

Clojure Programming Chas Emerick

Save time and trouble when using Scala to build object-oriented, functional, and concurrent applications. With more than 250 ready-to-use recipes and 700 code examples, this comprehensive cookbook covers the most common problems you'll encounter when using the Scala language, libraries, and tools. It's ideal not only for experienced Scala developers, but also for programmers learning to use this JVM language. Author Alvin Alexander (creator of DevDaily.com) provides solutions based on his experience using Scala for highly scalable, component-based applications that support concurrency and distribution. Packed with real-world scenarios, this book provides recipes

Read Free Clojure Programming Chas Emerick

for: Strings, numeric types, and control structures
Classes, methods, objects, traits, and packaging
Functional programming in a variety of situations
Collections covering Scala's wealth of classes and methods
Concurrency, using the Akka Actors library
Using the Scala REPL and the Simple Build Tool (SBT)
Web services on both the client and server sides
Interacting with SQL and NoSQL databases
Best practices in Scala development

The Clojure standard library is a treasure trove of functions and macros that have been battle-tested over the years to solve the most challenging programming problems. Clojure: The Essential Reference is an extensive reference to the standard library but it doesn't read as a dull list of functions. In addition to providing clear explanations for each topic, this

Read Free Clojure Programming Chas Emerick

guide is full of real-world examples, links, and background information. Purchase of the print book includes a free eBook in PDF, Kindle, and ePub formats from Manning Publications. GIMP for Absolute Beginners gives us an easy way to use the power of GIMP, and enjoy digital photography and graphics without worrying about application features. GIMP is a powerful open source graphics and digital imaging application with almost all the features of its proprietary counterpart, Photoshop. But many of us are photographers and artists first, and we like to use our tools without having to think about them too much. GIMP for Absolute Beginners gets us to this stage without patronising users who haven't used GIMP before. But what can we actually do with photos and graphical images? These questions are answered in a

Read Free Clojure Programming Chas Emerick

photo projects chapter as well as a chapter on digital painting: Starting with making small changes to photos and adding information to them to creating paintings and drawings without photographic input. GIMP for Absolute Beginners covers the newest version of GIMP, version 2.8, so you can be sure the book covers the latest GIMP features. **The print version of this book is black & white, but the ebook version is in color on color-enabled devices. **

If you're an experienced programmer who has not worked with Clojure before, this guide is the perfect thorough but gentle introduction for you. Author Carin Meier not only provides a practical overview of this JVM language and its functional programming concepts, but also includes a complete hands-on training course to help you learn Clojure

Read Free Clojure Programming Chas Emerick

in a structured way. The first half of the book takes you through Clojure's unique design and lets you try your hand at two Clojure projects, including a web app. The holistic course in second half provides you with critical tools and resources, including ways to plug into the Clojure community.

Understand the basic structure of a Clojure expression
Learn how to shape and control code in a functional way
Discover how Clojure handles real-world state and concurrency
Take advantage of Java classes and learn how Clojure handles polymorphism
Manage and use libraries in a Clojure project
Use the core.async library for asynchronous and concurrent communication
Explore the power of macros in Clojure programming
Learn how to think in Clojure by following the book's seven-week training course

Read Free Clojure Programming Chas Emerick

The Tcl Programming Language

GIMP for Absolute Beginners

From Journeyman to Master

Enterprise Data Workflows with Cascading

Clojure for the Brave and True

Professional Clojure

*Clojure Programming"O'Reilly Media, Inc."
Learn how to build complete client-side
applications with ClojureScript, the
Clojure language variant that compiles to
optimized JavaScript. This hands-on
introduction shows you how ClojureScript
not only has similarities to*

Read Free Clojure Programming Chas Emerick

JavaScript—without the flaws—but also supports the full semantics of its parent language. You’ll delve into ClojureScript’s immutable data structures, lazy sequences, first-class functions, macros, and support for JavaScript libraries. No previous experience with Clojure or ClojureScript is necessary. If you’re familiar with JavaScript, HTML, CSS, and the DOM, you’ll quickly discover that ClojureScript has the same reach as JavaScript, but with more power. Start writing ClojureScript code with the

Read Free Clojure Programming Chas Emerick

Leiningen build system Learn how the ClojureScript compiler works to produce optimized JavaScript Use JavaScript functions and libraries directly from ClojureScript code Explore functions in Clojure's sequence library such as map, reduce, and filter Use macros to define new control structures or embed domain-specific languages Compile manually or script your own workflow with ClojureScript's compiler tools Integrate ClojureScript with Clojure on the JVM to build powerful client-server applications

Read Free Clojure Programming Chas Emerick

Anyone who develops software for a living needs a proven way to produce it better, faster, and cheaper. The Productive Programmer offers critical timesaving and productivity tools that you can adopt right away, no matter what platform you use. Master developer Neal Ford not only offers advice on the mechanics of productivity-how to work smarter, spurn interruptions, get the most out your computer, and avoid repetition-he also details valuable practices that will help you elude common traps, improve your code,

Read Free Clojure Programming Chas Emerick

*and become more valuable to your team.
You'll learn to: Write the test before you
write the code Manage the lifecycle of
your objects fastidiously Build only what
you need now, not what you might need
later Apply ancient philosophies to
software development Question authority,
rather than blindly adhere to standards
Make hard things easier and impossible
things possible through meta-programming
Be sure all code within a method is at the
same level of abstraction Pick the right
editor and assemble the best tools for the*

Read Free Clojure Programming Chas Emerick

job This isn't theory, but the fruits of Ford's real-world experience as an Application Architect at the global IT consultancy ThoughtWorks. Whether you're a beginner or a pro with years of experience, you'll improve your work and your career with the simple and straightforward principles in The Productive Programmer.

