
GREASEMONKEY HACKS

*Tips & Tools for Remixing
the Web with Firefox*



Mark Pilgrim

*Foreword by Aaron Boodman,
creator and lead developer of Greasemonkey*

O'REILLY®

GREASEMONKEY HACKS

Tips & Tools for Remixing the Web with Firefox



Greasemonkey is a Firefox extension that allows you to alter the web pages you visit. You can use it to make web sites more usable, more accessible, or a little less annoying. You can fix bugs and add features to web applications such as Gmail, Yahoo! Search, and Amazon.com. You can do anything

JavaScript can do—and you can do it on other people’s sites! *Greasemonkey Hacks* provides complete, fully developed user scripts you can use to modify web pages, the tools to customize these scripts, and the guidance to develop your own scripts from scratch. Whether you’re an experienced web developer or a novice with no JavaScript experience, you’ll learn how to:

- Install, configure, and debug your first Greasemonkey script
- Insert links into web pages, fix broken pop-up links, and follow links without clicking them
- Beautify the Web by enhancing fonts, images, tooltips, lists, and tables
- Intercept and modify Web forms, generate developer reports, and debug Ajax web applications
- Make search engines auto-complete your search terms, prefetch your results, and remember where you’ve been—without invading your privacy!
- Add accessibility features that make sites easier to read and navigate
- Download embedded movies, automate site registrations, and route around brain-dead browser sniffers

“Without [Mark Pilgrim’s] simple explanations and painstaking documentation, the Greasemonkey community could not have grown into the worldwide collection of programmers, IT professionals, and hobbyists that it is today. If you read and understand the examples in this book...[the] next time you find yourself frustrated by a broken web site, you won’t have to live with it. You’ll have the tools and knowledge to fix it yourself.”

—Aaron Boodman, creator and lead developer of Greasemonkey

Got a Hack? Go to: hacks.oreilly.com

US \$24.95

CAN \$34.95

ISBN: 978-0-596-10245-6



O'REILLY®
www.oreilly.com

Safari Includes
BOOKS ONLINE FREE 45-Day
ENABLED Online Edition

GREASEMONKEY HACKS™

Other resources from O'Reilly

Related titles	Firefox Hacks™	Ambient Findability
	JavaScript and DHTML Cookbook™	Web Design in a Nutshell
	Webmaster in a Nutshell	Yahoo! Hacks™
	Google Hacks™	eBay Hacks™
	Don't Click on the Blue E!	Windows XP Hacks™

Hacks Series Home *hacks.oreilly.com* is a community site for developers and power users of all stripes. Readers learn from each other as they share their favorite tips and tools for Mac OS X, Linux, Google, Windows XP, and more.

oreilly.com *oreilly.com* is more than a complete catalog of O'Reilly books. You'll also find links to news, events, articles, weblogs, sample chapters, and code examples.



oreillynet.com is the essential portal for developers interested in open and emerging technologies, including new platforms, programming languages, and operating systems.

Conferences O'Reilly brings diverse innovators together to nurture the ideas that spark revolutionary industries. We specialize in documenting the latest tools and systems, translating the innovator's knowledge into useful skills for those in the trenches. Visit *conferences.oreilly.com* for our upcoming events.



Safari Bookshelf (*safari.oreilly.com*) is the premier online reference library for programmers and IT professionals. Conduct searches across more than 1,000 books. Subscribers can zero in on answers to time-critical questions in a matter of seconds. Read the books on your Bookshelf from cover to cover or simply flip to the page you need. Try it today with a free trial.

GREASEMONKEY HACKS™

Mark Pilgrim

O'REILLY®

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

Greasemonkey Hacks™

by Mark Pilgrim

Copyright © 2006 O'Reilly Media, Inc. All rights reserved.
Printed in the United States of America.

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

O'Reilly books may be purchased for educational, business, or sales promotional use. Online editions are also available for most titles (*safari.oreilly.com*). For more information, contact our corporate/institutional sales department: (800) 998-9938 or *corporate@oreilly.com*.

Editor:	Brian Sawyer	Production Editor:	Mary Anne Weeks Mayo
Series Editor:	Rael Dornfest	Cover Designer:	Marcia Friedman
Executive Editor:	Dale Dougherty	Interior Designer:	David Futato

Printing History:


November 2005: First Edition.

