

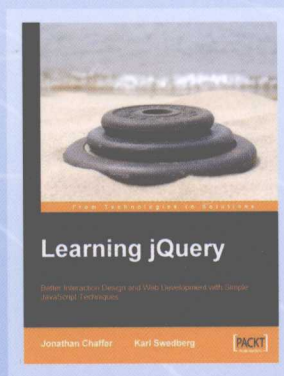
Learning jQuery

Better Interaction Design and Web Development
with Simple JavaScript Techniques

jQuery基础教程

[美] Jonathan Chaffer 著
Karl Swedberg
李松峰 李炜 等译

- 全球第一部jQuery著作
- Amazon全五星盛誉
- jQuery官方网站推荐



人民邮电出版社
POSTS & TELECOM PRESS

TURING

图灵程序设计丛书 Web开发系列

TP312/2893

2008

Learning jQuery
Better Interaction Design and Web Development
with Simple JavaScript Techniques

jQuery基础教程

[美] Jonathan Chaffer 著
Karl Swedberg

李松峰 李炜 等译

人民邮电出版社
北 京

图书在版编目 (CIP) 数据

jQuery 基础教程 / (美) 查弗 (Chaffer, J.), (美) 斯威德伯格 (Swedberg, K.) 著; 李松峰等译. —北京: 人民邮电出版社, 2008.7

(图灵程序设计丛书)

书名原文: Learning jQuery: Better Interaction Design and Web Development with Simple JavaScript Techniques
ISBN 978-7-115-18110-7

I. j… II. ①查…②斯…③李… III. JAVA 语言—程序设计—教材 IV. TP312

中国版本图书馆CIP数据核字 (2008) 第067863号

内 容 提 要

本书以通俗易懂的方式介绍了 jQuery 的基本概念, 主要包括 jQuery 的选择符、事件、特效、DOM 操作、AJAX、表格操作、表单函数、拖曳与旋转和插件等内容, 最后几章以实例操作为主, 在前面内容的基础上, 提出了常见的客户端实际问题并给出了其解决方案。

本书是一本注重理论与实践结合的基础教程, 适合 Web 开发人员阅读和参考。

图灵程序设计丛书

jQuery基础教程

-
- ◆ 著 [美] Jonathan Chaffer Karl Swedberg
译 李松峰 李 炜 等
责任编辑 杨 爽
 - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街14号
邮编 100061 电子函件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
三河市海波印务有限公司印刷
新华书店总店北京发行所经销
 - ◆ 开本: 800×1000 1/16
印张: 19.5
字数: 458千字
印数: 1—4 000册
- 2008年7月第1版
2008年7月河北第1次印刷

著作权合同登记号 图字: 01-2008-2029号

ISBN 978-7-115-18110-7/TP

定价: 45.00元

读者服务热线: (010)88593802 印装质量热线: (010)67129223

反盗版热线: (010)67171154

版 权 声 明

Copyright © Packt Publishing 2007. First published in the English language under the title *Learning jQuery : Better Interaction Design and Web Development with Simple JavaScript Techniques*.

Simplified Chinese-language edition copyright © 2008 by Posts & Telecom Press. All rights reserved.

本书中文简体字版由Packt Publishing授权人民邮电出版社独家出版。未经出版者书面许可，不得以任何方式复制或抄袭本书内容。

版权所有，侵权必究。

译者序

说起我与jQuery结缘，还要感谢2006年发生在台湾海峡的地震。2006年12月26日，中国南海台湾附近发生7.2级地震，数分钟后又发生了6.7级地震。受强烈地震影响，中美海缆等多条国际海底通信光缆发生中断，造成附近国家和地区的国际和地区性通信受到严重影响。2007年1月29日，电信网通宣布，经过20多天的抢修，受地震影响中断的国际通信业务已全部恢复。在此期间，中国雅虎在邮箱主页顶部发布了一个由于海缆中断可能会造成邮件收发有问题的通告。当时，通告是在页面加载完成大约1s后，以渐变和动画形式出现在页面顶部的——跟jQuery官方网站首页那个“The quick and dirty”的演示效果很相似。而且，通告显示了大约几秒钟后又以动画形式自动消失，整个页面好像什么都没有发生过一样。这个动画效果深深地吸引了我。以前，我也试着写过像卓越亚马逊网站首页“所有20类商品”按钮的鼠标悬停动画（可以在<http://www.amazon.com.cn/>上面看到这个动画效果），但使用了几十行代码，如今这个更酷的效果是怎么实现的呢？于是，我怀着强烈的好奇心开始查看它的源代码（这要感谢JavaScript天生的开源特性）。惊奇地发现，这个效果仅用了寥寥几行代码！惊讶之余，溯本求源，最后“认识”了精巧而美妙的jQuery，特别是它优雅的方法连缀能力，更令我如获至宝、兴奋不已！后来我查了很多jQuery的资料，发现它的文档没有汉化，就用一周的休息时间翻译了它的API（1.1版）文档。这份汉化文档在jQuery中文资料匮乏的时候为广大jQuery网友提供了一点帮助，也获得了大家的认可和好评。

