

HTML 5+CSS 3+JavaScript 网页设计与制作全程揭秘

杨阳 编著

1350个素材源文件

178个实战案例

760分钟视频教程



Book features



内容全面，循序渐进

遵循从基础到高级，从入门到精通的思路，帮助读者轻松建立起正确的知识体系。



技术专业，易于操作

在技术讲解的同时，配合丰富的操作练习，理论与操作相结合，使读者轻松理解和掌握知识点。



专业作者，如临亲授

作者来自于一线专业公司和高校教师，实践与教学的融合，更清楚学习者的知识需求。



扫码微课，碎片学习

读者可以扫描书中的二维码，利用零碎的时间随时随地通过视频课程进行学习。



源于实践，回归实战

穿插大量的实战操作案例，均来源于实践，通过一步步操作的方式再现给读者。



海量资源，辅助学习

提供立体化的学习资源：所有案例的素材、源文件；多媒体语音视频教程；与教学配套的PPT课件。

KEYWORD

专业学习手册+完全配套资源+PPT课件+6部网页设计电子书+海量辅助设计资源+作者技术交流群

SUBSCRIBE

清华大学出版社

HTML5+CSS3+ JavaScript 网页设计与 制作全程揭秘

杨阳 编著

清华大学出版社

北 京

内 容 简 介

Web 标准是所有网页前台技术的发展方向, 本书学习的三大技术是 Web 标准的主要组成部分。本书全面、系统地介绍了使用 HTML5、CSS3 和 JavaScript 进行网页设计制作各方面的内容和技巧。

本书内容简洁、通俗易懂, 通过知识点与案例相结合的方式, 让读者能够清晰明了地理解书中的相关技术内容, 从而达到理想的学习效果。全书共分 17 章, 从初学者的角度出发, 全面讲解了 HTML5、CSS3 和 JavaScript 的相关知识, 其中包括认识 HTML 与 HTML5、HTML 主体标签、文字与图片标签的应用、超链接与表格标签的应用、多媒体标签的应用、表单标签的应用、HTML5 中 <canvas> 标签的应用、HTML5 文档结构标签的应用、CSS 样式基础、CSS 布局与定位方式、CSS 基础属性详解、CSS3 属性详解、使用 CSS3 实现动画效果、JavaScript 基础、JavaScript 中的函数与对象、JavaScript 中的事件和 JavaScript 综合应用案例等内容。

本书结构清晰、实例经典、技术实用, 适合 Web 前端开发、网页设计、网页制作、网站建设等行业人员阅读和参考, 也可供网页设计爱好者自学使用, 同时还可作为高等院校网页设计与制作课程的教材, 以及网页平面设计的培训教材。

本书封面贴有清华大学出版社防伪标签, 无标签者不得销售。

版权所有, 侵权必究。侵权举报电话: 010-62782989 13701121933

图书在版编目 (CIP) 数据

HTML 5+CSS 3+JavaScript 网页设计与制作全程揭秘 / 杨阳 编著. — 北京: 清华大学出版社, 2019

(网页设计与开发殿堂之路)

ISBN 978-7-302-52695-7

I . ① H… II . ① 杨… III . ① 超文本标记语言—程序设计 ② 网页制作工具 ③ JAVA 语言—程序设计
IV . ① TP312.8 ② TP393.092.2

中国版本图书馆 CIP 数据核字 (2019) 第 057421 号

责任编辑: 李 磊 焦昭君

封面设计: 王 晨

版式设计: 思创景点

责任校对: 牛艳敏

责任印制:

出版发行: 清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址: 北京清华大学学研大厦A座 邮 编: 100084

社 总 机: 010-62770175 邮 购: 010-62786544

投稿与读者服务: 010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈: 010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者:

装 订 者:

经 销: 全国新华书店

开 本: 185mm×260mm 印 张: 21.25 字 数: 614千字

版 次: 2019年8月第1版 印 次: 2019年8月第1次印刷

定 价: 69.80元

产品编号: 077882-01



前言

随着互联网信息技术的发展，网页设计与制作技术已经成为一些网络用户必备的知识技能。现如今，大部分制作网页的方式都是运用可视化网页编辑软件，这些软件的功能相当强大。但对于高级的网页制作人员而言，仍然需要掌握 HTML、CSS 和 JavaScript 等网页设计语言和技术，从而更好地设计自己所想象的页面，以实现一般网页设计软件所不能实现的许多重要功能。

本书要求读者边学习边实践，几乎每个重要的知识点都有实例练习供读者参考，避免学习知识的表面仅限于理论。技术学习的关键是方法，本书在很多实例中体现了方法的重要性，读者只要掌握了各种技术的运用方法，在学习更深入的知识时，可大大提高学习的效率。

本书内容丰富、条理清晰，全面介绍 HTML5、CSS3 和 JavaScript 的相关知识，每个知识点都配有实例，让读者真正做到学以致用。全书共分 17 章，各章内容介绍如下。

第 1 章 认识 HTML 与 HTML5，本章主要介绍网站建设的常用软件和技术，重点讲解 HTML 与 HTML5 的相关基础知识，以及它们之间的关系。通过本章内容的学习，读者可以了解 HTML5 新增的标签。

第 2 章 HTML 主体标签，本章主要介绍如何在 HTML 中对网页头部的 <head> 标签和网页主体 <body> 标签进行设置，从而达到控制网页整体属性的目的。

第 3 章 文字与图片标签的应用，文字与图片是网页中不可或缺的基础元素。本章主要介绍在 HTML 中对文字、段落、列表和图像进行设置的相关标签和属性，这些都是对网页基础元素的控制，读者需要细心体会，认真掌握。

第 4 章 超链接与表格标签的应用，本章主要介绍 HTML 中超链接标签的应用及各种超链接的创建和设置方法，并且还介绍了表格标签的应用，以及数据表格的创建方法。

第 5 章 多媒体标签的应用，本章主要介绍如何在网页中应用各种不同类型的多媒体元素，包括 HTML5 新增的 <video> 与 <audio> 标签的使用和属性设置方法。

第 6 章 表单标签的应用，表单是网页交互的重要途径。本章主要介绍 HTML 页面中各种传统的表单元素及其属性设置，以及 HTML5 新增的表单元素和属性，并且通过实例的制作，使读者能够快速掌握 HTML 表单页面的制作。

第 7 章 HTML5 中 <canvas> 标签的应用，canvas 元素是 HTML5 的亮点之一，通过使用 canvas 元素可以在网页中绘制出各种几何图形。本章详细介绍使用 HTML5 中的 canvas 元素在网页中绘制各种图形、文字的方法。

第 8 章 HTML5 文档结构标签的应用，本章介绍 HTML5 中新增的文档结构标签和语义模块标签，并通过实例的制作，使读者能够理解并掌握 HTML5 结构标签的作用和使用方法。

第 9 章 CSS 样式基础，本章主要介绍有关 CSS 样式的基础知识，包括 CSS 语法的语法、CSS 选择器及应用 CSS 样式的方法。

第 10 章 CSS 布局与定位方式，本章主要介绍有关 CSS 基础盒模型的知识，以及 HTML 页面中的元素定位属性，使读者掌握使用 CSS 样式进行网页布局制作的方法。



第 11 章 CSS 基础属性详解,本章主要介绍通过 CSS 样式对网页中各种元素进行设置的方法,包括文字、段落、背景、列表、边框、超链接伪类等。通过本章内容的学习,读者可以使用 CSS 样式对网页中各种元素的外观进行控制。

第 12 章 CSS3 属性详解,本章分类向读者介绍 CSS3 新增的属性,以及各种属性的使用和设置方法,使读者能够实现很多特殊的效果。

第 13 章 使用 CSS3 实现动画效果,动画效果是 CSS3 最突出的亮点,通过对相关 CSS 属性的设置,即可轻松在网页中实现各种动画效果。本章详细介绍 CSS3 各种动画效果的制作方法和技巧。

第 14 章 JavaScript 基础,本章介绍有关 JavaScript 的基础知识,包括 JavaScript 在网页中的作用、JavaScript 语法基础、变量、数据类型、常用运算符、条件与循环语句等内容,使读者深入地了解 JavaScript。

第 15 章 JavaScript 中的函数与对象,本章介绍 JavaScript 中函数与对象的概念,以及在 JavaScript 中如何使用函数,并且还介绍了 JavaScript 中的内置对象和浏览器对象的使用方法。

第 16 章 JavaScript 中的事件,本章介绍 JavaScript 事件的概念和常用事件,并且通过实例练习的形式介绍常用 HTML 事件的使用方法和技巧。

第 17 章 JavaScript 综合应用案例,本章列举了多个在网页中常见的 JavaScript 特效案例进行分析讲解,使读者能够快速掌握网页中常见特效的实现方法。

