

SEW3

IT-Medientechnik

Softwareentwicklung

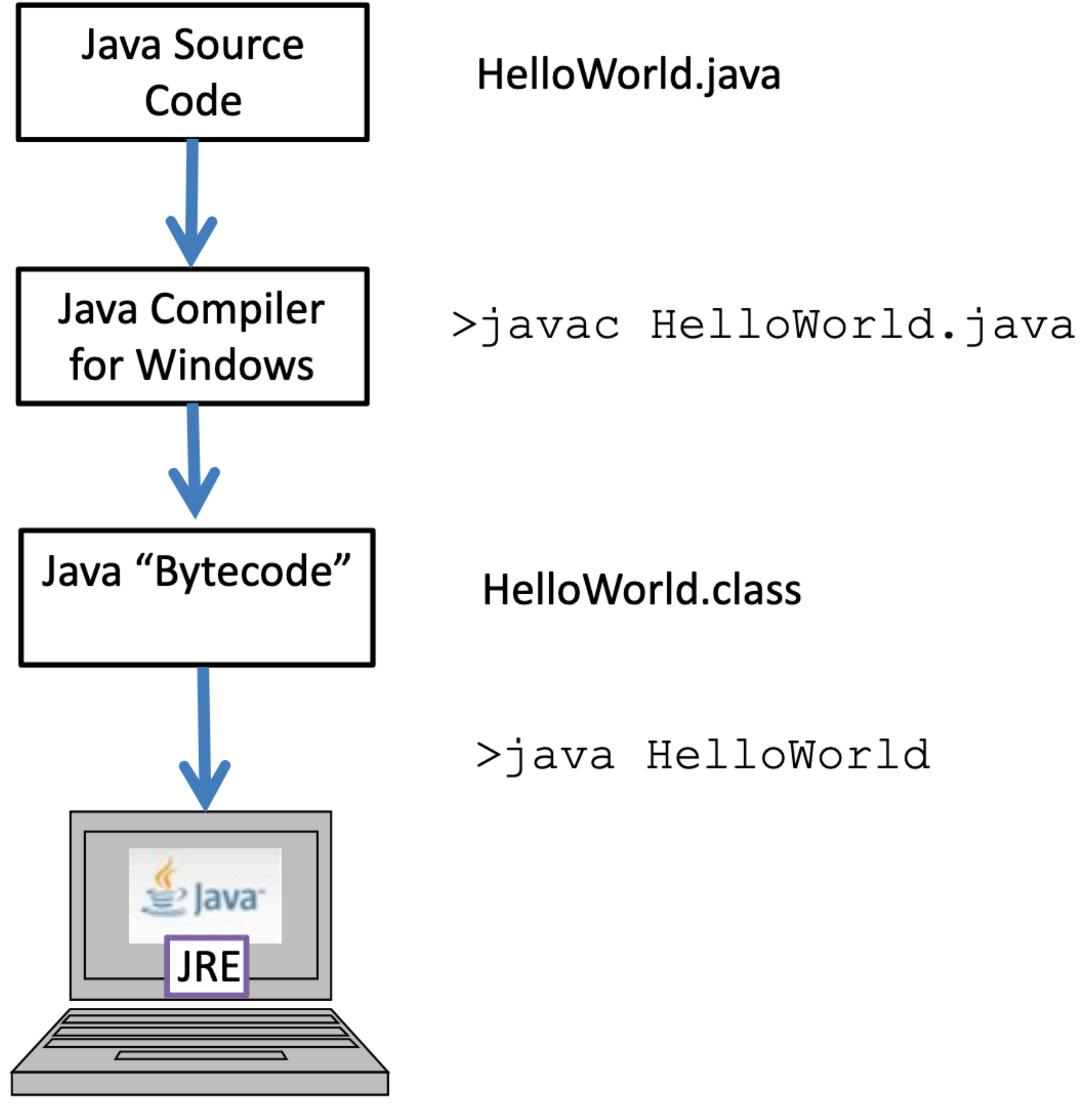
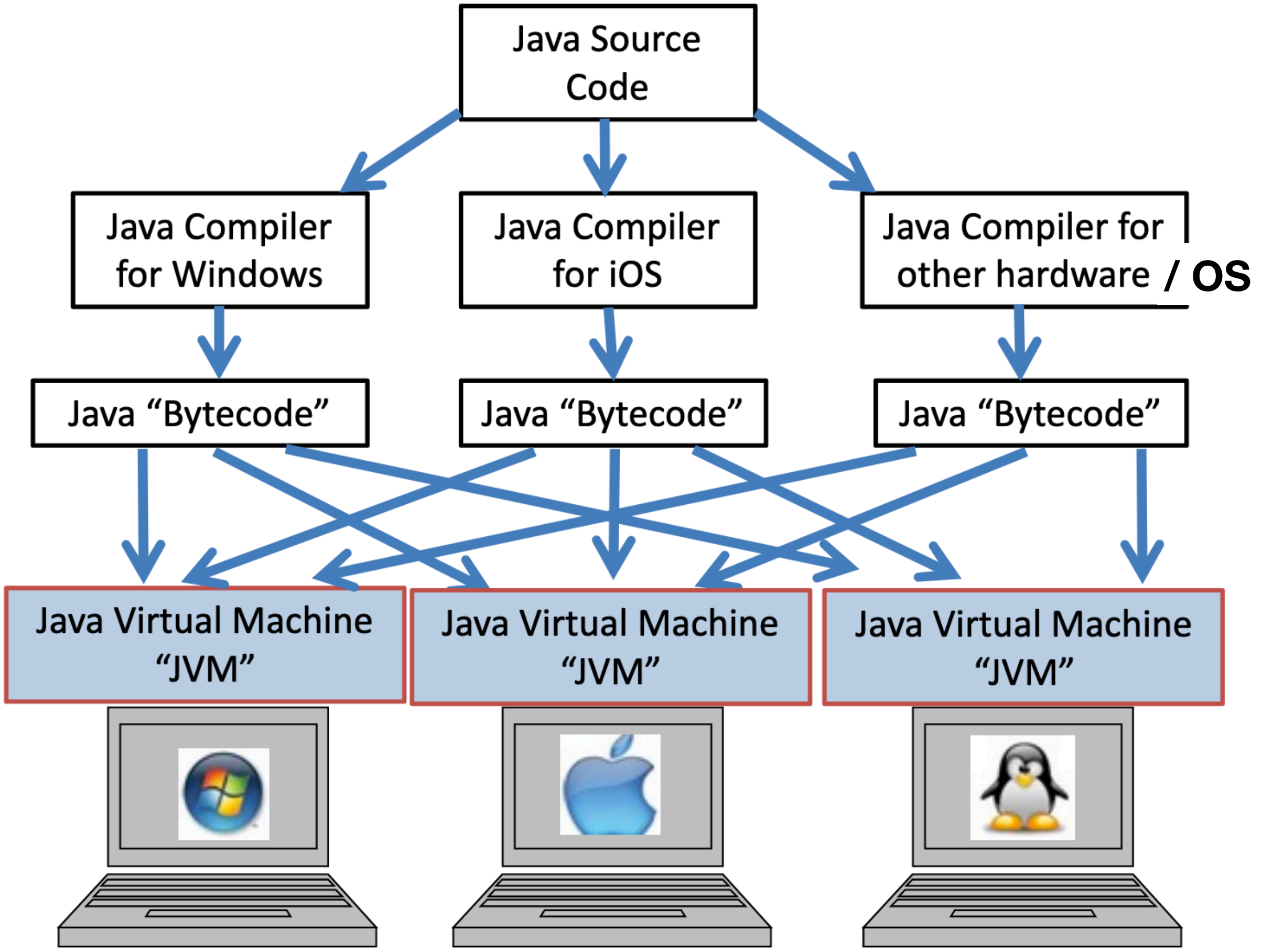
3. JG

