

第 9 章 使用 CSS 制作表单

表单是功能型网站中经常使用的元素，是网站交互中最重要的元素。在网页中，小到搜索按钮，大到用户注册表单，以及用户控制面板，都需要使用到表单及其表单元素。

表单元素用来收集用户信息，帮助用户进行功能性控制，表单的交互设计与视觉设计都是网站设计中的重要部分。从表单视觉设计来说，经常需要摆脱 XHTML 提供的默认的比较粗糙的视觉样式。本章将介绍如何使用 CSS 设计表单。

9.1 制作登录表单

登录表单是各大网站都会用到的一个非常实用的模块，是用来进入其他相关页面的入口，如图 9-1 所示是一个带有登录模块的网页。



图 9-1 带有登录模块的网页

本节将使用 form 表单元素以及 label 标签制作一个样式精美的登录表单。

9.1.1 label 标签语法结构

本节将使用到 XHTML 语言中表单常用到的 label 标签。在使用表单时，对于小小的单选框和复选框，总是需要用光标精确的点击到小框或是小圆上时，才能够完成交互相应，长期使用时也许会觉得这是一个非常麻烦的事情。表单可用性问题的便显露出来，一个不方便用户简单实用的表单是不可取的。XHTML 中提供了一个改善表单交互问题的标签 label，label 标签具体语法结构代码如下。

```
<label for="#" accesskey="#"></label>
```

label 标签具有提示表单的含义（标签说明）的功能，是成对出现的，以<label>开始，以</label>结束，两个 label 之间的内容是要显示的文字。label 中有两个属性是非常有用的，一个是 for，另外一个就是 accesskey，可用属性及其属性值如下。

- ❑ for：表示 label 标签要绑定的 XHTML 元素，当点击这个标签的时候，所绑定的元素将获取焦点。

```
<label for="InputBox">姓名</label><input id="InputBox" type="text">
```

- ❑ accesskey：表示访问 label 标签所绑定的元素的热键，当按下热键，所绑定的元素将获取焦点。

```
<label for="InputBox" accesskey="N">姓名</label><input id="InputBox" type="text">
```

注意：label 标签的 for 属性用于指定该标签所关联的表单元素，当 for 中所指的名称与表单某元素的 id 值相同时，该标签将与该元素关联，点击该标签的同时，该元素也将得到响应。

label 标签是提升表单可用性的简单易行的方法，在能够用到的时候使用这个标签，将会使表单的操作更舒畅。

9.1.2 制作登录表单实例

现在开始使用表单元素和 label 标签设计一个登录表单。分成两个 div 层级关系，外层为 layout 层，用来定义表单框架样式，内层为 member 层，用来放置表单等主体元素。内层 member 又分成标题区域和表单区域，分别由 .title 和 form 控制样式。

（1）制作页面的 XHTML 代码，如下。

```
<div id="layout">
  <div id="member">
    <div class="title"></div>
    <form name="form1" id="form1" method="post" action="#">
      <div>
        <label for="name">Name:</label>
        <input name="name" id="name" type="text" size="10" maxlength="10" />
      </div>
      <div>
        <label for="pass">Pass:</label>
        <input id="pass" name="pass" type="password" size="10" maxlength="10" />
        <input type="image" src="images/11_1_03.png" class="bt" />
      </div>
    </form>
  </div>
</div>
```

```

    </div>
  </form>
</div>
</div>

```

以上代码是在制作一个会员登录窗口的表单。登录窗口是一个需要由来访者输入会员账号和密码才能够进入参与相关活动的入口。在整个表单设计中，包含了输入框（input）、图片提交按钮两个表单元素和 label 标签，并且使用 CSS 改进了浏览器默认的外观。

div.title 元素指定了表单标题的图片，标题图片如图 9-2 所示。label 标签用来显示 Name 和 Pass 的文字，使用 for 属性分别指定捆绑的 XHTML 元素是 id 为 name 的 input 元素和 id 为 pass 的 input 元素。input 输入框表单元素，定义两个输入文本框和一个图片提交按钮。输入文本框一个是类型为 text 的单行文本输入框，一个是类型为 password 的密码输入框，文本框宽度都定义为 10 个，最大输入字符为 10。图片提交按钮也是通过 input 元素将类型定义为 image 来实现的，如图 9-3 所示。



图 9-2 标题图片



图 9-3 图片提交按钮

页面结构预览效果如图 9-4 所示。

（2）制作#layout 元素的 CSS 样式，代码如下。

```

#layout {
    background-color: #999999;           /*设置背景颜色*/
    width: 163px;                        /*设置宽度*/
    height: 142px;                       /*设置高度*/
    font-size: 10px;                    /*设置字体大小*/
    font-family: Arial;                  /*设置字体类型*/
}

```

在以上代码中，定义表单框架背景颜色为灰色，并为其指定宽度和高度，预览效果如图 9-5 所示。

（3）制作#member 元素的 CSS 样式，代码如下。

```

#member {
    text-align: center;                 /*设置文本居中显示*/
    width: 153px;                      /*设置宽度*/
    height: 132px;                     /*设置高度*/
    background: url(images/11_1_01.gif) repeat-y; /*设置背景样式*/
    margin: 5px;                       /*设置外边距*/
}
.title {
    margin-left: 3px;                  /*设置左外边距*/
    margin-top: 7px;                   /*设置上外边距*/
}

```

在以上代码中，定义了#member 元素的背景图片，并且纵向平铺，背景图如图 9-6 所示；指定了#member 元素的宽度和高度，以及外边距，用来放置表单各个元素。另外在.title 元素中，指定了标题图片的左和上外边距，保证图片的准确定位，预览效果如图 9-7 所示。



图 9-4 页面结构预览效果

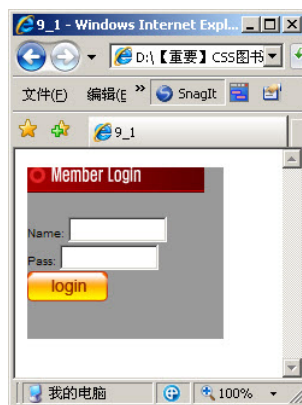


图 9-5 添加了#layout 样式的预览效果

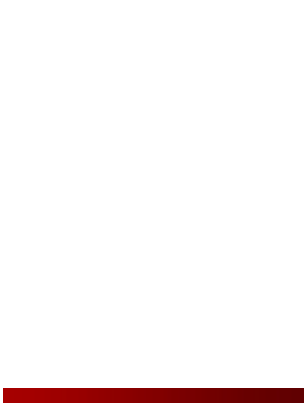


图 9-6 背景图片



图 9-7 添加了#member 和.title 样式的预览效果

从图 9-7 可以看出，输入文本框和 login 按钮都不对齐，继续添加样式。

(4) 制作表单元素的 CSS 样式，代码如下。

```
form {  
    margin: 0px; /*设置外边距*/  
    padding: 4px; /*设置内边距*/  
}  
label {  
    width: 40px; /*设置宽度*/  
    float: left; /*设置向左浮动定位*/  
    text-align: right; /*设置文本向右对齐*/  
    padding: 3px 0px 1px; /*设置内边距*/  
    color: #FFF; /*设置字体颜色*/  
}  
input {  
    border: 1px solid #000; /*设置边框样式*/  
    margin: 1px; /*设置外边距*/  
    padding: 1px; /*设置内边距*/  
    font-family: Arial; /*设置字体类型*/  
    font-size: 12px; /*设置字体大小*/  
    color: #666666; /*设置字体颜色*/  
}
```

}

以上代码是这个实例的关键。制作这个登录表单的实例，主要应用到的的是 form 元素、label 标签和 input 元素，在这里分别对它们进行定义 CSS 样式。使用标签指定式选择符 form 定义了表单外边距和内边距；使用标签指定式选择符 label 定义了文本区域的宽度、内边距、字体颜色以及浮动方式；使用标签指定式选择符 input 定义了输入框和图片按钮的边框、外边距、内边距、字体类型、大小以及颜色属性，预览效果如图 9-8 所示。

从图 9-8 可以看出，login 按钮图片周围还存在黑色的边框，这是 img 标签默认的风格，为了美观，现在需要把边框去掉，继续添加样式。

(5) 制作.bt 元素的 CSS 样式，代码如下。

```
.bt {
    width: 67px;                /*设置宽度*/
    height: 25px;               /*设置高度*/
    border: none;                /*设置边框样式*/
    margin-top: 10px;           /*设置上外边距*/
}
```

