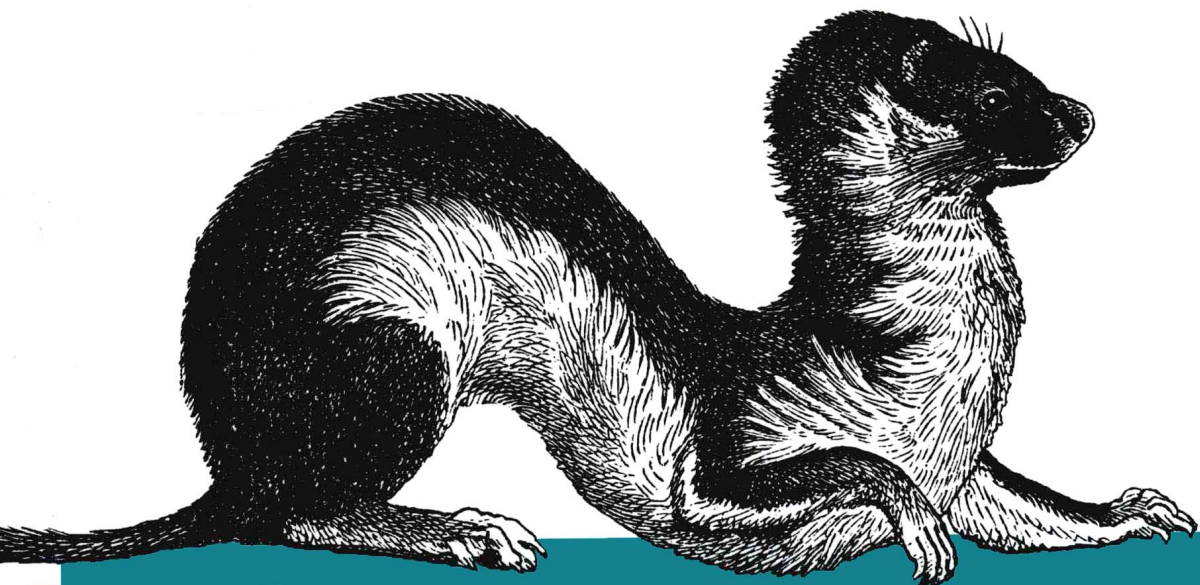


jQuery Cookbook (影印版)



jQuery Cookbook

O'REILLY®

東南大學出版社

jQuery Community Experts 著

jQuery Cookbook (影印版)

jQuery Cookbook

O'REILLY®

Beijing • Cambridge • Farnham • Köln • Sebastopol • Taipei • Tokyo

O'Reilly Media, Inc. 授权东南大学出版社出版

东南大学出版社

图书在版编目 (CIP) 数据

jQuery 参考手册 = jQuery Cookbook: 英文 / jQuery 社区
专家著. — 影印本. — 南京: 东南大学出版社, 2010.10
书名原文: jQuery Cookbook
ISBN 978-7-5641-2415-1

I. ① j… II. ① j… III. ① JAVA 语言—程序设计—英文
IV. ① TP312

中国版本图书馆 CIP 数据核字 (2010) 第 169611 号

江苏省版权局著作权合同登记

图字: 10-2010-282 号

©2009 by O'Reilly Media, Inc.

Reprint of the English Edition, jointly published by O'Reilly Media, Inc. and Southeast University Press, 2010. Authorized reprint of the original English edition, 2009 O'Reilly Media, Inc., the owner of all rights to publish and sell the same.

All rights reserved including the rights of reproduction in whole or in part in any form.

