



# HTML5 新增标签和属性

在网站开发过程中,表单是获取用户信息的重要手段,HTML5 大大加强了表单的功能,增加了从用户手机获得特定类型数据的新方法和在浏览器中检查数据的能力,以便提高用户交互的黏性,而且 HTML5 为了使页面元素更加丰富、多元,新增了 video 和 audio 等媒体标签。本章主要介绍表单新增特性、语义化结构标签及媒体标签的应用。

本章思维导图如图 3-1 所示。

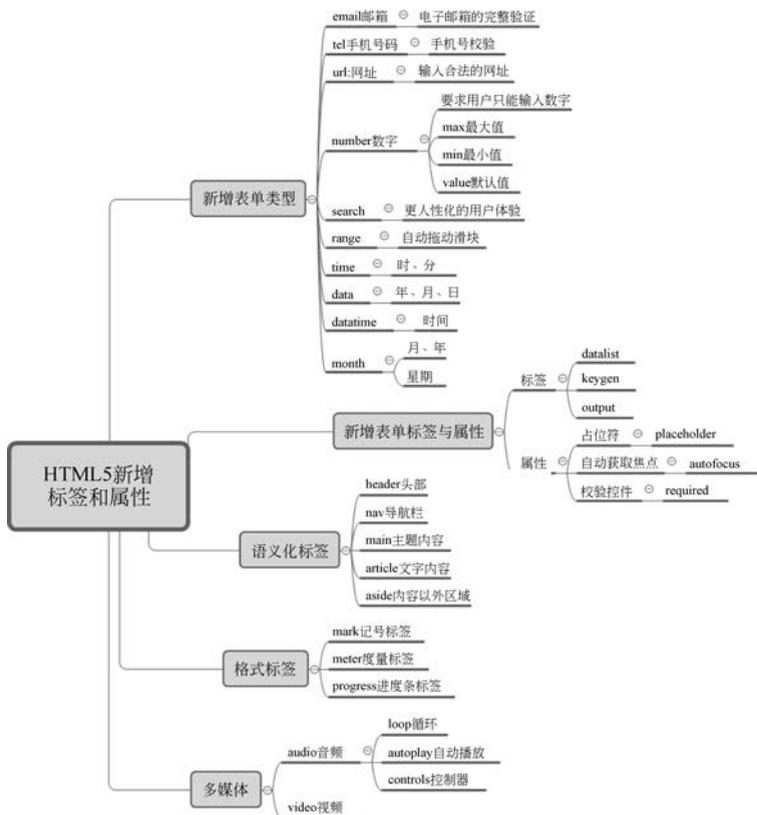


图 3-1 思维导图

## 3.1 HTML5 表单新增

HTML5 对表单系统进行了彻底的改造,新特性提供了更多语义明确的表单类型,并能够及时响应用户交互,以适应当前的应用。下面从表单新增类型、新增元素及新增属性等方面分别进行介绍。

### 3.1.1 HTML5 新增输入类型

HTML5 新增了多项表单输入类型,这些新类型具有更明确的含义,提供了更好的输入控制和验证,而不用借助其他前端脚本语言(如 JavaScript),为开发人员带来了极大的方便。新增的输入类型如表 3-1 所示。

表 3-1 HTML5 新增输入类型

类 型	示 例	描 述
email	<input type="email" />	输入邮箱格式
tel	<input type="tel" />	输入手机号码格式
url	<input type="url" />	输入 URL 格式
number	<input type="number" />	输入数字格式(只能是数字)
search	<input type="search" />	搜索框(体现语义化)
range	<input type="range" />	包含数值范围的滚动条
time	<input type="time" />	选中时间(包含时、分)
date	<input type="date" />	选择日期(包含年、月、日)
datetime	<input type="datetime" />	UTC 时间(包含年、月、日、时、分)
month	<input type="month" />	选择月份(包含年、月)
week	<input type="week" />	选择星期(包含年、第几周)
datetime-local	<input type="datetime-local">	本地时间和日期
color	<input type="color" />	颜色选择器

其中,datetime、datetime-local、time、date、week 和 month 类型是 6 种样式不同的时间日期选择器控件,统称为日期选择器。目前主流浏览器一般支持新的 input 类型,即使不支持,也可以显示常规的文本域。