本书由杨阳编著,另外张晓景、李晓斌、高鹏、胡敏敏、张国勇、贾勇、林秋、胡卫东、姜玉声、周晓丽、郭慧等人也参与了部分编写工作。本书在写作过程中力求严谨,由于作者水平所限,书中难免有疏漏和不足之处,希望广大读者批评、指正,欢迎与我们沟通和交流。QQ 群名称:网页设计与开发交流群;QQ 群号:705894157。

为了方便读者学习,本书为每个实例提供了教学视频,只要扫描一下书中实例名称旁边的二维码,即可直接打开视频进行观看,或者推送到自己的邮箱中下载后进行观看。本书配套的立体化学习资源中提供了书中所有实例的素材源文件、最终文件、教学视频和 PPT 课件,并附赠海量实用资源。读者在学习时可扫描下面的二维码,然后将内容推送到自己的邮箱中,即可下载获取相应的资源(注意:请将这几个二维码下的压缩文件全部下载完毕后,再进行解压,即可得到完整的文件内容)。



编 者

Search

目录



第1章 认识 HTML 与 HTML5

1.1 了解动态网站的开发技术	1
1.1.1 网页编辑软件 Dreamweaver	1
1.1.2 网页标记语言 HTML	2
1.1.3 网页表现语言 CSS	2
1.1.4 网页特效脚本语言 JavaScript	2
1.1.5 动态网页编程语言 ASP、PHP 和 JSP 等	3
1.2 HTML 基础	3
1.2.1 什么是 HTML	3
1.2.2 HTML 的主要功能	4
1.2.3 HTML 的编辑环境	4
1.2.4 认识 Dreamweaver	5
实战 制作第一个 HTML 页面	6
1.3 HTML5 基础	7
1.3.1 HTML5 概述	7
1.3.2 HTML5 的优势	8
1.4 认识 HTML5	9
1.4.1 HTML5 的文档结构	9
1.4.2 HTML5 的基本语法	9
1.4.3 HTML5 精简的头部	10
1.5 HTML5 中新增的标签	10
1.5.1 结构标签	11
1.5.2 文本标签	11
1.5.3 应用和辅助标签	11
1.5.4 进度标签	11
1.5.5 交互性标签	11
1.5.6 在文档和应用中使用的标签	12
1.5.7 <ruby> 标签	12
1.5.8 其他标签	12
1.6 HTML5 中废弃的标签	12
1.7 HTML、CSS 与 JavaScript 的结合	13

第2章 HTML 主体标签

2.1 HTML 头部 <head> 标签设置	14
2.1.1 <title> 标签	14
实战 设置网页标题	14
2.1.2 <base> 标签	15
实战 设置网页基底网址	16
2.1.3 <meta> 标签	16
实战 设置网页关键字、说明以及页面定时跳转	18
2.2 HTML 主体 <body> 标签设置	19
2.2.1 边距属性 margin	19
实战 设置网页整体内容边距	20
2.2.2 背景颜色属性 bgcolor	21
实战 设置网页背景颜色	21
2.2.3 背景图像属性 background	21
实战 为网页主体添加背景图像	22
2.2.4 文字属性 text	22
实战 设置网页文字默认颜色	23
2.2.5 默认链接属性 link	23
实战 设置网页中超链接文字的默认颜色	24
2.3 在 HTML 代码中添加注释	25
2.4 在 HTML 中调用外部程序	25
2.4.1 调用外部 JavaScript 程序	25
实战 在网页中显示当前系统时间	25
2.4.2 区分客户端与服务器端程序	27

第3章 文字与图片标签的应用

3.1 文字修饰	28
3.1.1 文字样式 标签	28
实战 设置文字样式	28
3.1.2 文字倾斜 <i> 和 标签	29
实战 设置倾斜文字效果	30

5.5 <audio> 与 <video> 标签的属性	70
5.5.1 元素的标签属性	70
5.5.2 元素的接口属性	71
5.6 <audio> 与 <video> 标签的接口方法与事件	72
5.6.1 元素的接口方法	72
5.6.2 元素的事件	72
实战 自定义播放控制组件	73

第 6 章 表单标签的应用

6.1 了解 HTML 表单	77
6.1.1 表单的作用	77
6.1.2 表单 <form> 标签	77
6.1.3 表单的数据传递方式	78
6.2 普通的 HTML 表单元素	78
6.2.1 文本域	78
6.2.2 密码域	78
6.2.3 文本区域	79
6.2.4 隐藏域	79
6.2.5 复选框	79
6.2.6 单选按钮	79
6.2.7 选择域	80
6.2.8 文件域	80
6.2.9 按钮	80
6.2.10 图像域	80
实战 制作登录表单	81
6.3 HTML5 新增表单输入类型	83
6.3.1 url 类型	83
6.3.2 email 类型	83
6.3.3 range 类型	83
6.3.4 number 类型	83
6.3.5 tel 类型	84
6.3.6 search 类型	84
6.3.7 color 类型	84
6.3.8 date 类型	84
6.3.9 month、week、time、datetime、 datetime-local 类型	85
实战 制作留言表单页面	85
6.3.10 浏览器对 HTML5 表单的支持情况	88
6.4 HTML5 新增表单属性	88
6.4.1 form 属性	88
6.4.2 formaction 属性	89
6.4.3 formmethod、formenctype、 formnovalidate、formtarget 属性	89
6.4.4 placeholder 属性	89
6.4.5 autofocus 属性	89

实战 为表单元素设置默认提示内容	90
6.4.6 autocomplete 属性	90
6.5 使用 HTML5 表单验证	91
6.5.1 用于实现验证的表单元素属性	91
实战 验证网页表单元素	91
6.5.2 表单验证方法	92
6.5.3 表单验证事件	93

第 7 章 HTML5 中 <canvas> 标签的应用

7.1 <canvas> 标签	95
7.1.1 了解 <canvas> 标签	95
7.1.2 在网页中使用 <canvas> 标签	95
7.1.3 使用 <canvas> 标签实现绘图的过程	96
7.2 绘制基本图形	97
7.2.1 绘制直线	97
实战 在网页中绘制直线	97
7.2.2 绘制矩形	98
实战 在网页中绘制矩形	99
7.2.3 绘制圆形	100
实战 在网页中绘制圆形	101
7.2.4 绘制三角形	101
实战 在网页中绘制三角形	102
7.2.5 图形组合	102
7.3 绘制文本	104
7.3.1 使用文本	104
实战 在网页中绘制文字	104
7.3.2 创建对象阴影	105
实战 为网页中所绘制的文字添加阴影	106
7.4 在网页中绘制特殊形状图像	106
7.4.1 绘制图像	106
实战 在网页中绘制图像	107
7.4.2 裁切区域	108
实战 在网页中实现圆形图像效果	108

第 8 章 HTML5 文档结构标签的应用

8.1 构建 HTML5 页面主体内容	111
8.1.1 文章 <article> 标签	111
8.1.2 章节 <section> 标签	113
8.1.3 导航 <nav> 标签	114
8.1.4 辅助内容 <aside> 标签	115
8.1.5 日期时间 <time> 标签	116
8.2 HTML5 文档中的语义模块标签	116
8.2.1 标题 <header> 标签	116
8.2.2 标题分组 <hgroup> 标签	117
8.2.3 页脚 <footer> 标签	118



8.2.4 联系信息 <address> 标签	118
8.3 制作 HTML5 文章页面	119
实战 制作 HTML5 文章页面	119

第9章 CSS 样式基础

9.1 了解 CSS 样式	123
9.1.1 什么是 CSS 样式	123
9.1.2 CSS 样式的发展	123
9.2 CSS 样式语法	124
9.2.1 CSS 样式基本语法	124
9.2.2 CSS 样式规则	125
9.3 创建和使用 CSS 选择器	125
9.3.1 通配符选择器	125
实战 使用通配符选择器控制网页中的所有 标签	126
9.3.2 标签选择器	126
实战 使用标签选择器设置网页整体样式	127
9.3.3 ID 选择器	127
实战 创建和使用 ID CSS 样式	128
9.3.4 类选择器	129
实战 创建和使用类 CSS 样式	129
9.3.5 伪类选择器	130
实战 设置网页中超链接伪类样式	131
9.3.6 派生选择器	132
实战 创建派生选择器样式	133
9.3.7 群组选择器	133
实战 使用群组选择器同时定义多个网页元素 样式	134
9.4 在网页中应用 CSS 样式的 4 种 方式	135
9.4.1 内联 CSS 样式	135
实战 创建并应用内联 CSS 样式	135
9.4.2 内部 CSS 样式	136
实战 创建并应用内部 CSS 样式	136
9.4.3 外部 CSS 样式	137
实战 创建并链接外部 CSS 样式表 文件	137
9.4.4 导入外部 CSS 样式	138
实战 导入外部 CSS 样式表文件	139
9.5 CSS 样式的特性	139
9.5.1 CSS 样式的继承性	139
9.5.2 CSS 样式的特殊性	139
9.5.3 CSS 样式的层叠性	140
9.5.4 CSS 样式的重要性	140

