括 px(像素)、em(元素字体高度)、ex(字体×高度)。

3. 百分数长度

在 CSS 中可以使用百分数指定属性的长度。例如,100%表示原始大小,200%表示原始大小的 2 倍,50%表示原始数据的一半。

5.4.3 布局属性

布局属性用于设定 XML 元素在整个页面所处的位置。

1. 显示属性

XML 不同于 HTML,由于 XML 没有内置的层次结构,因此使用 CSS 时需要设置 display 属性,用于在建立布局时生成显示框类型。display 属性常用的值包括以下几种。

- ① none: 隐藏或显示属性值,表示元素不会显示。
- ② block: 段落显示属性值,表示元素显示为段落样式,元素前后会带有换行符,使得元素处于一个独立的段落中。
- ③ inline: 行内显示属性值,也是默认属性值,表示元素前后没有换行符,显示在页面的一行中。
- ④ list-item: 列表显示属性值,表示该元素以列表方式显示。列表可以进一步通过 list-style-type、list-style-position 或 list-style-image 属性来修饰。

以文件 5-2-1.xml 为例,设置格式如文件 5-4-3-1.css 所示。

文件 5-4-3-1. css

```
name { display:list-item;
        list-style-type:disc;
        margin:30px; }
card { display:list-item;
        list-style-type:square;
        margin:30px; }
passwd { display:list-item;
        list-style-type:circle;
        margin:30px; }
```

文件 5-2-1. xml 引用文件 5-4-3-1. css, 在浏览器显示的效果如图 5-6 所示。

2. 边界属性

CSS 的边界属性用于设定元素与页面上、下、左、右边界的距离, 属性包括 margin-top、margin-bottom、margin-left、margin-right 和 margin。

- ① margin-top: 设置上边界距离。
- ② margin-bottom: 设置下边界距离。
- ③ margin-left: 设置左边界距离。
- ④ margin-right: 设置右边界距离。
- ⑤ margin: 用于同时设置上、右、下、左边界的距离。

边界属性值可以使用绝对长度、相对长度、百分比长度设定。

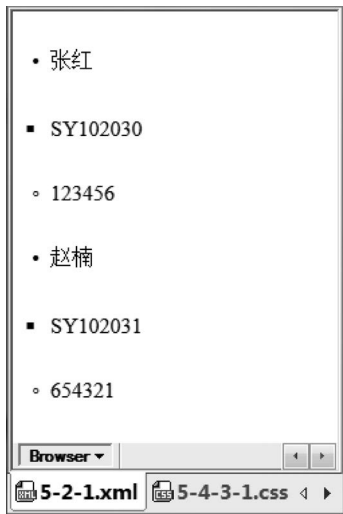


图 5-6 使用 CSS 的显示效果

3. 填充属性

CSS 的填充属性用于设定元素边框与元素内容之间的空间。属性包括 padding-top、padding-bottom、padding-left、padding-right 和 padding。

- ① padding-top: 设置上内边距。
- ② padding-bottom: 设置下内边距。
- ③ padding-left: 设置左内边距。
- ④ padding-right: 设置右内边距。
- ⑤ padding: 同时设置上、右、下、左内边距。

填充属性值可以使用绝对长度、相对长度、百分比长度设定。分别举例如下。

name{padding-top:8px;}表示上内边距为 8px。

name{padding:2cm;}表示上、下、左、右内边距都为 2cm。

name{padding:2cm 4cm 3cm 4cm;}表示上、右、下、左内边距分别为 2cm、4cm、3cm 和 4cm。

5.4.4 边框属性

在 display 属性值为 block 的情况下, 边框属性可以设定元素边框的样式、颜色和宽度。

1. 边框样式属性

边框样式属性包括 border-top-style、border-bottom-style、border-left-style、border-right-style 和 border-style。