英文原版由 O'Reilly Media, Inc. 出版 2009。

英文影印版由东南大学出版社出版 2010。此影印版的出版和销售得到出版权和销售权的所有者——O'Reilly Media, Inc. 的许可。

版权所有, 未得书面许可, 本书的任何部分和全部不得以任何形式重制。

jQuery Cookbook (影印版)

出版发行: 东南大学出版社

地 址: 南京四牌楼 2 号 邮编: 210096

出 版 人: 江 汉

网 址: <http://www.seupress.com>

电子邮件: press@seu.edu.cn

印 刷: 扬中市印刷有限公司

开 本: 787 毫米 × 980 毫米 16 开本

印 张: 30 印张

字 数: 588 千字

版 次: 2010 年 10 月第 1 版

印 次: 2010 年 10 月第 1 次印刷

书 号: ISBN 978-7-5641-2415-1

印 数: 1~1600 册

定 价: 58.00 元 (册)

本社图书若有印装质量问题, 请直接与读者服务部联系。电话 (传真): 025-83792328

Foreword

When I first started work on building jQuery, back in 2005, I had a simple goal in mind: I wanted to be able to write a web application and have it work in all the major browsers—without further tinkering and bug fixing. It was a couple of months before I had a set of utilities that were stable enough to achieve that goal for my personal use. I thought I was relatively done at this point; little did I know that my work was just beginning.

Since those simple beginnings, jQuery has grown and adapted as new users use the library for their projects. This has proven to be the most challenging part of developing a JavaScript library; while it is quite easy to build a library that'll work for yourself or a specific application, it becomes incredibly challenging to develop a library that'll work in as many environments as possible (old browsers, legacy web pages, and strange markup abound). Surprisingly, even as jQuery has adapted to handle more use cases, most of the original API has stayed intact.

One thing I find particularly interesting is to see how developers use jQuery and make it their own. As someone with a background in computer science, I find it quite surprising that so many designers and nonprogrammers find jQuery to be compelling. Seeing how they interact with the library has given me a better appreciation of simple API design. Additionally, seeing many advanced programmers take jQuery and develop large, complex applications with it has been quite illuminating. The best part of all of this, though, is the ability to learn from everyone who uses the library.

A side benefit of using jQuery is its extensible plugin structure. When I first developed jQuery, I was sure to include some simple ways for developers to extend the API that it provided. This has blossomed into a large and varied community of plugins, encompassing a whole ecosystem of applications, developers, and use cases. Much of jQuery's growth has been fueled by this community—without it, the library wouldn't be where it is today, so I'm glad that there are chapters dedicated to some of the most interesting plugins and what you can do with them. One of the best ways to expand your preconceived notion of what you can do with jQuery is to learn and use code from the jQuery plugin community.

This is largely what makes something like a cookbook so interesting: it takes the cool things that developers have done, and have learned, in their day-to-day coding and distills it to bite-sized chunks for later consumption. Personally, I find a cookbook to be one of the best ways to challenge my preconceived notions of a language or library. I love seeing cases where an API that I thought I knew well is turned around and used in new and interesting ways. I hope this book is able to serve you well, teaching you new and interesting ways to use jQuery.

—John Resig
Creator, Lead Developer, jQuery

Contributors

Chapter Authors

Jonathan Sharp has been passionate about the Internet and web development since 1996. Over the years that have followed, he has worked for startups and for Fortune 500 corporations. Jonathan founded Out West Media, LLC, in greater Omaha, Nebraska, and provides frontend engineering and architecture services with a focus on custom XHTML, CSS, and jQuery development. Jonathan is a jQuery core team member and an author and presenter when not coding. Jonathan is most grateful for his wife, Erin; daughter, Noel; two dogs, and two horses.

Rob Burns develops interactive web applications at A Mountain Top, LLC. For the past 12 years he has been exploring website development using a wide range of tools and technologies. In his spare time, he enjoys natural-language processing and the wealth of opportunity in open source software projects.

Rebecca Murphey is an independent frontend architecture consultant, crafting custom frontend solutions that serve as the glue between server and browser. She also provides training in frontend development, with an emphasis on the jQuery library. She lives with her partner, two dogs, and two cats in Durham, North Carolina.

Ariel Flesler is a web developer and a video game programmer. He's been contributing to jQuery since January 2007 and joined the core team in May 2008. He is 23 years old and was born in Buenos Aires, Argentina. He's studying at the National Technological University (Argentina) and is hoping to become a systems analyst by 2010 and a systems engineer by 2012. He started working as an ASP.NET(C#) programmer and then switched to client-side development of XHTML sites and Ajax applications. He's currently working at QB9 where he develops AS3-based casual games and MMOs.

