

第 4 章 HTML 标记语言

经过第 1 章的 JSP 概述和第 3 章的实例介绍,读者应该对 JSP 基本结构和编写过程有了清晰的了解。JSP 是由 HTML 标签和嵌入的 Java 脚本语言组成,其中还会使用到浏览器脚本语言 JavaScript 和层叠样式表 (CSS),这些也是 Ajax 高级应用的一个组成部分。这一章以及下面两章将向读者简单地介绍这三方面的知识,这些都是在 Web 开发过程中经常使用到技术。

本章要点包括以下内容:

- ☐ HTML 基本标签的使用方法
- ☐ HTML 格式标签的使用方法
- ☐ HTML 文本标签的使用方法
- ☐ HTML 图像标签的使用方法
- ☐ HTML 表格标签的使用方法
- ☐ HTML 链接标签的使用方法
- ☐ HTML 框架标签的使用方法
- ☐ HTML 表单标签的使用方法

4.1 HTML 简介

HTML 是英文 HyperText Markup Language 的缩写,中文意思是“超文本标志语言”,用它编写的文件的扩展名是.html 或.htm。HTML 是一组用来表示数据显示格式的标签,并由浏览器负责解析和处理。开发者可以使用记事本、写字板或类似 UltraEdit 等编辑工具来编写 HTML 文件(开发前期建议使用 Dreamweaver 调试格式,后期修改使用 UltraEdit,两者的结合使用是最佳的)。HTML 语言使用成对的标签方法编写文件(XML 使用了更加严格的成对格式),既简单又方便,它通常使用<标签名></标签名>来表示标志的开始和结束(例如<html></html>标志对)。接下来逐一介绍 HTML 的各基本标志。

4.2 HTML 基本标签

HTML 文件中基本包括的标签如下:

- ☐ <html></html>标签: 标记 HTML 文档的开始和结束
- ☐ <head></head>标签: 标记 HTML 文档的开头部分
- ☐ <title></title>标签: 标记 HTML 文档在浏览器头部的显示标题
- ☐ <body></body>标签: 标记 HTML 文档的主体部分

4.2.1 开始结束标签<html></html>

<html>标签用于 HTML 文档的最前端,用来标识整个 HTML 文档的开始。对应的</html>标签放在

HTML 文档文件的最末端，用来表示 HTML 文档的结束。这两个标签是配套使用的。例如下面的语句：

```
<html>
<!-- Html 文档的头部 -->
<!-- Html 文档的主体部分 -->
</html>
```

代码说明：在<html></html>标签对之间可以插入 HTML 文档的头部标签<head></head>和主体标签<body></body>，这两个标签对将在下面介绍。

4.2.2 头标签<head></head>

HTML 文档的头部说明就是由标签对<head>和</head>标识的。在这两个标签标识的 HTML 文档头部还可以插入<title>和</title>标签对来标记浏览器端显示的文档标题，以及使用<script>和</script>标签对来定义 JavaScript 语句等、使用<style>和</style>标签对定义层叠样式表。<head>和</head>标签对一起使用来描述 HTML 文档的相关信息，并且它们之间的内容是不会在浏览器的框中显示出来的。

例如下面代码：

```
<html>
  <head>
    <title>...</title>
    <script>...</script>
    <style>...</style>
    .....
  </head>
  .....
</html>
```

代码说明：在<head>和</head>标签对之间可以使用<title>和</title>标签对来定义该 HTML 文档的标题，<script>和</script>标签对定义浏览器脚本语言 JavaScript，以及<style>和</style>标签对定义 CSS 格式语言。关于这些标签将在下面内容有所介绍。

注意：<title>和</title>标签对是必须嵌入到<head>和</head>标签对体中，而<script>和</script>标签对、<style>和</style>标签对则不一定。

4.2.3 标题标签<title></title>

如果读者经常浏览网页的话，一定注意过浏览器左上端显示的该页面文档的标题信息（一般称为页面的“主题”），而这些标题信息就是通过插入的<title></title>标签对来定义的。这里的<title>和</title>标签必须成对出现，并且只能放在<head>和</head>标签对之间。例如下面代码：

```
<html>
  <head>
    <title>我的首页</title>
    .....
  </head>
  .....
</html>
```

代码说明：通过以上<title>和</title>标签对的定义，该页面的标题就会显示“我的首页”信息，通过浏览器访问时，在左上端会显示出来，如图 4.1 所示。



图 4.1 <title></title>标签对定义页面标题

4.2.4 <body></body>主体标签

<body>和</body>标签对定义整个 HTML 文档的主体部分。此标签对之间可以包含诸如<p>和</p>、<h1>和</h1>以及
、<hr>等标签，使用这些标签可以定义页面中的文本、图像等信息。另外，需要注意的，<body>和</body>标签必须成对使用。

另外，<body>标签中还可以定义如表 4.1 所列的多个属性。

表 1 <body>标签中定义的各类属性

| 属性 | 用途 | 说明 |
|-----------------------------------|---|---------------------|
| <body bgcolor="red"> | 设置页面的背景颜色。 | 背景颜色设置成了红色 |
| <body background="images/bg.gif"> | 设置页面的背景图片 | images/bg.gif作为背景图片 |
| <body text="green"> | 设置文本颜色。 | 文本颜色为绿色 |
| <body BgProperties=FIXED> | 设置页面背景图片为固定 | 背景图片固定，不随滚动 |
| <body link="blue"> | 设置链接颜色。 | 链接颜色为蓝色 |
| <body vlink="red"> | 设置已使用的链接的颜色。 | 使用过的链接颜色为红色 |
| <body alink="#FFFF00"> | 设置正在被击中的链接的颜色。 | 被击中链接颜色为黄色 |
| 其它 | TopMargin上边距，LeftMargin左边距、像onclick等触发属性。 | |

