Build Web Application With Golang Gitbook

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 application 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 modular code in Go; in-depth informative examples build the base, one step at a time. You will learn how to create a server, work with static files, SQL, NoSQL databases, and Beego. You will also learn how to create and secure REST services, and create and deploy Go web application and Go Docker containers on Amazon Cloud Services. By the end of the book, you will be able to apply the skills you've gained in Go to create and explore web applications in any domain. What you will learn Create a simple HTTP and TCP web server and understand how it works Explore record in a MySQL and MongoDB database Write and consume RESTful web service in Go Invent microservices in Go using Micro - a microservice toolkit Create and Deploy the Beego application with Nginx Deploy Go web application and Docker containers on an AWS EC2

instance Who this book is for This book is for Go developers interested in learning how to use Go to build powerful web applications. A background in web development is expected. 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 both beginners and developers as it extensively covers topics ranging from fundamentals to advanced 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 Your Hands-On Guide to Go, the Revolutionary New Language Designed for Concurrency, Multicore Hardware, and Programmer Convenience Today's most exciting new programming language, Go, is designed from the ground up to help you easily leverage all the power of today's multicore hardware. With this guide, pioneering Go programmer Mark Summerfield shows how to write code that takes full advantage of Go's breakthrough features and idioms. Both a tutorial and a language reference, Programming in Go brings together all the knowledge you need to evaluate Go, think in Go, and write high-performance software with Go. Summerfield presents multiple idiom comparisons showing exactly how Go improves upon older languages, calling special attention to Go's key innovations. Along the way, he explains everything from the absolute basics through Go's lock-free channel-based concurrency and its flexible and unusual duck-typing type-safe approach to object-orientation. Throughout, Summerfield's approach is thoroughly practical. Each chapter offers multiple live code examples designed to encourage experimentation and help you quickly develop mastery. Wherever possible, complete programs and packages are presented to provide realistic use cases, as well as exercises. Coverage includes Quickly getting and installing Go, and building and running Go programs Exploring Go's syntax, features, and extensive standard library Programming Boolean values, expressions, and

numeric types Creating, comparing, indexing, slicing, and formatting strings Understanding Go's highly efficient built-in collection types: slices and maps Using Go as a procedural programming language Discovering Go's unusual and flexible approach to object orientation Mastering Go's unique, simple, and natural approach to fine-grained concurrency Reading and writing binary, text, JSON, and XML files Importing and using standard library packages, custom packages, and third-party packages Creating, documenting, unit testing, and benchmarking custom packages

Master key features of Go, including advanced concepts like concurrency and working with JSON, to create and optimize real-world services, network servers, and clients Key Features Third edition of the bestselling guide to advanced Go programming, expanded to cover RESTful servers, the WebSocket protocol, and Go generics Use real-world exercises to build high-performance network servers and powerful command line utilities Packed with practical examples and utilities to apply to your own development work and administrative tasks Clearly explains Go nuances and features to simplify Go development Book Description Go is the language of the future for high-performance systems due to its simplicity and clear principles. Mastering Go shows you how to put Go to work on real production systems. This new edition has been updated to include topics like creating RESTful servers and clients, understanding Go generics, and developing gRPC servers and clients. Mastering Go, Third Edition explores the capabilities of Go in practice. You will become confident with advanced concepts, including concurrency and the operation of the Go Garbage Collector, using Go with Docker, writing powerful command-line utilities, working with JavaScript Object Notation (JSON) data, and interacting with databases. You will also improve your

understanding of Go internals to optimize Go code and use data types and data structures in new and unexpected ways. This Go programming book also covers the nuances and idioms of Go with exercises and resources to fully embed your newly acquired knowledge. Become an expert Go programmer by building Go systems and implementing advanced Go techniques in your projects. What you will learn Use Go in production Write reliable, high-performance concurrent code Manipulate data structures including slices, arrays, maps, and pointers Develop reusable packages with reflection and interfaces Become familiar with generics for effective Go programming Create concurrent RESTful servers, and build gRPC clients and servers Define Go structures for working with JSON data Who this book is for This book is for Go programmers with previous coding experience, who are familiar with the basics of the language and want to become expert Go practitioners.