** Treats LISP as a language for commercial applications, not a language for academic AI concerns. This could be considered to be a secondary text for the Lisp course*

Read Free Clojure Programming Chas Emerick

*that most schools teach . This would appeal to students who sat through a LISP course in college without quite getting it - so a "nostalgia" approach, as in "wow-lisp can be practical..." * Discusses the Lisp programming model and environment. Contains an introduction to the language and gives a thorough overview of all of Common Lisp's main features. * Designed for experienced programmers no matter what languages they may be coming from and written for a modern audience—programmers who are familiar with languages like Java,*

Read Free Clojure Programming Chas Emerick

*Python, and Perl. * Includes several examples of working code that actually does something useful like Web programming and database access.*

The Essential Reference

Learn You a Haskell for Great Good!

Designing for Scalability with Erlang/OTP

Programming Clojure

Flex & Bison

Programming Rust

If you have a working knowledge of Haskell, this hands-on book shows you how to use the

Read Free Clojure Programming Chas Emerick

language's many APIs and frameworks for writing both parallel and concurrent programs. You'll learn how parallelism exploits multicore processors to speed up computation-heavy programs, and how concurrency enables you to write programs with threads for multiple interactions. Author Simon Marlow walks you through the process with lots of code examples that you can run, experiment with, and extend. Divided into separate sections on Parallel and Concurrent Haskell, this book also includes exercises to help you become

Read Free Clojure Programming Chas Emerick

familiar with the concepts presented: Express parallelism in Haskell with the Eval monad and Evaluation Strategies Parallelize ordinary Haskell code with the Par monad Build parallel array-based computations, using the Repa library Use the Accelerate library to run computations directly on the GPU Work with basic interfaces for writing concurrent code Build trees of threads for larger and more complex programs Learn how to build high-speed concurrent network servers Write distributed programs that run on multiple

Read Free Clojure Programming Chas Emerick

machines in a network

This book tries to put words to what most experienced programmers already know. It provides a framework for making better design choices, and a vocabulary for teams to discuss the software they collaborate on.

An insightful exploration of what social media, AI, robot technology, and the digital world are doing to our relationships with each other and with ourselves. There's no doubt that technology has made it easier to communicate. It's also easier to shut someone

Read Free Clojure Programming Chas Emerick

out when we are confronted with online discourse. Why bother to understand strangers--or even acquaintances--when you can troll them, block them, or just click "Unfriend" and never look back? However briefly satisfying that might be, it's also potentially eroding one of our most human traits: empathy. So what does the future look like when something so vital to a peaceful, healthy, and productive society is fading away? The cautionary, yet hopeful, answer is in this champion for an endangered emotion.

Read Free Clojure Programming Chas Emerick

In *The Future of Feeling*, Kaitlin Ugolik Phillips shares her own personal stories as well as those of doctors, entrepreneurs, teachers, journalists, and scientists about moving innovation and technology forward without succumbing to isolation. This book is for anyone interested in how our brains work, how they're subtly being rewired to work differently, and what that ultimately means for us as humans.

How can you overcome JavaScript language oddities and unsafe features? With this book,

Read Free Clojure Programming Chas Emerick

you'll learn how to create code that's beautiful, safe, and simple to understand and test by using JavaScript's functional programming support. Author Michael Fogus shows you how to apply functional-style concepts with Underscore.js, a JavaScript library that facilitates functional programming techniques. Sample code is available on GitHub at <https://github.com/funjs/book-source>. Fogus helps you think in a functional way to help you minimize complexity in the programs you build. If you're a JavaScript

Read Free Clojure Programming Chas Emerick

programmer hoping to learn functional programming techniques, or a functional programmer looking to learn JavaScript, this book is the ideal introduction. Use applicative programming techniques with first-class functions Understand how and why you might leverage variable scoping and closures Delve into higher-order functions—and learn how they take other functions as arguments for maximum advantage Explore ways to compose new functions from existing functions Get around JavaScript's limitations

Read Free Clojure Programming Chas Emerick

for using recursive functions Reduce, hide, or eliminate the footprint of state change in your programs Practice flow-based programming with chains and functional pipelines Discover how to code without using classes
60 Recipes for Apache CloudStack
Lisp in Small Pieces
Getting Clojure
Elements of Clojure
Thinking Functionally with Haskell
Using the CloudStack Ecosystem
What others in the trenches say about The

Read Free Clojure Programming Chas Emerick

Pragmatic Programmer... "The cool thing about this book is that it's great for keeping the programming process fresh. The book helps you to continue to grow and clearly comes from people who have been there." —Kent Beck, author of *Extreme Programming Explained: Embrace Change* "I found this book to be a great mix of solid advice and wonderful analogies!" —Martin Fowler, author of *Refactoring* and *UML Distilled* "I would buy a copy, read it twice, then tell all my colleagues to run out and grab a copy. This is a book I would never loan because I

