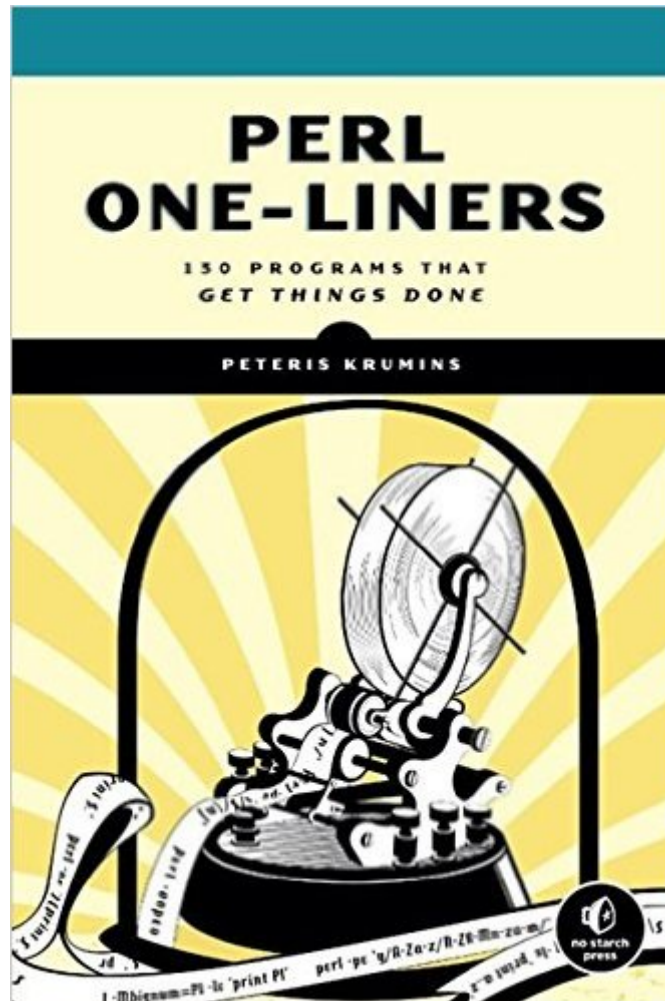The book was found

Perl One-Liners: 130 Programs That Get Things Done

Synopsis

Part of the fun of programming in Perl lies in tackling tedious tasks with short, efficient, and reusable code. Often, the perfect tool is the one-liner, a small but powerful program that fits in one line of code and does one thing really well.In Perl One-Liners, author and impatient hacker Peteris Krumins takes you through more than 100 compelling one-liners that do all sorts of handy things, such as manipulate line spacing, tally column values in a table, and get a list of users on a system. This cookbook of useful, customizable, and fun scripts will even help hone your Perl coding skills, as Krumins dissects the code to give you a deeper understanding of the language.You'll find one-liners that:Encode, decode, and convert stringsGenerate random passwordsCalculate sums, factorials, and the mathematical constants Ï and eAdd or remove spacesNumber lines in a filePrint lines that match a specific patternCheck to see if a number is prime with a regular expressionConvert IP address to decimal formReplace one string with anotherAnd many more! Save time and sharpen you coding skills as you learn to conquer those pesky tasks in a few precisely placed keystrokes with Perl One-Liners.

Book Information

Paperback: 168 pages

Publisher: No Starch Press; 1 edition (December 30, 2013)

Language: English

ISBN-10: 159327520X

ISBN-13: 978-1593275204

Product Dimensions: 7 x 0.5 x 9.2 inches

Shipping Weight: 13.4 ounces (View shipping rates and policies)

Average Customer Review: 4.4 out of 5 starsÂ Â See all reviewsÂ (15 customer reviews)

Best Sellers Rank: #139,419 in Books (See Top 100 in Books)   #8 inÂ Books > Computers & Technology > Programming > Languages & Tools > Perl   #161 inÂ Books > Computers & Technology > Networking & Cloud Computing > Network Administration   #235 inÂ Books > Computers & Technology > Programming > Software Design, Testing & Engineering > Object-Oriented Design

Customer Reviews

Perl one-liners are an important part of its power and flexibility. The ability to process a file quickly without having to write a program is often really useful. Any Perl programmers should take the time to get to know the command like switches that make this possible. This book is a pretty good

introduction to this way of using Perl.So why only three stars? Well, I have a couple of reservations about the book. Firstly, there are a few technical errors which the editors should have caught. For example, a few times the author refers to "array context" where he means "list context". The difference between arrays and lists is often difficult for beginners to master and it doesn't help when books blur the distinction.My other reservation is with the programs themselves. The book boasts "130 programs that get things done". But I think they have had to stretch things a bit to get to that number. One program might be "print lines that match a pattern". Then the next program will be "print lines that don't match a pattern". I'm not sure that inverting the logic in a one-liner is enough of a difference to justify counting it separately. Sometimes you'll come across two or three pages of examples all of which are only tiny permutations of each other.But it's good to see publishers bringing out books on Perl. And this is certainly an area of Perl that hasn't received much coverage before. I just think it's a rather thin concept to spin out to a book. Even this stretched, it's a rather thin book (140 pages - 50 of which are appendices). It might have been better as a cheap Kindle-only publication.

