

Java Web 是原 Sun 公司(已被 Oracle 公司收购)在 Java Servlet 规范中提出的通用技术,指的是 仅用 HTML/JSP、JavaBean 和 Servlet 等开发互联网 Web 应用的一系列技术的总和。Java Web 是 Java EE 程序的传统开发方式,也是 Java EE 的基础。只有学好了 Java Web,才能理解 Java EE 领域各类框架 (Struts 2/Hibernate/Spring/Spring MVC/MyBatis 等)的真实用途!

2.1 HTML 语言

Java 开发的 Web 网页是由 HTML、JSP 和 Java 程序片段组成的。下面先介绍 HTML 文件的基本 构成及用法。

2.1.1 HTML 文件的基本结构

HTML 文件的基本结构如图 2.1 所示。

<html></html>		
<head></head>		
文档头部分		
<body></body>		
文档主体部分	r	

图 2.1 HTML 文件的基本结构

从图 2.1 中可以看出,基本网页从<html>标记开始,到</html>标记结束。它们之间是文档头部分和文档主体部分。文档头部分用<head>...</head>标记界定,一般包含网页标题、文档属性参数等不在页面上显示的元素。文档主体部分是网页的主体,其内容均会反映在页面上,用<body>...</body>标记界定,主要包括文字、图像、动画、链接等。

读者可以看下面的例子,文档头部分有<title>网页标题标记,文档主体部分有图片标记、<a>...链接标记及一些文字。把这段代码命名为 a.html,保存在想要保存的路径。双击该文件就可以运行了。只要该路径下有 b.html 文件,单击链接就可以跳转到 b.html。

【例 2.1】第一个网页。

```
<html>
<head>
<title>第一个 html 网页</title>
</head>
<body>
<img src="image\njnu.jpg">
<br>
```

```
<a href="b.html">链接</a>
<hr align=center>
这是我的第一个网页
</body>
</html>
```

2.1.2 HTML 文件的语言标记

从上面简单的 HTML 文件中可以看出,HTML 文件是由很多标记组成的,下面介绍几个重要的标记。

1. HTML 标记

<html>...</html>

HTML 标记表示文档内容的开始和结束。<html>是开始标记,</html>是结束标记,其他所有的HTML 代码都位于这两个标记之间。浏览器将该标记中的内容视为一个 Web 文档,按照 HTML 语言规则对文档内的标记进行解释。<html>...</html>标记是可选的,但是最好不要省略这两个标记,以保持 Web 文档结构的完整性。

2. 首部标记

<head>...</head>

首部标记中提供与网页有关的各种信息。在首部标记中,一般使用下列标记。 <title>...</title>: 制定网页的标题。 <style>...</style>: 定义文档内容样式表。 <script>...</script>: 插入脚本语言程序。 <meta>: 描述网页信息。 这些信息首先向浏览器提供,但不作为文档内容提交。

3. 标题栏标记

<title>...</title>

标题栏标记的内容是在浏览器标题栏中显示的文本。通常,Web 搜索工具用它作为索引。

4. 描述标记

<meta 属性="值"...>

描述标记描述文档属性参数。

5. 正文标记

正文标记的格式如下:

<body 属性="值"...事件="执行的程序"...>...</body>

正文标记中包含文档的内容。其常用属性如下。

- background: 文档背景图像的 URL 地址。
- bgcolor: 文档的背景颜色。
- text: 文档中文本的颜色。
- link: 文档中链接的颜色。
- vlink: 文档中已被访问过的链接的颜色。
- alink: 文档中正被选中的链接的颜色。

此外,HTML 文件还有很多用来设置文本格式的标记,下面介绍几个常用的标记。

6. 分段标记

段落是文档的基本信息单位。利用分段标记,可以忽略文档中原有的回车和换行。定义一个新段 落,就是换行并插入一个空行。

单独使用标记时会空一行,使后续内容隔一行显示。同时使用,则将段落包围起来, 表示一个分段的块。其最常用的属性为 align,其值如下。

left: 左对齐(默认值) center: 居中对齐 right: 右对齐 justify: 两边对齐

下面介绍的属性也遵循这一规则。

7. 换行标记

该标记强行中断当前行,使后续内容在下一行显示。这个标记很简单,也很常用。

8. 标题标记

标题标记的格式如下:

```
<h1 属性="值"...>...</h1><h2 属性="值"...>...</h2><h3 属性="值"...>...</h3><h4 属性="值"...>...</h3><h5 属性="值"...>...</h5><h6 属性="值"...>...</h6>
```

