

在网页中,表单的作用非常重要,主要负责采集浏览者的输入信息,例如常见的注册表、登录表、调查表和留言表等。表单是网页交互的基本工具,在网页的制作过程中,常常需要使用表单实现用户与服务器之间的信息交互。

本章学习重点:

- 了解表单的组成
- 掌握表单中常用的控件
- 熟练使用表单元素布局表单

3.1 表单的基本结构



10min

HTML 表单的主要作用是接收用户的输入,从而能采集客户端信息,使网页具有交互功能。当用户提交表单时,浏览器将用户在表单中输入的数据打包,并发送给服务器,从而实现用户与 Web 服务器的交互,如图 3-1 所示。

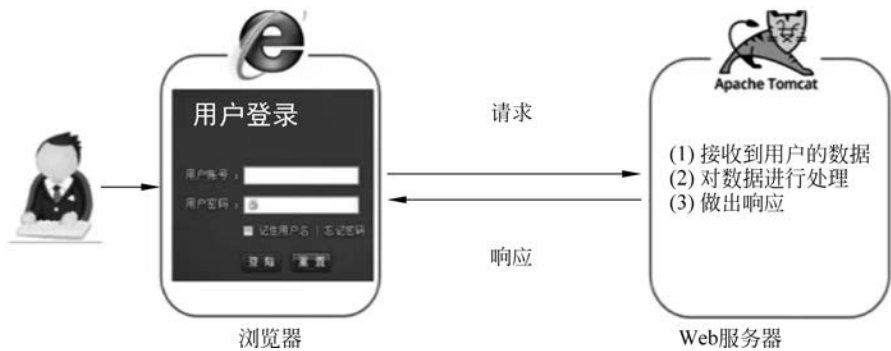


图 3-1 网页的交互性

在 HTML 中,一个完整的表单通常由表单标签(< form >)、表单域和表单按钮 3 部分构成,如图 3-2 所示。



图 3-2 表单的组成部分

(1) 表单标签(<form>): <form>标签用来创建表单,包含所有表单对象。<form>标签包含处理表单数据的各种属性,如表单的提交路径、提交方式等。

(2) 表单域(<input>、<select>等标签): 主要用来收集用户数据,包括文本框、密码框、隐藏域、单选框、复选框、下拉列表及文件上传等。

(3) 表单按钮: 包括提交按钮、重置按钮和一般按钮。提交按钮和一般按钮可用于把表单数据发送到服务器,重置按钮用于重置表单,把整个表单恢复到初始状态。

每个表单元素都有一个 name 属性,用于在提交表单时对表单数据进行识别。访问者通过提交按钮提交表单,表单提交后,填写的数据就会发送到服务器端进行处理。用户登录的表单如例 3-1 所示。

【例 3-1】 用户登录

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>用户登录</title>
</head>
<body>
  <form action="#" method="post">
    <p>用户名: <input type="text" name="username" /></p>
    <p>密码: <input type="password" name="password" /></p>
```

```
<input type = "submit" value = " 登 录 " />
</form>
</body>
</html>
```

在浏览器中的显示效果如图 3-3 所示。

在上述代码中,由 form 元素创建了一个表单,表单中包括用户名、密码输入型控件和登录按钮。



图 3-3 用户登录表单



3.2 表单标签< form >

在 HTML 中,< form >标签被用于定义表单,相当于一个容器,表示其他的表单标签需要在其范围内才有效,即在< form ></form >之间的一切都属于表单的内容。

创建表单的基本语法格式如下:

```
<form action = "URL 网址" method = "提交方式" name = "表单名称">
    //各种表单域
</form>
```

< form >标签包含很多属性,这里重点介绍 3 个常用基本属性: action、method 和 enctype。

(1) action: 在表单收集到信息后,需要将信息传递给服务器进行处理,action 属性用于指定接收并处理表单数据的服务器程序的 URL 网址。当设置 action = "#"时,表示提交给当前页面。

(2) method: 用于设置表单数据的提交方式,常用的两种方式为 get 和 post。

- get 方式传输的数据量少,执行效率高。当提交数据时,在浏览器网址栏中可以看到提交的数据,安全性不好。
- post 方式传输的数据量大,在浏览器网址栏中看不到提交的数据,适合传输重要信息,安全级别相对较高。在进行数据添加、删除等操作时可以设置为 post 方式。

(3) enctype: 用于规定表单数据发送时的编码方式,有以下 3 种属性值。

- text/plain: 主要用于向服务器传递大量文本数据,比较适用于电子邮件的应用。
- multipart/form-data: 上传二进制数据,只有使用了 multipart/form-data 才能完整地 对文件数据进行上传操作。
- application/x-www-form-urlencoded: 是其默认值。该属性主要用于处理少量文本 数据的传递。在向服务器发送大量的文件包含非 ASCII 字符的文本或二进制数据 时效率很低。

提示: 在实际网页开发中通常采用 post 方式提交表单数据。

3.3 表单控件

表单域和表单按钮都属于表单控件。

3.3.1 输入标签<input>

<input>标签为单标签,type 属性为最基本的属性,根据其属性值的不同可以显示多种表单元素样式,如单行文本框、密码框、单选和复选框等。除了 type 属性之外,<input>标签还有一系列属性用于表单输入控件的设置,如表 3-1 所示。

表 3-1 <input>标签的常用属性

属 性	属 性 值	描 述
type	text	单行文本输入框
	password	密码输入框
	radio	单选按钮
	checkbox	复选按钮
	submit	提交按钮
	reset	重置按钮
	button	普通按钮
	image	图像形式的提交按钮
	file	文件上传控件
	hidden	隐藏域
name	自定义名称	指定表单元素的名称。如果没有写 name 属性值,则表单组件的数据不能被正确提交
value	文本值	元素的初始值
size	数值	指定表单元素的初始宽度
maxlength	数值	指定输入框中输入的最大字符数
checked	checked	当 type 为 radio 或 checkbox 时,指定按钮是否被选中
disabled	disabled	指定加载时禁用此元素
readonly	readonly	将文本框内容指定为只读

<input>标签的常见语法格式如下:

```
<input type = "输入类型" name = "自定义名称"/>
```

1. 单行文本框

在<input>标签中,当 type 的属性值为 text 时表示单行文本框,这也是 type 属性的默认值。单行文本框也是使用最多的,例如登录时输入用户名、注册时输入电话号码、输入电子邮件、输入家庭住址等。

语法格式如下:



```
<input type="text" name="自定义名称">
```

单行文本的 name 属性值必须是唯一的,可以使用 size 属性将文本框可见字符的宽度为设置 20,使用 maxlength 属性设置最多允许输入 10 个字符。在默认情况下,单行文本框首次加载时内容为空,可以为其 value 属性预设初始文本,示例代码如下:

```
<input type="text" name="username" size="20" maxlength="10" value="admin">
```

2. 密码框

在<input>标签中,当 type 的属性值为 password 时表示密码输入框,此时输入的字符会被密码专用符号所遮挡,以保证文本的安全性。

语法格式如下:

```
<input type="password" name="自定义名称">
```

使用<input>标签中的 text 和 password 类型生成简易表单页面,如例 3-2 所示。

【例 3-2】 简易表单页面

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>简易表单页面</title>
</head>
<body>
  <form action="">
    姓名:<input type="text" name="username" size="20" maxlength="10"><br>
    <!-- readonly 表示只读状态,不可修改 -->
    身份:<input type="text" name="sf" readonly value="学生"><br>
    密码:<input type="password" name="pwd">
  </form>
</body>
</html>
```

在浏览器中的显示效果如图 3-4 所示。

3. 单选框

在<input>标签中,当 type 的属性值为 radio 时表示单选框,其样式为空心圆圈。

语法格式如下:

```
<input type="radio" name="自定义名称" value="值">
```



图 3-4 简易表单页面

(1) name: 单选框的名称,需要注意的是一组单选按钮组中,name 值必须相同,即单选按钮组同一时刻只能选择一个。

(2) value: 表单元素在提交数据时传递的数据值。value 必须指定一个值,因为单选框是通过用户选择的,而不是需要用户手动输入的值。

多个单选按钮可以组合在一起使用,为它们添加相同的 name 属性值即可表示这些单选按钮属于同一个组,示例代码如下:

```
<input type="radio" name="gender" value="woman"/> 女
<input type="radio" name="gender" value="man"/> 男
```

单选按钮可以使用 checked 属性设置默认选项。checked 属性的完整写法为 checked="checked",可简写为 checked。如果没有使用 checked 属性,则在首次加载时所有选项均处于未被选中状态。

```
<input type="radio" name="gender" value="woman"/> 女
<input type="radio" name="gender" value="man" checked/> 男
```

