Java 12 Runtime Programming

IO, Processes, Serialization, Asynchronous, Security, Networking, Web Access

After developers learn the Java 12 language they next must learn about the Java 12 runtime environment and the APIs it provides. Java and its add-on packages offer a vast range of APIs and often it can be daunting for developers new to Java to figure out what goes where. Initially, to simply get work done for their specific assignments can be a challenge. This course aims to overcome this and takes developers already proficient in the Java 12 language on a walkthrough of common scenarios – we look at relevant APIs and the runtime ideas underlying them and help attendees write code efficiently and become productive as Java devs.

Beneath Java on every implementation is an OS, whose capabilities are exposed to Java applications via an API. The Java runtime itself, known as the JVM, adds additional capabilities. Base class libraries and layered libraries offer even more functionality. Taken together, a rich multi-layer of readily available functionality is provided for application developers to exploit in their own applications.

The aim of this rapid-paced course is to cover as much as possible of the fundamental APIs that devs need and provide them a good grounding in practical API usage.

Target Audience

Developers wishing to create libraries and applications using Java 12's runtime capabilities.

Prerequisites

Attendees must already have attended our *Java 12 Language* course or have similar Java language programming experience.

Note: This course does not cover Java multithreading. We offer a separate complete course on this topic.

Contents of One-Day Training Course

Overview of Java Runtime

Documentation and tooling
How everything works together
Overview of module layout
Interaction with the JVM
Tour of all major runtime features

Garbage Collector

Interacting with the GC
Impact of various approaches
Z Garbage Collector

java.io

File handling
File and string readers and writers
Buffering
java.util.zip

Serializable

Serializing and deserializing an object
NotSerializableException
Object stream APIs

Managing Processes

Process class represents a process
Creating a process with `ProcessBuilder`
Launch mechanisms – e.g. `VFORK` and (new to Java 12) `POSIX_SPAWN`
Redirect via `ProcessBuilder.Redirect`

Asynchronous I/O

Java async I/O design pattern
Streams
Asynchronous channel APIs

JMS – Java Message Service

Rich message exchange framework
Reliable / asynchronous
Point to point vs. pubsub

Utilities

Text handling & regex
Internationalization (e.g. new Unicode 11)
Time / mathematics / etc.

Java And Security

A comprehensive security framework
authentication, authorization, auditing
Java and PKI
Use of cryptographic algorithms

Advanced Security Features

SecurityManager
Keystore
Code security – code signing, bytecode verification, avoiding common threats

Java Networking

Socket programming with Java
specifying network addresses
Socket options
Creating UDP and TCP connections

Web Access

HTTP 1.1, HTTP/2 (JEP110) and HTTP/3
URI
SSL/TLS

Reflection

`java.lang.Object.getClass()`
How to use `java.lang.Class`
Reflection namespace – java.lang.reflect
Constructor, Field, Method, Parameter

Additional Libraries

java.instrument and logging
Transactions
Java Management Extensions (JMX)
Java Naming & Directory Interface (JNDI)