在以上代码中，为图片按钮专门定义了一套 CSS 样式，去除了标签指定式选择符 label 中定义的边框，又重新指定了上边距，使得按钮距离上面的输入框拉开一定距离，美观大方。同时为按钮定义了宽度和高度，保证图片完整显示出来。这样登录表单的实例就制作完成，预览效果如图 9-9 所示。

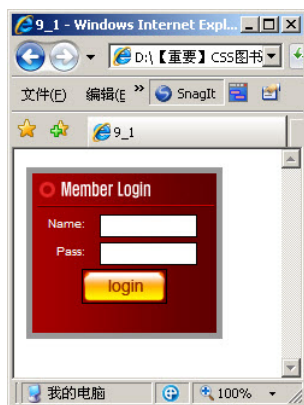


图 9-8 为表单标签添加样式的预览效果

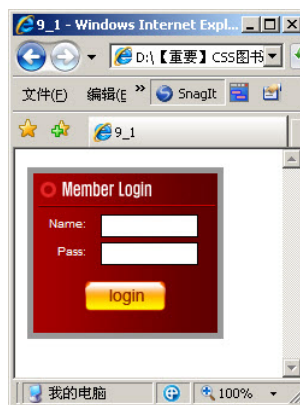


图 9-9 登录表单的预览效果

9.1.3 兼容问题

为了符合 Web 标准，制作完成的页面至少需要在 3 个主流浏览器（IE 7、Firefox、IE 6）中测试。下面首先在 Firefox 浏览器下测试。

1. Firefox 浏览器测试

在 Firefox 浏览器中打开该实例，预览效果如图 9-10 所示。从图 9-10 可以看出，本实例在 Firefox 浏览器中出现问题了，表单上边界被切下一块，问题的关键在于类标签 #layout 选择符设置的内边距 padding，重新设置 padding 的值，代码如下。

```
#layout {  
    background-color: #999999; /*设置背景颜色*/  
    width: 163px; /*设置宽度*/  
    height: 146px; /*设置高度*/  
    font-size: 10px; /*设置字体大小*/  
    font-family: Arial; /*设置字体类型*/  
    padding-top: 1px; /*设置上内边距*/  
}
```

在以上代码中，最后一行添加上了上内边距的设置 `padding-top: 1px`，这样就解决了表单上边界缺少一块灰色背景的问题，预览效果如图 9-11 所示。

2. IE 6 浏览器测试

在 IE 6 浏览器中打开该实例，预览效果如图 9-12 所示。

从图 9-12 可以看出，除了 login 图片边缘出现问题以外，其他没有样式问题。而图片边缘的问题，正是 IE 6 浏览器对 png 格式图片的不支持造成的，所以为了避免出现这样的问题，可以改为使用 jpg 等格式的图片。



图 9-10 Firefox 浏览器中的预览效果



图 9-11 解决兼容问题后 Firefox 浏览器中的预览效果



图 9-12 IE 6 浏览器中的预览效果

9.2 制作用户注册表单

表单布局除了需要应用表单中的各个元素之外，还需要使用到 `table` 表格，表格是表单布局的得力助手。对于最终的表单来说，表格对数据的排列方式非常适合于表单元格的排列。对于常见的左右式表单，即左侧为项目名称，右侧为项目输入框的形式，也非常符合表格具有表头及单元格的属性，因此使用表格进行表单排版再合适不过了。如图 9-13 所示为一个带有表单的网页。

本节尝试使用 `XHTML` 的表格将各表单元素放置于表格中。

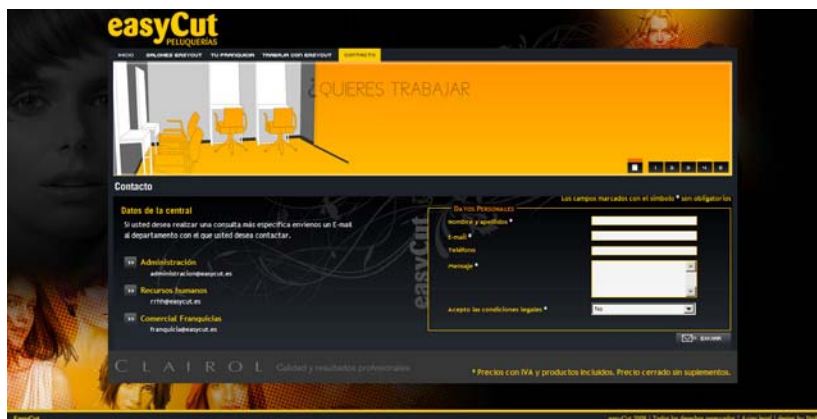


图 9-13 一个带有表单的网页

9.2.1 制作页面的 XHTML 代码

制作页面的 XHTML 代码，如下。

```
<div id="pageHolder">
  <div class="action">
    <form name="registerActionForm" method="POST" action="#">
      <h2>注册</h2>
      <p>欢迎注册，注册过程将不会收取您任何费用。我们承诺保护您的信息安全，不会将它提供给任何第三方。</p>
      <table>
        <tr>
          <td><label for="username">用户名:</label></td>
          <td><input type="text" name="userId" maxlength="99" value="" style="width:150px" class="textInput" id="username" />
            <br/>
            <span class="note">Use alphanumeric characters ( <strong>-</strong>, <strong>_</strong> and <strong>.</strong> are okay too)</span></td>
        </tr>
        <tr>
          <td><label for="password">密码:</label></td>
          <td><input type="password" name="password" maxlength="99" value="" class="textInput" /></td>
        </tr>
        <tr>
          <td><label for="passwordAgain">确认密码:</label></td>
          <td><input type="password" name="passwordConfirm" maxlength="99" value="" class="textInput" id="passwordAgain" /></td>
        </tr>
        <tr>
          <td><label for="email">电子邮件:</label></td>
          <td><input type="text" name="email" value="" style="width:250px" class="textInput" id="email" /></td>
        </tr>
      </table>
      <p>请设置机密问题，当此账号修改密码时会要求您回答机密问题，这将确保是您本人在进行密码修改操作，使您的账号更安全。</p>
    </form>
  </div>
</div>
```



```

<table>
<tr>
<td><label for="secretQ">密码问题:</label></td>
<td><input type="text" name="secretQ" value="" style="width:300px" class="textInput"
id="secretQ" />
<br />
<span class="note">输入一个你可以很容易回答，别人却很难猜到答案的问题。例如“我
的一个宠物的名字叫什么？”</span></td>
</tr>
<tr>
<td><label for="secretA">问题答案:</label></td>
<td><input type="text" name="secretA" value="" style="width:300px" class="textInput"
id="secretA" /></td>
</tr>
</table>
<div class="formAction">
<input type="submit" value="提交" class="button" />
</div>
</form>
</div>
</div>

```

在以上代码中，首先为整个界面添加了一个 `div#pageHolder` 包含页面中的所有元素，以便实现表单的整体外观。然后又添加了一个 `div .action` 包含表单所有元素，表单具体内容部分放在 `<form></form>` 元素中，`h2` 元素表示的是表单标题，`p` 元素表示的是说明文字，`table` 元素里面是表单的各项注册信息，包括用户名、密码、确认密码、电子邮件以及密码提示问题等一些注册所需要填写的信息。表单代码经过表格的嵌套，使得实际表单元素整齐地分入了整个单元格中。

提交按钮使用 `input` 元素的 `submit` 属性定义的，修改其 `value` 的值为“提交”，页面结构预览效果如图 9-14 所示。

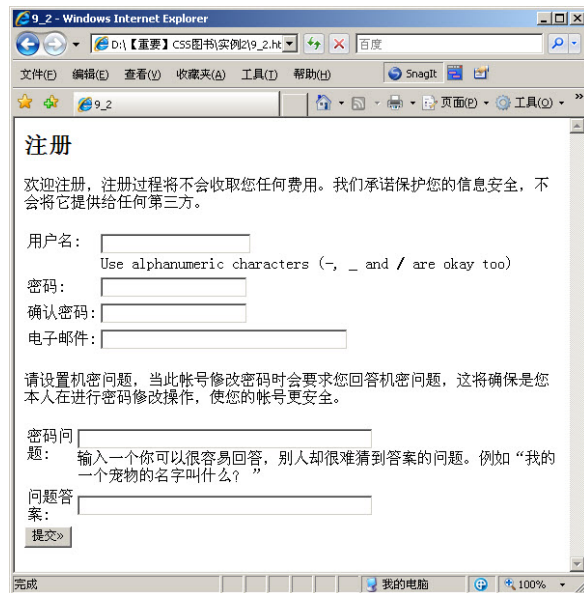


图 9-14 页面结构预览效果

9.2.2 制作页面的 CSS 样式

(1) 制作 body 的 CSS 样式，代码如下。

```
body {
    background: #eee url(images/backTileGray.jpg) repeat-x; /*设置背景样式*/
    text-align: center; /*设置文本居中显示*/
    font: 12px Arial, Helvetica, sans-serif; /*设置字体的大小、字体类型*/
    color: black; /*设置字体颜色*/
}
```