Nutshell Handbook, the Nutshell Handbook logo, and the O'Reilly logo are registered trademarks of O'Reilly Media, Inc. The *Hacks* series designations, *Greasemonkey Hacks*, the image of a hand beater, and related trade dress are trademarks of O'Reilly Media, Inc.

Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this book, and O'Reilly Media, Inc. was aware of a trademark claim, the designations have been printed in caps or initial caps.

While every precaution has been taken in the preparation of this book, the publisher and author assume no responsibility for errors or omissions, or for damages resulting from the use of the information contained herein.

Small print: The technologies discussed in this publication, the limitations on these technologies that technology and content owners seek to impose, and the laws actually limiting the use of these technologies are constantly changing. Thus, some of the hacks described in this publication may not work, may cause unintended harm to systems on which they are used, or may not be consistent with applicable user agreements. Your use of these hacks is at your own risk, and O'Reilly Media, Inc. disclaims responsibility for any damage or expense resulting from their use. In any event, you should take care that your use of these hacks does not violate any applicable laws, including copyright laws.

 This book uses RepKover™, a durable and flexible lay-flat binding.

ISBN: 0-596-10165-1

[C]

For Wesley

Contents

Foreword	xi
Credits	xiii
Preface	xix
Chapter 1. Getting Started	1
1. Install a User Script	2
2. Provide a Default Configuration	6
3. Master the @include and @exclude Directives	8
4. Prevent a User Script from Executing	10
5. Configure a User Script	13
6. Add or Remove Content on a Page	16
7. Alter a Page's Style	19
8. Master XPath Expressions	21
9. Develop a User Script "Live"	23
10. Debug a User Script	25
11. Embed Graphics in a User Script	29
12. Avoid Common Pitfalls	33
Chapter 2. Linkmania!	45
13. Turn Naked URLs into Hyperlinks	45
14. Force Offsite Links to Open in a New Window	49
15. Fix Broken Pop-up Links	51
16. Remove URL Redirections	55
17. Warn Before Opening PDF Links	58

18. Avoid the Slashdot Effect	60
19. Convert UPS and FedEx Tracking Numbers to Links	64
20. Follow Links Without Clicking Them	69
Chapter 3. Beautifying the Web	73
21. Banish the Scourge of Arial	73
22. Add Stripes to Data Tables	76
23. Straighten Smart Quotes	79
24. Convert Graphical Smileys to Text	81
25. Make Amazon Product Images Larger	84
26. Convert Straight Quotes	86
27. Add Dynamic Highlighting to Tables	89
28. Make Pop-up Titles Prettier	93
Chapter 4. Web Forms	100
29. Display Form Actions in a Tool Tip	100
30. Show Hidden Form Fields	103
31. Identify Password Fields	104
32. Allow Password Remembering	107
33. Confirm Before Closing Modified Pages	109
34. Resize Text Input Fields with the Keyboard	112
35. Enter Textile Markup in Web Forms	116
36. Select Multiple Checkboxes	121
37. Keep Track of Secure Site Passwords	124
38. Automatically Log into Web Mail and Other Sites	136
39. Build Calendar Events	139
Chapter 5. Developer Tools	159
40. Remove All Page Styles on Selected Sites	159
41. Refresh Pages Automatically	163
42. Make External Stylesheets Clickable	164
43. Show Image Information	168
44. Filter Code Examples on MSDN	171
45. Intercept and Modify Form Submissions	175
46. Trace XMLHttpRequest Activity	186

Chapter 6. Search	192
47. Add a Site Search	192
48. Remove Spammy Domains from Search Results	195
49. Find Similar Images	199
50. Search Wikipedia with Google Site Search	202
51. Link to Other Search Engines from Google	204
52. Prefetch Yahoo! Search Results	208
53. Browse the Web Through Google's Cache	212
54. Add More Book Reviews to Google Print	216
55. Autocomplete Search Terms as You Type	218
56. Highlight Search Terms	220
57. Remember Recent Google Searches	224
58. Add Keyboard Shortcuts to Google Search Results	230
59. Use Recent Searches and Google SearchKeys Together	233
Chapter 7. Web Mail	239
60. Force Gmail to Use a Secure Connection	239
61. Warn Before Replying to Multiple Recipients in Gmail	241
62. Warn Before Sending Gmail Messages with Missing Attachments	244
63. Compose Your Mail in Gmail	246
64. Add a Delete Button to Gmail	250
65. Select Your Yahoo! ID from a List	256
66. Add Saved Searches to Gmail	260
Chapter 8. Accessibility	274
67. Highlight Images Without Alternate Text	275
68. Add an Access Bar with Keyboard Shortcuts	277
69. Remove Conflicting Keyboard Shortcuts	281
70. Make Image alt Text Visible	282
71. Add a Table of Contents to Long Pages	285
72. Use Real Headers on Google Web Search	289
73. Add a Toolbar to Zoom Images Easily	292
74. Make Apache Directory Listing Prettier	298
75. Add a Text-Sizing Toolbar to Web Forms	305
76. Make Google More Accessible for Low-Vision Users	311

