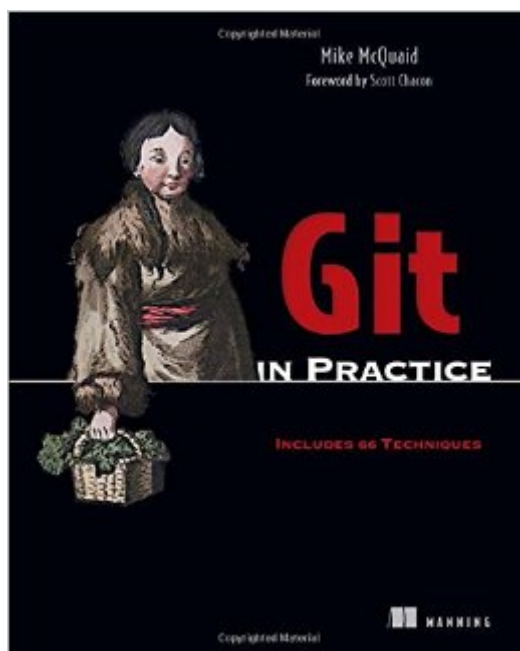


The book was found

# Git In Practice



## Synopsis

SummaryGit in Practice is a collection of 66 tested techniques that will optimize the way you and your team manage your development projects. The book begins with a brief reminder of the core version control concepts you need when using Git and moves on to the high-value features you may not have explored yet. Then, you'll dig into cookbook-style techniques like history visualization, advanced branching and rewriting history each presented in a problem-solution-discussion format. Finally you'll work out how to use Git to its full potential through configuration, team workflows, submodules and using GitHub pull requests effectively. Purchase of the print book includes a free eBook in PDF, Kindle, and ePub formats from Manning Publications.

About the TechnologyGit is a source control system, but it's a lot more than just that. For teams working in today's agile, continuous delivery environments, Git is a strategic advantage. Built with a decentralized structure that's perfect for a distributed team, Git manages branching, committing, complex merges, and task switching with minimal ceremony so you can concentrate on your code.

About the BookGit in Practice is a collection of battle-tested techniques designed to optimize the way you and your team manage development projects. After a brief overview of Git's core features, this practical guide moves quickly to high-value topics like history visualization, advanced branching and rewriting, optimized configuration, team workflows, submodules, and how to use GitHub pull requests. Written in an easy-to-follow Problem/Solution/Discussion format with numerous diagrams and examples, it skips the theory and gets right to the nitty-gritty tasks that will transform the way you work. Written for developers familiar with version control and ready for the good stuff in Git.

What's InsideTeam interaction strategies and techniquesReplacing bad habits with good practicesJuggling complex configurationsRewriting history and disaster recovery

About the AuthorMike McQuaid is a software engineer at GitHub. He's contributed to Qt and the Linux kernel, and he maintains the Git-based Homebrew project.

Table of ContentsPART 1 INTRODUCTION TO GITLocal GitRemote GitPART 2 GIT ESSENTIALSFilesystem interactionsHistory visualizationAdvanced branchingRewriting history and disaster recoveryPART 3 ADVANCED GITPersonalizing GitVendoring dependencies as submodulesWorking with SubversionGitHub pull requestsHosting a repositoryPART 4 GIT BEST PRACTICESCreating a clean historyMerging vs. rebasingRecommended team workflows

## Book Information

Paperback: 272 pages

Publisher: Manning Publications; 1 edition (October 12, 2014)

Language: English

ISBN-10: 1617291978

ISBN-13: 978-1617291975

Product Dimensions: 0.8 x 7.5 x 9.2 inches

Shipping Weight: 1 pounds (View shipping rates and policies)

Average Customer Review: 4.2 out of 5 stars [See all reviews](#) (12 customer reviews)

Best Sellers Rank: #359,513 in Books (See Top 100 in Books) #49 in [Books > Computers & Technology > Web Development & Design > Content Management](#) #86 in [Books > Computers & Technology > Programming > Software Design, Testing & Engineering > Tools](#) #141 in [Books > Computers & Technology > Programming > Software Design, Testing & Engineering > Testing](#)

## Customer Reviews

This book covers all aspects of using Git with the Github repository. Although the book addresses Linux and Windows, particular attention is paid to OSX users. The book can be read cover to cover, but it is aimed primarily at Git users who periodically run into a new situation or are interested in extending their expertise in new directions. As such, the book is presented as a collection of 66 "techniques", ranging from the simple to the complex. Discussion of each technique is accompanied by a concrete example of using a Github environment that the author uses throughout the book. The author has a lot of experience with Git, and he shares nuggets of wisdom throughout the book, including a section near the end of the book that is dedicated to Git best practices. At the very end of the book are appendixes containing instructions on Git installation and GitHub account setup, heavily annotated copies of the author's Git configuration files and a short treatise on the value of version control systems. I recommended the book particularly for practitioners with at least a basic understanding of Git. I read the book through from front to back and ran into only one area of minor frustration. The book uses https as the protocol of choice for interacting with the GitHub repository. For that reason, there was an almost constant prompting for my user ID and password. Eventually, I took to the web in search of a way to minimize this. I discovered a credential helper that could be activated to cache credentials for a user-designated period of time. Later on, I found excellent documentation on the GitHub web site that described how to set up an SSH agent, and I switched to that instead. This problem would not be an issue for someone using the book primarily as a reference as the prompting would only occur for the one or two techniques used at that point. Overall the book is excellent, and would be a worthwhile addition to a programmer's library.

