HTML5实战宝典

Chen

# 第1章 HTML5概述

## 一个新的web开发平台

hmtl5的出现,掀起了Web时代的新浪潮, 各大浏览器也都纷纷支持html5. html5可以使网页内容更加丰富, 不仅可以显示三维图形, 还可以在不使用Flash插件的基础上实现音频. html5是向下兼容html4的, 它是在html4的基础之上, 加进了一些新的标记、属性、功能的一个新的超文本标记语言。 例如： html5拥有新的html文档结构、新的CSS标准、API等。

html5提供了很多应用程序接口（API）， 例如基于浏览器支持的图形API、地理信息API、本地存储API和视频播放相关的API等。

## html5为什么受欢迎

### 语法特点

1. 简单的doctype

* <!doctype html> 省略了版本号，更为简洁。

1. 直观的结构

* 在html4中对于网页结构的划分大量使用了div， 需要靠类名、加注释才能很好的解释我们的结构，在html5中可以使用article、footer、header、nav、section等标签来更好地了解我们的结构。

### 1.2.2 功能特点

1. 音频、视频

* 在网页中想要实现音频、视频播放，一般都需要引入 Flash 或第三方媒体插件，并且要写很多代码，很烦琐。而在 HTML5 中，只需要写：
  + 音频: <audio src="url" autoplay loop></audio>
  + 视频: <video src="url" width="300" height="200" controls></video>
* 就可以了，可以像使用<img>标签一样来实现音频、视频播放。
* 本地存储
* HTML5 可以帮助浏览器存储一些用户的信息、缓存的数据、应用的使用状态等，这样一来，可以加快访问应用的速度；可以记录用户上一次的使用状态，在重新加载时只加载修改过的状态，节省资源。
* 强大的Canvas
* 使用 Canvas 可以达到 Flash 的效果，它可以实现动画设计和游戏开发。
* 地理信息
* HTML5 中提供了地理位置信息的 API（geolocation），通过浏览器来获取用户当前位置。在获取位置信息前，浏览器会给用户一个提示信息，只有用户同意以后才能使用。通过此特性可以开发基于位置的服务应用。

