The book was found

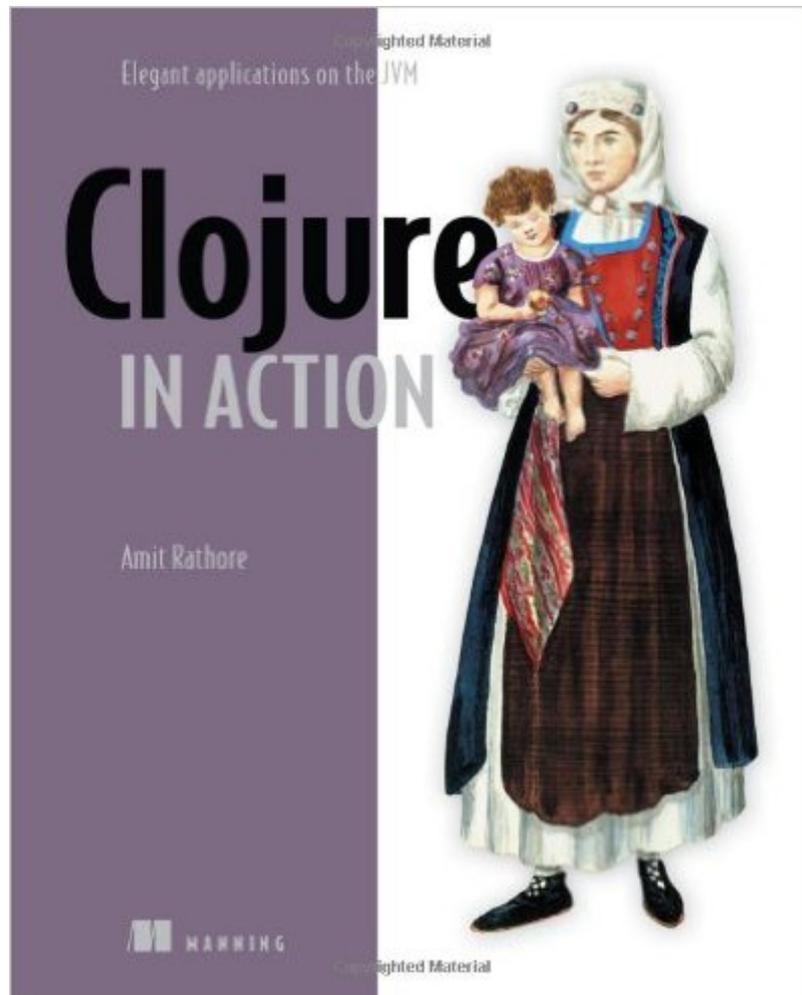# Clojure In Action

Elegant applications on the JVM

Clojure
IN ACTION

Amit Rathore

MANNING

Synopsis

Summary Clojure in Action is a hands-on tutorial for the working programmer who has written code in a language like Java or Ruby, but has no prior experience with Lisp. It teaches Clojure from the basics to advanced topics using practical, real-world application examples. Blow through the theory and dive into practical matters like unit-testing and environment set-up, all the way through building a scalable web-application using domain-specific languages, Hadoop, HBase, and RabbitMQ. About the Technology Clojure is a modern Lisp for the JVM, and it has the strengths you'd expect: first-class functions, macros, support for functional programming, and a Lisp-like, clean programming style. About this Book Clojure in Action is a practical guide focused on applying Clojure to practical programming challenges. You'll start with a language tutorial written for readers who already know OOP. Then, you'll dive into the use cases where Clojure really shines: state management, safe concurrency and multicore programming, first-class code generation, and Java interop. In each chapter, you'll first explore the unique characteristics of a problem area and then discover how to tackle them using Clojure. Along the way, you'll explore practical matters like architecture, unit testing, and set-up as you build a scalable web application that includes custom DSLs, Hadoop, HBase, and RabbitMQ. Purchase of the print book comes with an offer of a free PDF, ePub, and Kindle eBook from Manning. Also available is all code from the book. What's InsideA fast-paced Clojure tutorial Creating web services with Clojure Scaling through messaging Creating DSLs with Clojure's macro system Test-driven development with Clojure Distributed programming with Clojure and moreThis book assumes you're familiar with an OO language like Java, C#, or C++, but requires no background in Lisp or Clojure itself.

================================ Table of ContentsPART 1 GETTING STARTED Introduction to Clojure A whirlwind tour Building blocks of Clojure Polymorphism with multimethods Clojure and Java interop State and the concurrent world Evolving Clojure through macros PART 2 GETTING REAL Test-driven development and more Data storage with Clojure Clojure and the web Scaling through messaging Data processing with Clojure More on functional programming Protocols, records, and type More macros and DSLs

Book Information

ISBN-13: 978-1935182597

Product Dimensions: 7.4 x 0.9 x 9.2 inches

Shipping Weight: 1.5 pounds (View shipping rates and policies)

Average Customer Review: 4.4 out of 5 starsÂ Â See all reviewsÂ (9 customer reviews)

Best Sellers Rank: #1,574,608 in Books (See Top 100 in Books)   #89 inÂ Books > Computers & Technology > Programming > Languages & Tools > Lisp   #1599 inÂ Books > Computers & Technology > Programming > Languages & Tools > Java   #5070 inÂ Books > Textbooks > Computer Science > Programming Languages