Read Free Clojure Programming Chas Emerick

would worry about it being lost.” —Kevin Ruland, Management Science, MSG-Logistics “The wisdom and practical experience of the authors is obvious. The topics presented are relevant and useful.... By far its greatest strength for me has been the outstanding analogies—tracer bullets, broken windows, and the fabulous helicopter-based explanation of the need for orthogonality, especially in a crisis situation. I have little doubt that this book will eventually become an excellent source of useful information for journeymen programmers and

Read Free Clojure Programming Chas Emerick

expert mentors alike.” —John Lakos, author of Large-Scale C++ Software Design “This is the sort of book I will buy a dozen copies of when it comes out so I can give it to my clients.” —Eric Vought, Software Engineer “Most modern books on software development fail to cover the basics of what makes a great software developer, instead spending their time on syntax or technology where in reality the greatest leverage possible for any software team is in having talented developers who really know their craft well. An excellent book.” —Pete McBreen,

Read Free Clojure Programming Chas Emerick

Independent Consultant "Since reading this book, I have implemented many of the practical suggestions and tips it contains. Across the board, they have saved my company time and money while helping me get my job done quicker! This should be a desktop reference for everyone who works with code for a living."

—Jared Richardson, Senior Software Developer, iRenaissance, Inc. "I would like to see this issued to every new employee at my company...." —Chris Cleeland, Senior Software Engineer, Object Computing, Inc. "If I'm putting

Read Free Clojure Programming Chas Emerick

together a project, it's the authors of this book that I want. . . . And failing that I'd settle for people who've read their book." —Ward Cunningham Straight from the programming trenches, *The Pragmatic Programmer* cuts through the increasing specialization and technicalities of modern software development to examine the core process--taking a requirement and producing working, maintainable code that delights its users. It covers topics ranging from personal responsibility and career development to

Read Free Clojure Programming Chas Emerick

architectural techniques for keeping your code flexible and easy to adapt and reuse. Read this book, and you'll learn how to Fight software rot; Avoid the trap of duplicating knowledge; Write flexible, dynamic, and adaptable code; Avoid programming by coincidence; Bullet-proof your code with contracts, assertions, and exceptions; Capture real requirements; Test ruthlessly and effectively; Delight your users; Build teams of pragmatic programmers; and Make your developments more precise with automation. Written as a series of self-contained sections

Read Free Clojure Programming Chas Emerick

and filled with entertaining anecdotes, thoughtful examples, and interesting analogies, The Pragmatic Programmer illustrates the best practices and major pitfalls of many different aspects of software development. Whether you're a new coder, an experienced programmer, or a manager responsible for software projects, use these lessons daily, and you'll quickly see improvements in personal productivity, accuracy, and job satisfaction. You'll learn skills and develop habits and attitudes that form the foundation for long-term success in your career.

Read Free Clojure Programming Chas Emerick

You'll become a Pragmatic Programmer.

The Tcl Programming Language is a comprehensive guide to the current version (8.6) of this immensely flexible and versatile language. Starting with the basic features, it expands its scope to include the more advanced concepts, facilities and programming idioms from which the language derives its power. Begin with the basics of Tcl syntax and commands for operating on data. Get acquainted with the flexible and uniform execution model that enables metaprogramming, custom control

Read Free Clojure Programming Chas Emerick

structures etc. Learn to modularize your code with namespaces, object-oriented design and packages. See how intrinsic support for Unicode and encodings makes it a breeze to localize your applications. Become conversant with the integrated event loop and how it facilitates efficient asynchronous I/O models and the reactive style of programming. Delve into Tcl's sophisticated I/O framework and write your own reflected channels, transforms and virtual file systems. Understand the built-in facilities for inter-process communication using pipes or the

Read Free Clojure Programming Chas Emerick

network. See how concurrent programming facilities like coroutines and threads can simplify your code and make it more performant. Learn how to secure your application through the use of safe interpreters for sandboxing. Interact with databases through the Tcl Database Connectivity interface. Discover how software distribution and installation headaches are eliminated with starkits and single file deployment. The breadth of coverage and numerous examples will familiarize newcomers to every aspect of Tcl programming. At the same

Read Free Clojure Programming Chas Emerick

time, the depth and level of detail, and an exhaustive index, make The Tcl Programming Language a valuable reference in every Tcl programmer's library.

If you're familiar with functional programming basics and want to gain a much deeper understanding, this in-depth guide takes you beyond syntax and demonstrates how you need to think in a new way. Software architect Neal Ford shows intermediate to advanced developers how functional coding allows you to step back a level of abstraction so you can see

Read Free Clojure Programming Chas Emerick

your programming problem with greater clarity. Each chapter shows you various examples of functional thinking, using numerous code examples from Java 8 and other JVM languages that include functional capabilities. This book may bend your mind, but you'll come away with a much better grasp of functional programming concepts. Understand why many imperative languages are adding functional capabilities Compare functional and imperative solutions to common problems Examine ways to cede control of routine chores to the runtime Learn

Read Free Clojure Programming Chas Emerick

how memoization and laziness eliminate hand-crafted solutions Explore functional approaches to design patterns and code reuse View real-world examples of functional thinking with Java 8, and in functional architectures and web frameworks Learn the pros and cons of living in a paradigmatically richer world If you're new to functional programming, check out Josh Backfield's book *Becoming Functional*. If you need to build a scalable, fault tolerant system with requirements for high availability, discover why the Erlang/OTP platform stands out