Cody Lindley is a Christian, husband, son, father, brother, outdoor enthusiast, and professional client-side engineer. Since 1997 he has been passionate about HTML, CSS, JavaScript, Flash, interaction design, interface design, and HCI. He is most well known in the jQuery community for the creation of ThickBox, a modal/dialog solution. In 2008 he officially joined the jQuery team as an evangelist. His current focus has been

on client-side optimization techniques as well as speaking and writing about jQuery. His website is <http://www.codylindley.com>.

Remy Sharp is a developer, author, speaker, and blogger. Remy started his professional web development career in 1999 as the sole developer for a finance website and, as such, was exposed to all aspects of running the website during, and long after, the dotcom boom. Today he runs his own development company called Left Logic in Brighton, UK, writing and coding JavaScript, jQuery, HTML 5, CSS, PHP, Perl, and anything else he can get his hands on.

Mike Hostetler is an inventor, entrepreneur, programmer, and proud father. Having worked with web technologies since the mid-1990s, Mike has had extensive experience developing web applications with PHP and JavaScript. Currently, Mike works at the helm of A Mountain Top, LLC, a web technology consulting firm in Denver, Colorado. Heavily involved in open source, Mike is a member of the jQuery core team, leads the Qcubed PHP5 Framework project, and participates in the Drupal project. When not in front of a computer, Mike enjoys hiking, fly fishing, snowboarding, and spending time with his family.

Ralph Whitbeck is a graduate of the Rochester Institute of Technology and is currently a senior developer for BrandLogic Corporation (<http://brandlogic.com>) in Rochester, New York. His responsibilities at BrandLogic include interface design, usability testing, and web and application development. Ralph is able to program complex web application systems in ASP.NET, C#, and SQL Server and also uses client-side technologies such as XHTML, CSS, and JavaScript/jQuery in order to implement client-approved designs. Ralph officially joined the jQuery team as an evangelist in October 2009. Ralph enjoys spending time with his wife, Hope, and his three boys, Brandon, Jordan, and Ralphie. You can find out more about Ralph on his personal blog (<http://ralphwhitbeck.com>).

Nathan Smith is a goofy guy who has been building websites since late last century. He enjoys hand-coding HTML, CSS, and JavaScript. He also dabbles in design and information architecture. He has written for online and paper publications such as Adobe Developer Center, Digital Web, and .NET Magazine. He has spoken at venues including Adobe MAX, BibleTech, Drupal Camp, Echo Conference, Ministry 2.0, Refresh Dallas, and Webmaster Jam Session. Nathan works as a UX developer at FellowshipTech.com. He holds a Master of Divinity degree from Asbury Theological Seminary. He started Godbit.com, a community resource aimed at helping churches and ministries make better use of the Web. He also created the 960 Grid System (<http://www.960.gs>), a framework for sketching, designing, and coding page layouts.

Brian Cherne is a software developer with more than a decade of experience blue-printing and building web-based applications, kiosks, and high-traffic e-commerce websites. He is also the author of the hoverIntent jQuery plugin. When not geeking out with code, Brian can be found ballroom dancing, practicing martial arts, or studying Russian culture and language.

Jörn Zaefferer is a professional software developer from Cologne, Germany. He creates application programming interfaces (APIs), graphical user interfaces (GUIs), software architectures, and databases, for both web and desktop applications. His work focuses on the Java platform, while his client-side scripting revolves around jQuery. He started contributing to jQuery in mid-2006 and has since cocreated and maintained QUnit, jQuery's unit testing framework; released and maintained a half dozen very popular jQuery plugins; and contributed to jQuery books as both author and tech reviewer. He is also a lead developer for jQuery UI.

James Padolsey is an enthusiastic web developer and blogger based in London, UK. He's been crazy about jQuery since he first discovered it; he's written tutorials teaching it, articles and blog posts discussing it, and plenty of plugins for the community. James' plans for the future include a computer science degree from the University of Kent and a career that allows him to continually push boundaries. His website is <http://james.padolsey.com>.