Customer Reviews

I'm coming to Clojure from C# and, as the author puts it, it's a daunting journey. There is a lot to learn. The functional paradigm, lisp specifics (macros etc), clojure specifics (STM, persitent data structures, protocols, etc). It's not like learning python or ruby. It's fundamentally differnt ball game. I read a lot of stuff online and I read "Practical Clojure" that was published before this book but all of them were ranting about how great and superior Clojure is to everything else which wasn't very helpful and actually is annoying.This book was different. It's all practical and zero BS and it shows step by step the clojure way. For example I liked how clojure multi-methods polymorphism was explained. It starts by showing the limitation of single dispatch and then showing alternatives in Java through the visitor pattern and then shows clojure multimethods. All of course is done through writing code. Now I know exactly the problem that multimethods solves and I'm confident to apply them in my design.A similar example is protocols where the author builds an equivalent functionality to protocols with multi-methods and then replaces them at the end with protocols which make you understand exactly why protocls exist and when to use them.I also learned about things I wasn't expecting to learn when I bought this book like the expression problem, multiple dispatch and also HBase, Hadoop, RabbitMQ and other components that are essential to modern application architecture and how to work with them in clojure.I would like to thank the author for the apparent effort and passion he put in this book. Congratulations, job perfectly done.When you learn Clojure (and Lisp in general) you don't learn about a better technology only, you also learn about better people.

CiA has gotten me farther down the road towards understanding functional programming, and it includes one of the best introductions to the Lisp family and the theory behind functional programming I have read. The book strives to teach concepts from the ground-up, which is it's

biggest strength and weakness. Instead of expalining -how- you use Clojure to create programs, it also strives to give you the background behind the -why- of Clojure, delving into concepts like abstract-syntax-trees, polymorphic inheritence, and software transactional memory in depth.The author does an admirable job of covering these difficult topics in the first part of the book called "Getting Started". After discussing syntax and key concepts, the second part, "Getting Real" tries to bridge the divide between theory and practice by walking through TDD and common application implementations.The second half of the book is definitely the weaker of the two. It feels like Amit bites off more than he can chew by blasting through concepts like data storage, messaging and DSLs. For someone new to functional programming languages, this is a lot like drinking from the fire hose. Because so much ground is covered, I was left floundering.The book also glosses over the layout of the clojure architecture itself, which leads to trouble when following along with examples. For example, he seems to assume you know theres a "src" directory, and are familiar enough with Java to work through issues related to requiring files and classpaths.The book does not do an adequate job covering tools like lein which greatly simplify tasks like setting up testing environments, the REPL, etc. I think the ground-up approach is laudible, but building a real-world application realistically requires some sort of framework and set of guiding principles, and CiA fails to provide this kind of guidance. Apparently lein and other tools were not heavily used in 2011 when the book was released, and so this is a somewhat understandable omission, would love to see a second edition with another who section with this info.The good: * an excellent explanation of functional programming concepts * a good introduction to Clojure syntax and common functionsThe bad: * Too much detail, too fast * Does not cover Clojure structure or programming tools like Lein in depth * The author succumbs to the annoying language cheerleader role technical book writers always seem to fall into. For example he says on page 6: "Clojure is an extremely simple language to learn; from a syntax point of view, there's nearly nothing to it." This is followed by 150 pages of syntax description, and another 200 pages of implementation examples, which stretches a reasonable definition of the concept of simple.Other examples of wide-eyed optimism like "We'll write own little web framework to demonstrate how easy and straightforward Clojure is." and "Clojure makes the complicated task of writing multi- threaded programs that work correctly downright easy." are sprinkled liberally throughout the book. Learning a new language and a new programming paragigm is a marathon, and it's hard (albeit enjoyable) work, and telling me polymorphic multimethods are easy doesn't make them easy, any more than telling me mile 20 is easy magically gets me to mile 26.

This is one of the unsung greats in the Clojure book genre. This books brings the features of Clojure together and shows how they can be used cohesively to implement a number of engineering solutions. Each solution is stunningly simple and elegant. I highly recommend this book.

I find Amit's book very easy to follow. It starts with great introduction of Clojure. Then it dives into how to use Clojure in real world application. One thing that I really liked about the book is TDD in action namely test, repl and refactoring in easy to follow examples, which the book offers plenty. I strongly recommend the book to anyone as the first book to read before starting with other great Clojure books such as Joy of Clojure.

Download to continue reading...

How to Draw Action Figures: Book 2: More than 70 Sketches of Action Figures and Action Poses (Drawing Action Figures, Draw Action Figures Book, How Draw Action Poses, Draw Comic Figures) The Joy of Clojure: Thinking the Clojure Way Clojure Reactive Programming - How to Develop Concurrent and Asynchronous Applications with Clojure Clojure in Action Clojure for the Brave and True: Learn the Ultimate Language and Become a Better Programmer The Joy of Clojure Professional Clojure Mastering Clojure Macros: Write Cleaner, Faster, Smarter Code Clojure Recipes (Developer's Library) Programming Clojure (Pragmatic Programmers) Clojure for Machine Learning Clojure for Java Developers Living Clojure The Action Bible Collector's Edition: God's Redemptive Story (Action Bible Series) BMX Freestyle (Torque Books: Action Sports) (Torque: Action Sports) Tennis in Action (Sports in Action) Lacrosse in Action (Sports in Action (Paperback)) Lacrosse in Action (Sports in Action) Badminton in Action (Sports in Action (Paperback)) The Action Bible Devotional: 52 Weeks of God-Inspired Adventure (Action Bible Series)

Dmca