第10章 CSS 布局与定位方式

10.1 创建 Div	142
10.1.1 了解 Div	142
10.1.2 如何插入 Div	142
10.1.3 块元素与行内元素	144
10.2 CSS 基础盒模型	145
10.2.1 CSS 基础盒模型概述	145
10.2.2 margin 属性——元素边距	146
10.2.3 border 属性——元素边框	146
10.2.4 padding 属性——元素填充	146
实战 设置网页元素的盒模型相关属性	147
10.3 网页元素的定位方式	148
10.3.1 CSS 定位属性	148
10.3.2 相对定位 relative	149
实战 实现网页元素的叠加显示	149
10.3.3 绝对定位 absolute	150
实战 网页元素固定在右侧显示	150
10.3.4 固定定位 fixed	151
实战 实现固定位置的顶部 Logo	151
10.3.5 浮动定位	152
实战 制作顺序排列的图像列表	152
10.3.6 空白边叠加	155
实战 网页中空白边叠加的应用	155
10.4 常见的网页布局方式	156
10.4.1 居中的布局	156
10.4.2 浮动的布局	157
10.4.3 自适应高度的解决方法	161

第11章 CSS 基础属性详解

11.1 使用 CSS 设置文字样式	162
11.1.1 font-family 属性——字体	162
11.1.2 font-size 属性——字体大小	162
11.1.3 color 属性——文字颜色	163
实战 设置网页文字基本效果	163
11.1.4 font-weight 属性——字体粗细	164
实战 设置网页文字加粗效果	165
11.1.5 font-style 属性——字体样式	165
实战 设置网页文字倾斜效果	166
11.1.6 text-transform 属性——英文字体 大小写	166
实战 设置网页中英文字体大小写	167
11.1.7 text-decoration 属性——文字 修饰	168

实战 为网页中的文字添加下画线、顶画线和删除线效果	168	实战 制作复杂的新闻列表	188
11.1.8 letter-spacing 属性——字符间距	169	11.6 超链接 CSS 样式伪类	189
实战 设置网页中文字的字符间距	169	11.6.1 :link 伪类	189
11.2 使用 CSS 设置段落样式	170	11.6.2 :hover 伪类	190
11.2.1 line-height 属性——行间距	170	11.6.3 :active 伪类	190
11.2.2 text-indent 属性——段落首行缩进	170	11.6.4 :visited 伪类	190
实战 美化网页中的段落文本	170	实战 设置网页中超链接文字效果	191
11.2.3 段落首字下沉	171	11.6.5 按钮式超链接	193
实战 在网页中实现段落文字首字下沉	172	实战 制作网站导航菜单	193
11.2.4 text-align 属性——水平对齐方式	172	11.7 cursor 属性——光标指针效果	194
实战 设置网页文本水平对齐效果	173	实战 在网页中实现多种光标指针效果	195
11.2.5 vertical-align 属性——垂直对齐方式	174		
实战 设置网页文本垂直对齐效果	174	第 12 章 CSS3 属性详解	
11.3 使用 CSS 设置背景颜色和背景图像	175	12.1 了解 CSS3	196
11.3.1 background-color 属性——背景颜色	176	12.1.1 CSS3 的发展	196
实战 设置网页的背景颜色	176	12.1.2 浏览器对 CSS3 的支持情况	196
11.3.2 background-image 属性——背景图像	176	12.1.3 了解 CSS3 的全新功能	197
11.3.3 background-repeat 属性——背景图像重复方式	177	12.2 CSS3 新增选择器	197
实战 设置网页背景图像及其重复方式	177	12.2.1 属性选择器	197
11.3.4 background-position 属性——背景图像位置	179	实战 在网页中使用属性选择器	198
实战 定位网页中的背景图像	179	12.2.2 结构伪类选择器	199
11.3.5 background-attachment 属性——固定背景图像	180	12.2.3 UI 元素状态伪类选择器	200
实战 设置网页中的背景图像固定不动	180	12.2.4 伪元素选择器	200
11.4 使用 CSS 设置边框与图片缩放样式	181	实战 在网页中使用伪元素选择器	201
11.4.1 border 属性——边框	181	12.3 新增的颜色定义方法	202
实战 为网页元素添加边框效果	182	12.3.1 RGBA 颜色定义方法	202
11.4.2 图片缩放	184	实战 使用 RGBA 方式设置半透明颜色	202
实战 设置网页中图片的等比例缩放效果	184	12.3.2 HSL 颜色定义方法	203
11.5 使用 CSS 设置列表样式	185	12.3.3 HSLA 颜色定义方法	203
11.5.1 list-style-type 属性——设置列表符号	185	实战 使用 HSL 和 HSLA 方式定义网页元素的背景颜色	203
实战 设置新闻列表效果	186	12.4 新增的文字设置属性	204
11.5.2 list-type-image 属性——自定义列表符号	187	12.4.1 text-shadow 属性——文字阴影	204
实战 使用自定义图像作为列表符号	187	实战 为网页文字添加阴影效果	205
11.5.3 设置定义列表样式	188	12.4.2 text-overflow 属性——文本溢出	206
		实战 设置文字溢出处理方式	206
		12.4.3 word-wrap 属性	207
		实战 设置长文本内容换行	207
		12.4.4 word-break 和 word-space 属性	208
		实战 设置文字内容换行处理方式	209
		12.4.5 @font-face 规则——嵌入 Web 字体	210
		实战 在网页中实现特殊字体效果	210
		12.5 新增的背景设置属性	212



12.5.1	background-size 属性——背景图像大小	212
实战	实现始终全屏显示的网页背景	212
12.5.2	background-origin 属性——背景图像原点位置	215
实战	设置背景图像显示原点位置	215
12.5.3	background-clip 属性——背景图像裁剪区域	216
实战	设置元素背景图像的显示区域	217
12.5.4	background 属性——多背景图像	217
实战	为网页设置多背景图像效果	218
12.6	新增的边框设置属性	219
12.6.1	border-colors 属性——多种颜色边框	219
实战	实现多彩绚丽的边框效果	219
12.6.2	border-radius 属性——圆角边框	220
实战	为网页元素设置圆角效果	221
12.6.3	border-image 属性——图像边框	222
实战	为网页元素设置图像边框效果	222
12.6.4	box-shadow 属性——元素阴影	223
实战	为网页元素添加阴影效果	224
12.7	新增的多列布局属性	225
12.7.1	columns 属性——多列布局	225
实战	快速对网页内容分列	225
12.7.2	column-width 属性——列宽度	226
12.7.3	column-count 属性——列数	226
12.7.4	column-gap 属性——列间距	227
12.7.5	column-rule 属性——列分隔线	227
12.7.6	column-span 属性——横跨列	228
实战	设置网页内容的分列布局效果	228
12.8	新增的盒模型设置属性	230
12.8.1	opacity 属性——元素不透明度	230
实战	设置网页元素的不透明度	230
12.8.2	overflow-x 和 overflow-y 属性——溢出内容处理方式	231
实战	设置网页元素内容溢出的处理方式	232
12.8.3	resize 属性——自由缩放	233
实战	实现网页元素尺寸任意拖动缩放	233
12.8.4	outline 属性——轮廓外边框	234
实战	为网页元素添加轮廓外边框效果	235
12.8.5	content 属性——赋予内容	236
实战	为网页元素赋予文字内容	237

第 13 章 使用 CSS3 实现动画效果

13.1	实现元素变形	238
13.1.1	transform 属性	238
13.1.2	使用 rotate() 函数实现元素旋转	238
实战	实现网页元素的旋转变形效果	239
13.1.3	使用 scale() 函数实现元素缩放和翻转变形	239
实战	实现网页元素的缩放和翻转效果	240
13.1.4	使用 translate() 函数实现元素移动	240
实战	实现网页元素的移动效果	241
13.1.5	使用 skew() 函数实现元素倾斜	241
实战	实现网页元素的倾斜效果	242
13.1.6	使用 matrix() 函数实现元素矩阵变形	242
实战	实现网页元素的矩阵变形效果	243
13.1.7	定义变形中心点	244
实战	设置网页元素的变形中心点位置	244
13.1.8	同时使用多个变形函数	245
实战	为网页元素同时应用多种变形效果	245
13.2	CSS3 实现过渡动画效果	246
13.2.1	transition 属性	246
13.2.2	transition-property 属性——实现过渡效果	247
实战	制作网页元素旋转并放大动画效果	247
13.2.3	transition-duration 属性——设置过渡时间	248
实战	设置网页元素变形动画持续时间	248
13.2.4	transition-delay 属性——实现过渡延迟效果	249
实战	设置网页元素变形动画延迟时间	249
13.2.5	transition-timing-function 属性——设置过渡方式	250
实战	设置网页元素变形动画过渡方式	251

