
Learning Concurrent Programming In Scala

John DeGoes- Asynchronous and Concurrent Programming in Scala Part 1- λC 2019
The Red Book - \"Functional Programming in #Scala\" Review Safe and Scalable
Concurrent Programming in Scala - Philipp Haller Adam Warski - Concurrency in
Scala and on the JVM Best Books To Learn Scala Quick start for Chymyst in Scala
(Concurrent programming in the Chemical Machine) Concurrent programming in
2019: Akka, Monix or ZIO? - Adam Warski Introduction to functional programming in
Scala Playing with Scala Programming Language Scala Language | Scala Tutorial For
Beginners | Scala Functional Programming | Edureka Scala Programming Full Course
| Scala tutorial For Beginners | Part 1 The Laws of Programming with Concurrency
Building a Data Science Platform in Scala Composable Futures with Akka 2.0 Scala
Tutorial | Scala Tutorial For Beginners | Scala Programming | Spark Training |
Edureka Scala Tutorial Engineering Concurrent Library Components \"Programming
in #Scala\" Review Functional Concurrency in Scala with ZIO - Itamar Ravid Manning
Introduces: Learn Concurrent Programming with Go John DeGoes- Asynchronous and
Concurrent Programming in Scala Part 2- λC 2019 \"Concurrency Options on the
JVM\" by Jessica Kerr Can we make concurrency in Scala safer?—Philipp Haller Scala /
Akka Concurrency Options Concurrency in Scala Scala Programming - Introduction to
Threads and Futures John DeGoes- Asynchronous and Concurrent Programming in
Scala Part 3- λC 2019 Concurrency on the JVM using Scala
Scala Cookbook
Concurrent Application Development using Akka with Scala
Practical Functional Programming for the JVM
Scala Programming Projects
A Beginner's Guide
Scala Reactive Programming
Programming in Scala
Learning Concurrent Programming in Scala
Learn You Some Erlang for Great Good!
Tools for Better Concurrency, Abstraction, and Agility
Scala Functional Programming Patterns
Learning Scala Programming
Design Principles and Patterns
Scalability = Functional Programming + Objects
Hands-on Scala Programming: Learn Scala in a Practical, Project-Based Way
Scala for Machine Learning
A comprehensive guide covering functional and reactive programming with Scala
2.13, Akka, and Lagom
Functional techniques for sequential and parallel programming with Scala

OROZCO REILLY

Scala Cookbook Packt Publishing Ltd Offers information on how to exploit the parallel architectures in a computer's GPU to improve code performance, scalability, and resilience.

Concurrent Application Development using Akka with Scala "O'Reilly Media, Inc."

Save time and trouble building object-oriented, functional, and concurrent applications with Scala. The latest edition of this comprehensive cookbook is packed with more than 250 ready-to-use recipes and 1,000 code examples to help you solve the most common problems when working with Scala 3 and its popular libraries. Scala changes the way you think about programming--and that's a good thing. Whether you're working on web, big data, or distributed applications, this cookbook provides recipes based on real-world scenarios for both experienced Scala developers and programmers just learning to use this JVM language. Author Alvin Alexander includes practical solutions from his experience using Scala for component-based, highly scalable applications that support concurrency and distribution. Recipes cover: Strings, numbers, and control structures Classes, methods, objects, traits, packaging, and imports Functional programming techniques Scala's wealth of collections classes and methods Building and publishing Scala applications with sbt Actors and concurrency with Scala Future and Akka Typed Popular libraries, including Spark, Scala.js, Play Framework, and GraalVM

Types, such as variance, givens, intersections, and unions Best practices, including pattern matching, modules, and functional error handling

Practical Functional Programming for the JVM "O'Reilly Media, Inc."

Explains how to use Java's portable platforms to program and use threads effectively and efficiently while avoiding common mistakes

Scala Programming Projects Learning

Concurrent Programming in ScalaThis book is a must-have tutorial for software developers aiming to write concurrent programs in Scala, or broaden their existing knowledge of concurrency.This book is intended for Scala programmers that have no prior knowledge about concurrent programming, as well as those seeking to broaden their existing knowledge about concurrency. Basic knowledge of the Scala programming language will be helpful. Readers with a solid knowledge in another programming language, such as Java, should find this book easily accessible.Learning Concurrent Programming in Scala - Second EditionLearn the art of building intricate, modern, scalable, and concurrent applications using ScalaAbout This Book* Make the most of Scala by understanding its philosophy and harnessing the power of multicores* Get acquainted with cutting-edge technologies in the field of concurrency, through practical, real-world applications* Get this step-by-step guide packed with pragmatic examplesWho This Book Is ForIf you are a Scala programmer with no prior knowledge about concurrent programming, or seeking to broaden your existing knowledge about concurrency, this book is for you. Basic knowledge of the Scala

programming language will be helpful. Also if you have a solid knowledge in another programming language, such as Java, you should find this book easily accessible.

