

## Build Web Application With Golang Gitbook

Whether you want to automate tasks, analyze data, parse logs, talk to network services, or address other systems requirements, writing your own command-line tool may be the fastest - and perhaps the most fun - way to do it. The Go programming language is a great choice for developing tools that are fast, reliable, and cross-platform. Create command-line tools that work with files, connect to services, and even manage external processes, all while using tests and benchmarks to ensure your programs are fast and correct. When you want to develop cross platform command-line tools that are fast and reliable, use Go, a modern programming language that combines the reliability of compiled languages with the ease of use and flexibility of dynamic typed languages. Work through practical examples to develop elegant and efficient tools by applying Go's rich standard library, its built in support for concurrency, and its expressive syntax. Use Go's integrated testing capabilities to automatically test your tools, ensuring they work reliably even across code refactoring. Develop CLI tools that interact with your users by using common input/output patterns, including environment variables and flags. Handle files to read or persist data, and manipulate paths consistently in cross-platform scenarios. Control processes and handle signals, and use a benchmark driven approach and Go's concurrency primitives to create tools that perform well. Use powerful external libraries such as Cobra to create modern and flexible tools that handle subcommands, and develop tools that interact with databases, APIs, and network services. Finally, leverage what you learned by tackling additional challenges at the end of each chapter. What You Need: Go 1.8 or higher, an internet connection to download the example files and additional libraries, and a text editor to write your programs.

This step-by-step guide to the common patterns and practices, showing you how to apply these using the Go programming language About This Book This short, concise, and practical guide is packed with real-world examples of building microservices with Go. It is easy to read and will benefit smaller teams who want to extend the functionality of their existing systems Using this practical approach will save your money in terms of maintaining a monolithic architecture and demonstrate capabilities in ease of use Who This Book Is For You should have a working knowledge of programming in Go, including writing and compiling basic applications. However, no knowledge of RESTful architecture, microservices, or web services is expected. If you are looking to apply techniques to your own project, taking your first steps into microservice architecture, this book is for you. What You Will Learn Plan a microservice architecture and design a microservice Write a microservice with a RESTful API and a database Understand the common idioms and common patterns in microservices architecture Leverage tools and automation that help build microservices become horizontally scalable Get a grounding in containerization with Docker and Docker-Compose, which will greatly accelerate your development lifecycle Manage and secure Microservices at scale with monitoring, logging, service discovery, and automation Test microservices and integrate API tests in Go. In Detail Microservice architecture is sweeping the world as the de facto pattern to build web-based applications. Golang is a language particularly well suited to building them. Its strong community, encouragement of idiomatic style, and statically-linked binary artifacts make integrating it with other technologies and managing microservices at scale consistent and intuitive. This book will teach you the common patterns and practices, showing you how to apply these using the Go programming language. It 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 on how to put these concepts and patterns into practice with Go. Whether you are planning a new application or working in an existing monolith, this book will explain and illustrate with practical examples how teams of all sizes can start solving problems with microservices. It will help you understand Docker and Docker-Compose and how it can be used to isolate microservice dependencies and build environments. We finish off by showing you various techniques to monitor, test, and secure your microservices. By the end, you will know the benefits of system resilience of a microservice and the advantages of Go stack. Style and approach The step-by-step tutorial focuses on building microservices. Each chapter expands upon the previous one, teaching you the main skills and techniques required to be a successful microservice practitioner.

Explore the necessary concepts of REST API development by building five real world services from scratch. About This Book Follow best practices and explore techniques such as clustering and caching to achieve a reliable, scalable web service Leverage the Gin Framework to quickly implement RESTful endpoints Learn to implement a client library for a RESTful web service using Go Who This Book Is For This book is intended for those who want to learn to build RESTful web services with a framework like Gin. To make best use of the code samples included in the book, you should have a basic knowledge of Go programming. What You Will Learn Create HTTP handler and intercept the Gorilla Mux router OAuth 2 implementation with Go Build RESTful API with Gin Framework Create REST API with MongoDB and Go Build a working client library and unit test for REST API Debug, test, and profile RESTful APIs with each of the frameworks Optimize and scale REST API using microservices In Detail RESTful is an architectural style that tackles the challenges of building scalable web services and in today's connected world, APIs have taken a central role on the web. APIs provide the fabric through which systems interact, and REST has become synonymous with APIs. The depth, breadth, and ease of use of Go, makes it a breeze for developers to work with it to build robust web APIs. This book takes you through the design of RESTful web services and leverages a framework like Gin to implement these services. The book starts with a brief introduction to REST API development and how it transformed the modern web. You will learn how to handle routing and authentication of web services along with working with middleware for internal service. The book explains how to use Go frameworks to build RESTful web services and work with MongoDB to create REST API. You will learn how to integrate Postgres SQL and JSON with a Go web service and build a client library in Go for consuming REST API. You will learn how to scale APIs using the microservice architecture and deploy the REST APIs using Nginx as a proxy server. Finally you will learn how to metrize a REST API using an API Gateway. By the end of the book you will be proficient in building RESTful APIs in Go. Style and Approach This book is a step-by-step, hands-on guide to designing and building RESTful web services.

Echo is a leading framework for creating web applications with the Go language. This book will show you how to develop scalable real-world web apps, RESTful services, and backend systems with Echo. After a thorough understanding of the book, you'll be introduced to all the concepts for a building real-world web system with Echo. You will start with the Go HTTP standard library, and setting up your work environment. You will move on to Echo handlers, group routing, data binding, and middleware processing. After that, you will learn how to use the Go application and use templates. By the end of this book you will be able to build your very own high performance apps using Echo. A Quick Start Guide is a focussed, shorter title which provides a faster paced introduction to a technology. They are for people who don't need all the detail at this point in the learning curve. The presentation has been streamlined to concentrate on the things you really need to know, rather than everything. What you will learn Key design considerations for high performance Echo applications How Echo handles routing over context is managed through the lifetime of the request and response pipeline Decrease complexity of your apps by developing middleware functions Interact with the response through response data bindings Interact with the response through response data renderings within the framework Use Echo's logging and error handling facilities Render all templates within Echo to allow for server side rendering of content Who this book is for You will need to know the basics of the Go language, and the general concepts of web development.