说明：以上列举的各个属性可以结合使用，例如<body bgcolor="red" text="#0000ff">。在 text 属性中使用六个十六进制数表示某种 RGB（即红、绿、蓝三色的组合）颜色，例如#ff0000 对应的是红色。此外，还可以使用 HTML 语言中所给定的常量名来表示颜色：Black、White、Green、Maroon、Olive、Navy、Purple、Gray、Yellow、Lime、Agua、Fuchsia、Silver、Red、Blue 和 Teal，如<body text="Blue">表示在<body>和</body>标签对中的文本使用蓝色显示在浏览器的框内。

<body>标签中可以包含的属性不仅仅以上所介绍的那几个，有兴趣的读者可以查阅相关书籍。

4.2.5 基本标签的综合实例

介绍了以上几个最为基本的 HTML 标签之后，下面通过一个简单的实例来加深对这些基本标签的理解：

```
<html>

<head>
<title>显示在浏览器最上边蓝色条中的文本</title>
</head>

<body bgcolor="white" text="red">
<p>红色背景、蓝色文本</p>
</body>

</html>
```

代码说明：创建的该 HTML 文档是由<html>标签开始，并由</html>标签结束。另外，使用<title>和</title>标签对定义该页面的标题为“显示在浏览器最上边蓝色条中的文本”。在<body>标签中定义了 bgcolor 属性，用来指定该页面背景颜色为白色（默认为白色），文字颜色为红色（默认为黑色）。<p>

和<p>标签对嵌入在<body>和</body>标签对体中，并且定义了一行文本信息。

该实例页面的运行效果如图 4.2 所示。



图 4.2 实例 1 的运行效果

4.3 HTML格式标签

上一小节介绍了有关 HTML 文档的基本也是最为重要的标签，但有关浏览器如何显示文本之类的信息内容，读者可能还一无所知。这正是在这一小节中重点将要谈到的。在学习之前，必须强调一下，所有用于显示格式的标签统统都是用于<body></body>标签对之间。

用于 HTML 格式显示的标签包括如下：

- ☐ <p></p>标签对：该标签用于创建一个段落。
- ☐
标签：用于表示一个回车换行，该标签没有对应的结束标签。
- ☐ <blockquote></blockquote>标签对：用于表示两边缩进格式。
- ☐ <dl></dl>、<dt></dt>、<dd></dd>等标签对：用于创建普通列表。
- ☐ 、、等标签对：用于创建带有数字的列表。
- ☐ <div></div>标签对：用于标记一段文本信息，进行格式化。

4.3.1 段落标签<p></p>

该标签对是用来创建一个段落，插入在这对标签之间的文本将按照段落的格式显示。在<p>标签中可以使用 align 等属性，从而可以设置文本的对齐方式，语法格式为<p align="left|center|right">文本信息</p>。其中 align 属性值“left”表示左对齐、“center”表示居中、“right”表示右对齐。

4.3.2 换行标签

标签非常简单，它没有对应的结束标签，也没有任何属性的设置，它用来创建一个回车换行。这里在使用
标签时有一点需要注意的，将
标签加在<p></p>标签对外边，将创建一个大的回车换行；如果创建在<p></p>标签对内部，则生成的换行距离比较小。

4.3.3 缩排标签<blockquote></blockquote>

该介绍的标签对是实现页面文字的段落缩排，欲实现更宽长度的缩排可以使用多次缩排标签。基本语法：<blockquote>...</blockquote>。下面通过一个实例介绍该标签对的具体使用。

实例代码如下：

```
<html>
<head><title>文字的缩排</title></head>
<body>
<blockquote>这是《JSP 大全》</blockquote>
<blockquote><blockquote>这是《JSP 大全》</blockquote></blockquote>
<blockquote><blockquote><blockquote>这是《JSP 大全》</blockquote></blockquote></blockquote>
</body>
</html>
```

该页面的运行效果如图 4.3 所示。



图 4.3 blockquote 标签的使用

4.3.4 无序列表标签<dl></dl>、<dt></dt>、<dd></dd>

这些标签对是用来定义列表是一种两个层次的列表，以及用于解释名词的定义。<dl></dl>标签对用来定义列表的声明，<dt></dt>标签作为名词的标题，而使用<dd></dd>标签用作名词解释。示范实例的代码如下：

```
<html>
<head><title>建立定义列表</title></head>
<body>
<dl>
<dt>Photoshop</dt><dd>Adobe 公司的图像处理软件</dd>
<dt>Illustrator</dt><dd>Adobe 公司的矢量绘图软件</dd>
<dt>CorelDraw</dt><dd>Corel 公司的图形图像软件</dd>
</dl>
</body>
</html>
```

该页面代码的运行效果如图 4.4 所示。

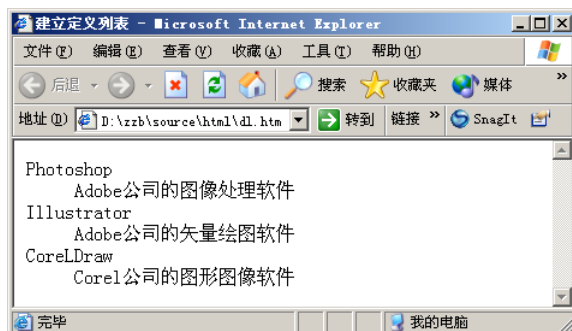


图 4.4 用于创建列表的标签演示

也可以说成<dt></dt>用来创建列表中的上层项目，<dd></dd>用来创建列表中最下层项目。另外需要注意的，<dt></dt>和<dd></dd>都必须放在<dl></dl>标志对之间。

4.3.5 有序列表标签、、

标志对用来创建一个标有序列的列表；标志对用来创建一个标有圆点的列表（即没有序列）；标志对只能在或标志对之间使用，此标志对用来创建一个列表项。另外，在标签中还可以指定 type 属性值，不同的属性值表示使用不同的序列方式，如表 4.2 所示。