- ① border-top-style: 设定上边框样式。
 - ② border-bottom-style: 设定下边框样式。
 - ③ border-left-style: 设定左边框样式。
 - ④ border-right-style: 设定右边框样式。
 - ⑤ border-style: 同时设定上、右、下、左边框的样式。
- 边框样式的属性可以有多种,常用边框样式的属性值如表 5-2 所示。

表 5-2 常用边框样式的属性值

值	描 述
none	定义无边框,默认值
dotted	定义点画线
dashed	定义虚线
solid	定义实线
double	定义双实线
groove	定义 3D 凹槽边框,效果取决于 border-color 的值
nidge	定义 3D 垄状边框,效果取决于 border-color 的值
inset	定义 3D inset 边框,效果取决于 border-color 的值
outset	定义 3D outset 边框,效果取决于 border-color 的值

以文件 5-2-1.xml 为例,设置格式如文件 5-4-4-1.css 所示。

文件 5-4-4-1.css

```
name { display:block;
        border-style: dotted;
        margin:20px; }
card { display:block;
        border-style:solid;
        margin:20px; }
passwd { display:block;
        border-style: dotted solid double dashed;
        margin:20px; }
```

文件 5-2-1.xml 引用文件 5-4-4-1.css,在浏览器显示的效果如图 5-7 所示。

2. 边框颜色属性

边框颜色属性包括 border-top-color、border-bottom-color、border-left-color、border-right-color 和 border-color。

- ① border-top-color: 设定上边框颜色。

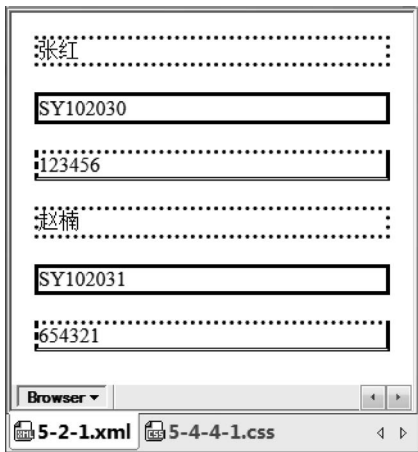


图 5-7 使用 CSS 的显示效果

- ② border-bottom-color: 设定下边框颜色。
- ③ border-left-color: 设定左边框颜色。
- ④ border-right-color: 设定右边框颜色。
- ⑤ border-color: 同时设定上、右、下、左边框的颜色。

边框颜色属性可以使用任何合法的颜色属性值。

以文件 5-2-1.xml 为例,设置格式如文件 5-4-4-2.css 所示。

文件 5-4-4-2.css

```
name { display:block;
        border-style:dotted;
        border-color: #0000ff;
        margin:20px; }
card { display:block;
        border-style:solid;
        border-color:rgb(100%,0%,0%);
        margin:20px; }
passwd { display:block;
        border-style: dotted solid double dashed;
        border-color: #FF0000 #00FF00 rgb(250,0,255) rgb(100%,0%,0%);
        margin:20px; }
```

文件 5-2-1.xml 引用文件 5-4-4-2.css,在浏览器显示的效果如图 5-8 所示。

3. 边框宽度属性

边框宽度属性包括 border-top-width、border-bottom-width、border-left-width、border-right-width 和 border-width。

- ① border-top-width: 设定上边框宽度。
- ② border-bottom-width: 设定下边框宽度。
- ③ border-left-width: 设定左边框宽度。
- ④ border-right-width: 设定右边框宽度。
- ⑤ border-width: 同时设定上、右、下、左边框的宽度。

边框宽度属性值可以使用绝对长度、相对长度和特定关键字设定。特定关键字包括 thin、medium 和 thick,分别表示细的边框、中等边框(默认值)和粗的边框。

以文件 5-2-1.xml 为例,设置格式如文件 5-4-4-3.css 所示。

文件 5-4-4-3.css

```
name { display:block;
        border-style:dotted;
        border-color: #0000ff;
        border-width:thin medium thick 12px;
        margin:20px; }
```

