

第四章 表单新功能解析及地理位置定位

4.1 HTML5新的input类型

1. Input 类型: color

color 类型用在input字段主要用于选取颜色

选择你喜欢的颜色: `<input type="color" name="favcolor">`

2. Input 类型: date

date 类型允许你从一个日期选择器选择一个日期。

出生日期(日期): `<input type="date" name="bday">`

3. Input 类型: datetime

datetime 类型允许你选择一个日期 (UTC 时间) 。

生日 (日期和时间): `<input type="datetime" name="bdaytime">`

4. Input 类型: datetime-local

datetime-local 类型允许你选择一个日期和时间 (无时区).

生日 (日期和时间): `<input type="datetime-local" name="bdaytime">`

5. Input 类型: email

email 类型用于应该包含 e-mail 地址的输入域(在提交表单时, 会自动验证 email 域的值是否合法有效)。

6. Input 类型: month

month 类型允许你选择一个月份。

生日 (月和年): `<input type="month" name="bdaymonth">`

7. Input 类型: number

number 类型用于应该包含数值的输入域。还能够设定对所接受的数字的限定

数量 (1 到 5 之间): `<input type="number" name="quantity" min="1" max="5">`

使用下面的属性来规定对数字类型的限定:

属性	描述
disabled	规定输入字段是禁用的
max	规定允许的最大值
maxlength	规定输入字段的最大字符长度
min	规定允许的最小值
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	规定输入字段中的可见字符数
step	规定输入字段的合法数字间隔
value	规定输入字段的默认值

8. Input 类型: range

range 类型用于应该包含一定范围内数字值的输入域。

```
<!-- 定义一个不需要非常精确的数值（类似于滑块控制）：-->
<input type="range" name="points" min="1" max="10">
```

下面的属性来规定对数字类型的限定：

- 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
- value - 规定默认值

9. Input 类型: search

search 类型用于搜索域，比如站点搜索或 Google 搜索。

```
Search Google: <input type="search" name="googlesearch">
```

10. Input 类型: tel

tel 类型用于定义输入电话号码字段

```
电话号码: <input type="tel" name="usrtel">
```

11. Input 类型: time

time 类型允许你选择一个时间。

```
选择时间: <input type="time" name="usr_time">
```

12. Input 类型: url

url 类型用于应该包含 URL 地址的输入域。（在提交表单时，会自动验证 url 域的值）

```
添加您的主页: <input type="url" name="homepage">
```

13. Input 类型: week

week 类型允许你选择周和年。

选择周: `<input type="week" name="week_year">`

4.2 HTML5新的表单元素

1. HTML5

`<datalist>` 元素规定输入域的选项列表。

`<datalist>` 属性规定 `form` 或 `input` 域应该拥有自动完成功能。当用户在自动完成域中开始输入时, 浏览器应该在该域中显示填写的选项:

使用 `<input>` 元素的列表属性与 `<datalist>` 元素绑定。

2. HTML5 元素

元素的作用是提供一种验证用户的可靠方法。

标签规定用于表单的密钥对生成器字段。

当提交表单时, 会生成两个键, 一个是私钥, 一个公钥。

私钥 (private key) 存储于客户端, 公钥 (public key) 则被发送到服务器。公钥可用于之后验证用户的客户端证书 (client certificate) 。

```
<!--带有keygen字段的表单-->
<form action="demo_keygen.asp" method="get">
  用户名: <input type="text" name="usr_name">
  加密: <keygen name="security">
  <input type="submit">
</form>
```

3. HTML5 元素

元素用于不同类型的输出, 比如计算或脚本输出

```
<form oninput="x.value=parseInt(a.value)+parseInt(b.value)">0
  <input type="range" id="a" value="50">100 +
  <input type="number" id="b" value="50">=
  <output name="x" for="a b"></output>
</form>
```

4.3 HTML5 新的表单属性

1. / autocomplete 属性

autocomplete 属性规定 form 或 input 域应该拥有自动完成功能。有两个值, 一个是off, 一个是on,on代表记录已经输入的值。

注意: autocomplete 适用于

标签, 以及以下类型的 标签: text, search, url, telephone, email, password, datepickers, range 以及 color。

使用这个属性, 必须具备以下两个条件:

- autocomplete首先必须有提交按钮
- 使用这个属性的表单或表单元素必须给名字（即给name属性）

2. novalidate属性

novalidate 是一个布尔（true 或 false）属性。

novalidate 属性是 HTML 表单元素的一个布尔属性，用于设置浏览器不对表单进行验证。

当该属性被添加到

元素上时，浏览器将不会执行默认的表单验证，不会检查输入字段是否符合指定的验证规则。

使用 novalidate 属性可以让开发者完全控制表单验证的逻辑，可以通过 JavaScript 或其他方式来自定义表单验证的行为。

3. autofocus 属性

autofocus 属性是一个布尔属性。autofocus 属性规定在页面加载时，域自动地获得焦点。

```
<!--让 "First name" input 输入域在页面载入时自动聚焦-->
First name:<input type="text" name="fname" autofocus>
```

4. form 属性

form 属性规定输入域所属的一个或多个表单。

```
<!--位于form表单外的 input 字段引用了 HTML form（该 input 表单仍然属于form表单的一部分）：-->
<form action="demo-form.php" id="form1">
  First name: <input type="text" name="fname"><br>
  <input type="submit" value="提交">
</form>

Last name: <input type="text" name="lname" form="form1">
```

提示:如需引用一个以上的表单，请使用空格分隔的列表。

5. formaction 属性

The formaction 属性用于描述表单提交的URL地址。The formaction 属性会覆盖元素中的action属性

```
<form action="demo-form.php">
  First name: <input type="text" name="fname"><br>
  Last name: <input type="text" name="lname"><br>
  <input type="submit" value="提交"><br>
  <input type="submit" formaction="demo-admin.php"
    value="提交">
</form>
```

注意: The formaction 属性用于 type="submit" 和 type="image".