Scott González is a web application developer living in Raleigh, North Carolina, who enjoys building highly dynamic systems and flexible, scalable frameworks. He has been contributing to jQuery since 2007 and is currently the development lead for jQuery UI, jQuery's official user interface library. Scott also writes tutorials about jQuery and jQuery UI on nemikor.com and speaks about jQuery at conferences.

Michael Geary started developing software when editing code meant punching a paper tape on a Teletype machine, and "standards-compliant" meant following ECMA-10 Standard for Data Interchange on Punched Tape. Today Mike is a web and Android developer with a particular interest in writing fast, clean, and simple code, and he enjoys helping other developers on the jQuery mailing lists. Mike's recent projects include a series of 2008 election result and voter information maps for Google; and StrataLogic, a mashup of traditional classroom wall maps and atlases overlaid on Google Earth. His website is <http://mg.to>.

Maggie Wachs, Scott Jehl, Todd Parker, and Patty Toland are Filament Group. Together, they design and develop highly functional user interfaces for consumer- and business-oriented websites, wireless devices, and installed and web-based applications, with a specific focus on delivering intuitive and usable experiences that are also broadly accessible. They are sponsor and design leads of the jQuery UI team, for whom they designed and developed ThemeRoller.com, and they actively contribute to ongoing development of the official jQuery UI library and CSS Framework.

Richard D. Worth is a web UI developer. He is the release manager for jQuery UI and one of its longest-contributing developers. He is author or coauthor of the Dialog, Progressbar, Selectable, and Slider plugins. Richard also enjoys speaking and consulting on jQuery and jQuery UI around the world. Richard is raising a growing family in Northern Virginia (Washington, D.C. suburbs) with his lovely wife, Nancy. They have been blessed to date with three beautiful children: Naomi, Asher, and Isaiah. Richard's website is <http://rdworth.org/>.

Tech Editors

Karl Swedberg, after having taught high school English, edited copy for an advertising agency, and owned a coffee house, began his career as a web developer four years ago. He now works for Fusionary Media in Grand Rapids, Michigan, where he specializes in client-side scripting and interaction design. Karl is a member of the jQuery project team and coauthor of *Learning jQuery 1.3* and *jQuery Reference Guide* (both published by Packt). You can find some of his tips and tutorials at <http://www.learningjquery.com>.

Dave Methvin is the chief technology officer at PCPitstop.com and one of the founding partners of the company. He has been using jQuery since 2006, is active on the jQuery help groups, and has contributed several popular jQuery plugins including Corner and Splitter. Before joining PC Pitstop, Dave served as executive editor at both *PC Tech Journal* and *Windows Magazine*, where he wrote a column on JavaScript. He continues to write for several PC-related websites including InformationWeek. Dave holds bachelor's and master's degrees in computer science from the University of Virginia.

David Serduke is a frontend programmer who is recently spending much of his time server side. After programming for many years, he started using jQuery in late 2007 and shortly after joined the jQuery core team. David is currently creating websites for financial institutions and bringing the benefits of jQuery to ASP.NET enterprise applications. David lives in northern California where he received a bachelor's degree from the University of California at Berkeley in electrical engineering and an MBA from St. Mary's College.

Scott Mark is an enterprise application architect at Medtronic. He works on web-based personalized information portals and transactional applications with an eye toward maintaining high usability in a regulated environment. His key interest areas at the moment are rich Internet applications and multitouch user interface technologies. Scott lives in Minnesota with his lovely wife, two sons, and a black lab. He blogs about technology at <http://scottmark.wordpress.com> and long-distance trail running at <http://runlikemonkey.com>.

Preface

The jQuery library has taken the frontend development world by storm. Its dead-simple syntax makes once-complicated tasks downright trivial—enjoyable, even. Many a developer has been quickly seduced by its elegance and clarity. If you’ve started using the library, you’re already adding rich, interactive experiences to your projects.

Getting started is easy, but as is the case with many of the tools we use to develop websites, it can take months or even years to fully appreciate the breadth and depth of the jQuery library. The library is chock-full of features you might never have known to wish for. Once you know about them, they can dramatically change how you approach the problems you’re called upon to solve.