注意: 只能为单选按钮组中的一个选项使用 checked 属性。

4. 复选框

在<input>标签中,当 type 的属性值为 checkbox 时表示复选框,复选框也叫多选框。语法格式如下:

```
<input type="checkbox" name="自定义名称" value="值" >
```

多个 checkbox 类型的按钮可以组合在一起使用,添加相同的 name 属性值,表示这些复选框属于同一个组,即使在同一组内复选框也允许多选。value 属性值作为提交表单时传递的数据值,示例代码如下:

```
<input type="checkbox" name="hobby" value="zq"/> 足球
<input type="checkbox" name="hobby" value="ymq"/> 羽毛球
<input type="checkbox" name="hobby" value="lq"/> 篮球
<input type="checkbox" name="hobby" value="ppq"/> 乒乓球
```

复选框也可以使用 checked 属性设置默认被选中的选项,与单选框不同的是,它运行多个选项同时使用该属性,示例代码如下:

```
<input type="checkbox" name="car" value="jc">紧凑型
<input type="checkbox" name="car" value="zx" checked="checked">中型车
<input type="checkbox" name="car" value="suv" checked="checked"> SUV
<input type="checkbox" name="car" value="zd">主打车
```

使用单选框和复选框构建汽车之家注册页面,如例 3-3 所示。

【例 3-3】 汽车之家注册页面

```
<!DOCTYPE html >
<html lang="en">
```

```

< head >
  < meta charset = "UTF - 8">
  < title>汽车之家注册页面</title>
</head>
< body>
  < h3>注册后,可以使用汽车之家和二手车之家的相关服务</h3>
  < hr>
  < form action = "#">
    < p>手机:< input type = "text"></p>
    < p>密码:< input type = "password"></p>
    性别:< input type = "radio" checked name = "gender">男
        < input type = "radio" name = "gender">女
    < br>
    请输入你喜欢的车型:< input type = "checkbox" name = "car">紧凑型
        < input type = "checkbox" name = "car" checked = "checked">中型车
        < input type = "checkbox" name = "car" checked = "checked"> SUV
        < input type = "checkbox" name = "car">主打工
    < br>
    < br>
    < img src = "images/xy.png">
    < br>
    < input type = "button" value = "提交">
    < input type = "reset">

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

```

在浏览器中的显示效果如图 3-5 所示。



图 3-5 汽车之家注册页面

5. 提交按钮

在<input>标签中,当 type 的属性值为 submit 时表示提交按钮。用户在单击按钮时,会将当前表单中的所有数据整理成名称(name)和值(value)的形式进行参数传递,提交给服

务器处理。

语法格式如下：

```
<input type = "submit" value = "值" />
```

6. 重置按钮

在<input>标签中,当 type 的属性值为 reset 时表示重置按钮,用户单击后可以清除表单的内容。

语法格式如下：

```
<input type = "reset" value = "按钮的取值" />
```

注意：当按钮没有设置取值时,该按钮上默认显示的文字为“重置”。

7. 普通按钮

在<input>标签中,当 type 的属性值为 button 时表示普通按钮,value 值为按钮上显示的文本。

语法格式如下：

```
<input type = "button" value = "按钮的取值" />
```

普通按钮一般情况下要和 JavaScript 配合使用。例如,为按钮添加单击事件,当用户单击时跳转到百度页面,示例代码如下：

```
<input type = "button" value = "GO"
onclick = "window.open('http://www.baidu.com')">
```

8. 图片提交按钮

在<input>标签中,当 type 的属性值为 image 时表示图片按钮,其样式来源于自定义图片,使网页色彩更加丰富,如果使用默认的按钮形式,则往往会觉得单调。

语法格式如下：

```
<input type = "image" src = "URL 网址" alt = "替代文本"/>
```

表单按钮应用,如例 3-4 所示。

【例 3-4】 表单按钮应用

```
<!DOCTYPE html >
<html lang = "en">
<head>
  <meta charset = "UTF - 8">
  <title>表单按钮应用</title>
</head>
<body>
  <form action = "http://www.baidu.com" method = "get">
```



```
    姓名:< input type = "text" name = "username" value = "">< br >
    密码:< input type = "password" name = "pwd" >< br >
    < input type = "submit" value = "提交">
    < input type = "reset" value = "清空">
    < input type = "image" src = "images/btn.png" width = "130px" alt = "登录" />
</form>
</body>
</html>
```