其常用属性也是 align。

9. 对中标记

<center>...</center>

该标记中间的内容全部居中。

10. 块标记

<div 属性="值"...>...</div>

块标记的作用是定义文档块。其常用属性也是 align。

11. 水平线标记

<hr 属性="值"...>

在<hr>标记位置画一条线。常用属性如下。

● align: 段落的水平对齐方式,其值如下。

left: 左对齐 (默认值) center: 居中对齐 right: 右对齐

● color: 线的颜色。

● size: 线的宽度(以像素为单位)。

第1部分 实用教程

50

- width: 线的长度(像素或占页面宽度的百分数)。
- noshade: 显示一条无阴影的实线。

12. 字体标记

...

字体标记用来设置文本的字符格式,主要包括字体、字号和颜色等。常用属性如下。

- face: 字体名表。
- size: 字号值。
- color: 颜色值。设置字体的颜色。

13. 图像标记

图像标记的常用属性如下。

- src: 图像文件的 URL 地址。
- alt: 图像的简单文本说明, 在浏览器下不能显示图像或图像加载时间过长时显示该文本。
- height: 显示图像的高度(像素或百分比)。
- width:显示图像的宽度(像素或百分比)。
- align: 图像大小小于显示区域大小时的对齐方式。

使用 align 属性设置图像与文本在垂直方向的对齐方式,此时 align 属性的取值如下。

top: 图像与文本顶部对齐 middle: 图像与文本中央对齐 bottom: 图像与文本底部对齐

当图像在左右绕排文本时, align 属性的取值如下。

left: 图像居左, 文本居右 right: 图像居右, 文本居左

14. 链接标记

<a 属性: "值"...>链接内容

链接的常用属性如下。

● href: 目标端点的 URL 地址(可以包含一个或多个参数)。

如前面的例子中:

链接

单击此链接,就会跳转到名为 b.html 的页面。该属性是必选项。

● target: 窗口或框架的名称。

target 属性的取值既可以是窗口或框架的名称,也可以是如下保留字。

_blank: 未命名的新浏览器窗口

_parent: 父框架页或窗口,如果包含链接的框架不是嵌套的,则链接的目标文件加载到整个浏览器窗口中_self: 所在的同一框架或窗口

_top: 整个浏览器的窗口,并删除所有框架

2.1.3 HTML 基本说明

HTML 中常用下列描述。

1. 颜色

许多标记也用到了颜色属性,颜色值一般用颜色名称或十六进制数值来表示。

① 使用颜色名称来表示。例如,红色、绿色和蓝色分别用 red、green 和 blue 表示。

② 使用十六进制格式数值#RRGGBB 来表示, RR、GG 和 BB 分别表示颜色中的红、绿、蓝三 原色的 2 位十六进制数据。例如,红色、绿色和蓝色分别用#FF0000、#00FF00 和#0000FF 表示。表 2.1 列出了 16 种标准颜色的名称及其十六进制数值。

颜 色	名 称	十六进制数值	颜色	名 称	十六进制数值
淡蓝	aqua(cyan)	#00FFFF	海蓝	navy	#000080
黑	black	#000000	橄榄色	olive	#808000
蓝	blue	#0000FF	紫	purple	#800080
紫红	fuchsia(magenta)	#FF00FF	红	red	#FF0000
灰	gray	#808080	银色	silver	#C0C0C0
绿	green	#00FF00	淡青	teal	#008080
橙	lime	#00FF00	白	white	#FFFFFF
褐红	maroon	#800000	黄	yellow	#FFFF00

表 2.1 16 种标准颜色的名称及其十六进制数值

2. 字符实体

有些字符在 HTML 里有特别的含义,比如小于号<就表示 HTML Tag 的开始,这个小于号不显示 在网页中。如果我们希望在网页中显示一个小于号,就要讲到 HTML 字符实体。

一个字符实体以&符号打头后跟实体名字或者是#加上实体编号,最后是一个分号。最常用的字符 实体如表 2.2 所示。

表 2.2 最常用的字符实体

显示结果	说明	实体名	实体号
	显示一个空格		
<	小于	<	<
>	大于	>	>
&	&符号	&	& #38;
"	双引号	"	"
©	版权	©	©
®	注册商标	®	®
×	乘号	×	×
÷	除号	÷	÷