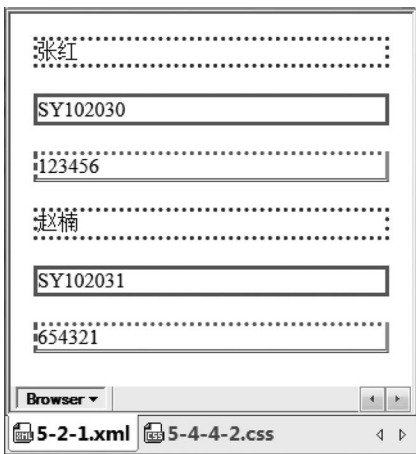


图 5-8 使用 CSS 的显示效果

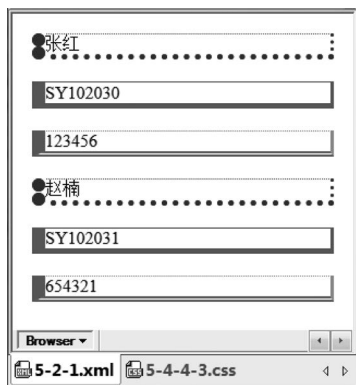


图 5-9 使用 CSS 的显示效果

```
card { display:block;
border - style:solid;
border - color:rgb(100%,0%,0%);
border - width:thin medium thick 12px;
margin:20px; }
passwd { display:block;
border - style: dotted solid double dashed;
border - color: # FF0000 # 00FF00 rgb(250,0,255) rgb
(100%,0%,0%);
border - width:thin medium thick 12px;
margin:20px; }
```

文件 5-2-1.xml 引用文件 5-4-4-3.css,在浏览器显示的效果如图 5-9 所示。

5.4.5 背景属性

背景属性可以对元素的背景颜色、背景图像、背景重复、背景定位、背景关联等进行设置。

1. 背景颜色属性

在 CSS 中可以使用 background-color 属性为元素设置背景颜色,该属性接受任何合理的颜色属性值。

以文件 5-2-1.xml 为例,设置格式如文件 5-4-5-1.css 所示。

文件 5-4-5-1.css

```
name { display:block;
background - color:rgb(255,0,255);
margin:20px; }
card { display:block;
background - color:blue;
margin:20px; }
passwd { display:block;
background - color:rgb(0%,100%,0%);
margin:20px; }
```

文件 5-2-1.xml 引用文件 5-4-5-1.css,在浏览器显示的效果如图 5-10 所示。

2. 背景图像属性

background-image 用于设定背景图像,默认情况下,background-image 的属性值是 none,表示背景上没有设置任何图像。如果需要设置背景图像,需要为这个属性设置一个 URL 值。

以文件 5-2-1.xml 为例,设置格式如文件 5-4-5-2.css 所示。

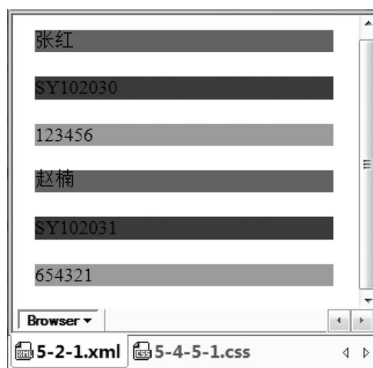


图 5-10 使用 CSS 的显示效果

文件 5-4-5-2. css

```
users { display:block;
        background-image:url(image.jpg);
        margin:20px; }
```

文件 5-2-1. xml 引用文件 5-4-5-2. css,在浏览器显示的效果如图 5-11 所示。由图可见,背景图像在垂直方向和水平方向重复显示。



图 5-11 使用 CSS 的显示效果

需要注意的是,如果同时指定背景颜色和背景图像,背景图像会覆盖背景颜色。

3. 背景重复属性

如果需要在页面上对背景图像进行平铺,可以使用 background-repeat 属性。常用背景重复的属性值如表 5-3 所示。

表 5-3 常用背景重复属性值

值	描 述
repeat	背景图像在垂直方向和水平方向重复,为默认值
repeat-x	背景图像在水平方向重复
repeat-y	背景图像在垂直方向重复
no-repeat	不允许图像在任何方向上平铺

