

Warum funktionale Programmierung (What's the Hype)?

Teaser: Was kann man mit funktionaler Programmierung machen?

Pandoc

DSLs Interpreter ...

Crush

GenNativeSql

Was ist funktionale Programmierung?

Im engeren Sinn: Alle Programmiermodelle denen das Lambdakalkül zugrunde liegt

Im weiteren Sinn: Alle Programmiermodelle die an das Lambdakalkül angelehnt sind

Definition Lambdakalkül

Frage: wie kommt man darauf - so eine Definition fällt ja nicht vom Himmel

Prelude: Berechenbarkeit

Prelude: Abstraktion / Essenz

Prelude: Die Essenz berechenbarer Funktionen

Definition Lambdakalkül (weitere Erläuterungen)

Definition Lambdakalkül mit interpretierten Funktionen (INT, CHAR)

Definition Monoid

Eigenschaften des Lambdakalklüs

Buzzword Stack: Lambdas - anonyme Funktionen Currying / Higher Order Functions

Immutable Nebeneffektfreiheit / Referenzielle Transparenz (https://de.wikipedia.org/wiki/Referenzielle_Transparenz)

Interlude: Sinn und Bedeutung

Motivation Typensysteme: Vermeidung un- definierter Programmzustände (Run-Time)

Operatorenüberladung INT und CHAR und “Addition” -> Einstellige Verknüpfung

Tracing Addition(+) Char

Vermeidung solcher Kategorienfehler zur Laufzeit möglich?

Einführung einfach typisiertes Lambdakalkül (in Haskell)

Definition mconcat_char Definition mconcat_string

Ad-Hoc Polymorphie (Overloading)

Typeclasses: Typen die eine bestimmte Operation unterstützen

Parametrische Polymorphie (

Identity-Funktion

Notes

Zielgruppe

Keine Erfahrung notwendig Schlaraffenland Griesbrei ... anstrengender Teil bis man merkt das es sich lohnt

Vokabular

Größte Klarheit durch Benennung abstrakter Konzepte; Methaphern tragen nur soweit - nichts in der Realität erfasst solch einen Begriff in seiner Reduziertheit / Abtraktheit

Oberfläche und Volumen eines Problems

weiterführende Programmierung

General Purpose FP * haskell * Scala * F Sharp

Web * purescript / elm

Dependent Types / Beweistheorie * idris * Coq

Links

- Vortrag
- Haskell
- Stack
- Crush