

Advanced Java

Duration:	2 days
Type:	advanced

Description

This course is designed to enable experienced Java developers to deepen their knowledge of the Java language and platform. It can be used to transition delegates from one version of Java to another, to formalize the education of self-taught developers or to assist programmers taking on new responsibilities as senior developers and/or architects.

Unfortunately there is no 'one size fits all' agenda for an advanced Java course. Each team will have its own needs depending on their current skill-set and the requirements of upcoming projects. The list of modules below provides an overview of topics that are commonly requested, but additional items can be included as required.

Prerequisites

Delegates should have at least 3 years experience in commercial Java programming

List of Modules

Generics in Depth

- Understanding how non-reified generics is implemented in Java 5

Writing Java 6 Annotation Processors

- Extending the Java language by creating your own compiler plug-ins

Object Oriented Design Patterns

- The most common patterns used in Java development

Writing Java Web Services

- Detailed coverage of JSR 181, JAX-WS, JAX-RS and JAXB

Java Language Idioms and Refactorings

- Improving existing code via incremental improvements and best practises

Using the Google Collections Library

- Taking advantage of the Google extensions to the standard collection types

The Log4J library

- Configuring and using Log4J correctly

Scripting on the JVM

- Automating build and testing tasks with JRuby, Jython and Groovy

Preview of new language features arriving in Java 7

- Project Coin, Closures and the Fork / Join framework

Test Driven Development

- Applying TDD using tools such as JUnit, TestNG, JMock and EasyMock

Overview of the Scala language

- Using Scala as a simpler and more powerful way of developing JSE applications

Modularizing Large Java Applications with OSGi

- Overcoming JAR versioning issues using OSGi bundles

Manipulating XML in Java

- Using JDOM, StAX, XPath, XSLT and XQuery

Using the Java Persistence API V2

- Simplifying your Data Access Layer using JPA and/or Hibernate