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 components using AngularJS and server-side templates. He walks through Go-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.
If you think you're well versed in ASP.NET, think again. This exceptional guide gives you a master class in site building with ASP.NET 3.5 and other cutting-edge Microsoft technologies. You learn how to develop rock-solid web portal applications that can withstand millions of hits every day while surviving scalability and security pressures -- not just for mass-consumer homepages, but also for dashboards that deliver powerful content aggregation for enterprises. Written by Omar AL Zabir, co-founder and CTO of Pageflakes, Building a Web 2.0 Portal with ASP.NET 3.5 demonstrates how to develop portals similar to My Yahoo!, iGoogle, and Pageflakes using ASP.NET 3.5, ASP.NET AJAX, Windows Workflow Foundation, LINQ and .NET 3.5. Through the course of the book, AL Zabir builds an open source Ajax-enabled portal prototype (available online at www.dropthings.com), and walks you though the design and architectural challenges, advanced Ajax concepts, performance optimization techniques, and server-side scalability problems involved. You learn how to: Implement a

highly decoupled architecture following the popular n-tier, widget-based application model Provide drag-and-drop functionality, and use ASP.NET 3.5 to build the server-side part of the web layer Use LINQ to build the data access layer, and Windows Workflow Foundation to build the business layer as a collection of workflows Build client-side widgets using JavaScript for faster performance and better caching Get maximum performance out of the ASP.NET AJAX Framework for faster, more dynamic, and scalable sites Build a custom web service call handler to overcome shortcomings in ASP.NET AJAX 1.0 for asynchronous, transactional, cache-friendly web services Overcome JavaScript performance problems, and help the user interface load faster and be more responsive Solve scalability and security problems as your site grows from hundreds to millions of users Deploy and run a high-volume production site while solving software, hardware, hosting, and Internet infrastructure problems Building a Web 2.0 Portal with ASP.NET 3.5 also presents real-world ASP.NET challenges that the author has solved in building educational and enterprise portals, plus thirteen production disasters common to web applications serving millions of users. If you're ready to build state-of-the art, high-volume web applications, this book has exactly what you need.

?? Go ????(???)??????????

Everything web designers need to build sites with Dreamweaver Dreamweaver is the leading website creation tool, with 90 percent of the market share. The nine minibooks that make up this guide cover getting started with Dreamweaver CS5, creating and publishing great sites, making pages dynamic, building web applications, and much more, including all the upgrades in Dreamweaver CS5. Dreamweaver is the gold standard for website development software; this complete reference covers what both beginners

and intermediate-level users need to know to make the most of Dreamweaver CS5 and create professional-quality sites Nine minibooks cover getting started, mastering Dreamweaver basics, working like a pro, energizing your site, publishing your site, working collaboratively, building Web applications, making pages dynamic, and developing applications rapidly Teaches how to customize the workspace; understand the Panels and Properties Inspector; plan, design, and manage your site; work with text and graphics; add Flash, movies, and sound; work with Cascading Style Sheets, JavaScript, and Ajax, connect to a database, and much more Dreamweaver CS5 All-in-One For Dummies gives web designers essential information for creating, managing, and maintaining all types of websites.

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

Even non-techies can build their first web application with this

friendly, approachable guide, which teaches the principles of programming plus two popular practical frameworks: Foundation for Apps and Angular.js. It walks users through every step of the process, from designing something that works on all formats to choosing a programming framework, along with assignments in every chapter. If you know some HTML and CSS, but want to go further, this book is perfect! 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. Key Features The easiest way to learn how to build web apps with Echo Build a full working project For Go developers with only basic web development knowledge required Book Description 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 basics, you'll be introduced to all the concepts for a building real-world web system with Echo. You will start with the 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 test your 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 their 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 How context is managed through the lifetime of the request and response pipeline Decrease

complexity of your apps by developing middleware functions Interact with the request through request data bindings Interact with the response through response data renderings within the framework Use Echo's logging and error handling facilities Render Go 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.