• • 注意:

并不是所有的浏览器都支持最新的字符实体名字。对于字符实体编号,各种浏览器都能处理。字符实体是区分大小写的。更多字符实体请参见ISO Latin-1 字符集。

3. 常用属性

有些属性在 HTML 许多标记中出现。

① 类名: class。

② 唯一标识: id。

③ 内样式: style。

第1部分 实用教程

④ 提示信息: title。

4. 常用事件

事件处理描述是一个或一系列以分号隔开的 JavaScript 表达式、方法和函数调用,并用引号引起来。当事件发生时,浏览器会执行这些代码。

事件包括窗口事件、表单及其元素事件、键盘事件、鼠标事件。

2.1.4 表单与表格

在代码中看到这样的标记: <form action="" method="">…</form>, 这就是表单。HTML 文件 中使用表单是很常见的。表格主要用来组织和显示信息,也用来安排页面布局。下面分别介绍表 单与表格。

1. 表单

表单用来从用户(站点访问者)处收集信息,然后将这些信息提交给服务器进行处理。表单中可 以包含允许用户进行交互的各种控件,例如,文本框、列表框、复选框和单选按钮等。用户在表单中 输入或选择数据后提交,该数据就会提交到相应的表单处理程序,以各种不同的方式进行处理。表单 结构如下:

<form 定义> [<input 定义>] [<textarea 定义>] [<select 定义>] [<button 定义>]

</form>

form 标记的属性如下。

● name: 表单的名称。

● method: 表单数据传输到服务器的方法。其属性值如下:

post: 在 HTTP 请求中嵌入表单数据 get: 将表单数据附加到请求该页的 URL 中

● action: 接收表单数据的服务器端程序或动态网页的 URL 地址。

● target: 目标窗口,其属性值如下。

```
_blank: 在未命名的新窗口中打开目标文档
_parent: 在显示当前文档的窗口的父窗口中打开目标文档
_self: 在提交表单所使用的窗口中打开目标文档
_top: 在当前窗口内打开目标文档,确保目标文档占用整个窗口
```

form 标记有以下事件。

● onsubmit: 提交表单时调用的时间处理程序。

● onreset: 重置表单时调用的处理程序。

下面具体介绍表单中的控件。

2. 表单: 输入控件

<input 属性="值" ... 事件="代码"...>

(1) 单行文本框。

创建单行文本框方法如下:

<input type="text" 属性="值" ... 事件="代码"...>

单行文本框的属性如下。

- name: 单行文本框的名称, 通过它可以在脚本中引用该文本框控件。
- value: 文本框的值。
- default value: 文本框的初始值。
- size: 文本框的宽度(字符数)。
- maxlength: 允许在文本框内输入的最大字符数。
- form: 所属的表单 (只读)。

单行文本框的方法如下。

- click(): 单击该文本框。
- focus(): 得到焦点。
- blur(): 失去焦点。
- select():选择文本框的内容。

单行文本框的事件如下。

- onclick: 单击该文本框时执行的代码。
- onblur: 失去焦点时执行的代码。
- onchange: 内容变化时执行的代码。
- onfocus: 得到焦点时执行的代码。
- onselect: 选择内容时执行的代码。

(2) 密码文本框。

创建密码文本框方法如下:

<input type="password" 属性 = "值"...事件="代码"...>

密码文本框的属性、方法和事件与单行文本框的设置基本相同,只是密码文本框没有 onclick 事件。 (3) 隐藏域。

创建隐藏域方法如下:

<input type="hidden" 属性= "值" ...>

隐藏域的属性、方法和事件与单行文本框的设置基本相同,只是没有 default value 属性。

(4) 复选框。

创建复选框方法如下:

<input type="checkbox" 属性 = "值" ... 事件="代码" ...>选项文本

复选框的属性如下。

- name: 复选框的名称。
- value: 选中时提交的值。
- checked: 当第一次打开表时,该复选框处于选中状态。
- defaultchecked: 判断复选框是否定义了 checked 属性。

复选框的方法如下。

- focus(): 得到焦点。
- blur(): 失去焦点。

第1部分 实用教程

```
● click():单击该复选框。
复选框的事件如下。
```

- onfocus: 得到焦点时执行的代码。
- onblur: 失去焦点时执行的代码。
- onclick: 单击该文本框时执行的代码。
- 例如,要创建以下复选框:

☑苹果 □香蕉 □橘子

应在 body 体内设置代码如下:

```
<input name="fruit" type="checkbox" checked>苹果
<input name="fruit" type="checkbox">香蕉
<input name="fruit" type="checkbox">橘子
```

(5) 单选按钮。

创建单选按钮方法如下:

```
<input type="radio" 属性 = "值"...事件="代码"...>选项文本
```

单选按钮的属性如下。

- name: 单选按钮的名称, 若干个名称相同的单选按钮构成一个控件组, 在该组中只能选中一个选项。
- value: 提交时的值。
- checked: 当第一次打开表单时,该单选按钮处于选中状态。该属性是可选的。

例如, 创建以下单选按钮:

性别: ◎男 ○女

```
应在 body 中设置代码如下:
```

```
性别: <input name="sex" type="radio" checked>男
<input name="sex" type="radio">女
```

(6) 按钮。

使用 input 标记可以在表单中添加 3 种类型的按钮:提交按钮、重置按钮和自定义按钮。创建按钮的方法如下:

```
<input 属性="值"...onclick="代码">
```

按钮的属性如下。

- type: 按钮种类, 具体如下。
 - ▶ submit: 创建一个提交按钮。
 - ▶ reset: 创建一个重置按钮。
 - ▶ button: 创建一个自定义按钮。
- name: 按钮的名称。
- value:显示在按钮上的标题文本。 按钮的事件如下。
- onclick: 单击按钮执行的脚本代码。

(7) 文件域。

创建文件域的方法如下:

```
<input type ="file" 属性 ="值"...>
```

其中,对"属性="值""部分可以进行如下设置。

- name: 文件域的名称。
- value: 初始文件名。
- size: 文件名输入框的宽度。

3. 表单: 滚动文本框

<textarea 属性="值"...事件="代码"...>初始值</textarea>

其属性如下。

- name: 滚动文本框控件的名称。
- rows: 控件的高度。
- cols: 控件的宽度。
- readonly: 表示文本框中的内容是只读的,不能被修改。

该标记的其他属性、方法和相关事件与单行文本框基本相同。

4. 表单:选项选单

创建选项选单方法如下:

```
<select name= "值" size="值" [multiple]>
<option[selected] value="值">选项 1</option>
<option[selected] value="值">选项 2</option>
```

</select>

其属性如下。

- name: 选项选单控件的名称。
- size: 在列表中一次可以看到的选项数目, 默认值为1。
- multiple: 允许做多项选择。
- selected: 该选项的初始状态为选中。

下面就用这些控件做一个综合的例子来简单体验这些控件的用法,页面如图 2.2 所示。

【例 2.2】表单控件的使用。

将下面的文件命名为 all.html, 保存到磁盘上。

<html> <head> <title>综合实例</title> </head> <body bgcolor="#E3E3E3"> <h2 align="left">综合展现 HTML 标记</h2> <hr align="left" size="2" width="200">

```
<font size="4">下面展示表单的应用</font>
```

```
<form action="" method="post">
```

姓名: <input type="text" name="username" maxlength="10">

密码: <input type="password" name="pwd">

<input type="hidden" name="action" value="隐藏的">

```
性别: <input name="sex" type="radio" checked>男
```

```
<input name="sex" type="radio">女<br>
```

水果: <input checked="" name="fruit" type="checkbox"/> 苹果
<input name="fruit" type="checkbox"/> 香蕉
<input name="fruit" type="checkbox"/> 橘子
备注: <textarea cols="25" rows="3">滚动文本框</textarea>
专业: <select name="zy" size="1"></select>
<option value="计算机">计算机</option>
<option value="英语">英语</option>
<option value="数学">数学</option>
课程: <select multiple="" name="kc" size="3"></select>
<option value="计算机导论">计算机导论</option>
<option value="数据结构">数据结构</option>
<option value="软件工程">软件工程</option>
<option value="高等数学">高等数学</option>
<option value="离散数学">离散数学</option>
<input type="submit" value="提交"/>
<input type="reset" value="重置"/>

双击 all.html 文件,就会出现如图 2.2 所示的页面。读者也可以做一些相关的实验,把其他属性都 演示一遍。可以看出,上面所做的页面,其功能虽然实现了,但整体的外观效果不是很好,没有统一 规划。如果把它们都放在表格中,则能解决该问题。下面具体介绍表格的使用。