#### 1. 电子邮箱类型 email

在提交表单时,会自动验证 email 域的值是否符合 email 的标准格式,再也不用自己用正则表达式去写 email 的格式验证了,代码如下:

```
Email: <input type="email" name="useremail" />
```

#### 2. 电话号码类型 tel

tel 类型的元素用于让用户输入和编辑电话号码,在提交表单之前,输入值不会被自动

验证为特定格式,因为世界各地的电话号码格式差别很大。尽管 tel 类型的输入在功能上和 text 输入一致,但它们确实有用,其中最明显的就是移动浏览器,特别是在手机上可能会选择专为输入电话号码而优化的自定义键盘,如图 3-2 所示。



图 3-2 手机端运行效果

由图 3-2 可知 `type="tel"` 会唤起系统的数字键盘。或者 tel 类型的文本框也可以和 `pattern` 属性一起使用,达到校验的效果,代码如下:

```
<input type="tel" id="phone" name="phone" pattern="[0-9]{10}">
```

### 3. 地址类型 url

将 `type` 属性设置为 `url`,在提交表单时,会自动验证 `url` 域的值是否符合 `url` 的标准格式,输入的内容中必须包含 `http://`,后面必须有内容,如百度网址或谷歌网址,代码如下:

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

### 4. 数字类型 number

`number` 类型是用来专门输入数字的文本框,在提交时会检测其中内容是否为数字。此类型的 `input` 标签的常用属性如表 3-2 所示,代码如下:

```
<input type="number" name="user_num" min="1" max="10" />
```

表 3-2 数值设置限定属性

属 性	描 述
max	规定允许的最大值
min	规定允许的最小值
step	规定合法的数字间隔(如果 step="3",则合法的数字是-3、0、3、6 等)
value	规定默认值
disabled	规定输入字段是禁用的
maxlength	规定输入字段的最大字符长度
pattern	规定用于验证输入字段的模式
readonly	规定输入字段的值无法修改
required	规定输入字段的值是必需的
size	规定输入字段的可见字符

### 5. 搜索类型 search

search 类型是一种专门用来输入搜索关键词的文本框。不同于普通类型的文本框,当用户开始输入时,输入框的右边会出现一个用于清除内容的图标,单击此图标可以快速清除,代码如下:

```
<input type = "search" name = "googlesearch">
```

### 6. 数值范围类型 range

range 类型用于包含一定范围内数字值的输入域。range 类型显示为滑动条。还能够设定对所接收的数字的限定,代码如下:

```
<input type = "range" name = "user_range" min = "1" max = "10" />
```

其用于数字限定的属性同 number 类型的前 4 个。

### 7. 颜色类型 color

color 类型用来选取颜色,它提供了一种颜色选取器,代码如下:

```
<input type = "color" name = "favcolor">
```

### 8. (Date Pickers)日期选择器

<input >标签中与时间日期选择相关的 type 属性值有以下 6 种:

- (1) date 用于选取日、月、年。
- (2) month 用于选取月、年。
- (3) week 用于选取周和年。
- (4) time 用于选取时间(小时和分钟)。