表 4.2 Type属性值列表

| Type属性值 | 描述 |
|---------|----------------------|
| 1 | 数字1、2、3..... |
| a | 小写字母a、b、c..... |
| A | 大写字母A、B、C..... |
| i | 小写罗马数字i、ii、iii..... |
| I | 大写罗马数字 I、II、III..... |

下面还是通过一个实例来介绍这些标签的具体使用方法，其详细代码如下：

```
<html>
<head><title>有序和无序列表的建立</title></head>
<body>
<h2>无序列表</h2>
<ul>
  <li>项目一</li>
  <li>项目二</li>
  <li>项目三</li>
</ul>
<h2>有序列表</h2>
<ol type=I start=4>
  <li>Photoshop</li>
  <li>Illustrator</li>
  <li>Freehand</li>
</ol>
</body>
</html>
```

代码说明：标签中还可以设置 start 属性，指定从什么序列号开始。该页面运行效果如图 4.5 所示。



图 4.5 演示实例效果

4.3.6 标记标签<div></div>

<div></div>标签对用来排版大块 HTML 段落，也可以用于格式化表，此标签对的用法与<p></p>标签对非常相似，同样也有 align 对齐方式属性，读者可以自己尝试一下。

4.4 HTML文本标签

上一小节介绍的是有关文本格式的设置标签，这一小节将介绍有关字体的设置，包括斜体、黑体字以及下划线等等。最后将以一个综合实例作为演示。有关文本的标签有如下：

- ❑ <pre></pre>标签对：预格式化标签。
- ❑ <h1></h1>.....<h6></h6>等标签对：设置标题字的标签。
- ❑ <tt></tt>、<cite></cite>、、等标签对：斜体标签。
- ❑ 标签对：设置字体格式（例如大小和颜色）的标签。

4.4.1 预格式化标签<pre></pre>

该标签对是用来对文本进行预格式化操作的。所谓预格式化，就是保留文字在源代码中的显示格式，页面中显示的效果和源代码中的效果完全一样。如果使用了<pre></pre>标签，浏览器在显示其中的内容时，会完全按照其真正的文本格式来显示，即原封不动地保留文档中的空白（例如空格和制表等等）。

4.4.2 文字标题标签<h1></h1>.....<h6></h6>

HTML 语言提供了一系列对文本中的标题进行操作的标签对：<h1></h1>.....<h6></h6>，一共有六对标题的标志对。<h1></h1>标签显示最大的标题，而<h6></h6>则标记最小的标题，也即是标志中 h 后面的数字越大标题文本就越小。如果 HTML 文档中需要输出标题文本的话，便可以根据需要显示的标题大小来使用这六对标题标志对中的其中一对。

4.4.3 文字格式标签、<i></i>、<u></u>

这三个标签对的具体介绍如下：

- ❑ 标签对是用来使文本以黑体字的形式显示；
- ❑ <i></i>标签对用来使文本显示斜体；
- ❑ <u></u>标签对用来使文本显示下划线的格式。

4.4.4 文字风格标签<tt></tt>、<cite></cite>、、

这些标签对的使用与上面介绍的相类似，差别只是在显示的文本字体不太一样。具体介绍如下：

- ❑ <tt></tt>标签对用来显示打字机风格的字体文本；
- ❑ <cite></cite>标签对用来显示引用风格的字体，通常也是斜体；
- ❑ 标签对用来显示需要强调的文本，通常是黑体加斜体；
- ❑ 标签对则用来显示加重格式的文本，通常也是斜体加黑体。

这些标签对的具体使用方法将在后面综合实例中出现。

4.4.5 字体标签

标签对是一对很有用的标志对，它可以对输出文本的字体大小、颜色进行随意地改变，这些改变主要是通过对它的两个属性 size 和 color 的控制来实现的。size 属性用来改变字体的大小，它可以取值：-1、1 和+1；而 color 属性则用来改变文本的颜色，颜色的取值是本章的 4.2.2 小节中讲过的十六进制 RGB 颜色码或 HTML 语言给定的颜色常量名。具体用法请看下边综合的例子。

4.4.6 文本标签的综合实例

下面通过一个综合实例，具体演示以上所介绍的标签功能，创建的页面代码如下：

```
<html>
<head>
<title>文本标志的综合示例</title>
</head>

<body text="blue">

<h1>最大的标题</h1>
<h3>使用 h3 的标题</h3>
<h6>最小的标题</h6>
<p><b>黑体字文本</b> </p>
<p><i>斜体字文本</i> </p>
<p><u>下加一划线文本</u> </p>
<p><tt>打字机风格的文本</tt></p>
<p><cite>引用方式的文本</cite></p>
<p><em>强调的文本</em></p>
<p><strong>加重的文本</strong></p>
<p><font size="+1" color="red">size 取值"+1"、color 取值"red"时的文本</font></p>
</body>
```



```
</html>
```