在以上代码中，先为页面的 body 标签定义了页面的背景属性，背景图片不平铺，背景如图 9-15 所示。同时定义了整个页面的文本居中对齐，字体大小、类型以及颜色，页面预览效果如图 9-16 所示。

从图 9-16 可以看出，表单内的元素全都居中显示，排列没有秩序，继续添加样式。

(2) 制作 #pageHolder 元素的 CSS 样式，代码如下。

```
#pageHolder {
    width: 500px; /*设置宽度*/
    margin: 0 auto; /*设置外边距，实现居中自适应*/
    position: relative; /*设置相对定位*/
    text-align: left; /*设置文本向左对齐*/
}
```

在以上代码中，定义了注册表单的框架的宽度，通过指定外边距 margin，使该区域居中页面显示，这是表单排版很关键的一步，预览效果如图 9-17 所示。



图 9-15 body 背景图

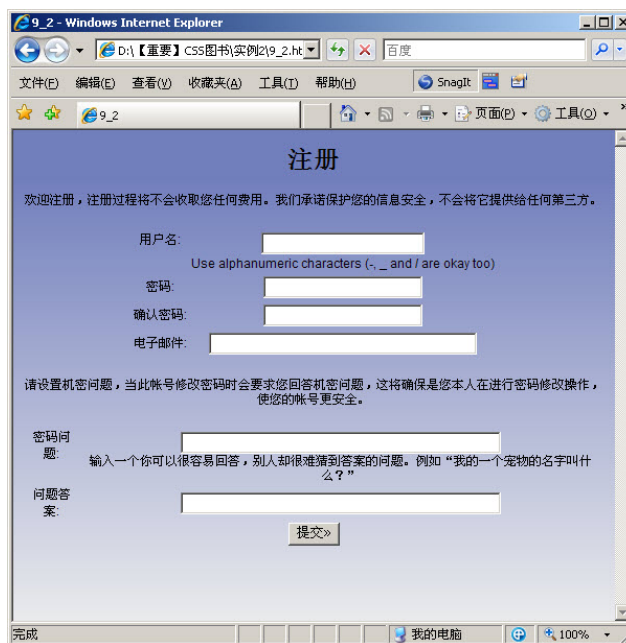


图 9-16 添加 body 样式的页面预览效果

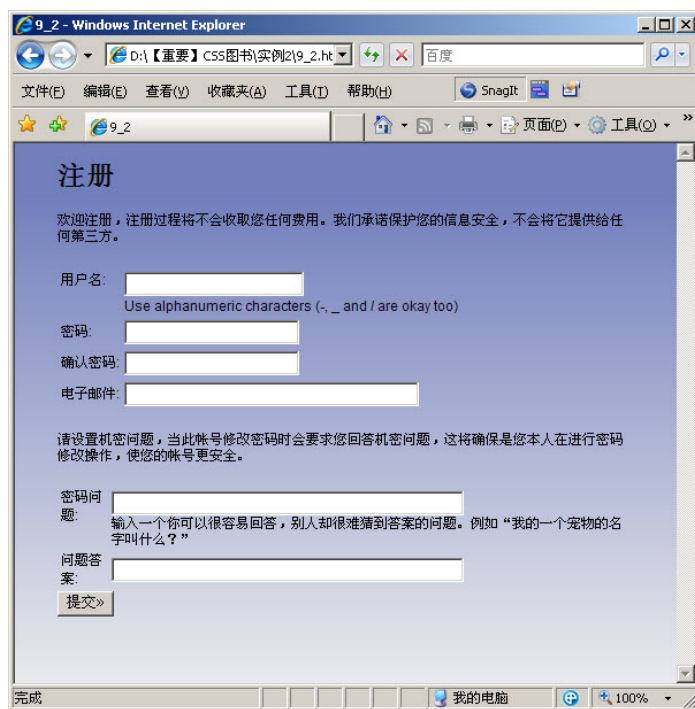


图 9-17 添加 pageHolder 样式的预览效果

(3) 制作.action 元素的 CSS 样式，代码如下。

```
.action {
    background: white;                                /*设置背景颜色*/
    padding: 12px;                                    /*设置内边距*/
    margin-top: 12px;                                  /*设置上外边距*/
    border-bottom: 1px solid #777;                    /*设置下边框样式*/
    border-left: 1px solid #aaa;                      /*设置左边框样式*/
    background: transparent url(images/bg_rounded.gif) no-repeat right top; /*设置背景样式*/
}
```

在以上代码中，定义了.action 元素的背景颜色、内边距、上外边距以及边框样式。背景定义了一个白色的圆角图片，不平铺，定位于.action 元素的右上角，指定圆角图片使页面具有设计感，预览效果如图 9-18 所示。

从图 9-18 可以看出，表单所有内容都被放在一个白色背景的区域中来。

(4) 制作 label 标签的 CSS 样式，代码如下。

```
label {
    float: left;                                       /*设置向左浮动定位*/
    width: 120px;                                     /*设置宽度*/
    text-align: right;                                 /*设置文本向右对齐*/
    margin-right: 5px;                                 /*设置右外边距*/
    font-weight: bold;                                /*设置字体为粗体*/
    font-size: 12px;                                  /*设置字体大小*/
    margin-top: 1px;                                  /*设置上外边距*/
}
```