### 对移动端的支持

# 第2章 HTML5规范

## 新的文档声明和语法规范

### 文档声明

1. <!doctype>的定义

* <!doctype>声明必须位于 HTML5 文档中的第一行，也就是位于<html>标签之前。该标签告知浏览器文档所使用的 HTML 规范。
  + （1）doctype 声明不属于 HTML 标签；它是一条指令，告诉浏览器编写页面所用的标记的版本。
  + （2）在所有 HTML 文档中，规定 doctype 是非常重要的，这样浏览器就能了解预期的文档类型。
  + （3）HTML 4.01 中的 doctype 需要对 DTD 进行引用，因为 HTML 4.01 基于 SGML。而HTML 5 不基于 SGML，因此不需要对 DTD 进行引用，但是需要 doctype 来规范浏览器的行为（让浏览器按照它们的方式来运行）。
* <!doctype>的用法
* 代码案例：
* <!doctype html>  
  <html lang="en">  
  <head>  
  <meta charset="UTF-8">  
  <title>Document</title>  
  </head>  
  <body>  
  HTML5 的主体结构。  
  </body>  
  </html>
* 一个注意点
* <!doctype>对大小写不敏感，而且它没有结束标签。
* 定义文档信息的元标签
* <meta>位于头部中，它用于提供页面的元信息，用来描述网页的关键词、网页更新的频度，同时也可以为搜索引擎的搜索提供便利。它的属性还定义了与文档相关联的名称/值对（元数据总是以名称/值的形式被成对传递的）。
* meta 的属性
  1. content属性 content 属性提供了名称/值对中的值。该值可以是任何有效的字符串。content 属性始终要和 name 属性或 http-equiv 属性一起使用。
  2. http-equiv属性 http-equiv 属性为名称/值对提供了名称。并指示服务器在发送实际的文档之前，先在要传送给浏览器的 MIME 文档头部包含名称/值对。当服务器向浏览器发送文档时，会先发送许多名称/值对。虽然有些服务器会发送许多这种名称/值对，但是所有服务器都至少要发送一个：content-type:text/html。这将告诉浏览器准备接收一个 HTML 文档。使用带有 http-equiv 属性的<meta>标签时，服务器将把名称/值对添加到发送给浏览器的内容头部。例如添加：
  + <meta http-equiv="charset" content="iso-8859-1">  
    <meta http-equiv="expires" content="31 Dec 2008">
* 这样发送到浏览器的头部就应该包含：
  + content-type: text/html
  + charset:iso-8859-1
  + expires:31 Dec 2008
* 当然，只有当浏览器可以接收这些附加的头部字段，并能以适当的方式使用它们时，这些字段才有意义。
  1. name属性 name 属性提供了名称/值对中的名称。HTML 和 XHTML 标签都没有指定任何预先定义的<meta>名称。通常情况下，用户可以自由使用对自己和源文档的读者来说富有意义的名称。例如"keywords" 是一个经常被用到的名称。它为文档定义了一组关键字，某些搜索引擎在遇到这些关键字时，会用这些关键字对文档进行分类。类似的 meta 标签可能对于进入搜索引擎的索引有帮助：
  + <meta name="keywords" content="HTML,ASP,PHP,SQL">
* **说明：** keywords为 "名称/值" 对中的“名称” 如果没有提供 name 属性，那么名称/值对中的名称会采用 http-equiv 属性的值。
  1. charset属性： 提供编码方式
  + <meta charset="utf-8">
* UTF-8（8-bit Unicode Transformation Format）是一种针对 Unicode 的可变长度字符编码，也是一种前缀码，又称万国码。它可以用来表示 Unicode 标准中的任何字符，因此，它逐渐成为电子邮件、网页及其他存储或传送文字的应用中，优先采用的编码。
* GB-2312 是计算机可以识别的编码，适用于汉字处理、汉字通信等系统之间的信息交 换，它共收入汉字 6763 个和非汉字图形字符 682 个。
* GBK 是 GB2312 的扩展版本。
  1. 适用于移动端的 meta 标签
  + <meta name="viewport" content="width=device-width,initial-scale=1, maximum-  
    scale=1, minimum-scale=1,user-scalable=no">
* 其中 meta 的参数说明如下：
  + name="viewport"：表示网页窗口。
  + content 属性中的值可以有以下几种情况：
  + width：控制 viewport 的宽度，可以指定一个具体的值，用于移动端时赋值为 device-
  + width，表示它将与设备一样宽。
  + height：控制 viewport 的高度，与 width 使用方式一样。
  + initial-scale：初始缩放比例，页面第一次加载时的比例。
  + maximum-scale：最大缩放比例，取值范围为 0～10.0。
  + minimum-scale：最小缩放比例，取值范围为 0～10.0。
  + user-scalable：是否允许用户手动缩放。当值为 yes 或 true 时，表示可以缩放；当值为no 或 false 时，表示不能缩放。
  + viewport 用来重新设置设备的分辨率，让网页在设备上都正好满屏显示。

### 语法规则

① 标签要小写。 ② 属性值可以不加""或‘’。 ③ 可以省略某些标签：html、body、head、tbody。 ④ 可以省略某些结束标签：tr、td、li。 ⑤ 单标签不用加结束标签：img、input。 ⑥ 废除的标签：font、center、big。 ⑦ 新添加的标签，请查阅本章 2.3 节。

## 废弃的标签和属性

### 废弃的标签

1. 表现性的标签

* HTML 中的有些标签只是为了画面展示而服务的，比如 u 标签，在使用时，想要显示下画线的字体需要用 u 标签包起来，它只是与其他文字显示的状态（样式）不同而已，在HTML5 中像这类的状态（样式）都可以用 CSS 属性编辑，所以将这类标签废弃掉了。与 u标签类似的还有 basefont、big、center、font、s、strike、tt 等标签，它们都可以用相应的CSS 属性来代替。