该页面的运行效果如图 4.6 所示。

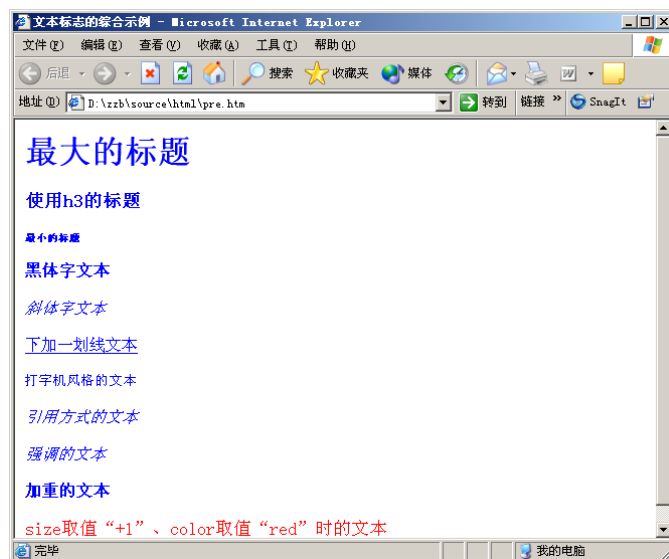


图 4.6 文本标签的示范实例

4.5 HTML 图像标签

有时需要在网页中插入图片，这时就可以使用标签，该标签允许在页面中插入一个图片。有时也需要在页面中进行显示数据的简单分隔处理，这时可以使用

水平线标签。这两个标签的具体讲解如下：

4.5.1 图片标签

页面中经常需要使用到图片，图片可以起到美化页面的效果。但是仅仅使用标签还不够，通常还需要在标签中配置相应的属性。标签的基本使用语法如下：

```

```

可以设置的属性如表 4.2 所示。

表 4.2 标签属性

| 可设置属性 | 描述 |
|--------------|----------------------------------|
| SRC | 设置图片的路径，同样可以引用网络上的图片URL地址 |
| ALT | 当鼠标停放在该图片上时，会出现相应文字提示 |
| Width、Height | 设置图片的宽度、高度 |
| Border | 设置图片的边框宽度 |
| Vspace | 设置图片的垂直间距 |
| Hspace | 设置图片的水平间距 |
| Align | 设置left、center或者right使图片向左、中或者右靠齐 |

各类属性的具体设置方法，见下面的综合实例。

4.5.2 分隔标签<hr/>

<hr/>水平线标签用于某段文字与另一段文字之间的分隔，使文档结构更加清晰、明朗，使文字的编排更加整齐。该标签自身也可以带有多个属性，例如 size、color、width 和 noshade 属性。其中 noshade 属性不需要赋值，而是直接加入到标签中即可使用，它是用来加入一条没有阴影的水平线。水平线在 HTML 文档中经常使用到，合适使用水平线可获得非常好的效果。具体使用方法见下面的综合实例。

4.5.3 综合实例

创建一个实例，具体演示以上介绍的两个标签。页面代码如下：

```
<html>
<head>
<title>图像标签的综合实例</title>
</head>
<body>
<h2>图像标签的综合实例</h2>
<hr width="600" size="1" color="#0000FF" noshade>
<p align="center"></p>
</body>
</html>
```

该页面程序的运行效果如图 4.7 所示。

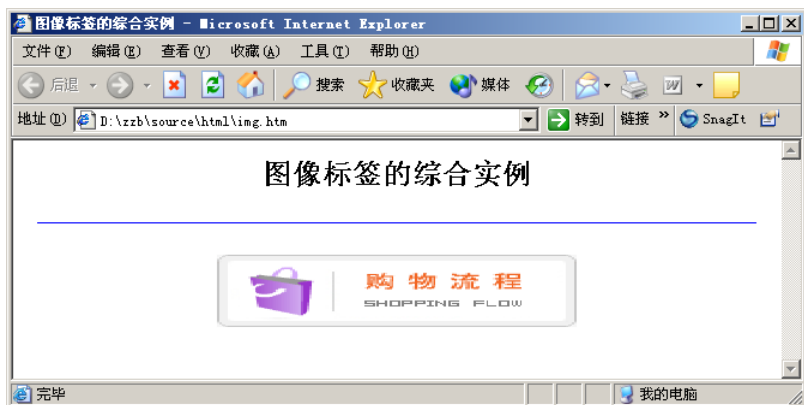


图 4.7 图像标签的综合使用

由于在标签中使用了 alt 属性，所以当鼠标停留在图片上时，会显示“购物流程—图片显示”信息提示。

4.6 HTML表格标签

表格标签在制作页面时经常使用，也非常重要，它是页面排版的灵魂。现在大部分页面都使用多重表格，主要是因为表格可以固定文本或者图像的输出。另外还是可以用它进行背景和前景颜色的设置。与表格制作相关的诸多标签如下：

- ❑ <table></table>标签对：该标签对用来创建一个表格。
- ❑ <tr></tr>、<td></td>标签对：用来创建表格的每一行、每一列。

□ `<th></th>` 标签对：该标签对用来设置表格头。

4.6.1 初始表格标签 `<table></table>`

该标签用来初始化一个表格。在这里，读者需要详细了解该标签的属性设置。各类属性描述如表 4.3 所示。

表 4.3 `<table>` 标签属性

| <code><table></code> 标签中属性设置 | 描述 |
|--|------------------------------|
| <code><table bgcolor=""></code> | 设置表格的背景颜色 |
| <code><table border=""></code> | 设置表格边框的宽度，默认设置为 0 |
| <code><table bordercolor=""></code> | 设置边框的颜色 |
| <code><table bordercolorlight=""></code> | 设置边框明亮部分的颜色（border 值不为 0 有效） |
| <code><table bordercolordark=""></code> | 设置边框昏暗部分的颜色（border 值不为 0 有效） |
| <code><table cellspacing=""></code> | 设置表格格子之间间隔大小 |
| <code><table cellpadding=""></code> | 设置表格格子边框与其内部内容之间间隔大小 |
| <code><table width=""></code> | 设置表格的宽度，可以使用像素值或者百分比值 |

说明：以上表格中的属性可以结合使用。有关宽度、大小的单位可以使用绝对像素值或者百分比。而有关颜色的属性值设置使用十六进制 RGB 颜色码或者 HTML 语言给定的颜色变量（例如 `:red`、`green` 等）。

4.6.2 行列标签 `<tr></tr>`、`<td></td>`

`<tr></tr>` 标签对是用来创建一个表格的行，而 `<td></td>` 标签对是使用来创建一个列。并且 `<tr></tr>` 标签必须放在 `<table></table>` 标签对之间，`<td></td>` 标签放置在 `<tr></tr>` 标签对之间。具体的使用格式如下：

```
<table>
  <tr>
    <td>...</td>
    <td>...</td>
  </tr>
</table>
```

