

# Installation of the Environment

This lesson gives you an introduction to what a local setup for these examples will look like!

## WE'LL COVER THE FOLLOWING ^

- Introduction
- Installing necessary software

## Introduction #

This course focuses first on introducing technologies. An example system is presented for each technology in each chapter. In addition, the quick start allows the reader to rapidly gain practical experience with the different technologies and to understand how they work with the help of the examples.

- First, the necessary software has to be *installed* on the computer. The installation is described in this lesson.
- *Maven* handles the build of the examples. The [next lesson](#) explains how to use Maven.
- All the examples use *Docker* and *Docker Compose*. A couple of lessons in this appendix describe the most important commands for Docker and Docker Compose.

For the Maven-based build and also for Docker and Docker Compose, the chapters contain basic instructions and advice on troubleshooting.

## Installing necessary software #

The source code of the examples is available on Github. For access, version control *git* must be installed, see <https://git-scm.com/book/en/v2/Getting-Started-Installing-Git>. If the installation was successful, a call of `git` in the command prompt will work.

The examples are implemented in Java. Therefore, *Java* has to be installed. Instructions can be found at [https://www.java.com/en/download/help/download\\_options.xml](https://www.java.com/en/download/help/download_options.xml). Since the examples have to be compiled, a JDK (Java Development Kit) has to be installed. The JRE (Java Runtime Environment) is not enough. When the installation is completed, it should be possible to start `java` and `javac` in the command prompt.

The examples run in Docker containers. This requires an installation of *Docker Community Edition*, see <https://www.docker.com/community-edition/>. Docker can be called with `docker`. This should work without errors after the installation.

The examples require a lot of memory in some cases. *Docker should have about 4 GB* available. Otherwise, Docker containers may be terminated due to lack of memory. Under Windows and macOS you can find the settings for this in the Docker application under Preferences/Advanced. If there is not enough memory, Docker containers are terminated. This is shown by the entry `killed` in the logs of the containers.

After installing Docker, you should be able to call `docker-compose`. If *Docker Compose* cannot be invoked, a separate installation is necessary, see <https://docs.docker.com/compose/install/>.

---

In the next example, we'll take a closer look at Maven!