(5) `datetime` 用于选取时间、日、月、年(UTC 时间,有时区)。

(6) `datetime-local` 用于选取时间、日、月、年(本地时间)。

使用日期选择器的语法格式如下:

```
<input type = "日期类型" name = "date"/>
```

其中, `type` 属性值可以填写上边 6 种类型中的任意一种。

新增 `input` 标签类型的综合应用如例 3-1 所示。

### 【例 3-1】 表单新增输入类型

```
<!DOCTYPE html >
<html lang = "en">
<head >
  <meta charset = "UTF - 8">
  <title>表单新增输入类型</title >
</head >
<body >
  <form action = "">
    <fieldset >
      <legend>学生档案</legend>
      姓名: <input type = "text" name = "username"/><br >
      相片: <input type = "file" name = "file"
        multiple = "multiple"/><br >
      邮箱: <input type = "email" name = "mail"><br >
      博客:<input type = "url" name = "blog"><br >
      身高: <input type = "number" max = "226" min = "80" step = "10"
        value = "170" /><br >
      体重: <input type = "range" max = "500" min = "30" step = "5"
        value = "65" /><br >
      电话: <input type = "tel" name = "tel"
        pattern = "^\\d{3} - \\d{8} | \\d{4} - \\d{7} $"><br >
      肤色: <input type = "color" name = "color"><br >
      艺名: <input type = "search" name = "name"><br >
      出生日期: <input type = "date" name = "date"><br >
      生日: <input type = "time" name = "time"><br >
      入学日期: <input type = "datetime" name = "datetime"><br >
      毕业日期: <input type = "datetime - local"
        name = "datetime - local"><br >
      毕业月份: <input type = "month" name = "month"><br >
      毕业星期: <input type = "week" name = "week" id = ""><br >
      <input type = "submit">
    </fieldset >
  </form >
</body >
</html >
```

在浏览器中的显示效果如图 3-3 所示。



图 3-3 表单新增输入类型综合应用效果

### 3.1.2 HTML5 新增元素标签

HTML5 新增表单元素有 datalist、keygen、output 等,其功能描述如表 3-3 所示。

表 3-3 HTML5 新增表单元元素标签

标签名称	描述
datalist	定义选项列表,需与 input 标签配合使用
keygen	规定用于表单的密钥对生成器
output	定义不同类型的输出

#### 1. datalist 标签

datalist 标签用来定义选项列表,它无法单独使用,需要与 input 标签配合使用,来定义 input 可能的值。初始加载时 datalist 元素及其选项不会被显示出来,它仅仅是合法的输入值列表。

input 标签的 list 属性要与 datalist 标签的 id 属性一致才能进行绑定,如例 3-2 所示。

#### 【例 3-2】 datalist 标签应用

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

```
< head >
  < meta charset = "UTF - 8">
  < title> datalist 标签应用</title>
</head >
< body >
  < form action = "">
    < input id = "input" list = "input_list">
    < datalist id = "input_list">
      < option value = "小红">
      < option value = "小明">
      < option value = "小丽">
      < option value = "大华">
    </datalist >
  </form >
</body >
</html >
```

在浏览器中的显示效果如图 3-4 所示。

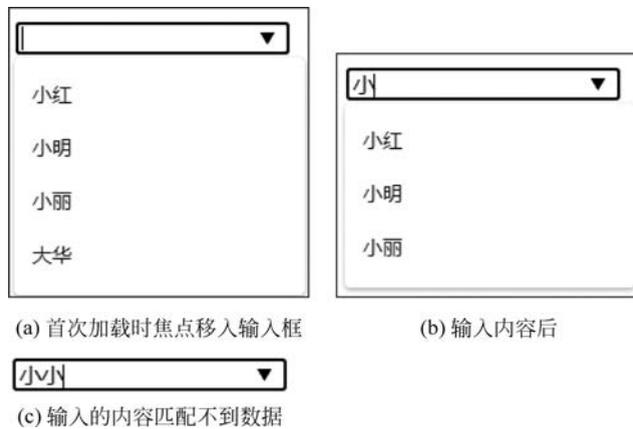


图 3-4 datalist 标签应用效果

## 2. keygen 标签

keygen 标签的作用是提供一种验证用户的可靠方法。keygen 标签是密钥对生成器 (key-pair generator)。当提交表单时, 会生成两个键, 一个是私钥, 另一个是公钥。私钥 (Private Key) 存储于客户端, 公钥 (Public Key) 则被发送到服务器端。公钥可用于之后验证用户的客户端证书 (Client Certificate), 代码如下:

```
< form action = "">
  < input type = "text" name = "name" />< br >
  Encryption:
```

```
<keygen name = "security" /><!-- 加入密钥安全 -->
<br><input type = "submit" />
</form>
```

目前,各大浏览器对此元素的支持程度都不高。

### 3. output 标签

output 标签用定义不同类型的输出(例如脚本的输出)。output 标签通常和 form 表单一起使用,用来输出显示计算结果。output 标签有 3 个属性,如表 3-4 所示。

表 3-4 output 标签属性

属 性	值	描 述
for	元素的 id 名称	定义输出域相关的一个或多个元素
form	表单的 id 名称	定义输入字段所属的一个或多个表单
name	自定义名称	定义对象的唯一名称(表单提交时使用)