第 14 章 JavaScript 基础

14.1	了解 JavaScript	252
14.1.1	JavaScript 概述	252
14.1.2	JavaScript 在网页中的作用	252
14.1.3	如何在网页中应用 JavaScript	253
实战	编写一个简单的 JavaScript 脚本	253
14.2	JavaScript 语法基础	254

14.2.1	JavaScript 代码格式	254
14.2.2	<script> 标签声明	254
14.2.3	大小写规范	255
14.2.4	添加注释	255
14.2.5	JavaScript 中的保留字	255
14.2.6	输出方法	255
	实战 使用 JavaScript 程序在网页中输出文字	256
14.3	变量	256
14.3.1	什么是变量	257
14.3.2	变量的声明和使用	257
	实战 定义并输出变量	257
14.4	数据类型	258
14.4.1	什么是数据类型	258
14.4.2	基本的数据类型	258
	实战 数据类型的使用	259
14.5	运算符	259
14.5.1	运算符与表达式	259
	实战 合并表达式	260
14.5.2	算术运算符	260
	实战 算术运算符的应用	260
14.5.3	赋值运算符	261
	实战 赋值运算符的应用	261
14.5.4	递增和递减运算符	262
	实战 递增和递减运算符的应用	262
14.5.5	关系运算符	263
	实战 关系运算符的应用	263
14.5.6	逻辑运算符	264
	实战 逻辑运算符的应用	264
14.5.7	条件运算符	265
	实战 条件运算符的应用	265
14.6	条件和循环语句	266
14.6.1	if 条件语句	266
	实战 if 条件语句的应用	266
14.6.2	switch 条件语句	267
	实战 switch 条件语句的应用	267
14.6.3	while 循环语句	268
	实战 while 语句的应用	268
14.6.4	do...while 循环语句	268
	实战 do...while 循环语句的应用	269
14.6.5	for 循环语句	269
	实战 for 循环语句的应用	269
14.6.6	for...in 循环语句	270
	实战 for...in 循环语句的应用	270

第 15 章 JavaScript 中的函数与对象

15.1	JavaScript 函数	272
15.1.1	什么是函数	272
15.1.2	函数的使用	272
	实战 自定义函数的应用	273
15.1.3	函数传递参数	273
	实战 使用函数传递参数	274
15.1.4	函数中变量的作用域	274
	实战 了解函数中变量的作用域	274
15.1.5	函数的返回值	275
	实战 接收函数返回值	275
15.1.6	函数嵌套	276
	实战 函数嵌套的应用	276
15.2	声明和引用对象	277
15.2.1	对象的声明	277
	实战 在 JavaScript 中声明对象	277
15.2.2	引用对象	278
	实战 引用内置对象输出系统时间	278
15.2.3	对象属性	279
	实战 设置对象属性	279
15.2.4	对象构造函数	279
	实战 对象构造函数的应用	279
15.2.5	自定义对象方法	280
	实战 自定义对象的应用	280
15.3	JavaScript 内置对象	281
15.3.1	date 对象	281
	实战 date 对象的应用	282
15.3.2	math 对象	283
	实战 math 对象的应用	283
15.3.3	string 对象	284
	实战 string 对象的应用	284
15.3.4	array 对象	284
	实战 array 对象的应用	285
15.3.5	函数对象	286
	实战 函数对象的应用	286
15.4	浏览器对象	287
15.4.1	浏览器对象 navigator	287
	实战 navigator 对象的应用	287
15.4.2	窗口对象 window	288
	实战 window 对象的应用	289
15.4.3	位置对象 location	289
	实战 location 对象的应用	290
15.4.4	历史对象 history	291
	实战 history 对象的应用	291



15.4.5 屏幕对象 screen	291
实战 screen 对象的应用	292
15.4.6 文档对象 document	292
实战 document 对象的应用	293

第 16 章 JavaScript 中的事件

16.1 了解 JavaScript 事件	294
16.1.1 JavaScript 事件类别	294
16.1.2 JavaScript 事件处理	294
16.1.3 HTML 元素常用事件	295
16.1.4 常用事件方法	295
16.2 常用事件在网页中的应用	296
16.2.1 click 事件	296
实战 使用 click 事件实现关闭网页窗口	296
16.2.2 change 事件	297
实战 使用 change 事件实现弹出提示框	297
16.2.3 select 事件	297
实战 使用 select 事件实现弹出提示框	298
16.2.4 focus 事件	298
实战 focus 事件在网页中的应用	299
16.2.5 load 事件	299
实战 load 事件在网页中的应用	299
16.2.6 鼠标移动事件	300
实战 在网页中应用鼠标移动事件	300

16.2.7 onblur 事件	301
实战 onblur 事件在网页中的应用	301

第 17 章 JavaScript 综合应用案例

17.1 实现可选择字体大小	303
17.1.1 思路分析	303
17.1.2 功能实现	303
实战 实现可选择字体大小	303
17.2 实现图像滑动切换效果	304
17.2.1 思路分析	304
17.2.2 功能实现	305
实战 实现图像滑动切换效果	305
17.3 实现滚动宣传广告	307
17.3.1 思路分析	308
17.3.2 功能实现	308
实战 实现滚动宣传广告	308
17.4 制作保健品网站	316
17.4.1 思路分析	316
17.4.2 功能实现	316
实战 制作保健品网站	316
17.5 制作儿童产品宣传网站	322
17.5.1 思路分析	322
17.5.2 功能实现	322
实战 制作儿童产品宣传网站	322

第1章 认识 HTML 与 HTML5



网页中包括的图像、动画、表单和多媒体等复杂的元素，其基础本质都是 HTML。随着互联网的飞速发展，网页设计语言也在不断地变化和发展，从 HTML 到 HTML5，每一次的发展变革都是为了适应互联网的需求。本章将介绍有关 HTML 和 HTML5 的基础知识，使读者对 HTML 的发展有所了解，并且理解 HTML5 与 HTML 之间有哪些共同点及有哪些改进。

本章知识点：

- 了解网站建设常用的软件和技术
- 了解 HTML
- 认识 HTML 的编辑环境和代码工具
- 了解 HTML5 的优势
- 理解并掌握 HTML5 的文档结构和语法规则
- 认识 HTML5 中新增的标签

1.1

了解动态网站的开发技术



网站不仅仅是把各种信息简单地堆积起来或者表达清楚就行，还要考虑通过各种设计手段和技术让受众能更多、更有效地接收网站页面中的各种信息。要想制作出精美的网站页面，需要综合运用各种网页制作工具和技术才能完成，本节将向读者简单地介绍网站开发常用的软件和技术。

1.1.1 网页编辑软件 Dreamweaver



Dreamweaver 是网页设计与制作领域中用户最多、应用广泛、功能强大的软件，无论是在国内还是在国外，它都受到专业网站开发人员的喜爱。Dreamweaver 可以用于网页的整体布局和设计，以及对网站的创建和管理，还提供了许多与编码相关的工具和功能，利用它可以轻而易举地制作出充满动感的网页。本书涉及的 HTML、CSS 和 JavaScript 都将在 Dreamweaver 中进行编写和处理，如图 1-1 所示。

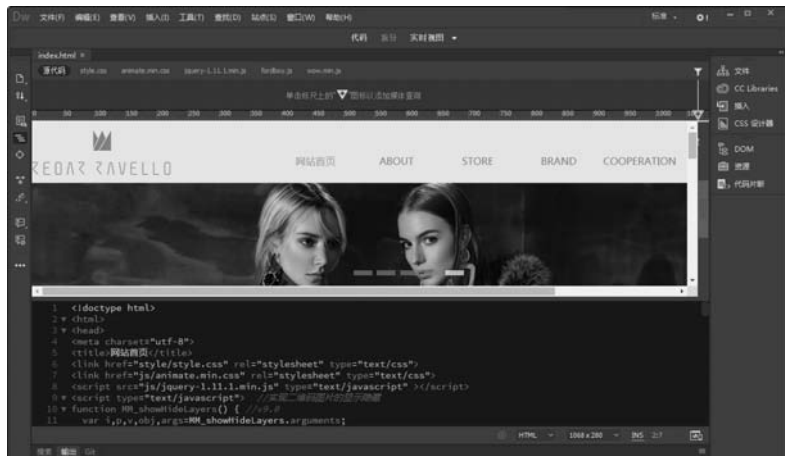


图 1-1



1.1.2 网页标记语言 HTML



要想专业地进行网页的设计和编辑,最好还要具备一定的 HTML 语言知识。虽然现在有很多可视化的网页设计制作软件,但网页的本质都是 HTML 语言构成的,如果要想精通网页制作,必须要对 HTML 语言相当了解。HTML 语言示例如图 1-2 所示。