Chapter 9. Taking Back the Browser	320
77. Reenable Context Menus on Sites That Disable Them	321
78. Bypass Weight Watchers' Browser Checker	323
79. Easily Download Embedded Movies	325
80. Break Out of Frames	327
81. Disable Targets for Downloads	330
82. Automatically Link to Printer-Friendly Versions	332
83. Restore Functionality in Google Print	334
84. Bypass Annoying Site Registrations	338
Chapter 10. Syndication	350
85. Automatically Display Unread Items in Bloglines	350
86. Zap Ugly XML Buttons	352
87. Squeeze More Feeds into the Bloglines Sidebar	355
88. Automatically Collect Syndicated Feeds	358
89. Syndicate Encrypted Content	365
Chapter 11. Site Integration	379
90. Translate Any Web Page	379
91. Warn Before Buying an Album	383
92. Find Out Who's Reading What You're Reading	386
93. Add Wikipedia Links to Any Web Page	393
94. Compare Book Prices	400
Chapter 12. Those Not Included in This Classification	409
95. Maximize HomestarRunner Cartoons	409
96. Refine Your Google Search	412
97. Check Whether Pages Really Validate	415
98. Animate Wikipedia History	416
99. Create Greasemonkey Scripts Automatically	434
100. Remember Everything You Read	439
Index	461

Foreword

It has been occasionally noted that Greasemonkey is a hacker's tool. I take some pride in that, since I come from a family of relentless hackers.

My father was a landscape contractor who moonlighted doing home renovation. Sometimes, he worked on other people's homes, but he mostly focused on our own. My childhood living arrangement was in a constant state of refactoring. At one point, it featured a giant saltwater aquarium mounted in the living room wall and a freshwater koi pond with live turtles in the entryway. My dad drilled an eight-foot-wide hole in our home's foundation, dug a hole in the dirt beneath, cemented it, and filled it with water, rocks, and fish, without any training whatsoever. He read a few books and figured out the rest on his own.

My mother and stepfather own a landscape-maintenance firm, but they are also tireless improvers of their surroundings. At this point, I think their home is at least 50% custom-built. They are architecting the next one themselves from scratch. Finding that none of the canned blueprints for mountain cottages adequately addressed their lounging-on-the-front-porch-on-Saturday-morning needs, they determined they had no choice but to draw their own.

This desire to improve one's surroundings isn't limited to my crazy family. And despite our love for big trucks and Home Depot, it isn't strictly an American thing, either. Hacking can be traced all the way back to the first Homo sapiens crafting stone cutting tools. To hack is to be human. Our species' entire history can be defined in terms of creating new tools to make our lives better.

Greasemonkey is a tool for making your life on the Web better. Think of it like a power drill for the Internet: fast, efficient, flexible, useful for a variety of tasks, easy, fun, and generally a good thing to have around the house.

You won't use it to build fine furniture, but when you just need some more shelves for the cupboard, it's the perfect thing.

But even with a great tool, you need a good teacher to show you how to use it and ensure you don't hurt yourself. Mark Pilgrim is a master writer and teacher. His *Dive into Python* is the bible for Python programmers and the book that I turn to most often when using that language. His previous Greasemonkey work, *Dive into Greasemonkey* (<http://www.diveintogreasemonkey.org>), is Greasemonkey's definitive online reference. Without his simple explanations and painstaking documentation, the Greasemonkey community couldn't have grown into the worldwide collection of programmers, IT professionals, and hobbyists that it is today.

If you read and understand the examples in this book, you'll become a member of this community. More important, you'll be well on your way to becoming a master of your web environment. The next time you find yourself frustrated by a broken web site, you won't have to live with it. You'll have the tools and knowledge to fix it yourself.