基本语法格式如下:

```
<output name = "名称" for = "element_id">默认内容</output >
```

**注意:** output 标签中的内容为默认显示内容,它会随着相关元素的改变而变化。使用 output 标签实现加法计算器的应用,如例 3-3 所示。

#### 【例 3-3】 output 标签应用

```
<!DOCTYPE html >
<html lang = "en">
<head >
  <meta charset = "UTF - 8">
  <title> output 标签应用</title >
</head >
<body >
  <h4 > output 标签演示: </h4 >
  <h5 >加法计算器</h5 >
  <form oninput = "x. value = parseInt(a. value) + parseInt(b. value) ">
    <input type = "number" id = "a" value = "0"> +
    <input type = "number" id = "b" value = "0"> =
    <output name = "x" for = "a b"> 0 </output >
    <br >
    <input type = "submit">
  </form >
</body >
</html >
```

在浏览器中的显示效果如图 3-5 所示。



(a) 首次加载后的效果

(b) 计算后的效果

图 3-5 加法计算器效果

### 3.1.3 HTML5 新增属性

HTML5 给表单新增了一些属性,如表 3-5 所示。

表 3-5 新增常用属性

属性	值	描述
placeholder	<input type="text" placeholder="请输入用户名">	占位符,提供可描述输入字段预期值的提示信息
autofocus	<input type="text" autofocus >	规定当页面加载时 input 元素应该自动获得焦点
multiple	<input type="file" multiple >	多文件上传
autocomplete	<input type="text" autocomplete="off">	规定表单是否应该启用自动完成功能,有两个值: on 和 off,on 表示记录已经输入的值
required	<input type="text" required >	校验控件为必填项,内容不能为空
accesskey	<input type="text" accesskey="s">	规定激活(使元素获得焦点)元素的快捷键,采用 Alt+s 的形式

#### 1. placeholder 属性

placeholder 属性为 input 控件提供一种提示信息,该属性的值将会以灰色的字体显示在文本框中,当文本框获得焦点时,提示信息消失;当失去焦点时,提示信息显示(前提是该文本框的内容为空)。

语法格式如下:

```
<input type="text" name="userName" placeholder="请输入用户名">
```

#### 2. autofocus 属性

autofocus 是指在页面加载时,控件自动获得焦点,可以直接输入内容。这个属性在注册登录页面及表单的第 1 项 input 中比较实用。

语法格式如下：

```
<input type = "text" name = "username" autofocus >
```

**注意：**一个页面只能有一个控件有该属性。

### 3. autocomplete 属性

autocomplete 属性用于启用/关闭自动完成功能。其作用是对表单字段的值填写完成后记录所填写的值,当再返回时,恢复之前表单字段的值,即用户正在输入的内容是否显示曾经填写过的内容选项,此功能取决于两个属性值 on(开启)、off(关闭)。autocomplete 属性适用于 < form >, 以及下面的 < input > 类型: text、search、url、telephone、email、password、datepickers、range 及 color,如例 3-4 所示。

#### 【例 3-4】 autocomplete 属性的应用

```
<!DOCTYPE html >
<html lang = "en">
<head >
  <meta charset = "UTF - 8">
  <title> autocomplete 属性的应用</title >
</head >
<body >
<form action = "http://www.baidu.com" method = "get">
  姓氏: <input type = "text" name = "xing" autocomplete = "on"><br >
  名字: <input type = "text" name = "ming" autocomplete = "on"><br >
  地址: <input type = "text" name = "dizhi" autocomplete = "off"><br >
  <input type = "submit" value = "提交">
</form >
</body >
</html >
```

在浏览器中的显示效果如图 3-6 所示。



图 3-6 autocomplete 属性的应用效果

为标签添加 autocomplete="on"属性,开启自动提示内容效果。图 3-6(a)为页面首次加载后的效果,由图可见与普通框没有什么区别。先在文本框中输入一次关键词(如 admin)并单击“提交”按钮后重新回到该页面,在第 2 次重新输入内容时会在输入框下方自动显示出曾经填写过的关键词内容,如图 3-6(b)所示的效果。