What You Will Learn*

- Get to grips with the fundamentals of concurrent programming on modern multiprocessor systems, with a particular focus on the JVM concurrency model*
- Build high-performance concurrent systems from simple, low-level concurrency primitives*
- Express asynchrony in concurrent computations with futures and promises*
- Seamlessly accelerate sequential programs by using data-parallel collections*
- Design safe, scalable, and easy-to-comprehend in-memory transactional data models*
- Transparently create distributed applications that scale across multiple machines*
- Integrate different concurrency frameworks together in large applications*
- Develop and implement scalable and easy-to-understand concurrent applications in Scala 2.12

In Detail

Scala is a modern, multiparadigm programming language designed to express common programming patterns in a concise, elegant, and type-safe way. Scala smoothly integrates the features of object-oriented and functional languages.

In this second edition, you will find an updated coverage of the Scala 2.12 platform. The Scala 2.12 series targets Java 8 and requires it for execution. It starts by introducing you to the foundations of concurrent programming on the JVM, outlining the basics of the Java Memory Model, and then shows some of the classic building blocks of concurrency, such as the atomic variables, thread pools, and concurrent data structures, along with the caveats of traditional concurrency. It then walks you through different high-

level concurrency abstractions, each tailored toward a specific class of programming tasks, while touching on the latest advancements of Async programming capabilities of Scala. It also covers some useful patterns and idioms to use the techniques described. Finally, the book presents an overview of when to use which concurrency library and demonstrates how they all work together.

Learning Concurrent Programming in Scala

This book is an in-depth introduction to Erlang, a programming language ideal for any situation where concurrency, fault tolerance, and fast response is essential. Erlang is gaining widespread adoption with the advent of multi-core processors and their new scalable approach to concurrency. With this guide you'll learn how to write complex concurrent programs in Erlang, regardless of your programming background or experience. Written by leaders of the international Erlang community -- and based on their training material -- Erlang Programming focuses on the language's syntax and semantics, and explains pattern matching, proper lists, recursion, debugging, networking, and concurrency. This book helps you:

- Understand the strengths of Erlang and why its designers included specific features
- Learn the concepts behind concurrency and Erlang's way of handling it
- Write efficient Erlang programs while keeping code neat and readable
- Discover how Erlang fills the requirements for distributed systems
- Add simple graphical user interfaces with little effort
- Learn Erlang's tracing mechanisms for debugging concurrent and distributed systems
- Use the built-in Mnesia database and other table storage features

Erlang Programming provides exercises at the end of each chapter and

simple examples throughout the book.

A BEGINNER'S GUIDE

Packt Publishing Ltd

Erlang is the language of choice for programmers who want to write robust, concurrent applications, but its strange syntax and functional design can intimidate the uninitiated. Luckily, there's a new weapon in the battle against Erlang-phobia: *Learn You Some Erlang for Great Good!* Erlang maestro Fred Hébert starts slow and eases you into the basics: You'll learn about Erlang's unorthodox syntax, its data structures, its type system (or lack thereof!), and basic functional programming techniques. Once you've wrapped your head around the simple stuff, you'll tackle the real meat-and-potatoes of the language: concurrency, distributed computing, hot code loading, and all the other dark magic that makes Erlang such a hot topic among today's savvy developers. As you dive into Erlang's functional fantasy world, you'll learn about: -Testing your applications with EUnit and Common Test -Building and releasing your applications with the OTP framework -Passing messages, raising errors, and starting/stopping processes over many nodes -Storing and retrieving data using Mnesia and ETS -Network programming with TCP, UDP, and the inet module -The simple joys and potential pitfalls of writing distributed, concurrent applications Packed with lighthearted illustrations and just the right mix of offbeat and practical example programs, *Learn You Some Erlang for Great Good!* is the perfect entry point into the sometimes-crazy, always-thrilling world of Erlang. [Scala Reactive Programming](#) "O'Reilly Media, Inc."