1. 框架类标签

* HTML 中的框架标签 frame、frameset、noframes 对页面的可用性有负面影响，所以在HTML5 中将它们废弃了，使用 iframe 标签代替，它就像 img 标签一样，写入地址，设置好宽高就可以了 高就可以了。

1. 局限性标签

* 只有部分浏览器支持 Applet（Java 小应用程序）、bgsound（页面添加背景音乐）、blink（可以闪烁的字体）、marquee（滚动字幕）等标签，这些标签在 HTML5 中也被废弃掉了。

1. 其他被废除的标签:

* 这是一些很少用到的标签，它们特殊但没有被 W3C 列入规范，所以在 HTML5 中也被废弃了。 例如：
  + rb 标签用来设定被标示的元素对象，为 ruby 的子元素，使用 ruby 替代。
  + acronym 标签定义首字母缩写，使用 abbr 替代。
  + dir 标签定义目录列表，使用 ul 替代。
  + isindex 标签显示输入框，使用 form 与 input 相结合的方式替代。
  + listing 标签显示静态页面源代码，使用 pre 替代。
  + xmp 标签原样输出代码，使用 code 替代。
  + plaintex 标签，使用“text/plain”（无格式正文）MIME 类型替代。

## 新的结构标签和属性

### 新的标签

1. 结构性标签

* 结构性标签（construct tag）主要负责 Web 的上下文结构的定义，确保 HTML 文档的完整性，使网页的文档结构更加明确。这类标签包括以下几个：
  1. section 标签用于表达文档的一部分或一章，或者一章内的一节。在 Web 页面应用中，该标签也可以用于区域的章节表述。它用来表现普通的文档内容或应用区块，通常由内容及其标题组成。但 section 标签并非一个普通的容器元素，它表示一段专题性的内容，一般会带有标题。 代码实例：
  + <section>  
    <h1>新章节的标题</h1>  
    <article>  
    <h2>第一节的标题</h2>  
    <p>第一节的内容......</p>  
    </article>  
    </section>
  1. hgroup 标签对网页或区段（section）的标题进行组合。 代码实例：
  + <hgroup>  
    <h1>第二章 HTML 规范</h1>  
    <h2>第一节 新的结构标签和属性</h2>  
    <h3>新的标签</h3>  
    <h4>结构性标签</h4>  
    </hgroup>
  1. header 标签相当于页面主体上的头部（页眉），注意区别于 head 标签。这里可以给初学者提供一个判断区别的小技巧：head 标签中的内容往往是不可见的，而 header 标签往往在一对 body 标签之中。
  + <header>  
    <h1>网页的标题</h1>  
    <nav>上导航部分</nav>  
    </header>
  1. footer 标签相当于页面的底部（页脚）。通常，人们会在这里标出网站的一些相关信息，例如关于我们、法律申明、邮件信息、管理入口等。 代码示例：
  + 码示例：  
    <footer>  
    &copy;网页的版权声明。  
    </footer>
  1. nav 标签时专门用于菜单导航、链接导航的标签， 是navigatior的缩写。 代码实例：
  + <nav>  
    <ul>  
    <li><a href="#">首页</a></li>  
    <li><a href="#">电视</a></li>  
    <li><a href="#">平板</a></li>  
    <li><a href="#">路由器</a></li>  
    <li><a href="#">笔记本</a></li>  
    </ul>  
    </nav>
  1. article 标签用于表示一篇文章的主体内容，一般为文字集中显示的区域。 代码实例：
  + <article>  
    <header>  
    <h1>文章的标题</h1>  
    <time datetime="2015-08-08">2015.08.08</time>  
    </header>  
    <p>文章的内容</p>  
    </article>