JavaScript库和框架解决的问题，无非就是（跨浏览器的）DOM操作、事件处理、样式更换和外部通信（AJAX）。但jQuery独特的集合对象、隐式迭代、方法连缀、自定义选择符和事件方法，加之只有不到20 KB的超轻巧和执行速度超快，赢得了众多JavaScript开发者的青睐。

jQuery不仅支持各式各样的CSS选择符表达式，而且还支持XPath和自定义的选择符表达式，这一点在JavaScript库和框架领域中无出其右者，使开发者找到要操作的元素或集合简单得难以置信；它细腻灵巧而又富有弹性的事件处理机制，包括事件注册、触发和自定义，特别是令JavaScript的Guru级人物都喜不自禁的`hover()`方法，使它在JavaScript库和框架之林中独树一帜、个性十足；它在操作DOM文档时的大处着眼、小处着手，提供的丰富而实用的各种遍历和操作DOM结构及元素的方法，令人耳目一新，简直“直逼每个JavaScript爱好者的心理防线”，那种令人怦然心动的感觉，历久弥新；它处理AJAX请求和响应的简洁明快、它的简单易用、它超级方便的扩展机制、它丰富的插件支持（Interface等）、它背后的强大社区……所有这些，引无数JavaScript高手竞折腰！

事实上,因特网上的JavaScript库和框架数以百千计,为什么唯独jQuery对我们这些爱好者有如此大的吸引力呢?就是因为jQuery采取了与其他库和框架截然不同的理念,处处匠心独运,别出心裁——具体细节,请参考第1章。

本书作为第一本全面、深入介绍jQuery库的图书,可以说是应运而生的。书中包含了jQuery教程、jQuery实例和JavaScript最佳实践。jQuery教程部分是本书第2章至第6章,分别介绍了jQuery中的选择符、事件处理、DOM操作、动画效果和AJAX方法。其中,第3章、第4章、第5章结尾,特别归纳了相应方法及适用情形,既简明又实用。jQuery实例部分是本书第7章、第8章、第9章,分别围绕Web开发中最常见的表格、表单和动画效果,详尽地探讨了使用jQuery的方方面面。这几章的实例,深入讨论诸多Web开发问题,深入浅出、娓娓道来,时不时令人拍案叫绝、感叹很多百思不得其解的问题,其实只有一层窗户纸!第10章介绍了jQuery强大的扩展能力,介绍了扩展jQuery或者编写自己的jQuery插件的方法。这一章深入到jQuery核心,把整个库的架构全部展现给了读者,并向读者揭示出jQuery库中的“陷阱”和“关键”,令人有豁然开朗、恍然大悟之感。

现代JavaScript开发的一个基准点就是最佳实践。为了让读者不走弯路、不浪费宝贵的时间,本书在介绍通过jQuery进行JavaScript开发的过程中,实践了“渐进增强”和“平稳退化”这两个不唐突(unobtrusive)的JavaScript开发原则。把抽象的概念形象化、具体化,字里行间,渗透着作者对这些先进理念的阐发与启示。

值得一提的是,本书附录C是名符其实的“压轴好戏”。这么举重若轻、浅显易懂地讨论JavaScript闭包,在译者看来还是头一次。几个精心设计的例子,读者跟着走下来,不知不觉中就能领略到JavaScript这一高级特性的精髓所在(也许没有说得那么容易)。