以文件 5-2-1. xml 为例,设置格式如文件 5-4-5-3. css 所示。

文件 5-4-5-3. css

```
users { display:block;
        background-image:url(image.jpg);
        background-repeat:no-repeat;
        margin:20px; }
```

文件 5-2-1. xml 引用文件 5-4-5-3. css,在浏览器显示的效果如图 5-12 所示。由图可见,图像没有在任何方向上平铺。

4. 背景定位属性

在 CSS 中可以利用 background-position 属性定位背景图像,用于改变图像在背景中的位置。默认情况下,背景图像显示在页面的左上角。通常情况下该属性的属性值可以分为



图 5-12 使用 CSS 的显示效果

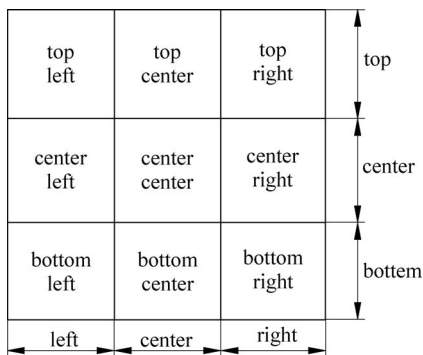


图 5-13 相对位置属性值的示意图

3 类,即相对位置、绝对位置和百分比位置。

相对位置可以使用 top(顶部)、bottom(底部)、center(居中)、left(左侧)、right(右侧)等关键字表示。其中,垂直位置关键字(top、bottom、center)和水平位置关键字(center、left、right)需要结合使用表示位置的属性值。相对位置属性值的示意图如图 5-13 所示。

注意: 如果 background-position 的属性值仅使用了一个关键字,则第二个关键字的值默认为 center。

绝对位置可以使用 CSS 中长度属性值设定,包括绝对长度单位和相对长度单位。

百分比位置是使用百分比数值设定图像位置。百分比位置属于相对位置设置的一种方式。如果 background-position 设定的值为 0%,表示该图像位于页面左上角;如果 background-position 设定的值为 100%,表示该图像位于页面右下角。因而图像可以在 0%~100%页面的相对位置进行设置。

以文件 5-2-1.xml 为例,设置格式如文件 5-4-5-4.css 所示。

文件 5-4-5-4.css

```
users { display:block;
        background-image:url(image.jpg);
        background-repeat:no-repeat;
        background-position:center;
        margin:20px; }
```

文件 5-2-1.xml 引用文件 5-4-5-4.css,在浏览器显示的效果如图 5-14 所示。由图可见,图像显示在界面的居中位置。

5. 背景关联属性

background-attachment 是用于设定背景关联的属性,其属性常用的值包括以下两个。

① scroll: 默认值,表示在文本比较长的情况下,当文本向下滚动时,背景图像也会随之滚动;当文本滚动到超过图像的位置时,图像就会消失。



图 5-14 使用 CSS 的显示效果

② fixed: 表示该元素声明的图像相对于可见区域是固定的,不会受到滚动的影响。
例如:

```
users { background - image:url(image.gif);  
         background - attachment:fixed; }
```

6. 背景图像尺寸属性

通常情况下,背景图像的尺寸是由图像自身大小决定的。但是 CSS3 推出的 background-size 属性可以设定背景图像的尺寸,它主要通过像素或百分比数值在横向和纵向上改变图像尺寸。如果以百分比规定尺寸,那么尺寸是相对于原始图像的宽度和高度设定的。

以文件 5-2-1. xml 为例,设置格式如文件 5-4-5-5. css 所示。

文件 5-4-5-5. css

```
users { display:block;  
        background - image:url(image.jpg);  
        background - repeat:no - repeat;  
        background - size : 80px 80px;  
        margin:20px; }
```