Head First Go

Level Up Your Web Apps With Go

Explore the power of Golang to secure host, web, and cloud services

Building Distributed Applications in Gin

Security with Go

Learning Website Development with Django

Go is an open-source language from Google that's a bit like C. Designed for programmer productivity, it's got a clean syntax, and emphasizes concurrency. This book gives you all you need to use Go in your web applications. You'll learn the basic concepts - language structures, the standard library, and Go tools - then tackle more advanced features like concurrency concepts, testing methodologies, and package structures. At each step, you'll get advice for better coding in Go. You'll see how to structure projects, how to use concurrency effectively, and best practices for testing - as well as many valuable hints and tips gleaned from real world experience of developing web applications with Go. You'll learn: Get to grips with Go language basics (types, the standard library, tools) Use Go with HTTP Work with images Understand concurrency Test effectively Master deployment And much more ...

Take a deep dive into web development using the Go programming language to build web apps and RESTful services to create reliable and efficient software. Web Development with Go provides Go language fundamentals and then moves on to advanced web development concepts and successful deployment of Go web apps to the cloud. Web Development with Go will teach you how to develop scalable real-world web apps, RESTful services, and backend systems with Go. The book starts off by covering Go programming language fundamentals as a prerequisite for web development. After a thorough understanding of the basics, the book delves into web development using the built-in package, net/http. With each chapter you'll be introduced to new concepts for gradually building a real-world web system. The book further shows you how to integrate Go with other technologies. For example, it provides an overview of using MongoDB as a means of persistent storage, and provides an end-to-end REST API sample as well. The book then moves on to demonstrate how to deploy web apps to the cloud using the Google Cloud platform. Web Development with Go provides: Fundamentals for building real-world web apps in Go Thorough coverage of prerequisites and practical code examples Demo web apps for attaining a deeper understanding of web development A reference REST API app which can be used to build scalable real-world backend services in Go A thorough demonstration of deploying web apps to the Cloud using the Google Cloud platform Go is a high-performance language while providing greater level of developer productivity, therefore Web Development with Go equips you with the necessary skills and knowledge required for effectively building robust and efficient web apps by leveraging the features of Go.

Docker practical techniques for building real-world web apps that are scalable, reliable, and always available. Key Features Build well-designed and secure microservices. Enrich your microservices with continuous integration and monitoring. Containerize your application with Docker Deploy your application to AWS. Learn how to utilize the powerful AWS services from within your application Book Description Awarded as one of the best books of all time by BookAuthority, Cloud Native Programming with Golang will take you on a journey into the world of microservices and cloud computing with the help of Go. Cloud computing and microservices are two very important concepts in modern software architecture. They represent key skills that ambitious software engineers need to acquire in order to design and build software applications capable of performing and scaling. Go is a modern cross-platform programming language that is very powerful yet simple; it is an excellent choice for microservices and cloud applications. Go is gaining more and more popularity, and becoming a very attractive skill. This book starts by covering the software architectural patterns of cloud applications, as well as practical concepts regarding how to scale, distribute, and deploy those applications. You will also learn how to build a JavaScript-based front-end for your application, using TypeScript and React. From there, we dive into commercial cloud offerings by covering AWS. Finally, we conclude our book by providing some overviews of other concepts and technologies that you can explore, to move from where the book leaves off. What you will learn Understand modern software applications architectures Build secure microservices that can effectively communicate with other services Get to know about event-driven architectures by diving into message queues such as Kafka, RabbitMQ, and AWS SQS. Understand key modern database technologies such as MongoDB, and Amazon's DynamoDB Leverage the power of containers Explore Amazon cloud services fundamentals Know how to utilize the power of the Go language to access key services in the Amazon cloud such as S3, SQS, DynamoDB and more. Build front-end applications using ReactJS with Go Implement CD for modern applications Who this book is for This book is for developers who want to begin building secure, resilient, robust, and scalable Go applications that are cloud native. Some knowledge of the Go programming language should be sufficient.To build the front-end application, you will also need some knowledge of JavaScript programming.

Go is rapidly becoming the preferred language for building web services. While there are plenty of tutorials available that teach Go's syntax to developers with experience in other programming languages, tutorials aren't enough. They don't teach Go's idioms, so developers end up recreating patterns that don't make sense in a Go context. This practical guide provides the essential background you need to write clear and idiomatic Go. No matter your level of experience, you'll learn how to think like a Go developer. Author Jon Bodner introduces the design patterns experienced Go developers have adopted and explores the rationale for using them. You'll also get a preview of Go's upcoming generics support and how it fits into the language. Learn how to write idiomatic code in Go and design a Go project Understand the reasons for the design decisions in Go Set up a Go development environment for a solo developer or team Learn how and when to use reflection, unsafe, and cgo Discover how Go's features allow the language to run efficiently Know which Go features you should use sparingly or not at all

Go Web Development Cookbook

The Go Programming Language

An Introduction to Programming in Go

Go Programming by Example

Network Programming with Go

Hands-On Serverless Applications with Go

Like the best-selling Black Hat Python, Black Hat Go explores the darker side of the popular Go programming language. This collection of short scripts will help you test your systems, build and automate tools to fit your needs, and improve your offensive security skillset. Black Hat Go explores the darker side of Go, the popular programming language revered by hackers for its simplicity, efficiency, and reliability. It provides an arsenal of practical tactics from the perspective of security practitioners and hackers to help you test your systems, build and automate tools to fit your needs, and improve your offensive security skillset, all using the power of Go. You'll begin your journey with a basic overview of Go's syntax and philosophy and then start to explore examples that you can leverage for tool development, including common network protocols like HTTP, DNS, and SMB. You'll then dig into various tactics and problems that penetration testers encounter, addressing things like data pilfering, packet sniffing, and exploit development. You'll create dynamic, pluggable tools before diving into cryptography, attacking Microsoft Windows, and implementing steganography. You'll learn how to: • Make performant tools that can be used for your own security projects • Create usable tools that interact with remote APIs • Scrape arbitrary HTML data • Use Go's standard package, net/http, for building HTTP servers • Write your own DNS server and proxy • Use DNS tunneling to establish a C2 channel out of a restrictive network • Create a vulnerability fuzzer to discover an application's security weaknesses • Use plugins and extensions to future-proof productsBuild an RC2 symmetric-key brute-force • Implant data within a Portable Network Graphics (PNG) image. Are you ready to add to your arsenal of security tools? Then let's Go!

