

普.通.高.等.学.校
计算机教育“十三五”规划教材

HTML5+CSS3 基础开发教程

(第2版)

HTML5+CSS3 PROGRAMMING
(2nd edition)

张珈珣 范立锋 ◆ 编著



中国工信出版集团



人民邮电出版社
POSTS & TELECOM PRESS

普.通.高.等.学.校
计算机教育“十三五”规划教材

HTML5+CSS3 基础开发教程

(第2版)

HTML5+CSS3 PROGRAMMING
(2nd edition)

张珈珣 范立锋 ◆ 编著



人民邮电出版社

北京

图书在版编目 (C I P) 数据

HTML5+CSS3基础开发教程 / 张珈珣, 范立锋编著
- 2版. -- 北京: 人民邮电出版社, 2017.8
普通高等学校计算机教育“十三五”规划教材
ISBN 978-7-115-46040-0

I. ①H… II. ①张… ②范… III. ①超文本标记语言—程序设计—高等学校—教材②网页制作工具—高等学校—教材 IV. ①TP312.8②TP393.092.2

中国版本图书馆CIP数据核字(2017)第133686号

内 容 提 要

本书内容主要分为三大部分。第一部分介绍了 HTML5 的相关开发技术, 包括 HTML 的发展, HTML5 与之前版本的区别, HTML5 的表单元素应用, HTML5 的多媒体元素应用, HTML5 的图像及动画应用, HTML5 的元素拖曳技术, HTML5 的数据存储技术, HTML5 的离线应用及地理位置应用, 通过 HTML5 对本地文件进行操作, Web Worker 进行复杂任务调度, SSE 和 WebSocket 进行浏览器与服务器通读。第二部分介绍了 CSS3 的相关知识, 包括 CSS3 选择器的应用, 使用 CSS3 控制页面样式, 使用 CSS3 在页面中插入信息, 使用 CSS3 控制元素变形以及 CSS3 中的元素过渡应用。第三部分为 HTML5 与 CSS3 的综合应用实例, 介绍 HTML5+CSS+JS 进行移动端 Web 网页的响应式设计的核心代码。

本书主要面向高等院校学生, 以及没有开发经验或者仅有少量程序设计基础的读者, 因此书中所讲解内容较为基础、细致。书中每一章节都提供了上机实践环节, 使读者在理论学习的基础上结合实际操作, 力求使读者真正做到学以致用。

-
- ◆ 编 著 张珈珣 范立锋
责任编辑 刘 博
责任印制 陈 彝
 - ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路 11 号
邮编 100164 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
三河市海波印务有限公司印刷
 - ◆ 开本: 787×1092 1/16
印张: 16.5
字数: 433 千字
-

定价: 49.80 元

读者服务热线: (010)81055256 印装质量热线: (010)81055316

反盗版热线: (010)81055315

广告经营许可证: 京东工商广登字 20170147 号

HTML5 和 CSS3 是近年来 Web 应用开发技术中最为热门的两项新技术。HTML5 在原 HTML 版本的基础上,提出了大量创新的、实用的元素和规范。HTML5 不仅在传统的台式机 Web 应用方面可以发挥重要作用,对移动设备上的 Web 应用也提供了良好的支持。CSS3 在样式定义方面同样提出了大量新的元素,在丰富原有样式的基础上,使开发变得更加方便、快捷。

本书全面、翔实地介绍了 HTML5 各种基础知识及应用技巧,以及 CSS3 的基础应用和高级应用。通过对本书的学习,读者可以快速、全面地掌握使用 HTML5 和 CSS3 开发 Web 应用程序的方法,并且能够达到融会贯通、灵活运用之目的。

本书特点

本书的特点如下。

(1) 介绍最新的开发技术。HTML5 和 CSS3 是目前最新的两种网页开发技术,且发展前景良好,未来将被广泛应用。

(2) 教材知识体系结构合理。知识安排强调整体性和系统性,知识表达强调层次性和有序性,便于读者学习和理解。

(3) 理论与应用紧密结合。本书的每一章节在对相关知识点进行详细介绍之后,都会提供该技术点的具体应用示例,通过理论与实际相结合,读者能更好地掌握相关知识点。

(4) 提升读者综合应用能力。本书在每章(除个别章节外)最后都提供了一个综合应用本章知识点的上机实践,通过完成每章上机实践内容,读者不仅能够更加深入理解理论知识,还能检测自己对本章知识的综合应用能力。