另外，需要输入的文本信息可以插入在 `<td></td>` 标签对之间。在 `<tr>` 和 `<td>` 中都可以设置 `align` 和 `valign` 等属性：

- `align` 属性用来设置水平对齐方式，可以取值 `left`（左对齐）、`center`（居中）、`right`（右对齐）；
- `valign` 是用来设置垂直对齐方式，取值可以是 `top`（靠顶端对齐）、`middle`（居中对齐）或者 `bottom`（靠底部对齐）。

在 `<td>` 标签中还可以设置 `width`、`height` 属性，用于设置表格列的宽度、行的高度。另外，读者最需要注意的是如下两个属性：

- `colspan` 属性：用于设置一个表格格子跨占的列数（默认为 1）；
- `rowspan` 属性：设置一个表格格子跨占的行数（默认为 1）。

具体属性的设置方法见下面的综合实例。

4.6.3 表格头标签<th></th>

该标签对用于设置表格头，通常显示的是黑体居中文字。详细见综合实例中的示范。

4.6.4 表格标签的综合实例

介绍完基本的表格标签后，下面还是通过一个综合实例来具体演示这些标签的使用，详细的代码如下：

```
<html>
<head>
<title>表格标签的综合示例</title>
</head>

<body>
<table border="1" width="90%" bgcolor="#E8E8E8" cellpadding="2" bordercolor="#0000FF"
bordercolorlight="#7D7DFF" bordercolordark="#0000A0">
  <tr>
    <th width="33%" colspan="2" valign="bottom">意大利</th>
    <th width="36%" colspan="2" valign="bottom">英格兰</th>
    <th width="36%" colspan="2" valign="bottom">西班牙</th>
  </tr>
  <tr>
    <td width="16%" align="center">AC 米兰</td>
    <td width="16%" align="center">佛罗伦萨</td>
    <td width="17%" align="center">曼联</td>
    <td width="17%" align="center">纽卡斯尔</td>
    <td width="17%" align="center">巴塞罗那</td>
    <td width="17%" align="center">皇家社会</td>
  </tr>
  <tr>
    <td width="16%" align="center">尤文图斯</td>
    <td width="16%" align="center">桑普多利亚</td>
    <td width="17%" align="center">利物浦</td>
    <td width="17%" align="center">阿申纳</td>
    <td width="17%" align="center">皇家马德里</td>
    <td width="17%" align="center">.....</td>
  </tr>
  <tr>
    <td width="16%" align="center">拉齐奥</td>
    <td width="16%" align="center">国际米兰</td>
    <td width="17%" align="center">切尔西</td>
    <td width="17%" align="center">米德尔斯堡</td>
    <td width="17%" align="center">马德里竞技</td>
    <td width="17%" align="center">.....</td>
  </tr>
</table>
</body>
</html>
```

实例页面的运行效果如图 4.8 所示。



图 4.8 表格标签的综合实例

4.7 HTML链接标签

链接是网页的一大特色，正是因为有了它，所有浏览器中的内容才能有其灵活和网络性。使用在链接中的标签只有[<a>](#)，它有两种使用方法。

4.7.1 链接标签[](#)

该标签中的 href 属性值是必不可少的，用来设置一个超文本链接。它可以是 URL 地址（即网址或者相对路径），还可以是“mailto:”形式，即发送 E-Mail。

例如：[百度](#)，提供“百度”网站的链接；[给我发邮件](#)，提供发送邮件的链接。

另外，该标签还存在一个 target 属性，它用来设置浏览的目标帧。如果 target 属性不进行设置，单击页面链接之后，会在当前页面上出现新的页面；如果 target 设置为“_blank”，表示打开一个新的窗口来浏览新弹出的页面。

4.7.2 标记链接标签[](#)

一般[](#)是和[](#)配合使用的。[](#)用来创建一个文本标记，以便能定位到该标记的文本信息，一般使用在篇幅比较长的页面中。而[](#)提供定位到标记位置的链接。

下面还是通过一个实例来形象地展示该标签的使用。实例代码如下：

```
<html>
<head>
<title>链接标签实例</title>
</head>
<body>
<ul>
<li><a href="#tech">科技类图书</a></li>
<li><a href="#computer">计算机类图书</a></li>
</ul>
```

```
<pre>
<a name="tech">
科技类图书：
    《UFO 的发现》
    《宇宙大爆炸》
    《生命起源》
    《转基因》
</a>

<a name="computer">
计算机类图书：
    《JAVA 编程思想》
    《JSP 实战》
    《JSP 大全》
    《J2EE》
    《Struts》
</a>
</pre>
</body>
</html>
```

代码说明：页面中使用到了之前介绍的<pre></pre>标签，它将代码编辑格式不变地显示在浏览器中。单击页面中的两个标记链接，会定位相应的位置。该标签的组合使用在篇幅特别长的页面中非常有效，类似目录，可以迅速查找到相应文本信息。

4.8 HTML框架标签

框架是由 Frame 单词翻译过来的，使用它可以向浏览器窗口中装载多个 Html 页面。这时窗口中每一个页面就占据一个框架，窗口中的所有框架就组成一个最大的框架集。框架的使用通过是在一个框架中显示目录导航，另外框架显示具体文本信息。有关框架使用的标签如下：

- ❑ <frameset></frameset>标签对：用来定义主文档中有几个框架，并且是如何排列的。
- ❑ <frame>标签：定义一个具体的框架。
- ❑ <noframes></noframes>标签对：定义在那些不支持框架的浏览器中显示的内容。