Read Free Clojure Programming Chas Emerick

for the breadth, depth, and consistency of its features. This hands-on guide demonstrates how to use the Erlang programming language and its OTP framework of reusable libraries, tools, and design principles to develop complex commercial-grade systems that simply cannot fail. In the first part of the book, you'll learn how to design and implement process behaviors and supervision trees with Erlang/OTP, and bundle them into standalone nodes. The second part addresses reliability, scalability, and high availability in your overall system design. If

Read Free Clojure Programming Chas Emerick

you're familiar with Erlang, this book will help you understand the design choices and trade-offs necessary to keep your system running. Explore OTP's building blocks: the Erlang language, tools and libraries collection, and its abstract principles and design rules Dive into the fundamentals of OTP reusable frameworks: the Erlang process structures OTP uses for behaviors Understand how OTP behaviors support client-server structures, finite state machine patterns, event handling, and runtime/code integration Write your own

Read Free Clojure Programming Chas Emerick

behaviors and special processes Use OTP's tools, techniques, and architectures to handle deployment, monitoring, and operations

Practical Common Lisp

Clojure Cookbook

Develop event-driven, scalable, and reactive microservices with real-time monitoring

Clojure

Implement Robust, Fault-Tolerant Systems

Odo Development Cookbook

Clear, practical Clojure for the professional programmer Professional

Read Free Clojure Programming Chas Emerick

Clojure is the experienced developer's guide to functional programming using the Clojure language. Designed specifically to meet the needs of professional developers, this book briefly introduces functional programming before skipping directly to the heart of using Clojure in a real-world setting. The discussion details the read-eval-print workflow that enables fast feedback loops, then dives into enterprise-level Clojure

Read Free Clojure Programming Chas Emerick

development with expert guidance on web services, testing, datomics, performance, and more. Read from beginning to end, this book serves as a clear, direct guide to Clojure programming—but the comprehensive coverage and detail makes it extraordinarily useful as a quick reference for mid-project snags. The author team includes four professional Clojure developers, ensuring professional-level instruction from a

Read Free Clojure Programming Chas Emerick

highly practical perspective. Clojure is an open-source programming language maintained and supported by Cognitect., and quickly gaining use across industries at companies like Amazon, Walmart, Facebook, Netflix, and more. This guide provides a concise, yet thorough resource for professional developers needing to quickly put Clojure to work. Parse the difference between functional and object-oriented programming Understand Clojure

Read Free Clojure Programming Chas Emerick

performance and capabilities Develop reactive web pages using ClojureScript Adopt an REPL-driven development workflow Clojure is a modern dialect of Lisp, designed for concurrency and Java compatibility. It can be used with the Java virtual machine, Microsoft's Common Language Runtime, and JavaScript engines, providing a level of both versatility and functionality that is appealing to more and more enterprise-level developers. As requirements grow

Read Free Clojure Programming Chas Emerick

increasingly complex, stepping away from imperative programming can dramatically streamline the development workflow. Professional Clojure provides the expert instruction that gets professionals up to speed and back to work quickly.

Behind every programming language lies a vision of how programs should be built. The vision behind Clojure is of a radically simple language framework holding together a sophisticated

Read Free Clojure Programming Chas Emerick

collection of programming features. Learning Clojure involves much more than just learning the mechanics of the language. To really get Clojure you need to understand the ideas underlying this structure of framework and features. You need this book: an accessible introduction to Clojure that focuses on the ideas behind the language as well as the practical details of writing code. Clojure attracts developers on the cutting edge

Read Free Clojure Programming Chas Emerick

and is arguably the best language for learning to program in the functional style without compromise. But this comes with a steep learning curve. Getting Clojure directly addresses this by teaching you how to think functionally as it teaches you the language. You'll learn about Clojure's powerful data structures and high-level functions, but you'll also learn what it means for a language to be functional, and how to think in

Read Free Clojure Programming Chas Emerick

Clojure's functional way. Each chapter of Getting Clojure takes a feature or two or three from the language, explains the syntax and the mechanics behind that feature so that you can make it work before digging into the deeper questions: What is the thinking behind the feature? And how does it fit in with the rest of the language? In Getting Clojure you'll learn Clojure's very simple syntax, but you'll also learn why that syntax is integral the

Read Free Clojure Programming Chas Emerick

way the language is constructed. You'll discover that most data structures in Clojure are immutable, but also why that leads to more reliable programs. And you'll see how easy it is to write Clojure functions and also how you can use those functions to build complex and capable systems. With real-world examples of how working Clojure programmers use the language, Getting Clojure will help you see the challenges of programming through the

Read Free Clojure Programming Chas Emerick

eye of experienced Clojure developers.

What You Need: You will need to some background in programming. To follow along with the examples in the book, you will need Java 6 or new, Clojure 1.8 or 1.9, and Leiningen 2.

Describes the fundamentals of Clojure, covering such topics as data structures, concurrency, macros, multimethods, JVM, REPL-oriented programming, and relational databases.

"Clojure programming ... This

Read Free Clojure Programming Chas Emerick

functional programming language not only lets you take advantage of Java libraries, services, and other JVM resources, it rivals other dynamic languages such as Ruby and Python. With this comprehensive guide, you'll learn Clojure fundamentals with examples that relate it to languages you already know" --P. [4] of cover.