(5) 语言通俗易懂,读者容易理解。书中采用程序结构、页面交互图、流程图、表格等多种方式,描述问题及解决问题的过程,使读者从多个角度来理解问题。

(6) 案例的实用性。本书中最后提供了两个具有代表性的案例:Web 游戏和 Web 网站,这两个应用分别代表了当前 HTML5 和 CSS3 的应用方向。通过这两个案例的练习,读者可以对本书所讲解的知识点进行整体回顾,提升读者应用 HTML5 和 CSS3 进行开发的水平。

本书结构

本书围绕 HTML5 及 CSS3 基础知识和高级应用进行讲解,书中提供了许多实例,而且每章后附有习题,用于巩固所学内容。全书共分 15 章,各章具体内容如下表所示。

表 全书各章的主要内容

章 名	描 述
第1章 初识 HTML5	介绍 HTML 发展过程以及 HTML5 学习前的准备工作
第2章 HTML5——全新的 HTML	通过与之前 HTML 对比，着重介绍 HTML5 的变化，包括新的语法结构及新的元素、属性等
第3章 HTML5 的表单	介绍 HTML5 新的 input 输入类型及属性，同时介绍了 HTML5 中表单的验证方式
第4章 HTML5 的多媒体	介绍如何应用 HTML5 的多媒体元素，包括多媒体元素的属性、方法以及事件的讲解和应用
第5章 HTML5 的图像及动画	介绍 HTML5 中绘图的方法，包括 canvas 元素应用，绘制简单图形、图像，以及对图形、图像的相关操作方法
第6章 HTML5 的元素拖曳	介绍 HTML5 中元素拖曳的实现方法
第7章 HTML5 的数据存储	介绍 HTML5 中数据存储的作用以及实现方法
第8章 HTML5 离线应用及地理位置应用	介绍 HTML5 中离线应用的原理、操作，以及地理位置的相关应用方法
第9章 文件系统	介绍 HTML5 中的 FileAPI 进行本地文件的操作方法
第10章 Web Worker	介绍 HTML5 通过 Web Worker 进行后台进程运行的方法
第11章 SSE 和 WebSocket	介绍 SSE 和 WebSocket 进行服务器向浏览器进行数据推送的方法
第12章 CSS3 入门与基础	介绍 CSS3 的基础知识，包括选择器的用法及控制页面样式的方法
第13章 CSS3 高级应用	介绍 CSS3 的高级应用，包括在页面中插入信息，控制页面文字样式，控制页面元素变形以及样式过渡的实现方法
第14章 综合案例	提供两个完整实例，一个为 Web 游戏应用，另一个为 Web 网站应用
第15章 移动应用前端开发	介绍 App 前端开发示例，通过 HTML5+CSS+JS 进行移动端 Web 网页的响应式设计核心代码

本书面向的读者

本书面向的是 HTML 程序设计的初学者。读者无需掌握任何开发技术就可以根据书中介绍的方法和实例，构建 HTML5 的 Web 应用。本书在内容编排上由浅入深，循序渐进，注重理论与实际相结合，特别适合高等院校的教师作为授课教材。

如果您具备一定的网页开发基础，但又希望掌握最新的网页开发技术，本书也非常适合。虽然本书的编写初衷是面向没有开发经验的读者，但是如果您具备以下方面的知识，学习起来将事半功倍：

- 熟悉 HTML；
- 熟悉 CSS；
- 熟悉 JavaScript。

本书实例的运行环境

由于目前浏览器对 HTML5 及 CSS3 的支持性有所不同，个别示例可能需要在不同浏览器上运行方能得到正确效果，因此建议读者安装 Firefox、Chrome 以及 Opera 浏览器。

技术支持

本书实例开发中用到的程序源代码，可以在“人邮教育社区（www.ryjiaoyu.com）”上免费下载，以供读者学习和使用。

编 者
2017 年 5 月

目 录

第 1 章 初识 HTML5.....1

1.1 HTML 发展史.....1
1.2 为什么要学习 HTML5.....2
1.3 HTML5 的开发环境.....2
1.4 浏览器对 HTML5 支持性检测.....3
小结.....6
习题.....6

第 2 章 HTML5——全新的 HTML.....7

2.1 新的语法结构.....7
2.2 新的页面架构.....8
2.3 元素的改变.....10
2.3.1 新增的元素.....10
2.3.2 停止使用的元素.....14
2.4 属性的改变.....15
2.4.1 新增的属性.....15
2.4.2 停止使用的属性.....16
2.4.3 全局属性.....16
小结.....18
习题.....18