6. formenctype 属性

formenctype 属性描述了表单提交到服务器的数据编码（只对form表单中 method="post" 表单）。formenctype 属性覆盖 form 元素的 enctype 属性。

主要: 该属性与 type="submit" 和 type="image" 配合使用。

```
<!--第一个提交按钮以默认编码发送表单数据，第二个提交按钮以 "multipart/form-data" 编码格式发送表单数据-->
<form action="demo-post_enctype.php" method="post">
  First name: <input type="text" name="fname"><br>
  <input type="submit" value="提交">
  <input type="submit" formenctype="multipart/form-data"
    value="以 Multipart/form-data 提交">
</form>
```

7. formmethod 属性

formmethod 属性定义了表单提交的方式。formmethod 属性覆盖了元素的 method 属性。

注意: 该属性可以与 type="submit" 和 type="image" 配合使用。

```
<form action="demo-form.php" method="get">
  First name: <input type="text" name="fname"><br>
  Last name: <input type="text" name="lname"><br>
  <input type="submit" value="提交">
  <input type="submit" formmethod="post" formaction="demo-post.php"
    value="使用 POST 提交">
</form>
```

8. formnovalidate 属性

novalidate 属性是一个 boolean 属性。novalidate 属性描述了 元素在表单提交时无需被验证。formnovalidate 属性会覆盖元素的 novalidate 属性。

注意: formnovalidate 属性与 type="submit" 一起使用

```
<!--两个提交按钮的表单(使用与不适用验证)-->
<form action="demo-form.php">
  E-mail: <input type="email" name="userid"><br>
  <input type="submit" value="提交"><br>
  <input type="submit" formnovalidate value="不验证提交">
</form>
```

9. formtarget 属性

formtarget 属性指定一个名称或一个关键字来指明表单提交数据接收后的展示。formtarget 属性覆盖

元素的 target 属性。

注意: formtarget 属性与 type="submit" 和 type="image" 配合使用。

```
<form action="demo-form.php">
  First name: <input type="text" name="fname"><br>
  Last name: <input type="text" name="lname"><br>
  <input type="submit" value="正常提交">
  <input type="submit" formtarget="_blank" value="提交到一个新的页面上">
</form>
```

10. height 和 width 属性

height 和 width 属性规定用于 image 类型的 标签的图像高度和宽度。

注意: height 和 width 属性只适用于 image 类型的 标签。

```
<input type="image" src="img_submit.gif" alt="Submit" width="48" height="48">
```

11. list 属性

list 属性规定输入域的 datalist。datalist 是输入域的选项列表。

```
<input list="browsers">

<datalist id="browsers">
  <option value="Internet Explorer">
  <option value="Firefox">
  <option value="Chrome">
  <option value="Opera">
  <option value="Safari">
</datalist>
```

12. min 和 max 属性

min、max 和 step 属性用于为包含数字或日期的 input 类型规定限定（约束）。

注意: min、max 和 step 属性适用于以下类型的 标签：date pickers、number 以及 range。

```
Enter a date before 1980-01-01:
<input type="date" name="bday" max="1979-12-31">

Enter a date after 2000-01-01:
<input type="date" name="bday" min="2000-01-02">

Quantity (between 1 and 5):
<input type="number" name="quantity" min="1" max="5">
```

13. multiple 属性

multiple 属性是一个 boolean 属性。multiple 属性规定 元素中可选择多个值。

注意: multiple 属性适用于以下类型的 标签：email 和 file

```
Select images: <input type="file" name="img" multiple>
```

14. pattern 属性

pattern 属性描述了一个正则表达式用于验证 元素的值。

注意: pattern 属性适用于以下类型的 标签：text, search, url, tel, email, 和 password。

提示: 是用来全局 [title](#) 属性来描述模式。

```
Country code: <input type="text" name="country_code" pattern="[A-Za-z]{3}"
title="Three letter country code">
```

15. **placeholder 属性**

placeholder 属性提供一种提示 (hint) , 描述输入域所期待的值。简短的提示在用户输入值前会显示在输入域上。

注意: placeholder 属性适用于以下类型的 标签: text, search, url, telephone, email 以及 password。

```
<input type="text" name="fname" placeholder="First name">
```

16. **required 属性**

required 属性是一个 boolean 属性.required 属性规定必须在提交之前填写输入域 (不能为空) 。

注意:required 属性适用于以下类型的 标签: text, search, url, telephone, email, password, date pickers, number, checkbox, radio 以及 file。

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

17. **step 属性**

step 属性为输入域规定合法的数字间隔。如果 step="3", 则合法的数是 -3,0,3,6 等。

提示: step 属性可以与 max 和 min 属性创建一个区域值。

注意: step 属性与以下type类型一起使用: number, range, date, datetime, datetime-local, month, time 和 week.

```
<input type="number" name="points" step="3">
```