#### 4. required 属性

一旦为某输入型控件设置了 required 属性,则此项必填,不能为空,否则无法提交表单。

以文本输入框为例,要将其设置为必填项,按照以下方式添加 required 属性即可:

```
<input type = "text" name = "username"required>
```

required 属性是最简单的一种表单验证方式。

#### 5. accesskey 属性

accesskey 属性的示例代码如下:

```
用户名:<input type = "text" name = "name" accesskey = ";">  
密码:<input type = "text" name = "password" accesskey = "p">  
<a href = "http://www.baidu.com" accesskey = "x">单击此处</a>
```

将这些标签设置 accesskey 属性后,使用 Alt+accesskey 属性中设置的值,就可激活对应的 HTML 元素,例如,在浏览器中按快捷键 Alt+x 时会跳转到超链接的百度页面;按快捷键 Alt+p 时密码文本框会自动获取焦点。

**注意:** accesskey 的值可以为任意字母、数字、标点符号,即键盘上存在的字符(前提是这个快捷键没有被占用)。

表单新增属性的综合应用如例 3-5 所示。

#### 【例 3-5】 表单新增属性综合应用

```
<!DOCTYPE html >  
<html lang = "en">  
<head >  
  <meta charset = "UTF - 8">  
  <title>表单新增属性综合应用</title >  
</head >  
<body >  
  <form action = "">  
    <fieldset >  
      <legend align = "center">学生档案</legend >  
      <!-- autofocus 聚焦 required 必填项 placeholder 提示信息 -->  
      用户名: <input type = "text" name = "username" autofocus  
        placeholder = "请输入用户名" required><br >  
      住宿费: <input type = "number" name = "money" min = "5" max = "1000"><br >
```

```

        <!-- autocomplete 启动记忆功能 -->
        手机号: <input type = "tel" name = "phone"
                required pattern = "[0-9].{10}" autocomplete = "on"><br >
        相片: <input type = "file" name = "files" multiple><br >
        <input type = "submit">
    </fieldset>
</form>
</body>
</html>

```

在浏览器中的显示效果如图 3-7 所示。



图 3-7 表单新增属性综合应用效果

## 3.2 HTML5 新增结构标签

### 3.2.1 新增文档结构标签

在 HTML5 之前,使用机器来阅读一个网页是非常困难的,通常用<div>对网页整体布局,常见网页布局包括页眉、页脚、导航菜单和正文部分,为了区分文档中的不同<div>内容,一般会为其配上不同的 id 来标识,代码如下:

```

<div id = "header">
    这是网页的页眉
</div>
<div id = "nav">
    这是网页的导航
</div>
<div id = "content">
    这是网页的正文
</div>
<div id = "footer">
    这是网页的页脚
</div>

```

这些元素的用途,从机器搜索引擎的角度出发,它并不认识这些 div 元素具体是用来做什么的,因为它看不懂这些 id 的意义,所以为了能够让机器理解这些元素的意义,HTML5 新增了语义化标签来代替之前的 div 布局方式,这样的网页结构对于搜索引擎更加友好,使网页内容能够更好地被搜索引擎抓取,新增文档结构标签如表 3-6 所示。

表 3-6 HTML5 新增文档结构标签

标 签	描 述
< header >	页眉标签,定义文档的标题
< nav >	导航标签,定义导航菜单栏
< section >	节标签,定义节段落
< aside >	侧栏标签,定义网页正文两侧的侧栏内容
< article >	文章标签,定义正文内容
< footer >	页脚标签,定义文档的页脚