第 3 章 HTML5 的表单.....19

3.1 新的 input 输入类型及属性.....19
3.1.1 新的 input 输入类型.....19
3.1.2 新的 input 公用属性.....25
3.2 表单的验证方式.....30
3.2.1 自动验证方式.....30
3.2.2 调用 checkValidity()方法实现验证.....32
3.2.3 自定义提示信息.....33
3.2.4 设置不验证.....33
3.3 上机实践——设计注册页面.....34
3.3.1 实践目的.....34

3.3.2 设计思路.....34
3.3.3 实现过程.....34
3.3.4 演示效果.....36
小结.....37
习题.....37

第 4 章 HTML5 的多媒体.....38

4.1 HTML5 的多媒体元素.....38
4.2 多媒体元素的属性.....38
4.2.1 autoplay 属性.....38
4.2.2 controls 属性.....39
4.2.3 error 属性.....39
4.2.4 poster 属性.....41
4.2.5 networkState 属性.....41
4.2.6 width 与 height 属性.....43
4.2.7 readyState 属性.....43
4.2.8 其他属性.....45
4.3 多媒体元素的方法.....45
4.3.1 多媒体支持性检测方法.....46
4.3.2 多媒体播放方法.....49
4.4 多媒体元素的事件.....50
4.4.1 事件捕捉方法.....50
4.4.2 支持的事件类型.....51
4.4.3 播放事件的应用.....52
4.5 上机实践——DIY 视频播放器.....53
4.5.1 实践目的.....53
4.5.2 设计思路.....53
4.5.3 实现过程.....53
4.5.4 演示效果.....56
小结.....57
习题.....57

第 5 章 HTML5 的图像及动画.....58

5.1 了解 canvas 元素.....58
5.1.1 canvas 的用法.....58

5.1.2 一个简单的 canvas 画图实例	59
5.2 使用路径画图	59
5.2.1 理解 canvas 的坐标系	60
5.2.2 使用 moveTo、lineTo 画线	60
5.2.3 使用 arc 方法画弧	61
5.2.4 绘制贝塞尔图形	64
5.3 图形操作	65
5.3.1 图形样式设置	65
5.3.2 渐变图形	66
5.3.3 图形坐标变换	69
5.3.4 图形组合处理	72
5.3.5 图形阴影	74
5.4 图像操作	75
5.4.1 绘制图像	76
5.4.2 图像平铺	77
5.4.3 图像剪裁	79
5.4.4 像素处理	81
5.5 canvas 其他操作	82
5.5.1 绘制文字	83
5.5.2 保存、恢复图形	84
5.6 制作动画	86
5.7 上机实践——绘制时钟	87
5.7.1 实践目的	87
5.7.2 设计思路	88
5.7.3 实现过程	88
5.7.4 演示效果	89
小结	90
习题	90
第 6 章 HTML5 的元素拖曳	91
6.1 曾经的拖曳解决方案	91
6.2 HTML5 中拖曳的实现方法	91
6.3 dataTransfer 对象	94
6.4 文件拖曳操作	95
6.5 上机实践——拖曳式点菜界面	97
6.5.1 实践目的	97
6.5.2 设计思路	97
6.5.3 实现过程	97
6.5.4 演示效果	99
小结	100

习题	101
----------	-----

第 7 章 HTML5 的数据存储

7.1 为什么需要数据存储	102
7.2 Web Storage	102
7.2.1 Web Storage 与 Cookie 的比较	103
7.2.2 Web Storage 的两种存储方式	103
7.2.3 localStorage 的多数据操作	105
7.3 Web SQL 数据库	107
7.3.1 创建数据库	107
7.3.2 Web SQL 的增删改查	108
7.4 上机实践——注册与登录	113
7.4.1 实践目的	113
7.4.2 设计思路	113
7.4.3 实现过程	113
7.4.4 演示效果	116
小结	118
习题	118