Go: Building Web ApplicationsPackt Publishing Ltd Step-by-step instruction on writing your first production-ready servers with Golang Google's Go language, otherwise known as Golang, is a fast, simple, and reliable language that is rapidly becoming a highly popular choice for developers of all kinds. With particular utility in cloud-native environments, Golang is being adopted in major projects like Docker and Ethereum thanks to its user-friendly features, like concurrency and easy deployment. In Practical Golang: Building Scalable Network and Non-Network Applications, expert coder and devops engineer Amit Saha delivers a step-by-step guide to writing production-ready HTTP 1.1, HTTP2, RPC, and TCP/UDP servers. Walking you through the entire process of learning this already straightforward language, from your first application to your first deployed server, the authors rely solely on the most popular open-source projects to ensure you can apply the book's advice in any cloud environment. In this book, you'll get: Fulsome descriptions of best practices on load balancing, scaling, and failure handling Stepwise guidance on writing an HTTP service from scratch using only Golang's standard library Easy tutorials on implementing RPC and HTTP interfaces for RPC services Straightforward instructions on using SQL databases Perfect for software developers, devops engineers, and other programming professionals, Practical Golang is also an indispensable resource for anyone who wants to go beyond the basics of Golang and deploy robust and practical servers.

Take a deep dive into web development using the Go programming language to build web apps and RESTful services to create reliable and efficient software. Web Development with Go provides Go language fundamentals and then moves on to advanced web development concepts and successful deployment of Go web apps to the cloud. Web Development with Go will teach you how to develop scalable real-world web apps, RESTful services, and backend systems with Go. The book starts off by covering Go programming language fundamentals as a prerequisite for web development. After a thorough understanding of the basics, the book delves into web development using the built-in package, net/http. With each chapter you'll be introduced to new concepts for gradually building a real-world web system. The book further shows you how to integrate Go with other technologies. For example, it provides an overview of using MongoDB as a means of persistent storage, and provides an end-to-end REST API sample as well. Developers looking for a full-fledged web development framework for building web apps will be introduced to Beego. The book then moves on to demonstrate how to deploy web apps to the cloud using the Google Cloud platform. Finally, the book introduces Docker, a revolutionary container technology platform for deploying containerized Go web apps to the cloud. Web Development with Go provides: Fundamentals for building real-world web apps in Go Thorough coverage of prerequisites and practical code examples Demo web apps for attaining a deeper understanding of web development A reference REST API app which can be used to build scalable real-world backend services in Go A thorough demonstration of deploying web apps to the Cloud using the Google Cloud platform, and Docker for deploying Go servers Go is a high-performance language while providing greater level of developer productivity, therefore Web Development with Go equips you with the necessary skills and knowledge required for effectively building robust and efficient web apps by leveraging the features of Go.

Powerful Command-Line Applications in Go

Go Web Scraping Quick Start Guide

Hands-On GUI Application Development in Go

Learning Go

Web Development with Go

Go in Practice

The first step for your security needs when using Go, covering host, network, and cloud security for ethical hackers and defense against intrusion Key Features First introduction to Security with Golang Adopting a Blue Team/Red Team approach Take advantage of speed and inherent safety of Golang Works as an introduction to security for Golang developers Works as a guide to Golang security packages for recent Golang beginners Book Description Go is becoming more and more popular as a language for security experts. Its wide use in server and cloud environments, its speed and ease of use, and its evident capabilities for data analysis, have made it a prime choice for developers who need to think about security. Security with Go is the first Golang security book, and it is useful for both blue team and red team applications. With this book, you will learn how to write secure software, monitor your systems, secure your data, attack systems, and extract information. Defensive topics include cryptography, forensics, packet capturing, and building secure web applications. Offensive topics include brute force, port scanning, packet injection, and social engineering, and post exploitation techniques. What you will learn The basic concepts and principles of secure programming Write secure Golang programs and applications Understand classic patterns of attack Write Golang scripts to defend against network-level attacks Learn how to use Golang security packages Apply and explore cryptographic methods and packages Learn the art of defending against brute force attacks Secure web and cloud applications Who this book is for Security with Go is aimed at developers with basics in Go to the level that they can write their own scripts and small programs without difficulty. Readers should be familiar with security concepts, and familiarity with Python security applications and libraries is an advantage, but not a necessity.

An effective guide to learning how to build a large-scale distributed application using the wide range of functionalities in Gin Key Features Explore the commonly used functionalities of Gin to build web applications Become well-versed with rendering HTML templates with the Gin engine Solve commonly occurring challenges such as scaling, caching, and deployment Book Description Gin is a high-performance HTTP web framework used to build web applications and microservices in Go. This book is designed to teach you the ins and outs of the Gin framework with the help of practical examples. You'll start by exploring the fundamentals of the Gin framework, before progressing to build a real-world RESTful API. Along the way, you'll understand how to store and retrieve data at scale with a NoSQL database such as MongoDB, and how to implement a caching layer with Redis. Next, you'll understand how to secure and test your API endpoints with authentication protocols such as OAuth 2 and JWT. Later chapters will guide you through rendering HTML templates on the server-side and building a frontend application with the React web framework to consume API responses. Finally, you'll deploy your application on Amazon Web Services (AWS) and learn how to automate the deployment process with a continuous integration/continuous delivery (CI/CD) pipeline. By the end of this Gin book, you will be able to design, build, and deploy a production-ready distributed application from scratch using the Gin framework. What you will learn Build a production-ready REST API with the Gin framework Scale web applications with event-driven architecture Use NoSQL databases for data persistence Set up authentication middleware with JWT and Auth0 Build a Gin-based RESTful API on AWS with Docker and Kubernetes Implement a CI/CD workflow for Gin web apps Who this book is for This book is for Go developers who are comfortable with the Go language and seeking to learn REST API design and development with the Gin framework. Beginner-level knowledge of the Go programming language is required to make the most of this book.