IT-Medientechnik

TWS

Overview

Write Once; Run Anywhere!



Java-Versionen

Java SE
Standard Edition



- Java SE beinhaltet die Basistechnologien für Desktops

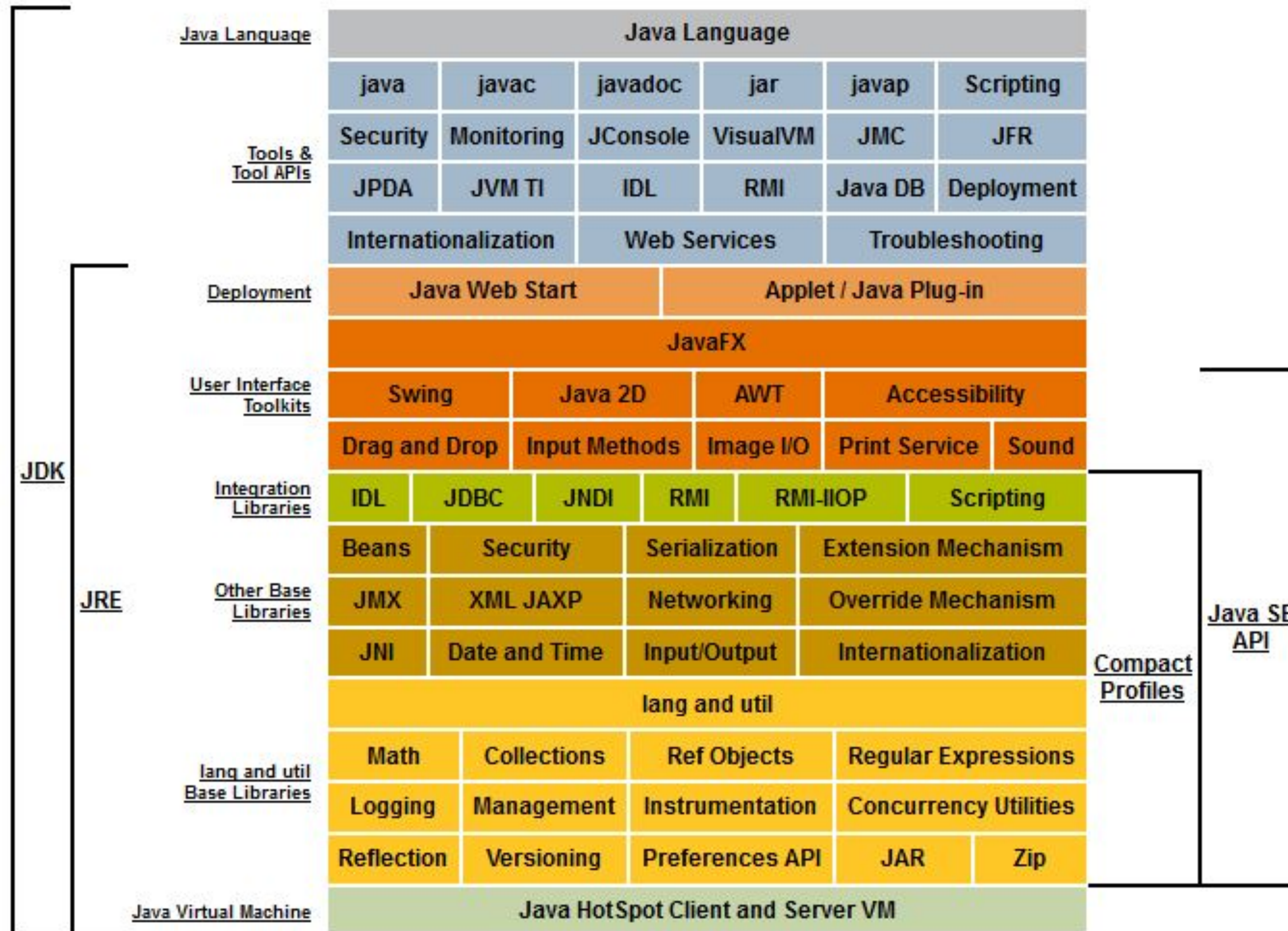
Jakarta EE
Enterprise Edition



JAKARTA EE

- früher Java EE
- Java EE wurde von der Eclipse Foundation übernommen und weiterentwickelt
- beinhaltet Technologien für verteilte Systeme
- wird immer mehr mit Microprofile kombiniert (Quarkus, Helidon, ...)

Java SE



Welches Java JDK?

- Oracle JDK (lizenzenrechtlich eingeschränkt nutzbar)
<https://www.oracle.com/technetwork/java/javase/downloads/index.html>
- **Oracle openJDK** (kann als vollwertiger Ersatz verwendet werden)
<https://openjdk.java.net/> , <https://jdk.java.net/>
- **AdoptOpenJDK** (wird von IBM, Microsoft usw unterstützt, kann bedenkenlos verwendet werden)
<https://adoptopenjdk.net/> (Hinweis: Mac-User können sich AdoptOpenJDK per HomeBrew installieren)
- **Amazon Corretto** (wird in der AWS verwendet, ebenfalls zu empfehlen)
<https://aws.amazon.com/de/corretto/>
- **Zulu openJDK** (könnte auch verwendet werden)
<https://www.azul.com/downloads/zulu-community/>

In den meisten Linux Systemen wird über den Paketmanager Oracle openJDK installiert und ist daher am meisten verbreitet. AdoptOpenJDK und Amazon Corretto JDK sind durch ihren Einsatz in der jeweiligen Cloud (azure bzw aws) sicherlich produktionsbereit

Java installieren

Empfehlung: Es ist sinnvoll, vor allem eine LTS-Version (Long Term Support) zu installieren, also derzeit 8 oder 11



Linux



Oracle Java (nur für Developer)