第 8 章 HTML5 离线应用及地理位置应用

8.1 离线应用	119
8.1.1 离线应用的工作原理	119
8.1.2 管理本地缓存	119
8.1.3 applicationCache 检测及更新缓存	120
8.1.4 检测在线状态	121
8.2 地理位置应用	121
8.2.1 Geolocation 的工作原理	121
8.2.2 获取当前地理位置	122
8.2.3 监视地理位置信息	123
8.2.4 停止获取当前地理位置信息	123
8.3 上机实践——在搜狗地图中定位	123
8.3.1 实践目的	123
8.3.2 设计思路	123
8.3.3 实现过程	123
8.3.4 演示效果	125
小结	126
习题	126

第 9 章 文件系统	127	第 12 章 CSS3 入门与基础	153
9.1 FileAPI 用途	127	12.1 CSS3 是什么	153
9.2 FileAPI 数据结构及接口标准	128	12.2 CSS3 的一个简单应用	153
9.3 核心代码示例	129	12.3 CSS3 的常用选择器	157
9.3.1 判断浏览器是否支持	129	12.3.1 为什么要使用选择器	157
9.3.2 获取本地文件	129	12.3.2 属性选择器	158
9.3.3 Blob 对象	132	12.3.3 类选择器	160
9.4 浏览器对 File API 的支持情况	133	12.3.4 伪类选择器	162
小结	133	12.4 控制页面样式	165
习题	133	12.4.1 控制圆角边框样式	166
第 10 章 Web Worker	134	12.4.2 控制背景样式	168
10.1 Web Worker 应用场景	134	12.4.3 控制颜色样式	173
10.2 如何使用 Web Worker	134	12.4.4 控制页面布局	176
10.3 核心代码示例	136	12.5 上机实践——购物车结算界面	179
10.4 Web Worker 访问对象的限制	137	12.5.1 实践目的	179
10.5 Web Worker 传递 JSON	137	12.5.2 设计思路	179
10.6 浏览器对 Web Worker 的支持情况	139	12.5.3 实现过程	179
小结	139	12.5.4 显示效果	181
习题	139	小结	181
第 11 章 SSE 和 WebSoceket	140	习题	181
11.1 关于数据推送	140	第 13 章 CSS3 高级应用	182
11.2 SSE 示例	141	13.1 在页面中插入内容	182
11.2.1 SSE 工作原理——客户端	141	13.1.1 插入文字	182
11.2.2 SSE 工作原理——服务端	143	13.1.2 插入图像	184
11.3 WebSocket 工作原理	143	13.1.3 插入项目编号	185
11.3.1 WebSocket 工作原理 ——客户端	144	13.2 文字样式控制	186
11.3.2 WebSocket 工作原理 ——服务端	145	13.2.1 为文字增加阴影效果	187
11.4 上机实践——使用 WebSocket 实现 聊天室	146	13.2.2 设置单词及网址自动换行	188
11.4.1 实践目的	146	13.2.3 使用服务器端字体	189
11.4.2 设计思路	146	13.3 元素变形处理	189
11.4.3 实现过程	146	13.3.1 缩放效果	190
11.4.4 演示效果	151	13.3.2 旋转效果	190
小结	151	13.3.3 移动效果	191
习题	152	13.3.4 倾斜效果	192
		13.4 样式过渡	193
		13.5 更为复杂的样式过渡	194
		13.6 上机实践——个性留言板	195
		13.6.1 实践目的	195

13.6.2 设计思路	195
13.6.3 实现过程	195
13.6.4 显示效果	198
小结	199
习题	199

第14章 综合案例 200

14.1 马里奥大逃亡游戏	200
14.1.1 游戏介绍	200
14.1.2 需求分析	200
14.1.3 详细设计	201
14.1.4 游戏效果	204
14.2 欧美风格企业网站	205

14.2.1 需求分析	205
14.2.2 概要设计	206
14.2.3 详细设计	207
14.2.4 网站效果	226

第15章 移动应用前端开发 230

15.1 引导页的设计	230
15.2 登录页的设计	231
15.3 注册页的设计	234
15.4 首页的设计	237
15.5 我的页面设计	241
15.6 新房源列表页的设计	243
15.7 新房内容页的设计	251