文件 5-2-1. xml 引用文件 5-4-5-5. css,在浏览器显示的效果如图 5-15 所示。由图可见,图像以长 80 像素和宽 80 像素显示。



图 5-15 使用 CSS 的显示效果

7. 背景属性

background 是一个简写的背景属性,它能够将以上规定的部分或全部背景属性设置在

声明中,通常建议使用这个属性,它需要输入的字母更少,应用更加方便。例如:

```
users { background:url(image.gif) repeat - y fixed top left; }
```

5.4.6 文本属性

文本属性可定义文本的外观。通过文本属性,可以改变文本的颜色和字间距、对齐文本、装饰文本、对文本进行缩进等。

1. 文本缩进属性(text-indent)

text-indent 属性用于实现文本的缩进,它规定块级显示样式元素的第一行缩进一个给定的长度,长度可以使用度量单位,包括精确度量单位、百分比单位等。

以文件 5-2-1.xml 为例,设置格式如文件 5-4-6-1.css 所示。

文件 5-4-6-1.css

```
name { display:block;
        text-indent:1cm;
        margin:20px; }
```

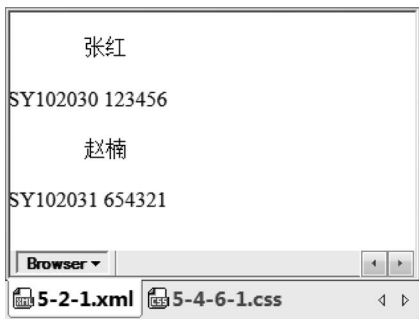


图 5-16 使用 CSS 的显示效果

文件 5-2-1.xml 引用文件 5-4-6-1.css,在浏览器显示的效果如图 5-16 所示。

2. 文本对齐属性(text-align)

text-align 属性用于设置一个元素中文本行互相之间的对齐方式,它影响 display 属性值为 block 的元素。常用属性值包括以下几个。

- ① left: 左对齐。
- ② right: 右对齐。
- ③ center: 居中对齐。

④ justify: 两端对齐。

以文件 5-2-1.xml 为例,设置格式如文件 5-4-6-2.css 所示。

文件 5-4-6-2.css

```
name { display:block;
        text-align:right;
        margin:20px; }
```

文件 5-2-1.xml 引用文件 5-4-6-2.css,在浏览器显示的效果如图 5-17 所示。

3. 字间距属性(letter-spacing)

word-spacing 属性可以改变单词之间的标准间隔,其默认值为 normal。word-spacing 属性接受一

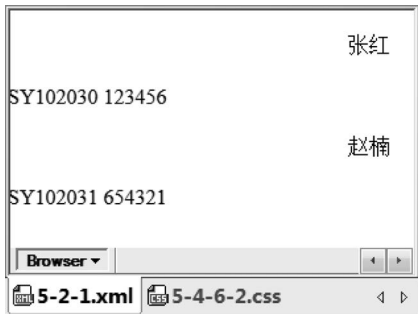


图 5-17 使用 CSS 的显示效果

个正数值或负数值,如果设为一个正数值,那么单词之间的间隔就会增加;如果设为一个负数值,单词之间的间隔缩小。需要注意的是,word-spacing 只针对英文内容有效,如果是中文字符,可以使用 letter-spacing。

以文件 5-2-1.xml 为例,设置格式如文件 5-4-6-3.css 所示。

文件 5-4-6-3.css

```
name { display:block;
        letter-spacing:100px;
        margin:20px; }
```

文件 5-2-1.xml 引用文件 5-4-6-3.css,在浏览器显示的效果如图 5-18 所示。

4. 字符转换属性(text-transform)

text-transform 是设置字符转换的属性,用于改变字母大小写,一般描述英文字符,其属性值包括以下几个。

- ① capitalize: 每个单词第一个字母大写。
- ② uppercase: 所有字母转换为大写。
- ③ lowercase: 所有字母转换为小写。
- ④ none: 默认值,不改变大小写。