在以上代码中，为注册信息的栏目标签定义了宽度、字体大小及字体粗细，并且指定成

浮动显示，指定了右外边距和上外边距，预览效果如图 9-19 所示。

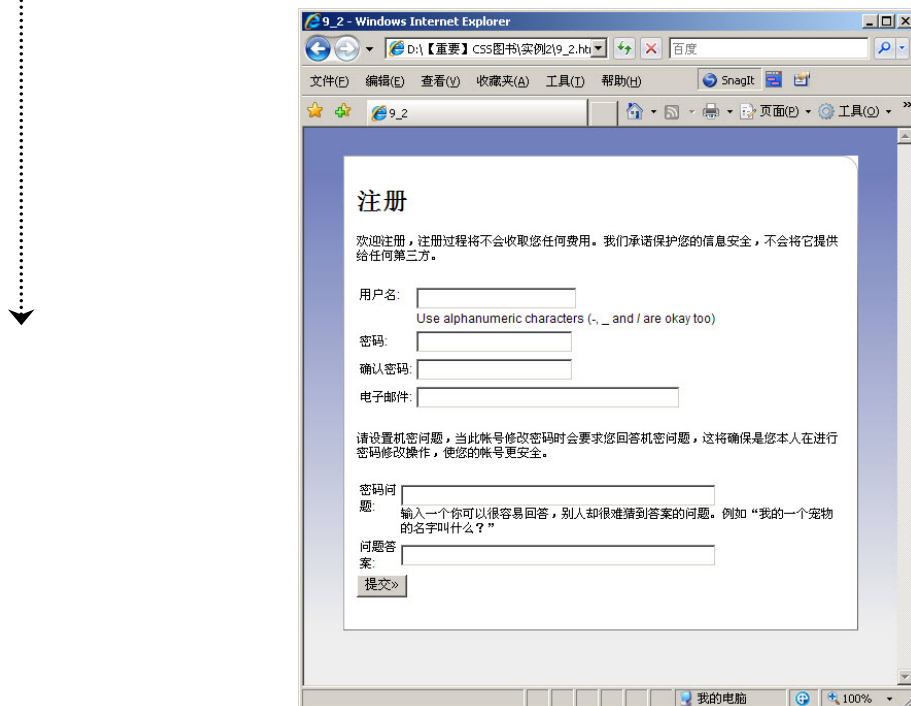


图 9-18 添加.action 样式的预览效果

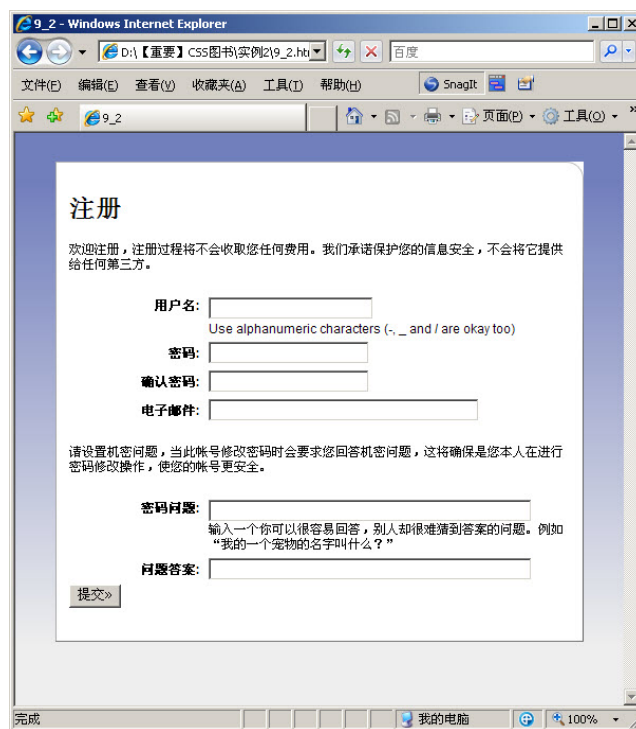


图 9-19 添加 label 样式的预览效果

(5) 制作 `input.textInput` 元素的 CSS 样式，代码如下。

```
input.textInput {
    border: 1px solid #bbb;           /*设置边框样式*/
    border-top-color: #666;          /*重新设置上边框颜色*/
    border-left-color: #666;         /*重新设置左边框颜色*/
    padding: 3px;                    /*设置内边距*/
    width: 300px;                    /*设置宽度*/
    background: white url(images/bg_textfield.gif) repeat-x top;          /*设置背景样式*/
}
```

在以上代码中，使用包含选择符 `input.textInput` 定义输入单行输入文本框的边框、内边距、宽度及背景样式，定义了不一样的上边框和左边框的颜色。为了使输入框呈现出立体的效果，背景通过 `background` 属性改变了浏览器默认的样式，重新定义了一张背景图片，横向平铺，并且居顶部显示，背景色为白色，预览效果如图 9-20 所示。

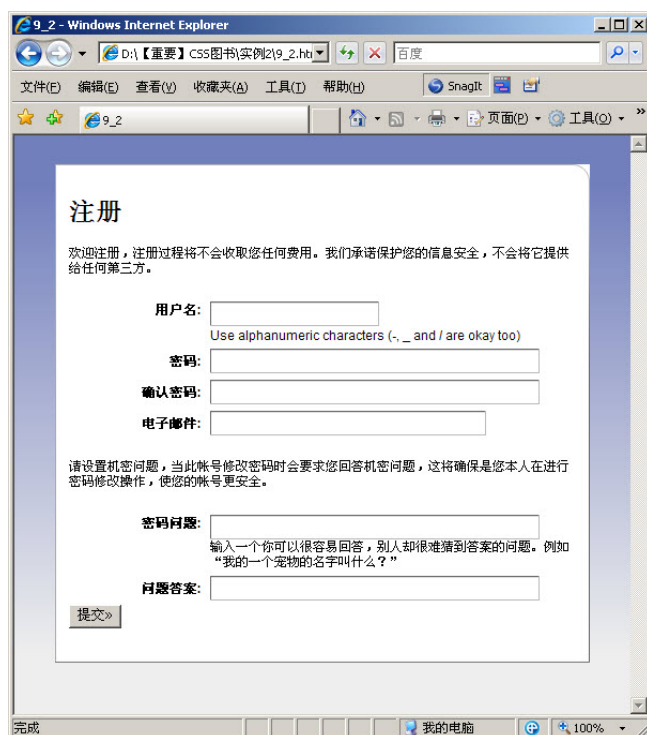


图 9-20 添加输入文本框样式的预览效果

(6) 制作提交按钮区域的 CSS 样式，代码如下。

```
div.formAction {
    margin: 20px -12px -12px -12px; /*设置外边距*/
    background: url(images/bg_gray.gif); /*设置背景样式*/
    text-align: left;                /*设置文本向左对齐*/
    padding: 12px 10px;              /*设置内边距*/
    width: 478px;                    /*设置宽度*/
    font-size: 12px;                 /*设置字体大小*/
}
div.formAction input {
```

```
font-size: 16px; /*设置字体大小*/
margin-left: 12px; /*设置左外边距*/
}
```

在以上代码中，定义的是提交按钮区域的 CSS 样式。首先，在 `div.formAciton` 元素中，定义了按钮背景区域的图片，并且将这个区域文本定义成向左对齐，指定了该区域的内边距和外边距等属性。然后再通过包含选择符 `div.formAction input` 定义了图片类型按钮的字体大小和按钮的左外边距。

(7) 制作提交按钮内部的 CSS 样式，代码如下。

```
input.button {
    border: 2px solid #aaa; /*设置边框样式*/
    border-top-color: #ddd; /*重新设置上边框颜色*/
    border-left-color: #ddd; /*重新设置左边框颜色*/
    color: #444; /*设置字体颜色*/
    font-weight: bold; /*设置文字为粗体*/
    background: url(images/bg_button.jpg) left bottom repeat-x; /*设置背景样式*/
}
```

在以上代码中，定义的是提交按钮内部样式，通过 CSS 改变了浏览器中按钮的默认样式。首先，定义了按钮的边框样式，将其左边框和上边框定义成不同的颜色，使其呈现出具有立体感的效果。然后指定了背景图片，如图 9-21 所示横向平铺，并且居左下显示。



图 9-21 按钮背景图片

这样注册表单实例就制作完成，预览效果如图 9-22 所示。

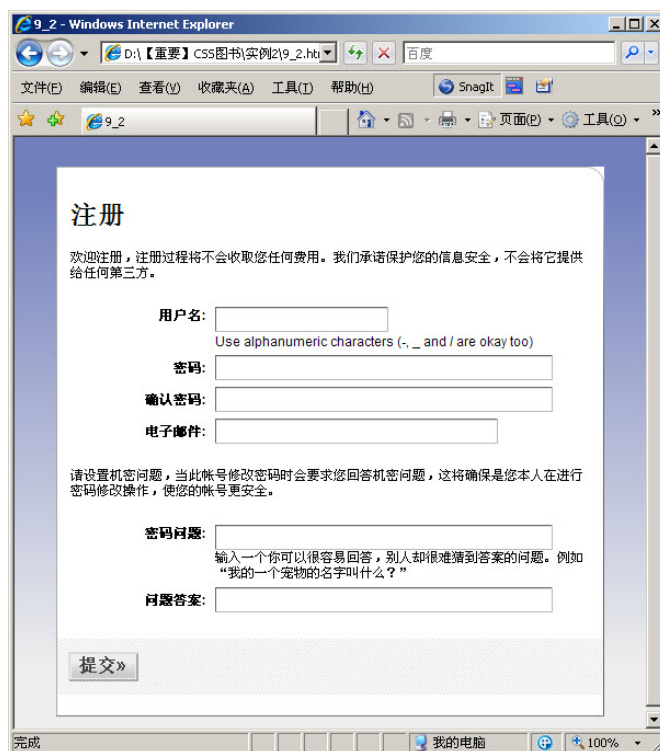


图 9-22 注册表单的预览效果

9.3 制作符合 W3C 标准的表单

所谓符合 W3C 标准的表单，就是指能够在 IE 7、Firefox 和 IE 6 等主流浏览器全兼容、符合用户体验、并且符合 Web 标准的表单。

目前 W3C 标准已经成为高端客户设计网站的首选。国内用户上网用 IE 浏览器的比较多，但从全世界的上网用户来看，有些客户并不是用 IE 来上网浏览内容的，他们会用一些其他的浏览工具，例如 Netscape、Mozilla、FireFox、Opera 等，如果网站没有采用 W3C 标准，那么使用其他浏览器的用户，就无法看到这样的网站。即此标准是国际上的通用标准，符合此标准的网站，可以使用任何浏览器来浏览。

本节将举例讲解使用 CSS 制作符合 W3C 标准的表单的方法。

9.3.1 制作页面的 XHTML 代码

(1) 制作页面总体结构的 XHTML 代码，如下。

```
<h3>Standard Form Layout</h3>
<div id="container">
  <p id="fm-intro">Fields in <strong>bold</strong> are required.</p>
  <form id="fm-form" method="post" action="#" >

    </form>
</div>
```

在以上代码中，表单的标题使用 h3 标签的默认样式，然后下面表单的主体包含在使用 id 为 container 的 div 容器里面。表单的主体由 p 标签和 form 标签组成。p 标签内的内容是表单的附加说明文字，用来提示这个表单的粗体部分为必填内容，form 标签放置表单的各个表单项内容，例如“个人信息”、“地址”、“联系方式”等都要放在这对 form 标签内。