The goal of this cookbook is to expose you, dear reader, to the patterns and practices of some of the leading frontend developers who use jQuery in their everyday projects. Over the course of 18 chapters, they’ll guide you through solutions to problems that range from straightforward to complex. Whether you’re a jQuery newcomer or a grizzled JavaScript veteran, you’re likely to gain new insight into harnessing the full power of jQuery to create compelling, robust, high-performance user interfaces.

Who This Book Is For

Maybe you’re a designer who is intrigued by the interactivity that jQuery can provide. Maybe you’re a frontend developer who has worked with jQuery before and wants to see how other people accomplish common tasks. Maybe you’re a server-side developer who’s frequently called upon to write client-side code.

Truth be told, this cookbook will be valuable to anyone who works with jQuery—or who hopes to work with jQuery. If you’re just starting out with the library, you may want to consider pairing this book with *Learning jQuery 1.3* from Packt, or *jQuery in Action* from Manning. If you’re already using jQuery in your projects, this book will serve to enhance your knowledge of the library’s features, hidden gems, and idiosyncrasies.

What You'll Learn

We'll start out by covering the basics and general best practices—including jQuery in your page, making selections, and traversing and manipulation. Even frequent jQuery users are likely to pick up a tip or two. From there, we move on to real-world use cases, walking you through tried-and-true (and tested) solutions to frequent problems involving events, effects, dimensions, forms, and user interface elements (with and without the help of jQuery UI). At the end, we'll take a look at testing your jQuery applications and integrating jQuery into complex sites.

Along the way, you'll learn strategies for leveraging jQuery to solve problems that go far beyond the basics. We'll explore how to make the most of jQuery's event management system, including custom events and custom event data; how to progressively enhance forms; how to position and reposition elements on the page; how to create user interface elements such as tabs, accordions, and modals from scratch; how to craft your code for readability and maintainability; how to optimize your code to ease testing, eliminate bottlenecks, and ensure peak performance; and more.

Because this is a cookbook and not a manual, you're of course welcome to cherry-pick the recipes you read; the individual recipes alone are worth the price of admission. As a whole, though, the book provides a rare glimpse into the problem-solving approaches of some of the best and brightest in the jQuery community. With that in mind, we encourage you to at least skim it from front to back—you never know which line of code will provide the "Aha!" moment you need to take your skills to the next level.

jQuery Style and Conventions

jQuery places a heavy emphasis on *chaining*—calling methods on element selections in sequence, confident in the knowledge that each method will give you back a selection of elements you can continue to work with. This pattern is explained in depth in Chapter 1—if you're new to the library, you'll want to understand this concept, because it is used heavily in subsequent chapters.

jQuery's features are organized into a handful of simple categories: core functionality, selecting, manipulating, traversing, CSS, attributes, events, effects, Ajax, and utilities. Learning these categories, and how methods fit into them, will greatly enhance your understanding of the material in this book.

One of the best practices this book will cover is the concept of storing element selections in a variable, rather than making the same selection repeatedly. When a selection is stored in a variable, it is commonplace for that variable to begin with the \$ character, indicating that it is a jQuery object. This can make code easier to read and maintain, but it should be understood that starting the variable name with the \$ character is merely a convention; it carries no special meaning, unlike in other languages such as PHP.

In general, the code examples in this book strive for clarity and readability over compactness, so the examples may be more verbose than is strictly necessary. If you see an opportunity for optimization, you should not hesitate to take it. At the same time, you'll do well to strive for clarity and readability in your own code and use minification tools to prepare your code for production use.

Other Options

If you're looking for other jQuery resources, here are some we recommend:

- *Learning jQuery 1.3*, by Jonathan Chaffer, Karl Swedberg, and John Resig (Packt)
- *jQuery in Action*, by Bear Bibeault, Yehuda Katz, and John Resig (Manning)
- *jQuery UI 1.6: The User Interface Library for jQuery*, by Dan Wellman (Packt)

If You Have Problems Making Examples Work

Before you check anything else, ensure that you are loading the jQuery library on the page—you'd be surprised how many times this is the solution to the "It's not working!" problem. If you are using jQuery with another JavaScript library, you may need to use `jQuery.noConflict()` to make it play well with others. If you're loading scripts that require the presence of jQuery, make sure you are loading them after you've loaded the jQuery library.

