# 第二章 HTML5中的表单

1.新增的表单属性

2.新增的表单类型

3.新增的表单的验证

4.新增的页面元素

在创建web应用中，一定会应用到大量的表单元素。在HTML5标准中，吸纳了WebForms2.0的标准，大大强化表单元素的功能，使表单的开发更加快捷，更方便。

一) 新增的属性

1.1 placeholder

当用户还没有输入值的时候，输入型控件可以通过placeholder属性向用户描述性说明或者提示信息。使用placeholder属性只需要将说明性文字作为该属性值即可。

|  |
| --- |
| <input type="text" placeholder="...."/> |

1.2 autocomplete属性

浏览器通过autocomplete属性能够知晓是否应该保存输入值以备将来使用。

autocomplete属性应该用来保护敏感用户数据，避免浏览器对它们进行不安全的存储。autocomplete属性，可以指定on,off两个值，设置为on的时候，可以显示指定候补输入的数据列表。

|  |
| --- |
| <input type="text" placeholder="...." autocomplete="on"/> |

1.3 autofocus属性

给文本框,选择框或按钮组件加上该属性，当画面打开时，该控件自动获取光标焦点。

|  |
| --- |
| <input type="text" placeholder="...." autofocus/> |

注意:一个页面上只能有一个控件具有该属性，不可以出现多个。

1.4 list属性

在HTML5中，为单行文本增加一个list属性，该属性的值为某个datalist元素的id。dataList元素也是HTML5新增元素，该元素类似于选择框，但是当用户想要设置的值不在选择列表之内，允许其自行输入。该元素本身并不显示，而是当文本框获得焦点时以输入的方式显示。

|  |
| --- |
| <input type="text" list="greet" />  <datalist id="greet">  <option>吃饭</option>  <option>喝水</option>  <option>打球</option>  </datalist> |

1.5 min和max属性

通过属性min和max属性，可以将range输入框的数值范围限定在最低值和最高值之间。这两个特性既可以值设置一个，也可以两个设置，当然也可以不设置。例如：创建一个表示型大小的range空间，值范围为0%-100%

|  |
| --- |
| <input id="text" type="range" min="0" max="100" value="0" /> |

注意:默认的min为0，max的值为100

1.6 step属性

对于输入型控件，设置其step属性能够设定输入值递增或递减的梯度。例如，如下的方式将表示型大小的range控件的step属性值设置为5

|  |
| --- |
| <input id="text" type="range" min="0" max="100" value="0" step="5" /> |

设置完成后，控件可接受的输入只能是初始值和5的倍数。

二）新增的表单类型

在HTML5中，增加与改良了input元素的种类，可以简单地使用这些元素来实现HTML5之前需要使用javascript才能实现的许多功能。

2.1 email输入类型

email类型的input元素是一种专门用来输入Email地址的文本框。提交时如果该文本框中的内容不是E-mail地址格式的文字则不允许提交，但是它不检查Email是否存在。

|  |
| --- |
| <input id="text" type="email" required=""/> |

2.2 url输入类型

url类型的input元素是一种专门用来输入url地址的文本框。提交时如果该文本框中内容不是url地址格式的文字，则不允许提交。

|  |
| --- |
| <input id="text" type="url" /> |

2.3 date输入类型

date输入类型比较受欢迎的一种元素，我们常常在网页中要求输入各种各样的日期，如生日，购买日期，订单等。data类型的input元素以日历的形式方便用户输入。

|  |
| --- |
| <input id="text" type="date" /> |

2.4 time输入类型

time类型的input元素是一种专门用来输入时间的文本框，并且在提交时会对输入时间的有效性进行检查。它的外观取决于浏览器。

|  |
| --- |
| <input id="text" type="time" /> |

2.5 datetime输入类型

datetime类型的input元素是一种专门用来输入UTC日期和事件的文本框，并在提交时对输入的日期和时间进行有效的检查

|  |
| --- |
| <input id="text" type="datetime" /> |

2.6 datetime-local输入类型

datetime-local类型的input元素是一种专门用来输入本地日期和时间的文本框，并且在提交时会对输入的日期和时间进行有效性的检查。datetime-local类型的input元素的使用方法如下

|  |
| --- |
| <input id="text" type=" datetime-local " /> |

2.7 month输入类型

month类型的input元素是一种专门用来输入月份的文本框，并且在提交时会对输入的月份的有效性进行检查。

|  |
| --- |
| <input id="text" type=" month " /> |

2.8 number输入类型

number类型的input元素是一种专门用来输入数字的文本框，并且在提交的时候会检查其中的内容是否是数字。

|  |
| --- |
| <input id="text" type=" number " /> |

2.9 tel输入类型

tel类型的input元素输入电话号码的专用文本框。但是它没有特殊的检验规则。

## 2.10 color输入类型

Color类型的input元素用来选取颜色，它提供了一个颜色选择器。

## 2.11 search输入类型

Search类型的input元素是一种专门用来搜索关键词的文本框。

1. 对表单验证

在HTML5中，可以实现在表单提交时执行自动验证的功能。下面是关于在HTML5中追加的关于对元素内输入内容进行限制属性的指定。

## 3.1 required属性

required属性的主要目的的是确保表单控件中的值已填写。在提交时，如果元素中内容为空白，则不允许提交，同时在浏览器中显示文字，提示用户该元素中必须输入内容。

## 3.2 pattern属性

pattern属性根据表单控件上设置的格式规则验证输入是否为有效格式。对input元素使用pattern属性，并且将属性设为某个格式的正则表达式，在提交时会检查其内容是否符合给定格式，并且会给出相对应的提示。

|  |
| --- |
| <form>  <input type="text" placeholder="请输入1个数字与三个大写字母" name="username" pattern="[0-9][a-z]{3}" required="required"/>  <input type="submit" />  </form> |

## 3.3 使用setCustomValiduty方法自定义错误信息

HTML5中许多新的input元素都带有对于输入内容的有效性检查，如果检查不通过，浏览器会针对于该元素提示错误信息，但有时开发者不想使用这些默认的错误信息提示，而是想自定义错误信息框。那么，就可以使用到我们setCustomValiduty（）这个方法。

|  |
| --- |
| <form action="article.html">  <input type="text" placeholder="请输入第一次密码" required="required" id="password1" />  <input type="text" placeholder="请输入第二次密码" required="required" id="password2"/>  <input type="submit" id="btn" />  </form>  <script>  var oBtn = document.getElementById("btn")  oBtn.addEventListener("click",function(){  var oPassword1 = document.getElementById("password1")  var oPassword2 = document.getElementById("password2")  if(oPassword1.validity.valueMissing){  oPassword1.setCustomValidity("不能为空啊..")  }else {  oPassword1.setCustomValidity("")  }    if(oPassword2.validity.valueMissing){  oPassword2.setCustomValidity("不能为空啊..")  }else {  oPassword2.setCustomValidity("")  }    if(oPassword1.value!=oPassword2.value){  oPassword2.setCustomValidity("不一致..")  }else {  oPassword2.setCustomValidity("")  }  })  </script> |

# 总结

HTML5表单极大地提高了开发者的工作效率，并给用户带来了一些新的体验。HTML5提供了很多直接使用的功能，而以前需要大量的定制代码才能实现这些功能。