360安全浏览器 10.0	> 文件	查看 收藏 工	具帮助 🗎 🗕 🗆 🗙
← C △ ☆ ⊕ file:///D:/Ja	ava%20EE基础实用	敗程(第3版)/源	程序/第2章/小实例 🗲 🗸
▶ □ 综合实例	× +		C
综合展现HTMI标记			
下面展示表单的应用			
姓名:			
密码:			
性别: ●男 ○女			
水果: 🗹 苹果 🗆 香蕉 🗆 橘子			
滚动文本框			
备注:	1		
专业: 计算机 ▼			
计算机导论 🔺			
数据结构			
课程: 软件程 ▼			
提交 重置			

图 2.2 综合展现 HTML 表单标记实例

5. 表格

表格由表头、行和单元格组成,常用于组织和显示信息,还可以用于安排页面布局。表格的表头、行和单元格可分别用不同的标记来定义。可以用 table 标记定义表格;表格中的每一行用 tr 标记来表示;

行中的单元格用 td 或 th 标记定义。其中, th 标记定义表格的列标题单元格, 表格的标题说明则用 caption 标记来定义。定义表格的格式图解如图 2.3 所示。

∖tab.	10/	<caption></caption>	学生成绩	表 <\captio	n>	
	专业 ^{<\th}	^{~th>} 学号 ^{<\th>}	姓名 ^{<\th>}	计算机导论	数据结构	<\tr
	<\td>	<\td><\td><\td><	≦ ^d 本 <∖td>	80 <\td>	78 <\td>	<\tr>
	计算机	≪ nd> <\nd> 081102	程明	≥id> <\id> 90	60	<\tr>
		081104	TID> TID> 韦延平	adə sudə 83	std> std>	<\tr
	道信工程 ^{\td>}		≝啦> <\td>	<\td>	100 <\td>	<\tr>
<\ta	ble>					

图 2.3 定义表格的格式图解

【例 2.3】表格的使用。

```
<html>
<head>
 <title>学生成绩显示</title>
</head>
<body>
  <center>
  <caption>学生成绩表</caption>
    专业学号姓名计算机导论
     数据结构数据库原理
    计算机
     051101王  林
     8078
     90
    051102程  明
     8578
     91
    051103=新延平
     8488
     96
    通信工程
     051104王  敏
     8381
     80
    </center>
</body>
</html>
```

把这段代码保存,命名为 table.html,双击该文件会看到如图 2.4 所示的页面。





1) 设置表格的属性

用 table 标记创建表格时,可以设置以下属性。

● align: 表格的对齐方式, 其属性值如下。

left: 左对齐(默认值) center: 居中对齐 right: 右对齐

- background: 表格背景图片的 URL 地址。
- bgcolor: 表格的背景颜色。
- border: 表格边框的宽度(像素), 默认值为0。
- bordercolor: 表格边框的颜色, border≠0 时起作用。
- bordercolordark: 三维边框的阴影颜色, border≠0 时起作用。
- bordercolorlight: 三维边框的高亮显示颜色, border≠0 时起作用。
- cellpandding: 单元格内数据与单元格边框之间的间距(像素)。
- width: 表格的宽度(像素或百分比)。

2) 设置行的属性

表格中的每一行都是用 tr 标记来定义的,可以设置下列属性。

● align: 行中单元格的水平对齐方式, 其属性值如下。

```
left: 左对齐(默认值)
center: 居中对齐
right: 右对齐
```

- background: 行的背景图片的 URL 地址。
- bgcolor: 行的背景颜色。
- bordercolor: 行的边框颜色,只有当 table 标记的 border≠0 时起作用。
- bordercolordark: 行的三维边框的阴影颜色,只有当 table 标记的 border≠0 时起作用。
- bordercolorlight: 行的三维边框的高亮显示颜色,只有当 table 标记的 border≠0 时起作用。
- valign: 行中单元格内容的垂直对齐方式,其属性值如下。

top: 顶端对齐 middle: 居中对齐 bottom: 底端对齐 baseline: 基线对齐

3) 设置单元格的属性

td 标记和 th 标记的属性如下。

```
● align: 行中单元格的水平对齐方式, 其属性值如下。
```

```
left: 左对齐(默认值)
center: 居中对齐
right: 右对齐
```