书是人类进步的阶梯,这话一点不假,但“尽信书不如无书”。要想学习jQuery不能不看jQuery的图书,但是,只看是不管用的,还要动手实践——打开文本编辑器和浏览器,亲手写jQuery代码!书中很多地方讲的只是要点,而动手实践才能收获书中没有讲到的东西。

本书由李松峰负责翻译,参加翻译工作的还有李炜、秦绪文、李丽、程宝杰、宋连海、付荣艳、封耀杰、贾爱华、左艳坡、熊俊芹、刘英、宋会敏等。

最后,也是最重要的,我要感谢在翻译此书过程中,傅志红老师给我提供的帮助和建议。感谢武卫东老师、刘江老师对译稿的指点。感谢本书的责任编辑杨爽对译稿的认真审核和修改,如果不是她创造性地与我沟通,本书恐怕要留下不少遗憾。此外,还要感谢图灵俱乐部“明月星光”网友的热心建言,他的建议解决了一些我在翻译过程中遇到的问题。不过,囿于个人水平和能力,翻译中的错误和不当之处在所难免。如果读者发现了书中的问题,请在我的个人网站<http://www.cn-cuckoo.com>中指出,或者将电子邮件发送到lsf.email@yahoo.com.cn。

译者

2008年2月于北京

前言

jQuery是一个强大的JavaScript库。无论你具有什么编程背景，都可以通过它来增强自己的网站。

由John Resig创建的jQuery是一个开源项目，其核心团队由富有献身精神的顶尖JavaScript开发人员组成。jQuery在一个紧凑的文件中提供了丰富多样的特性、简单易学的语法和稳健的跨平台兼容性。此外，百余种为扩展jQuery功能而开发的插件，更使得它几乎成为适用于各类客户端脚本编程的必备工具。

本书以通俗易懂的方式介绍了jQuery的基本概念，通过学习本书，即使曾经因编写JavaScript而受过挫折的人，也能够掌握为网页添加交互和动态效果的技术。本书将引导读者跨越AJAX、事件、效果及高级JavaScript语言特性中的各种陷阱。

本书网站<http://book.learningjquery.com>中，包含书中各章的在线示例。

本书内容

本书的第一部分是jQuery简介，用来帮助读者对jQuery有个大概的了解。第1章的内容主要涉及如何下载和设置jQuery库，同时也会指导你使用jQuery编写第一个脚本。

本书的第二部分将深入讨论jQuery库的各个主要方面。第2章讲述如何取得我们想要的一切。通过jQuery中的选择符表达式，你可以在页面中的任何地方找到想要的元素。这一章将使用各种选择符表达式为页面中的不同元素添加样式，其中一些是通过纯CSS方式做不到的。

第3章讲述如何“扣动扳机”。本章介绍如何通过jQuery的事件处理机制，在浏览器发生事件时触发行为。同时，还会介绍jQuery的独家秘笈——以不唐突的方式添加事件（甚至在页面加载完成之前）。

第4章讲述如何增加操作的艺术感。这一章介绍通过jQuery实现动画的技术，从中我们能够体会到隐藏、显示和移动页面元素时那种轻松自如的感觉。

第5章讲述如何通过指令改变页面。本章讲述的是动态修改HTML文档结构的技术。

第6章讲述如何让你的网站跻身主流行列。在学习完本章后，你也可以做到不用像过去那样刷新页面而访问服务器端功能。

本书的第三部分与前两部分不同。这一部分主要以实例为主，即在前几章学习的基础上，创建常见问题的稳健jQuery解决方案。第7章将讲述排序、筛选和为信息添加样式并创建优美实用的数据布局。