Go is rapidly becoming the preferred language for building web services. While there are plenty of tutorials available that teach Go's syntax to developers with experience in other programming languages, but tutorials aren't enough. They don't teach Go's idioms, so developers end up recreating patterns that don't make sense in a Go context. This practical guide provides the essential background you need to write clear and idiomatic Go. No matter your level of experience, you'll learn how to think like a Go developer. Author Jon Bodner reveals design patterns that experienced Go developers have adopted and the rationale for them. You'll learn how to structure your project and choose the proper tools and libraries to create successful software. Learn how to write idiomatic code in Go and design a Go project Understand the reasons for the design decisions in Go Set up a Go development environment for a solo developer or team Learn how and when to use reflection, unsafe, and CGo Learn how Go's features allow the language to run efficiently Know which Go features you should use sparingly, or not at all Learn the future of Go, including Generics

Deep dive into the essential topics in Go programming KEY FEATURES: Understand the fundamentals of Go language, its history, purpose and success stories. Learn how to work with Variables, Constants, Data types, Operators, Control structures and Functions. Get familiar and work with the standard Golang libraries. Learn how to create custom packages and third-party package installation. Understand how concurrency is achieved in Go with the use of Goroutines, Mutex and Channels. Understand how an error is handled in Golang and supported libraries. DESCRIPTION This book is a unique read for those who are looking for an extensively covers topics in Go programming. Basics such as Data types, Control structures and Loops in have been explained in-depth. A detailed description of Structs, Interfaces, Polymorphism and Concurrency will enable you to write professional codes using Golang. You will get an idea of error data type and how to recover it in Golang. You will be capable of using standard libraries, create custom packages and install third party packages in Go. Creation of functions and invoking them in Go have been vividly explained. By the end, you will be able to write advanced Golang code and at the same time, develop an application with Golang server. WHAT YOU WILL LEARN: Learn how to write codes using Control structures and Loops in Go Get familiar with the type of Operators in Go Learn how to work with Arrays and Slices in Go Get familiar and work with the functions in Go Learn how to implement Concurrent programming in Go WHO THIS BOOK IS FOR This book is for anyone who wants to learn the Golang programming language. Programmers and developers who are currently using Golang can use this book as a reference guide. TABLE OF CONTENTS 1. Introduction to Go 2. Environment Setup 3. Beginning With Go 4. Variables, Data Types and Constants 5. Operators 6. Control Structures 7. Functions 8. Packages in Go 9. Arrays and Slices 10. Strings 11. Pointers 12. Structures 13. Composition 14. Interfaces and polymorphism 15. Maps 16. Concurrency with Go 17. Mutex & Channels 18. Error Handling 19. Reflection 20. Build Web Application

Building Microservices with Go

An Authoritative Guide to Building Microservices, Web and Enterprise Applications, and Best Practices

Build ScalableNext-Gen Web Application using Golang (English Edition)

Build responsive, cross-platform, graphical applications with the Go programming language

Build full-stack web applications with Go

A hands-on guide for Go developers to build and deploy distributed web apps with the Gin framework

What Do Docker, Kubernetes, and Prometheus have in common? All of these cloud native technologies are written in the Go programming language. This practical book shows you how to use Go's strengths to develop cloud native services that are scalable and resilient, even in an unpredictable environment. You'll explore the composition and construction of these applications, from lower-level features of Go to mid-level design patterns to high-level architectural considerations. Each chapter builds on the lessons of the last, walking intermediate to advanced developer value store. You'll learn best practices for adopting Go as your development language for solving cloud native management and deployment issues. Learn how cloud native applications differ from other software architectures Understand how Go can solve the challenges of designing scalable distributed services Leverage Go's lower-level features, such as channels and goroutines, to implement a reliable cloud native service Explore what "service reliability" is and what it has to do with cloud native Apply a variety of patterns, abstractions, and tooling to build an application Discover how to extract information from the web using various tools that perform scraping and crawling Go is emerging as the language of choice for scraping using a variety of libraries. This book will quickly explain to you how to scrape data data from various websites using Go libraries such as Colly and Gocolly.

Quickly and productively develop complex Spring applications and microservices out of the box, with minimal concern over things like configurations. This revised book will show you how to fully leverage the Spring Boot 2 technology and how to apply it to create enterprise ready applications that just work. It will also cover what's been added to the new Spring Boot 2 release, including Spring Framework 5 features like WebFlux, Security, Actuator and the new way to expose Metrics through Micrometer framework, and more. This book is your authoritative hands-on application productivity while decreasing development time. It's a no nonsense guide with case studies of increasing complexity throughout the book. The author, a senior solutions architect and Principal Technical Instructor with Pivotal, the company behind the Spring Framework, shares his experience, insights and first-hand knowledge about how Spring Boot technology works and best practices. Pro Spring Boot 2 is an essential book for your Spring learning and reference library. What You Will Learn Configure and use Spring Boot Use non-functional requirements Spring Boot Persistence with JDBC, JPA and NoSQL Databases Messaging with JMS, RabbitMQ and WebSockets Test and deploy with Spring Boot A quick look at the Spring Cloud projects Microservices and deployment to the Cloud Extend Spring Boot by creating your own Spring Boot Starter and @Enable feature Who This Book Is For Experienced Spring and Java developers seeking increased productivity gains and decreased complexity and development time in their applications and software services.

Essential Skills for Using and Securing Networks

Implement the power of Go to scrape and crawl data from the web

Building Scalable Network and Non-Network Applications

Build full stack web applications with Go, React, Gin, and GopherJS

Learn how to build powerful RESTful APIs with Golang that scale gracefully

Go Programming Blueprints

86 recipes on how to build fast, scalable, and powerful web services and applications with Go Key Features Become proficient in RESTful web services Build scalable, high-performant web applications in Go Get acquainted with Go frameworks for web development Book Description Go is an open source programming language that is designed to scale and support concurrency at the language level. This gives you the liberty to write large concurrent web applications with ease. From creating web applications to deploying them on Amazon Cloud Services, this book will be your one-stop guide to learn web development in Go. The Go Web Development Cookbook teaches you how to create REST services, write microservices, and deploy Go Docker containers. Whether you are new to programming or a professional developer, this book will help get you up to speed with web development in Go. We will focus on writing applications in Go, using various apps in near real-time with services such as AWS Cloudwatch and X-ray. This book will also teach you how to secure the access with AWS Cognito. By the end of this book, you will have mastered designing, building, and deploying a Go serverless application. What you will learn Understand how AWS Lambda works and use it to create an application Understand how to scale serverless applications Design a cost-effective serverless application in AWS Build a highly scalable and fault-tolerant CI/CD pipeline Understand how to troubleshoot and monitor serverless apps in AWS Discover the working of APIs and single page applications Build a production-ready serverless application in Go Who this book is for This book is for Go developers who would like to learn about serverless architecture. Go programming knowledge is assumed. DevOps and Solution Architects who are interested in building serverless applications in Go can also choose this book.