以语义化文档结构标签来搭建网页的结构,当去掉 CSS 后,网页结构不会变,这样可使页面具有逻辑性结构、容易维护,并且对数据挖掘服务更友好,如图 3-8 所示。

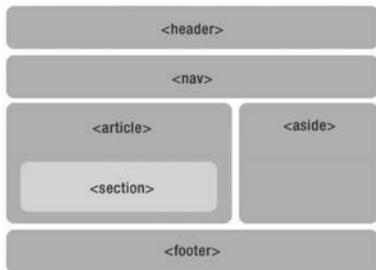


图 3-8 以语义化标签搭建网页结构

### 3.2.2 新增格式标签

#### 1. 记号标签< mark >

< mark >标签表示为引用或符号目的而标记或突出显示的文本,这是由标记的段落在封闭上下文中的相关性或重要性造成的,会给标记文字添加黄色底色。

具体语法格式如下:

```
<p>你是< mark >大长腿</mark >吗?</p >
```

**注意:** Internet Explorer 9+、Firefox、Opera、Chrome 及 Safari 支持< mark >标签。

#### 2. 度量标签< meter >

< meter > 标签用来定义度量衡,仅用于已知最大值和最小值的度量。例如,磁盘使用情况、查询结果等。

基本语法格式如下：

```
<meter value = "值"></meter>
```

<meter> 标签是 HTML5 的新标签，有一系列属性用于辅助显示效果，属性列表如表 3-7 所示。

表 3-7 <meter> 标签属性

属性	值	描 述
high	number	设置范围最高的值。 如果该属性值小于 low 属性值，则把 low 属性值视为 high 属性值； 如果该属性值大于 max 属性值，则把 max 属性值视为 high 属性值
low	number	设置范围最低的值，必须小于或等于 high 属性值。 如果 low 属性值小于 min 属性值，则浏览器会把 min 属性值视为 low 属性值
max	number	规定范围的最大值，默认值为 1。 如果设定该属性值小于 min 属性的值，则浏览器会把 min 设置为最大值
min	number	规定范围的最小值(值不可小于 0)，默认值为 0
optimum	number	设置度量衡的最佳值。 必须在 min 属性值与 max 属性值之间，可以大于 high 属性值
value	number	必须设置度量的当前值。默认值为 0，可指定浮点数小数位

注意：

- (1) <meter> 不能作为一个进度条来使用，进度条应用 progress 标签。
- (2) <meter> 标签是双标签，但在 <meter> 和 </meter> 之间的元素内容是不可见的，也就是此内容不在浏览器中显示。

<meter> 标签的具体用法如下：

- (1) <meter> 标签用于定义度量衡为 0% 的情况。

当 <meter> 标签无 value 属性、value 属性值为空或为 0 时，整个度量衡区间为灰色，如图 3-9 所示。

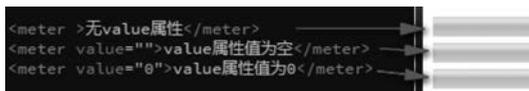


图 3-9 <meter> 标签无 value 值时的效果

当 value 属性值小于或等于 min 属性值时，整个度量衡区间为灰色，如图 3-10 所示。



图 3-10 当 value 属性值小于或等于 min 属性值时的效果

说明：实际上，当 value 属性值小于 0 或小于 min 属性值时没有意义。

(2) 当只有 value 属性的情况下，min 属性值默认为 0，max 属性值默认为 1。当 value 属性值小于或等于 0 或 min 属性值时，整个度量衡区间为灰色，如图 3-11 所示。

当 value 属性值大于或等于 max 属性值时，整个度量衡区间为绿色，如图 3-12 所示。



图 3-11 value 小于或等于 min 值时



图 3-12 value 大于或等于 max 值时

当 value 属性值在 min 属性值和 max 属性值之间时，min 到 value 的区间为绿色，而 value 到 max 区间为灰色，如图 3-13 所示。

(3) <meter> 标签度量衡显示比例。

以下 3 种情况，度量衡显示比例一致，如图 3-14 所示。



图 3-13 value 在 min 和 max 值之间时

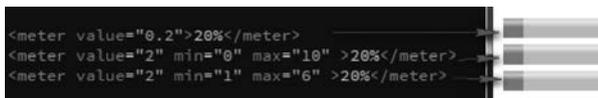


图 3-14 度量衡显示比例效果

(4) <meter> 标签度量衡颜色变化。

当 <meter> 标签同时出现了 min、max、low、high、value 属性时，度量衡显示的颜色会有以下几种变化：

当 value 值在 low 和 high 之间时，显示为绿色，代码如下：

```
<meter value = "5" min = "0" low = "3" high = "6" max = "10"></meter >
```

以上代码在浏览器中的显示效果如图 3-15 所示。



图 3-15 value 值在 low 和 high 之间时

当 value 值在 min 和 low 之间或在 high 和 max 之间时，显示为黄色，代码如下：

```
<meter value = "2" min = "0" low = "3" high = "6" max = "10"></meter >
<meter value = "8" min = "0" low = "3" high = "6" max = "10"></meter >
```

