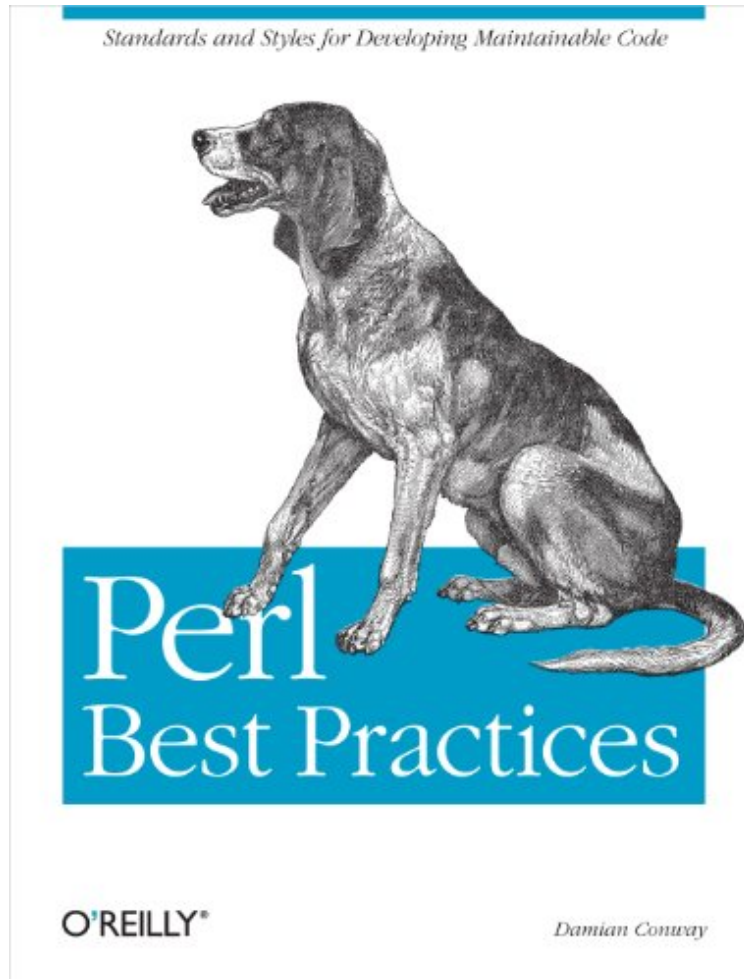


Perl Best Practices: Standards and Styles for Developing Maintainable Code

Von Damian Conway

**Download PDF | ePub | DOC | audiobook | ebooks*



 Download

 Read Online

Produktinformation -Verkaufsrank: #380454 in eBooksVerffentlicht am: 2005-07-12Erscheinungsdatum: 2009-06-18File Name: B002L4EXI8 | File size: 16.Mb

Von Damian Conway : Perl Best Practices: Standards and Styles for Developing Maintainable Code before purchasing it in order to gage whether or not it would be worth my time, and all praised Perl Best Practices: Standards and Styles for Developing Maintainable Code:

KundenrezensionenHilfreichste Kundenrezensionen16 von 16 Kunden fanden die folgende Rezension hilfreich. Conways zweites MeisterwerkVon Ein KundeDamien Conway ist neben Larry Wall und Randal L. Schwartz der wohl bekannteste Perl Hacker. Sein erstes Buch, Object Oriented Perl, ist bereits ein Klassiker und fr viele das beste Perl Buch fr Fortgeschrittene überhaupt. Sein neustes Buch ist eine Sammlung von Perl Coding Guidelines, die zu sauberen, lesbaren und wartbaren Code fhren sollen. In gewohnt humorvollem Schreibstil gibt Conways hier einfache aber sehr effektive Tipps. Die ersten beiden Kapitel widmet er dem Code Layout und den Naming Conventions. Schn:

Konfigurationsdateien für vim, emacs und perltidy liefert er gleich mit. Dann (die meisten Coding Guidelines im Netz enden nach diesen beiden Themen schon) wirds richtig interessant. Alle, die selbst Perl Module erstellen wollen, finden auf gut 150 Seiten wertvolle Tipps: wie dokumentiert man richtig? Wie am besten OO programmieren? Exception handling? Testing? Kaum jemand hat soviel Erfahrung mit OO Perl wie Conway, und das merkt man sofort. Aber nicht nur für Autoren von Perl Modulen ist PBP ein must-have. Er stellt zwischendurch viele tolle Module vor, von denen ich trotz 5 jähriger Perl Erfahrung nie was gehört hatte, z.B. Readonly, Perl6::Slurp, List::Util, English... Einziger Kritikpunkt, und deswegen nur 4 von 5 Sternen: Einige Module wie Config::Std, Class::Std, Getopt::Clade werden in den Mittelpunkt der entsprechenden Kapitel gestellt. Zum jetzigen Zeitpunkt sind diese Module entweder noch nicht erhltlich (Getopt::Clade) oder inkonsistent (Class::Std), auf jeden Fall aber eher Beta Status mit zum Teil erheblichen Syntaxänderungen seit dem Druck. Conway hätte hier besser die Module etwas früher veröffentlicht sollen. Fazit: Jeder Perl Programmierer findet hier wertvolle Tipps, die in der Tat zu besserem Code führen. Conway zeigt, dass man nicht unbedingt in Ruby oder Python skripten muss, um lesbaren Code zu erzeugen. 3 von 3 Kunden fanden die folgende Rezension hilfreich. Wann kommt die zweite Auflage? Von M. Kutter Perl Best Practices ist das bislang fehlende Teil in der Perl-Bibliothek: Es beschreibt Coding-Guidelines für die Perl-Entwicklung, wie sie in anderen Sprachen (Java, C#, Python) ganz selbstverständlich als Teil der Sprach-Dokumentation mitgeliefert werden. Dabei ist egal, ob man andere persönliche Vorlieben hat oder nicht - Perl Best Practices kann erfolgreich endlose Diskussionen über Code-Formatierung etc. verhindern. Schließlich ist letztendlich fast jede (definierte) Coding-Guideline so gut wie die andere - und die Richtlinien in Perl Best Practices muss man nicht mehr selbst zusammenstellen. Leider bevorzugt Conway in seinen Richtlinien meist seine eigenen Perl-Module auch dann wenn sie kaum mehr gepflegt sind und bessere Alternativen vorhanden sind, so da eine (bearbeitete) 2. Auflage angebracht wäre. Für Perl-Anfänger ist es dabei sehr schwer, herauszufinden, welchen Kapiteln man besser nicht vertrauen sollte. Beim Einsatz im Team sollte daher mindestens ein Entwickler mit Kontakt zur Perl-Community unerfahrenere Kollegen an die Hand nehmen können. Die Englische Originalausgabe ist für alle dem Englischen mächtigen der deutschen Übersetzung vorzuziehen - Werkzeuge zur statischen Code-Analyse wie Perl::Critic verweisen auf die Seitenzahlen der Englischen Ausgabe. 0 von 0 Kunden fanden die folgende Rezension hilfreich. Dieses Buch sollte jeder Programmierer lesen und beherzigen! Von Reinhold Nagel Da schreibt mal jemand zusammen, was zu tun ist, damit ein Programm möglichst verständlich und gut zu pflegen ist. Das Buch ist Gold wert, für jeden, der mehr als einen Dreizeiler programmiert. Es kann professionell genutzt werden; angeblich gibt es Firmen, bei denen dieses Buch einfach der Programmierstandard ist. Und das ist auch gut so!

Kurzbeschreibung Many programmers code by instinct, relying on convenient habits or a "style" they picked up early on. They aren't conscious of all the choices they make, like how they format their source, the names they use for variables, or the kinds of loops they use. They're focused entirely on problems they're solving, solutions they're creating, and algorithms they're implementing. So they write code in the way that seems natural, that happens intuitively, and that feels good. But if you're serious about your profession, intuition isn't enough. Perl Best Practices author Damian Conway explains that rules, conventions, standards, and practices not only help programmers communicate and coordinate with one another, they also provide a reliable framework for thinking about problems, and a common language for expressing solutions. This is especially critical in Perl, because the language is designed to offer many ways to accomplish the same task, and consequently it supports many incompatible dialects. With a good dose of Aussie humor, Dr. Conway (familiar to many in the Perl community) offers 256 guidelines on the art of coding to help you write better Perl code--in fact, the best Perl code you possibly can. The guidelines cover code layout, naming conventions, choice of data and control structures, program decomposition, interface design and implementation, modularity, object orientation, error handling, testing, and debugging. They're designed to work together to produce code that is clear, robust, efficient, maintainable, and concise, but Dr. Conway doesn't pretend that this is the one true universal and unequivocal set of best practices. Instead, Perl Best Practices offers coherent and widely applicable suggestions based on real-world experience of how code is actually written, rather than on someone's ivory-tower theories on how software ought to be created. Most of all, Perl Best Practices offers guidelines that actually work, and that many developers around the world are already using. Much like Perl itself, these guidelines are about helping you to get your job done, without getting in the way. Praise for Perl Best Practices from Perl community members: "As a manager of a large Perl project, I'd ensure that every member of my team has a copy of Perl Best Practices on their desk, and use it as the basis for an in-house style guide."-- Randal Schwartz "There are no more excuses for writing bad Perl programs. All levels of Perl programmer will be more productive after reading this book."-- Peter Scott "Perl Best Practices will be the next big important book in the evolution of Perl. The ideas and practices Damian lays down will help bring Perl out from under the embarrassing heading of "scripting languages". Many of us have known Perl is a real programming language, worthy of all the tasks normally delegated to Java and C++. With Perl Best Practices, Damian shows specifically how and why, so everyone else can see, too."-- Andy

