Standard Kurs Google Go Concurrent Programming

Dieser Google Go Concurrent Programming Kurs zeigt wie Asynchrone (Concurrent) und parallele Ausführungen mit Google Go (Golang) zu programmieren sind.

Information

Kurscode: GOCODas Angebot Firmenkurs finden Sie hier.Das Angebot Online Kurs finden Sie hier.Infrastruktur: Bring your own Computer (Processor i7 oder vergleichbar, mindestens 8GB RAM), VMWare Workstation Player Version 12+.Alle Kursunterlagen werden elektronisch abgegeben, damit leisten wir einen Beitrag an die Umwelt.

Information

Kurscode: GOCOKursdauer: 1 TagKurspreis Firmenkurs Inhouse: sFr. 2099.- (3-5 Teilnehmer)sFr. 2939.- (6-8 Teilnehmer)sFr. 3499.- (9-12 Teilnehmer)Inhouse Firmenkurse werden vor Ort bei Ihnen durchgeführt. Je nach Ort und Distanz können zusätzliche Kosten für Übernachtung und Anfahrt anfallen.Das Angebot Standard Kurs finden Sie hier.Das Angebot Online Kurs finden Sie hier.Infrastruktur: Bring your own Computer (Processor i7 oder vergleichbar, mindestens 8GB RAM), VMWare Workstation Player Version 12+.Alle Kursunterlagen werden elektronisch abgegeben, damit leisten wir einen Beitrag an die Umwelt.

Information

Kurscode: GOCODas Angebot Firmenkurs finden Sie hier.Das Angebot Standard Kurs finden Sie hier.Infrastruktur: Bring your own Computer (Processor i7 oder vergleichbar, mindestens 8GB RAM), VMWare Workstation Player Version 12+.Alle Kursunterlagen werden elektronisch abgegeben, damit leisten wir einen Beitrag an die Umwelt.

Einleitung

Google Go (Golang) ist die Programmiersprache mit feinen Concurrency Konzepten. Golang bietet die direkte Unterstützung für die parallele Ausführung via goroutines direkt in der Syntax und unterscheidet sich damit von anderen Programmiersprachen wie z.B. Java.goroutines sind nicht Threads und damit wird das Scheduling nicht via das Betriebsystem sondern über die Go Runtime behandelt, welche das Multiplexing über ein definierte Anzahl von Threads oder Prozessoren übernimmt.On Java you can run 1000’s or tens of 1000’s threads. On Go you can run hundreds of thousands or millions of goroutines.Die Synchronisierung der parallelen Routinenen erfolgt via Channels (Signale) und Mutual Exclusion Locks. Mehrere Channels (Signale) sind via Select abfangbar. Diese Konzepte ermöglichen das effiziente Programmieren, aber die Konzepte müssen dabei verstanden werden.Mit dem Data Races Detector findet man in Golang mögliche Ablauffehler, die ansonsten nur sehr schwierig zu finden sind.Auch die effektive parallele Ausführung über mehrere Prozessoren (CPU's) hinweg wird direkt unterstützt, parallel heisst aber nicht concurrent.Dieser Kurs bietet Ihnen eine vertiefte Einführung in die parallele oder concurrent Programmierung mit Google Go. Anhand von Beispielen und Übungen lernen Sie wie man das Golang Concurrent Package richtig anwendet für Ihre Applikationen.

Ihr Nutzen

- Sie können multithreaded (concurrent) Anwendung mit goroutines programmieren.
- Sie können mit Channels umgehen und damit Events (Signale) verarbeiten.
- Sie können Ihre Anwendung auf Fehler analysieren mit dem Race Detector.
- Sie können die parallele von der concurrent Ausführung unterscheiden.
- Sie kennen die Best Practices und Patterns

Verwandte Kurse

- Google Go Web Programming (GOHT)
- Google Go für C# Programmierer (golang for c# programmers) (GOCS)
- Google Go für C++ Programmierer (golang for c++ programmers) (GOCP)

- Google Go für Java Programmierer (golang for java programmers) (GOJA)
- Google Go für PHP Programmierer (golang for php programmers) (GOPH)

Voraussetzungen

Guten Kenntnisse der Programmiersprache Go analog dem Kurs GOGO oder gleichwertige Kenntnisse.

Teilnehmerkreis

Google Go Programmierer und Anwender

Unterlagen

- Tutorial
- Code Walks
- Internet / Intranet

Inhalt

- Einführung
- gorutine vs Threads
- Channels und Select
- Synchronisierung
- Deadlocks
- Data Race Detector
- Parallele Ausführung (Multi CPU Processing
- Benchmark Tests
- Best Practices und Patterns

Kontakt

Simtech AG
Finkenweg 23
3110 Münsingen
Schweiz

Impressum

Das Copyright für sämtliche Inhalte dieser Website liegt bei Simtech AG, Schweiz.
Beachten Sie auch unsere Hinweise zum Urheberrecht, Datenschutz und Haftungsauschluss.
Jeder Hinweis auf Fehler nehmen wir gerne entgegen.

Copyright

2024 Simtech AG, All rights reserved, Powered by stack.ch written in Golang by Daniel Schmutz

https://www.simtech-ag.ch/race