下面针对这三对标签详细介绍：

4.8.1 框架集标签<frameset></frameset>

该标签对用来定义一个框架集，然后再使用<frame>标签具体定义每个框架。在使用框架集的页面中，页面中的<body>标签将被<frameset>标签所取代。<frameset>标签中使用到的属性如表 4.4 所示。

表 4.4 框架集标签的属性表

| <frameset>标签中使用到的属性 | 描述 |
|-----------------------------------|--------------------|
| <frameset rows="value,value,..."> | Rows属性设置上下分割的各框架高度 |
| <frameset cols="value,value,..."> | Cols属性设置左右分割的各框架宽度 |
| <frameset framespacing=value> | 设置框架集边框宽度 |

| | |
|--|-----------|
| <code><frameset bordercolor="ColorValue"></code> | 设置框架集边框颜色 |
|--|-----------|

另外，框架的分割方式主要三种，具体如下：

4.8.1.1 左右分割窗口

将使用到属性 `cols`，具体形式如下：

```
<frameset cols="value,value,.....">
  <frame>
  <frame>
  .....
</frameset>
```

代码说明：其中 `value` 指定百分比，或者像素值，用来指定某个框架的宽度或者高度。具体设置方法见下面综合实例。

4.8.1.2 上下分割窗口

将使用到 `rows` 属性，具体形式如下：

```
<frameset rows="value,value,.....">
  <frame>
  <frame>
  .....
</frameset>
```

代码说明：同样，这样 `value` 值可以设置成百分比或者像素值，只是它设置的是框架的高度。

4.8.1.3 嵌套分割窗口

嵌套分隔的意思是在某框架中再嵌套一个框架集，下面通过一个具体实例来示范一下：

```
<html>
<head><title>嵌套分割窗口</title></head>
<frameset cols="20%,*">
  <frame>
    <frameset rows="30%,*">
      <frame>
      <frame>
    </frameset>
  </frameset>
</html>
```

代码说明：首先定义一个左右分割的框架集，然后再在其中一个框架中定义一个框架集。`value` 属性指定左右或者上下分割的宽度或者高度，其中“*”表示剩余的宽度或者高度。该页面程序运行的效果如图 4.9 所示。

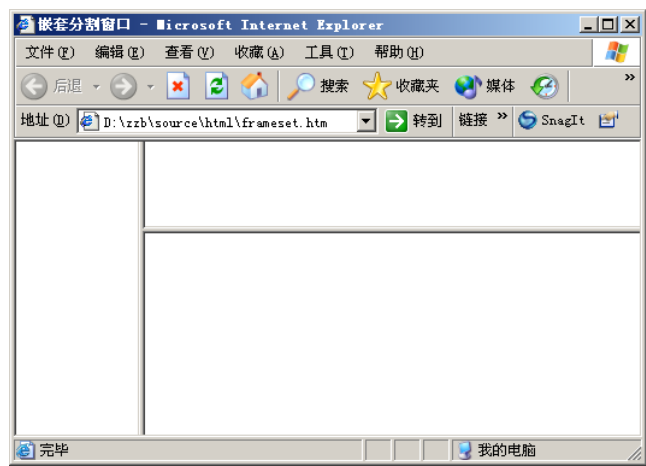


图 4.9 嵌套分割框架图

4.8.2 框架标签<frame>

<frame>标签用来装载一个具体的显示页面，这个页面文件就称为框架页面。<frame>标签必须放在<frameset></frameset>标签对之间，它有基本的 src 和 name 属性，这两个属性都是必须要设置的。其中 src 属性指定该框架装载的具体页面文件的地址，name 为该框架的名称。另外，<frame>标签中使用的其他属性如表 4.5 所示。

表 4.5 <frame>标签的属性

| <frame>标签属性 | 描述 |
|-------------------------------------|-----------------------------------|
| <frame name="FrameName" src="Path"> | 框架名称name和装载的页面文件地址属性src必须要设置 |
| <frame frameborder="0 1"> | Frameborder属性值为0表示不显示边框，否则显示 |
| <frame scrolling="yes no auto"> | Yes: 显示滚动条，no: 不显示，auto: 根据页面长度判断 |
| <frame noresize> | 该属性不设置值，代表禁止改变框架的尺寸大小 |
| <frame marginwidth="value"> | 设置框架的左右边缘宽度 |
| <frame marginHeight="value"> | 设置框架的上下边缘高度 |

4.8.3 无框架标签<noframes>

当浏览器不支持框架时，使用该标签定义需要显示的页面内容。一般在此标签对之间紧跟<body></body>标签，然后才可以使用之上介绍的各类标签。

4.8.4 框架标签的综合实例

下面通过一个实例更加形象地展示这些标签的使用方法和效果，需要创建四个页面：index.htm、menu.htm、page1.htm 和 page2.htm。

(1) index.htm 页面代码如下：

```
<html>
<head>
<title>框架标签的综合示例</title>
</head>
<frameset cols="25%,*">
```



```
<frame src="menu.htm" scrolling="no" name="Left">
<frame src="page1.htm" scrolling="auto" name="Main">
<noframes>
<body>
<p>对不起，您的浏览器不支持“帧”！</p>
</body>
</noframes>
</frameset>
</html>
```

代码说明：该代码实现了框架的左右分割，装载 menu.htm 页面的框架占据窗口左边的 25%。

(2) menu.htm 目录导航页面代码如下：

```
<html>
<head>
<title>目录</title>
</head>
<body>
<p><font color="#FF0000">目录</font></p>
<p><a href="page1.htm" target="Main">链接到第一页</a></p>
<p><a href="page2.htm" target="Main">链接到第二页</a></p>
</body>
</html>
```

代码说明：target 属性指定页面显示在哪个框架中，这里指定的是右边 main 框架。

(3) page1.htm 页面代码如下：

```
<html>
<head>
<title>第一页</title>
</head>
<body>
<p align="center"><font color="#8000FF">这是第一页！</font></p>
</body>
</html>
```