—Aaron Boodman
Creator and Lead Developer of Greasemonkey

Credits

About the Author

Mark Pilgrim is an accessibility architect by day. By night, he is a husband and father who lives in North Carolina with his wife, his two sons, and his dog. He spends his copious free time sunbathing, skydiving, and reading Immanuel Kant's *The Critique of Pure Reason* in the original Klingon. This is his first O'Reilly book.

Contributors

The following people contributed their hacks, writing, and inspiration to this book:

- Jesse Andrews is a developer who stumbled upon the Internet a few days ago. When conscious, he can be found spouting the virtues of Ruby on Rails, JavaScript, and wearing socks with sandals. You can rummage through his latest projects at <http://overstimulate.com> or have him rummage through yours at <http://voltaiccommerce.com>.
- Albert Bachand
- Michael Bolin is a software engineer at Google. Before coming to Google, he was a student at MIT, where his master's thesis, *End-User Programming for the Web*, won the William A. Martin Memorial Thesis Prize in computer science. The thesis introduces an extension to JavaScript called Chickenscratch that is designed to enable end-user programmers to script web pages. The language and development environment are bundled as a Firefox extension called Chickenfoot that has similarities to Greasemonkey. Michael holds master's and bachelor's degrees in computer science from MIT.

-
- Aaron Boodman is a 26-year-old web developer from Southern California. He wrote Greasemonkey in November 2004 in a fit of frustration with web pages that didn't work correctly. He expected his five closest friends to use it in the best case. When he isn't obsessing over a current project, he's probably moving. In the past five years, he has lived in Orange County, L.A., Atlanta, Queens, the East Village, Seattle, and San Francisco. The most reliable place to find him will always be at <http://youngpup.net>.
 - Julien Couvreur is a developer for Microsoft, in the MSN division. There he worked in Passport, mainly on the authentication and bot prevention, and then joined the MSN Business Intelligence team. Julien is greatly passionate about the possibilities of the Web and especially has lots of fun with Greasemonkey. Originally from France and after spending a year in a San Francisco in a now defunct startup (RedCart), he currently lives in the state of Washington with his lovely fiancée, Lina, and shares thoughts and software projects on his blog, "Curiosity is Bliss" (<http://blog.monstuff.com>).
 - Roberto De Almeida is a Brazilian oceanographer trying to understand how the South Atlantic Ocean interacts with climate. In his spare time, he likes to play with JavaScript, Python, and his dog.
 - Vasil Dinkov is a 22-year-old web developer living in Plovdiv, the second largest city in Bulgaria. He is finishing his BSc in Computer Sciences at the University of Plovdiv in 2005. He has been playing with web technologies since 1999. In 2003, he established his own company, Vadikom (<http://vadikom.com>) and transformed his hobby into profession. He adores spending his spare time with the love of his life, Vania, playing soccer with friends, and listening to Depeche Mode.
 - Chris Feldmann's first computer was an Apple. He used it to play Lode Runner, and didn't start learning to program until 20 years later. He's a freelance web developer living in Brooklyn who just doesn't have time to be reading Slashdot. Watch for him skulking the sidewalks in a Greasemonkey T-shirt, or contact him at <http://www.axlotl.net>. "¡Mas peligroso que mono con navaja!"
 - Jonathan Fenocchi (<http://www.SlightlyRemarkable.com>) is a freelance web developer based in Southern Texas. He works as a web and graphics designer and also does programming in PHP and JavaScript.
 - Matthew Gertner is the chief technology officer of AllPeers (<http://www.allpeers.com>). A Brit by birth, an American by upbringing, and a European by choice, he has spent the last 13 years living in Paris and Hamburg before finally settling in Prague. In his professional persona, he's

first and foremost a C++ software developer, although he has been wasting a lot of time lately writing frivolous Greasemonkey scripts. When not working, he is an avid cook and a dangerously obsessive player of backgammon and golf.

