

1. Record Nr.	UNINA9910523890303321
Autore	Feigenbaum Barry A.
Titolo	Go for Java programmers : learn the Google Go programming language // Barry Feigenbaum
Pubbl/distr/stampa	Berkeley, California : , : Apress Media LLC, , [2022] ©2022
ISBN	1-4842-7199-8
Descrizione fisica	1 online resource (596 pages)
Disciplina	005.133
Soggetti	Go (Computer program language) Java (Computer program language)
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Note generali	Includes index.
Nota di contenuto	Intro -- Table of Contents -- About the Author -- About the Technical Reviewer -- Acknowledgments -- Introduction -- Assessments -- Part I: First Look at Go -- Chapter 1: A Brief Look at Go vs. Java -- Go Is a Compiled (vs. Interpreted, As Java Is) Language -- Go and Java Share Similar Program Structure -- Go and Java Have Some Code Style Differences That Influence How Code Is Structured -- Both Go and Java Are Procedural Languages -- Java Is an Object-Oriented (OO) Language, While Go Is Not Fully OO -- Java Is a Highly Functional Language, Go Is Less So -- Java Is a Highly Declarative Language, Go Is Less So -- Many Java Features Are Annotation Driven -- Go Does Not Support Exceptions -- Both Java and Go Use Managed Memory (Garbage Collectors) -- Go and Java Both Support Concurrency but in Different Ways -- Go's Runtime Is Simpler Than the JRE -- The Go Program Build Process Is Different -- Go and Java Have Similar Release Cycles -- Chapter 2: What Java Has That Go Does Not -- Multiple Assignments -- Statements and Operators -- Assert Statement -- While and Do Statements -- Throw Statement/Throws Clause -- Strictfp, transient, volatile, synchronized, abstract, static -- Objects and Classes (OOP) and Inner Classes, Lambdas, this, super, Explicit Constructors -- Generic Types and Methods -- Extensive Functional Programming Capabilities -- Boxing of Primitive Values -- Source Annotations -- Multiple Visibilities -- Overloaded/Overridden