Git In Practice is a problem/solution style book that presents solutions as numbered techniques. It is

aimed at intermediate and advanced users of GIT but concedes that beginners may cope if they are willing to do a bit of extra research about each topic discussed. As well as the techniques, there is also discussion of the reasons for using version control systems and of the pros and cons of a number of possible GIT workflows for teams of programmers developing a product. I fall into the beginner camp; I've used GIT to commit versions of my work and push them to Github mostly as backup. The book has given me a good overview of the tools that GIT provides and a growing sense of when I might use them. However, right from the start, the discussions in the book reference further techniques, flags and options that were a bit overwhelming for me as an inexperienced user. As I worked through the book, many of those concepts started falling into place but many others will only really make sense when I have more experience. If you're a beginner who would rather be supported through your initial learning about GIT, then this is probably more of a three star book for you. I am willing to give the book the benefit of the doubt - it is mostly well written with clear examples and a sensible structure - and take the blame myself for any confusion. The book starts with the basics of adding, committing, pushing and cloning and moves steadily through to branching, merging, rebasing, logging and re-writing history. It also looks at some graphical clients like gitx. It spends a lot of time examining the output from GIT commands. I managed to follow along with most examples but wasn't always 100% sure of what was going on. The diagrams included with the text didn't always make the explanations any clearer for me, although those accompanying the sections on branching/merging workflows were helpful. The discussion of various approaches for teams was interesting for me as I work in a team of one. They've given me some ideas to play with as I start to make better use of GIT. Overall, the book is well structured with clear, accurate writing for the main text. However, the clarity is sometimes undermined by throwaway references to more advanced concepts. I'll definitely return to the book once I've gained more experience using the basics in the wild.

This book is written in the cookbook format where there are a series of 66 techniques which are different types of situations you may want to use with Git. This is a good book for intermediate readers and to have as a reference for the various cookbook recipes (techniques), however I think it would be a little rough intro to git for complete Git novices. If you are a novice user, I have found the Atlassian Git tutorials, <https://www.atlassian.com/git/tutorials/>, to be a valuable resource. I came to Git from Perforce and Subversion, where a centralized SCM was very familiar to me. Git is a decentralized SCM, although you can use it with a centralized workflow. Chapter 1 does decent job of getting a new user up and going with by a local filesystem directory under git. The one thing here

that would likely only affect new Git users would be the use of the Git Index/Staging Area. Traditional SCM systems don't have this buffer area between the working copy (the files you modify) and the Git datastore. Normally files go from the working copy to the SCM's datastore, but with Git they pass through the Staging Area 1st on their way to the datastore. The real meat of the book is in Chapter 2 where we see remote use of Git and Part 2 of the book which demonstrates various techniques that most Git users will use on a daily if not often basis. One thing I think, would have been better is if the chapter on personalizing git had of been moved closer to the front of the book than the end. Seeing as the book focuses on teaching you the Git commands, it is a lot easier to type "git co -b " over "git checkout -b " by creating git aliases for some of the most common commands. If you're a command line user then you WILL type a set of git commands all the time and every key stroke saved is one less chance for carpal tunnel syndrome.

[Download to continue reading...](#)

Git in Practice Better Git It in Your Soul: An Interpretive Biography of Charles Mingus Git Essentials Pragmatic Guide to Git (Pragmatic Guides) Version Control with Git: Powerful tools and techniques for collaborative software development Starting a Medical Practice: The Physician's Handbook for Successful Practice Start-Up (Practice Success! Series) Ready, Set, Practice: Elements of Landscape Architecture Professional Practice Practice Makes Perfect Algebra (Practice Makes Perfect (McGraw-Hill)) MCAT Chemistry and Physics: Strategy and Practice: Timed Practice for the Revised MCAT MCAT Psychology and Sociology: Strategy and Practice (MCAT Strategy and Practice) CPC Practice Exam 2016: Includes 150 practice questions, answers with full rationale, exam study guide and the official proctor-to-examinee instructions McGraw-Hill Education 5 TEAS Practice Tests, 2nd Edition (Mcgraw Hill's 5 Teas Practice Tests) Adult CCRN Strategies, Practice, and Review with 2 Practice Tests (Kaplan Test Prep) Common Core Practice - 8th Grade English Language Arts: Workbooks to Prepare for the PARCC or Smarter Balanced Test: CCSS Aligned (CCSS Standards Practice) (Volume 11) SSAT & ISEE 2017 Strategies, Practice & Review with 6 Practice Tests: For Private and Independent School Admissions (Kaplan Test Prep) NCLEX-RN 2016 Strategies, Practice and Review with Practice Test (Kaplan Test Prep) ACT 2016-2017 Strategies, Practice, and Review with 6 Practice Tests: Online + Book (Kaplan Test Prep) SAT 2017 Strategies, Practice & Review with 3 Practice Tests: Online + Book (Kaplan Test Prep) New GED® Test Strategies, Practice, and Review with 2 Practice Tests: Book + Online Fully Updated for the 2014 GED (Kaplan Test Prep) GED Test 2017 Strategies, Practice & Review with 2 Practice Tests: Online + Book (Kaplan Test Prep)

[Dmca](#)