

# 表单

本讲大纲：

- 1、表单标记-form
- 2、form标记属性
- 3、输入类控件
- 4、菜单类控件
- 5、表单的应用
- 6、表单数据中文乱码问题
- 7、html5新增的输入类型
- 8、fieldset、legend标记

# 使用表单标记——form

## 概述

表单在网页中主要负责数据采集功能。

一个表单有三个基本组成部分：

- **表单标签：**

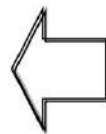
这里面包含了处理表单数据所用CGI程序的URL以及数据提交到服务器的方法。

- **表单域（表单控件）：**

包含了文本框、密码框、隐藏域、多行文本框、复选框、单选框、下拉选择框和文件上传框等。

- **表单按钮：**

包括提交按钮、复位按钮和一般按钮；用于将数据传送到服务器上的CGI脚本或者取消输入，还可以用表单按钮来控制其他定义了处理脚本的处理工作。

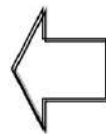


# 表单标签

在表单的<form>标记中，还可以设置表单的基本属性，包含：

- 表单的名称
- 处理程序
- 传送方法

一般情况下，表单的处理程序**action**和传送方法**method**是必不可少的参数。

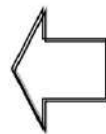


# <form>元素的属性

在HTML中，<form></form>标记对用来创建一个表单

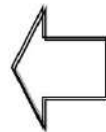
- **action**属性：用来定义表单处理程序的位置(相对地址或绝对地址)。
- **Name**属性：用来给表单命名。
- **Method**属性：定义表单结果从浏览器传送到服务器的方法，一般有两种方法：**get**、**post**。
- **enctype** 属性：用于设置表单信息提交的编码方式。

**enctype**属性为表单定义了**MIME**编码方式，编码方式的取值如下表所示



# MIME编码方式的取值

编码方式的取值	取值的含义
<b>text/plain</b>	以纯文本的形式传送
<b>application/x-www-form-urlencoded</b>	默认的编码形式
<b>multipart/form-data</b>	<b>MIME</b> 编码，上传文件的表单必须选择该项

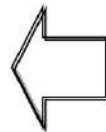


# <form>元素的属性

- **target属性**：用来指定目标窗口的打开方式。

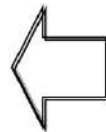
目标窗口的打开方式包含4个取值

- \_blank是指将返回的信息显示在新打开的窗口中；
- \_parent是指将返回信息显示在父级的浏览器窗口中；
- \_self则表示将返回信息显示在当前浏览器窗口；
- \_top表示将返回信息显示在顶级浏览器窗口中。



# Method属性中get方法与post方法的区别

- **method=get:** 使用这个设置时，表单数据会被视为**CGI**或**ASP**的参数发送，也就是来访者输入的数据会附加在**URL**之后，由用户端直接发送至服务器，所以速度上会比**post**快，但缺点是数据长度不能够太长。在没有指定**method**的情形下，一般都会视**get**为默认值。
- **method=post:** 使用这种设置时，表单数据是与**URL**分开发送的，用户端的计算机通知服务器来读取数据，所以通常没有数据长度上的限制，缺点是速度上会比**get**慢。



# 控件的添加

## 概述

按照[控件的填写方式](#)可以分为输入类和菜单列表类。

- 输入类的控件一般以标记开始，说明这一控件需要用户的输入
- 菜单列表类则以select开始，表示用户需要选择。

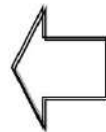
按照[控件的表现形式](#)则可以将控件分为文本类、选项按钮、菜单等几种。

在HTML表单中，input参数是最常用的控件标记，包括最常见的文本域、按钮都是采用这个标记，这个标记的基本语法是：

<form>

    <input name="控件名称" type="控件类型" />

</form>





# 输入类控件

<input name=? type=\* value=\*\* size=\*\*\* maxlength=??>

? → 文本框的名字

\* → text,password

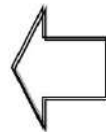
\*\* → 默认值

\*\*\* → 长度

?? → 最大输入字符数



1.html、2.html



# 单选按钮

单选框能够进行项目的单项选择，以一个圆框表示。

## 语法

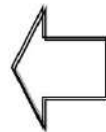
```
<input type="radio" value="单选按钮的取值" name="单选按钮名称" checked="checked"/>
```

## 语法解释

在该语法中，checked属性表示这一单选按钮默认被选中，而在一个单选按钮组中只能有一项单选按钮控件设置为checked。Value则用来设置用户选中该项目后，传送到处理程序中的值。



3.html



# 复选框

复选框在页面中以一个方框来表示。

## 语法

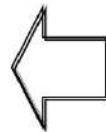
```
<input type="checkbox" value="复选框的值" name="名称"  
checked="checked" />
```

## 语法解释

在该语法中，checked参数表示该选项在默认情况下已经被选中，一个选择中可以有多多个复选框被选中。



8.html



# 图像域

## 概述

图像域是指可以用在提交按钮位置上的图片，这幅图片具有按钮的功能。

## 语法

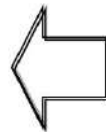
```
<input type="image" src="图地址" name="图像域名称" />
```