- Joe Gregorio is President of BitWorking, Inc. ([http:// bitworking.com](http://bitworking.com)). He has over 13 years of software design and project management experience, working on a range of applications from embedded and web-based systems to Windows desktop applications. He is an active member of the syndication community and the author of the Atom API. He spends his free time exploring the limits of XML and HTTP. He holds a Master of Arts degree in mathematics from Dartmouth College and Bachelor's degrees in mathematics and computer science from Eastern Connecticut State University. He maintains a personal weblog at [http:// bitworking.org](http://bitworking.org).
- Logan Ingalls is a web programmer, Linux administrator, and bad driver from Connecticut. When he isn't sitting in front of a screen, he's probably hiking the Appalachian Trail or trying not to burn dinner. His arm-chair complaints and newest half-done projects can be found at [http:// plutor.org](http://plutor.org).
- Prakash Kailasa is a firm believer in Free Software and a student of Dynamic Languages.
- Adam Langley
- Stuart Langridge has been playing with the Web since 1994, and is quite possibly the only person in the world to have a BSc in Computer Science and Philosophy. When not working on the Web, he's a keen Linux user and part of the team at the open source radio show LUGRadio (<http://www.lugradio.org>). He likes drinking decent beers, studying stone circles, and scripting the DOM, not necessarily in that order.
- Johannes la Poutre has a fascination for all things innovative in engineering and practical science. After getting a Master's degree in industrial engineering and having a short career in the professional advertising and media world, he switched to the front lines of Internet technology. Currently, he is the leader of web development team at a large Dutch ISP. In his spare time, he tries to keep a balance between his family, and mixing and mashing all things XML, JavaScript, and CSS just for the fun of it.
- Anthony Lieuallen is a 25-year-old web application development specialist living in New York. His personal home page is available at [http:// www.arantius.com](http://www.arantius.com), and his resume can be found at <http://portfolio.arantius.com>.

-
- Gervase Markham is a member of the Mozilla.org staff and part-time employee of the Mozilla Foundation. He is particularly interested in modern web development, usability, and security. He also enjoys football, writing his weblog “Hacking for Christ” (<http://weblogs.mozillazine.org/gerv>), and reorienting toilet rolls so the loose end hangs down at the back.
 - Evan Martin
 - Matt McCarthy
 - Jeff Minard, having worked in the tech industry since age 13, says it best: “Computers are just fun, ya know?”
 - Justin J. Novack is a systems administrator and an avid script hacker. He thanks everyone who contributes to the open source community. With them, many young and aspiring minds have learned tips, tricks, hacks, and entirely new programming languages. His recent scripting abominations can be found at <http://scripts.slightlyinsane.com>.
 - Leslie Michael Orchard is a hacker, tinkerer, and creative technologist who works in the Detroit area. He’s engaged to a very patient and understanding science genius girl, and they both live with two spotted Ocicats and two dwarf bunnies. On rare occasions when spare time comes in copious amounts, he plays around with odd bits of code and writing, sharing them on his web site named 0xDECAFBAD (<http://www.decafbad.com>).
 - Mihai Parparita is a software engineer at Google. In a former life, he was a Mac shareware developer while being a student at Princeton. He completely misunderstood the purpose of Greasemonkey and started to use it to add features to web sites, instead of doing simple hacks like he was supposed to. He writes about his Greasemonkey scripts and other hacks at <http://www.persistent.info>.
 - Dan Phiffer is a designer and web hacker from Los Angeles. For the past year, he has been working as a freelance web site builder and will soon start graduate school at NYU’s Interactive Telecommunications Program. Dan isn’t really sure where this web thing is headed, but he enjoys playing with technologies that challenge the read-only tradition of media.
 - Jon Ramsey (<http://idlevice.co.uk>) is a web developer living in Hertfordshire, United Kingdom. He is a founding member of PHP London (<http://www.phplondon.org>) and works on a few open source projects whenever he gets the chance. Greasemonkey has renewed Jon’s lapsed interest in JavaScript, which has the great benefit that a worthwhile project can be completed in a period that fits his attention span. When not messing

around with web stuff, Jon enjoys writing music, traveling, and playing snooker very badly.