第 1 章

初识 HTML5

随着计算机硬件及网络环境的不断发展，基于网页形式的各种 Web 应用技术也层出不穷。在众多前端技术中，HTML5 作为新一代 Web 开发技术得到越来越多开发者的关注和应用。HTML5 的出现，使 Web 开发标准发生了质的飞跃，使原本死板保守的 Web 应用变得更加绚丽多彩、功能强大。更为重要的是，在移动互联网越来越发达的时代，HTML5 在这一领域已经占有一席之地。虽然目前还未最终确定 HTML5 的开发标准，但是 HTML5 已经发展成为一种复杂的网页设计及 Web 应用开发的重要平台，其跨平台性已经在移动端应用开发占据主流地位。

通过本章的学习，读者可以对 HTML 的发展历程以及 HTML5 的基本特性有一个大体了解，为后续内容的学习奠定基础。

1.1 HTML 发展史

在真正开始接触 HTML5 之前，我们有必要首先了解一下 HTML 的发展史。HTML 是随着网页技术的出现而诞生的，它的全称是 HyperText Markup Language，即超文本标记语言，主要用于描述网页文档结构。通俗地说，HTML 规定了一组由尖括号组成的能够提供各种功能的标签，通过不同标签的组合使用来构建页面。

HTML 从诞生至今，主要经历了几个比较关键的版本。

1. HTML 雏形诞生

1991 年，蒂姆·伯纳斯·李（Tim Berners-Lee）编写了一份叫作“HTML 标签”的文档，该文档包括了大约 20 个用来标记网页的 HTML 标签。这是一个非官方的版本，是 HTML 的雏形。

2. 第一个官方版本

HTML 的第一个官方版本是由 IETF（因特网工程任务组）推出的 HTML 2.0，在该版本问世之前，一些标签的功能已经被实现。

3. HTML 发展拐点

当 W3C（万维网联盟）取代 IETF 成为 HTML 的标准组织后，HTML 的版本被频繁修改。随着标签数量的增加，HTML 能够提供的功能也越来越完善。直到 1999 年的 HTML 4.01 版本，HTML 到达了它的第一个拐点，并被普遍应用。

4. XHTML 的没落

在 HTML 4.01 之后的版本变为 XHTML 1.0，其中 X 代表 eXtensible（扩展）。XHTML 1.0 与 HTML 4.01 相比，并未引入任何新的标签或属性，只是在语法上进行了严格的要求。例如，

HTML 4.01 允许使用大写或小写字母标识标记元素和属性，而 XHTML 则只允许小写字母。严格的语法规则带来的好处是统一的代码风格，这在一定程度上为 Web 开发者提供了便利。

然而在 XHTML 1 的后续版本 XHTML 2 却发生了很大的变化，该版本不再兼容之前的版本（甚至之前的 HTML 规范）。由于 HTML 4 已被普遍应用和接受，要完全放弃原有标准是不现实的，无论对于 Web 开发者还是浏览器制造商来说都是不可接受的。这也注定了 XHTML 2 逐步走向没落。

5. HTML5 的萌芽

W3C 组织于 2009 年宣布终止 XHTML 2 的开发进程，转向一种新的规范——HTML 5。非常有趣的是，W3C 是以 WHATWG 组织的研究成果为基础进行发展的，而 WHATWG 组织正是当年 W3C 的反对者联盟。W3C 组织的方向转变造成了这一现状，目前，同时有两个组织在制定自己的规范。

6. HTML5 为移动而生

HTML5 之所以能够迅速受到广大开发者的青睐应当归功于乔布斯发表的公开信《关于 FLASH 的几点思考》论文，其提出 HTML5 更适合移动开发的六点主要原因更是奠定了 HTML5 在跨平台的移动设备上赢得最终胜利。HTML5 改变 Web 开发的局限性，基于 HTML5 开发方便构建类似客户端软件的网页版 App，可以访问磁盘系统和摄像头等敏感设备，将原本桌面应用软件开发所擅长的领域带到 Web 开发领域，摒弃了 Web 开发的种种痛点，将 Web 开发带入了新的纪元。

1.2 为什么要学习 HTML5

虽然目前 HTML5 还没有形成一个统一的规范，但是这并不能成为我们学习 HTML5 的阻碍，可以说 HTML5 在不久的将来将逐步甚至完全取代以往的规范，成为 Web 开发的主流。