## 语法解释

在该语法中，图像地址可以是绝对地址或相对地址。



8.html



# 文件域

## 概述

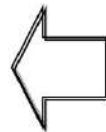
文件域在上传文件时常常用到，它用于查找硬盘中的文件路径，然后通过表单将选中的文件上传。在设置电子邮件的邮件、上传头像、发送文件时常常会看到这一控件。

## 语法

```
<input type="file" name="文件域的名称" />
```



8.html



# 列表/菜单标记

## 概述

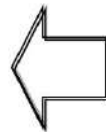
菜单列表类的控件主要用来进行选择给定答案中的一种，这类选择往往答案比较多，使用单选按钮比较浪费空间。可以说，菜单列表类的控件主要是为了节省页面空间而设计的。菜单和列表都是通过<select>和<option>标记来实现的。

## 语法

```
<select name="下拉菜单的名称">  
    <option value="" selected="selected">选项显示内容</option>  
    <option value="选项值">选项显示内容</option>  
    .....  
</select>
```



9.html



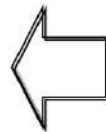
# 文本域标记textarea

## 概述

它与文字字段的区别在于可以添加多行文字，从而可以输入更多的文本。这类控件在一些留言本中最为常见。

## 语法

```
<textarea name="文本域名称" value="文本域默认值"  
rows="行数" cols="列数"  
>/textarea>
```

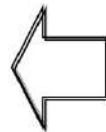


# 文本域标记textarea

文字域标记属性	描述
<b>Name</b>	文字域的名称
<b>Rows</b>	文字域的行数
<b>Cols</b>	文字域的列表
<b>Value</b>	文字域的默认值



10.html





# 表单按钮

## 提交按钮

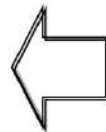
`<input type="submit" name="Submit" value="提交">`获得一个提交按钮

## 重置按钮

`<input type="reset" name="Reset" value="重写">`获得一个重置按钮

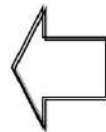


5.Html  
10.html



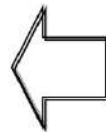
# 表单的应用

- 1、设计表单是为了让用户输入信息，网页最终需要将用户输入的信息转发到其他软件或服务器上。（看懂form标记中的action属性非常重要）
- 2、用于输入表单的网页既可以是静态网页，也可以是动态网页。
- 3、接收和处理表单数据的网页一般是动态网页（使用动态网页必须先按照web服务器，例如tomcat和IIS）



# 表单的应用

- 1、设计表单是为了让用户输入信息，网页最终需要将用户输入的信息转发到其他软件或服务器上。（看懂form标记中的action属性非常重要）
- 2、用于输入表单的网页既可以是静态网页，也可以是动态网页。
- 3、接收和处理表单数据的网页一般是动态网页（使用动态网页必须先按照web服务器，例如tomcat和IIS）



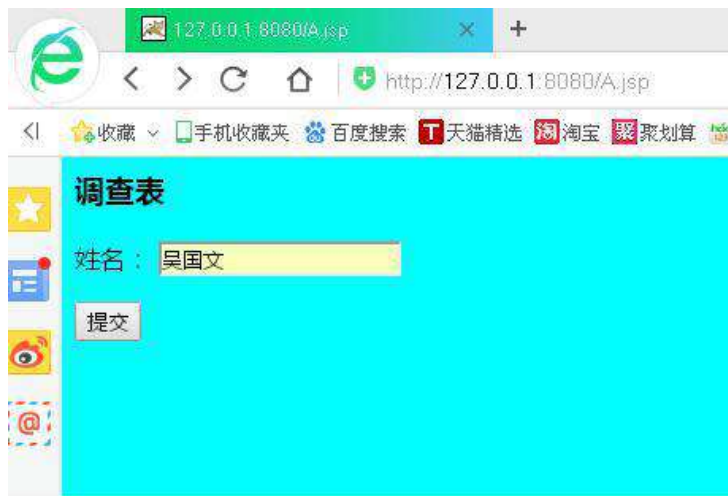
# 表单的应用

```
yle.css x A.jsp x
<%@ page contentType="text/html;Charset=utf-8" pageEncoding="utf-8"%>
<HTML>
<BODY bgcolor=cyan>
<h3>调查表</h3>
<FORM action="B.jsp" method=post name=form>
  姓名:
  <INPUT type="text" name="boy">
  <p>
  <INPUT TYPE="submit" value="提交" name="submit">
</FORM>
</BODY>
</HTML>
```

这行是jsp代码

- 1、A.Jsp网页中包含表单，
- 2、表单中包含一个名为“boy”的文本框
- 3、当点击“提交”按钮后，“boy”文本框中的文字被发送到“B.jsp”网页中