Paradigm Over Syntax

Building Empathy in a Tech-Obsessed World

Read Free Clojure Programming Chas Emerick

A Gentle Introduction to Symbolic
Computation
Scala Cookbook

A Beginner's Guide

Summary The Joy of Clojure, Second Edition is a deep look at the Clojure language. Fully updated for Clojure 1.6, this new edition goes beyond just syntax to show you the "why" of Clojure and how to write fluent Clojure code. You'll learn functional and declarative approaches to programming and will master the techniques that make Clojure so elegant

and efficient. Purchase of the print book includes a free eBook in PDF, Kindle, and ePub formats from Manning Publications. About the Technology The Clojure programming language is a dialect of Lisp that runs on the Java Virtual Machine and JavaScript runtimes. It is a functional programming language that offers great performance, expressive power, and stability by design. It gives you built-in concurrency and the predictable precision of immutable and persistent data structures. And it's really, really fast. The instant you see long blocks of Java or Ruby dissolve into a few lines of Clojure, you'll know why

Read Free Clojure Programming Chas Emerick

the authors of this book call it a "joyful language." It's no wonder that enterprises like Staples are betting their infrastructure on Clojure. About the Book The Joy of Clojure, Second Edition is a deep account of the Clojure language. Fully updated for Clojure 1.6, this new edition goes beyond the syntax to show you how to write fluent Clojure code. You'll learn functional and declarative approaches to programming and will master techniques that make Clojure elegant and efficient. The book shows you how to solve hard problems related to concurrency, interoperability, and performance, and how great it can be to think in the

Clojure way. Appropriate for readers with some experience using Clojure or common Lisp. What's Inside Build web apps using ClojureScript Master functional programming techniques Simplify concurrency Covers Clojure 1.6 About the Authors Michael Fogus and Chris Houser are contributors to the Clojure and ClojureScript programming languages and the authors of various Clojure libraries and language features. Table of Contents PART 1 FOUNDATIONS Clojure philosophy Drinking from the Clojure fire hose Dipping your toes in the pool PART 2 DATA TYPES On scalars Collection types

PART 3 FUNCTIONAL PROGRAMMING Being lazy
and set in your ways **Functional programming** **PART 4**
LARGE-SCALE DESIGN Macros **Combining data**
and code **Mutation and concurrency** **Parallelism**
PART 5 HOST SYMBIOSIS **Java.next** **Why**
ClojureScript? **PART 6 TANGENTIAL**
CONSIDERATIONS **Data-oriented programming**
Performance **Thinking programs** **Clojure changes the**
way you think

If you need to parse or process text data in Linux or
Unix, this useful book explains how to use flex and
bison to solve your problems quickly. flex & bison is

the long-awaited sequel to the classic O'Reilly book, `lex & yacc`. In the nearly two decades since the original book was published, the `flex` and `bison` utilities have proven to be more reliable and more powerful than the original Unix tools. `flex & bison` covers the same core functionality vital to Linux and Unix program development, along with several important new topics. You'll find revised tutorials for novices and references for advanced users, as well as an explanation of each utility's basic usage and simple, standalone applications you can create with them. With `flex & bison`, you'll discover the wide range of uses these

flexible tools offer. Address syntax crunching that regular expressions tools can't handle Build compilers and interpreters, and handle a wide range of text processing functions Interpret code, configuration files, or any other structured format Learn key programming techniques, including abstract syntax trees and symbol tables Implement a full SQL grammar-with complete sample code Use new features such as pure (reentrant) lexers and parsers, powerful GLR parsers, and interfaces to C++ With more than 150 detailed recipes, this cookbook shows experienced Clojure developers how to solve a

variety of programming tasks with this JVM language. The solutions cover everything from building dynamic websites and working with databases to network communication, cloud computing, and advanced testing strategies. And more than 60 of the world's best Clojurians contributed recipes. Each recipe includes code that you can use right away, along with a discussion on how and why the solution works, so you can adapt these patterns, approaches, and techniques to situations not specifically covered in this cookbook. Master built-in primitive and composite data structures Create, develop and publish libraries, using

the Leiningen tool Interact with the local computer that's running your application Manage network communication protocols and libraries Use techniques for connecting to and using a variety of databases Build and maintain dynamic websites, using the Ring HTTP server library Tackle application tasks such as packaging, distributing, profiling, and logging Take on cloud computing and heavyweight distributed data crunching Dive into unit, integration, simulation, and property-based testing Clojure Cookbook is a collaborative project with contributions from some of the world's best Clojurians, whose backgrounds range

from aerospace to social media, banking to robotics, AI research to e-commerce.

Collecting data is relatively easy, but turning raw information into something useful requires that you know how to extract precisely what you need. With this insightful book, intermediate to experienced programmers interested in data analysis will learn techniques for working with data in a business environment. You'll learn how to look at data to discover what it contains, how to capture those ideas in conceptual models, and then feed your understanding back into the organization through business plans,

metrics dashboards, and other applications. Along the way, you'll experiment with concepts through hands-on workshops at the end of each chapter. Above all, you'll learn how to think about the results you want to achieve -- rather than rely on tools to think for you. Use graphics to describe data with one, two, or dozens of variables Develop conceptual models using back-of-the-envelope calculations, as well as scaling and probability arguments Mine data with computationally intensive methods such as simulation and clustering Make your conclusions understandable through reports, dashboards, and other metrics

programs Understand financial calculations, including the time-value of money Use dimensionality reduction techniques or predictive analytics to conquer challenging data analysis situations Become familiar with different open source programming environments for data analysis "Finally, a concise reference for understanding how to conquer piles of data."--Austin King, Senior Web Developer, Mozilla "An indispensable text for aspiring data scientists."--Michael E. Driscoll, CEO/Founder, Dataspora

Recipes for Object-Oriented and Functional

Programming

ClojureScript: Up and Running

Learn to Program in Lisp, One Game at a Time!

