

Agendavorschlag zum Seminar

Von Java 8 zu Java 17

Dauer: 3 Tage

Autor:
Stephan Karrer

Zu dieser Agenda

Diese Agenda ist ein Vorschlag bzgl. der Inhalte und Unterrichtszeiten für das Seminar

Von Java 8 zu Java 17

Bitte überprüfen Sie, ob diese Konzeption Ihren Erfordernissen entspricht. Änderungen oder Ergänzungen hinsichtlich der Seminarinhalte oder des Zeitplans nehmen wir in Absprache mit Ihnen, soweit möglich, gerne vor.

Zeitliche Struktur des Seminars

Wir gehen von 9:00 -17:00 täglicher Seminarzeit aus. Individuelle Regelungen zu den Unterrichtszeiten und den Pausenregelungen stimmen wir mit Ihnen ab.

Hard/Software - Medien

Die Teilnehmer benötigen PCs unter Windows oder Linux mit folgender zusätzlicher Software:

- Java Development Kit Version 17 (JDK 17)
- optional: Java Development Kit Version 8 (JDK 8)
- Eclipse fürEnterprise Developers als Entwicklungsumgebung (alternativ IntelliJ IDEA)
- Internet-Zugang

Modularisiertes JDK

- Die Ziele von Projekt Jigsaw
- Überblick über die Systemmodule
- Classpath vs. Module Path
- Einschränkungen: Interne APIs, Deep Reflection
- Compiler- und Laufzeit-Optionen zur Übersteuerung
- Umstrukturierungen im JDK
- Remove-EE (was entfällt und welche Alternativen gibt es)

Modularisierung eigener Anwendungen

- Öffentliche und private Schnittstellen
- Modul-Deskriptoren und Abhängigkeiten zu System-Modulen
- Definition eigener Module und deren Abhängigkeiten
- Sichtbarkeiten und Export öffentlicher Schnittstellen
- Wie unterstützt IDE (Eclipse) die Modularisierung?
- Modul-Builds mit Maven
- ServiceLoader im Kontext von Modulen
- Öffnen für Reflection
- Überblick zum Java Linker

Fortsetzung: Modularisierung eigener Anwendungen

- Reflection-Erweiterung: Modul-API
- Migrationsstrategien
- Unbenannte und automatische Module

Migrationsbeispiele und -herausforderungen

- Java 8 Anwendung auf Java 11 bzw. Java 17 migrieren
- Wo steht Java EE bzw. Jakarta EE

Syntax-Erweiterungen

- „var“ für Variablen, Typ-Inferenz
- „switch“-Ausdrücke
- Records als Datenklassen
- Textblöcke
- Sealed Classes
- Pattern-Matching bei der Typprüfung
- Sonstiges

Neues im JDK und der VM

- Nettes Spielzeug: JShell
- Launch Single-File Source-code Programs
- Shared Archive: CDS
- Multirelease-Jars
- Flight Recorder (Einsatz und Möglichkeiten)
- Neues Lizenzmodell von Oracle
- Sonstiges

Neue APIs und API-Ergänzungen

- Immutable Collections
- Flow API (Einführung Reactive Programming)
- HTTP/2 Client
- String API-Änderungen
- Process API
- Erweiterungen bei Streams, CompletableFuture und Optional
- Sonstiges: XML Catalog, Stack-Walking API, UTF-8 Property Resource Bundles, ...
- Was ist Deprecated und vermutlich relevant