Much of the code in this book requires the document to be "ready" before JavaScript can interact with it. If you've included code in the head of the document, make sure your code is enclosed in `$(document).ready(function() { ... });` so that it knows to wait until the document is ready for interaction.

Some of the features discussed in this book are available only in jQuery 1.3 and later. If you are upgrading from an older version of jQuery, make sure you've upgraded any plugins you're using as well—outdated plugins can lead to unpredictable behavior.

If you're having difficulty getting an example to work in an existing application, make sure you can get the example working on its own before trying to integrate it with your existing code. If that works, tools such as Firebug for the Firefox browser can be useful in identifying the source of the problem.

If you're including a minified version of jQuery and running into errors that point to the jQuery library itself, you may want to consider switching to the full version of jQuery while you are debugging the issue. You'll have a much easier time locating the line that is causing you trouble, which will often lead you in the direction of a solution.

If you're still stuck, consider posting your question to the jQuery Google group. Many of this book's authors are regular participants in the group, and more often than not, someone in the group will be able to offer useful advice. The #jquery IRC channel on Freenode is another valuable resource for troubleshooting issues.

If none of this works, it's possible we made a mistake. We worked hard to test and review all of the code in the book, but errors do creep through. Check the errata (described in the next section) and download the sample code, which will be updated to address any errata we discover.

If You Like (or Don't Like) This Book

If you like—or don't like—this book, by all means, please let people know. Amazon reviews are one popular way to share your happiness (or lack of happiness), or you can leave reviews at the site for the book:

<http://oreilly.com/catalog/9780596159771/>

There's also a link to errata there. Errata gives readers a way to let us know about typos, errors, and other problems with the book. That errata will be visible on the page immediately, and we'll confirm it after checking it out. O'Reilly can also fix errata in future printings of the book and on Safari, making for a better reader experience pretty quickly. We hope to keep this book updated for future versions of jQuery, and will also incorporate suggestions and complaints into future editions.

Conventions Used in This Book

The following typographical conventions are used in this book:

Italic

Indicates Internet addresses, such as domain names and URLs, and new items where they are defined.

Constant width

Indicates command lines and options that should be typed verbatim; names and keywords in programs, including method names, variable names, and class names; and HTML element tags, switches, attributes, keys, functions, types, namespaces, modules, properties, parameters, values, objects, events, event handlers, macros, the contents of files, or the output from commands.

Constant width bold

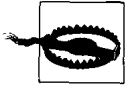
Indicates emphasis in program code lines.

Constant width *italic*

Indicates text that should be replaced with user-supplied values.



This icon signifies a tip, suggestion, or general note.



This icon indicates a warning or caution.

Using Code Examples

This book is here to help you get your job done. In general, you may use the code in this book in your programs and documentation. You do not need to contact us for permission unless you're reproducing a significant portion of the code. For example, writing a program that uses several chunks of code from this book does not require permission. Answering a question by citing this book and quoting example code does not require permission. Selling or distributing a CD-ROM of examples from O'Reilly books *does* require permission. Incorporating a significant amount of example code from this book into your product's documentation *does* require permission.

We appreciate, but do not require, attribution. An attribution usually includes the title, author, publisher, and ISBN. For example: “*jQuery Cookbook*, by Cody Lindley. Copyright 2010 Cody Lindley, 978-0-596-15977-1.” If you feel your use of code examples falls outside fair use or the permission given above, feel free to contact us at permissions@oreilly.com.

Safari® Books Online

Safari Books Online is an on-demand digital library that lets you easily search over 7,500 technology and creative reference books and videos to find the answers you need quickly.

With a subscription, you can read any page and watch any video from our library online. Read books on your cell phone and mobile devices. Access new titles before they are available for print, and get exclusive access to manuscripts in development and post feedback for the authors. Copy and paste code samples, organize your favorites, download chapters, bookmark key sections, create notes, print out pages, and benefit from tons of other time-saving features.

O'Reilly Media has uploaded this book to the Safari Books Online service. To have full digital access to this book and others on similar topics from O'Reilly and other publishers, sign up for free at <http://my.safaribooksonline.com>.