Summary Go from zero to production readiness with Docker in 22 bite-sized lessons! Learn Docker in a Month of Lunches is an accessible task-focused guide to Docker on Linux, Windows, or Mac systems. In it, you'll learn practical Docker skills to help you tackle the challenges of modern IT, from cloud migration and microservices to handling legacy systems. There's no excessive theory or niche-use cases—just a quick-and-easy guide to the essentials of Docker you'll use every day. Purchase of the print book includes a free eBook in PDF, Kindle, and ePub formats from Manning Publications. About the technology The idea behind Docker is simple: package applications in lightweight virtual containers that can be easily installed. The results of this simple idea are huge! Docker makes it possible to manage applications without creating custom infrastructures. Free, open source, and battle-tested, Docker has quickly become must-know technology for developers and administrators. About the book Learn Docker in a Month of Lunches introduces Docker concepts through a series of brief hands-on lessons. Follow-ing a learning path perfected by author Elton Stoneman, you'll run containers by chapter 2 and package applications by chapter 3. Each lesson teaches a practical skill you can practice on Windows, macOS, and Linux systems. By the end of the month you'll know how to containerize and run any kind of application with Docker. What's inside Package applications to run in containers Put containers into

production Build optimized Docker images Run containerized apps at scale About the reader For IT professionals. No previous Docker experience required. About the author Elton Stoneman is a consultant, a former architect at Docker, a Microsoft MVP, and a Pluralsight author. Table of Contents PART 1 - UNDERSTANDING DOCKER CONTAINERS AND IMAGES 1. Before you begin 2. Understanding Docker and running Hello World 3. Building your own Docker images 4. Packaging applications from source code into Docker Images 5. Sharing images with Docker Hub and other registries 6. Using Docker volumes for persistent storage PART 2 - RUNNING DISTRIBUTED APPLICATIONS IN CONTAINERS 7. Running multi-container apps with Docker Compose 8. Supporting reliability with health checks and dependency checks 9. Adding observability with containerized monitoring 10. Running multiple environments with Docker Compose 11. Building and testing applications with Docker and Docker Compose PART 3 - RUNNING AT SCALE WITH A CONTAINER ORCHESTRATOR 12. Understanding orchestration: Docker Swarm and Kubernetes 13. Deploying distributed applications as stacks in Docker Swarm 14. Automating releases with upgrades and rollbacks 15. Configuring Docker for secure remote access and CI/CD 16. Building Docker images that run anywhere: Linux, Windows, Intel, and Arm PART 4 - GETTING YOUR CONTAINERS READY FOR PRODUCTION 17. Optimizing your Docker images for size, speed, and security 18. Application configuration management in containers 19. Writing and managing application logs with Docker 20. Controlling HTTP traffic to containers with a reverse proxy 21. Asynchronous communication with a message queue 22. Never the end A complete guide to build robust and scalable web applications with Spring and Angular. About This Book This hands on guide will teach you how to build an end-to-end

modern web application using Spring and Angular. It is easy to read and will benefit Java developers who have been used to develop the back-end part of web application while front-end (UI) has been left for UI developers. Learn the core aspects involved in developing the backend and the UI, right from designing to integrating and deploying. Who This Book Is For This book is targeted towards Java Web Developers with a basic knowledge of Spring who want to build complete web applications in a fast and effective way. They will want to gain a stronghold on both frontend and backend development to advance in their careers. What You Will Learn Set up development environment for Spring Web App and Angular app. Process web request and response and build REST API endpoints. Create data access components using Spring Web MVC framework and Hibernate Use Junit 5 to test your application Learn the fundamental concepts around building Angular Configure and use Routes and Components. Protect Angular app content from common web vulnerabilities and attacks. Integrate Angular apps with Spring Boot Web API endpoints Deploy the web application based on CI and CD using Jenkins and Docker containers In Detail Spring is the most popular application development framework being adopted by millions of developers around the world to create high performing, easily testable, reusable code. Its lightweight nature and extensibility helps you write robust and highly-scalable server-side web applications. Coupled with the power and efficiency of Angular, creating web applications has never been easier. If you want build end-to-end modern web application using Spring and Angular, then this book is for you. The book directly heads to show you how to create the backend with Spring, showing you how to configure the Spring MVC and handle Web requests. It will take you through the key aspects such as building REST API endpoints, using Hibernate, working with Junit 5 etc. Once

you have secured and tested the backend, we will go ahead and start working on the front end with Angular. You will learn about fundamentals of Angular and Typescript and create an SPA using components, routing etc. Finally, you will see how to integrate both the applications with REST protocol and deploy the application using tools such as Jenkins and Docker. Style and approach This is a straightforward guide that shows how to build a complete web application in Angular and Spring.

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. Moreover, you'll learn how to troubleshoot and monitor 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 scaleup 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.