第8章以客户端数据验证为主题。届时，将设计一个具有适应能力的表单布局，还会实现基于客户端与服务器通信的交互式表单功能，例如自动完成。

第9章介绍如何在小窗口内显示页面元素来增强它们的美感和实用性。其中，动态显示和隐藏信息的方式既可是自动的，也可是用户控制的。

第10章讲述jQuery令人印象深刻的扩展能力。读者将在理解和掌握3个优秀jQuery插件的基础上，从头开始构建自己的插件。

附录A提供了很多与jQuery、JavaScript以及通常的Web开发有关的内容丰富的网站信息。

附录B推荐了一些有用的第三程序和实用工具，用于在个人的开发环境中编辑和调试jQuery代码。

附录C讨论JavaScript语言的常见壁垒之一——闭包。学了本附录，你将会依赖闭包的强大威力而不是害怕它的副作用。

本书读者对象

本书适合想在自己的设计中添加交互元素的Web设计者，也适合想在自己的Web应用中创建最佳用户界面的开发者。

读者需要具备基本的HTML和CSS知识，并且应该熟悉JavaScript语法。但是，不需要有jQuery的知识，也不必拥有其他JavaScript库的使用经验。

本书约定

在本书中，读者会发现针对不同信息类型的文本样式。下面是这些样式的示例和解释。

书中的代码有3种样式。正文中提到的代码如下所示：“总之，`$()`和`.addClass()`对我们修改诗歌文本的外观已经足够了。”

代码如下所示：

```
$(document).ready(function() {  
    $('span:contains(language)').addClass('emphasized');  
});
```

当需要读者特别注意代码块中的某一部分时，相关的代码行或项将以粗体显示：


```
$(document).ready(function() {  
    $('a[@href$=".pdf"]').addClass('pdflink');  
});
```

新术语及重要词汇将以粗体字显示。对于在屏幕上看到的文字（例如在菜单或对话框中）在正文中会表示为：下一步是单击“**All**”按钮运行那些测试。

说明 重要的注意事项以说明体例给出。

提示 提示和技巧以提示体例给出。

读者反馈

我们始终欢迎来自读者的反馈意见。我们想知道读者对本书的看法，读者喜欢哪些内容或不喜欢哪些内容。读者真正深有感触的反馈，对于我们开发图书产品至关重要。如有反馈意见，请将电子邮件发送到contact@turingbook.com，不要忘记在邮件标题中注明你要评论的书名。

客户支持

为了让你的付出得到最大的回报，请注意以下信息。

本书的示例代码下载

访问<http://www.packtpub.com/support>，然后从图书列表中选择本书，以便下载本书的示例代码及其他资源。在选择了本书之后，请根据提示完成下载。

下载的文件中包含使用说明。

勘误

虽然我们会全力确保本书内容的准确性，但错误仍在所难免。如果你发现了本书中的错误（包括文字和代码错误），而且愿意向我们提交这些错误，我们会十分感激。这样一来，不仅可以减少其他读者的疑虑，也有助于本书后续版本的改进。要提交你发现的错误，请访问<http://www.packtpub.com/support>，选择你的图书，单击Submit Errata链接，然后输入勘误信息。经过验证之后，你提交的勘误信息就会添加到已有的勘误列表中。现有的勘误信息也可以通过访问<http://www.packtpub.com/support>并选择你的图书查看到。