- background: 单元格的背景图片的 URL 地址。
- bgcolor: 单元格的背景颜色。
- bordercolor: 单元格的边框颜色, 只有当 table 标记的 border≠0 时起作用。
- bordercolordark: 单元格的三维边框的阴影颜色,只有当 table 标记的 border≠0 时起作用。
- bordercolorlight: 单元格的三维边框的高亮显示颜色, 只有当 table 标记的 border≠0 时起作用。
- colspan: 合并单元格时一个单元格跨越的表格列数。
- rowspan: 合并单元格时一个单元格跨越的表格行数。

● valign: 单元格中文本的垂直对齐方式, 其属性值如下。

```
top: 顶端对齐
middle: 居中对齐
bottom: 底端对齐
baseline: 基线对齐
```

● nowrap: 若指定该属性,则要避免 Web 浏览器将单元格里的文本换行。

下面再来看看【例 2.2】中 all.html 文件使用表格后的情况,布局后的页面如图 2.5 所示。可以看出,效果非常明显,布局明显比之前美观。

360安全浏览器 10.0	> 文件 查看 收藏 工具 帮助 (■ _ □ ×
← C ☆ ☆ file:	///D:/Java%20EE基础实用教程(第3版)/源程序/第2章	動化 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
▷ □ 综合实例	× +	C 5
	综合展现HTML标记	
	下面展示表单的应用	
姓名:		
密码:		
性别:	●男 ○女	
水果:	●苹果 □ 香蕉 □ 橘子	
备注:	滚动文本框	
专业:	计算机 ▼	
课程:	计算机导论 ▲ 数据结构 软件工程 ▼	
提交	重置	

图 2.5 使用表格后的表单

【例 2.4】用表格呈现表单。

将下面的文件命名为 form_table.html,保存到磁盘上:

```
<html>
<head>
<title>综合实例</title>
```

```
<body bgcolor="#E3E3E3">
```

```
<h2 align="center">综合展现 HTML 标记</h2>
      <hr align="center" size ="2" width ="300" >
      <center>
      <font size="4">下面展示表单的应用</font>
      <form action="" method="post">
      姓名: 
                maxlength ="10" size="20">
          密码: input type="password" name="pwd" size="21">
          性别: >= "sex" type="radio" checked>男
          <input name="sex" type="radio">女
          <input name="fruit" type="checkbox">香蕉
          <input name="fruit" type="checkbox">橘子
          专业: w: size="1">
                <option value="计算机">计算机</option>
                <option value="英语">英语</option>
                <option value="数学">数学</option>
             </select>
          课程: 
             <select name="kc" size="3" multiple>
                <option value="计算机导论">计算机导论</option>
                <option value="数据结构">数据结构</option>
                <option value="软件工程">软件工程</option>
                <option value="高等数学">高等数学</option>
                <option value="离散数学">离散数学</option>
             </select>
          <input type= "submit" value ="提交"/>
          <input type="reset" value ="重置"/>
          </form>
   </center>
</body>
</html>
```

2.1.5 多框架文档

框架可以将浏览器窗口划分为若干窗格,在每个窗格中都可以显示一个网页,从而可以取得在同 一个浏览器窗口中同时显示不同网页的效果。框架可以嵌套。

框架网页通过一个 frameset (框架集)标记和多个 frame (框架)标记来定义。在框架网页中,将 frameset 标记置于 head 之后,以取代 body 的位置,还可以使用 noframes 标记生成不能被浏览器显示时的替换内容。框架网页的基本结构如下:

```
<html>
<head>
<title>框架网页的基本结构</title>
</head>
<frameset 属性="值"...>
<frame 属性="值"...>
<frame 属性="值"...>
...
<frame 属性="值"...>
...
</frameset>
</html>
```

1. 框架集

```
<frameset 属性="值"...>
```

</frameset>

可以使用 frameset 标记的下列属性对框架的结构进行设置。

- cols: 创建纵向分隔框架时指定各个框架的列宽。取值有 3 种形式,即像素、百分比(%)和 相对尺寸(*)。
 - ▶ cols="*,*,*": 表示将窗口划分成3个等宽的框架。
 - ▶ cols="30%, 200, *": 表示将窗口划分为3个框架,其中第1个占窗口宽度的30%,第2个为200像素,第3个为窗口的剩余部分。
 - ▶ cols="*,3*,2*": 表示左边的框架占窗口宽度的 1/6,中间的框架占窗口宽度的 1/2,右边的 框架占窗口宽度的 1/3。
- rows: 横向分隔框架时各个框架的行高。
- frameborder: 框架周围是否显示三维边框。
- framespacing: 框架之间的间隔(以像素为单位, 默认值为0)。