Summary Get Programming with Go introduces you to the powerful Go language without confusing jargon or high-level theory. By working through 32 quick-fire lessons, you'll quickly pick up the basics of the innovative Go programming language! Purchase of the print book includes a free eBook in PDF, Kindle, and ePub formats from Manning Publications. About the Technology Go is a small programming language designed by Google to tackle big problems. Large projects mean large teams with people of varying levels of experience. Go offers a small, yet capable, language that can be understood and used by anyone, no matter their experience. About the Book Hobbyists, newcomers, and professionals alike can benefit from a fast, modern language; all you need is the right resource! Get Programming with Go provides a hands-on introduction to Go language fundamentals, serving as a solid foundation for your future programming projects. You'll master Go syntax, work with types and functions, and explore bigger ideas like state and concurrency, with plenty of exercises to lock in what you learn. What's inside Language concepts like slices, interfaces, pointers, and concurrency Seven capstone projects featuring spacefaring gophers, Mars rovers, ciphers, and simulations All examples run in the Go Playground - no installation required! About the Reader This book is for anyone familiar with computer programming, as well as anyone with the desire to learn. About the Author Nathan Youngman organizes the Edmonton Go meetup and is a mentor with Canada Learning Code. Roger Peppé contributes to Go and runs the Newcastle upon Tyne Go meetup. Table of Contents Unit 0 - GETTING STARTED Get ready, get set, Go Unit 1 - IMPERATIVE PROGRAMMING A glorified calculator Loops and branches Variable scope Capstone: Ticket

to Mars Unit 2 - TYPES Real numbers Whole numbers Big numbers Multilingual text Converting between types Capstone: The Vigenere cipher Unit 3 - BUILDING BLOCKS Functions Methods First-class functions Capstone: Temperature tables Unit 4 - COLLECTIONS Arrayed in splendor Slices: Windows into arrays A bigger slice The ever-versatile map Capstone: A slice of life Unit 5 - STATE AND BEHAVIOR A little structure Go's got no class Composition and forwarding Interfaces Capstone: Martian animal sanctuary Unit 6 - DOWN THE GOPHER HOLE A few pointers Much ado about nil To err is human Capstone: Sudoku rules Unit 7 - CONCURRENT PROGRAMMING Goroutines and concurrency Concurrent state Capstone: Life on Mars

Learn to build, secure, deploy, and manage your serverless application in Golang with AWS Lambda Key Features Implement AWS Lambda to build scalable and cost-efficient applications in Go Design and set the data flow between cloud services and custom business logic Learn to design Lambda functions using real-world examples and implementation scenarios Book Description Serverless architecture is popular in the tech community due to AWS Lambda. Go is simple to learn, straightforward to work with, and easy to read for other developers; and now it's been heralded as a supported language for AWS Lambda. This book is your optimal guide to designing a Go serverless application and deploying it to Lambda. This book starts with a quick introduction to the world of serverless architecture and its benefits, and then delves into AWS Lambda using practical examples. You'll then learn how to design and build a production-ready application in Go using AWS serverless services with zero upfront infrastructure investment. The book will help you learn how to scale up serverless applications and handle distributed serverless systems in production. You will also learn how to log and test your application. Along the way, you'll also discover how to set up a CI/CD pipeline to automate the deployment process of your Lambda functions. However, you'll learn how to troubleshoot and debug your apps in near real-time with services such as AWS Cloudwatch and X-ray. This book will also teach you how to secure the access with AWS Cognito. By the end of this book, you will have mastered designing, building, and deploying a Go serverless application. What you will learn Understand how AWS Lambda works and use it to create an application Understand how to scale serverless applications Design a cost-effective serverless application in AWS Build a highly scalable and fault-tolerant CI/CD pipeline Understand how to troubleshoot and monitor serverless apps in AWS Discover the working of APIs and single page applications Build a production-ready serverless application in Go Who this book is for This book is for Go developers who would like to learn about serverless architecture. Go programming knowledge is assumed. DevOps and Solution Architects who are interested in building serverless applications in Go can also choose this book.

Summary Go in Practice guides you through 70 real-world techniques in key areas like package management, microservice communication, and more. Following a cookbook-style Problem/Solution/Discussion format, this practical handbook builds on the foundational concepts of the Go language and introduces specific strategies you can use in your day-to-day applications. Purchase of the print book includes a free eBook in PDF, Kindle, and ePub formats from Manning Publications. About the Technology Go may be the perfect systems language. Built with simplicity, concurrency, and modern applications in mind, Go provides the core tool set for rapidly building web, cloud, and systems applications. If you know a language like Java or C#, it's easy to get started with Go; the trick is finding the practical dirt-under-the-fingernails techniques that you need to build production-ready code. About the Book In this practical guide, you'll discover dozens of real-world techniques in key areas. Following a cookbook-style Problem/Solution/Discussion format, this practical handbook builds on the foundational concepts of the Go language and introduces specific strategies you can use in your day-to-day applications. You'll learn techniques for building web services, using Go in the cloud, testing and debugging, routing, network applications, and much more. After finishing this book, you'll be ready to build sophisticated cloud-native Go applications. What's Inside Dozens of specific, practical Golang techniques Using Go for devops and Cloudops Writing RESTful web services and microservices Practical web dev techniques About the Reader Written for experienced developers who have already started exploring Go and want to use it effectively in a production setting. About the Authors Matt Farina is a software architect at Deis. Matt Butcher is a Principal Engineer in the Advanced Technology Group at Hewlett Packard Enterprise. They are both authors, speakers, and regular open source contributors. Table of Contents PART 1 - BACKGROUND AND FUNDAMENTALS Getting into Go A solid foundation Concurrency in Go PART 2 - WELL-ROUNDED APPLICATIONS Handling errors and panic Debugging and testing PART 3 - AN INTERFACE FOR YOUR APPLICATIONS HTML and email template

patterns Serving and receiving assets and forms Working with web services PART 4 - TAKING YOUR APPLICATIONS TO THE CLOUD Using the cloud Communication in cloud services Reflection and code generation Building Scalable Web Apps and RESTful Services