In large projects, programmers tend to

get overwhelmed by their complexity. It can be hard to keep track of all the interdependencies in the code-base and how its state changes on runtime. The solution to these problems is Functional Programming, a paradigm specifically designed to deal with the complexity of software development. [Mastering ... Programming in Scala](#) Packt Publishing Ltd

Hands-on Scala teaches you how to use the Scala programming language in a practical, project-based fashion. This book is designed to quickly teach an existing programmer everything needed to go from "hello world" to building production applications like interactive websites, parallel web crawlers, and distributed systems in Scala. In the process you will learn how to use the Scala language to solve challenging problems in an elegant and intuitive manner.

LEARNING CONCURRENT PROGRAMMING IN SCALA

Addison-Wesley Professional

Describes how to use Scala to create applications for the Java VM.

[Learn You Some Erlang for Great Good!](#) Packt Publishing Ltd

Summary Scala in Action is a comprehensive tutorial that introduces Scala through clear explanations and numerous hands-on examples. Because Scala is a rich and deep language, it can be daunting to absorb all the new concepts at once. This book takes a "how-to" approach, explaining language concepts as you explore familiar programming challenges that you face in your day-to-day work. About the Technology *Scala* runs on the JVM and combines object-orientation with functional programming. It's designed to produce succinct, type-safe code, which

is crucial for enterprise applications. Scala implements Actor-based concurrency through the amazing Akka framework, so you can avoid Java's messy threading while interacting seamlessly with Java. About this Book Scala in Action is a comprehensive tutorial that introduces the language through clear explanations and numerous hands-on examples. It takes a "how to" approach, explaining language concepts as you explore familiar programming tasks. You'll tackle concurrent programming in Akka, learn to work with Scala and Spring, and learn how to build DSLs and other productivity tools. You'll learn both the language and how to use it. Experience with Java is helpful but not required. Ruby and Python programmers will also find this book accessible. What's Inside A Scala tutorial How to use Java and Scala open source libraries How to use SBT Test-driven development Debugging Updated for Scala 2.10 Purchase of the print book includes a free eBook in PDF, Kindle, and ePub formats from Manning Publications. About the Author Nilanjan Raychaudhuri is a skilled developer, speaker, and an avid polyglot programmer who works with Scala on production systems. Table of Contents PART 1 SCALA: THE BASICS Why Scala? Getting started OOP in Scala Having fun with functional data structures Functional programming PART 2 WORKING WITH SCALA Building web applications in functional style Connecting to a database Building scalable and extensible components Concurrency programming in Scala Building confidence with testing PART 3 ADVANCED STEPS Interoperability between Scala and Java Scalable and distributed applications using Akka

TOOLS FOR BETTER CONCURRENCY, ABSTRACTION, AND AGILITY

Artima Press

More than ever, learning to program concurrency is critical to creating faster, responsive applications. Speedy and affordable multicore hardware is driving the demand for high-performing applications, and you can leverage the Java platform to bring these applications to life. Concurrency on the Java platform has evolved, from the synchronization model of JDK to software transactional memory (STM) and actor-based concurrency. This book is the first to show you all these concurrency styles so you can compare and choose what works best for your applications. You'll learn the benefits of each of these models, when and how to use them, and what their limitations are. Through hands-on exercises, you'll learn how to avoid shared mutable state and how to write good, elegant, explicit synchronization-free programs so you can create easy and safe concurrent applications. The techniques you learn in this book will take you from dreading concurrency to mastering and enjoying it. Best of all, you can work with Java or a JVM language of your choice - Clojure, JRuby, Groovy, or Scala - to reap the growing power of multicore hardware. If you are a Java programmer, you'd need JDK 1.5 or later and the Akka 1.0 library. In addition, if you program in Scala, Clojure, Groovy or JRuby you'd need the latest version of your preferred language. Groovy programmers will also need GPars.

Scala Functional Programming Patterns
"O'Reilly Media, Inc."

Software -- Programming Languages.