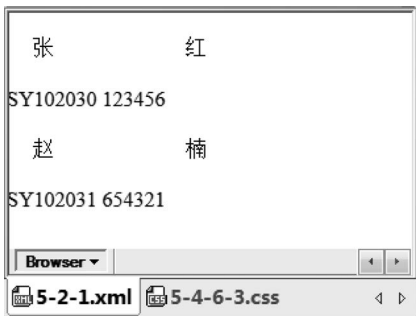


图 5-18 使用 CSS 的显示效果

5. 文字装饰属性(text-decoration)

text-decoration 属性能够为文字内容添加装饰效果,属性值包括以下几个。

- ① none: 默认值,无装饰效果。
- ② underline: 为文字添加下画线。
- ③ overline: 为文字添加上画线。
- ④ line-through: 为文字添加删除线。
- ⑤ blink: 为文字添加闪烁效果,个别浏览器支持。

以文件 5-2-1.xml 为例,设置格式如文件 5-4-6-4.css 所示。

文件 5-4-6-4.css

```
name { display:block;
        text-align:center;
        text-decoration:line-through;
        margin:20px; }
card { display:block;
        text-align:center;
        text-decoration:underline;
        margin:20px; }
passwd { display:block;
        text-align:center;
        text-decoration:overline;
        margin:20px; }
```

文件 5-2-1.xml 引用文件 5-4-6-4.css,在浏览器显示的效果如图 5-19 所示。

6. 文本行距属性(line-height)

line-height 属性设置文本行间的距离,该属性的值可以使用数字、绝对长度、相对长度和百分数长度和关键字 normal 等表示,其中 normal 为默认值,并且该属性不能使用负数值。

以文件 5-2-1 为例,设置格式如文件 5-4-6-5.css 所示。

文件 5-4-6-5.css

```
user { display:block;
        text-align:center;
        line-height:400%; }
```

文件 5-2-1.xml 引用文件 5-4-6-5.css,在浏览器显示的效果如图 5-20 所示。



图 5-19 使用 CSS 的显示效果



图 5-20 使用 CSS 的显示效果

5.4.7 字体属性

在 CSS 中定义多种设置字体的属性,包括字体类型、字体风格、变形、加粗和大小等。

1. 字体类型属性(font-family)

通常情况下,页面中的字体由系统或浏览器的默认值确定。在 CSS 中可以使用 font-family 属性设置页面中不同的字体类型,如宋体、黑体、微软雅黑、Times New Roman 等。

需要注意的是,如果是类似 Times New Roman 这种多个字符组合的字体,属性值需要使用双引号。例如:

```
name { font-family:"Times New Roman"; }
```

2. 字体风格属性(font-style)

font-style 属性用于设置字体风格,该属性的值包括以下几个。

① normal: 默认值, 文本正常显示。

② italic: 文本斜体显示。

③ oblique: 文本倾斜显示。

其中 italic 是一种简单的字体风格, 用于设置斜体字, 而 oblique 文本是指倾斜的文字, 对于没有斜体的字体, 可以使用 oblique 属性值来实现倾斜的文字效果。

以文件 5-2-1.xml 为例, 设置格式如文件 5-4-7-1.css 所示。

文件 5-4-7-1.css

```
@charset "gb2312";
name { display:block;
        font-family:华文彩云;
        font-style:italic;
        margin:20px; }
```

文件 5-2-1.xml 引用文件 5-4-7-1.css, 在浏览器显示的效果如图 5-21 所示。

3. 字体变形属性(font-variant)

font-variant 属性设置小型大写字母的字体显示文本, 这意味着所有的小写字母均会被转换为大写, 但是所有使用小型大写字母的字母与其余文本相比, 其字体尺寸更小。该属性的值包括以下几个