在浏览器中的显示效果如图 3-6 所示。



图 3-6 表单按钮应用效果

当单击提交或图片按钮时会跳转到百度页面。输入的信息数据以“name 值 = value 值”(键-值对)形式提交于服务器中,如 username=admin。

注意：表单控件要想正确提交,必须设置 name 属性。用户在控件中输入的数据默认会赋值给 value 属性。

9. 文件上传

在< input>标签中,当 type 的属性值为 file 时表示文件上传,使用户可以选择一个或多个元素以提交表单的方式上传到服务器上。

语法格式下：

```
< input type = "file" name = "自定义名称" />
```

除了 < input> 标签共享的公共属性,file 类型的< input> 标签还支持下列属性,如表 3-2 所示。

表 3-2 file 类型的< input>标签支持的特有属性

属 性	描 述
accept	一个或多个期望的文件类型
capture	捕获图像或视频数据的源
files	FileList 列出了已选择的文件
multiple	布尔值,如果出现,则表示用户可以选择多个文件

file 类型的<input> 标签特有属性详解。

(1) accept: accept 属性是一个字符串,它定义了文件 input 应该接受的文件类型。示例代码如下:

```
<input accept = "audio/* ,video/* ,image/* ">
```

(2) capture: capture 属性指出了文件 input 是图片还是视频/音频类型,如果存在,则请求使用设备的媒体捕获设备(如摄像机),而不是请求一个文件输入。

(3) files: FileList 对象列出每个已选择的文件。如果 multiple 属性没有指定,则这个列表只有一个成员。

(4) multiple: 当将布尔类型属性指定为 multiple 时,文件 input 允许用户选择多个文件。用户可以用他们选择的平台允许的任何方式从文件选择器中选择多个文件。如果只想让用户为每个<input> 选择一个文件,则可省略 multiple 属性。

注意: 当表单中包含文件域时,form 元素的 method 属性必须设置为 post, enctype 属性必须设置为 multipart/form-data。

接下来看一个完整文件上传的示例,如例 3-5 所示。

【例 3-5】 多格式文件上传应用

```
<!DOCTYPE html >
<html lang = "en">
<head>
  <meta charset = "UTF - 8">
  <title>多格式文件上传应用</title>
</head>
<body>
  <form action = "#" method = "POST" enctype = "multipart/form - data">
    <!-- accept 表示期望的文件类型。格式:
    image/*
    .jpg, .png 或其他文件扩展名(后缀名)
    accept 不是强制的,用户可以通过在弹出窗上选择“所有文件”来选择任何文件
    添加 multiple 属性支持多文件上传 -->
    <h5>图片格式文件:</h5>
    <input type = "file" name = "file01" accept = ".jpg, .png" multiple>
    <h5>Word 格式文件:</h5>
    <input type = "file" name = "file02"
      accept = ".doc, application/msword" multiple>
    <h5>所有格式文件:</h5>
    <input type = "file" name = "file03" ><br>
    <input type = "submit" value = "上传">
  </form>
</body>
</html>
```

在浏览器中的显示效果如图 3-7 所示。



图 3-7 多格式文件上传应用效果

10. 隐藏域

在<input>标签中,当 type 的属性值为 hidden 时表示隐藏域。在页面中对用户不可见,它包含的信息也被提供给服务器。

语法格式如下:

```
<input type = "hidden" name = "自定义名称" value = "值">
```

在表单中插入隐藏域的目的在于收集或发送信息,以利于被处理表单的程序所使用。用户单击“发送”按钮发送表单时,隐藏域的信息也被一起发送到服务器。有时我们要给用户一些信息,让他在提交表单时确定用户的身份,如例 3-6 所示。

【例 3-6】 隐藏域的应用

```

<!DOCTYPE html >
<html lang = "en">
<head>
    <meta charset = "UTF - 8">
    <title>隐藏域的应用</title>
</head>
<body>
    <form action = "http://www.baidu.com" method = "get">
        用户名:<input type = "text" name = "username"><br>
        密 码:<input type = "password" name = "pwd"><br>
        <!-- 插入隐藏域,用户看不见 -->
        <input type = "hidden" name = "country" value = "china">
        <input type = "submit" value = "登录">
    </form>
</body>
</html>