以上代码在浏览器中的显示效果如图 3-16 所示。

当 <meter> 标签同时出现了 min、max、low、high、value、optimum 属性时，度量衡的颜色变化会有更多情况，还会出现红色。

以上代码在浏览器中的显示效果如图 3-17 所示。



图 3-16 value 值在 min 和 low 之间或在 high 和 max 之间时



图 3-17 <meter>中同时出现 5 个属性

### 3. 进度条标签< progress >

< progress > 标签用于定义一个进度条,用途很广泛,可以用在文件上传的进度显示及文件下载的进度显示,也可以作为一种 loading 的加载状态条使用。通常与 JavaScript 一同使用,以此来显示任务的进度。

该属性加上 max 和 value 属性分别表示任务进度的最大值和当前值,示例代码如下:

```
< progress max = "100" value = "20"></progress >
```

## 3.3 HTML5 媒体标签

HTML5 新增了音频和视频标签,使用这些标签可以在页面上直接播放当前浏览器所支持的音频或视频格式,无须再安装插件(如 Flash)来播放。

### 3.3.1 音频

< audio > 标签用来定义声音(音乐或其他音频流),有了这个标签就可以在个人网站中引入声音了。目前,< audio > 元素支持的 3 种文件格式为 MP3、Wav、Ogg。

基本语法格式如下:

```
< audio src = "音频地址">此浏览器不支持 audio 标签。</audio >
```

**提示:** 可以在 < audio > 和 </audio > 之间放置文本内容,这些文本信息将会被显示在那些不支持 < audio > 标签的浏览器中。

< audio > 标签有一系列属性用于对音频文件的播放进行设置,如表 3-8 所示。

表 3-8 < audio > 标签属性

属 性	值	描 述
autoplay	autoplay	如果出现该属性,则音频就绪后马上播放
controls	controls	如果出现该属性,则向用户显示音频控件(例如播放/暂停按钮)
loop	loop	如果出现该属性,则每当音频结束时重新开始播放
muted	muted	如果出现该属性,则音频输出为静音

续表

属 性	值	描 述
preload	auto	规定当网页加载时,音频是否默认被加载及如何被加载
	metadata	
	none	
src	URL	规定音频文件的 URL,必需的属性

**注意:** preload 属性和 autoplay 属性不能同时使用。当属性名与值完全相同时,可以简写,如 autoplay="autoplay",可简写为 autoplay。

< audio > 标签支持多个 < source > 标签,< source > 标签可以嵌套在 < audio > 容器内,用来引入多个音频,浏览器会选择一个支持的音频格式进行加载,对于不支持 < audio > 标签的浏览器,< source > 元素也可以作为浏览器不识别的内容加入文档中,使用语法如下:

```
< audio >
  < source src = "music.mp3">
  < source src = "music.ogg">
  < source src = "music.wav">
</audio >
```

**注意:** Internet Explorer 8 及更早的版本不支持 < audio > 标签。使用 < audio > 标签来完成网页中音频的引入,如例 3-6 所示。

#### 【例 3-6】 < audio > 标签应用

```
<!DOCTYPE html >
< html lang = "en">
< head >
  < meta charset = "UTF-8">
  < title > audio 标签应用</title >
</head >
< body >
  < h5 >晚风 -- 伍佰</h5 >
  <!-- autoplay 自动播放 controls 音频控件 loop = "2"播放两次 muted 静音 -->
  < audio src = "song.mp3" autoplay controls loop = "2" muted ></audio >
  < hr />
  <!-- 为了浏览器兼容,可以提供 3 种声音文件 Ogg、Mp3、Wav -->
  < audio controls >
    < source src = "song.mp3" />
    < source src = "song.ogg" />
    此浏览器不支持音频播放
  </audio >
</body >
</html >
```

在浏览器中的显示效果如图 3-18 所示。



图 3-18 <audio>标签应用效果

### 3.3.2 视频

<video> 标签用于定义视频,例如电影片段或其他视频流。

基本语法格式如下:

```
<video src = "视频地址 URL" controls = "controls">
    此浏览器不支持 video 标签
</video>
```