1. 块级性标签

* 块级性标签（block tag）主要完成Web页面区域的划分， 确保内容的有效分隔， 这类标签包括以下几个。
  1. aside 标签是用以表达注记、贴士、侧栏、摘要、插入的引用等作为补充主体的内 容。从一个简单页面显示上看，就是侧边栏，可以在左边，也可以在右边。从一个页面的局 部看，就是摘要。
  + 代码实例：
  + <aside> <p>作者信息</p> </aside>
  1. figure 标签规定独立的流内容(图像、图表、照片等)，通常与 figcaption 联合使用。
  + <figure>  
    <figcaption>风景图的标题</figcaption>  
    <img src="fengjing.jpg" alt="风景图">  
    </figure>
  1. code 标签表示一段代码块
  + <code>一段电脑代码</code>
  1. dialog 标签定义对话框或窗口，配合<dt>、<dd>标签使用。它的属性 open 规定 dialog 元素是活动的，用户可与之交互。
  + <table border="1">  
     <tr>  
     <td>周一<dialog open>这是打开的对话窗口</dialog></td> <td>周二</td>  
     <td>周三</td>  
     </tr>  
     <tr>  
     <td>12</td>  
     <td>13</td>  
     <td>14</td>  
     </tr>  
     </table>
  1. Canvas 标签。它是一个画布标签，用它可以实现电脑上的画图工具，可以在网页中 画出不同的图形。

1. 行内标签

* 行内语义性标签(in-line tag)主要完成 Web 页面具体内容的引用和表述，是丰富内容 展示的基础，这类标签包括以下几个标签。
  1. meter 标签表示特定范围内的数值，可用于工资、数量、百分比等。max 表示最大 值，min 表示最小值，value 代表当前值
  + <meter value="6" min="0" max="10">6/10</meter><br>  
    <meter value="0.3">50%</meter>
  1. time 标签表示时间值，该元素能够以机器可读的方式对日期和时间进行编码，属性 datetime 强调日期和时间。
  + <p>  
    <time datetime="2015-09-27">中秋节</time>马上就到了。  
     </p>
  1. progress 标签用来表示任务的进度条，属性 max 表示最大任务值，属性 value 表示 完成了多少任务
  + <p>下载进度:</p>  
    <progress value="34" max="100"></progress>

1. 多媒体标签

* 多媒体标签(multimedia tag)，它可以让网页对视频和音频有着更好的实现，不用再与 其他的插件配合使用。HTML5 中提供了 video 视频标签与 audio 音频标签，详情参照第 8 章 HTML5 对多媒体的支持。

# 第3章 HTML5表单新功能 —— 注册和登录验证实战

## 本章重点

在 HTML 4 中，实现表单信息的录入需要用到的标签有 input 标签、select 标签、option 标签、textarea 标签、label 标签等，通过 input 的 type 属性可以调整输入框的类型。在 HTML 4 中提供了“text”“password”“radio”“checkbox”“submit”等，这些表单控件提供了不同类 型的功能，但是一些数据的验证工作还需要结合 Javascript 等脚本语言来实现。

HTML5 中的元素和特性提供了更加强大的语义标记，其中关于表单的修改令人眼前一 亮。其中有很多特性功能都非常实用，比如可以像在 Word 里面一样通过色板选择某种颜色， 或者直接选择一个日期。此外，还有一些非常好用的属性，比如单击清除的属性，正则验证属 性。虽然在不同的浏览器上的支持各有不同，但是这些新功能还是很值得我们去学习。

**本章内容包括:**

* 关于表单新的控件类型的详细介绍，包括 form 标签与 input 的标签的新写法， email 输入类型、url 输入类型、日期相关输入类型、time 输入类型、range 输入类型、search 输入类型、color 输入类型、datalist 标签等。
* 结合 HTML5 表单特性，制作一个登录页面和一个注册页面。
* 了解新的表单验证功能和属性，如 autofocus 属性、placeholder 属性、autocomplate 属性、novalidate 属性、multiple 属性、required 属性、pattern 属性、autocomplete 属性等， 并且将这些属性添加到我们所写的页面中。
* 完成一个完整的登录注册设计流程。