A.Jsp运行后效果图



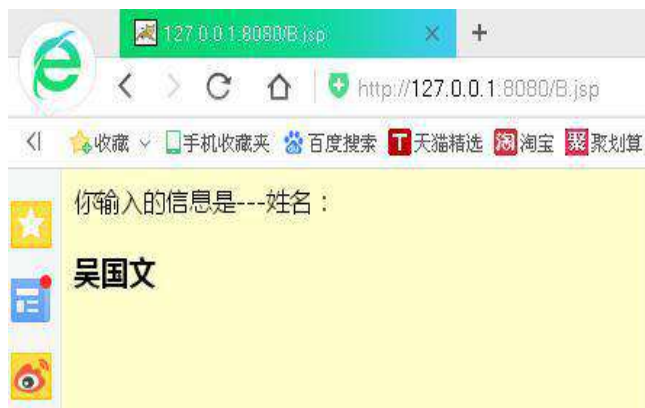
# 表单的应用



```
1 <%@ page contentType="text/html; charset=utf-8" %>
2 <MHTML><BODY bgcolor="#FFFFCC">
3 <P>你输入的信息是---姓名:
4 <% String textContent=request.getParameter("boy");
5     byte b[]=textContent.getBytes("ISO-8859-1");
6     textContent=new String(b,"utf-8");
7 %>
8 <h3> <%=textContent%> </h3>
9 </BODY></HTML>
```

在B.jsp中接收A.jsp表单文本框“boy”的值

在A.jsp表单中点击  
“提交”按钮后，网  
页自动跳转到B.jsp，  
并显示右图所示画面

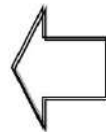


# 表单中文数据乱码

---

## 例10.html

```
<body>
用户调查留言：<br /><br />
<form action="mailto:04ljy@163.com" name="invest" method="post"
enctype="text/plain" accept-charset="utf-8">
用户名：<input name="username" type="text" size="20" id="username"
/><br />
密码：<input name="passworg" type="password" size="20" /><br /><br
/>
留言：<textarea name="liuyan" rows="5" cols="40">
    </textarea><br /><br />
    <input type="submit" name="submit" value="提交" />
    <input type="reset" name="Reset" value="重写">
</form>
</body>
```

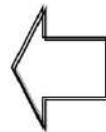




# html5新增的输入类型

- email类型
- url类型
- number类型
- 日期检出器类型
- color类型

HTML5新增的功能不是所有的浏览器都支持，对HTML5支持比较好的浏览器依次为版本比较高的IE浏览器，firefox、chrome浏览器





# email类型

- email类型的input元素是一种专门用于输入E-mail地址的文本输入框，在提交表单的时候，会自动验证email输入框的值。如果不是一个有效的E-mail地址，则该输入框不允许提交该表单。

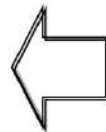
`<input type="email" name="email_address"/>`

- 表单会增加一个验证是否为邮箱地址的功能



示例

12.Html



# url类型

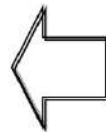
- url类型的input元素提供用于输入url地址这类特殊文本的文本框。当提交表单时，如果所输入的内容是url地址格式的文本，则会提交数据到服务器，如果不是url地址格式的文本，则不允许提交。

`<input type="url" name="url_info"/>`

- 表单增加一个验证是否为url地址的功能



12.Html



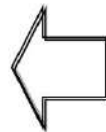
# number类型

- **number**类型的元素提供用于输入数值的文本框。实际使用中，可以设定对所接受的数字进行限制，例如，规定允许的最大值和最小值、合法的数字间隔或默认值等。如果所输入的数字不符合限制要求，则会出现错误提示。

```
<input type="number" name="content" min="1" max="100" step="5"/>
```



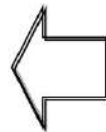
12.Html



# 日期检出器类型

- 日期检出器(Date Pickers)是网页中经常要用到的一种控件。HTML5提供了多个可用于选取日期和时间的输入类型，即6种日期检出器控件，分别用于选择以下日期格式：日期、月、星期、时间、日期+时间和日期+时间+时区。

输入类型	HTML代码	功能说明
date	<code>&lt;input type="date"&gt;</code>	选取日、月、年
month	<code>&lt;input type="month"&gt;</code>	选取月、年
week	<code>&lt;input type="week"&gt;</code>	选取周和年
time	<code>&lt;input type="time"&gt;</code>	选取时间，包括小时和分钟
datetime	<code>&lt;input type="datetime"&gt;</code>	选取时间、日、月、年（UTC时间）
datetime-local	<code>&lt;input type="datetime-local"&gt;</code>	选取时间、日、月、年（本地时间）



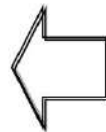
# color类型

- color类型的input元素提供专门用于设置颜色的文本框。通过单击文本框，可以快速打开调色器面板，方便用户可视化选择一种颜色。

`<input type="color" name="content"/>`



12.Html



# fieldset标记

- **fieldset**标记

浏览器会对用<fieldset>分组的表单外外加上一个边框

- **legend**标记

legend标记为fieldset分组提供文本描述。



```
<fieldset>  
<legend>联系方式</legend>  
<p>输入email:  
<input type="email" name="email1">  
<p>输入url地址:  
<input type="url" name="url1">  
</fieldset>
```

联系方式

输入email :

输入url地址 :