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 basics of the Gin framework, before progressing to build a real-world RESTful API. Along the way, you'll learn how to write custom middleware and understand the routing mechanism, as well as how to bind user data and validate incoming HTTP requests. The book also demonstrates 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 Deploy 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.

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.

This book provides the reader with a comprehensive overview of the new open source programming language Go (in its first stable and maintained release Go 1) from Google. The language is devised with Java / C#-like syntax so as to feel familiar to the bulk of programmers today, but Go code is much cleaner and simpler to read, thus increasing the productivity of developers. You will see how Go: simplifies programming with slices, maps, structs and interfaces incorporates functional programming makes error-handling easy and secure simplifies concurrent and parallel programming with goroutines

and channels And you will learn how to: make use of
Go's excellent standard library program Go the
idiomatic way using patterns and best practices in
over 225 working examples and 135 exercises This
book focuses on the aspects that the reader needs
to take part in the coming software revolution using
Go.

Google App Engine makes it easy to create a web
application that can serve millions of people as
easily as serving hundreds, with minimal up-front
investment. With Programming Google App Engine,
Google engineer Dan Sanderson provides practical
guidance for designing and developing your
application on Google's vast infrastructure, using
App Engine's scalable services and simple
development model. Through clear and concise
instructions, you'll learn how to get the most out of
App Engine's nearly unlimited computing power.
This second edition is fully updated and expanded to
cover Python 2.7 and Java 6 support, multithreading,
asynchronous service APIs, and the use of
frameworks such as Django 1.3 and webapp2.
Understand how App Engine handles web requests
and executes application code Learn about new
datastore features for queries and indexes,
transactions, and data modeling Create, manipulate,
and serve large data files with the Blobstore Use
task queues to parallelize and distribute computation
across the infrastructure Employ scalable services

for email, instant messaging, and communicating with web services Track resource consumption, and optimize your application for speed and cost effectiveness

In this IBM® Redbooks® publication we demonstrate that it is possible to combine the traditional strengths of the mainframe to manage large volumes of data and run business transactions with the Web 2.0 paradigm. We can get simpler interfaces, better integration among different services, lightweight protocols for communication, and much more, together with the availability, security, and reliability of mainframe data. And we will show how mainframe data can be accessed by smartphones such as Android or iPhone. But we can do more to demonstrate how flexible the mainframe platform is. Through the use of pervasive devices it is possible to add new possibilities to mainframe applications, extending System z® capabilities. We can receive notifications in real time, for example, of successful or unsuccessful termination of a TWS job stream, or we can immediately get alerts about abends that occurred in a critical application. This book is another demonstration that the mainframe is alive and kicking and can and should play a key role in modern application architectures.

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.

There has been an increase in the need for web apps, however, most programming languages which support app development are complex. This means that longer periods of times are spent while developing these apps. Python has a framework called "Django" which provides web developers with a mechanism to develop web apps in an easy and quick manner. So it's a good idea for you to learn how to use this framework for the development of web apps, all of which is explained in this book. Make sure that you install Python 2.6.5 or higher. Enjoy reading!This book will save you time. On many occasions we've seen clever students get stuck, spending hours trying to fight with Django and other aspects of web development. More often than not, the problem was usually because a key piece of information was not provided, or something was not made clear. While the occasional blip might set you back 10-15 minutes, sometimes they can take hours to resolve. We've tried to remove as many of these hurdles as possible. This will mean you can get on with developing your application instead of stumbling along.This book will lower the learning curve. Web application frameworks can save you a lot of hassle and lot of time. Well, that is if you know how to use

them in the first place! Often the learning curve is steep. This book tries to get you going - and going fast by explaining how all the pieces fit together.This book will improve your workflow. Using web application frameworks requires you to pick up and run with a particular design pattern - so you only have to fill in certain pieces in certain places. After working with many students, we heard lots of complaints about using web application frameworks - specifically about how they take control away from them (i.e. inversion of control).To help you, we've created a number of workflows to focus your development process so that you can regain that sense of control and build your web application in a disciplined manner.This book is not designed to be read. Whatever you do, do not read this book! It is a hands-on guide to building web applications in Django. Reading is not doing. To increase the value you gain from this experience, go through and develop the application. When you code up the application, do not just cut and paste the code. Type it in, think about what it does, then read the explanations we have provided to describe what is going on. If you still do not understand, then check out the Django documentation, go to Stack Overflow or other helpful websites and fill in this gap in your knowledge.