Learning Scala Programming

Pragmatic Bookshelf

Software development today is embracing functional programming (FP), whether it's for writing concurrent programs or for managing Big Data. Where does that leave Java developers? This concise book offers a pragmatic, approachable introduction to FP for Java developers or anyone who uses an object-oriented language. Dean Wampler, Java expert and author of *Programming Scala* (O'Reilly), shows you how to apply FP principles such as immutability, avoidance of side-effects, and higher-order functions to your Java code. Each chapter provides exercises to help you practice what you've learned. Once you grasp the benefits of functional programming, you'll discover that it improves all of the code you write. Learn basic FP principles and apply them to object-oriented programming Discover how FP is more concise and modular than OOP Get useful FP lessons for your Java type design—such as avoiding nulls Design data structures and algorithms using functional programming principles Write concurrent programs using the Actor model and software transactional memory Use functional libraries and frameworks for Java—and learn where to go next to deepen your functional programming skills

DESIGN PRINCIPLES AND PATTERNS

Packt Publishing Ltd
Learn the art of building intricate, modern, scalable, and concurrent applications using Scala About This Book Make the most of Scala by understanding its philosophy and harnessing the power of multicores Get acquainted with cutting-edge technologies in the field of concurrency, through practical, real-world applications Get this step-by-step guide packed with pragmatic examples Who This Book Is

For If you are a Scala programmer with no prior knowledge about concurrent programming, or seeking to broaden your existing knowledge about concurrency, this book is for you. Basic knowledge of the Scala programming language will be helpful. Also if you have a solid knowledge in another programming language, such as Java, you should find this book easily accessible. What You Will Learn Get to grips with the fundamentals of concurrent programming on modern multiprocessor systems Build high-performance concurrent systems from simple, low-level concurrency primitives Express asynchrony in concurrent computations with futures and promises Seamlessly accelerate sequential programs by using data-parallel collections Design safe, scalable, and easy-to-comprehend in-memory transactional data models Transparently create distributed applications that scale across multiple machines Integrate different concurrency frameworks together in large applications Develop and implement scalable and easy-to-understand concurrent applications in Scala 2.12 In Detail Scala is a modern, multiparadigm programming language designed to express common programming patterns in a concise, elegant, and type-safe way. Scala smoothly integrates the features of object-oriented and functional languages. In this second edition, you will find updated coverage of the Scala 2.12 platform. The Scala 2.12 series targets Java 8 and requires it for execution. The book starts by introducing you to the foundations of concurrent programming on the JVM, outlining the basics of the Java Memory Model, and then shows some of the classic building blocks of concurrency,

such as the atomic variables, thread pools, and concurrent data structures, along with the caveats of traditional concurrency. The book then walks you through different high-level concurrency abstractions, each tailored toward a specific class of programming tasks, while touching on the latest advancements of async programming capabilities of Scala. It also covers some useful patterns and idioms to use with the techniques described. Finally, the book presents an overview of when to use which concurrency library and demonstrates how they all work together, and then presents new exciting approaches to building concurrent and distributed systems. Style and approach The book provides a step-by-step introduction to concurrent programming. It focuses on easy-to-understand examples that are pragmatic and applicable to real-world applications. Different topics are approached in a bottom-up fashion, gradually going from the simplest foundations to the most advanced features.

"O'Reilly Media, Inc."

Summary Functional Programming in Scala is a serious tutorial for programmers looking to learn FP and apply it to the everyday business of coding. The book guides readers from basic techniques to advanced topics in a logical, concise, and clear progression. In it, you'll find concrete examples and exercises that open up the world of functional programming. Purchase of the print book includes a free eBook in PDF, Kindle, and ePub formats from Manning Publications. About the Technology Functional programming (FP) is a style of software development emphasizing functions that don't depend on program state. Functional code is easier to test

and reuse, simpler to parallelize, and less prone to bugs than other code. Scala is an emerging JVM language that offers strong support for FP. Its familiar syntax and transparent interoperability with Java make Scala a great place to start learning FP. About the Book Functional Programming in Scala is a serious tutorial for programmers looking to learn FP and apply it to their everyday work. The book guides readers from basic techniques to advanced topics in a logical, concise, and clear progression. In it, you'll find concrete examples and exercises that open up the world of functional programming. This book assumes no prior experience with functional programming. Some prior exposure to Scala or Java is helpful. What's Inside Functional programming concepts The whys and hows of FP How to write multicore programs Exercises and checks for understanding About the Authors Paul Chiusano and Rúnar Bjarnason are recognized experts in functional programming with Scala and are core contributors to the Scalaz library. Table of Contents PART 1 INTRODUCTION TO FUNCTIONAL PROGRAMMING What is functional programming? Getting started with functional programming in Scala Functional data structures Handling errors without exceptions Strictness and laziness Purely functional state PART 2 FUNCTIONAL DESIGN AND COMBINATOR LIBRARIES Purely functional parallelism Property-based testing Parser combinators PART 3 COMMON STRUCTURES IN FUNCTIONAL DESIGN Monoids Monads Applicative and traversable functors PART 4 EFFECTS AND I/O External effects and I/O Local effects and mutable state Stream processing and incremental I/O *Scalability = Functional Programming +*