- Jason Rhyley has been playing with the Web since 1995. Despite popular opinion, he feels that things are more exciting now than ever. A recovering child prodigy and lifelong autodidact, he lives at <http://www.rhyley.org> and resides in Charleston, SC. And yes, he'd love to work for you.
- Timothy Rice
- Jesse Ruderman
- Britt Selvitelle is an open source software developer whose enduring loves include Ruby, JavaScript, Greasemonkey, all things Mozilla, and the entire Spanish language and cultures. He is currently most excited about the newly formed <http://tr.emendo.us> project. His rock-and-roll lifestyle can be observed in real time at <http://lukewarmtapioca.com>.
- Gareth Simpson
- Ben Tesch
- Scott Turner has a Ph.D. in computer storytelling, won the ACM International Programming Contest in 1989, contributed Keystone Kops to Nethack, appeared at least 10 times in *rec.humor.funny*, and is the world's foremost expert on Rail Baron (retired). He is the author of Platypus and, strangely enough, actually is a rocket scientist.
- Simon Willison
- Phil Wilson is a 25-year-old software developer from Bristol in the United Kingdom. His web site doesn't have any cat photos, which makes him sad. You can check up on his feline-free lifestyle by visiting <http://www.philwilson.org>. This will make him happy again.
- Sencer Yurdagül studies information systems, as well as life, the Web, and everything. He enjoys building and experimenting with server-side web stuff, yet he had not written a single line of JavaScript until he discovered Greasemonkey.
- Carlo Zottmann (<http://G-Spotting.net>) is an application developer and geek at heart. He spends his spare time tinkering with web sites, Greasemonkey scripts, Firefox extensions, JavaScript, Python, Ruby on Rails, and (if it can't be helped) PHP and Perl. By the time you read this, he will once again be trying to get a grip on his love for World of Warcraft. He lives in Munich, Germany, with his wife, Dana.

Acknowledgments

First and foremost, I would like to thank Aaron Boodman for creating Greasemonkey, for taking me seriously when I reported security holes in Greasemonkey 0.3, and for working many long nights and weekends to make Greasemonkey 0.5 both secure and backward compatible. Without him, this book would not exist.

Second, I give my undying love and appreciation to my wife, Dora, who went to bed alone far too many times while I stayed up and wrote this book.

Third, I thank the members of the Greasemonkey mailing list (<http://greasemonkey.mozdev.org/list.html>) for maintaining such a high signal-to-noise ratio. They make my inbox a happy place.

Finally, I will forever appreciate my editor, Brian Sawyer, for his obsessive hatred of the passive voice; my technical editor, Simon Willison, for showing me that JavaScript can be elegant as well as functional; and everyone at Perkins on Highway 64 for providing late-night comfort food, electricity, and friendly conversation.



Preface

Greasemonkey is a Firefox extension that allows you to write scripts that alter the web pages you visit. You can use it to make a web site more readable or more usable. You can fix bugs that site owners can't be bothered to fix themselves. You can alter pages so they work better with assistive technologies that speak a web page out loud or convert it to Braille. You can even automatically retrieve data from other sites to make two sites more interconnected.

Greasemonkey by itself does none of these things. In fact, after you install it, you won't notice any change at all...until you start installing what are called *user scripts*. A user script is just a chunk of JavaScript, the same scripting language you use on your own web site. But user scripts don't run on your own web site; they can run on any web site, and they can do anything JavaScript can do. In fact, they can do more than that, because Greasemonkey provides special API functions that give user scripts even more power than traditional JavaScript.

Why Greasemonkey Hacks?

The term *hacking* has a bad reputation in the press. They use it to refer to someone who breaks into systems or wreaks havoc with computers as their weapon. Among people who write code, though, the term *hack* refers to a "quick-and-dirty" solution to a problem, or a clever way to get something done. And the term *hacker* is taken very much as a compliment, referring to someone as being *creative*, having the technical chops to get things done. The Hacks series is an attempt to reclaim the word, document the good ways people are hacking, and pass the hacker ethic of creative participation on to the uninitiated. Seeing how others approach systems and problems is often the quickest way to learn about a new technology.

Greasemonkey has existed for less than a year, and hundreds of people have already written thousands of Greasemonkey scripts to scratch their own personal itches. Web enthusiasts with zero JavaScript experience have written scripts to route around broken web sites, alter site styles, and *roll back* ill-conceived site redesigns. More experienced coders have created link trackers, password managers, and personal shopping agents. Some have even added entirely new features to complex web applications—without ever needing to talk to the application developers or wait for bureaucratic approval.

Clearly, there were a lot of itches waiting to be scratched.

Some hacks in this book are short and sweet; they do one thing and do it well. (One is just a single line of code!) Other hacks are hundreds of lines long, complete with their own user interface, their own data cache, and their own preferences. This book showcases the best of the best, from “Hey, that’s always bugged me,” to “Gee, I don’t know how I ever lived without this,” to “Wow, I had no idea a browser could do that.”

How This Book Is Organized

You can read this book from cover to cover if you like, but each hack stands on its own, so feel free to browse and jump to the different sections that interest you most. If there’s a prerequisite you need to know about, a cross-reference will guide you to the right hack.