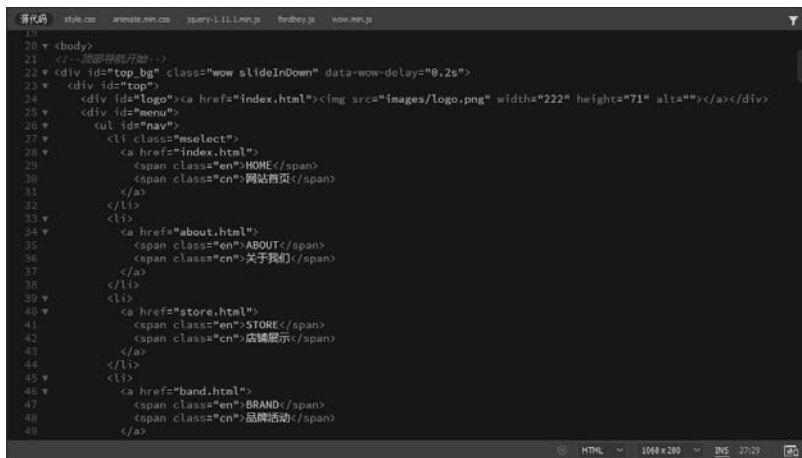


图 1-2

1.1.3 网页表现语言 CSS



如今的网页排版格式越来越复杂,很多效果都需要通过 CSS 样式来实现,即网页制作离不开 CSS 样式,采用 CSS 样式可以有效地对网页的布局、字体、颜色、背景和其他效果进行控制,只要对 CSS 样式代码做一些简单的编辑,就可以改变同一页面中不同部分或不同页面的外观和格式。使用 CSS 样式不仅可以制作出美观工整、令浏览者赏心悦目的网页,还能给网页添加许多神奇的效果。如图 1-3 所示为应用 CSS 样式的效果。

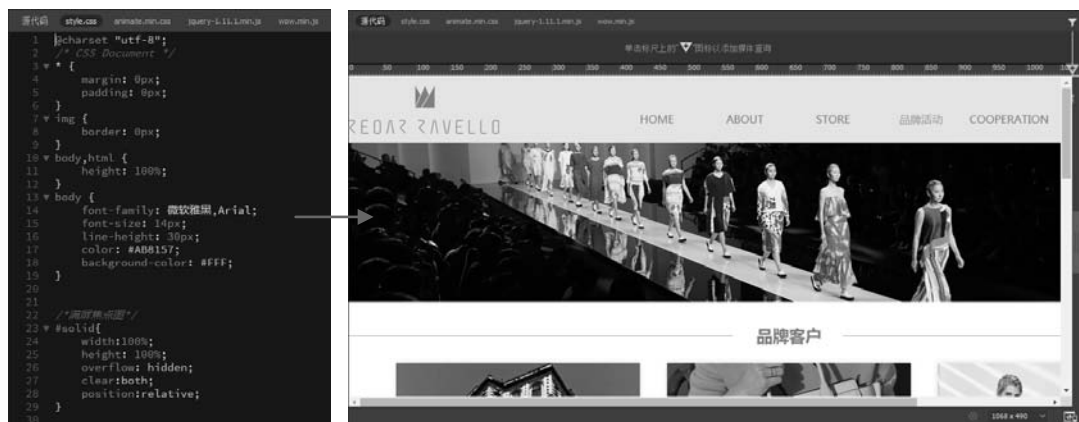


图 1-3

1.1.4 网页特效脚本语言 JavaScript



在网页设计中使用脚本语言,不仅可以减少网页的规模,提高网页的浏览速度,还可以丰富网页的表现力,因此脚本已成为网页设计中必不可少的一种技术。目前最常用的脚本有 JavaScript 和 VBScript 等,其中 JavaScript 是众多脚本语言中较为优秀的一种,是许多网页开发者首选的脚本语言。JavaScript 是一种描述性语言,它可以被嵌入 HTML 文件中。和 HTML 一样,用户可以用任何一种

文本编辑工具对它进行编辑，并在浏览器中进行预览。如图 1-4 所示为使用 JavaScript 实现的网页特效。

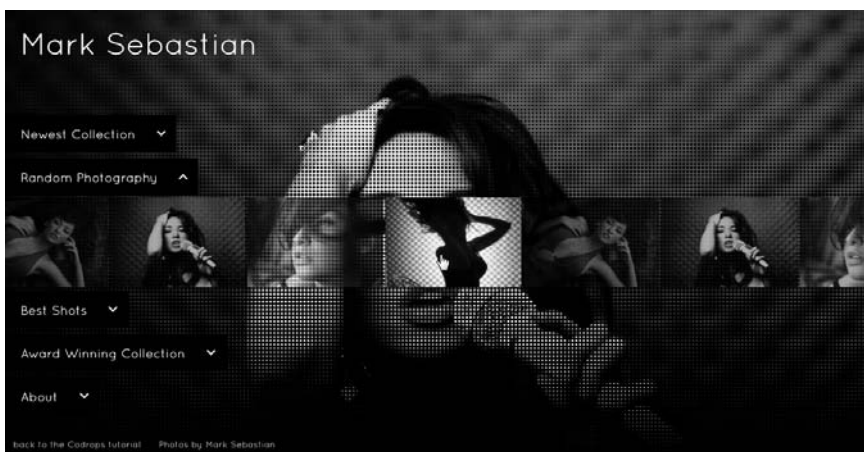


图 1-4

1.1.5 动态网页编程语言 ASP、PHP 和 JSP 等

随着互联网的快速发展，静态网站页面已经渐渐满足不了大多数网站的需求，需要通过动态网页设计语言来实现网站的交互操作和对网站内容的便捷管理。动态网站编程语言种类繁多，目前比较常用的有 ASP、PHP、JSP、CGI 和 ASP.NET 等。

ASP 是 Active Server Pages 的缩写，是 Microsoft 公司开发的 Web 服务器端脚本开发环境，利用它可以生成动态、高效的 Web 应用程序。ASP 就是嵌入了 ASP 脚本的 HTML 页面，它可以是 HTML 标签、文本和命令的任意组合。

PHP 全称为 Hypertext Preprocessor，同样是一种 HTML 内嵌式的服务器端语言，PHP 在 Windows 或 UNIX Like (UNIX、Linux、BSD 等) 平台下都能够运行，重要的是它的源代码是免费、开放的。

JSP 全称为 Java Server Pages，是由 Sun Microsystems 公司倡导，多家公司参与一起建立的一种动态网页技术标准。在传统的 HTML 网页文件中加入 Java 程序片段 (Scriptlet) 和 JSP 标记 (tag)，就构成了 JSP 网页，其文件扩展名称为 .jsp。

1.2 HTML 基础

HTML 主要运用标签使页面文件显示出预期的效果，也就是在文本文件的基础上，加上一系列的网页元素展示效果，最后形成扩展名为 .htm 或 .html 的文件。通过浏览器阅读 HTML 文件时，浏览器负责解释插入 HTML 文件中的各种标签，并以此为依据显示内容，将 HTML 语言编写的文件称为 HTML 文本，HTML 语言即网页的描述语言。

1.2.1 什么是 HTML

在介绍 HTML 语言之前，不得不介绍 World Wide Web (万维网)。万维网是一种建立在互联网上的全球性的、交互性的、动态多平台分布式图形信息系统。它采用 HTML 语法描述超文本 (Hypertext) 文件。Hypertext 一词有两个含义：一个是连接相关联的文件；另一个是内含多媒体对象的文件。

HTML 是英文 Hyper Text Markup Language 的缩写，它是一种文本类、解释执行的标记语言，是在标准一般化的标记语言 (SGML) 的基础上建立的。SGML 仅描述了定义一套标记语言的方法，而



没有定义一套实际的标记语言。而 HTML 就是根据 SGML 制定的特殊应用。

HTML 语言是一种简易的文件交换标准，有别于物理的文件结构，旨在定义文件内的对象和描述文件的逻辑结构，而并不定义文件的显示。由于 HTML 所描述的文件具有极高的适应性，所以特别适用于万维网的环境。

HTML 于 1990 年被万维网所采用，至今经历了众多版本，主要由万维网联盟 (W3C) 主导其发展。而很多编写浏览器的软件公司也根据自己的需要定义 HTML 标签或属性，所以导致现在的 HTML 标准较为混乱。

由于 HTML 语言编写的文件是标准的 ASCII 文本文件，所以可以使用任何文本编辑器来打开 HTML 文件。

提示

HTML 文件可以直接由浏览器解释执行，无须编译。当用浏览器打开网页时，浏览器读取网页中的 HTML 代码，分析其语法结构，然后根据解释的结果显示网页内容，正因为如此，网页显示的速度与网页代码的质量有很大的关系，保持精简和高效的 HTML 源代码是十分重要的。

