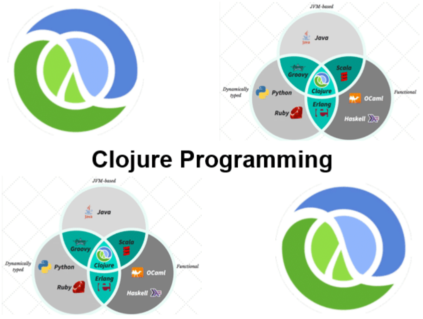
Clojure Programming

Duration: 3 days Price: € 1850

Open Schedule



Audience Course Clojure Programming

The course Clojure Programming is intended for anyone who wants to learn programming in the functional programming language Clojure.

Prerequisites Course Clojure Programming

In order to participate in this course basic knowledge of programming in another programming language is beneficial to the understanding but is not required.

Realization Training Clojure Programming

The theory is discussed on the basis of presentation slides. The theory is explained further through demos. After discussing a module there is the possibility to practice. Course times are from 9.30 to 16.30.

Certification Clojure Programming

After successful completion of the course the participants receive an official certificate Clojure Programming.

Content Course Clojure Programming

In the course Clojure Programming participants learn to program in the dynamic and functional programming language Clojure. Clojure has its roots in the Lisp programming language and there are compilers for both Java and .NET so that Clojure applications can run in both run time environments.

Clojure Intro

The course starts with an overview of the features of functional programming and how these are supported by Clojure. Attention is paid to expression evaluation, immutability and concurrency support. It also demonstrated how the REPL loop in Clojure can be used to run code.

Clojure Syntax

Subsequently the variables and data types of the Clojure language and the control flow constructions in Clojure are discussed. Exception handling is also treated.

Functions

Functions are of course a central element in a functional language like Clojure and it is discussed how functions work in Clojure. In addition to standard functions, anonymous functions, variadic functions and higher order functions such as map and reduce are also on the course program. The passing of functions as parameters and returning functions as return value is also treated.

Data Structures

Data Structures in Clojure are covered as well including sequences, lists, sets, vectors, maps and hash maps. And also attention is paid to the operations on data structures such as unions and intersections of sets and the sorting of maps.

Advanced Types

Also advanced types such as predicates, atoms, agents and watchers are discussed.

Multiple Threads

Finally it's time for multiple threads and concurrency in Clojure. In this respect object locking, transactions, agents and asynchronous calls are treated.

Modules Course Clojure Programming

Module 1 : Clojure Intro	**Module 2 : Language Syntax**	**Module 3 : Functions**
Functional Programming Clojure Roots in Lisp Expression Evaluation Clojure Compilers Running in JVM Running in .NET Immutability State Management Concurrency Support Installing Clojure Leiningen Tool CounterClockwise Plugin REPL Loop	Data Types Numbers and Strings char and Boolean nil and Atom Symbols Variables Require Statement Operators for and while doseq and dotimes if and if-do case and cond Exception Handling	Function Definition defn Macro Anonymous Functions Functions with Parameters Passing Arguments Variadic Functions Ampersand Symbol Higher Order Functions Passing Functions Returning Functions filter Function map and reduce iterate and remove
Module 4 : Data Structures	**Module 5 : Advanced Types**	**Module 6 : Clojure Concurrency**
Sequences seq Keyword Creating Lists ISeq Interface List Access Sets Set Intersection Set Union Vectors Maps Key Value Pairs HashMaps SortedMaps	Predicates Metadata vary-meta Atoms swap! Method compare-and-set! StructMaps Agents Watchers add-watch Macros macro-expand Date Class	Threads in Clojure Sharing Data References in Clojure Isolation in Vars Mutable State Object Locking Synchronization with Refs Transactions dosync Blocks Asynchronous Changes Agent State send and send-off await and await-for