Objects Simon and Schuster

Get up to speed on Scala, the JVM language that offers all the benefits of a modern object model, functional programming, and an advanced type system. Packed with code examples, this comprehensive book shows you how to be productive with the language and ecosystem right away, and explains why Scala is ideal for today's highly scalable, data-centric applications that support concurrency and distribution. This second edition covers recent language features, with new chapters on pattern matching, comprehensions, and advanced functional programming. You'll also learn about Scala's command-line tools, third-party tools, libraries, and language-aware plugins for editors and IDEs. This book is ideal for beginning and advanced Scala developers alike.

Program faster with Scala's succinct and flexible syntax Dive into basic and advanced functional programming (FP) techniques Build killer big-data apps, using Scala's functional combinators Use traits for mixin composition and pattern matching for data extraction Learn the sophisticated type system that combines FP and object-oriented programming concepts Explore Scala-specific concurrency tools, including Akka Understand how to develop rich domain-specific languages Learn good design techniques for building scalable and robust Scala applications

HANDS-ON SCALA PROGRAMMING: LEARN SCALA IN A PRACTICAL, PROJECT-BASED WAY

Packt Publishing Ltd

This book is a must-have tutorial for software developers aiming to write concurrent programs in Scala, or broaden their existing knowledge of

concurrency. This book is intended for Scala programmers that have no prior knowledge about concurrent programming, as well as those seeking to broaden their existing knowledge about concurrency. Basic knowledge of the Scala programming language will be helpful. Readers with a solid knowledge in another programming language, such as Java, should find this book easily accessible.

SCALA FOR MACHINE LEARNING

Simon and Schuster

Building scalable, concurrent systems is hard. Think parallelism and we think about threads. Using threads for parallelism is not only difficult but also not scalable. You can only create a certain number of threads on a box. On the other hand you can create a million Akka actors on a box. Also it is difficult to scale out (parallel scaling) using threads. Any multi threaded application likely has the keyword synchronized peppered throughout the code base. It does not follow any pattern which makes the code base and application difficult to manage and it also indicates the difficulty in managing threads. The Akka framework has brought Actors to the Java virtual machine. Actors way of doing parallelism is way simpler than multi-threaded applications. Scala is a modern programming language for the Java Virtual Machine (JVM) that combines the best features of object-oriented and functional programming languages. Using Scala one can write concise programs with the power of concurrency. Since Scala runs on the JVM, it can access any Java library and is interoperable with Java frameworks. This book is a practical guide to use Akka along with power of Scala to design business solutions for scalability, fault

tolerant concurrent systems. It introduces the topics of concurrency, Scala and Akka which are then blended together to provide a solution on steroids for modern day web scale applications. What you'll learn Scala, SBT: Functional concepts, starting a project in Scala using sbt Akka Toolkit: Actors, Futures, FSM, TestKit Akka Clusters (upcoming and latest addition) Application of technical concepts to business problems A list of business situations which can benefit by leveraging the stack Who this book is for It is meant for Java and Scala developers, architects and product development teams. It is also meant for stakeholders who wish to see what business problems can be solved easily with this stack.

[A comprehensive guide covering functional and reactive programming with Scala 2.13, Akka, and Lagom](#) Packt Publishing Ltd

A definitive guide to mastering and implementing concurrency patterns in your applications Key Features Build scalable apps with patterns in multithreading, synchronization, and functional programming Explore the parallel programming and multithreading techniques to make the code run faster Efficiently use the techniques outlined to build reliable applications Book Description Selecting the correct concurrency architecture has a significant impact on the design and performance of your applications. This book explains how to leverage the different characteristics of parallel architecture to make your code faster and more efficient. To start with, you'll understand the basic concurrency concepts and explore patterns around explicit locking, lock free programming, futures & actors. Then, you'll get insights

into different concurrency models and parallel algorithms and put them to practice in different scenarios to realize your application's true potential. We'll take you through multithreading design patterns, such as master, slave, leader, follower, map-reduce, and monitor, also helping you to learn hands-on coding using these patterns. Once you've grasped all of this, you'll move on to solving problems using synchronizer patterns. You'll discover the rationale for these patterns in distributed & parallel applications, followed by studying how future composition, immutability and the monadic flow help create more robust code. Toward the end of the book, you'll learn about the actor paradigm and actor patterns - the message passing concurrency paradigm. What you will learn Explore parallel architecture Get acquainted with concurrency models Internalize design themes by implementing multithreading patterns Get insights into concurrent design patterns Discover design principles behind many java threading abstractions Work with functional concurrency patterns Who this book is for This is a must-have guide for developers who want to learn patterns to build scalable and high-performing apps. It's assumed that you already have a decent level of programming knowledge.

