



Proseminar: Moderne Programmiersprachen

Themenvorstellung - Oleg Travkin

Was könnte wichtig sein?



- Programmierparadigmen
 - imperativ, funktional, logisch
 - objektorientiert
- Typisierungskonzepte
 - statisch, dynamisch ...und viele Varianten dazwischen
 - Typinferenz
 - Vererbung, Polymorphie
- Syntax
 - Kürze, Einrückung, Nähe zu natürlicher Sprache
- andere Besonderheiten...z.B. Nebenläufigkeit

- Vorlesungsmaterial (siehe Webseite von Prof. Kastens)
 - Grundlagen der Programmiersprachen
 - Objektorientierte Programmierung
 - Funktionale Programmierung
 - Skriptsprachen
 - Parallel Programming

- Clojure
 - „alles ist ein Ausdruck“, kurzer Code
 - funktional, dynamisch
- Python
 - gute Lesbarkeit, Einrückung mit Semantik
 - imperativ, OO, (funktional), dynamisch
- Go
 - besonders performant, verteilte/nebenläufige Appl.
 - imperativ, OO, funktional, statisch
- Scala
 - Java für Anhänger der funktionalen Programmierung
 - funktional, OO, (imperativ), statisch

- TypeScript
 - JavaScript mit statischer Typisierung, für größere Appl.
 - imperativ, funktional, OO, dynamisch
- Rust
 - Hardware-nah, Konfliktvermeidung durch Sprache/Compiler
 - imperativ, funktional, OO, statisch
- Erlang
 - Telekommunikation, Robustheit steht im Vordergrund
 - imperativ, funktional, OO, dynamisch
- Swift
 - Nachfolger von Objective-C, Apple und Linux Appl.
 - imperativ, funktional, OO, dynamisch

- Lua
 - minimalistisch, in Hard- oder Software eingebettete Appl.
 - imperativ, funktional, OO, dynamisch
- Ruby
 - Web-Anwendungen, Prototyping
 - imperativ, (funktional), OO, dynamisch
- R
 - statistische Berechnungen, Datenvisualisierung
 - funktional, OO, dynamisch
- Julia
 - aufwändige mathematische Berechnungen
 - imperativ, funktional, OO, dynamisch