疑难解答

如果你对本书的某些方面有疑问，请将电子邮件发送到contact@turingbook.com或questions@packtpub.com，我们会尽力解决。

目 录

第1章 jQuery 入门.....1

1.1 jQuery 能做什么.....1

1.2 jQuery 为什么如此出色.....2

1.3 第一个 jQuery 文档.....3

1.3.1 下载 jQuery.....3

1.3.2 设置 HTML 文档.....4

1.3.3 编写 jQuery 代码.....6

1.4 小结.....9

第2章 选择符——取得你想要的一切.....10

2.1 DOM.....10

2.2 工厂函数\$().....11

2.3 CSS 选择符.....11

2.4 XPath 选择符.....14

2.5 自定义选择符.....16

2.6 DOM 遍历方法.....18

2.7 访问 DOM 元素.....22

2.8 小结.....22

第3章 事件——扣动扳机.....23

3.1 在页面加载后执行任务.....23

3.1.1 代码执行的时机选择.....23

3.1.2 基于一个页面执行多个脚本.....24

3.1.3 缩短代码的简写方式.....25

3.2 简单的事件.....25

3.2.1 简单的样式转换器.....26

3.2.2 简写的事件.....32

3.3 复合事件.....33

3.3.1 显示和隐藏高级特性.....33

3.3.2 突出显示可单击的项.....34

3.3.3 事件的旅程.....36

3.3.4 事件冒泡的副作用.....37

3.4 限制和终止事件.....38

3.4.1 阻止事件冒泡.....38

3.4.2 移除事件处理程序.....40

3.5 模仿用户操作.....42

3.6 小结.....43

第4章 效果——为操作添加艺术性.....44

4.1 修改内联 CSS.....44

4.2 基本的隐藏和显示.....47

4.3 效果和速度.....49

4.4 多重效果.....50

4.4.1 构建具有动画效果的 show().....51

4.4.2 创建一种自定义的动画效果.....51

4.4.3 理解数字的含义.....53

4.4.4 改进自定义动画效果.....54

4.5 并发与排队效果.....55

4.5.1 处理一组元素.....55

4.5.2 处理多组元素.....57

4.6 简单概括.....60

4.7 小结.....60

第5章 DOM 操作——基于命令改变页面.....61

5.1 操作属性.....61

5.2 插入新元素.....64

5.3 移动元素.....66

5.3.1 标注、编号和链接到上下文.....69

5.3.2 插入脚注.....71

5.4 包装元素.....72

5.5 复制元素	72	7.8.1 筛选选项	150
5.5.1 复制的深度	73	7.8.2 同其他代码整合	152
5.5.2 通过复制创建突出引用	74	7.9 完成的代码	155
5.6 DOM 操作方法的简单归纳	79	7.10 小结	158
5.7 小结	80	第8章 构建功能型表单	159
第6章 AJAX——让网站与时俱进	81	8.1 渐进增强的表单设计	159
6.1 基于请求加载数据	81	8.1.1 图标符号	161
6.1.1 追加 HTML	83	8.1.2 必填字段的提示信息	162
6.1.2 操作 JavaScript 对象	85	8.1.3 根据条件显示的字段	166
6.1.3 加载 XML 文档	91	8.2 表单验证	168
6.2 选择数据格式	94	8.2.1 即时反馈	168
6.3 向服务器传递数据	95	8.2.2 最终检查	173
6.3.1 执行 GET 请求	96	8.3 复选框操作	174
6.3.2 执行 POST 请求	99	8.4 完成的代码	177
6.3.3 序列化表单	100	8.5 字段的占位符文本	180
6.4 关注请求	102	8.6 AJAX 自动完成	182
6.5 AJAX 和事件	105	8.6.1 服务器端代码	182
6.5.1 限定事件绑定函数的作用域	106	8.6.2 浏览器端脚本	183
6.5.2 利用事件冒泡	106	8.6.3 填充搜索字段	184
6.6 安全限制	107	8.6.4 键盘导航	184
6.7 小结	108	8.6.5 自动完成与实时搜索	188
第7章 表格操作	109	8.7 完成的代码	189
7.1 排序	110	8.8 输入掩码	191
7.1.1 服务器端排序	110	8.8.1 购物车表格结构	191
7.1.2 JavaScript 排序	111	8.8.2 拒绝非数字输入	194
7.2 分页	123	8.9 数字计算	194
7.2.1 服务器端分页	124	8.9.1 解析和格式化货币值	195
7.2.2 JavaScript 分页	125	8.9.2 处理小数位	196
7.3 完成的代码	129	8.9.3 其他计算	198
7.4 高级行条纹效果	132	8.10 删除商品	200
7.4.1 三色交替模式	135	8.11 修改送货信息	204
7.4.2 三行一组交替	137	8.12 完成的代码	207
7.5 突出显示行	141	8.13 小结	209
7.6 工具提示条	143	第9章 滑动和翻转	210
7.7 折叠和扩展	148	9.1 标题翻转效果	210
7.8 筛选	150	9.1.1 设置页面	210