Full Disclosure: I received a free review copy of this bookI'm primarily a backend web programmer. I try to write in an object-oriented way with test-driven development practices, with each object having a well-defined purpose. I do this primarily in Perl, a language that causes many developers to get a twitch. The reason is that Perl has an old reputation as being messy line noise. Part of the reason for that reputation is because of the one-liner obfuscation games that used to be quite popular.So when I first heard of this book, I was naturally inclined to avoid it. In a way, it's perpetuating an aspect of Perl that's best left as part of history. Besides, as a web developer, what did this book have to offer me?What I found was that one-liners offer a different perspective, and it's a useful perspective to have. Converting a text document to double-spaced lines isn't a problem I have every day, but maybe I will at some point. When that or one of the many other problems this book addresses comes up, I'll have something within easy reach.Perhaps more importantly than any specific problem is the attitude that programming in this fashion doesn't have to be inscrutable. Every one-liner here comes with a detailed explanation, and many come with alternative solutions with their own explanations. If we can document our concise solution in less space than a more "well-developed" solution would take, what's the problem?Donald Knuth had attempted a system called "Literate Programming", where documentation and code are carefully interleaved. Jon Bentley, writer of the Programming Pearls column in Communications of the ACM, challenged Knuth to write a Literate Program for the following problem: read a file of text, determine the n most

frequently used words, and print out a sorted list of those words along with their frequencies.Knuth wrote a 10-page program. Bentley then turned around and wrote a 6-command shell pipeline to do the same thing, which also happened to miss a few bugs that Knuth had stumbled over. The entire shell script and its documentation fit on a post-it note.*One-liners can look like inscrutable line noise. But if we can otherwise document them in a succinct way, does it really matter?That's what this book does: provides solid explanations for extremely concise code. No matter what you do in Perl or your skill level, there's something in here for everyone.* See: [...]

Perl one-liners is the quickest way to get a job done and this book offers 130 very interesting tasks that you can accomplish by using one-liners.Many of them definitely represent the most charming, elegant and brilliant way to do something and I found practicing these tasks really exciting.Quoting the author, Ã¢Â Âœ[Ã¢Â Â] PerlÃ¢Â Â™s motto is ThereÃ¢Â Â™s More Than One Way To Do It, which is abbreviated as TIMTOWTDI and pronounced Tim Toady[Ã¢Â Â]Ã¢Â Â•.Reading this book you will learn Ã¢Â Âœ[Ã¢Â Â] Perl golfing, a sport that involves writing the shortest Perl program to get a task doneÃ¢Â Â•.Just like a golfer trying to put the ball in the hole with as few strokes as possible, perhaps with a single blow.Just to give an idea, with a one-liner you can for example-Generate random passwords- Get a list of all users on the system- Find out how many packets have passed through iptables rules- Remove all blank lines from a file- Generate a certain amount of data for analysis purposes- Convert DOS/Windows newlines to UNIX newlines- Selectively Printing and Deleting Lines- And so onÃ¢Â Â|As you can easily guess, it is advisable to have one eye on the book and the other on the monitor to put immediately into practice what youÃ¢Â Â™re learning BUT, if the reading takes you and if you're feeling particularly inspired, always remember that it is good practice to use the -i argument before performing operations on original files Ã Â•Â (see one-liner 6.1)Ã¢Â Â|Despite of its 168 pages, it may be quicker to read a tome of 1000 pages rather than Ã¢Â ÂœPerl One-LinersÃ¢Â Â•, because every single consideration in this book requires great attention and reasoning.I think that by performing these one-liners you can definitely become more familiar with Perl but this reading must be accompanied or preceded by learning the fundamentals.So I would not suggest to approach this scripting language by reading this book but I would definitely recommend Ã¢Â ÂœPerl One-LinersÃ¢Â Â• to everyone who wants to improve his s

Download to continue reading...

Perl One-Liners: 130 Programs That Get Things Done Higher-Order Perl: Transforming Programs with Programs Learning Perl: Making Easy Things Easy and Hard Things Possible Perl: Crash

Course - The Ultimate Beginner's Course to Learning Perl Programming in Under 12 Hours Effective Perl Programming: Ways to Write Better, More Idiomatic Perl (2nd Edition) (Effective Software Development Series) Mastering Perl/Tk: Graphical User Interfaces in Perl Perl Programming Success in a Day: Beginners Guide to Fast, Easy, and Efficient Learning of Perl Programming Programming the Perl DBI: Database programming with Perl What's Best Next: How the Gospel Transforms the Way You Get Things Done Tame Your Gmail in 5 Easy Steps with David Allen's GTD: 5-Steps to Organize Your Mail, Improve Productivity and Get Things Done Using Gmail, Google Drive, Google Tasks and Google Calendar How To Focus: Stop Procrastinating, Improve Your Concentration & Get Things Done - Easily! Get Big Things Done: The Power of Connectional Intelligence Daily Routine Makeover: Guide To Focused Action, Productivity Hacks, Stress-Free Performance - Get Things Done In Less Time Influencing Virtual Teams: 17 Tactics That Get Things Done with Your Remote Employees Time to Get Things Done: Beat Procrastination, Stop Being Lazy, Take Actions, and Master Your Life in 24 Hours (2nd Edition) (Organize Yourself, Organize ... Self Organization, To Do List Book 6) How To Beat Your Procrastination Forever: Top 10 Simple Secrets Successful People Use To Get Things Done (Success Mindsets) 101 Things to Do Outside: Loads of fantastically fun reasons to get up, get out, and get active! Getting Organized in the Google Era: How to Get Stuff out of Your Head, Find It When You Need It, and Get It Done Right BREAKING IN: Over 130 Advertising Insiders Reveal How to Build a Portfolio That Will Get You Hired Ghost Liners: Exploring the World's Greatest Lost Ships

[Dmca](#)