HTML5 提供的功能丰富的标签，可以满足 Web 应用多元化的需求；通过使用 HTML5 标签，开发人员可以轻松地在网页中实现音频和视频的嵌入、动画效果、渐变效果、表单自动验证等等。而这些在现有技术层面，都是需要第三方插件以及大量编码才能够实现的。

此外，HTML5 可以很好地支持移动互联网的 Web 应用需求。随着手机和平板电脑的硬件配置、智能化操作系统的不断升级，移动互联网已经逐步渗透到我们周围的每一个人。HTML5 自身对音频、视频、地理定位等功能的良好支持，直接决定了其在移动设备的 Web 应用、游戏方面大有可为。

目前流行的“云”技术，更使 HTML5 大放异彩。试想我们在任何一台可以使用浏览器的计算机或移动终端设备上，通过“云”能够便捷地获取我们需要的资料、信息而无需预先安装任何应用。这对用户来说，可以彻底摆脱操作环境的束缚，能够更加方便、快捷地实现信息传递。

1.3 HTML5 的开发环境

HTML5 更多时候被用在基于 Web 的页面及应用开发，较为常用的集成开发工具有 Adobe 公司的 Dreamweaver、Adobe Edge，Microsoft 公司的 Visual Studio 和 JetBrains 公司的 WebStorm 等。当然我们也可以使用纯文本编辑器来编写 HTML5 代码，常用的文本编辑工具有 UltraEdit、NotePad++、EditPlus 等。

目前国内主流的 Hybrid 跨平台移动应用开发采用的技术均为 HTML5、CSS3 和 JavaScript 语言结合开发；开发者通过内置的 IDE 集成开发工具、在线编译系统以及云端打包器等，快速生成 Android、iOS、Windows Phone 平台上的本地应用，其中比较典型的平台为 Appcon、APICloud、WEX5 等，建议读者熟读其平台对应的 API，这样对开发 Hybrid 应用能够起到事半功倍的效果。

集成工具与普通文本编辑器比较起来各有利弊，集成工具一般体积庞大，但是功能强大，多数集成工具都提供了代码提示、代码校验以及集成的调试环境；普通文本编辑器体积轻巧，但是往往只具有一般的编辑功能。读者可根据实际需求，选择适合自己的开发工具。

1.4 浏览器对 HTML5 支持性检测

HTML5 应用不需要额外的服务器支持，只要在客户端使用浏览器即可运行。目前，已经有多种浏览器对 HTML5 的部分功能提供了支持，例如，微软公司的 Internet Explorer 9 浏览器、Mozilla 的 Firefox 浏览器、Google 的 Chrome 浏览器，以及 Opera 浏览器等。然而各浏览器还并未提供对 HTML5 所有功能的完善支持，因此在应用 HTML5 之前，应该先进行当前浏览器对 HTML5 支持性的检测。

检测浏览器是否支持 HTML5 特性有多种方法，主要包括以下几种。

(1) 检测指定元素的 DOM 对象是否能被浏览器正确识别。例如，以检测 canvas 元素为例，创建一个 html 页面 testHTML5.html，并输入以下代码。

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />
<title></title>
</head>
<body>
    <canvas id="canvas1" style="background-color:#0000FF">
        如果浏览器不支持 HTML5 的<canvas>元素，则会显示此句话。
    </canvas>
</body>
</html>
```

在此页面中，我们使用了 canvas 元素创建一个画布，如果浏览器支持该元素，则会显示出一块背景颜色为蓝色的画布；如果浏览器不支持该元素，则会直接显示中间的语句。保存后，使用 IE6.0 浏览器打开此页面，得到结果如图 1-1 所示。

使用 Chrome 浏览器打开此页面，得到结果如图 1-2 所示。

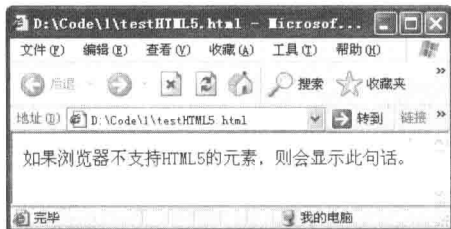


图 1-1 浏览器不支持 HTML5 的 canvas 元素



图 1-2 浏览器支持 HTML5 的 canvas 元素

(2) 检测指定元素是否拥有特定的方法, 并调用指定方法检查返回值。仍以 canvas 元素为例, 该元素使用方法 `getContext()` 获取该元素上下文。创建一个 HTML 页面 `testHTML5_2.html`, 并输入如下代码。

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />
<title></title>
<script>
function checkCanvasSupport()
{
    var canvas = document.getElementById('canvas1');
    alert(canvas.getContext());
}
</script>
</head>
<body onload="checkCanvasSupport()">
    <canvas id="canvas1" style="background-color:#0000FF">
        如果浏览器不支持 HTML5 的<canvas>元素, 则会显示此句话。
    </canvas>
</body>
</html>
```