Recipes for Functional Programming

The Joy of Clojure

Common LISP

Build effective applications by applying Odo

development best practices About This Book

Each recipe stands by itself as much as possible,

so that you can jump straight into the topics you

prefer The recipes included cover all the major

development areas of Odo and the most

important techniques explained through real-

life projects From seasoned authors, learn the tricks of becoming a productive developer with the Odoo framework Who This Book Is For If you are a Python developer who wants to learn or consolidate your Odoo development skills, then this book is for you! Some experience with the JavaScript programming language and web development is required to fully benefit from the front-end chapters. What You Will Learn Install and manage Odoo environments and instances Use Models to define your application's data structures Add business logic to your applications Implement automated tests and debug Odoo apps Use back-end views to

create a user interface Get to know about the access security model and internationalization features Develop front-end website features Extend the web client with new widgets and features In Detail Odoo is a full-featured open source ERP with a focus on extensibility. The flexibility and sustainability of open source is also a key selling point of Odoo. It is built on a powerful framework for rapid application development, both for back-end applications and front-end websites. The book starts by covering Odoo installation and administration, and provides a gentle introduction to application development. It then dives deep into

several of the areas that an experienced developer will need to use. You'll learn implement business logic, adapt the UI, and extend existing features. Style and Approach These practical and easy-to-follow recipes are presented step-by-step, with dozens of hands-on recipes to boost your Odoo skills. This book can also be used as a reference guide for your daily work.

Rust is a new systems programming language that combines the performance and low-level control of C and C++ with memory safety and thread safety. Rust's modern, flexible types ensure your program is free of null pointer

Read Free Clojure Programming Chas Emerick

dereferences, double frees, dangling pointers, and similar bugs, all at compile time, without runtime overhead. In multi-threaded code, Rust catches data races at compile time, making concurrency much easier to use. Written by two experienced systems programmers, this book explains how Rust manages to bridge the gap between performance and safety, and how you can take advantage of it. Topics include: How Rust represents values in memory (with diagrams) Complete explanations of ownership, moves, borrows, and lifetimes Cargo, rustdoc, unit tests, and how to publish your code on crates.io, Rust's public package repository High-

level features like generic code, closures, collections, and iterators that make Rust productive and flexible Concurrency in Rust: threads, mutexes, channels, and atomics, all much safer to use than in C or C++ Unsafe code, and how to preserve the integrity of ordinary code that uses it Extended examples illustrating how pieces of the language fit together

This is a comprehensive account of the semantics and the implementation of the whole Lisp family of languages, namely Lisp, Scheme and related dialects. It describes 11 interpreters and 2 compilers, including very recent

techniques of interpretation and compilation. The book is in two parts. The first starts from a simple evaluation function and enriches it with multiple name spaces, continuations and side-effects with commented variants, while at the same time the language used to define these features is reduced to a simple lambda-calculus. Denotational semantics is then naturally introduced. The second part focuses more on implementation techniques and discusses precompilation for fast interpretation: threaded code or bytecode; compilation towards C. Some extensions are also described such as dynamic evaluation, reflection, macros and objects. This

will become the new standard reference for people wanting to know more about the Lisp family of languages: how they work, how they are implemented, what their variants are and why such variants exist. The full code is supplied (and also available over the Net). A large bibliography is given as well as a considerable number of exercises. Thus it may also be used by students to accompany second courses on Lisp or Scheme. If you build your Scala application through Test-Driven Development, you'll quickly see the advantages of testing before you write production code. This hands-on book shows you

how to create tests with ScalaTest and the Specs2—two of the best testing frameworks available—and how to run your tests in the Simple Build Tool (SBT) designed specifically for Scala projects. By building a sample digital jukebox application, you'll discover how to isolate your tests from large subsystems and networks with mocking code, and how to use the ScalaCheck library for automated specification-based testing. If you're familiar with Scala, Ruby, or Python, this book is for you. Get an overview of Test-Driven Development Start a simple project with SBT and create tests before you write code Dive into SBT's basic commands,

***interactive mode, packaging, and history Use
ScalaTest both in the command line and with
SBT, and learn how to incorporate JUnit and
TestNG Work with the Specs2 framework,
including Specification styles, matchers DSLs,
and Data Tables Understand mocking by using
Java frameworks EasyMock and Mockito, and
the Scala-only framework ScalaMock Automate
testing by using ScalaCheck to generate fake
data***