(2) 制作“个人信息”模块的 XHTML 代码，如下。

```
<fieldset>
  <legend>Personal information</legend>
  <div class="fm-req">
    <label for="fm-firstname">First name:</label>
    <input name="fm-firstname" id="fm-firstname" type="text" />
  </div>
  <div class="fm-opt">
    <label for="fm-middlename">Middle name:</label>
    <input id="fm-middlename" name="fm-middlename" type="text" />
  </div>
  <div class="fm-req">
    <label for="fm-lastname">Last name:</label>
    <input name="fm-lastname" id="fm-lastname" type="text" />
  </div>
</fieldset>
```

在以上代码中，使用 fieldset 标签对表单分组，然后使用 legend 标签对表单每组内容进行说明，这里作为表单组的名称。表单的每一个填充项包含在 div 标签内。这里的填充项包

括“First name”、“Middle name”、“Last name”。

(3) 制作“地址”模块的 XHTML 代码，如下。

```
<fieldset>
<legend>Address </legend>
  <div class="fm-opt">
    <label for="fm-addr">Address:</label>
    <input id="fm-addr" name="fm-addr" type="text" />
  </div>
  <div class="fm-opt">
    <label for="fm-city">City or Town:</label>
    <input id="fm-city" name="fm-city" type="text" />
  </div>
  <div class="fm-opt">
    <label for="fm-state">State:</label>
    <select id="fm-state" name="fm-state">
      <option value="" selected="selected">Choose a State</option>
      <option value="UNK">Outside US / Canada</option>
      <option value="AL">Alabama</option>
    </select>
  </div>
  <div class="fm-req">
    <label for="fm-zipcode">Zip code:</label>
    <input id="fm-zipcode" name="fm-zipcode" type="text" />
  </div>
</fieldset>
```

在以上代码中，使用 fieldset 标签制作 Address 表单组，里面的填充项包括了“Address”、“City or Town”、“State”和“Zip code”。

(4) 制作“联系方式”模块的 XHTML 代码，如下。

```
<fieldset>
<legend>Contact information</legend>
  <div class="fm-req">
    <label for="fm-telephone">Telephone:</label>
    <input id="fm-telephone" name="fm-telephone" type="text" title="Enter Phone Number in
xxx-xxx-xxxx format" />
  </div>
  <div class="fm-opt">
    <label for="fm-fax">Fax:</label>
    <input id="fm-fax" name="fm-fax" type="text" title="Enter Fax Number in xxx-xxx-xxxx format"
/>
  </div>
  <div class="fm-opt">
    <label for="fm-mobile">Mobile:</label>
    <input id="fm-mobile" name="fm-mobile" type="text" />
  </div>
  <div class="fm-req">
    <label for="fm-email">Email:</label>
    <input id="fm-email" name="fm-email" type="text" tabindex="" />
  </div>
  <div class="fm-opt">
    <label for="fm-url">Web site address:</label>
    <input id="fm-url" name="fm-url" type="text" />
  </div>
  <div class="fm-opt">
    <label for="fm-comments">Comments:</label>
```



```

<textarea name="fm-comments" cols="10" rows="5" id="fm-comments"></textarea>
</div>
<div class="fm-multi">
  <div class="fm-opt">
    <p>Would you like to be notified of future updates?</p>
    <label for="fm-newsopt-yes">
      <input name="fm-newsopt" type="radio" id="fm-newsopt-yes" value="yes"
checked="checked" />
      Yes</label>
    <label for="fm-newsopt-no">
      <input id="fm-newsopt-no" name="fm-newsopt" type="radio" value="no" />
      No</label>
    </div>
  </div>
</fieldset>

```

在以上代码中，使用 fieldset 标签制作 Contact Information 表单组，里面的填充项包括了“Telephone”、“Fax”、“Mobile”、“Email”、“Web site address”、“Comments”和一组单选按钮。

(5) 制作提交按钮的 XHTML 代码，如下。

```

<div id="fm-submit" class="fm-req">
  <input name="Submit" value="Submit" type="submit" />
</div>
<br />
<div style="text-align: center;" class="fm-req"><a href="#">Back to {style:phreak;}</a></div>

```

在以上代码中，制作的是提交按钮。这样，整个表单的 XHTML 结构就编写完成，预览效果如图 9-23 所示。

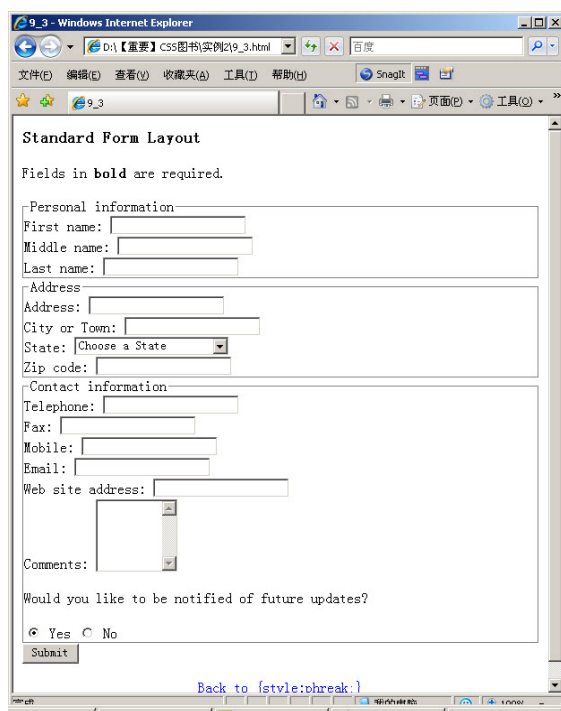


图 9-23 表单的 XHTML 结构的预览效果

9.3.2 制作页面的 CSS 样式

完成了表单的 XHTML 结构，现在开始为结构添加表现样式，即 CSS 样式表。

(1) 制作页面总体结构的 CSS 代码，代码如下。

```
body {
    padding: 0;                /*设置内边距*/
    margin: 20px;              /*设置外边距*/
    color: #333;               /*设置字体颜色*/
    background: #fff;          /*设置背景颜色*/
    font: 12px arial,verdana,sans-serif; /*设置字体样式*/
    text-align: center;        /*设置文本居中*/
}
#container {
    margin: 0 auto;            /*设置外边距，实现居中对齐*/
    padding: 1em;              /*设置内边距*/
    width: 350px;              /*设置宽度*/
    text-align: left;          /*设置文本左对齐*/
}
p#fm-intro {
    margin: 0;                 /*设置外边距*/
}
form {
    margin: 0;                 /*设置外边距*/
    padding: 0;                /*设置外边距*/
}
```

在以上代码中，首先为表单的整体结构添加 CSS 样式。使用 `body` 标签指定式选择符为页面设置了内外边距、背景色、字体类型和颜色以及文本的对齐方式。`#container` 设置表单容器的边距、宽度以及表单内文本的对齐方式，预览效果如图 9-24 所示。

(2) 制作标签 `fieldset` 的 CSS 代码，代码如下。

```
fieldset {
    margin: 1em 0;             /*设置外边距*/
    border: none;               /*设置边框为 none*/
    border-top: 1px solid #ccc; /*重新设置上边框样式*/
}
```

在以上代码中，使用 `fieldset` 标签指定式选择符为表单组设置了外边距、边框样式。首先使用 `border` 属性将边框取消，改变 `fieldset` 默认的边框样式，然后再添加 `border-top` 样式，重新设置表单组的边框样式，预览效果如图 9-25 所示。

(3) 制作标签 `legend` 的 CSS 代码，代码如下。

```
legend {
    margin: 1em 0;             /*设置外边距*/
    padding: 0 .5em;           /*设置内边距*/
    color: #036;               /*设置字体颜色*/
    font-size: 1.3em;          /*设置字体大小*/
    font-weight: bold;          /*设置字体为粗体*/
}
```