在此页面中, 我们通过编写 JavaScript 代码获取 canvas 元素的 DOM 对象, 并调用 `alert` 方法显示 canvas 元素的 `getContext()` 方法。保存后, 使用 IE6.0 浏览器打开此页面, 得到结果如图 1-3 所示。

使用 Chrome 浏览器打开此页面, 得到结果如图 1-4 所示。



图 1-3 浏览器不支持 canvas 元素的 `getContext` 方法

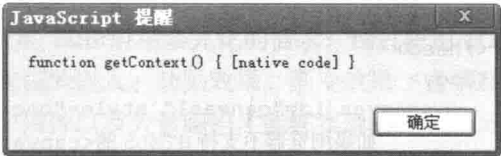


图 1-4 浏览器支持 canvas 元素的 `getContext` 方法

(3) 检测全局对象是否拥有特定的属性。例如以检测全局对象 `navigator` 的 `geolocation` 属性为例, 创建一个 HTML 页面 `testHTML5_3.html`, 输入如下代码。

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />
<title></title>
<script>
function checkNavigatorSupport()
{
    alert(navigator.geolocation);
}
</script>
</head>
<body onload="checkNavigatorSupport()">
```

```
</body>
</html>
```

在此页面中, 我们通过编写 JavaScript 代码, 使用 alert 方法显示 navigator 元素的 geolocation 属性。保存后, 使用 IE6.0 浏览器打开此页面, 得到结果如图 1-5 所示。



图 1-5 浏览器不支持 navigator 的 geolocation 属性

使用 Chrome 浏览器打开此页面, 得到结果如图 1-6 所示。

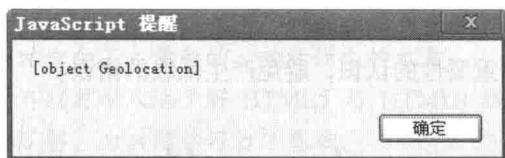


图 1-6 浏览器支持 navigator 的 geolocation 属性

(4) 使用 HTML 特性检测工具。Modernizr 是一个开源的 JavaScript 类库, 主要用于检测浏览器是否支持 HTML5 的新特性。我们可以在 Modernizr 的官方网站 <http://modernizr.com> 下载 Modernizr 的最新版本。下载后得到一个 JavaScript 文件, 在页面中引用该文件, 即可进行相关检测。例如, 创建一个 HTML 页面 testHTML5_4.html, 并输入以下代码。

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />
<title></title>
<script src="modernizr.custom.02219.js"></script>
<script>
function testHTML5()
{
    if(Modernizr.video)
    {
        alert('支持 video 元素')
    }
    else
    {
        alert('不支持 video 元素')
    }
}
</script>
</head>
<body onload="testHTML5()">
</body>
</html>
```

在此页面中, 我们调用 Modernizr 对 HTML5 的 video 元素进行检测, 如果浏览器支持该元素

则 Modernizr.video 返回 true，否则返回 false。保存后，使用 IE6.0 浏览器打开此页面，得到结果如图 1-7 所示。

使用 Chrome 浏览器打开此页面，得到结果如图 1-8 所示。



图 1-7 Modernizr 检测浏览器不支持 video 元素

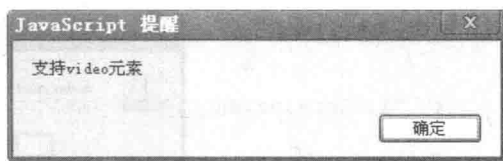


图 1-8 Modernizr 检测浏览器支持 video 元素

对于上述介绍的几种检测方法，读者应熟练掌握。因为 HTML5 还处于发展阶段，而各浏览器厂商对 HTML5 的支持性也在不断进步。对于某些新特性，如果在使用中没有达到预期的效果，不一定是因为错误地使用了 HTML5 的新特性，也有可能是因为当前浏览器没有提供此特性的支持。读者需提高对检测机制重要性的认识，避免产生低级的错误。