Data Analysis with Open Source Tools

Functional Thinking

Living Clojure

Techniques for Multicore and Multithreaded

Programming

Testing in Scala

Text Processing Tools

It's all in the name: Learn You a Haskell for Great Good! is a hilarious, illustrated guide to this complex functional language. Packed with the author's original artwork, pop culture references, and most importantly, useful example code, this book teaches functional fundamentals in a way you never thought possible. You'll start with the

kid stuff: basic syntax, recursion, types and type classes. Then once you've got the basics down, the real black belt master-class begins: you'll learn to use applicative functors, monads, zippers, and all the other mythical Haskell constructs you've only read about in storybooks. As you work your way through the author's imaginative (and occasionally insane) examples, you'll learn to: -Laugh in the face of side effects as you wield purely functional

programming techniques -Use the magic of Haskell's "laziness" to play with infinite sets of data -Organize your programs by creating your own types, type classes, and modules -Use Haskell's elegant input/output system to share the genius of your programs with the outside world Short of eating the author's brain, you will not find a better way to learn this powerful language than reading Learn You a Haskell for Great Good! There is an easier way to build Hadoop

applications. With this hands-on book, you'll learn how to use Cascading, the open source abstraction framework for Hadoop that lets you easily create and manage powerful enterprise-grade data processing applications—without having to learn the intricacies of MapReduce. Working with sample apps based on Java and other JVM languages, you'll quickly learn Cascading's streamlined approach to data processing, data filtering, and workflow optimization. This book

demonstrates how this framework can help your business extract meaningful information from large amounts of distributed data. Start working on Cascading example projects right away Model and analyze unstructured data in any format, from any source Build and test applications with familiar constructs and reusable components Work with the Scalding and Cascalog Domain-Specific Languages Easily deploy applications to Hadoop, regardless of cluster location or

data size Build workflows that integrate several big data frameworks and processes Explore common use cases for Cascading, including features and tools that support them Examine a case study that uses a dataset from the Open Data Initiative

The common patterns and practices of the microservice architecture and their application using the Clojure programming language. Key Features Relevance of the microservice

architecture and benefits of Clojure's functional and simple features to implement it. Learn best practices and common principles to avoid common pitfalls while developing microservices. Learn how to use Pedestal to build your next microservices, secure them using JWT, and monitor them using the ELK stack ***Book Description*** ***The microservice architecture is sweeping the world as the de facto pattern with which to design and build scalable, easy-to-maintain web***

applications. This book will teach you common patterns and practices, and will show you how to apply these using the Clojure programming language. This book will teach you the fundamental concepts of architectural design and RESTful communication, and show you patterns that provide manageable code that is supportable in development and at scale in production. We will provide you with examples of how to put these concepts and patterns into practice with

Clojure. This book will explain and illustrate, with practical examples, how teams of all sizes can start solving problems with microservices. You will learn the importance of writing code that is asynchronous and non-blocking and how Pedestal helps us do this. Later, the book explains how to build Reactive microservices in Clojure that adhere to the principles underlying the Reactive Manifesto. We finish off by showing you various ways to monitor, test, and

secure your microservices. By the end, you will be fully capable of setting up, modifying, and deploying a microservice with Clojure and Pedestal. What you will learn Explore the pros and cons of monolithic and microservice architectures Use Clojure to effectively build a reallife application using Microservices Gain practical knowledge of the Clojure Pedestal framework and how to use it to build Microservices Explore various persistence patterns and

***learn how to use Apache Kafka to build event-driven microservice architectures
Secure your Microservices using JWT
Monitor Microservices at scale using the ELK stack
Deploy Microservices at scale using container orchestration platforms such as Kubernetes
Who this book is for
You should have a working knowledge of programming in Clojure. However, no knowledge of RESTful architecture, microservices, or web services is expected. If you are looking to apply***

techniques to your own projects, taking your first steps into microservice architecture, this book is for you. Drowning in unnecessary complexity, unmanaged state, and tangles of spaghetti code? In the best tradition of Lisp, Clojure gets out of your way so you can focus on expressing simple solutions to hard problems. Clojure cuts through complexity by providing a set of composable tools--immutable data, functions, macros, and the interactive

REPL. Written by members of the Clojure core team, this book is the essential, definitive guide to Clojure. This new edition includes information on all the newest features of Clojure, such as transducers and specs. Clojure joins the flexibility and agility of Lisp with the reach, stability, and performance of Java. Combine Clojure's tools for maximum effectiveness as you work with immutable data, functional programming, and safe concurrency to

write programs that solve real-world problems. Start by reading and understanding Clojure syntax and see how Clojure is evaluated. From there, find out about the sequence abstraction, which combines immutable collections with functional programming to create truly reusable data transformation code. Clojure is a functional language; learn how to write programs in a functional style, and when and how to use recursion to your advantage. Discover

Clojure's unique approach to state and identity, techniques for polymorphism and open systems using multimethods and protocols, and how to leverage Clojure's metaprogramming capabilities via macros. Finally, put all the pieces together in a real program. New to this edition is coverage of Clojure's spec library, one of the most interesting new features of Clojure for describing both data and functions. You can use Clojure spec to validate data, destructure data,

***explain invalid data, and generate large numbers of tests to verify the correctness of your code. With this book, you'll learn how to think in Clojure, and how to take advantage of its combined strengths to build powerful programs quickly. What You Need: Java 6 or higher
Clojure 1.9***

