

第三篇

HTML5 和 CSS3 开发基础与应用

第 7 章

HTML5 简介



学习任务

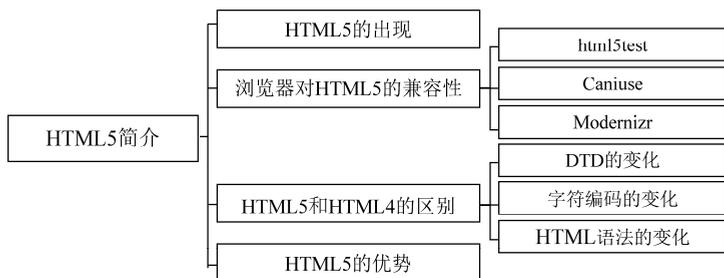
【任务 1】了解 HTML5 及主流浏览器对其的兼容性；

【任务 2】掌握 HTML5 的基本语法；

【任务 3】掌握 HTML5 和 HTML4 的区别。



学习路线



7.1 HTML5 的出现

在 HTML5 出现之前，W3C 希望用 XHTML 取代 HTML，因为当时 HTML 已经发展到一种极其混乱的程度。

XHTML 是一种更纯洁、更严格、更规范的 HTML 代码。这是因为 XHTML 要求 HTML 必须是一份 XML 文档。XML 文档是一种结构化文档，主要有 4 条基本规则。

- 在 XML 文档中有且只能有一个根元素，在 HTML 文档中仅有一对根元素 `<html></html>`。

- XML 文档中的元素必须由开始标签和结束标签组成，除非使用空元素语法，即单标签。

开始标签和结束标签示例: `<html></html>`

单标签示例: `
`

- 标签之间不得交叉使用。

错误写法: `<a>hello`

`<a>` 标签中就交叉嵌套了一个 `` 标签，这种交叉使用是错误且不规范的。

但在 HTML 文档，浏览器很多时候都可以解析这种错误且不规范的写法。

正确写法: `<a>hello`

- 属性值必须用" "括起来。

错误写法: `<input type=text />`

正确写法: `<input type="text" />`

当时互联网上存在大量不规范的 HTML 文档，而且绝大部分编写 HTML 文档的程序员并没有接受过专业的训练。所以，尽管 W3C 努力使程序员编写 HTML 文档时要遵守规范，或者按照 XHTML 规范编写 HTML，但是大部分编写 HTML 语言的程序员并不理会这种号召。

既然互联网存在这些不规范页面，并且大家也不愿意改进这些 HTML 页面，所以就产生了一个新的 HTML 标准，即 HTML5。

为了促进 Web 标准运动的发展，2004 年，一些浏览器厂商，如 Opera、Mozilla 等共同成立了 Web Hypertext Application Technology Working Group，简称 WHATWG，中文名为 Web 超文本应用技术工作组。

HTML5 草案的前身为 Web Applications 1.0，是 WHATWG 于 2004 年提出的，W3C 于 2007 年接受，并成立了一个新的 HTML 工作组，目的就是为 HTML5 完善。经过几年的努力，HTML5 第一份正式草案由 WHATWG 于 2008 年 1 月 22 日公布。

- HTML 5.1 的文档地址为 <https://www.w3.org/TR/2015/WD-html51-20151008/>。
- HTML 5.2 的文档地址为 <https://www.w3.org/TR/2017/REC-html52-20171214/>。
- HTML 5.3 的文档地址为 <https://www.w3.org/TR/2018/WD-html53-20181018/>。

HTML5 针对大量不规范的 HTML 文档，做到了良好的兼容性，如 XML 文档的 4 条基本规则在 HTML5 中并没有很严格的要求。

- 在 HTML 文档中可以有多根元素，即某些标签可以不写，浏览器会自动补全。
- 元素不一定有结束标签。
- 标签交叉使用没有问题，浏览器处理时会自动修正。
- 属性值不一定用引号括起来，甚至不一定有属性名。