① normal: 默认属性值, 用于保持字体原有状态。

② small-caps: 将字体设为小型的大写字母。

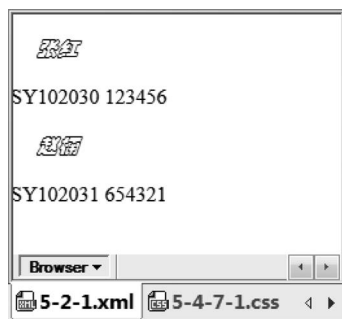


图 5-21 使用 CSS 的显示效果

4. 字体加粗属性(font-weight)

font-weight 属性用于设置字体的粗细程度, 该属性可以使用多种关键字设定, 该属性的值包括以下几个。

① normal: 默认值, 标准的字体显示。

② bold: 粗体字体显示。

③ bolder: 更粗的字体显示。

④ lighter: 更细的字体显示。

⑤ 100~900: 定义由粗到细的字体, 属性值 400 等同于 normal、700 等同于 bold。

以文件 5-2-1.xml 为例, 设置格式如文件 5-4-7-2.css 所示。

文件 5-4-7-2.css

```
name { display:block;
        font-weight:900;
        margin:20px; }
```

文件 5-2-1.xml 引用文件 5-4-7-2.css, 在浏览器显示的效果如图 5-22 所示。



图 5-22 使用 CSS 的显示效果

5. 字体大小属性(font-size)

font-size 属性用于设置字体的大小,它可以使用多种方式设定字体大小,包括特定关键字、绝对尺寸、相对尺寸、百分比尺寸等。

其中,特定关键字描述的属性值包括 xx-small、x-small、small、medium、large、x-large 和 xx-large,依次表示由小到大显示字体,其中 medium 为默认值,表示字体正常显示。此外,还可使用关键字 smaller 或 larger 设置比父元素更小或更大的尺寸。

以文件 5-2-1.xml 为例,设置格式如文件 5-4-7-3.css 所示。

文件 5-4-7-3.css

```
name { display:block;
        font-weight:900;
        font-size:xx-large;
        margin:20px }
```

文件 5-2-1.xml 引用文件 5-4-7-3.css,在浏览器显示的效果如图 5-23 所示。

6. 字体属性(font)

font 是一个字体的简写属性,它可以在声明中设置所有字体属性,通常按照规定的顺序设置字体属性,即 font-style、font-variant、font-weight、font-size/line-height、font-family。如果不设置其中的某个值,则未设置的属性会使用默认值显示。例如:

```
name { display:block;
        font: bolder italic small-caps 40pt Arial; }
```



图 5-23 使用 CSS 的显示效果

5.5 在 XML 中引用 CSS 文件

在 XML 中使用 CSS,通常需要将 CSS 文本引入 XML 文本中方可使用。引入 CSS 文本有两种方式:一种为内嵌式;另一种为外部引用方式。一般情况下,建议使用外部引用方式,因为它符合数据表示与数据显示相分离的要求,同时外部 CSS 文件能同时被多个 XML 使用;而使用内部 CSS 会破坏 XML 文档的可读性,本书只讲解外部引用方式。

在 XML 中使用外部引用 CSS 文本的方式,需要使用以下处理指令:

```
<?xml-stylesheet type="text/css" href="样式表的 URI"?>
```

