Memo Datei für Clojure Usergroup Demo

Übersicht (Clojure für die CLR)

* Sieben Sprachen in Sieben Wochen
* Buch mitnehemen

Übersicht (Erfinder)

* Rich Hickey +25 Jahre Berufserfahrung
* Erfinder von Clojure in 2007
* Hat in einen breiten Spektrum von Projekten gearbeitet
* Erfinder von Datomic einer auf Clojure basierenden Datenbank
* John McCarthy
* Erfinder von LISP in Jahre 1958
* Zweitälteste Programmiersprache nach Fortran

Übersicht (was ist Clojure)

* Lisp
  + Code ist Daten Daten ist Code
* Functional
  + Die Sprache arbeitet mit Funktionen, es gibt keine Objekte  
    (Einschub Funktional gegen Objektorientiert, keine Kriege führen
* Immutable Datastructures
  + Es werden keine Datenstrukturen geändert, sondern immer neue Datenstrukturen zurückgeliefert.
* Concurrency
  + Clojure hat wundarbare Concurreny Eigenschaften
  + Es gibt dafür eine STM (Software Transaktional Memory)
  + Es gibt verschieden Datentypen wie Atome, Refs und Agents
* REPL Based
  + Ein Hauptwerkzeug ist die REPL (Read Eval Print Loop)
* Interopabilittät
  + Clojure arbeitet mit verschiedenen Host System zusammen.
  + Investions Schutzt für bestehende Entwicklung
  + JAVA
  + CLR
  + Javascript

Funktion 1 (von c# nach Clojure)

* Lernen Clojure Funktionen zu lesen
* Clojure ist ein List, wo immer alles innerhalb eines Klammer Paares notiert wird

Funktion 1 (die REPL)

* An der Repl werden die Anweisungen eingelesen (Read)
* Die Anweisungen werden evaluiert (Eval)
* Die Ergebnisse werden ausgegeben. (Print)
* Das ganze wiederholt sich in einen Zyklus (Loop)

Funktion 1 (Demo erste Schritte mit der REPL)

* Welche Funktionen sollen hier gezeigt werden
* Println, str, +, -, /, \* ?????
* Siehe VS Project

Einfache Datentypen in Clojure (Datentypen)

* Class Funktion Präsentieren

Datenstrukturen in Clojure (Datentypen)

* Es gibt nur diese 4 Datenstrukturen
* Listen werden oft für Programme verwendet, können aber auch Daten beingalten.  
  Listen sind gut um vorne was dran zu hängen.
* Vektoren beinhalten Daten und eigen sich für eine wahlfreien Zugriff
* Maps sind Key/Value Pairs
* Sets haben keine Duplikate
* Key/Value Pairs in Maps werden oft komma seperiert. Komma ist aber Whitespace
* Alle diese Dinge können beliebig gemixt werden.

Demo Datentypen und Datenstrukturen

Funktionen 2 (Eigene Funktionen definieren)

* Wir beginnen mit einer runden Klammer
* defn steht für define function
* danach folgt der Funktionsname, in diesen Fall „greetings“
* danach folgt in einen eckigen Klammer Paar die Parameter Liste. Hier gibt es nur ein Parameter „name“.   
  Clojure ist dynamisch also keine Typ Deklaration
* Danach folgt der Funktionsrumpf, der hier aus den Aufruf der str Funktion besteht.
* str konkateniert die beiden parameter zusammen und gibt den zusamengesetzten String zurück.
* Es folgt Klammer zu für str und Klammer zu für defn.