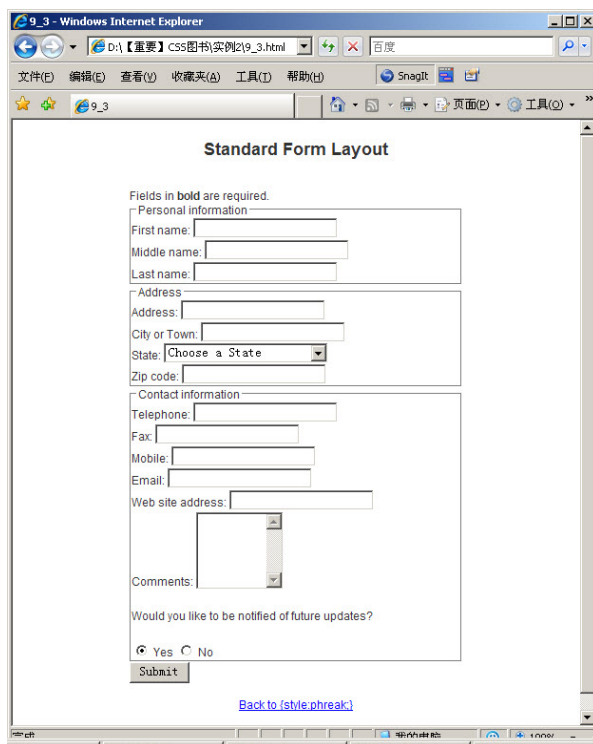


图 9-24 添加 body 等总体结构样式的表单预览效果

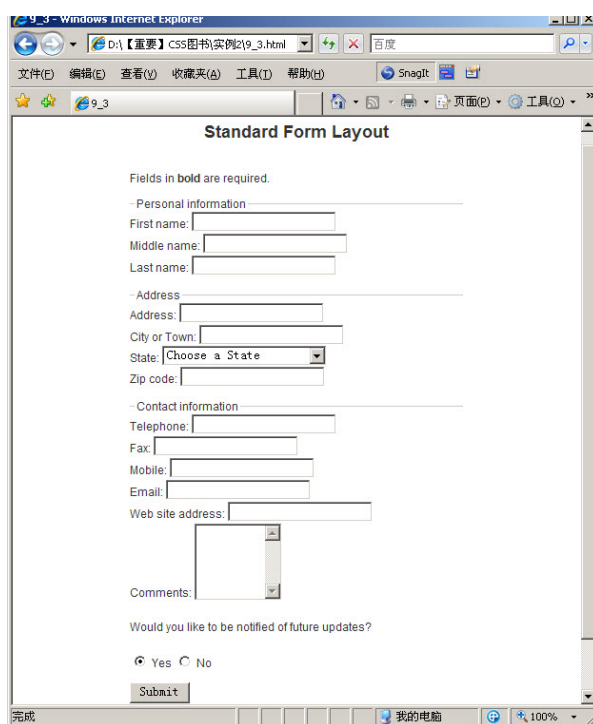


图 9-25 修改 fieldset 边框样式的表单预览效果

在以上代码中,使用 **legend** 标签指定式选择符为表单组的标题设置了内外边距、字体颜色、字体大小以及字体粗细,并且设置背景为 **transparent** 透明。经过修改表单组的标题样式,突出显示标题名称,使每个表单组分隔明显,预览效果如图 9-26 所示。

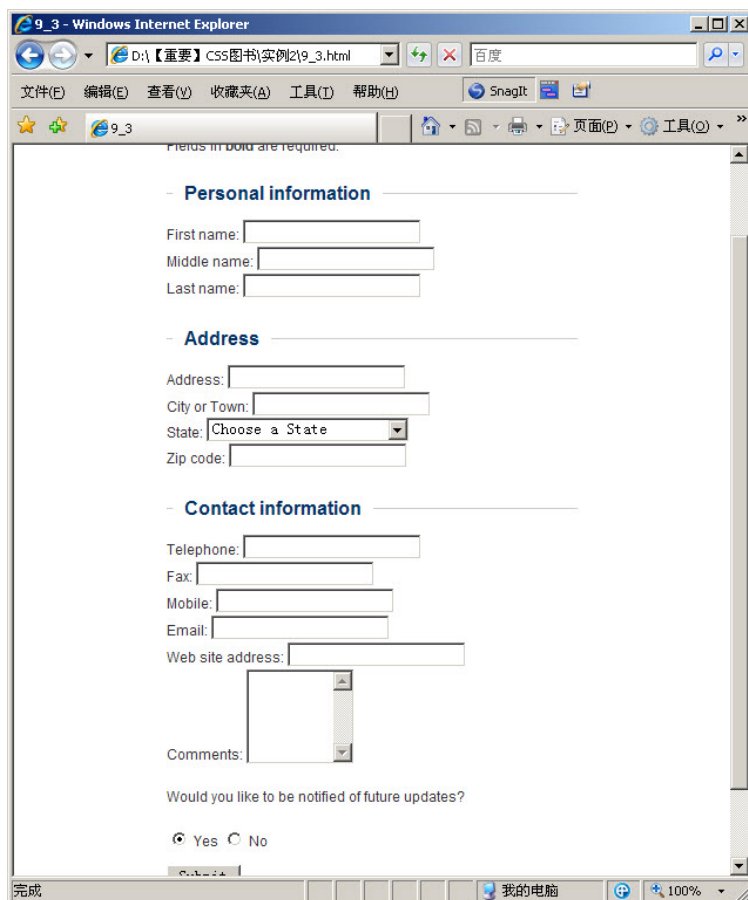


图 9-26 为 legend 标签添加 CSS 样式的预览效果

(4) 制作标签 label 的 CSS 代码,代码如下。

```
label {
    float: left;                                /*设置向左浮动样式*/
    width: 100px;                               /*设置宽度*/
    padding: 0 1em;                             /*设置内边距*/
    text-align: right;                          /*设置文本右对齐*/
}
```

在以上代码中,使用 **label** 标签指定式选择符为表单项的标题名称设置浮动属性,同时定义了标题名称的宽度、内间距以及文本对齐属性,使表单的整体工整、美观,预览效果如图 9-27 所示。

(5) 制作填充项的 CSS 代码,代码如下。

```
fieldset div {
    margin-bottom: .5em;                        /*设置下外边距*/
    padding: 0;                                /*设置内边距*/
}
```

```

display: block;                                /*设置块级元素显示方式*/
}
fieldset div input, fieldset div textarea {
    width: 150px;                              /*设置宽度*/
    border-top: 1px solid #555;                /*设置上边框样式*/
    border-left: 1px solid #555;               /*设置左边框样式*/
    border-bottom: 1px solid #ccc;             /*设置下边框样式*/
    border-right: 1px solid #ccc;              /*设置右边框样式*/
    padding: 1px;                              /*设置内边距*/
    color: #333;                               /*设置字体颜色*/
}
fieldset div select {
    padding: 1px;                              /*设置内边距*/
}

```

在以上代码中，使用包含选择符，分别为每个表单项的 `div` 容器、输入文本框、输入文本域和下拉框设置样式，主要重新设置了文本框、文本域的边框样式，调节每个表单项的间距，适当地留一定的留白，使页面简单精致，预览效果如图 9-28 所示。