1.2.2 HTML 的主要功能

HTML 语言作为一种网页编辑语言，易学易懂，能制作出精美的网页效果，其主要功能如下。

- ④ 利用 HTML 语言格式化文本。例如，设置标题、字体、字号、颜色；设置文本的段落、对齐方式等。
- ④ 利用 HTML 语言可以在页面中插入图像，使网页图文并茂，还可以设置图像的各种属性。例如大小、边框、布局等。
- ④ 利用 HTML 语言可以创建列表，将信息用一种易读的方式表现出来。
- ④ 利用 HTML 语言可以建立表格。表格为浏览者提供了快速找到需要信息的显示方式。
- ④ 利用 HTML 语言可以在页面中加入多媒体。可以在网页中加入音频、视频、动画，还能设定播放的时间和次数。
- ④ 利用 HTML 语言可以建立超链接。通过超链接检索在线的信息，只需用鼠标单击，就可连接到任何一处。
- ④ 利用 HTML 语言还可以实现交互式表单等效果。

1.2.3 HTML 的编辑环境

网页文件，即扩展名为 .htm 或 .html 的文件，本质上是文本类型的文件，网页中的图片、动画等资源是通过网页文件的 HTML 代码连接的，与网页文件分开存储。

由于 HTML 语言编写的文件是标准的 ASCII 文本文件，因此可以使用任意一种文本编辑器来打开或编辑 HTML 文件。例如，Windows 操作系统中自带的记事本或者专业的网页制作软件 Dreamweaver。

1. 记事本

HTML 是一个以文字为基础的语言，并不需要什么特殊的开发环境，可以直接在 Windows 操作系统自带的记事本中进行编辑，其优点是方便快捷，缺点是无任何语法提示、无行号提示和格式混乱等，初学者使用困难。

2. Dreamweaver

著名的网页设计制作软件，其优点是有所见即所得的设计视图，能够通过鼠标拖放直接创建并

编辑网页文件，自动生成相应的 HTML 代码。Dreamweaver 的代码视图有非常完善的语法自动提示、自动完成和关键词高亮等功能。可以说，Dreamweaver 是一个非常全面的网页制作工具，在本书内容的讲解和制作过程中就是使用的 Dreamweaver 软件。

3. EditPlus

EditPlus 是一个非常优秀的代码编辑器，可以很方便地创建和编辑网页文件。其优点是方便快捷、有语法高亮、行号提示和 HTML 代码快捷插入等。缺点是无语法自动提示、无所见即所得的网页设计视图。专业的代码编辑器比较适合代码熟练的用户，并不适合初学者。

1.2.4 认识 Dreamweaver

Dreamweaver 软件自从推出以来就作为网页制作工具的标准被人们广泛应用，它的后续升级版本在功能方面都紧跟当前的网页发展趋势，为用户提供方便、快捷的网页设计制作方法。最新版的 Dreamweaver CC 2018 为设计和开发人员提供了强大的可视化布局工具、应用开发功能和代码编辑支持功能。如图 1-5 所示为最新版本的 Dreamweaver CC 2018 软件的工作界面。



图 1-5

1. 菜单栏

菜单栏中包含所有 Dreamweaver 操作需要的命令。这些命令按照操作类别分为“文件”“编辑”“查看”“插入”“工具”“查找”“站点”“窗口”和“帮助”9 个菜单。

2. 视图选项

在 Dreamweaver 中提供了 3 种视图窗口形式，分别是“代码视图”“拆分视图”和“实时视图”，其中“实时视图”中还包含旧版本中的“设计视图”选项。默认我们通常使用“拆分视图”形式，这样可以在编写 HTML 代码的同时实时地查看页面的效果。

3. 实时视图

在“实时视图”窗口中显示当前制作页面的效果，也是可视化操作的窗口，可以使用各种工具，在该窗口中输入文字、插入图像等，是所见即所得的视图。

4. 代码视图

在“代码视图”窗口中将显示当前所编辑页面的 HTML 代码，可以直接在该窗口中进行 HTML 页面代码的编写操作。在“代码视图”中，Dreamweaver 通过不同的颜色对不同的 HTML 代码进行区别，并且在编写过程中实时为用户提供相应的提示，非常方便。



5. 工具栏

在 Dreamweaver 工作界面的左侧为用户提供了文档编辑操作的相关工具，可以实现对文档 HTML 代码的快速格式化、添加注释等操作，用户也可以自定义工具栏中所显示的工具选项。

6. 标签选择器

显示环绕当前选定内容的标签的层次结构。单击该层次结构中的任何标签，可以选择该标签及其全部内容。

7. 面板组

用于帮助用户进行监控和修改的工作。例如，“插入”面板、“CSS 设计器”面板。单击相应的面板名称，可以折叠或展开相应的工具面板。

实战 制作第一个 HTML 页面

最终文件：最终文件\第 1 章\1-2-4.html 视频：视频\第 1 章\1-2-4.mp4



01 打开 Dreamweaver，执行“文件”>“新建”命令，弹出“新建文档”对话框，选择 HTML 选项，如图 1-6 所示。单击“创建”按钮，新建 HTML5 文件，在代码视图中可以看到文档的 HTML 代码，如图 1-7 所示。



图 1-6

```
1 <!doctype html>
2 <html>
3 <head>
4 <meta charset="utf-8">
5 <title>无标题文档</title>
6 </head>
7
8 <body>
9 </body>
10 </html>
11
```

图 1-7

02 执行“文件”>“保存”命令，弹出“另存为”对话框，将该网页保存为“源文件\第 1 章\1-2-4.html”，如图 1-8 所示。在页面的 <title> 与 </title> 标签之间输入网页的标题，如图 1-9 所示。



图 1-8

```
<html>
<head>
<meta charset="utf-8">
<title>第一个HTML网页</title>
</head>

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

图 1-9

03 在 <body> 标签中添加 style 属性设置代码，如图 1-10 所示。在 <body> 与 </body> 标签之间编写相应的网页正文内容代码，如图 1-11 所示。

```
<html>
<head>
<meta charset="utf-8">
<title>第一个HTML网页</title>
</head>

<body style="background-color:#b2e4e5; text-align:center; color:#036e70;">

</body>
</html>
```

图 1-10

```
<body style="background-color:#b2e4e5; text-align:center; color:#036e70;">
<br>
<br>
<br>
<br>
<br>

<br>
<b>欢迎进入我们的网站设计工作室</b>
</body>
```

图 1-11

提示

在 <body> 标签中添加 style 属性设置，实际上是 CSS 样式的一种使用方式，称为内联 CSS 样式。此处通过内联 CSS 样式设置页面整体的背景颜色、水平对齐方式和文字颜色。

04 完成该网页 HTML 代码的编写，执行“文件”>“保存”命令，保存网页，在浏览器中预览该网页，可以看到网页的效果，如图 1-12 所示。



图 1-12

技巧

Dreamweaver 是一款专业的网页制作软件。在该软件中新建 HTML 页面，会自动给出 HTML 文件结构的基础代码，编写 HTML 代码还具有代码提示等功能，非常适合初学者使用。

1.3

HTML5 基础



HTML5 是近十年来 Web 标准巨大的飞跃。和以前的版本不同，HTML5 并非仅仅用来表示 Web 内容，它的使命是将 Web 带入一个成熟的应用平台，在这个平台上，视频、音频、图像、动画，以及与计算机的交互都被标准化。HTML5 目前仍然处于发展的阶段，基于互联网的应用已经越来越丰富，同时也对互联网应用提出了更高的要求。

1.3.1 HTML5 概述



W3C 在 2010 年发布了 HTML5 的工作草案，并于 2014 年制定出完整的 HTML5 标准规范。HTML5 的工作组包括 AOL、Apple、Google、IBM、Microsoft、Mozilla、Nokia、Opera 以及数百个其他的开发商。制定 HTML5 的目的是取代 1999 年 W3C 所制定的 HTML4.01 和 XHTML1.0 标准，希望在网络应用迅速发展的同时，网页语言能够符合网络发展的需求。



HTML5 实际上是指包括 HTML、CSS 样式和 JavaScript 脚本在内的一整套技术的组合,希望通过 HTML5 能够轻松地实现许多丰富的网络应用需求,而减少浏览器对插件的依赖,并且提供更多能有效增强网络应用的标准集。

在 HTML5 中添加了许多新的应用标签,其中包括 <video>、<audio> 和 <canvas> 等标签,添加这些标签是为了使设计者能够更轻松地网页中添加或处理图像和多媒体内容。其他新的标签还有 <section>、<article>、<header> 和 <nav>,这些新添加的标签是为了能够更加丰富网页中的数据内容。除了添加了许多功能强大的新标签和属性,同样还对一些标签进行了修改,以方便适应快速发展的网络应用。同时也有一些标签和属性在 HTML5 标准中已经被去除。

1.3.2 HTML5 的优势

对于用户和网站开发者而言,HTML5 的出现意义非常重大。因为 HTML5 解决了 Web 页面存在的诸多问题,HTML5 的优势主要表现在以下几个方面。

1. 化繁为简

HTML5 为了做到尽可能简化,避免了一些不必要的复杂设计。例如,DOCTYPE 声明的简化处理,在过去的 HTML 版本中,第一行的 DOCTYPE 过于冗长,在实际的 Web 开发中也没有什么意义,而在 HTML5 中 DOCTYPE 声明就非常简洁。

为了让一切变得简单,或避免造成误解,HTML5 对每一个细节都有非常明确的规范说明,不允许有任何的歧义和模糊出现。

2. 向下兼容

HTML5 有很强的兼容能力。在这方面,HTML5 没有颠覆性的革新,允许存在不严谨的写法。例如,一些标签的属性值没有使用英文引号括起来;标签属性中包含大写字母;有的标签没有闭合等。然而这些不严谨的错误处理方案,在 HTML5 的规范中都有明确的规定,也希望未来在浏览器中有一致的支持。当然对于 Web 开发者来说,还是遵循严谨的代码编写规范比较好。

对于 HTML5 的一些新特性,如果旧的浏览器不支持,也不会影响页面的显示。在 HTML 规范中,也考虑了这方面的内容。例如,在 HTML5 中 <input> 标签的 type 属性增加了很多新的类型,当浏览器不支持这些类型时,默认会将其视为 text。

3. 支持合理

HTML5 的设计者花费了大量的精力来研究通用的行为。例如,Google 分析了上百万份的网页,从中提取了 <div> 标签的 id 名称,很多网页开发人员都如下标记导航区域。