Building Web Applications and Microservices for the Cloud With Go and AngularJS

Clean Architecture

OpenShift for Developers

Develop microservice-based high performance web apps for the cloud with Go

Build SaaS Apps in Go

Today, companies and their developers need to respond to their market at breakneck speeds. Organizations that aren't built on highly-available, rapidly-evolving software are going the way of the dinosaurs. Cloud Native Go brings together all the knowledge developers need to build huge-scale cloud applications that meet the insatiable demands of today's customers and markets. Kevin Hoffman starts with a primer on Go, a programming language that's rapidly gaining traction due to its exceptional suitability for cloud development. Next, he describes the modern cloud-native application in detail, illuminating the 12 Factors associated with successful cloud-native development. Hoffman then guides you through building the skills you need to create microservices in Go, helping you master key techniques such as TDD in Go. Once you're comfortable building microservices, Hoffman shows how to add front-end web services to your cloud-based, cloud-native techniques for routing, RESTful service creation, JSON serialization, securing RESTful services, OAuth2 authentication, and more. You'll find practical techniques for working with web sockets, developing responsive/mobile-friendly UIs, making the most of concurrency, and integrating database access. At each appropriate stopping point, Hoffman shows you how to push your work in progress to cloud like Cloud Foundry/Pivotal Web Services, watch it run there, and assess its ability to dynamically scale, and to support failover, fault tolerance, and monitoring. With cloud-native development rapidly accelerating in importance, these are skills you need now - and no other book brings them together like this. By the time you're finished, you'll be ready to build cloud-native apps that dynamically scale to handle virtually any volume of data, traffic, or users.

Dive into key topics in network architecture and Go, such as data serialization, application level protocols, character sets and encodings. This book covers network architecture and gives an overview of the Go language as a primer, covering the latest Go release. Beyond the fundamentals, Network Programming with Go covers key networking and security issues such as HTTP and HTTPS, templates, remote procedure call (RPC), web sockets including HTML5 web sockets, and more. Additionally, author Jan Newmarch guides you in building and connecting to a complete web server based on Go. This book can serve as both as an essential learning guide and reference on Go networking. What You Will Learn Master network programming with Go Carry out data serialization Use application-level protocols Manage character sets and encodings Deal with HTTP(S) Build a complete Go-based web server With RPC, web sockets, and more Who This Book Is For Experienced Go programmers and other programmers with some experience with the Go language.

This book is a short, concise introduction to computer programming using the language Go. Designed by Google, Go is a general purpose programming language with modern features, clean syntax and a robust well-documented common library, making it an ideal language to learn as your first programming language. This book is a short, concise introduction to computer programming using the language Go. Designed by Google, Go is a general purpose programming language with modern features, clean syntax and a robust well-documented common library, making it an ideal language to learn as your first programming language.

The Go Programming Language is the authoritative resource for any programmer who wants to learn Go. It shows how to write clear and idiomatic Go to solve real-world problems. The book does not assume prior knowledge of Go nor experience with any specific language, so you'll find it accessible whether you're most comfortable with JavaScript, Ruby, Python, Java, or C++. The first chapter is a tutorial on the basic concepts of Go, introduced through programs for file I/O and text processing, simple graphics, and web clients and servers. Early chapters cover the structural elements of Go programs: syntax, control flow, data types, and the organization of a program into packages, files, and functions. The examples illustrate many packages from the standard library and show how to create new ones of your own. Later chapters explain the package mechanism in more detail, and how to build, test, and maintain projects using the go tool. The chapters on methods and interfaces introduce Go's unconventional approach to object-oriented programming, in which methods can be declared on any type and interfaces are implicitly satisfied. They explain the key principles of encapsulation, composition, and substitution using realistic examples. Two chapters on concurrency present in-depth approaches to this increasingly important topic. The first, which covers the basic mechanisms of goroutines and channels, illustrates the style known as communicating sequential processes for which Go is renowned. The second covers more traditional aspects of concurrency with shared variables. These chapters provide a solid foundation for programmers encountering concurrency for the first time. The final two chapters explore lower-level features of Go. One covers the art of metaprogramming using reflection. The other shows how to use the unsafe package to step outside the type system for special situations, and how to use the cgo tool to create Go bindings for C libraries. The book features hundreds of interesting and practical examples of well-written Go code that cover the whole language, its most important packages, and a wide range of applications. Each chapter has exercises to test your understanding and explore extensions and alternatives. Source code is freely available for download from <http://gopl.io/> and may be conveniently fetched, built, and installed using the go get command.

MasteringConcurrencyand

Go Programming For Hackers and Pentesters

Pro Spring Boot 2  
An Idiomatic Approach to Real-World Go Programming  
Learning Go Programming  
Performance, Concurrency, Scalability

**Building upon the success of best-sellers The Clean Coder and Clean Code, legendary software craftsman Robert C. "Uncle Bob" Martin shows how to bring greater professionalism and discipline to application architecture and design. As with his other books, Martin's Clean Architecture doesn't merely present multiple choices and options, and say "use your best judgment": it tells you what choices to make, and why those choices are critical to your success. Martin offers direct, is essential reading for every software architect, systems analyst, system designer, and software manager-- and for any programmer who aspires to these roles or is impacted by their work.**  
**Discover Golang's GUI libraries such as Go-GTK (GIMP Toolkit) and Go-Qt and build beautiful, performant, and responsive graphical applications Key Features Conceptualize and build state-of-art GUI applications with Golang (Go) Tackle the complexity of varying GUI application sizes with a structured and scalable approach Get hands-on experience of GUI development with Shiny, and labs/ui, Fyne, and Walk Book Description Go is often compared to C++ when it comes to low-level programming and implementations that require faster processing, such as Graphical User Interfaces (GUIs). In fact, many claim that Go is superior to C++ in terms of its concurrency and ease of use. Most graphical application toolkits, though, are still written using C or C++, and so they don't enjoy the benefits of using a modern programming language such as Go. This guide to programming GUIs with Go 1.11 explores the various toolkits available, including UI, Walk, Shiny, and Fyne. The book compares the vision behind each project to help you pick the right approach for your project. Each framework is described in detail, outlining how you can build performant applications that users will love. To aid you further in creating applications using these emerging technologies, you'll be able to easily refer to code samples and screenshots featured in the book. In addition to toolkit-specific discussions, you'll cover more complex topics, such as how to structure growing graphical applications, and how cross-platform applications can integrate with each desktop operating system to create a seamless user experience. By delving into techniques and best practices for organizing and scaling Go-based graphical applications, you'll also glimpse Go's impressive concurrency system. In the concluding chapters, you'll discover how to distribute to the main desktop marketplaces and distribution channels. By the end of this book, you'll be a confident GUI developer who can use the Go language to boost the performance of your applications. What you will learn Understand the benefits and complexities of building native graphical applications Gain insights into how Go makes cross-platform graphical application development simple Build platform-native GUI applications using andlabs/ui Develop graphical Windows applications using Walk Create multiplatform GUI applications using Shiny, Nuklear, and Fyne Use Go wrappers for GTK and Qt for GUI application development Streamline your requirements to pick the correct toolkit strategy Who this book is for This book is designed for Go developers who are interested in building native graphical applications for desktop computers and beyond. Some knowledge of building applications using Go is useful, but not essential. Experience in developing GUIs is not required as the book explores the benefits and challenges they pose. This book will also be beneficial for GUI application developers who are interested in trying Go.**

