[defsquare]

Clojure Training

Content & Organization

June 17th, 2024

Training Presentation

Goal: pass on the knowledge and good practice to **develop industrial-quality Clojure applications.**

Most of the **training is practical with simple but representative projects**. In addition, **we consolidate the fundamentals of software engineering, design practices and programming paradigms**. At the end of the course, the trainee will be autonomous enough to start up and maintain a Clojure project and can even lead a team towards a Clojure ecosystem.

Training conduct

Audience:

- Architects
- Developers

Prerequisites:

- A minimum of 2 years experience in software development

Training materials:

- Github repository, PDF slides (distributed at the end of the training), Miro, Teams recording

Duration:

- 4 days / 28 hours

Teaching mode proportions:

- 50% theorical, 50% practical. Exercises can be adapted to your context.

At the end of the training program, participants receive a certificate to confirm that they have achieved the objective.

Training organization

Location: training can be intra-company, or inter-company. In this case, the exact location will be provided during the registration.

Scheduling: according our website calendar for inter-company. For intra-company, scheduling is to be determined with the customer.

Costs:

- inter-company training : 1 900 €
- in-house training : price upon quotation

Contact us at +33 6 71 01 06 52 or formations@defsquare.com to discuss your needs. We will then send you a quote within a week.

For all individuals with disabilities, please feel free to contact us at +33 6 +33 6 71 01 06 52 or formations@defsquare.com.

Additionally, all our training sessions are available online.

Defsquare offers breakfast at the training location and lunch for each of your days with us.

[defsquare]

Clojure Immersion

Day 1	The Basics
Day 2	Let’s Build a Project
Day 3	Industrial-Grade Software
Day 4	Advanced Topics

Clojure Immersion

Day 1

Day 2

Day 3

Day 4

The basics

The first steps in the Clojure eco-system will be a **quick presentation on the history of Clojure** and from where it comes. We will present the **syntax with the main concepts and its philosophy**.

We will then introduce the **tools needed to work efficiently with Clojure**: the REPL and integration with code editors.

Finally, the trainees will handle the **various data structures and core functions** through several practical exercises.

Topics: Introduction, Language fundamentals, function, data structure, REPL, tooling, IDE.

Reached level ➡️ **Clojure Beginner**

Clojure Immersion

Day 1

Day 2

Day 3

Day 4

Let’s build a project

We will first consolidate our knowledge of **functional programming**. This will get us off to a good start and be able to express the behaviour of software in a more **expressive and maintainable** way. Besides, these foundations can be used in other languages.

Next, we'll create **a Clojure project with all lifecycles** through development and delivery. We will present the different **levels of testing** and their respective **tools**.

Finally, we will take stock of the **main libraries and frameworks** in the Clojure ecosystem and present **our own tech radar** as our selection of tools for developing industrial-quality applications.

Reached level ➡️ **Clojure Maintainer**

[defsquare]

Clojure Immersion

Day 1

Day 2

Day 3

Day 4

Industrial-grade software

The **shapes and structure of data are important** even in a non-typed language. **Controlling data structures** is essential to **ensure software consistency.**

During this day, we’ll look at some of **Domain-Driven Design concepts** and the **separation of code responsibilities between technical and domain needs**. We will then take a quick look at the different **styles of application architecture**, with their strengths and weaknesses.

Clojure(script) is a flexible language offering a high degree of freedom, but it is essential to master **the basics of industrial-strength software architecture**.

[defsquare]

Reached level ➡️ **Autonomous Clojure Developer**

Clojure Immersion

Day 1

Day 2

Day 3

Day 4

Advanced Topics

Finally, we'll look at topics such as the **compilation of Clojure into a browser**, offering the power to develop complete, high-performance **web interfaces**. By the same way as we explored server-side development, we will present the **ClojureScript ecosystem**, starting with the **main tools and frameworks** on the market, the **lifecycle of a web project** and how to implement it.

Finally, we will show how Clojure takes advantage of the richness of the JVM world and **the way it can plug-in** from/to JAVA tools.

[defsquare]

Reached level ➡️ **Clojure Lead Developer**

Design Software

Good Design: Domain-Driven, Stratified, Decoupled, Composable

[defsquare]

Build Software

Software Engineering, Functional Programming, Clojure

Run Software

Production-Ready Systems: Observable, Performant, Secure, Reliable, Cloud Native

[defsquare]