Discover practical techniques to build cloud-native apps that are scalable, reliable, and always

available. Key Features Build well-designed and secure microservices. Enrich your microservices with continous 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.

Simplified Chinese edition of Rework. Seth Godin, author of the international bestselling marketing Purple Cow that changed the way marketing is performed, says: "Stop reading the review. Buy the book." This small book is filled with common sense - yes, you know them, but the book tells you how to put them into practice. In Simplified Chinese.

Distributed by Tsai Fong Books, Inc.
Learning about isomorphic development in Go leads to a unified view of web and back-end development: It extends the remit of a server-side programming language to the browser. About This Book Learn how to build Isomorphic Go web applications Neatly organize your isomorphic codebase to enhance the maintainability of your application Leverage the same Go code across the web server and the web browser Who This Book Is For This book addresses web developers, who are comfortable with JavaScript and the full web development cycle. You may or may not be experienced in isomorphic development, but should have a basic understanding of Go's syntax. What You Will Learn Create Go programs inside the web browser using GopherJS Render isomorphic templates on both the client side and the server side Perform end-to-end application routing for greater search engine discoverability and an enhanced user experience Implement isomorphic handoff to seamlessly transition state between the web server and the web browser Build real-time web application functionality with websockets Create reusable components (cogs) that are rendered using the virtual DOM Deploy an Isomorphic Go application for production use In Detail Isomorphic Go is the methodology to create isomorphic web applications using the Go programming language. Isomorphic web applications have the capability to

reuse code across environments, increasing the synergy between the web server and the web browser. This book is a hands-on guide that will show you how to build and deploy an Isomorphic Go web application. Isomorphic Go begins with an in-depth exploration of the benefits provided by the isomorphic web application architecture. You'll be introduced to the Isomorphic Go toolchain, and how it can help you write Go code that functions across environments. You'll learn how to create Go programs in the web browser using GopherJS and how to render isomorphic templates. Then you'll be introduced to end-to-end application routing, use isomorphic handoff to seamlessly transition state from the web server to the web browser, and compose isomorphic forms that have the ability to reuse form validation logic. You'll be exposed to advanced concepts including the implementation of real-time web application functionality with websockets and the creation of reusable components (cogs) that are rendered using the virtual DOM. Finally, you'll see how to deploy an Isomorphic Go web application to a production environment. Style and Approach Kamesh guides beginner Go developers and JavaScript developers with basic Go syntax skills through a full Go tutorial until they emerge with a working knowledge of isomorphic Go development.

If you're a web developer interested in building scalable

single-page applications—full-stack, browser-based apps that connect to a backend—this practical guide shows you how to use Ember.js, the popular JavaScript framework based on the model-view-controller (MVC) architectural pattern. Through the course of the book, you'll learn how to build a prototype Ember application (a musician index called Rock'n'Roll Call), using routers, templates, models, controllers, and views. You'll also understand how Ember's convention over configuration approach helps you persist data, build backend technologies, and create widgets for developing production-capable applications that behave like desktop software. Set up workflow management and boilerplate code creation Learn how Ember's "developer ergonomics" help you use less code Write templates for the book's prototype with Handlebars.js Use routers to manage application states without reloading the page Connect controllers and views with events, and sync data with data-binding Build an Ember backend with a RESTful API or Ruby on Rails Use the Ember-Data library to persist data and talk to the backend Write reusable encapsulated widgets to extend your applications

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 ChitChat 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 The author Kanalakis gives in-depth and detailed guidance on how to build a single, scalable enterprise application with C# and using .NET technologies. Discover practical techniques to build cloud-native apps that are scalable, reliable, and always available. Key Features Build well-designed and secure microservices.

Enrich your microservices with continous 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 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.. The book will take you on a journey into the world of microservices and cloud computing with the help of Go. It will start 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 the reader 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 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 ...

?????Go??????????????????Go?????????????????Go??????????JavaScript?Ruby?Python?Java?C++?????????????
??????Go????????????????????????????????????? ???
??????Go??
?????????????????????????????????????Go??????
??????? ???????????Go?????????????????????????
????????????????????????????????? ??????????
???????????????goroutine?channel??????????Go????
??
?? ???????Go??????????reflection???????????????uns

afe????????????????????cgo????Go?C?????? ?????????
??Go??
?????????http://gopl.io/??????go get???????????? #????
GOTOP Information Inc.