(4) page2.htm 页面代码如下：

```
<html>
<head>
<title>第二页</title>
</head>
<body>
<p align="center"><font color="#FF0080">这是第二页！</font></p>
</body>
</html>
```

运行 index.htm 页面，出现图 4.10 和 4.11 所示效果。

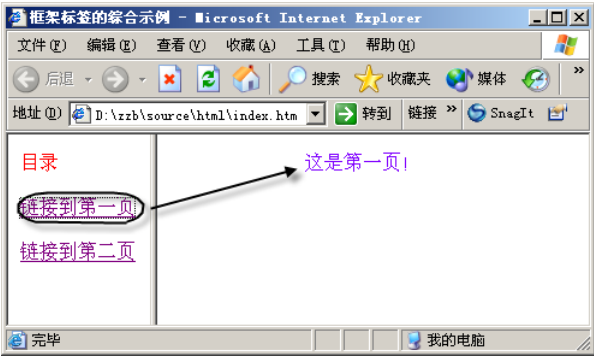


图 4.10 main 框架显示 page1.htm 页面

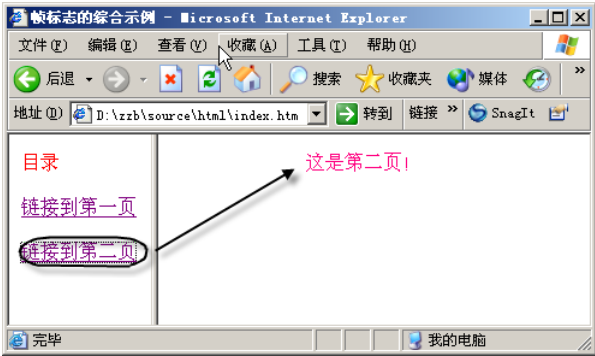


图 4.11 main 框架显示 page2.htm 页面

单击左边 left 框架中的“链接到第一页”链接，将在右边的 main 框架中显示 page1.htm 页面；单击“链接到第二页”，在 main 框架中显示 page2.htm 页面。

4.9 HTML 表单标签

表单在实际开发过程会经常使用，它是 HTML 页面和浏览器实现交互的重要手段，利用表单可以收集客户端用户提交的数据信息。表单的主要功能即为收集信息，具体讲收集用户在浏览器上填写的信息。例如，需要申请一个电子邮箱，就必须填写网站提供的表单，其内容主要有用户名、密码、真实姓名等信息。

表单信息的处理过程为：当用户单击表单中的提交按钮时，浏览器就会通过 HTTP 传输协议将表单信息传到服务器中，然后由服务器有关后台程序进行处理。一般服务器会把用户提交的信息存储在服务器端的数据库中，然后再将有关信息返回到客户端浏览器上。在表单中使用的标签包括如下：

- ❑ `<form></form>` 标签对：创建一个表单，标记表单的开始和结束
- ❑ `<input type="">` 标签：标注一个用户输入区
- ❑ `<select></select>`、`<option></option>` 标签对：创建一个下拉列表
- ❑ `<textarea></textarea>` 标签对：创建一个可以输入多行的文本框

4.9.1 表单标签 `<form></form>`

表单是页面中一个特定的区域，该区域就是通过 `<form></form>` 标签对定义。设置的该 `<form></form>` 标签对主要有两方面作用：一方面，限定表单的范围，即 `<form>` 标注表单的开始，`</form>` 标注表单的结束。另外一方面，通过在 `<form>` 标签中设置相关属性来决定表单提交给服务器端哪个应用程序处理，以及决定提交表单的方式等。

使用该标签的基本语法：

```
<form name="" method="" action="">
.....
</form>
```

`<form>` 标签中的属性依次如表 4.6 所示。

表 4.6 `<form>` 标签属性

| <code><form></code> 标签的属性 | 描述 |
|---------------------------------|------|
| Name | 表单名称 |

| | |
|--------|---------------------------------------|
| Method | 定义浏览器将表单提交给服务器的方法，使用最多的为get、post方法 |
| Action | 定义服务器端处理表单数据的应用程序，例如一个JSP、ASP或者CGI程序。 |

其中 method 属性，get 方法是将表单内容附加在 URL 地址后面，所以对提交的信息长度有限制。Post 方法是将用户在表单中填写的信息包含在表单的主体中，一起发送到服务器上进行处理，该方面没有字符长度的限制。当不指明哪种方式时，浏览器浏览器默认为 get 方式。

4.9.2 输入框标签

输入标签是表单中使用最为常见的标签之一，用来设置文本域、按钮等。基本语法：

```
<form>
.....
<input type="" name="" .....>
.....
</form>
```

标签中的属性依次如表 4.7 所示。

表 4.7 <input>标签属性

| <input>标签的属性 | 描述 |
|--------------|----------------------------|
| name | 输入域的名称 |
| Type | 指定该输入域的类型，例如text、password等 |

这里需要重点介绍一个 type 属性，在 type 属性中，可以包括如表 4.8 所示的属性值。

表 4.8 type属性值

| Type属性值 | 描述 |
|----------|-------------------|
| Text | 文字输入域 |
| Password | 密码输入域，输入的字符以“*”替代 |
| File | 文件输入域 |
| Checkbox | 复选框 |
| Radio | 单选框 |
| Button | 普通按钮 |
| Submit | 提交按钮 |
| Reset | 重置按钮 |
| Hidden | 隐藏域 |
| Image | 图片域（图像提交按钮） |