```

在浏览器中的显示效果如图 3-8 所示。



(a) 首次加载完后输入信息



(b) 数据提交后的效果

图 3-8 隐藏域的应用效果



14min

3.3.2 标记标签

标记标签<label>为<input>标签定义标注,用于绑定一个表单元素,当单击<label>标签包含的文本时,被绑定的表单元素就会获得输入焦点,从而提升用户的使用体验。

在<label>标签中使用 for 属性指向目标标签的 id 值,示例代码如下:

```

<label for = "male">男</label>
<input type = "radio" name = "sex" id = "male" />

```

也可以直接将文本内容与表单控件都放入<label></label>标签之间,无须再为<input>标签特别设置 id 属性,示例代码如下:

```

<label><input type = "radio" name = "sex" value = "男"/>男</label>

```

这两种方法的运行效果完全相同,如例 3-7 所示。

【例 3-7】 标记标签的应用

```
<!DOCTYPE html >
<html lang = "en">
< head >
    < meta charset = "UTF - 8">
    < title>标记标签的应用</title>
</head>
< body >
    < h3> label 标签的两种用法 </h3>
    < label >< input type = "radio" name = "sex" value = "男"/>男</label>
    < hr/>
    < label for = "color"> 颜色</label>
        < input type = "checkbox" name = "蓝色" id = "color1">< label for = "color1">蓝色
</label>
        < input type = "checkbox" name = "绿色" id = "color2">< label for = "color2">绿色
</label>
        < input type = "checkbox" name = "紫色" id = "color3">< label for = "color3">紫色
</label>
</body>
</html>
```

在浏览器中的显示效果如图 3-9 所示。

3.3.3 多行文本标签

如果需要输入大量的信息,就需要用到< textarea > </textarea>文本域标签。通过< textarea >控件可以轻松地创建多行文本输入框,一般应用在留言板或网友评论等表单中,如图 3-10 所示。



15min

图 3-9 标记标签的应用效果



图 3-10 网友评论

语法规则如下:

```
< textarea cols = "每行中的字符数" rows = "显示的行数">
    文本内容
</textarea>
```

如例 3-8 所示,为客户提供留言输入框。

【例 3-8】 文本域的应用

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>文本域的应用</title>
</head>
<body>
  <form action="#">
    <table width="600" align="center">
      <tr>
        <td>客户留言方式一:</td>
        <td>客户留言方式二:</td>
      </tr>
      <tr>
        <td><textarea name="" cols="40" rows="6" placeholder="请输入内容:">
</textarea></td>
        <td><textarea name="" cols="40" rows="6" disabled>
《剑指大前端全栈工程师》书籍可以帮助读者快速提升段位,学前端这本就够了。实
力打造大前端时代,走在时代的前端。用通俗易懂的语言直指前端开发者的痛点。
</textarea></td>
      </tr>
    </table>
  </form>
</body>
</html>
```

在浏览器中的显示效果如图 3-11 所示。



图 3-11 文本域的应用效果

3.3.4 下拉列表标签

在 HTML 表单中,<select>标签用来创建单选列表或多选列表,<option>标签用于定义下拉列表中的一个选项。

<select>标签和<option>标签必须配合使用,<select>标签中至少应包含一对<option>标签。它们的常用属性如表 3-3 所示。

表 3-3 <select>标签和<option>标签的常用属性

标 签 名 称	属 性 名 称	属 性 值	说 明
select	disabled	disabled	禁用列表菜单
	multiple	multiple	规定可选择多个选项
	name	自定义名称	规定下拉列表的名称
	size	数值	规定下拉列表中可见选项的数目
option	disabled	disabled	首次加载时禁用当前选项
	selected	selected	规定首次加载时当前选项为选中状态
	value	文本内容	规定提交表单时发送给服务器的选项值

常见的用法是<select>标签配合若干<option>标签使用,示例代码如下:

```
<select name = "car" >
    <option value = "bmw">宝马</option>
    <option value = "benz">奔驰</option>
    <option value = "audi">奥迪</option>
</select>
```

其中,value 属性值是提交表单时传递给服务器的数据,不显示在网页上。<option>首尾标签之间的文本是显示在网页上的内容。

下拉列表首次加载默认选中的是第 1 项,如果需要默认选中列表中的其他项,则可以在<option>标签设置 selected 属性,示例代码如下:

```
<select name = "car" >
    <option value = "bmw">宝马</option>
    <option value = "benz" selected>奔驰</option>
    <option value = "audi">奥迪</option>
</select>
```

在<select>标签中设置 multiple 属性,允许在下拉列表中进行多选,示例代码如下:

```
<select name = "car" multiple = "multiple" >
    <option value = "bmw">宝马</option>
    <option value = "benz">奔驰</option>
    <option value = "audi">奥迪</option>
</select>
```

在不同的操作系统中,选择多个选项的方法如下。

- (1) 对于 Windows: 按住 Ctrl 按钮来选择多个选项。
- (2) 对于 macOS: 按住 Command 按钮来选择多个选项。

由于上述差异的存在,同时由于需要告知用户可以使用多项选择,对用户更友好的方式是使用复选框。

如果列表选项较多,则需要进行分类,此时可以使用<optgroup> 标签定义选项组,示例代码如下:

```

<select>
  <optgroup label = "北京">
    <option value = "1">东城区</option>
    <option value = "2">西城区</option>
    <option value = "3">海淀区</option>
  </optgroup>
  <optgroup label = "河北省">
    <option value = "4">石家庄</option>
    <option value = "5">保定市</option>
    <option value = "6">沧州市</option>
  </optgroup>
</select>

```

<optgroup>标签中 label 属性的文本内容规定了选项组的标题。

在很多地方能见到下拉列表的使用,最常用在填写地址及用户自己选择地址中,如例 3-9 所示。

【例 3-9】 下拉列表的应用

```

<!DOCTYPE html>
<html lang = "en">
<head>
  <meta charset = "UTF - 8">
  <title>下拉列表的应用</title>
</head>
<body>
  <p>单选下拉列表</p>
  <select>
    <option>--- 请选择 ---</option>
    <option value = "hn">湖南</option>
    <option value = "hb">湖北</option>
    <option value = "zj">浙江</option>
    <option value = "gd">广东</option>
  </select>
  <p>多选下拉列表</p>
  <select multiple>
    <option>--- 请选择 ---</option>
    <option value = "hn">湖南</option>
    <option value = "hb">湖北</option>
    <option value = "zj">浙江</option>
    <option value = "gd">广东</option>
  </select>
  <p>分组下拉列表</p>
  <select>
    <optgroup label = "北京">
      <option value = "1">东城区</option>
      <option value = "2">西城区</option>
      <option value = "3">海淀区</option>
    </optgroup>
  </select>

```



```
</optgroup>
<optgroup label = "河北省">
    <option value = "4">石家庄</option>
    <option value = "5">保定市</option>
    <option value = "6">沧州市</option>
</optgroup>
</select>
</body>
</html>
```

在浏览器中的显示效果如图 3-12 所示。



图 3-12 下拉列表应用效果

3.3.5 按钮标签

按钮标签< button >可用于在网页上生成自定义样式的按钮。在< button ></button >之间可以包含普通文本、图像、文本格式化标签等内容,这意味着使用< button >可以创建图像、颜色、文字等更多样式效果的按钮,比< input >标签创建的按钮更加丰富。

< button >标签常用属性如表 3-4 所示。

表 3-4 < button >标签常用属性

属 性 名 称	属 性 值	说 明
autofocus	autofocus	在页面内容加载时,按钮是否获得焦点
disabled	disabled	禁止使用按钮
name	自定义名称	给按钮添加名字
type	button/reset/submit	指定按钮的类型
value	文本内容	指定按钮的初始值

注意：应始终为 < button > 元素规定 type 属性。不同的浏览器对 < button > 元素的 type 属性使用不同的默认值。建议尽量使用 < input > 在 HTML 表单中创建按钮。

button 按钮根据 type 属性的不同值有不同的用法,这里介绍 3 种常见用法。

1. button 的普通用法

将 button 的 type 属性设置为 button 是 button 的一般性用法,使用时,一般需要搭配 onclick 属性,示例代码如下:

```
<button type = 'button'>普通按钮</button><br><br>
<button type = 'button' onclick = 'alert("你好,世界")'>普通按钮,带 onclick 属性</button>
```

如上代码,当单击不带 onclick 的普通按钮时,按钮不执行任何操作,当单击带 onclick 属性值的按钮时,程序会执行 onclick 中设置的操作。

2. button 的提交用法

HTML 表单的提交可以通过将 input 标签的 type 属性值设置为 submit 实现,也可以用 button 设置,但也要将 button 的 type 属性值设置为 submit,示例代码如下:

```
<form method = 'GET' action = 'https://blog.csdn.net/beixishuo'>
  <input type = 'text' name = 'name' value = "贝西奇谈">
  <button type = 'submit'>搜索</button>
</form>
```

3. button 的表单重置 reset 用法

在有些网站的注册页面或登录页面,可能会设置一个一键清除内容或重置内容的按钮,这项功能可以通过将 button 中 style 值设置为 reset 实现,示例代码如下:

```
<form>
  <span>用户:</span><input type = 'text'><br><br>
  <span>密码:</span><input type = 'password'><br><br>
  <button type = 'reset'>重置</button>
</form>
```

3.3.6 域标签

域标签 < fieldset > 可将表单内的相关元素分组,通常和 < legend > 标签一起使用,< legend > 标签用于定义每组的标题。

基本语法格式如下:

```
<fieldset>
  <legend>健康信息</legend>
  <!-- 其他表单控件 -->
</fieldset>
```

在 HTML 表单中可以使用 < fieldset > 标签对表单元素进行分项分组,如例 3-10 所示。

【例 3-10】 域标签的应用

```

<!DOCTYPE html >
<html lang = "en">
<head>
    <meta charset = "UTF - 8">
    <title>域标签的应用</title>
</head>
<body>
    <h3>域标签 fieldset 和域标题 legend 的简单应用</h3>
    <hr>
    <form action = "url" method = "post">
        <fieldset>
            <legend>账户信息:</legend>
            用户名:<input type = "text" name = "username" /><br />
            密码:<input type = "text" name = "pwd" /><br />
        </fieldset>
        <br />
        <fieldset>
            <legend>个人信息:</legend>
            姓名:<input type = "text" name = "name" /><br />
            邮箱:<input type = "text" name = "email" /><br />
            职位:<input type = "text" name = "postion" />
        </fieldset>
        <p align = "center"><input type = "submit" value = "提交"></p>
    </form>
</body>
</html>

```

在浏览器中的显示效果如图 3-13 所示。



图 3-13 域标签的应用效果

注意：表单控件要想正确提交，必须设置 name 属性。



15min



10min



16min

3.4 综合案例

人人网是中国领先的实名制社交网络平台。人人网在用户数、页面浏览量、访问次数和用户花费时长等方面均占据优势地位。可以仿人人网官方注册页面实现其功能,如例 3-11 所示。

【例 3-11】 人人网注册页面

```
<!DOCTYPE html >
<html lang = "en">
<head>
    <meta charset = "UTF - 8">
    <meta http - equiv = "X - UA - Compatible" content = "IE = edge">
    <meta name = "viewport" content = "width = device - width, initial - scale = 1.0">
    <title>人人网注册页面</title>
</head>
<body>
    <table width = "895" border = "0" align = "center" >
        <tr>
            <td width = "120"><img src = "images/logo.gif" width = "106" height = "27" /></td>
            <td><strong>10 秒找到你的所有朋友</strong></td>
            <td align = "right">已有人人网账号,<a href = "#">登录</a></td>
        </tr>
    </table>
    <br />
    <table width = "895" border = "1" align = "center" cellpadding = "10" cellspacing = "0">
        <tr>
            <td>
                <p><strong>免费开通人人网账号</strong></p>
                <form id = "form1" name = "form1" method = "post" action = "">
                    <table width = "100%" border = "0" cellspacing = "0" cellpadding = "10">
                        <tr>
                            <td width = "100" align = "right">注册邮箱:</td>
                            <td>
                                <input name = "userName" required type = "text" id = "userName"
                                size = "30" maxlength = "50" />
                            </td>
                        </tr>
                        <tr>
                            <td align = "right">&nbsp;</td>
                            <td>还可以使用 <a href = "#">账号</a> 注册或者 <a href = "#">手机号</a> 注册</td>
                        </tr>
                        <tr>
                            <td align = "right">创建密码:</td>
                            <td><input name = "passWord" type = "password" id = "passWord" size =
                                "30" maxlength = "16" /></td>
                        </tr>
                    </table>
                </form>
            </td>
        </tr>
    </table>
```

```
< tr>  
    < td align = "right">真实姓名:</td>  
    < td>< input name = "trueName" type = "text" id = "trueName" value = ""  
size = "30" maxlength= "8" /></td>  
</tr>  
< tr>  
    < td align = "right">性别:</td>  
    < td>< input name = "sex" type = "radio" id = "male" value = "1"  
checked = "checked" />男  
  
