

Java 11 Runtime Programming

IO, Processes, Multithreading & Synchronization, Serialization, Security, Networking, Web Access

After developers learn the Java 11 Language they next must learn about the Java 11 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 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.

Contents of One-Day Training Course	
<p>Target Audience Developers wishing to create libraries and applications using Java's runtime capabilities.</p> <p>Notes this course does not cover user interface features of Java.</p> <p>Prerequisites Attendees must already have attended our <i>Java 11 Language</i> course or have similar Java language programming experience.</p>	<div style="display: flex; justify-content: space-between;"> <div style="width: 48%;"> <p>Overview of Java Runtime Documentation and tooling How everything works together Overview of module layout Interaction with the VM Tour of all major runtime features java.io</p> <p>File handling File and string readers and writers Buffering <code>java.util.zip</code></p> <p style="text-align: center;">Serializable</p> <p>Serializing and deserializing an object <code>NotSerializableException</code> Object stream APIs</p> <p style="text-align: center;">Managing Processes</p> <p>Process class represents a process Creating a process with <code>ProcessBuilder</code> Redirect via <code>ProcessBuilder.Redirect</code></p> <p style="text-align: center;">Multithreading</p> <p>Java's threading architecture The <code>Runnable</code> interface Creating threads with <code>java.lang.Thread</code> <code>ThreadLocal</code> – each thread has its own copy of the variable <code>ThreadGroup</code></p> <p style="text-align: center;">Synchronization</p> <p>A synchronized block <code>java.util.concurrent</code> for concurrency <code>java.util.concurrent.atomic</code>:* supports lock-free thread-safe programming on single variables <code>java.util.concurrent.locks</code>=locks</p> </div> <div style="width: 48%;"> <p style="text-align: center;">Utilities</p> <p>Text handling & regex Internationalization Time / mathematics / etc.</p> <p style="text-align: center;">Java And Security</p> <p>A comprehensive security framework authentication, authorization, auditing Java and PKI Use of cryptographic algorithms</p> <p style="text-align: center;">Advanced Security Features</p> <p><code>SecurityManager</code> Keystore Code security – code signing, bytecode verification, avoiding common threats</p> <p style="text-align: center;">Java Networking</p> <p>Socket programming with Java specifying network addresses Socket options Creating UDP and TCP connections</p> <p style="text-align: center;">Web Access</p> <p>HTTP 1.1, HTTP/2 (JEP110) and HTTP/3 URI SSL/TLS</p> <p style="text-align: center;">Reflection</p> <p><code>java.lang.Object.getClass()</code> How to use <code>java.lang.Class</code> Reflection namespace – <code>java.lang.reflect</code> Constructor, Field, Method, Parameter</p> <p style="text-align: center;">Additional Libraries</p> <p><code>java.instrument</code> and logging Transactions Java Management Extensions (JMX) Java Naming & Directory Interface (JNDI)</p> </div> </div>