

Java Programmierung auf dem Mainframe – Jakarta für Mainframe Entwickler*innen

Zielgruppe

- **Anwendungsentwickler*innen**
- **COBOL Anwendungsentwickler*innen**

Version 1.1 vom Dezember 2022

Autoren: Wolfram Greis, Prof. Dr. Philipp Brune

European Mainframe Academy GmbH
Am Klostergarten 3
D 78337 Öhningen
Tel. +49-7735-9388668
info@mainframe-academy.de

European Mainframe Academy AG
Obergass 23
CH 8260 Stein am Rhein
Tel. +41-52-558 20 40
info@mainframe-academy.de

Inhaltsverzeichnis

| | | |
|---|-----------------------------------------|---|
| 1 | Ziele des Ausbildungsmoduls | 3 |
| 2 | Randbedingungen / Voraussetzungen | 3 |
| 3 | Inhalte | 4 |

1 Ziele des Ausbildungsmoduls

Ziel des Moduls ist es, den Teilnehmern **Java auf der Serverseite** beizubringen. Das lief ursprünglich mal unter **Java Enterprise Edition**, seit 2018 jedoch unter **Jakarta**. Hierbei wird speziell die Zielgruppe Entwickler*innen für den Mainframe angesprochen. Wir haben speziell Erfahrung in der Ausbildung von COBOL Entwickler*innen, da unsere Referenten sowohl Java als auch COBOL beherrschen. Die APIs für die Zugriffe in das Backend werden detailliert behandelt. Es geht hierbei um Servlets, JDBC, JPA, EJBs, JSPs, JSF, JMS. Es wird vermittelt, wie Unit Tests organisiert werden. RESTful Web Services mit JAX-RS werden behandelt und Java Batch Verarbeitung ist ein Thema.

2 Randbedingungen / Voraussetzungen

Java Grundlagen (Java SE), wie sie in unserem Basics-Seminar vermittelt werden, werden vorausgesetzt.

Den Teilnehmern wird für die Dauer der Ausbildung ein Zugang auf dem z/OS System der EMA (aktuell: z/OS 2.4) zur Verfügung gestellt. Dieser Zugang steht auch nach Beendigung der Ausbildung zur Verfügung. Die EMA garantiert diesen Zugang für mindestens drei Monate nach Ausbildungsende.

Wir setzen ausserdem voraus, dass die Teilnehmer diverse Werkzeuge auf ihrer Workstations installieren können.

Als Begleitmaterial werden Unterlagen und Folien der EMA und von Partnerunternehmen eingesetzt. Ergänzend wird auf Redbooks und weitere (offene) Literatur verwiesen.

Die Ausbildung wird nach dem bewährten Blended-Learning Konzept der EMA durchgeführt. Nähere Informationen dazu gibt es auf unserer Website;

www.mainframe-academy.de

3 Inhalte

Java EE und Jakarta

- Java SE / Java EE / Jakarta
- Wie kam es zu Jakarta?

Java Database Connectivity (JDBC)

- API für den Zugriff auf (R)DBMS
- Typen von JDBC Treibern
- Grundelemente der JDBC API
- Abfrage- und Änderungsoperationen
- Transaktionssteuerung

Java Persistence API (JPA)

- Object-Relational Mapping (ORM)
- JPA Provider
- ObjectDB
- Grundelemente von JPA
- Konfiguration von JPA
- Annotationen
- Lazy Loading mit FetchType
- Zugriff per Entity-Manager
- Java Persistence Query Language
- Konfiguration in persistence.xml

Unit Tests

- Tests auf der Ebene von Codebausteinen
- Unit-Test Frameworks (z.B. JUnit)
- Testabdeckung
- Simulation durch Mock-Up-Objekte

RESTful Web Services mit JAX-RS

- Was sind Web Services?
- HTTP Request-Typen
- GET/PUT/POST/DELETE
- JavaScript Object Notation (JSON)
- Implementierung von REST Services

Enterprise Java Beans (EJBs)

- Software-Komponenten für Business-Logik
- EJBs als Java-Klassen
- Dienste eines EJB Containers
- Transaktionssteuerung / verteilte Transaktionen
- EJB Container
- Entity Beans, Session Beans und MDBs
- Stateless, Stateful und Singleton EJBs
- JMS, verknüpfbar mit IBM MQ
- Dependency Injection (Inversion of Control)
- Java Transaction API (JTA) und XA Transaktionen
- Verteilte Transaktionen und Transaktionsmonitore
- Container- oder Bean-managed Transaktionen

Java Message Services (JMS)

- MoM Unterstützung für Java
- JMS Provider
- Queues und Publish/Subscribe (Topics)
- Versenden von Messages
- Aufbau einer Message (Header und Body)
- ConnectionFactory Objekt
- Polling von Queues
- MDBs als Listener für Queue / Topic
- MDB Konfiguration (server.xml)

Java Batch Verarbeitung

- JSR 352
- Standardisierte API
- Liberty Implementierung
- Chunk-Verarbeitung und Batchlet
- REST-API oder WOLA
- Job Specification Language (JSL) in XML implementiert
- Batch Interfaces
- ItemReader, ItemProcessor, ItemWriter