9.1.2 取得新闻源.....	212	10.2 流行的插件.....	251
9.1.3 设置翻转效果.....	214	10.2.1 Dimensions	251
9.1.4 标题翻转函数.....	215	10.2.2 Form.....	253
9.1.5 悬停时暂停.....	217	10.2.3 Interface	255
9.1.6 从不同的域中取得新闻源.....	219	10.3 查找插件文档.....	259
9.1.7 附加的内部渐变效果.....	221	10.4 开发新插件.....	261
9.2 图像传送带.....	223	10.4.1 添加新的全局函数.....	261
9.2.1 设置页面.....	223	10.4.2 添加 jQuery 对象方法.....	263
9.2.2 通过 JavaScript 修改样式	225	10.4.3 DOM 遍历方法	264
9.2.3 通过单击滑移图像.....	226	10.4.4 添加新的简写方法.....	267
9.2.4 添加滑移效果.....	228	10.4.5 维护多事件日志.....	268
9.2.5 显示操作图标.....	229	10.4.6 添加选择符表达式.....	270
9.3 放大图像.....	232	10.4.7 创建缓动样式.....	272
9.3.1 隐藏大幅封面.....	234	10.4.8 做个好公民.....	274
9.3.2 更有价值的标记.....	236	10.5 小结	275
9.3.3 为封面放大添加动画效果	237		
9.4 完成的代码.....	244	附录 A 在线资源.....	277
9.5 小结	249	附录 B 开发工具	283
第 10 章 插件.....	250	附录 C JavaScript 闭包	287
10.1 使用插件.....	250		



*Up on the buzzer
Quick on the start
Let's go! Let's go! Let's go!
—Devo^①,
“Let's Go”*

今天的万维网是一个动态的环境，Web用户对网站的设计和功能都提出了高要求。为了构建有吸引力的交互式网站，开发者们借助于像jQuery这样的JavaScript库，实现了常见任务的自动化和复杂任务的简单化。jQuery库广受欢迎的一个原因，就是它对种类繁多的开发任务都能游刃有余地提供帮助。

由于jQuery的功能如此丰富多样，找到合适的切入点似乎都成了一项挑战。不过，这个库的设计秉承了一致性与对称性原则，它的大部分概念都是从HTML和CSS（Cascading Style Sheet，层叠样式表）的结构中借用而来的。鉴于很多Web开发人员对这两种技术比对JavaScript更有经验，所以编程经验不多的设计者能够快速学会使用该库。实际上，在本书开篇第1章中，只需3行代码就能编写一个有用的jQuery程序。另一方面，经验丰富的程序设计人员也会受益于这种概念上的一致性，通过学习后面的更高级内容，你会感受到这一点。

但是，在举例说明如何使用这个库之前，我们应该首先讨论一下选择它的理由。

1.1 jQuery 能做什么

jQuery库为Web脚本编程提供了通用的抽象层，使得它几乎适用于任何脚本编程的情形。由于它容易扩展而且不断有新插件面世增强它的功能，所以一本书根本无法涵盖它所有可能的用途和功能。抛开这些不谈，仅就其核心特性而言，jQuery能够满足下列需求：

① 本书各章都以著名“后朋克乐队Devo（退化）”的一首歌的歌词开头，歌词大意与相应章的名称或内容有微妙的对应。要了解“后朋克”的更多信息，请参考<http://baike.baidu.com/view/638729.htm>。要了解有关“朋克”的更多信息，请参考<http://baike.baidu.com/view/966.htm>。——译者注

- ❑ **取得页面中的元素。**如果不使用JavaScript库，遍历**DOM**（Document Object Model，文档对象模型）树，以及查找HTML文档结构中某个特殊的部分，必须编写很多行代码。jQuery为准确地获取需要检查或操纵的文档元素，提供了可靠而富有效率的选择符机制。
- ❑ **修改页面的外观。**CSS虽然为影响文档呈现的方式提供了一种强大的手段，但当所有浏览器不完全支持相同的标准时，单纯使用CSS就会显得力不从心。jQuery可以弥补这一不足，它提供了跨浏览器的标准解决方案。而且，即使在页面已经呈现之后，jQuery仍然能够改变文档中某个部分的类或者个别的样式属性。
- ❑ **改变页面的内容。**jQuery能够影响的范围并不局限于简单的外观变化，使用少量的代码，jQuery就能改变文档的内容。可以改变文本、插入或翻转图像、对列表重新排序，甚至，对HTML文档的整个结构都能重写和扩充——所有这些只需一个简单易用的API。
- ❑ **响应用户的页面操作。**即使是最强大和最精心设计的行为，如果我们无法控制它何时发生，那它也毫无用处。jQuery提供了截取形形色色的页面事件（比如用户单击一个链接）的适当方式，而不需要使用事件处理程序搞乱HTML代码。此外，它的事件处理API也消除了经常困扰Web开发人员的浏览器不一致性。
- ❑ **为页面添加动态效果。**为了实现某种交互式行为，设计者也必须向用户提供视觉上的反馈。jQuery中内置的一批淡入、擦除之类的效果，以及制作新效果的工具包，为此提供了便利。
- ❑ **无需刷新页面即可从服务器获取信息。**这种编程模式就是众所周知的AJAX（Asynchronous JavaScript and XML，异步JavaScript和XML），它能辅助Web开发人员创建出反应灵敏、功能丰富的网站。jQuery通过消除这一过程中的浏览器特定的复杂性，使开发人员得以专注于服务器端的功能设计。
- ❑ **简化常见的JavaScript任务。**除了这些完全针对文档的特性之外，jQuery也提供了对基本的JavaScript结构^①（例如迭代和数组操作等）的增强。