这些类型的输入框将在下面的综合实例中具体讲解。

4.9.3 下拉框标签<select></select>、<option></option>

当在表单中需要添加一项内容，例如城市，这时就不能使用单选按钮将所有的程序罗列出来选择。于是，在表单中使用菜单和列表就显得很占优势，它可以节省很多的页面空间。

<select></select>标签就是用来创建一个下拉列表框或可以复选列表框。此标签对插入到<form></form>标志对之间。<select>具有 multiple、name 和 size 属性。multiple 属性不用赋值，直接加入标志中即可使用，加入了此属性后列表框就成了可多选的了；name 是此列表框的名字，它与上边讲的 name 属性作用是一样的；size 属性用来设置列表的高度，缺省值为 1，若没有设置 multiple 属性，显示的将是一个弹出式的列表框。

<option>标志用来指定列表框中的一个选项，它放在<select></select>标志对之间。此标志具有

selected 和 value 属性，selected 用来指定默认的选项，value 属性用来给<option>指定的那一个选项赋值，这个值是要传送到服务器上的，服务器正是通过调用<select>区域的名字的 value 属性来获得该区域选中的数据项的。

下面通过一个实例来形象地演示这两个标签的使用，详细代码如下：

```
<html>
<head><title>插入菜单和列表</title></head>
<body>
<H2>用户调查</H2>
<form action="mailto:zzb@163.com" method="get" name=invest>
请选择你喜欢的音乐：<br>
<select name="music" size=4 multiple>
  <option value="rock" selected>摇滚乐
  <option value="pop">流行乐
  <option value="jazz">爵士乐
  <option value="nation">民族乐
</select><br>
请选择你居住的城市：<br>
<select name="city">
  <option value="Beijing" selected>背景
  <option value="shanghai">上海
  <option value="nanjing">南京
  <option value="Guangzhou">广州
</select>
<input type="submit" name="Submit" value="提交表单">
</form>
</body>
</html>
```

代码说明：第一个列表框定义的是多选，第二个列表框为单选。提交的处理程序为邮件地址。页面运行效果如图 4.12 所示。



图 4.12 列表框

4.9.4 文本输入框标签<textarea>

这个标签用来制作多行的文字输入框，可以在其中输入更多文本信息。基本语法如下：

```
<textarea name="name" rows=value cols=value value="value">
</textarea>
```

在标签中使用到的属性如表 4.9 所示。

表 4.9 <textarea>标签属性

| 文字域标签属性 | 描述 |
|---------|---------|
| Name | 文字域的名称 |
| Rows | 文字域的行数 |
| Cols | 文字域的列数 |
| Value | 文字域的默认值 |

该标签的使用具体见下面的综合实例。

4.9.5 表单标签的综合实例

实例页面用于输入用户的注册信息，详细代码如下：

```
<html>
<head>
<title>表单标签综合实例</title>
</head>

<body>
<form name="form1" action="http://log2.163.com/reguser.cgi">
<table width="80%" border="0" align="center">
  <tr>
    <td width="20%">&nbsp;&nbsp;&nbsp;用户名： </td>
    <td width="30%">&nbsp;&nbsp;&nbsp;<input type="text" name="username" size=15></td>
    <td width="20%">&nbsp;&nbsp;&nbsp;密码： </td>
    <td width="30%">&nbsp;&nbsp;&nbsp;<input type="password" name="userpassword" size=15></td>
  </tr>
  <tr>
    <td>&nbsp;&nbsp;&nbsp;性别： </td>
    <td>&nbsp;&nbsp;&nbsp;<input type="radio" name="sex">男&nbsp;&nbsp;&nbsp;<input type="radio" name="sex">女</td>
    <td>&nbsp;&nbsp;&nbsp;籍贯： </td>
    <td>&nbsp;&nbsp;&nbsp;
      <select name="jiguan">
        <option value="北京" selected>北京</option>
        <option value="江苏">江苏</option>
        <option value="山东">山东</option>
      </select>
    </td>
  </tr>
  <tr>
    <td width="20%">&nbsp;&nbsp;&nbsp;音乐爱好： </td>
    <td width="30%">&nbsp;&nbsp;&nbsp;<input type="checkbox" name="m1" value="rock" checked>摇滚乐
      <input type="checkbox" name="m1" value="pop">流行乐
      <input type="checkbox" name="m1" value="jazz">爵士乐
    </td>
    <td width="20%">&nbsp;&nbsp;&nbsp;上传照片： </td>
    <td width="30%">&nbsp;&nbsp;&nbsp;<input type="file" name="mypic" size="15"></td>
  </tr>
</table>
</form>
</body>
</html>
```

```
</tr>
<tr>
  <td>&nbsp;&nbsp;&nbsp;自我简介: </td>
  <td colspan="3">&nbsp;&nbsp;&nbsp;
    <textarea name="introduce" cols="45" rows="4" value="自我介绍: "></textarea></td>
</tr>
<tr>
  <td colspan="4" align="center"><input type="submit" value="提交">|<input type="reset" value="重置"></td>
</tr>
</table>

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

代码说明：在<input>标签中，指定 type 的不同属性值，可以产生文本输入框、单选按钮、复选按钮以及提交按钮和重置按钮等。<input>、<select>以及<textarea>标签一般和<form>表单标签一起组成用户信息输入区，用于提交用户信息，例如信息注册等。这几个标签在 Web 应用开发中使用非常频繁。页面运行效果如图 4.13 所示。



图 4.13 表单标签综合实例

至此，有关 HTML 常用标签基本介绍完，更详细和深入的知识（例如多媒体标签等），请有兴趣的读者自己查看相关书籍。

4.10 本章小结

本章简单介绍了 HTML 标签编写规则和使用方法。由于 JSP 就是由 HTML 标签语言（HTML 主要负责数据显示格式）和嵌入的 Java 程序片组成，所以对那些 Web 开发没有任何基础的读者来说，本章的学习还是很有必要的。下一章节将向读者介绍浏览器的脚本语言 JavaScript，它实现了客户端的浏览器和页面之间的交互，达到动态效果。并且二十一章将要介绍的 Web2.0 开发的 AJAX 就是以 JavaScript 为基础。