<https://www.oracle.com/technetwork/java/javase/downloads/index.html>

freies Java: zB AdoptOpenJDK



open JDK

```
sudo apt update

sudo apt install default-jdk
oder
sudo apt install openjdk-11-jdk

java -version
```

siehe auch:
<https://wiki.ubuntuusers.de/Java/Installation/OpenJDK/>

Liste aller verfügbaren Version:
`brew search adoptopenjdk --casks`

AdoptOpenJDK

Empfehlung:
Verwendung von Homebrew



https://brew.sh/index_de

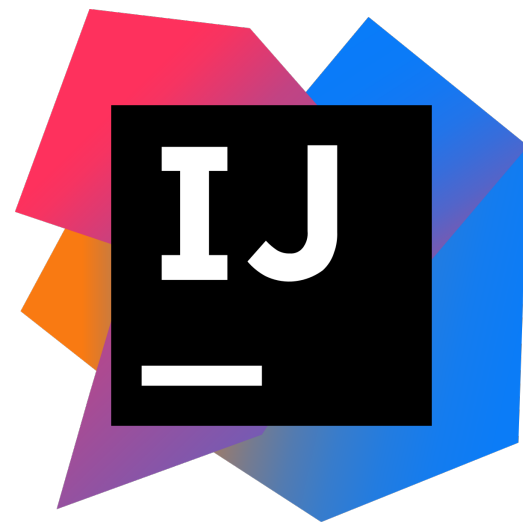
anschließend Installation
der gewünschten JDK-Version

```
brew tap AdoptOpenJDK/openjdk
brew cask install <version>
```

```
brew tap AdoptOpenJDK/openjdk
brew cask install adoptopenjdk-openjdk11
```

Liste aller Versionen:
<https://github.com/AdoptOpenJDK/homebrew-openjdk>

Tooling



IDE
IntelliJ IDEA Ultimate



Build Tool
Apache Maven



Test Framework
JUnit 5

<https://junit.org/junit5/docs/current/user-guide/>

<https://github.com/junit-team/junit5-samples/tree/r5.5.1/junit5-jupiter-starter-maven>