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.
Take your Go programming skills to the next level in this

Advanced Topics in Go video series from Go Programming expert Zanis Khan. Click here to watch Zanis' other videos including the Definitive Guide to Writing Reliable and Efficient Code in Go on O'Reilly . There are 12 topics within this video series: Structures in Go . This first topic in the Go programming video series covers structures (structures.go) in the Go programming language. Follow along with Zanis in this hands-on session and practice creating structures and then accessing data from these structures. Slices in Go . This second topic in the Go programming video series covers Go Slices (slice.go), which are abstractions of the Go Array. Follow along with Zanis in this hands-on session and practice using the len() and cap() functions. Maps in Go . This third topic in the Go programming video series covers maps (map.go) in the Go programming language. Follow along with Zanis and map unique keys to values in this hands-on session. Ranges in Go . This fourth topic in the Go programming video series covers ranges (range.go) in the Go programming language. Follow along with Zanis in this hands-on session, and practice using ranges in FOR loops to iterate over items of an array, slice, channel, or map. Recursion in Go . This fifth topic in the Go programming video series covers recursion in the Go programming language. Follow along with Zanis in this hands-on session and create a function that calls itself. Interfaces in Go . This sixth topic in the Go programming video series covers interfaces in the Go programming language. Follow along with Zanis in this hands-on session and create interfaces, which are sets of method signatures. Internet Access in Go . This

seventh topic in the Go programming video series covers internet access in the Go programming language. Follow along with Zanis in this hands-on session and learn how to access the internet as well as how to parse data sourced from web pages. Building a Simple Web Application in Go . This eighth topic in the Go programming video series shows you how to build a simple Web application in the Go programming language. HTML Templates in Go . This ninth topic in the Go programming video series explains how to use and create HTML templates in the Go programming language. Follow along with Zanis in this hands-on session and create a pre-built website. Go Routines . This tenth topic in the Go programming video series covers Go routines...

By integrating the Web with traditional TV, Google TV offers developers an important new channel for content. But creating apps for Google TV requires learning some new skills—in fact, what you may already know about mobile or desktop web apps isn't entirely applicable. Building Web Apps for Google TV will help you make the transition to Google TV as you learn the tools and techniques necessary to build sophisticated web apps for this platform. This book shows you how Google TV works, how it fits into the web ecosystem, and what the opportunities are for delivering rich content to millions of households. Discover the elements of a compelling TV web app, and what comprises TV-friendly navigation Learn the fundamentals for designing the 10-foot user experience Work with the Google Chrome browser on a TV display, and migrate an existing site Use examples

for developing a TV web app, including the UI, controls, and scrolling Understand how to optimize, deliver, and protect video content for Google TV Help users discover your content by optimizing your site for Search—especially videos

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. This IBM® RedpaperTM publication presents a series of tutorials for cloud native developers just getting started

with IBM CloudTM and IBM Cloud Object Storage. Within the context of a car insurance application, this paper presents an introductory series of linked modules that allow developers unfamiliar with either IBM Cloud or cloud native development to get started with application development using IBM starter kits. This allows you to become familiar with the types of services available on IBM Cloud, and to develop a sense of which patterns and choices are appropriate for different use cases. Some of the technologies and products covered in this book are Cloudant®, WatsonTM Analytics, machine learning, elastic search, Kubernetes, containers, pre-signed URLs, Aspera®, and SQL Query. In addition to the technical integration steps, it also presents a business case for integrating these technologies and products with IBM Cloud Object Storage. The target audience for this paper is cloud native developers and cloud object storage specialists.

Your one-stop 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 projects, 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 helps 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.
???,?
??????????????,??????Java???C#?????.
This is a practical guide to the features and uses of JDO, with example programs and learning tools provided. Explore the necessary concepts of REST API development by building few real world services from scratch. Key Features Follow best practices and explore techniques such as clustering and caching to achieve a reactive, 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 Book Description REST 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 metricize a REST API using an API Gateway. By the end of the book you will be proficient in building RESTful APIs in Go. What you will learn Create HTTP handler and introspect 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 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.

Copyright: 966f8ab4efad48075d23168905abc6ad