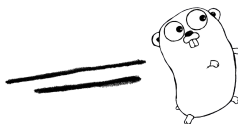


Die Programmiersprache Go - Eine Einführung

Konzeptskizze

Student: Adrian Helberg
Prüfer: Prof. Dr. Axel Schmolitzky

4. März 2018



Geschichte

Überblick

Merkmale

Syntax

Paradigmen

Zentrale Fragestellung

Pro & Contra

Compiler

Einzelnachweise

Geschichte

- ▶ Entwickler
- ▶ Entwurfsphase
- ▶ Veröffentlichung
- ▶ Go-Community

Überblick

- ▶ Motivation
- ▶ Zielsetzung
- ▶ Skalierbarkeit
- ▶ Einsatzgebiete
- ▶ Erster Vergleich mit JavaTM

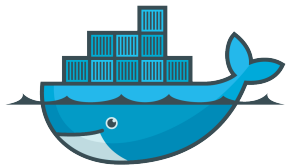


Merkmale

- ▶ Closures
- ▶ Reflexion
- ▶ Typsicherheit
- ▶ Automatische Speicherbereinigung
- ▶ Interfaces, Mixins → Objektorientierung
- ▶ Keine Klassen → Java Vergleich
- ▶ Pakete

Projekte

Go



docker

Java



elasticsearch

Syntax

Go

```
package main

import "fmt"

func main() {
    fmt.Println("Hello World!")
}
```

Java

```
public class HelloWorld
{
    public static void main (String[] args)
    {
        System.out.println("Hello World!");
    }
}
```

Paradigmen

- ▶ Nebenläufig
- ▶ Imperativ
- ▶ Strukturiert
- ▶ Modular
- ▶ Objektorientiert

Zentrale Fragestellung

„flexibel wie dynamisch getyped,
aber mit statischer Typsicherheit?“

Pro & Contra

Pro

- ▶ Minimalismus
- ▶ Statisches Duck-Typing
- ▶ Parallelisierung
- ▶ Aufgeräumte Syntax
- ▶ Schneller Compiler

Contra

- ▶ Keine generische Programmierung
- ▶ nil statt Option
- ▶ Wenig grundlegende Datenstrukturen
- ▶ Keine Methodenüberladung

Pro & Contra

Pro

- ▶ Statisch gelinkte Binärdateien
- ▶ Laufzeiteigenschaften
- ▶ Integriertes Unit-Test-Framework
- ▶ Paketmanager

Contra

- ▶ Unbefriedigende API-Dokumentation
- ▶ Teilweise umständliche APIs
- ▶ Umständliches Mocking
- ▶ Kleines Ökosystem

Compiler

- ▶ Gc
- ▶ Gccgo

Einzelnachweise

► ...