The book is divided into 12 chapters, roughly organized by topic:

Chapter 1, *Getting Started*

It’s hard to do cool stuff when you don’t even know where to click. The hacks in this chapter will get you started using the Greasemonkey interface, installing user scripts, and developing your own.

Chapter 2, *Linkmania!*

The Web revolves around links. Make them work for you! Learn how to control links that try to open a new window, launch unwanted applications, execute JavaScript, or otherwise behave badly. Plus, see how to follow links without clicking.

Chapter 3, *Beautifying the Web*

The Web is a grim place to visit. Spruce it up! These hacks will show you how to fine-tune tool tips, banish the scourge of Arial, and get rid of those God-awful smileys.

Chapter 4, *Web Forms*

You can't spit in this town without hitting a web form. `<input>` boxes in particular make a satisfying "ping" when you hit them. Discover what all those forms are doing behind your back. Plus, never forget a web site password again.

Chapter 5, *Developer Tools*

You don't just live online; you occasionally work there, too. Make the browser a better tool for web development with these hacks. Does the term *AJAX* ring a bell? You're going to love this chapter.

Chapter 6, *Search*

Hey there! Yes, you. Stop searching for porn long enough to read this chapter. It's all about how to make searching the Web easier and faster. What you do with that information is between you and your webcam.

Chapter 7, *Web Mail*

Web-based email sucks. But you can't live without it. Have you ever accidentally hit "Reply All" instead of "Reply"? (Did your coworkers ever forgive you?) Who the hell put them one inch away from each other? Find out how to add essential features to web mail services such as Gmail and Yahoo! Mail.

Chapter 8, *Accessibility*

Accessibility affects everyone, in every walk of life. One in 13 workers reported some form of disability last year. Learn how to make your own web site more accessible, and see how Greasemonkey can help people with disabilities use the Web.

Chapter 9, *Taking Back the Browser*

Invasive site registrations. Brain-dead browser sniffers. Frames. Something about the Web makes content providers stupid. Learn how to route around them.

Chapter 10, *Syndication*

Blogs are all the rage. They've reached the tipping point. They've jumped the shark. They've taken the meme by the horns and the cliché by the throat. Dive into the wonderful world of syndicated feeds.

Chapter 11, *Site Integration*

Find a book on Amazon.com and get price quotes from five other retailers *without leaving the page*. Enough said.

Chapter 12, *Those Not Included in This Classification*

Good things come to those who wait. No, wait; this is a Hacks book! Jump right to the end; I've saved the best for last.

Conventions Used in This Book

The following is a list of the typographical conventions used in this book:

Italics

Used to indicate URLs, filenames, filename extensions, and directory/folder names. For example, a path in the filesystem appears as */Developer/Applications*.

Constant width

Used to show code examples, the contents of files, console output, as well as the names of variables, commands, and other code excerpts.

Constant width bold

Used to highlight portions of code, typically new additions to old code.

Constant width italic

Used in code examples and tables to show sample text to be replaced with your own values.

Gray type

Used to indicate a cross-reference within the text.

You should pay special attention to notes set apart from the text with the following icons:



This is a tip, suggestion, or general note. It contains useful supplementary information about the topic at hand.



This is a warning or note of caution, often indicating that your money or your privacy might be at risk.

The thermometer icons, found next to each hack, indicate the relative complexity of the hack:



beginner



moderate



expert

Using Code Examples

This book is here to help you get your job done. Thus, you may use the code in this book in your programs and documentation without contacting us for permission as long as you are reproducing limited portions for use in

- [Love in a World of Sorrow: A Teenage Girl's Holocaust Memoirs book](#)
- [read Think and Grow Rich.pdf](#)
- [read online Landscaping With Fruit: Strawberry ground covers, blueberry hedges, grape arbors, and 39 other luscious fruits to make your yard an edible paradise](#)
- [Peacock \(Animal\) book](#)

- <http://pittiger.com/lib/Blogging-All-in-One-For-Dummies--2nd-Edition-.pdf>
- <http://fortune-touko.com/library/Myths-and-Symbols-in-Indian-Art-and-Civilization.pdf>
- <http://monkeybubblemedia.com/lib/Night-Stalker.pdf>
- <http://xn--d1aboelcb1f.xn--p1ai/lib/The-Superpromoter--The-Power-of-Enthusiasm.pdf>