**注意:** 可以在 <video> 和 </video> 标签之间放置文本内容,这样不支持 <video> 元素的浏览器就可以显示出该标签的信息。

<video> 元素支持 3 种视频格式: MP4、WebM、Ogg。

- (1) MP4 = 带有 H.264 视频编码和 AAC 音频编码的 MPEG 4 文件。
- (2) WebM = 带有 VP8 视频编码和 Vorbis 音频编码的 WebM 文件。
- (3) Ogg = 带有 Theora 视频编码和 Vorbis 音频编码的 Ogg 文件。

<video> 标签提供了播放、暂停和音量控件来控制视频。同时,<video> 元素也提供了 width 和 height 属性控制视频的尺寸。如果设置了高度和宽度,所需的视频空间则会在页面加载时保留。当然还提供了其他属性对视频播放进行设置,如表 3-9 所示。

表 3-9 <video> 标签属性

属性	值	描述
autoplay	autoplay	如果出现该属性,则视频就绪后马上播放
controls	controls	如果出现该属性,则向用户显示控件,例如“播放”按钮
height	pixels	设置视频播放器的高度
loop	loop	如果出现该属性,则当媒介文件完成播放后再次开始播放

续表

属性	值	描述
muted	muted	规定视频的音频输出应该被静音
poster	URL	规定视频下载时显示的图像(视频封面),或者在用户单击“播放”按钮前显示的图像
preload	preload	如果出现该属性,则视频在页面加载时进行加载,并预备播放。如果使用 "autoplay",则忽略该属性
src	url	要播放的视频的 URL
width	pixels	设置视频播放器的宽度,必需的属性

< video >标签支持多个 < source > 标签。< source >标签可以连接不同格式的视频文件,这样浏览器会自动选择一个它所支持的视频格式类型来展示,而忽略其他类型,使用格式如下:

```
< video width = "320" height = "240" controls >
  < source src = "movie.mp4" type = "video/mp4">
  < source src = "movie.ogg" type = "video/ogg">
  此浏览器不支持 video 标签
</video >
```

使用< video >标签来完成网页中视频的引入,如例 3-7 所示。

### 【例 3-7】 < video >标签应用

```
<!DOCTYPE html >
< html lang = "en">
< head >
  < meta charset = "UTF - 8">
  < title > video 标签应用</title >
</head >
< body >
  <!-- controls 视频控件 loop 循环播放 muted 静音 poster 封面 -->
  < video src = "video.mp4" width = "320" height = "240" controls loop muted
    poster = "images/guangtouqiang. jpg"></video >

  < video width = "320" height = "240" controls >
  < source src = "video.mp4" type = "video/mp4">
  < source src = "video.ogg" type = "video/ogg">
  此浏览器不支持 video 标签
  </video >
</body >
</html >
```

在浏览器中的显示效果如图 3-19 所示。

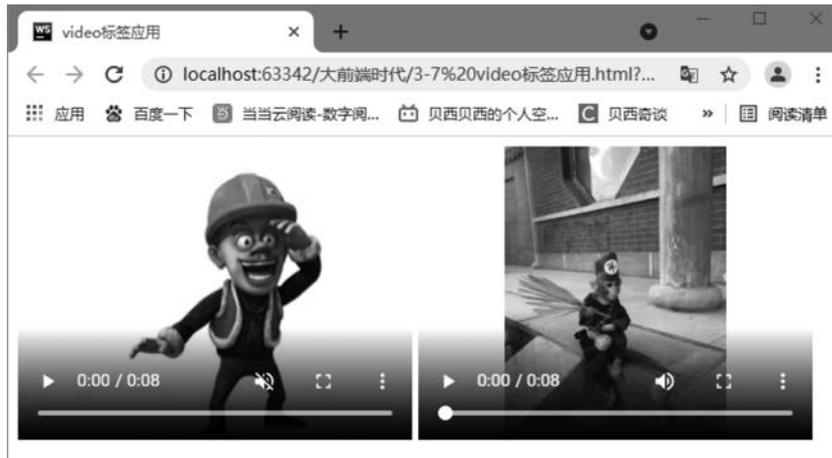


图 3-19 <video>标签应用效果