```
<div id="nav">
  // 导航区域内容
</div>
```

既然该行为已经大量存在,HTML5 就会想办法去改进,所以就直接增加了一个 <nav> 标签,用于网页导航区域。

4. 实用性

对于 HTML 无法实现的一些功能,用户会寻求其他方法来实现。例如,对于绘图、多媒体、地理位置和实时获取信息等应用,通常会开发一些相应的插件间接地去实现。HTML5 的设计者研究了这些需求,开发了一系列用于 Web 应用的接口。

HTML5 规范的制定是非常开放的,所有人都是可以获取草案的内容,也可以参与进来提出宝贵的意见。因为开放,所以可以得到更加全面的发展,一切以用户需求为最终目的。所以,当用户在使用 HTML5 的新功能时,会发现正是期待已久的功能。

5. 用户优先

当遇到无法解决的冲突时，HTML5 规范把最终用户的诉求放在第一位。因此，HTML5 的绝大部分功能都是非常实用的。用户与开发者的重要性远远高于规范和理论。例如，有很多用户都需要实现一项新的功能，HTML5 规范的设计者们会研究这种需求，并纳入规范；HTML5 规范了一套错误处理机制，以便当 Web 开发者写了不够严谨的代码时，接纳这种不严谨的写法。HTML5 比以前版本的 HTML 更加友好。

1.4 认识 HTML5

HTML5 的语法结构和 HTML 的语法结构基本一致，下面将分别介绍 HTML5 的文档结构与基本语法。

1.4.1 HTML5 的文档结构

编写 HTML 文件时，必须遵循 HTML 的语法规则。一个完整的 HTML 文件由标题、段落、列表、表格、单词和嵌入的各种对象组成。这些逻辑上统一的对象统称为元素，HTML 使用标签来分隔并描述这些元素。实际上整个 HTML 文件就是由元素与标签组成的。

HTML 文件基本结构如下。

<code><html></code>	<code><!--HTML 文件开始 --></code>
<code><head></code>	<code><!--HTML 文件的头部开始 --></code>
头部内容	
<code></head></code>	<code><!--HTML 文件的头部结束 --></code>
<code><body></code>	<code><!--HTML 文件的主体开始 --></code>
主体内容	
<code></body></code>	<code><!--HTML 文件的主体结束 --></code>
<code></html></code>	<code><!--HTML 文件结束 --></code>

可以看到，代码分为以下 3 部分。

1. `<html>...</html>`

告诉浏览器 HTML 文件开始和结束，`<html>` 标签出现在 HTML 文件的第一行，用来表示 HTML 文件的开始。`</html>` 标签出现在 HTML 文件的最后一行，用来表示 HTML 文件的结束。两个标签一定要一起使用，网页中的所有其他内容都要放在 `<html>` 与 `</html>` 之间。

2. `<head>...</head>`

网页的头标签，用来定义 HTML 文件的头部信息，该标签也是成对使用的。

3. `<body>...</body>`

在 `<head>` 与 `</head>` 标签之后就是 `<body>` 与 `</body>` 标签，该标签也是成对出现的。`<body>` 与 `</body>` 标签之间为网页主体内容和其他用于控制内容显示的标签。

1.4.2 HTML5 的基本语法

绝大多数元素都有起始标签和结束标签，在起始标签和结束标签之间的部分是元素体。例如，`<body>...</body>`。每一个元素都有名称和可选择的属性，元素的名称和属性都在起始标签内进行设置。

1. 普通标签

普通标签是由一个起始标签和一个结束标签组成的，其语法格式如下。

```
<x> 内容 </x>
```



其中, x 代表标签名称。`<x>` 和 `</x>` 就如同一组开关: 起始标签 `<x>` 为开启某种功能, 而结束标签 `</x>` (通常为起始标签加上一个斜线 `/`) 为关闭功能, 受控制的内容便放在两标签之间。例如下面的代码。

```
<b> 加粗文字 </b>
```

标签之中还可以附加一些属性, 用来实现或完成某些特殊效果或功能。例如下面的代码。

```
<x a1="v1" a2="v2"...an="vn"> 内容 </x>
```

其中, $a_1, a_2, \dots, a_n$ 为属性名称, 而 $v_1, v_2, \dots, v_n$ 则是其所对应的属性值。属性值加不加引号, 目前所使用的浏览器都可接受, 但根据 W3C 的新标准, 属性值是需要加引号的, 所以最好养成加引号的习惯。

2. 空标签

虽然大部分的标签是成对出现的, 但也有一些是单独存在的, 这些单独存在的标签称为空标签, 其语法格式如下。

```
<x>
```

同样, 空标签也可以附加一些属性, 用来完成某些特殊效果或功能。例如下面的代码。

```
<x a1="v1" a2="v2"...an="vn">
```

以及下面的代码。

```
<hr color="#0000FF">
```

1.4.3 HTML5 精简的头部

HTML5 避免了不必要的复杂性, DOCTYPE 和字符集都极大地简化了。

DOCTYPE 声明是 HTML 文件中必不可少的内容, 它位于 HTML 文件的第一行, 声明了 HTML 文件遵循的规范。HTML4.01 的 DOCTYPE 声明代码如下。

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
```

这么长的一串代码恐怕极少有人能够默写出来, 通常都是通过复制、粘贴的方式添加这段代码。而在 HTML5 中的 DOCTYPE 代码则非常简单, 如下所示。

```
<!DOCTYPE html>
```

这样就简洁了许多, 不需要再复制、粘贴代码。同时这种声明也标识性地让人感觉到这是符合 HTML5 规范的页面。如果使用了 HTML5 的 DOCTYPE 声明, 则会触发浏览器以标准兼容的模式来显示页面。

字符集的声明也是非常重要的, 它决定了页面文件的编码方式。在过去都是使用如下方式来指定字符集的。

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
```

HTML5 对字符集的声明也进行了简化处理, 简化后的声明代码如下。