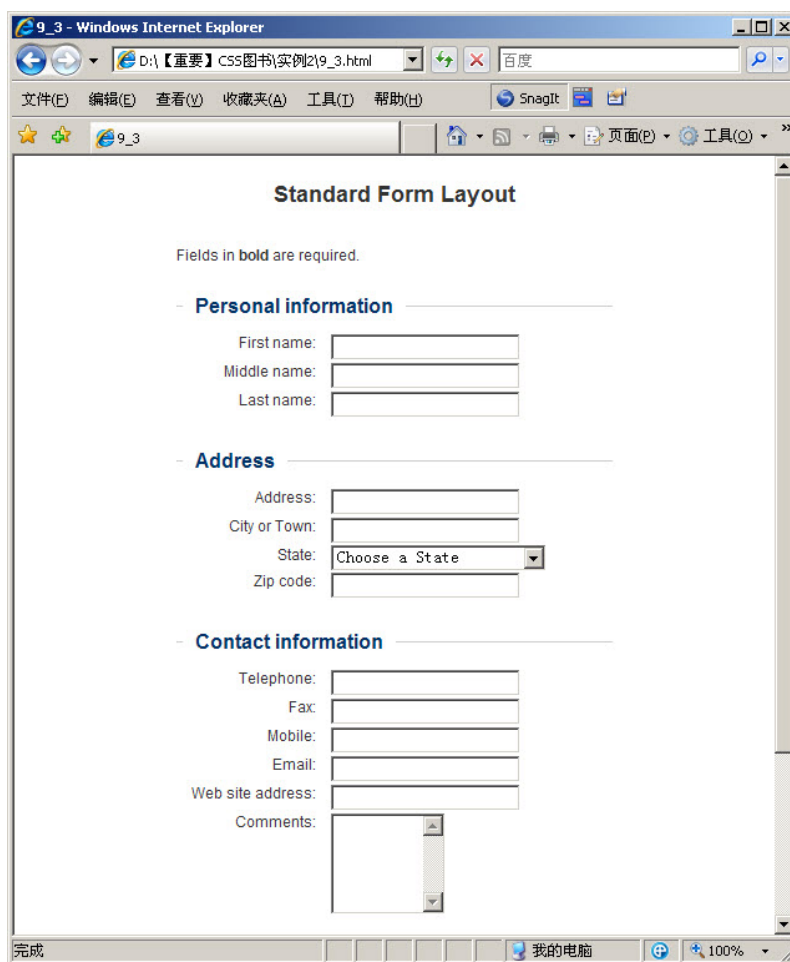


图 9-27 为 label 标签设置样式的预览效果

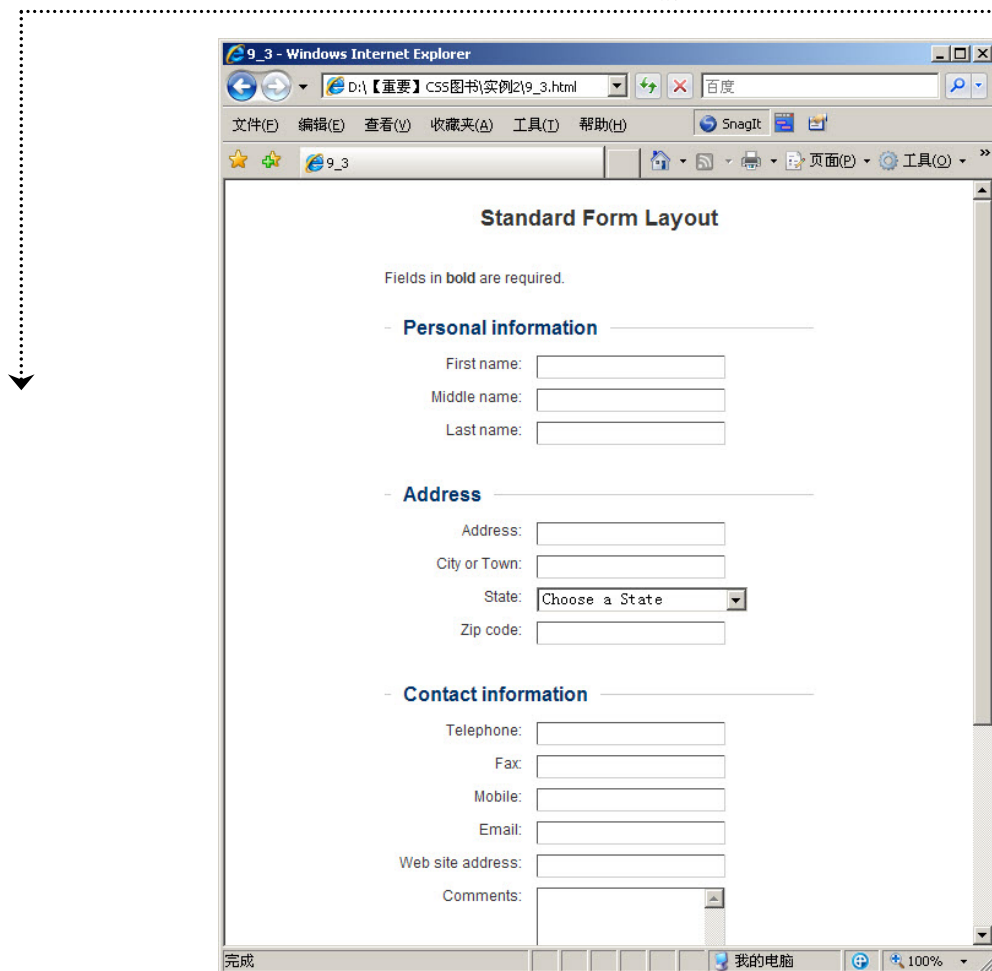


图 9-28 为输入文本框、文本域及下拉框添加样式的预览效果

(6) 制作单选按钮的 CSS 代码，代码如下。

```
div.fm-multi div {
    margin: 5px 0;                                /*设置外边距*/
}
div.fm-multi input {
    width: 1em;                                    /*设置宽度*/
}
div.fm-multi label {
    display: block;                                /*设置块级元素显示方式*/
    width: 200px;                                  /*设置宽度*/
    padding-left: 5em;                             /*设置左内边距*/
    text-align: left;                              /*设置文本左对齐*/
}
```

在以上代码中，使用组合选择符为单选按钮设置了 CSS 样式。在 `div.fm-multi label` 标签中设置了 `display: block` 样式，使每个单选按钮独立一行显示，这里面的选项分别是 “Yes” 和 “No”，预览效果如图 9-29 所示。

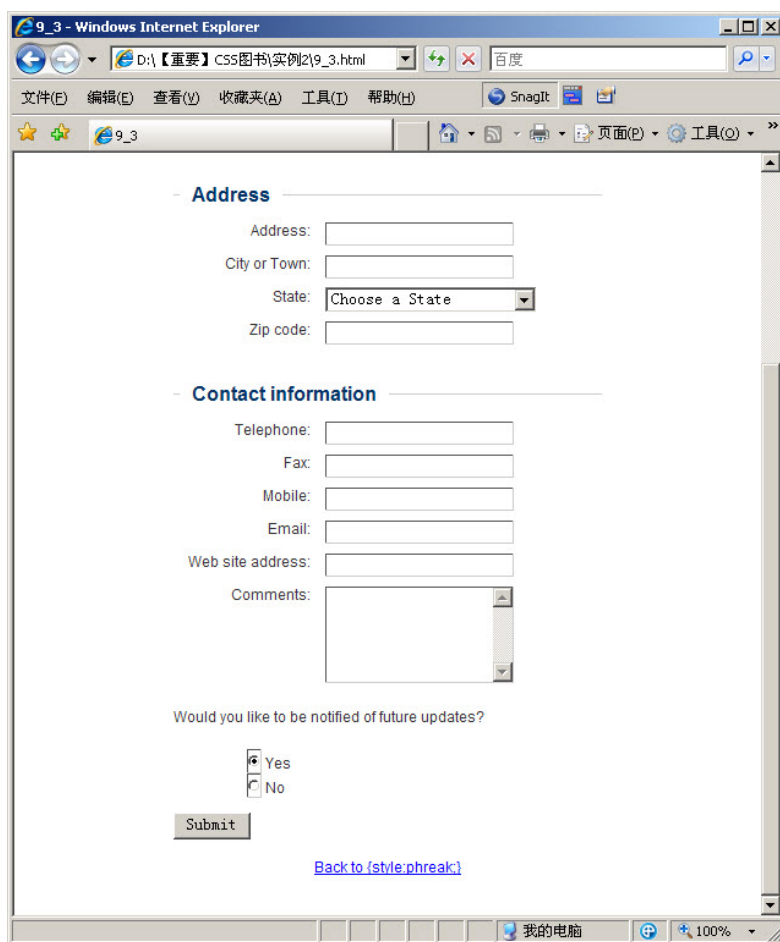


图 9-29 添加单选按钮样式的预览效果

(7) 制作“submit”按钮的 CSS 代码，代码如下。

```
#fm-submit {  
    clear: both;                                /*清楚左右浮动元素*/  
    padding-top: 1em;                            /*设置上内边距*/  
    text-align: center;                          /*设置文本居中对齐*/  
}  
#fm-submit input {  
    border: 1px solid #333;                      /*设置边框样式*/  
    padding: 2px 1em;                            /*设置内边距*/  
    background: #555;                            /*设置背景颜色*/  
    color: #fff;                                /*设置字体颜色*/  
}
```

在以上代码中设置了“submit”按钮的 CSS 样式。在#fm-submit 标签中，使用 clear:both 属性使按钮独立一行，禁止了两边的浮动元素出现。XHTML 按钮是有默认样式的，这里通过#fm-submit input 标签，重新设置了按钮的边框样式、内边距、背景色、字体颜色及字体大小，预览效果如图 9-30 所示。