不规范写法:

```
<!DOCTYPE html>
```

这里省去了 `html` 元素，导致根元素不唯一。

```
<head>
  <meta charset="utf-8">
</head>
<body>
  <img src=hello.gif alt=World>  这里属性值没有用""括起来
  <p>Hello Word!                 这里没有</p>结束标签
</body>
规范写法:
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
  </head>
  <body>
    
    <p>Hello Word! </p>
  </body>
</html>
```

后面还会对 HTML5 兼容这种不规范 HTML 文档做一个比较系统的汇总。

7.2 浏览器对 HTML5 的兼容性

由于现在的浏览器厂商很多，每个厂商的浏览器版本也很多，这势必会存在浏览器对 HTML5 的兼容性，本节提供的几个网址工具在 HTML5 开发中经常会用到。

- <http://html5test.com/>。html5test 是一个提供浏览器对 HTML5 兼容性测试，以及与其他浏览器进行对比的网站，虽然其是一个第三方 HTML5 兼容性测试网站，但现在的主流浏览器厂商都会在该网站进行 HTML5 的兼容性测试，甚至以该网站得分为自家的浏览器做宣传。
- <https://caniuse.com/>。Caniuse 是一个检测浏览器对 JS、HTML5、CSS、SVG 或者其他 Web 前端相关特性支持程度的列表，可以检测的浏览器包括桌面和移动版的主流浏览器。
- <https://modernizr.com/>。Modernizr 可以检测当前使用的浏览器是否实现了某个 feature，它可以检测很多项 HTML5 和 CSS3 的功能，比传统检测浏览器名称的方式更加可靠。

7.3 HTML5 和 HTML4 的区别

7.3.1 DTD 的变化

DTD，即文档类型定义（Document Type Definition），是通用标记语言的一部分，主要用于定义通用标记语言可以使用的元素，以及各元素支持的属性等。

HTML 4.01、XHTML 1.0 和 HTML5 中 DTD 写法如下表所示。

HTML 4.01	严格模式	<code><!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd"></code>
	过渡模式	<code><!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional //EN" "http://www.w3.org/TR/html4/loose.dtd"></code>
	框架集	<code><!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN" "http://www.w3.org/TR/html4/frameset.dtd"></code>
XHTML 1.0	严格模式	<code><!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd"></code>
	过渡模式	<code><!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional //EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd"></code>
	框架集	<code><!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd"></code>
HTML5		<code><!DOCTYPE html></code>

7.3.2 字符编码的变化

字符编码的变化如下表所示。

HTML 4.01	<code><meta http-equiv="content-type" content="text/html; charset=utf-8"></code>
HTML5	与 HTML 4.0.1 的写法相同
	新增写法: <code><meta charset="utf-8"></code>

7.3.3 HTML5 语法的变化

HTML5 语法的变化主要是为了兼容各种不规范的 HTML 文档，上文提到 HTML5 有时不是一个严格规范的 XML 文档，这些不规范的写法可归纳为以下几点。

- 某些元素可以省略结束标签。
 - 仅仅可以省略结束标签的元素有 `<p>`、``、`<dt>`、`<dd>`、`<rb>`、`<rt>`、`<rtc>`、`<rp>`、`<thead>`、`<tbody>`、`<tfoot>`、`<tr>`、`<td>`、`<th>`、`<optgroup>`、`<option>`、`<colgroup>`。
 - 绝对没有结束标签的元素有 `<area>`、`<base>`、`
`、`<col>`、`<command>`、`<embed>`、`<hr>`、``、`<input>`、`<keygen>`、`<link>`、`<menuitem>`、`<meta>`、`<param>`、`<source>`、`<track>`、`<wbr>`。
 - 可以将开始标签和结束标签一起省略的元素有 `<html>`、`<head>`、`<body>`、`<colgroup>`、`<tbody>`。
- 属性值可以不用引号括起来。

```
<!DOCTYPE html>
<head>
  <meta charset="utf-8">
</head>
<body>
  <img src=hello.gif alt=World>
```

```
<p>Hello World!
</body>
```