其中,type 表示样式表的 MIME 类型,CSS 的 MIME 类型为 text/css; href 表示样式表文件的相对路径或绝对路径。

5.6 CSS 应用实例

下面通过一个示例综合应用 CSS 规范显示 XML 文档。

文件 5-6-1. xml

```
<?xml version = "1.0" encoding = "gb2312"?>
<?xml - stylesheet type = "text/css" href = "5 - 6 - 1. css" ?>
< members >
    < name >会员信息一览表</name >
    < member >
        < card > SY102030 </card >
        < information >
            < name >张红</name >
            < birthday > 1986 - 02 - 15 </birthday >
            < address >金地滨河小区 2 号楼 203 室</address >
            < tel > 13233339999 </tel >
        </information >
    </member >
    < member >
        < card > SY102031 </card >
        < information >
            < name >赵楠</name >
            < birthday > 1988 - 11 - 20 </birthday >
            < address >万科新里程 5 号楼 1103 室</address >
            < tel > 13212341122 </tel >
        </information >
    </member >
    < member >
        < card > SY102032 </card >
        < information >
            < name >王刚</name >
            < birthday > 1990 - 08 - 25 </birthday >
            < address >青年居易 2 号楼 203 室</address >
            < tel > 18612345678 </tel >
        </information >
    </member >
</members >
```

文件 5-6-1. css

```
@charset "gb2312";
member, name, information, birthday, address, tel
{
    text-align :center ;
    display :block ;
}
member
{
    margin:3px;
    padding :1cm;
    background-color : #FFFFFF;
```

```
}  
name  
{ margin:10px;  
  font-size:50pt;  
  font-family:华文彩云 ;  
  text-align:center ;  
}  
information name  
{ font-size:30pt;  
  font-family:黑体 ;  
}  
member card  
{ font-size:25pt;  
  text-decoration:underline;  
  font-family:"Arial Black" ;  
}  
information  
{ font-size:25pt;  
  margin-top:1cm;  
  font-family:仿宋;  
}
```

文件 5-6-1. xml 引用文件 5-6-1. css, 在浏览器显示的效果如图 5-24 所示。



图 5-24 使用 CSS 的显示效果

5.7 小结

CSS 层叠样式表是一种用来表现 HTML 或 XML 等文件样式的语言,使用样式表技术,可以使枯燥无味的标记语言呈现不同的样式。多变的样式技术不仅提供给用户视觉上的享受,同时在数据处理和数据交互上也起到重要作用。

在 CSS 中,可以定义不同的属性描述元素的样式,CSS 技术能够对网页中的布局、背景、颜色、字体以及其他样式效果实现更加精确的控制。使用 CSS,只需简单地改变样式,对应的元素就会自动更新,因而呈现不同的显示效果。在 XML 中使用 CSS,需要在 XML 中加入相关指令方可见效。

CSS 在使用中,存在一定的局限性,由于它的初衷是为 HTML 服务,因而定义的 XML 文本中的元素不能使用中文,更为适合 XML 文档的样式表为 XSL 可扩展样式表。

5.8 习题

1. 选择题

- (1) 下面()不是 CSS 的特点。
A. 方便控制页面布局
B. 数据显示和内容相分离
C. 加快页面访问速度
D. 增加了页面文件大小
- (2) 下列()是 CSS 的布局属性。
A. Border-style B. font C. display D. line-height
- (3) ()选择器结合“#”符号设置相应属性,并且只允许定义一个唯一的属性值。
A. 类选择器 B. ID 选择器 C. 通用选择器 D. 后代选择器
- (4) 下列()不属于字体属性。
A. font-family B. font-style C. font-variant D. font-color
- (5) 独立的层叠样式表的后缀名为()。
A. css B. xml C. xsl D. html

2. 简答题

- (1) 什么是 CSS? 它有什么作用?
- (2) 在 XML 中如何引用 CSS 文本?
- (3) 在 CSS 属性值中,颜色值的 RGB 有几种表现方式? 分别是什么?
- (4) 什么是类选择器? 什么是 ID 选择器? 它们有什么区别?
- (5) 已知一个 XML 文档片段如下:

```
<?xml version="1.0" encoding="gb2312"?>
<book>
  <name>XML 基础教程</name>
  <author>靳新</author>
```

</book>

编写相应的 CSS 文档,使得< book>元素中的信息显示效果为:

- ① 使用块样式方式显示;
- ② 文字颜色为蓝色(blue);
- ③ 背景颜色为黄色(yellow)。

3. 上机操作

使用 XML 中的相关标记,制作一份个人简历,然后使用 CSS 设计该 XML 文档。