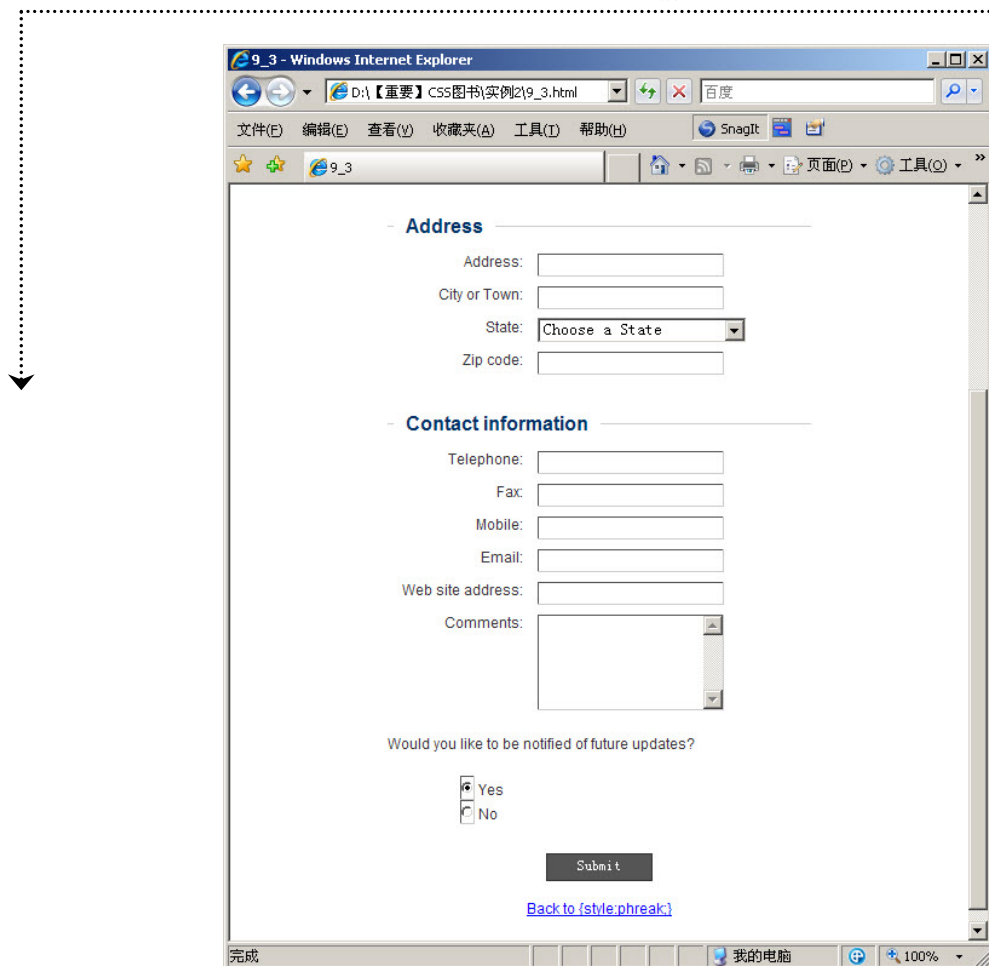


图 9-30 添加按钮样式的预览效果

(8) 补充一些相关 CSS 样式，代码如下。

```
input:focus, textarea:focus {
    background: #efefef;
    color: #000;
}
fieldset div.fm-req {
    font-weight: bold;
}
```

/*设置背景颜色*/
/*设置字体颜色*/

/*设置字体为粗体*/

在以上代码中，使用伪类:focus，在输入文本框和输入文本域获得焦点时向它们添加特殊的样式，即添加背景色和字体颜色样式。这种伪类在 IE 浏览器中是不显示的，在 Firefox 浏览器等标准浏览器中可以显示，如图 9-31 所示。

使用组合选择符 fieldset div.fm-req，为必填的内容添加醒目提示样式，即字体加粗显示。这样，整个表单页面就制作完成了，通过 IE 7、Firefox 及 IE 6 浏览器测试，没有不兼容问题，最终预览效果如图 9-32 所示。

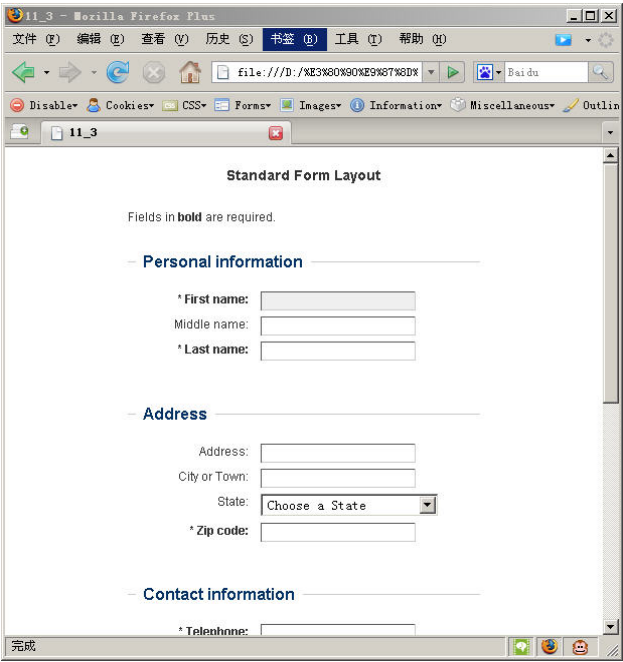


图 9-31 Firefox 浏览器中的获得焦点效果

Standard Form Layout

Fields in **bold** are required.

Personal information

First name:

Middle name:

Last name:

Address

Address:

City or Town:

State:

Zip code:

Contact information

Telephone:

Fax:

Mobile:

Email:

Web site address:

Comments:

Would you like to be notified of future updates?

☐ Yes

☐ No

[Back to Table of Contents](#)

图 9-32 符合 W3C 标准的表单预览效果

9.3.3 兼容问题

这个实例制作的是符合 W3C 标准的表单，因此，不应该再存在兼容问题。符合 W3C 标准的页面，顾名思义，就是使用能够兼容标准浏览器的 CSS 样式代码。

不过，由于至今为止，依然存在一些标签以及 CSS 在某些浏览器中无法显示的现象，这样的现象是无法通过兼容问题能够解决的，它是代码本身的缺陷。例如本实例的单选按钮，在 IE 浏览器中显示带有边框的样式。而在 Firefox 这样的标准浏览器中，却能够得到很好的显示，如图 9-33 所示。

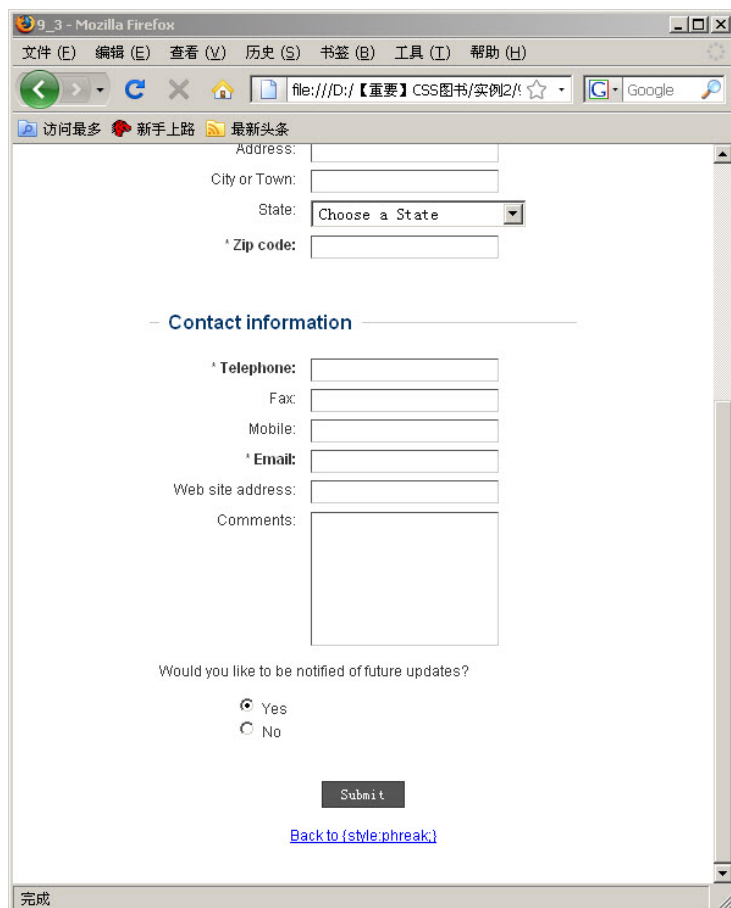


图 9-33 在 Firefox 浏览器中单选按钮得到完美显示

9.4 小结

本章讲解使用 CSS 制作表单的实例，包括了制作登录表单、制作用户注册的表单、制作符合 W3C 标准的表单，主要涉及到了 1 个 XHTML 标签 label。

对本章的知识点前后联系进行归纳总结，知识点结构导图如图 9-34 所示。



图 9-34 本章知识点结构导图