- 标签不再区分大小写。

规范写法：`<p>Hello World!</p>`

不规范写法：`<p>Hello World!</P>`

- 某些标志性的属性可以省略属性值，通常为布尔型的。

规范写法：

```
<input type="checkbox" checked="checked">
```

不规范写法：

```
<input type="checkbox" checked="">
```

```
<input type="checkbox" checked=checked>
```

```
<input type="checkbox" checked>
```

如果完全省略这个属性，则表示该属性的属性值为 `false`。

在 HTML5 中允许省略的属性如下表所示。

XHTML	HTML5
<code>checked="checked"</code>	<code>checked</code>
<code>defer="defer"</code>	<code>defer</code>
<code>disabled="disabled"</code>	<code>disabled</code>
<code>ismap="ismap"</code>	<code>ismap</code>
<code>multiple="multiple"</code>	<code>multiple</code>
<code>nohref="nohref"</code>	<code>nohref</code>
<code>noresize="noresize"</code>	<code>noresize</code>
<code>noshade="noshade"</code>	<code>noshade</code>
<code>nowarp="nowarp"</code>	<code>nowarp</code>
<code>readonly="readonly"</code>	<code>readonly</code>
<code>selected="selected"</code>	<code>selected</code>

以上 4 点除了单标签元素，即没有结束标签的元素，其他的基本上可以认为是不规范的写法，建议 Web 开发者日常编写 HTML5 的代码时（虽然 HTML5 支持各种不规范的 HTML 文档）应尽量写得规范一些，这样便于其他开发者阅读和理解，以及进行扩展或者修改。

7.4 HTML5 的优势

- 解决了跨浏览器、跨平台问题。

HTML5 本身是由 W3C 推荐的，它是 Google、Apple、诺基亚、中国移动等几家公司共同开发的技术，是一项公开的技术。这也就意味着每一个浏览器或者每一个平台都可以实现。

目前市面上 95% 的浏览器都支持 HTML5，这里面同时包含移动端等设备上使用的浏览器。

对开发者来说，各浏览器可以更好地支持 HTML5，前端程序员开发 HTML+CSS+JavaScript 页面将会更加轻松。

HTML5 的优势还体现在跨平台上，开发的浏览器用的是 HTML5 页面或者游戏，可以轻松移植到其他平台，如常见的使用 HTML5 开发的手机 App。

- 使用了更明确的标签。

使用明确的标签可以使代码结构更加清晰易读。尤其在复杂的嵌套结构中，能大大优化程序员的开发体验。

```
HTML 4.01 写法：
<div>
  <div>我是此结构头部</div>
  <div>我是此结构身体</div>
  <div>我是此结构底部</div>
</div>
HTML5 写法：
<div>
  <header>我是此结构头部</header>
  <div>我是此结构身体</div>
  <footer>我是此结构底部</footer>
</div>
```

由此可知，HTML5 的代码阅读性比之前提高了很多。

- 增强了 Web 的应用程序。

以前的 HTML 考虑到安全性，很多扩展功能都需要 JavaScript 和 Flash 支持。现在有很多东西已经不需要 JavaScript 和 Flash，最直观的地方就是以前在网页上播放视频都需要使用 Flash 播放器，而现在直接使用 HTML5 就可以播放视频，使手机网页看视频成为可能。

早在 2011 年，Adobe 公司做出了一个令人惊讶的决定：停止 Android 移动设备 Flash 播放器的开发工作。Adobe 公司在 iOS 系统上从未得到机会，在 Android 系统上虽然有机会，但已经放弃了。另外，iOS 从未加入对 Flash 的支持，Android 4.4 版本就已经默认移除了对 Flash 的支持。正是这两方面的原因导致了手机网页看视频必须用到 HTML5。

7.5 本章小结

本章大体介绍了 HTML5 的历史、出现的原因，以及 HTML5 和 HTML4 之间的区别等内容。

通过回顾，可以很清晰地了解到 HTML4 规范存在的问题，从而更好地了解 HTML5 规范的优势。

本章的重点是 HTML5 和 HTML4 的区别，以及 HTML5 语法的变化。下面会详细讲述 HTML5 新增的元素、属性，以及废弃的元素、属性等相关内容。