小 结

HTML 从其诞生至今一直在不断地发展、完善，而 HTML5 更是 HTML 发展史上一个具有划时代意义的版本。HTML5 则是第一个将 Web 作为应用开发平台的 HTML 语言，熟悉和掌握 HTML5 开发技巧，在今后的求职、就业中，也将成为一大制胜的法宝。

习 题

- (1) HTML 发展过程中经历了哪几个重要版本？
- (2) 请给出 3 个以上 HTML5 的开发工具。
- (3) 检测浏览器对 HTML5 支持的方法有哪几种？
- (4) 如何使用 Modernizr 工具检测当前浏览器是否支持 audio 元素？

第 2 章

HTML5——全新的 HTML

HTML5 是 HTML 诞生至今最具有划时代意义的一个版本，它在之前的 HTML 版本基础上，做出了大量更新。HTML5 除了保留了 HTML4 中一些基本元素及属性的用法外，还删除了部分利用率低或不合理的元素，同时增加了大量新的、功能强大的元素。

通过本章的学习，读者可以更深入地了解 HTML5 与 HTML4 版本的区别，同时对 HTML5 的新元素、新语法有个初步了解，为后续学习打下基础。

2.1 新的语法结构

在了解 HTML5 的新语法结构之前，我们先来看一个例子。同样一个网页，在 HTML4 中编写的代码如下。

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />
<title>HTML4</title>
</head>
<body>
<p>这是一个 HTML 页面</p>
</body>
</html>
```

在 HTML5 中编写的代码如下。

```
<!DOCTYPE html>
<html>
<head>
<meta charset="gb2312">
<title>HTML5</title>
</head>
<body>
<p>这是一个 HTML 页面</p>
</body>
</html>
```

分析这两段代码的详细结构不难发现，与 HTML4 的语法结构相比，HTML5 的语法结构更加简练，省去了一些不必要的配置信息。

最初的 HTML 版本借用了标准通用置标语言(Standard Generalized Markup Language, SGML) 的标记规范, 并且在后续的版本中一直遵循着这一规范。但是 SGML 的语法非常复杂, 想要开发出一款完美解析 SGML 的程序, 无疑是一件非常困难的事情。目前大多数浏览器都不提供 SGML 的解析功能, 相同的 HTML 代码在不同的浏览器中执行, 结果也会有所区别。

针对 HTML4 对于各浏览器兼容性的问题, 开发者们也曾经想出过一些解决方案, 例如针对不同浏览器编写不同代码片段, 程序会根据不同浏览器执行环境, 选择合适的代码段就能解析。虽然通过某些手段可以在一定程度上解决不同浏览器之间的兼容性问题, 但是这对开发者来说, 不仅增加了工作量及工作难度, 最重要的是始终未能从根本上解决这一问题。HTML5 的一个目标就是消除不同浏览器的兼容性问题, 通过制定统一标准, 保证相同代码在不同浏览器上执行, 都能够按照同一标准解析, 产生相同的结果。

2.2 新的页面架构

如果读者有过 HTML 的开发经验, 对目前的页面架构应该不会陌生。无论是简单的页面还是复杂的页面, 都可以被分割为几个不同的区域, 用于放置不同的信息。在 HTML4 中要想实现这一功能, 目前多数开发者都是使用 div 元素来实现的。例如在 HTML4 中, 一个常见的分块页面代码如下。

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />
<style type="text/css">
#sidebar{float:left;width:20%}
.main{float:right;width:80%}
#footer{clear:both}
</style>
<title></title>
</head>
<body>
  <div id="header">
    <p>网站标题</p>
  </div>
  <div id="sidebar">
    <ul>
      <li>菜单 1</li>
      <li>菜单 2</li>
      <li>菜单 3</li>
    </ul>
  </div>
  <div class="main">
    <p>主体内容 1</p>
  </div>
  <div class="main">
    <p>主体内容 2</p>
  </div>
```