How to Contact Us

Please address comments and questions concerning this book to the publisher:

O'Reilly Media, Inc.
1005 Gravenstein Highway North
Sebastopol, CA 95472

800-998-9938 (in the United States or Canada)
707-829-0515 (international or local)
707-829-0104 (fax)

To comment or ask technical questions about this book, send email to:

bookquestions@oreilly.com

For more information about our books, conferences, Resource Centers, and the O'Reilly Network, see our website at:

http://oreilly.com

—Rebecca Murphey and Cody Lindley

Table of Contents

Foreword	xi
Contributors	xiii
Preface	xvii
1. jQuery Basics	1
1.1 Including the jQuery Library Code in an HTML Page	9
1.2 Executing jQuery/JavaScript Coded After the DOM Has Loaded but Before Complete Page Load	11
1.3 Selecting DOM Elements Using Selectors and the jQuery Function	13
1.4 Selecting DOM Elements Within a Specified Context	15
1.5 Filtering a Wrapper Set of DOM Elements	16
1.6 Finding Descendant Elements Within the Currently Selected Wrapper Set	18
1.7 Returning to the Prior Selection Before a Destructive Change	19
1.8 Including the Previous Selection with the Current Selection	21
1.9 Traversing the DOM Based on Your Current Context to Acquire a New Set of DOM Elements	22
1.10 Creating, Operating on, and Inserting DOM Elements	23
1.11 Removing DOM Elements	25
1.12 Replacing DOM Elements	26
1.13 Cloning DOM Elements	27
1.14 Getting, Setting, and Removing DOM Element Attributes	29
1.15 Getting and Setting HTML Content	31
1.16 Getting and Setting Text Content	32
1.17 Using the \$ Alias Without Creating Global Conflicts	33
2. Selecting Elements with jQuery	35
2.1 Selecting Child Elements Only	36
2.2 Selecting Specific Siblings	37

2.3	Selecting Elements by Index Order	39
2.4	Selecting Elements That Are Currently Animating	41
2.5	Selecting Elements Based on What They Contain	42
2.6	Selecting Elements by What They Don't Match	43
2.7	Selecting Elements Based on Their Visibility	43
2.8	Selecting Elements Based on Attributes	44
2.9	Selecting Form Elements by Type	46
2.10	Selecting an Element with Specific Characteristics	47
2.11	Using the Context Parameter	48
2.12	Creating a Custom Filter Selector	50
3.	Beyond the Basics	53
3.1	Looping Through a Set of Selected Results	53
3.2	Reducing the Selection Set to a Specified Item	56
3.3	Convert a Selected jQuery Object into a Raw DOM Object	59
3.4	Getting the Index of an Item in a Selection	62
3.5	Making a Unique Array of Values from an Existing Array	64
3.6	Performing an Action on a Subset of the Selected Set	67
3.7	Configuring jQuery Not to Conflict with Other Libraries	69
3.8	Adding Functionality with Plugins	72
3.9	Determining the Exact Query That Was Used	74
4.	jQuery Utilities	77
4.1	Detecting Features with jQuery.support	77
4.2	Iterating Over Arrays and Objects with jQuery.each	79
4.3	Filtering Arrays with jQuery.grep	80
4.4	Iterating and Modifying Array Entries with jQuery.map	81
4.5	Combining Two Arrays with jQuery.merge	81
4.6	Filtering Out Duplicate Array Entries with jQuery.unique	82
4.7	Testing Callback Functions with jQuery.isFunction	82
4.8	Removing Whitespace from Strings or Form Values with jQuery.trim	83
4.9	Attaching Objects and Data to DOM with jQuery.data	84
4.10	Extending Objects with jQuery.extend	85
5.	Faster, Simpler, More Fun	87
5.1	That's Not jQuery, It's JavaScript!	87
5.2	What's Wrong with \$(this)?	88
5.3	Removing Redundant Repetition	91
5.4	Formatting Your jQuery Chains	92
5.5	Borrowing Code from Other Libraries	94
5.6	Writing a Custom Iterator	96
5.7	Toggling an Attribute	99