git on GitHub

<https://books.goalkicker.com/GitBook/>



GitHub Classroom

GitHub Classroom

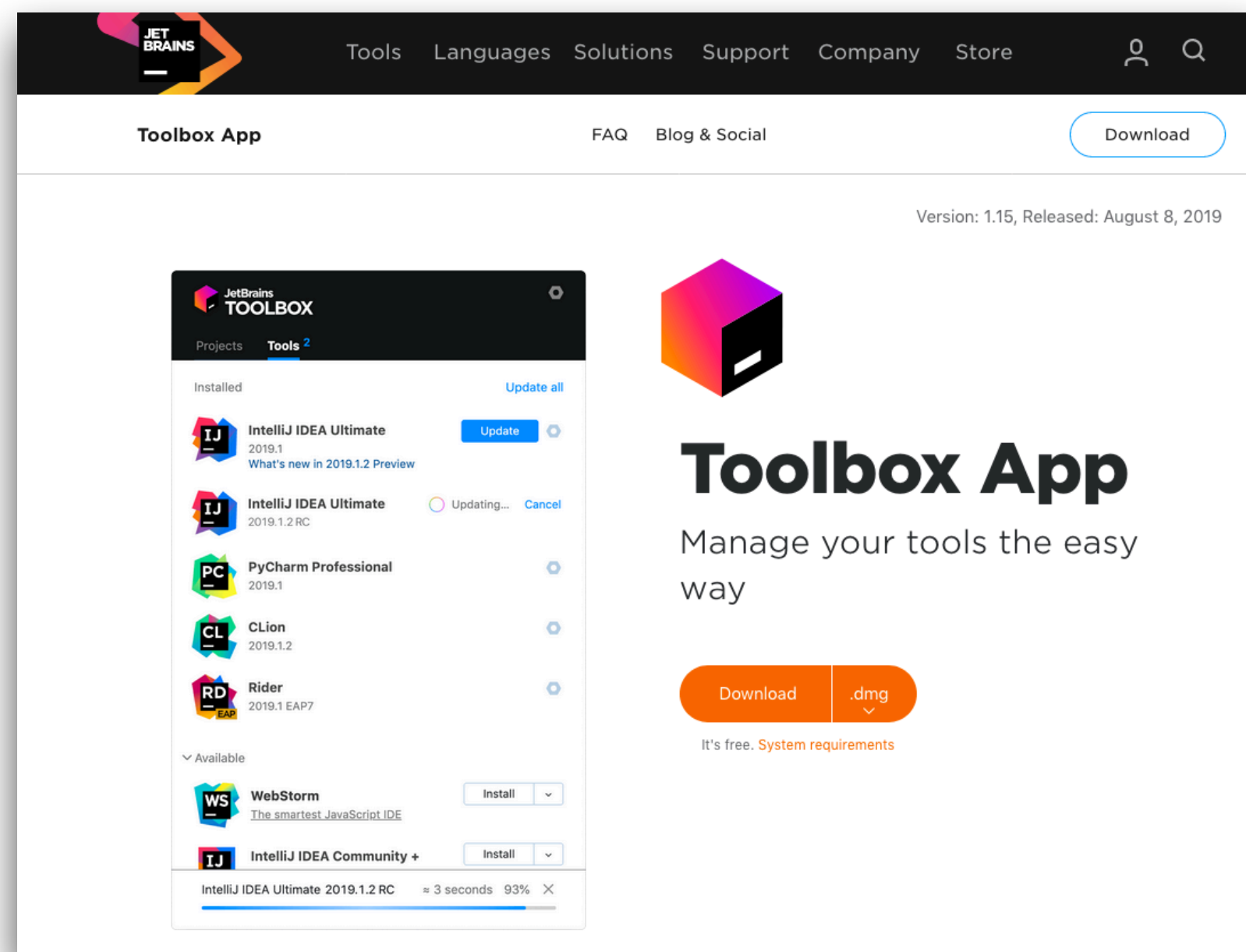


<https://edufs.edu.htl-leonding.ac.at/moodle/login/index.php>

Installieren von JetBrains-Produkten

1 Installieren der JetBrains Toolbox

<https://www.jetbrains.com/toolbox-app/>



2

das gewünschte Produkt installieren:
IntelliJ IDEA Ultimate

3

Ein Konto bei JetBrains erstellen und sich
gratis alle JetBrains-Lizenzen besorgen