1.2 jQuery 为什么如此出色

随着近年来人们对动态HTML兴趣的复苏，催生了一大批JavaScript框架。有的特别专注于上述任务中的一项或两项，有的则试图以预打包的形式囊括各种可能的行为和动态效果。为了在维持上述各种特性的同时仍然保持紧凑，jQuery采用了如下策略：

- ❑ **利用CSS的优势。**通过将查找页面元素的机制构建于CSS选择符之上，jQuery继承了简明清晰地表达文档结构的方式。由于进行专业Web开发的一个必要条件是掌握CSS语法，因而jQuery成为希望向页面中添加行为的设计者们的切入点。

^① 本书在提到JavaScript中的函数、对象、语句、变量时，经常使用“结构”泛指。——译者注

- ❑ **支持扩展。**为了避免特性蠕变（feature creep）^①，jQuery将特殊情况下的用途归入插件当中。创建新插件的方法很简单，而且拥有完备的文档说明，这促进了大量有创意和有实用价值的模块的开发。甚至在下载的基本jQuery库文件当中，多数特性在内部都是通过插件架构实现的。而且，如有必要，可以移除这些内部插件，从而生成更小的库文件。
- ❑ **抽象浏览器不一致性。**Web开发领域中一个令人遗憾的事实是，每种浏览器对颁布的标准都有自己的一套不一致的实现方案。任何Web应用程序中都会包含一个用于处理这些平台间特性差异的重要组成部分。虽然不断发展的浏览器前景使得为某些高级特性提供浏览器中立的完美的基础代码（code base）不大可能，但jQuery添加一个抽象层来标准化常见的任务，从而有效地减少了代码量，同时，也极大地简化了这些任务。
- ❑ **总是面向集合。**当我们指示jQuery“找到带有‘collapsible’类的全部元素，然后隐藏它们”时，不需要循环遍历每一个返回的元素。相反，`.hide()`之类的方法被设计成自动操作对象集合，而不是单独的对象。这种称作**隐式迭代**（implicit iteration）的技术，使得大量的循环结构变得不再必要，从而大幅地减少代码量。
- ❑ **将多重操作集于一行。**为了避免过度使用临时变量或不必要的代码重复，jQuery在其多数方法中采用了一种被称作**连缀**（chaining）^②的编程模式。这种模式意味着基于一个对象进行的大多数操作的结果，都会返回这个对象自身，以便于为该对象应用下一次操作。

这些策略不仅确保了jQuery包的小型化——压缩文件约20 KB，同时，也为我们使用这个库的自定义代码保持简洁提供了技术保障。

jQuery库的适用性一方面归因于其设计理念，另一方面则得益于围绕这个开源项目涌现出的活跃社区的促进作用。jQuery用户聚集到一起，不仅会讨论插件的开发，也会讨论如何增强核心库。附录A中提供了很多jQuery开发人员可用的社区资源的详细信息。

除了为工程师提供灵活且稳健的系统之外，jQuery的最终产品对所有人都是免费的。而且，这个开源项目还具有**GNU Public License**（适合包含在很多其他开源项目中）和**MIT License**（便于在专有的软件中使用jQuery）的双重许可。

1.3 第一个 jQuery 文档

在了解了jQuery能够为我们提供的丰富特性之后，我们可以看一看这个库的实际应用了。

1.3.1 下载 jQuery

jQuery官方网站（<http://jquery.com/>）始终都包含与该库有关的最新代码和新闻的第一手资源。

① 术语feature creep也有人译为特性蔓延，指软件应用开发中过分强调新的功能以至于损害了其他的设计目标，例如简洁性、轻巧性、稳定性及错误出现率等。——译者注