例如,创建一个嵌套框架集。

```
<html>
<
```

```
<body>
此网页使用了框架,但您的浏览器不支持框架。
</body>
</noframes>
</frameset>
</html>
```

2. 框架

62

<frame 属性="值"...>

frame 标记具有下列属性。

- name: 框架的名称。
- frameborder: 框架周围是否显示三维边框。
- marginheight: 框架的高度(以像素为单位)。
- marginwidth: 框架的宽度(以像素为单位)。
- noresize: 不能调整框架的大小。
- scrolling: 指定框架是否可以滚动,其属性值如下。

```
yes: 框架可以滚动
no: 框架不能滚动
auto: 在需要时,框架添加滚动条
```

● src: 在框架中显示的 HTML 文件。

下面结合表格及表单的特性综合展现它们的应用。

【例 2.5】学生信息管理系统。

"学生信息管理系统"主页如图 2.6 所示。



360安全浏览器 10.0	> 文件 查看 収織 上	
C C ☆ ☆ 0	file:///D:/Java%20EE基础实用教程(第3版)/测	原程序/第2章/小实例 🗲 🗸
> 在框架中显:	示网页 × +	
	学生信息管理系统	
学生成绩查询	这里是首页。	D
学生信息显示		
返回开始页面		

图 2.6 "学生信息管理系统"主页

单击"学生成绩查询"链接,出现如图 2.7 所示的页面。

第2章 Java Web 开发基础 63

360安全浏览器 10.0			> 3	文件 查看 收藏	工具 帮助	♥ _ □	×
← C û ☆ Ū file;	///D:/Java%20E	E基础实用教	如程(第3版	反)/源程序/第23	章/小实例/fra	me.html 🗲	\sim
▶ ■ 在框架中显示网	٦ ×	+				C 5	
学生信息管理系统							
学生成绩查询			ŝ	学生成绩表			
	专业	学号	姓名	计算机导论	数据结构	数据库原理	
学生信息显示		051101	王林	80	78	90	
返回开始页面	计算机	051102	程 明	85	78	91	
		051103	韦延平	84	88	96	
	通信工程	051104	王敏	83	81	80	
							_

图 2.7 学生成绩查询页面

单击"学生信息显示"链接,出现如图 2.8 所示的页面。

360安全浏览器 10.0		> X14 E/d (X80) 14	
← C û ☆ 🛡 file	:///D:/Java%20EE基础实用]教程(第3版)/源程序/第2章/小	实例/frame.html 🦩 🗸
ト 📃 🗅 在框架中显示网	页 × +		€ ⊅
	学生信息	息管理系统	
学生成绩查询		综合展现HTML标识	5
学生信息显示		下面展示表单的应用	
返回开始页面	姓名:		
	密码:		
	性别:	◎男○女	
	水果:	☑苹果 □ 香蕉 □ 橘子	
	备注:	滚动文本框	
	专业:	计算机 ▼	
	课程:	计算机导论 ▲ 数据结构 软件工程 ▼	
	提交	重置	

图 2.8 学生信息显示页面

代码实现如下。
1) head.html
<html></html>
<head></head>
<body bgcolor="#E3E3E3"></body>
<center><h1>学生信息管理系统</h1></center>
2) left.html
<html></html>
<head></head>

```
</head>
<body bgcolor="#e3e3e3">
     <a href="table.html" target="right">学生成绩查询<br><br>
     <a href="form table.html" target="right">学生信息显示<br>
     <a href="right.html" target="right">返回开始页面<br><br>
</body>
</html>
3) right.html
<html>
<head>
</head>
<body bgcolor="#e3e3e3">
     <h2 align="center">这里是首页。</h2>
</body>
</html>
4) frame.html
<html>
<head><title>在框架中显示网页</title></head>
          <frameset rows="80,*">
                <frame src ="head.html" name ="head">
                     <frameset cols ="25%,*">
                          <frame src ="left.html" name ="left">
                         <frame src ="right.html" name ="right">
                     </frameset>
          <noframes>
          <body>
          此网页使用了框架, 但您的浏览器不支持框架
          </body>
          </noframes>
          </frameset>
</html>
```

💽 💽 说明:

因在该例中用到了前面例子中的 table.html 和 form_table.html 文件,故需要放在同一文件夹下, 双击 frame.html 文件就可以进行测试了。

2.2 JSP 基础

JSP(Java Server Pages)是由原 Sun 公司倡导、许多公司一起建立的一种动态网页技术标准。 它是在传统的网页 HTML 文件(*.htm,*.html)中插入Java程序段(Scriptlet)和 JSP 标记(tag),从而 形成 JSP 文件(*.jsp)。

2.2.1 JSP 语法

1. JSP 数据定义

在 JSP 中可以用<%! 和%>定义一个或多个变量。在其中定义的变量为该页面级别的共享变量, 可以被访问此页面的所有用户访问。其语法格式如下:

```
<%! 变量声明 %>
```

```
如下面的代码片段:
```

<%!

```
String name="liu";
int i=0;
```

%>

此外,这种声明方式还可以定义一个方法或类,定义方法的格式如下:

<%!

```
返回值数据类型 函数名(数据类型,参数,...){
语句;
retum (返回值);
}
```

%>

```
定义一个类,如下面的代码片段:
```

<%!

```
public class A{...}
%>
```

2. JSP 程序块

下面这段 JSP 代码, 命名为 circle.jsp:

```
<%@ page language="java" pageEncoding="ISO-8859-1"%>
<html>
<body>
<% double r=10.0, s;
s=3.14 * r * r;
out.print(s);
%>
</body>
</html>
```

创建项目,名称为 Practice (具体操作见第1章),在项目中创建该.jsp 文件,部署、启动 Tomcat, 在浏览器中输入 "http://localhost:8080/Practice/circle.jsp",将在窗口中显示圆面积的值 "314.0"。今后 运行书中的小程序皆用这个 Practice 项目,在其中创建源文件部署执行即可。

从上面的这段代码中可以发现,在<%与%>之间是一个 Java 片段代码。这就是在 HTML 脚本中嵌入 Java 片段的方法,而在其中还可以定义数据类型,也就是说在<%与%>之间可以是任意的操作 Java 代码,这样为编写 JSP 文件带来了很大的方便。

3. JSP 表达式

从上面的例子中可以发现,要输出面积 *s* 的值,先计算 *s* 的值,然后输出结果。JSP 中提供了一种表达式,可以很方便地输出运算结果,其格式如下:

<%=Java 表达式 %>

于是,对 circle.jsp 文件的代码可以修改如下:

```
<%@ page language="java" pageEncoding="ISO-8859-1"%>
<html>
<body>
<%double r=10.0.s;
```

65

```
%>
<%=3.14*r*r%>
```

</body>

</html>

可以输出同样的运算结果。

4. JSP 指令

JSP 指令主要用来提供整个 JSP 页的相关信息和设定 JSP 页的相关属性,如设定网页的编码方式、 脚本语言及导入需要用到的包等。其语法格式如下:

<%@ 指令名 属性名="属性值"%>

常用的有3条指令: page、include 和 taglib。

1) page 指令

page 指令主要用来设定整个 JSP 文件的属性和相关功能,例如:

<%@ page contentType="text/html, charset=gb2312"%>

一般来说, page 指令还可以用于导入所需要的包, 用法如下:

<%@ page import="java.util.List" %>

2) include 指令

include 指令用于导入静态文件,如 JSP 网页文件、HTML 网页文件,但不能包含用<%=和%>表示的代表表达式的文件。其语法格式如下:

<%@ include file="被包含文件 url" %>

如有 head.jsp 文件,其内容如下:

<%@page language="java" contentType="text/html;charset=gb2312"%> <%@page import="java.sql.ResultSet"%>

现在,在另一个文件中调用它:

<%@include file="head.jsp"%>

<html>

<head><title>输出页面</title></head>

<body>这句话是我想输出的</body>

</html>

3) taglib 指令

taglib 指令语法格式如下:

<%@ taglib uri="tagLibraryURI" prefix="tagPrefix" %>

其中,uri="tagLibraryURI"指明标签库文件的存放位置。而 prefix="tagPrefix"则表示该标签使用时 的前缀。例如,在 Struts 2 中用到标签:

<%@ taglib uri="/struts-tags" prefix="s"%>

这就需要导入这段代码,此内容将在 Struts 2 中具体讲解。

5. JSP 动作

1) <jsp:param>

<jsp:param>的语法规则如下:

<jsp:param name="paramName" value="paramValue"/>