**Go programming has been rapidly adopted by developers for building web applications. With its ecosystem growing in size and its stable architecture, Go offers a library for building scalable and high-performant web services and apps. Hands-On Full Stack Development with Go is a comprehensive guide that covers all aspects of full-stack ....**  
**\*\*\* Make sure to send me a photo of the book via my email shared in the introduction. I'll enroll you to the digital product where you'll have access to additional materials like videos and the source code.If you've never thought of using Go for a web API let me teach you. It's easy, quick and it's fun! Together, we'll build a strong, API-first, reusable code base suitable for building a SaaS or web application. By the end of the book you'll have a solid framework to use as the starting point for future projects. I've built two successful SaaS applications in the last four years using these techniques. They are LeadFuze and Roadmap and I use the same stack, techniques and process that I teach in this book.Go is a fantastic language, you'll be productive in less than one week. We'll dive deep in to the excellent HTTP package and you'll learn useful knowledge that can be used with any other language.Pre-requisites:The book assumes you already have Go setup and that you've followed some getting started tutorials and written at least one function by yourself in Go.Basics knowledge of the HTTP requests/responses life-cycle would be helpful. Knowing what a REST API is, HTTP methods, JSON format. The book would be best if you've already built a web application in another language.**

Cloud Native Programming with Golang

Build Fast and Maintainable Tools

Learn How to Build an API-First SaaS / Web Application

Echo Quick Start Guide

Go Web Programming

How To Code in Go

**Summary Go in Action** introduces the Go language, guiding you from inquisitive developer to Go guru. The book begins by introducing the unique features and concepts of Go. Then, you'll get hands-on experience writing real-world applications including websites and network servers, as well as techniques to manipulate and convert data at speeds that will make your friends jealous. Purchase of the print book includes a free eBook in PDF, Kindle, and ePub formats from Manning Publications. About the Technology Application development can be tricky enough even when you aren't dealing with complex systems programming problems like web-scale concurrency and real-time performance. While it's possible to solve these common issues with additional tools and frameworks, Go handles them right out of the box, making for a more natural and productive coding experience. Developed at Google, Go powers nimble startups as well as big enterprises—companies that rely on high-performing services in their infrastructure. About the Book *Go in Action* is for any intermediate-level developer who has experience with other programming languages and wants a jump-start in learning Go or a more thorough understanding of the language and its internals. This book provides an intensive, comprehensive, and idiomatic view of Go. It focuses on the specification and implementation of the language, including topics like language syntax, Go's type system, concurrency, channels, and testing. What's Inside Language specification and implementation Go's type system Internals of Go's data structures Testing and benchmarking About the Reader This book assumes you're a working developer proficient with another language like Java, Ruby, Python, C#, or C++. About the Authors William Kennedy is a seasoned software developer and author of the blog *GoingGo.Net*. Brian Ketelsen and Erik St. Martin are the organizers of *GopherCon* and coauthors of the Go-based *SkyNet* framework. Table of Contents Introducing Go Go quick-start Packaging and tooling Arrays, slices, and maps Go's type system Concurrency Concurrency patterns Standard library Testing and benchmarking

**Ready to build cloud native applications? Get a hands-on introduction to daily life as a developer crafting code on OpenShift, the open source container application platform from Red Hat. Creating and packaging your apps for deployment on modern distributed systems can be daunting. Too often, adding infrastructure value can complicate development. With this practical guide, you'll learn how to build, deploy, and manage a multitiered application on OpenShift. Authors Joshua Wood and Brian Tannous, principal developer advocates at Red Hat, demonstrate how OpenShift speeds application development. With the Kubernetes container orchestrator at its core, OpenShift simplifies and automates the way you build, ship, and run code. You'll learn how to use OpenShift and the Quarkus Java framework to develop and deploy apps using proven enterprise technologies and practices that you can apply to code in any language. Learn the development cycles for building and deploying on OpenShift, and the tools that drive them Use OpenShift to build, deploy, and manage the ongoing lifecycle of an n-tier application Create a continuous integration and deployment pipeline to build and deploy application source code on OpenShift Automate scaling decisions with metrics and trigger lifecycle events with webhooks**

**What will you learn from this book? Go makes it easy to build software that's simple, reliable, and efficient. Andthis book makes it easy for programmers like you to get started. Googledesigned Go for high-performance networking and multiprocessing, but—like Python and JavaScript—the language is easy to read and use. With thispractical hands-on guide, you'll learn how to write Go code using clearexamples that demonstrate the language in action. Best of all, you'll understandthe conventions and techniques that employers want entry-level Godevelopers to know. Why does this book look so different? Based on the latest research in cognitive science and learning theory, HeadFirst Go uses a visually rich format to engage your mind rather than a textheavyapproach that puts you to sleep. Why waste your time struggling withnew concepts? This multisensory learning experience is designed for theway your brain really works.**