beim HTL-Leonding Studentenemail einloggen

<https://studentsmx.htl-leonding.ac.at/roundcube/>

ein Konto bei JetBrains erstellen (mit Studentenemail)

<https://account.jetbrains.com/login>

Tipp: Passwort merken, da die Lizenz jährlich zu verlängern ist

**Auf der JetBrains-Seite mit neuem Konto anmelden
und um die Studentenlizenz ansuchen**

<https://www.jetbrains.com/student/>

APPLY NOW

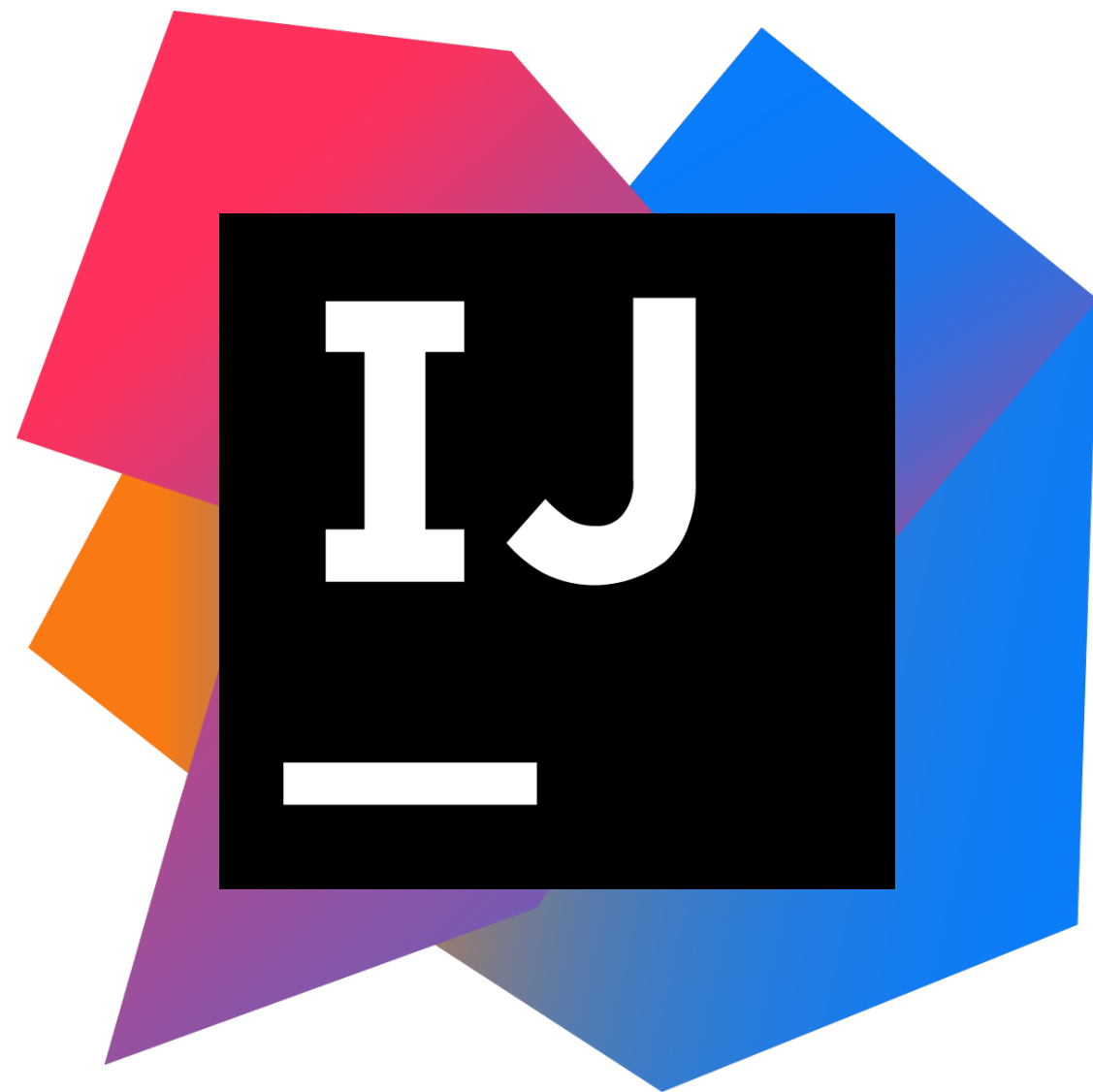
siehe auch:

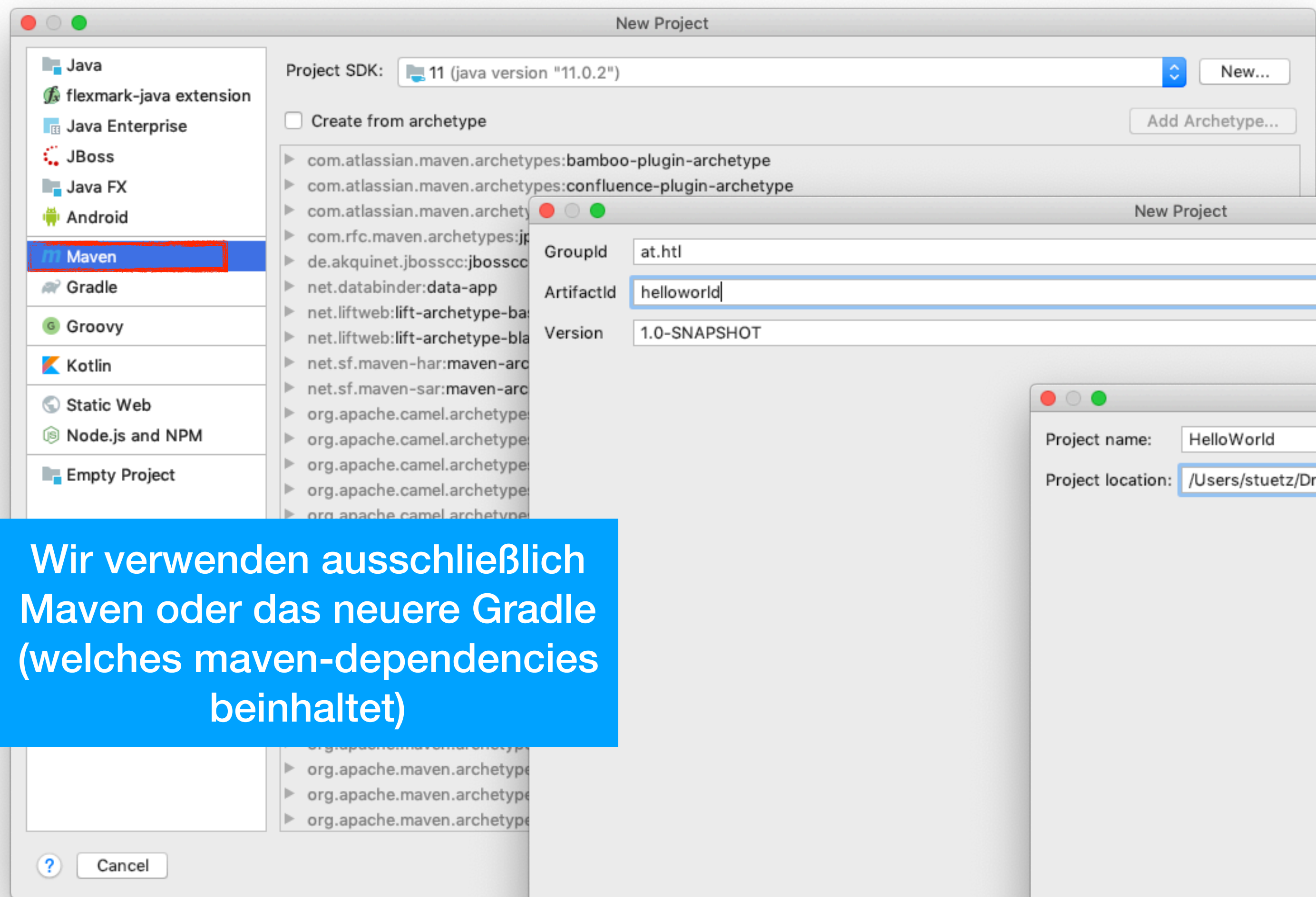
<http://edufs.edu.htl-leonding.ac.at/~t.stuetz/download/nvs/JetBrains.Registrierung.Studentenaccount.pdf>