Lester "Damian's done what many thought impossible: show how to build large, maintainable Perl applications, while still letting Perl be the powerful, expressive language that programmers have loved for years." -- Bill Odom "Finally, a means to bring lasting order to the process and product of real Perl development teams." -- Andrew Sundstrom "Perl Best Practices provides a valuable education in how to write robust, maintainable Perl, and is a definitive citation source when coaching other programmers." -- Bennett Todd "I've been teaching Perl for years, and find the same question keeps being asked: Where can I find a reference for writing reusable, maintainable Perl code? Finally I have a decent answer." -- Paul Fenwick "At last a well researched, well thought-out, comprehensive guide to Perl style. Instead of each of us developing our own, we can learn good practices from one of Perl's most prolific and experienced authors. I recommend this book to anyone who prefers getting on with the job rather than going back and fixing..." -- Pressestimmen "If you are looking for a book to teach you how to program Perl, this is definitely not what you need. Also, if you are cranking out quick Perl scripts to solve one-time tasks, it might not be worth the effort to read this book. However, if you are fairly comfortable with the language and are looking for ways to improve your code, this book would be a wonderful addition to your bookshelf." - James Mohr, Linux Magazine, November 2005

Kurzbeschreibung Many programmers code by instinct, relying on convenient habits or a "style" they picked up early on. They aren't conscious of all the choices they make, like how they format their source, the names they use for variables, or the kinds of loops they use. They're focused entirely on problems they're solving, solutions they're creating, and algorithms they're implementing. So they write code in the way that seems natural, that happens intuitively, and that feels good. But if you're serious about your profession, intuition isn't enough. Perl Best Practices author Damian Conway explains that rules, conventions, standards, and practices not only help programmers communicate and coordinate with one another, they also provide a reliable framework for thinking about problems, and a common language for expressing solutions. This is especially critical in Perl, because the language is designed to offer many ways to accomplish the same task, and consequently it supports many incompatible dialects. With a good dose of Aussie humor, Dr. Conway (familiar to many in the Perl community) offers 256 guidelines on the art of coding to help you write better Perl code--in fact, the best Perl code you possibly can. The guidelines cover code layout, naming conventions, choice of data and control structures, program decomposition, interface design and implementation, modularity, object orientation, error handling, testing, and debugging. They're designed to work together to produce code that is clear, robust, efficient, maintainable, and concise, but Dr. Conway doesn't pretend that this is the one true universal and unequivocal set of best practices. Instead, Perl Best Practices offers coherent and widely applicable suggestions based on real-world experience of how code is actually written, rather than on someone's ivory-tower theories on how software ought to be created. Most of all, Perl Best Practices offers guidelines that actually work, and that many developers around the world are already using. Much like Perl itself, these guidelines are about helping you to get your job done, without getting in the way.

Praise for Perl Best Practices from Perl community members: "As a manager of a large Perl project, I'd ensure that every member of my team has a copy of Perl Best Practices on their desk, and use it as the basis for an in-house style guide." -- Randal Schwartz "There are no more excuses for writing bad Perl programs. All levels of Perl programmer will be more productive after reading this book." -- Peter Scott "Perl Best Practices will be the next big important book in the evolution of Perl. The ideas and practices Damian lays down will help bring Perl out from under the embarrassing heading of "scripting languages". Many of us have known Perl is a real programming language, worthy of all the tasks normally delegated to Java and C++. With Perl Best Practices, Damian shows specifically how and why, so everyone else can see, too." -- Andy Lester "Damian's done what many thought impossible: show how to build large, maintainable Perl applications, while still letting Perl be the powerful, expressive language that programmers have loved for years." -- Bill Odom "Finally, a means to bring lasting order to the process and product of real Perl development teams." -- Andrew Sundstrom "Perl Best Practices provides a valuable education in how to write robust, maintainable Perl, and is a definitive citation source when coaching other programmers." -- Bennett Todd "I've been teaching Perl for years, and find the same question keeps being asked: Where can I find a reference for writing reusable, maintainable Perl code? Finally I have a decent answer." -- Paul Fenwick "At last a well researched, well thought-out, comprehensive guide to Perl style. Instead of each of us developing our own, we can learn good practices from one of Perl's most prolific and experienced authors. I recommend this book to anyone who prefers getting on with the job rather than going back and fixing..."