**Summary "Go Web Programming" teaches you how to build scalable, high-performance web applications in Go using modern design principles. Purchase of the print book includes a free eBook in PDF, Kindle, and ePub formats from Manning Publications. About the Technology The Go language handles the demands of scalable, high-performance web applications by providing clean and fast compiled code, garbage collection, a simple concurrency model, and a fantastic standard library. It's perfect for writing microservices or building scalable, maintainable systems. About the Book "Go Web Programming" teaches you how to build web applications in Go using modern design principles. You'll learn how to implement the dependency injection design pattern for writing test doubles, use concurrency in web applications, and create and consume JSON and XML in web services. Along the way, you'll discover how to minimize your dependence on external frameworks, and you'll pick up valuable productivity techniques for testing and deploying your applications. What's Inside BasicsTesting and benchmarkingUsing concurrencyDeploying to standalone servers, PaaS, and DockerDozens of tips, tricks, and techniques About the Reader This book assumes you're familiar with Go language basics and the general concepts of web development. About the Author Sau Sheong Chang is Managing Director of Digital Technology at Singapore Power and an active contributor to the Ruby and Go communities. Table of Contents PART 1 GO AND WEB APPLICATIONSGo and web applications Go ChatChat PART 2 BASIC WEB APPLICATIONSHandling requests Processing requests Displaying content Storing data PART 3 BEING REALGo web services Testing your application Leveraging Go concurrency Deploying Go**

A Craftsman's Guide to Software Structure and Design

Building RESTful Web services with Go

Build Lightweight and high-performance web apps with Echo

Build real-world, production-ready applications with AWS Lambda

Cloud Native Go

Build real-world, production-ready solutions by harnessing the powerful features of Go About This Book An easy-to-follow guide that provides everything a developer needs to know to build end-to-end web applications in Go Write interesting and clever, but simple code, and learn skills and techniques that are directly transferable to your own projects A practical approach to utilize application scaffolding to design highly scalable programs that are deeply rooted in go routines and channels Who This Book Is For This book is intended for developers who are new to Go, but have previous experience of building web applications and APIs. What You Will Learn Build a fully featured REST API to enable client-side single page apps Utilize TLS to build reliable and secure sites Learn to apply the nuances of the Go language to implement a wide range of start-up quality projects Create websites and data services capable of massive scale using Go's net/http package, exploring RESTful patterns as well as low-latency WebSocket APIs Interact with a variety of remote web services to consume capabilities ranging from authentication and authorization to a fully functioning thesaurus Explore the core syntaxes and language features that enable concurrency in Go Understand when and where to use concurrency to keep data consistent and applications non-blocking, responsive, and reliable Utilize advanced concurrency patterns and best practices to stay low-level without compromising the simplicity of Go itself In Detail Go is an open source programming language that makes it easy to build simple, reliable, and efficient software. It is a statically typed language with syntax loosely derived from that of C, adding garbage collection, type safety, some dynamic-typing capabilities, additional built-in types such as variable-length arrays and key-value maps, and a large standard library. This course starts with a walkthrough of the topics most critical to anyone building a new web application. Whether it's keeping your application secure, connecting to your database, enabling token-based authentication, or utilizing logic-less templates, this course has you covered. Scale, performance, and high availability lie at the heart of the projects, and the lessons learned throughout this course will arm you with everything you need to build world-class solutions. It will also take you through the history of concurrency, how Go utilizes it, how Go differs from other languages, and the features and structures of Go's concurrency core. It will make you feel comfortable designing a safe, data-consistent, and high-performance concurrent application in Go. This course is an invaluable resource to help you understand Go's powerful features to build simple, reliable, secure, and efficient web applications. Style and approach This course is a step-by-step guide, which starts off with the basics of go programming to build web applications and will gradually move on to cover intermediate and advanced topics. You will be going through this smooth transition by building interesting projects along with the authors, discussing significant options, and decisions at each stage, while keeping the programs lean, uncluttered, and as simple as possible.

Being a beginner's guide this book has a very simple and clear approach. It is a practical guide that will help you learn the features of Django and help you build a dynamic website using those features. This book is for web developers who want to see how to build a complete site with Web 2.0 features, using the power of a proven and popular development system, but do not necessarily want to learn how a complete framework functions in order to do this. Basic knowledge of Python development is required for this book, but no knowledge of Django is expected.

Build real-world, production-ready solutions in Go using cutting-edge technology and techniques About This Book Get up to date with Go and write code capable of delivering massive world-class scale performance and availability Learn to apply the nuances of the Go language, and get to know the open source community that surrounds it to implement a wide range of start-up quality projects Write interesting and clever but simple code, and learn skills and techniques that are directly transferrable to your own projects Who This Book Is For If you are familiar with Go and are want to put your knowledge to work, then this is the book for you. Go programming knowledge is a must. What You Will Learn Build quirky and fun projects from scratch while exploring patterns, practices, and techniques, as well as a range of different technologies Create websites and data services capable of massive scale using Go's net/http package, exploring RESTful patterns as well as low-latency WebSocket APIs Interact with a variety of remote web services to consume capabilities ranging from authentication and authorization to a fully functioning thesaurus Develop high-quality command-line tools that utilize the powerful shell capabilities and perform well using Go's in-built concurrency mechanisms Build microservices for larger organizations using the Go Kit library Implement a modern document database as well as high-throughput messaging queue technology to put together an architecture that is truly ready to scale Write concurrent programs and gracefully manage the execution of them and communication by smartly using channels Get a feel for app deployment using Docker and Google App Engine In Detail Go is the language of the Internet age, and the latest version of Go comes with major architectural changes. Implementation of the language, runtime, and libraries has changed significantly. The compiler and runtime are now written entirely in Go. The garbage collector is now concurrent and provides dramatically lower pause times by running in parallel with other Go routines when possible. This book will show you how to leverage all the latest features and much more. This book shows you how to build powerful systems and drops you into real-world situations. You will learn to develop high-quality command-line tools that utilize the powerful shell capabilities and perform well using Go's in-built concurrency mechanisms. Scale, performance, and high availability lie at the heart of our projects, and the lessons learned throughout this book will arm you with everything you need to build world-class solutions. You will get a feel for app deployment using Docker and Google App Engine. Each project could form the basis of a start-up, which means they are directly applicable to modern software markets. Style and approach This book provides fun projects that involve building applications from scratch. These projects will teach you to build chat applications, a distributed system, and a recommendation system.

Go in Action

Hands-On Full Stack Development with Go

Black Hat Go

Practical Go

Get Programming with Go

Go: Building Web Applications