Functional techniques for sequential and parallel programming with Scala

Pragmatic Bookshelf

Learn the art of building intricate, modern, scalable, and concurrent applications using Scala About This Book* Make the most of Scala by understanding its philosophy and harnessing the power of multicores* Get acquainted with cutting-edge technologies in the field of concurrency, through practical, real-world

applications* Get this step-by-step guide packed with pragmatic examplesWho This Book Is ForIf you are a Scala programmer with no prior knowledge about concurrent programming, or seeking to broaden your existing knowledge about concurrency, this book is for you. Basic knowledge of the Scala programming language will be helpful. Also if you have a solid knowledge in another programming language, such as Java, you should find this book easily accessible.What You Will Learn* Get to grips with the fundamentals of concurrent programming on modern multiprocessor systems, with a particular focus on the JVM concurrency model* Build high-performance concurrent systems from simple, low-level concurrency primitives* Express asynchrony in concurrent computations with futures and promises* Seamlessly accelerate sequential programs by using data-parallel collections* Design safe, scalable, and easy-to-comprehend in-memory transactional data models* Transparently create distributed applications that scale across multiple machines* Integrate different concurrency frameworks together in large applications* Develop and implement scalable and easy-to-understand concurrent applications in Scala 2.12In DetailScala is a modern, multiparadigm programming language designed to express common programming patterns in a concise, elegant, and type-safe way. Scala smoothly integrates the features of object-oriented and functional languages.In this second edition, you will find an updated coverage of the Scala 2.12 platform. The Scala 2.12 series targets Java 8 and requires it for execution. It starts by introducing you to the foundations of concurrent

programming on the JVM, outlining the basics of the Java Memory Model, and then shows some of the classic building blocks of concurrency, such as the atomic variables, thread pools, and concurrent data structures, along with the caveats of traditional concurrency.It then walks you through different high-level concurrency abstractions, each tailored toward a specific class of programming tasks, while touching on the latest advancements of Async programming capabilities of Scala. It also covers some useful patterns and idioms to use the techniques described. Finally, the book presents an overview of when to use which concurrency library and demonstrates how they all work together.

CONCURRENT PATTERNS AND BEST PRACTICES

Pearson Education

A multi-user game, web site, cloud application, or networked database can have thousands of users all interacting at the same time. You need a powerful, industrial-strength tool to handle the really hard problems inherent in parallel, concurrent environments. You need Erlang. In this second edition of the bestselling *Programming Erlang*, you'll learn how to write parallel programs that scale effortlessly on multicore systems. Using Erlang, you'll be surprised at how easy it becomes to deal with parallel problems, and how much faster and more efficiently your programs run. That's because Erlang uses sets of parallel processes-not a single sequential process, as found in most programming languages. Joe Armstrong, creator of Erlang, introduces this powerful language in small steps, giving you a complete overview of Erlang and how to use it in common scenarios. You'll

start with sequential programming, move to parallel programming and handling errors in parallel programs, and learn to work confidently with distributed programming and the standard Erlang/Open Telecom Platform (OTP) frameworks. You need no previous knowledge of functional or parallel programming. The chapters are packed with hands-on, real-world tutorial examples and insider tips and advice, and finish with exercises for both beginning and advanced users. The second edition has been extensively rewritten. New to this edition are seven

chapters covering the latest Erlang features: maps, the type system and the Dialyzer, WebSockets, programming idioms, and a new stand-alone execution environment. You'll write programs that dynamically detect and correct errors, and that can be upgraded without stopping the system. There's also coverage of rebar (the de facto Erlang build system), and information on how to share and use Erlang projects on github, illustrated with examples from cowboy and bitcask. Erlang will change your view of the world, and of how you program. What You Need The Erlang/OTP system. Download it from erlang.org.

Related with Learning Concurrent Programming In Scala:

[© Learning Concurrent Programming In Scala Records Management Navy Training](#)

[© Learning Concurrent Programming In Scala Realidades 2 Capitulo 2a Workbook Answers](#)

[© Learning Concurrent Programming In Scala Record Store Day History](#)