② 术语chaining可译为链接，但为避免与人们耳熟能详的超级链接混淆（如常见的“单击链接”等），所以才译为更贴切的连缀。——译者注

为了开始学习，我们需要从官方网站上面下载一个jQuery库文件。官方网站在任何时候都会提供几种不同版本的jQuery库，但其中最适合我们的是该库最新的未压缩版。

使用jQuery库不需安装，只要把下载的库文件放到网站上的一个公共位置即可。因为JavaScript是一种解释型语言，所以使用它不必进行编译或者构建。无论什么时候，当我们想在某个页面上使用jQuery时，只需在相关的HTML文档中简单地引用该库文件的位置。

1.3.2 设置 HTML 文档

本书多数jQuery应用示例都包含以下3个部分：HTML文档、为该文档添加样式的CSS文件和为该文档添加行为的JavaScript文件。在本书的第一个例子中，我们使用一个包含图书内容提要的页面，同时，该页面中的很多部分都添加了相应的类。

```
<?xml version="1.0" encoding="UTF-8" ?>
<!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" xml:lang="en" lang="en">
  <head>
    <meta http-equiv="Content-Type" content="text/html;
                                          charset=utf-8"/>
    <title>Through the Looking-Glass</title>
    <link rel="stylesheet" href="alice.css" type="text/css"
                                          media="screen" />
    <script src="jquery.js" type="text/javascript"></script>
    <script src="alice.js" type="text/javascript"></script>
  </head>
  <body>
    <div id="container">
      <h1>Through the Looking-Glass</h1>
      <div class="author">by Lewis Carroll</div>
      <div class="chapter" id="chapter-1">
        <h2 class="chapter-title">1. Looking-Glass House</h2>
        <p>There was a book lying near Alice on the table, and while
          she sat watching the White King (for she was still a
          little anxious about him, and had the ink all ready to
          throw over him, in case he fainted again), she turned over
          the leaves, to find some part that she could read, <span
            class="spoken">"&mdash;for it's all in some language I
            don't know,"</span> she said to herself.</p>
        <p>It was like this.</p>
        <div class="poem">
          <h3 class="poem-title">YKCOWREBAJ</h3>
          <div class="poem-stanza">
            <div>sevot yhtils eht dna ,gillirb sawT'</div>
            <div>;ebaw eht ni elbmig dna eryg diD</div>
            <div>,sevogorob eht erew ysmim llA</div>
            <div>.ebargtuo shtar emom eht dnA</div>
          </div>
        </div>
      </div>
    </div>
```



```

<p>She puzzled over this for some time, but at last a bright
thought struck her. <span class="spoken">"Why, it's a
Looking-glass book, of course! And if I hold it up to a
glass, the words will all go the right way again."</span></p>
<p>This was the poem that Alice read.</p>
<div class="poem">
  <h3 class="poem-title">JABBERWOCKY</h3>
  <div class="poem-stanza">
    <div>'Twas brillig, and the slithy toves</div>
    <div>Did gyre and gimble in the wabe;</div>
    <div>All mimsy were the borogoves,</div>
    <div>And the mome raths outgrabe.</div>
  </div>
</div>
</div>
</body>
</html>

```

说明 这个示例文档在服务器上的实际布局并不重要。读者可以根据自己选中的组织结构，调整一个文件中指向另一个文件的引用。在本书多数例子中，都将使用相对路径引用文件（../images/foo.png），不使用绝对路径（/images/foo.png）。这样，我们不需要使用Web服务器就能在本地运行代码。

文档中，紧随常规的HTML开头代码之后的是加载样式表文件的代码。在这个例子中，我们使用了一个简单的样式表。

```

body {
  font: 62.5% Arial, Verdana, sans-serif;
}
h1 {
  font-size: 2.5em;
  margin-bottom: 0;
}
h2 {
  font-size: 1.3em;
  margin-bottom: .5em;
}
h3 {
  font-size: 1.1em;
  margin-bottom: 0;
}
.poem {
  margin: 0 2em;
}
.emphasized {
  font-style: italic;
  border: 1px solid #888;
  padding: 0.5em;
}

```