                                < input type = "radio" name = "sex" id = "female" value = "2" />  
女</td>  
</tr>  
< tr>  
    < td align = "right">生日:</td>  
    < td>< select name = "birthYear" id = "birthYear">  
        < option value = "1990"> 1990 </option>  
        < option value = "1991" selected = "selected"> 1991 </option>  
        < option value = "1992"> 1992 </option>  
</select>年 < select name = "birthMonth" id = "birthMonth">  
        < option value = "12"> 12 </option>  
        < option value = "11"> 11 </option>  
        < option value = "10" selected = "selected"> 10 </option>  
</select>月 < select name = "birthDay" id = "birthDay">  
        < option value = "31"> 31 </option>  
        < option value = "30" selected = "selected"> 30 </option>  
        < option value = "29"> 29 </option>  
</select>日</td>  
</tr>  
< tr>  
    < td align = "right">我现在:</td>  
    < td>< select name = "work" id = "work">  
        < option value = "正在上学">正在上学</option>  
        < option value = "已毕业">已毕业</option>  
</select></td>  
</tr>  
< tr>  
    < td align = "right"> &nbsp;&nbsp;&nbsp;</td>  
    < td>< img src = "images/verycode.gif" width = "132" height = "55"  
align = "absmiddle" /> &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&~<a href = "#">看不清,换一张?</a></td>  
</tr>  
< tr>  
    < td align = "right">验证码:</td>  
    < td>< input name = "veryCode" type = "text" id = "veryCode" size =  
"15" maxlength = "5" /></td>  
</tr>  
< tr>  
    < td align = "right"> &nbsp;&nbsp;&nbsp;</td>  
    < td>< input type = "image" name = "btnReg" id = "btnReg" src =  
"images/btn_reg.gif" /></td>  
</tr>  
</table>
```

```

        </form>
        </td>
        <td width = "284" valign = "top" bgcolor = "# f7f7f7"><table width = "100 %"
border = "0" cellspacing = "0" cellpadding = "0">
            <tr>
                <td height = "200" align = "center"><img src = "images/intro - new.
png" width = "242" height = "116" /></td>
            </tr>
            <tr>
                <td height = "25"><strong>最热门公共主页</strong></td>
            </tr>
            <tr>
                <td><table width = "100 %" border = "0" cellspacing = "0"
cellpadding = "0">
                    <tr>
                        <td width = "94" align = "center"><img src = "images/icon_
zhaowei.gif" width = "73" height = "73" /></td>
                        <td width = "94" align = "center"><img src = "images/icon_
likaifu.gif" width = "73" height = "73" /></td>
                        <td align = "center"><img src = "images/icon_shangjie.gif"
width = "73" height = "73" /></td>
                    </tr>
                    <tr>
                        <td height = "25" align = "center"></td>
                        <td align = "center">李开复</td>
                        <td align = "center">商界</td>
                    </tr>
                </table></td>
            </tr>
            <tr>
                <td>&nbsp;</td>
            </tr>
            <tr>
                <td height = "25"><strong>最热门游戏</strong></td>
            </tr>
            <tr>
                <td><table width = "100 %" border = "0" cellspacing = "0"
cellpadding = "0">
                    <tr>
                        <td width = "94" align = "center"><img src = "images/icon_
xxzz.gif" width = "73" height = "73" /></td>
                        <td width = "94" align = "center"><img src = "images/icon_
rrnc.gif" width = "73" height = "73" /></td>
                        <td>&nbsp;</td>
                    </tr>
                    <tr>
                        <td height = "25" align = "center">小小战争</td>
                        <td align = "center">人人农场</td>
                        <td>&nbsp;</td>
                    </tr>
                </table></td>
            </tr>
        </td>
    </tr>
</table>

```

```

        </tr>
      </table></td>
    </tr>
  </table>
  <br />
  <table width = "895" border = "0" align = "center" cellpadding = "0" cellspacing = "0">
    <tr>
      <td align = "right">单击免费开通账号表示同意并遵守<a href = "#">人人服务条款
    </a></td>
  </tr>
</table>
</body>
</html>
```

在浏览器中的显示效果如图 3-14 所示。



图 3-14 人人网注册页面显示效果图

注意：非空验证可以通过 HTML 中的 required 属性实现。