```
<meta charset="utf-8">
```

在 HTML5 中, 以上两种字符集的声明方式都可以使用, 这是由 HTML5 向下兼容的原则决定的。

1.5 HTML5 中新增的标签

在 HTML5 中新增了许多新的有意义的标签, 为了方便学习和记忆, 本节将对 HTML5 中新增的

标签进行分类介绍。

1.5.1 结构标签

HTML5 中新增的结构标签说明如表 1-1 所示。

表 1-1 HTML5 新增的结构标签说明

标签	说明
<article>	<article> 标签用于在网页中标识独立的主体内容区域，可用于论坛帖子、报纸文章、博客条目和用户评论等
<aside>	<aside> 标签用于在网页中标识非主体内容区域，该区域中的内容应该与附近的主体内容相关
<section>	<section> 标签用于在网页中标识文档的小节或部分
<footer>	<footer> 标签用于在网页中标识页脚部分，或者内容区块的脚注
<header>	<header> 标签用于在网页中标识页首部分，或者内容区块的标头
<nav>	<nav> 标签用于在网页中标识导航部分

1.5.2 文本标签

HTML5 中新增的文本标签说明如表 1-2 所示。

表 1-2 HTML5 新增的文本标签说明

标签	说明
<bdi>	<bdi> 标签在网页中允许设置一段文本，使其脱离其父元素的文本方向设置
<mark>	<mark> 标签用于在网页中标识需要高亮显示的文本
<time>	<time> 标签用于在网页中标识日期或时间
<output>	<output> 标签在用于网页中标识一个输出的结果

1.5.3 应用和辅助标签

HTML5 中新增的应用和辅助标签说明如表 1-3 所示。

表 1-3 HTML5 新增的应用和辅助标签说明

标签	说明
<audio>	<audio> 标签用于在网页中定义声音，如背景音乐或其他音频流
<video>	<video> 标签用于在网页中定义视频，如电影片段或其他视频流
<source>	<source> 标签为媒介标签（如 video 和 audio），用于在网页中定义媒介资源
<track>	<track> 标签在网页中为如 video 元素之类的媒介规定外部文本轨道
<canvas>	<canvas> 标签用于在网页中定义图形。例如，图标和其他图像。该标签只是图形容器，必须使用脚本来绘制图形
<embed>	<embed> 标签用于在网页中标识来自外部的互动内容或插件

1.5.4 进度标签

HTML5 中新增的进度标签说明如表 1-4 所示。

表 1-4 HTML5 新增的进度标签说明

标签	说明
<progress>	<progress> 标签用于在网页中标识任务进度显示的进度条
<meter>	<meter> 标签可以显示进度条，在该标签中可以设置最大和最小值，并且可以根据 value 属性值来显示当前的进度位置

1.5.5 交互性标签

HTML5 中新增的交互性标签说明如表 1-5 所示。



表 1-5 HTML5 新增的交互性标签说明

标签	说明
<command>	<command> 标签用于在网页中标识一个命令元素（单选、复选或者按钮）；当且仅当这个元素出现在 <menu> 标签里面时才会被显示，否则将只能作为键盘快捷方式的一个载体
<datalist>	<datalist> 标签用于在网页中标识一个选项组，与 <input> 标签配合使用，来定义 input 元素可能的值

1.5.6 在文档和应用中使用的标签

HTML5 中新增的在文档和应用中使用的标签说明如表 1-6 所示。

表 1-6 在文档和应用中使用的标签说明

标签	说明
<details>	<details> 标签用于在网页中标识描述文档或者文档某个部分的细节
<summary>	<summary> 标签用于在网页中标识 <details> 标签内容的标题
<figcaption>	<figcaption> 标签用于在网页中标识 <figure> 标签内容的标题
<figure>	<figure> 标签用于在网页中标识一块独立的流内容（图像、图表、照片和代码等）
<hgroup>	<hgroup> 标签用于在网页中标识文档或内容的多个标题。用于将 h1 至 h6 元素打包，优化页面结构在 SEO 中的表现

1.5.7 <ruby> 标签

HTML5 中新增的 <ruby> 标签说明如表 1-7 所示。

表 1-7 <ruby> 标签说明

标签	说明
<ruby>	<ruby> 标签用于在网页中标识 ruby 注释（中文注音或字符）
<rp>	<rp> 标签在 ruby 注释中使用，以定义不支持 <ruby> 标签的浏览器所显示的内容
<rt>	<rt> 标签用于在网页中标识字符（中文注音或字符）的解释或发音

1.5.8 其他标签

HTML5 中新增的其他标签说明如表 1-8 所示。

表 1-8 其他标签说明

标签	说明
<keygen>	<keygen> 标签用于标识表单密钥生成器元素。当提交表单时，私钥存储在本地，公钥发送到服务器
<wbr>	<wbr> 标签用于标识单词中适当的换行位置，可以用该标签为一个长单词指定合适的换行位置

1.6

HTML5 中废弃的标签

在 HTML5 中也废弃了一些以前 HTML 中的标签，主要是以下几个方面的标签。

1. 可以使用 CSS 样式替代的标签

在 HTML5 之前的一些标签中，有一部分是纯粹用作显示效果的标签。而 HTML5 延续了内容与表现分离，对于显示效果更多地交给 CSS 样式去完成。所以，在这方面废弃的标签有 <basefont>、<big>、<center>、、<s>、<strike>、<tt> 和 <u>。

2. 不再支持 frame 框架

由于 frame 框架对网页可用性存在负面影响，因此在 HTML5 中已经不再支持 frame 框架，但

是支持 iframe 框架。所以 HTML5 中废弃了 frame 框架的 <frameset>、<frame> 和 <noframes> 标签。

3. 其他废弃标签

在 HTML5 中其他被废弃的标签主要是因为有了更好的替代方案。

废弃 <bgsound> 标签，可以使用 HTML5 中的 <audio> 标签替代。

废弃 <marquee> 标签，可以在 HTML5 中使用 JavaScript 程序代码来实现。

废弃 <applet> 标签，可以使用 HTML5 中的 <embed> 和 <object> 标签替代。

废弃 <rb> 标签，可以使用 HTML5 中的 <ruby> 标签替代。

废弃 <acronym> 标签，可以使用 HTML5 中的 <abbr> 标签替代。

废弃 <dir> 标签，可以使用 HTML5 中的 标签替代。

废弃 <isindex> 标签，可以使用 HTML5 中的 <form> 和 <input> 标签结合的方式替代。

废弃 <listing> 标签，可以使用 HTML5 中的 <pre> 标签替代。

废弃 <xmp> 标签，可以使用 HTML5 中的 <code> 标签替代。

废弃 <nextid> 标签，可以使用 HTML5 中的 GUIDS 替代。

废弃 <plaintext> 标签，可以使用 HTML5 中的 “text/plain” MIME 类型替代。

1.7

HTML、CSS 与 JavaScript 的结合



网页中包括文本、图像、动画、多媒体和表单等多种复杂的元素，但是其基础架构仍然是 HTML 语言。HTML 是互联网用于设计网页的主要语言，注意 HTML 只是一种标记语言，与其他程序设计语言不同的是，HTML 只能建议浏览器以什么方式或结构显示网页内容。HTML 相对比较简单，初学者只要掌握 HTML 的一些常用标签即可。

CSS 样式是为了弥补 HTML 的不足而出现的，最初 HTML 是可以标记页面中的标题、段落、表格和链接等格式的。但是随着网络的发展，用户需求的增加，HTML 越来越不能满足不同页面表现的需求。为了解决这个问题，1997 年 W3C 颁布 HTML4 标准的同时也公布了有关样式表的第一个标准 CSS1。随着网络的发展，CSS 样式又得到了更多的完善、充实和发展，目前最新的 CSS 样式版本为 CSS3，其功能也越来越强大。

随着网络的普及，用户对网站的要求越来越高，已经不再满足于使用 HTML 和 CSS 样式配合制作出的静态页面，需要有更多的交互性，使网页使用更方便，浏览过程更有趣。出于这样的需求，JavaScript 在网页中的应用越来越广泛，JavaScript 用于开发 Internet 客户端的应用程序，它可以与 HTML 和 CSS 样式相结合，实现在网页中与浏览者进行交互的功能。

第2章

HTML 主体标签



HTML 网页文件是组成网站的基本单位，有完整的结构。本章从 HTML 的主体标签入手，全面开始对 HTML 中的标签及其相关属性进行学习。通过本章的学习，读者将掌握 HTML 文件的头部信息设置和页面主体的基本设置。

本章知识点：

- 了解 HTML 的主体标签
- 理解并掌握 <meta> 标签的设置与使用方法
- 掌握主体标签 <body> 中相关属性的设置方法
- 了解在 HTML 代码中添加注释的方法
- 理解并掌握在 HTML 代码中调用外部 JavaScript 程序的方法

2.1

HTML 头部 <head> 标签设置



通过前面的学习，了解到 HTML 的基本结构分为 <head></head> 部分和 <body></body> 部分。head 的中文意思即头部，因此一般把 <head></head> 部分称为网页的头部信息。头部信息部分的内容虽然不会在网页中显示，但它能影响网页的全局设置。

2.1.1 <title> 标签



网页标题与文章标题的性质是一样的，它们都表示重要的信息，允许用户快速浏览网页，找到他们需要的信息。网页标签的设置是很重要的，因为网站访问者并不总是阅读网页上的所有文字。为网页设置标题，只需要在 HTML 文件的头部 <title></title> 标签之间输入标题信息就可以在浏览器上显示。

网页的标题只有一个，位于 HTML 文件的头部 <head> 与 </head> 标签之间，<title> 标签的基本语法如下。

```
<head>
<title>...</title>
</head>
```

实 战

设置网页标题



最终文件：最终文件\第2章\2-1-1.html 视频：视频\第2章\2-1-1.mp4



01 打开页面“源文件\第2章\2-1-1.html”，可以看到页面的 HTML 代码，如图 2-1 所示。在浏览器中预览该页面，网页的标签默认为“无标题文档”，如图 2-2 所示。

02 返回网页的 HTML 代码中，在页面头部的 <title> 与 </title> 标签之间输入网页的标题，如图 2-3 所示。执行“文件”>“保存”命令，保存该页面，在浏览器中预览页面，可以看到设置的网页标题，如图 2-4 所示。