



Software Voraussetzungen für den Coding Thursday

Übersichtsliste

- Browser (Chrome, Firefox, ...)
- Postman
- Docker (inkl. docker-compose)
- Java **JDK** >=8 (bitte nicht die Version 11)
- Git
- Maven

Vorab Installationen

Bitte unbedingt folgende Software vorab runterladen und installieren.

- Browser (Chrome, Firefox, ...)
- Postman (<https://www.getpostman.com/apps>)
- Docker (inkl. docker-compose)
- JAVA IDE der Wahl (z.B. Eclipse, IntelliJ)
- Java **JDK** >=8

Die restlichen Teile aus dem unteren Teil der Konfiguration werden auch im Workshop besprochen.

Details zur Docker Installation:

1. Bitte unter <https://store.docker.com/signup> einen User (genannt Docker ID) anlegen, um Docker und den Docker-Hub nutzen zu können!
2. Für das entsprechende Betriebssystem herunterladen
 - a. Windows Prof. + MacOS: (<https://www.docker.com/get-started>)
 - b. Windows Home: https://docs.docker.com/toolbox/toolbox_install_windows/
 - c. Andere Betriebssysteme (z.B. Ubuntu):
<https://store.docker.com/search?architecture=amd64&offering=community&type=edition>
3. Mit dem grafischen Interface (Rechte Maustaste auf das Wal-Icon) oder in der Kommandozeile mit „*docker login*“ anmelden
4. Mit „*docker run hello-world*“ testen ob wirklich das hello-world Image runtergeladen und gestartet wird (starten=nur kurzes Infofeedback)
5. Wenn Schritt 4 erfolgreich war bitte das openjdk java image herunterladen mit dem Befehl:
„*docker run -it openjdk:8-jdk-alpine java -version*“
Falls die Ausführung erfolgreich gestartet wird exekutiert der Container den Befehl „*java -version*“ und gibt daher eine kurze Info über die (!im Container!) installierte JDK.

Details zur Java Installation:

Idealerweise die JDK in der Version 8 installieren. Falls eine andere Version >8 aber <11 vorhanden ist sollte es auch keine Probleme geben.

- **Oracle JDK 8:** <https://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>
- **Open JDK 8:** <https://openjdk.java.net/install/>

Details zur Java IDE

- Eclipse download von <https://www.eclipse.org/downloads/>
- IntelliJ download von <https://www.jetbrains.com/idea/download/>
- Oder nach Belieben ...

Konfiguration

Hier nun Details zu der weiteren Konfiguration, welche auch in dem Workshop kurz behandelt werden.

Java

JAVA_HOME Variable setzen. Siehe <https://www.java.com/de/download/help/path.xml>

Docker

Windows und MacOS: In den Docker-Einstellungen die Option "Expose daemon on tcp://localhost:2375 without TLS" aktivieren (für Maven-Plugin „com.spotify“)

Git Installieren

Git mit Hilfe von <https://git-scm.com/downloads> installieren. Für Windows folgende Einstellungen im Install-Wizard vornehmen:

- „Use VIM“ wählen
- Use Git from Git Bash only
- Use the OpenSSL Library
- Checkout Windows-Style ...
- Use MinTTY ...
- „Enable Symbolic Links“ anklicken
- keine „Experimentals“ anklicken

Danach Workshop-Vorlage mit Hilfe von

`git clone` <https://github.com/FH-Tutorials/spring-boot-docker-example.git>

in ein Verzeichnis der Wahl klonen.

Maven

Installieren für

- Windows (in Powershell):
 1. Package Manager für Windows (<https://scoop.sh/>):
`iex (new-object net.webclient).downloadstring('https://get.scoop.sh')`
 2. Maven Install:
`scoop install maven`
 3. Optional curl für Windows (z.B. statt Postman):
`scoop install curl`
- Linux-User: `sudo apt-get install maven`
- Mac-User: `brew install maven`

Anschließen mit

```
mvn -version
```

überprüfen ob die Installation korrekt funktioniert hat (wenn nicht wurden evtl. JAVA_HOME nicht korrekt gesetzt).

Zum Schluss kann das Projekt im Verzeichnis *spring-boot-docker-example* gebaut werden mit dem Befehl:

```
mvn clean install
```

Danach kann das Beispiel mit „*docker-compose up*“ mit 3 docker Container starten. Zum Stoppen kann „*docker-compose down*“ verwendet werden.