Land of Lisp

The Little LISPer

Functional JavaScript

A Hands-On Guide for Programmers and

Data Scientists

Grokking Simplicity

The Pragmatic Programmer

Planning to deploy and maintain a public, private, or hybrid cloud service? This cookbook's handy how-to recipes help you quickly learn and install Apache CloudStack, along with several API clients, API wrappers, data architectures, and configuration management technologies that work as part of CloudStack's ecosystem. You'll learn how to use

Read Free Clojure Programming Chas Emerick

Vagrant, Ansible, Chef, Fluentd, Libcloud, and several other open source tools that let you build and operate CloudStack better and faster. If you're an experienced programmer, system administrator, or DevOps practitioner familiar with bash, Git, package management, and some Python, you're ready to go. Learn basic CloudStack installation from source, including features such as DevCloud, the CloudStack sandbox Get a step-by-step guide for installing CloudStack from packages on Ubuntu 14.04

Read Free Clojure Programming Chas Emerick

using KVM Write your own applications on top of the CloudStack API, using CloudMonkey, Libcloud, jclouds, and CloStack Expose different APIs on CloudStack with the EC2Stack, Boto, and Eutester API wrappers Deploy applications easily, using Puppet, Salt, Ansible, Chef, and Vagrant Dive into cloud monitoring and storage with RiakCS, Fluentd, and Apache Whirr

Introduces fundamental techniques for reasoning mathematically about functional programs. Ideal for a first- or second-

Read Free Clojure Programming Chas Emerick

year undergraduate course.

Lisp has been hailed as the world's most powerful programming language, but its cryptic syntax and academic reputation can be enough to scare off even experienced programmers. Those dark days are finally over—Land of Lisp brings the power of functional programming to the people! With his brilliantly quirky comics and out-of-this-world games, longtime Lisper Conrad Barski teaches you the mysteries of Common Lisp. You'll start with the basics, like list manipulation, I/O, and recursion,

Read Free Clojure Programming Chas Emerick

then move on to more complex topics like macros, higher order programming, and domain-specific languages. Then, when your brain overheats, you can kick back with an action-packed comic book interlude! Along the way you'll create (and play) games like Wizard Adventure, a text adventure with a whiskey-soaked twist, and Grand Theft Wumpus, the most violent version of Hunt the Wumpus the world has ever seen. You'll learn to: –Master the quirks of Lisp's syntax and semantics –Write concise and elegant functional programs –Use

Read Free Clojure Programming Chas Emerick

macros, create domain-specific languages, and learn other advanced Lisp techniques –Create your own web server, and use it to play browser-based games –Put your Lisp skills to the test by writing brain-melting games like Dice of Doom and Orc Battle With Land of Lisp, the power of functional programming is yours to wield. For weeks, months–nay!–from the very moment you were born, you’ve felt it calling to you. At long last you’ll be united with the programming language you’ve been longing for: Clojure! As a

Read Free Clojure Programming Chas Emerick

Lisp-style functional programming language, Clojure lets you write robust and elegant code, and because it runs on the Java Virtual Machine, you can take advantage of the vast Java ecosystem. Clojure for the Brave and True offers a "dessert-first" approach: you'll start playing with real programs immediately, as you steadily acclimate to the abstract but powerful features of Lisp and functional programming. Inside you'll find an offbeat, practical guide to Clojure, filled with quirky sample programs that

Read Free Clojure Programming Chas Emerick

catch cheese thieves and track glittery vampires. Learn how to: –Wield Clojure’s core functions –Use Emacs for Clojure development –Write macros to modify Clojure itself –Use Clojure’s tools to simplify concurrency and parallel programming Clojure for the Brave and True assumes no prior experience with Clojure, the Java Virtual Machine, or functional programming. Are you ready, brave reader, to meet your true destiny? Grab your best pair of parentheses—you’re about to embark on an epic journey into the world of

Read Free Clojure Programming Chas Emerick

Clojure!

The Productive Programmer

Taming Complex Software with Functional Thinking

Microservices with Clojure

Functional Programming for the Web

Fast, Safe Systems Development

Learn the Ultimate Language and Become a Better Programmer

Even experienced developers struggle with software systems that sprawl across distributed servers and APIs, are filled with redundant code, and are difficult to reliably test and modify. Grokking Simplicity is a friendly, practical guide that will change the way you

Read Free Clojure Programming Chas Emerick

approach software design and development. Even experienced developers struggle with software systems that sprawl across distributed servers and APIs, are filled with redundant code, and are difficult to reliably test and modify. Grokking Simplicity is a friendly, practical guide that will change the way you approach software design and development. Grokking Simplicity guides you to a crystal-clear understanding of why certain features of modern software are so prone to complexity and introduces you to the functional techniques you can use to simplify these systems so that they're easier to read, test, and debug. Through hands-on examples, exercises, and numerous self-assessments, you'll learn to organize your code for maximum reusability and internalize methods to keep unwanted complexity out of your codebase. Purchase of the print book includes a free eBook in PDF, Kindle, and ePub formats from

Read Free Clojure Programming Chas Emerick

Manning Publications.

Highly accessible treatment covers cons cell structures, evaluation rules, programs as data, recursive and applicable programming styles. Nearly 400 illustrations, answers to exercises, "toolkit" sections, and a variety of complete programs. 1990 edition.

Clojure Programming

A Comprehensive Guide

Parallel and Concurrent Programming in Haskell

Introducing Functional Programming with Underscore.js

The Future of Feeling

Build Your Functional Skills One Idea at a Time