Functions -- Formal Enums -- Built-in Binary Data Self-Serialization --  
Concurrent Collections -- Chapter 3: A Deeper Comparison  
of Go and Java -- Part II: The Go Language -- Chapter 4: Key  
Go Aspects -- Simple Go Program Example -- Go Packages -- Go  
Comments -- Go Build/Run Process -- Go Playground -- Go Integrated  
Development Environments -- Running Go Programs -- Building  
Go Programs -- Bytecode vs. Real Code.  
Go Command-Line Tools -- Tools Bundled in the Go Command --  
Other Tools -- Go Runs Programs Instead of Classes -- Go Memory  
Management -- Go Identifiers -- Go Scopes -- Go Scopes vs.  
Go Source Files -- Initializing Go Variables -- Lifetimes  
of Go Identifiers -- Go Module Summary -- Go Assignments  
and Expressions -- Text Formatting in Go -- Goroutines -- Issues  
with Concurrency -- Go Concurrency -- Goroutines by Example --  
Chapter 5: Go Basic Features -- Language Keywords -- Operators  
and Punctuation -- Go Operator Precedence -- Go Built-in Functions --  
Chapter 6: Go Types -- Primitive/Built-in Types -- Numbers --  
Characters and Strings -- Reference vs. Pointer Types -- Drill Down  
on Basic Data Types -- Boolean Type -- Integer Types -- Floating-  
Point Types -- When nil Can Go Wrong -- Programmer-Defined Types  
-- Arrays -- Slices -- Maps -- Functions -- Methods As Functions --  
Any Declared Type Can Have Custom Functions -- Functions As Values  
-- Structs -- Structure Fields -- Structure Literals -- Nested Structures  
-- Structure Field Alignment -- Interfaces -- Application of Interfaces  
-- Composite Interfaces -- Channels -- Variable Declarations --  
Constant Declarations -- Type Casting -- Type Testing -- Derived  
Type Declarations -- Chapter 7: Errors and Panics -- Go Errors -- Go  
Panics -- Errors and Panics Illustrated -- Chapter 8: Go Statements --  
Package and Import Statements -- Assignment Statements -- Declaring  
Variables -- Declaring Named Constants -- If/Else Statements --  
Switch Statement -- While Statement -- Do-While Statement -- For  
with Index Statement -- For over a Collection Statement -- Forever  
Statement -- Break and Continue Statements -- Goto Statement --  
Return Statement -- Defer Statement -- Go Statement -- Select  
Statement -- Chapter 9: Applications for Interfaces -- An Interface Is  
the Key -- On Dependency Injection.  
On Aspect-Oriented Programming -- Chapter 10: Go Unit Tests  
and Benchmarks -- Test Cases and Benchmarks in Go -- Test Cases  
in Java -- Chapter 11: Going into the Future -- Improved Error  
Handling -- Go Generics -- Capstone Game of Life Example -- Game  
of Life Go Source -- Capstone Project in Java -- Main.java -- Server.  
java -- Game.java -- Utility.java -- Formatter.java --  
ParameterProvider.java -- AnimatedGifWriter.java -- Logger.java --  
Capstone Project in Go -- Main.go -- Server.go -- Game.go -- Utility.  
go -- Go Doc Output -- API Outputs -- Game Outputs -- Part III: Go  
Library Survey -- Chapter 12: Key Package Comparison -- Java Lang  
Packages -- Java IO Package -- Java Text Package -- Java Time  
Packages -- Java Util Packages -- Chapter 13: Key Method/Function  
Comparison -- Chapter 14: Go Package Survey -- File Access with Go  
-- Compression Services -- Archive Packages -- Compression  
Packages -- Image -- Input/Output (I/O) -- Bytes Package -- Format  
Package -- Data Collections -- Sorting -- Context Package --  
Cryptography and Hashing and Data Encoding -- Encoding Packages --  
Unicode Encoding Packages -- Chapter 15: SQL Database Access --  
Chapter 16: Client and Server Support -- MIME Packages -- Network  
Packages -- The Net Package -- The HTTP Template Package -- The  
Net.HTTP Package -- URL Package -- Chapter 17: Go Runtime -- Errors  
Package -- Flag Package -- Log Package -- Math Package -- Operating

System Support Packages -- Reflection Package -- Regular Expression Package -- Go Runtime Packages -- String Processing Packages -- Concurrency and Goroutines -- Testing Package -- Time and Date Package -- Appendix A: Installing Go -- Appendix B: Some Go FAQs -- Appendix C: Go Gotchas to Look Out For -- Appendix D: Mark-Sweep Pseudocode -- Appendix E: ASCII vs. UTF-8 -- Index.

---

## Sommario/riassunto

Get an in-depth introduction to the Go programming language and its associated standard runtime libraries. This book is targeted towards programmers that already know the Java programming language and uses that Java knowledge to direct the learning of Go. You will get a deep understanding of the Go language and obtain a good introduction to the extensive Go standard libraries. This book teaches Go through clear descriptions of Go features, contrasting them with similar Java features and via providing extensive code examples. After reading this book you will be knowledgeable enough about Go and its libraries to begin doing effective programming using the Go language. Go for Java Programmers is structured more like a tutorial than a reference document. It covers key features of Go, but not every little detail as a reference might. Its goal is to get you competent enough in Go and its runtime that you can begin to effectively write Go programs. What You Will Learn Examine the key Go Runtime libraries and how they compare to Java libraries See when it is appropriate to use the Go language instead of the Java language Read and understand programs written in Go Write many programs in Go Determine when Go is an appropriate language to develop applications in Discover how the Go and Java languages and development experience compare and contrast Who This Book Is For Primarily existing professional Java programmers or students that already know something about Java. A basic understanding of Java is expected. Some basic programming experience with imperative languages is expected.

---