Syllabus

- Java basics
- Unit testing
- Collections
- Exception Handling
- Java FX
- Lambda expressions
- Java Streaming API
- Java FX / MVC Pattern
- Threads
- Database access: JDBC
- Sockets / Serialization
- (Remote Method Invocation – RMI)

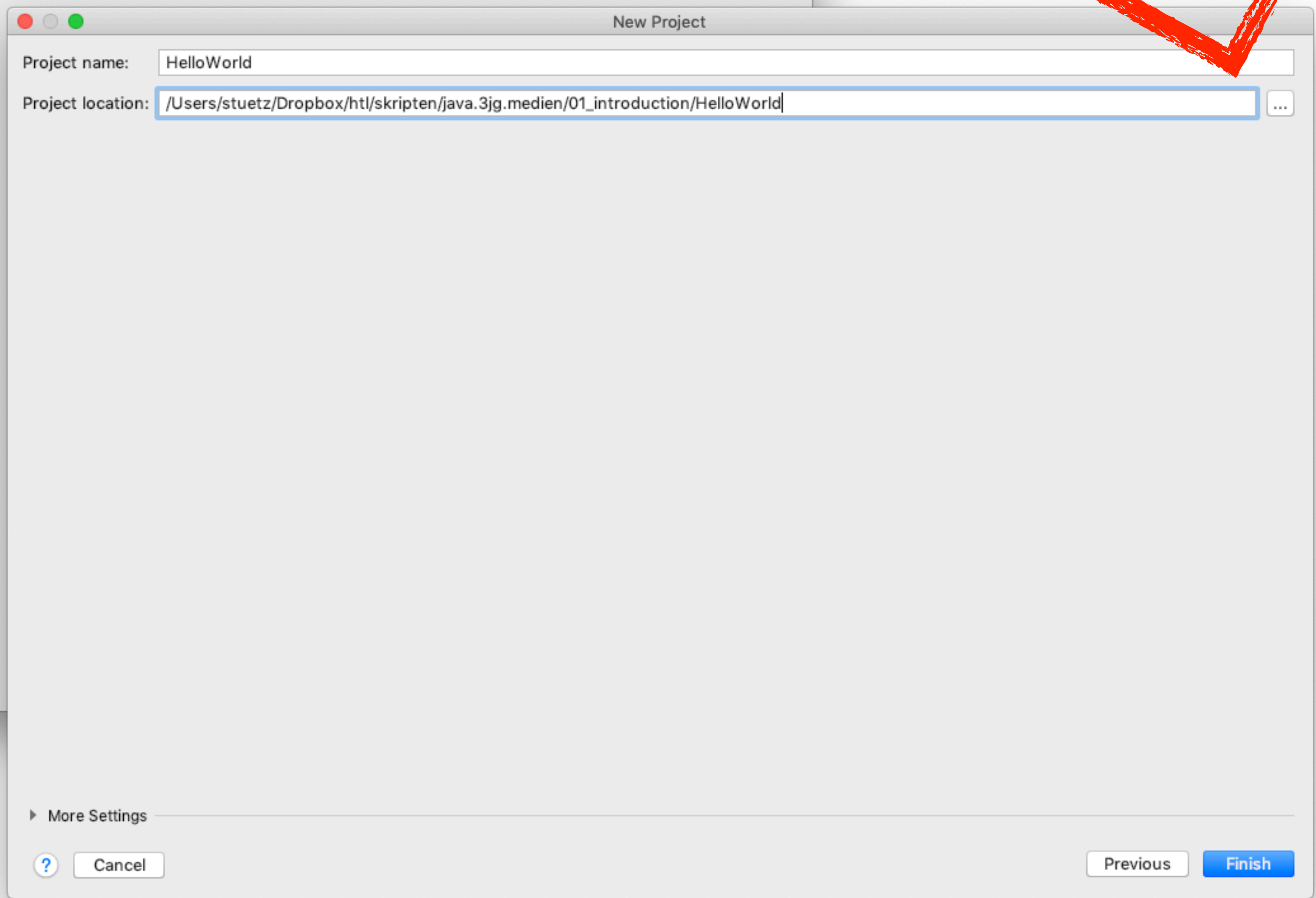
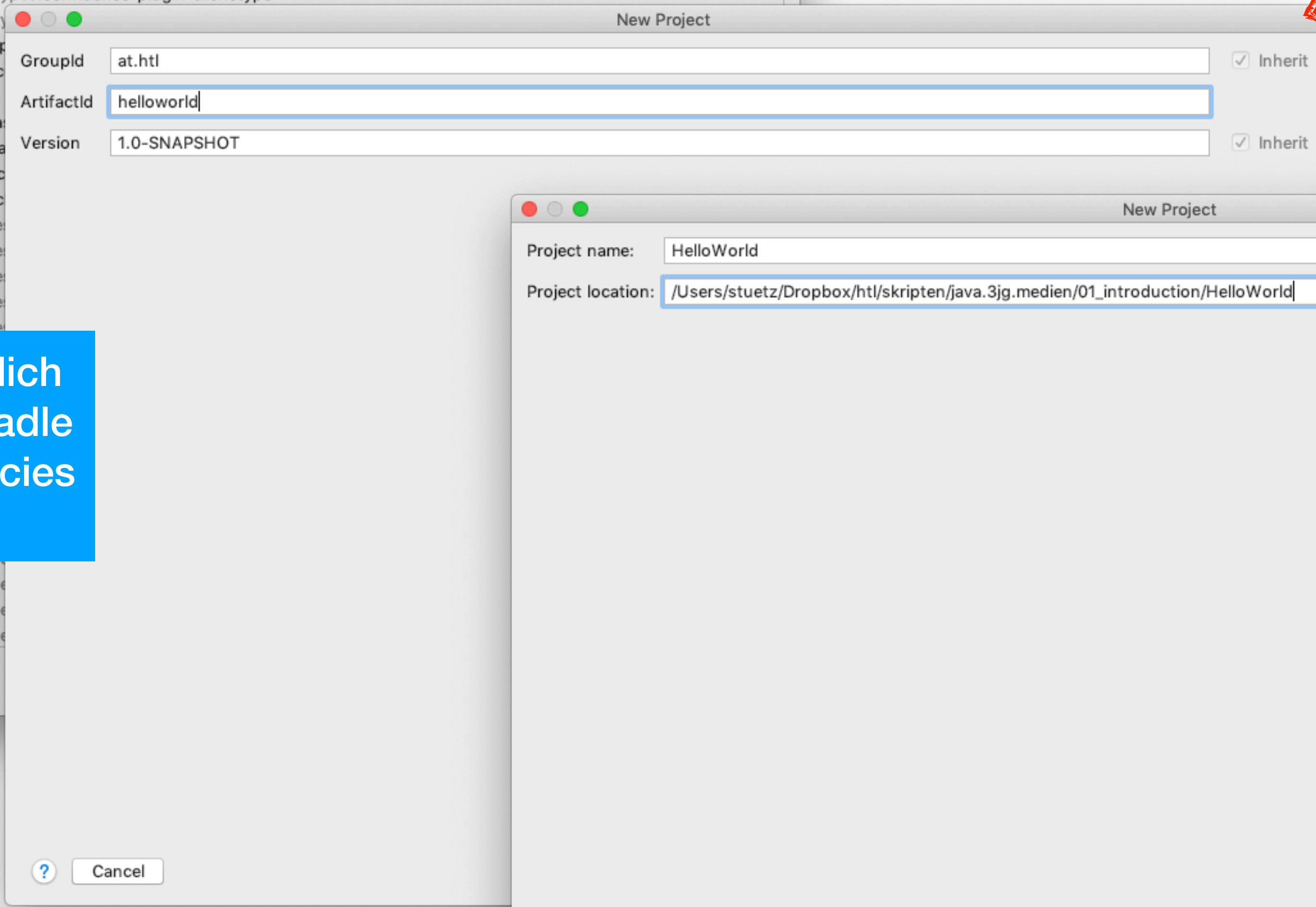
Create a project with IntelliJ IDEA Ultimate





wir verwenden
Maven

Wir verwenden ausschließlich
Maven oder das neuere Gradle
(welches maven-dependencies
beinhaltet)



Maven projects need to be imported
Import Changes **Enable Auto-import**

pom.xml

Intellisense verwenden

<https://stuetzpunkt.wordpress.com/2019/07/16/pom-xml-for-junit5/>

```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>

  <groupId>at.htl</groupId>
  <artifactId>helloworld</artifactId>
  <version>1.0-SNAPSHOT</version>
  <packaging>jar</packaging>

  <properties>
    <maven.compiler.source>11</maven.compiler.source>
    <maven.compiler.target>11</maven.compiler.target>
    <junit.jupiter.version>5.5.1</junit.jupiter.version>
  </properties>

  <dependencies>
    <dependency>
      <groupId>org.junit.jupiter</groupId>
      <artifactId>junit-jupiter</artifactId>
      <version>${junit.jupiter.version}</version>
      <scope>test</scope>
    </dependency>
    <dependency>
      <groupId>org.hamcrest</groupId>
      <artifactId>hamcrest-all</artifactId>
      <version>1.3</version>
      <scope>test</scope>
    </dependency>
  </dependencies>

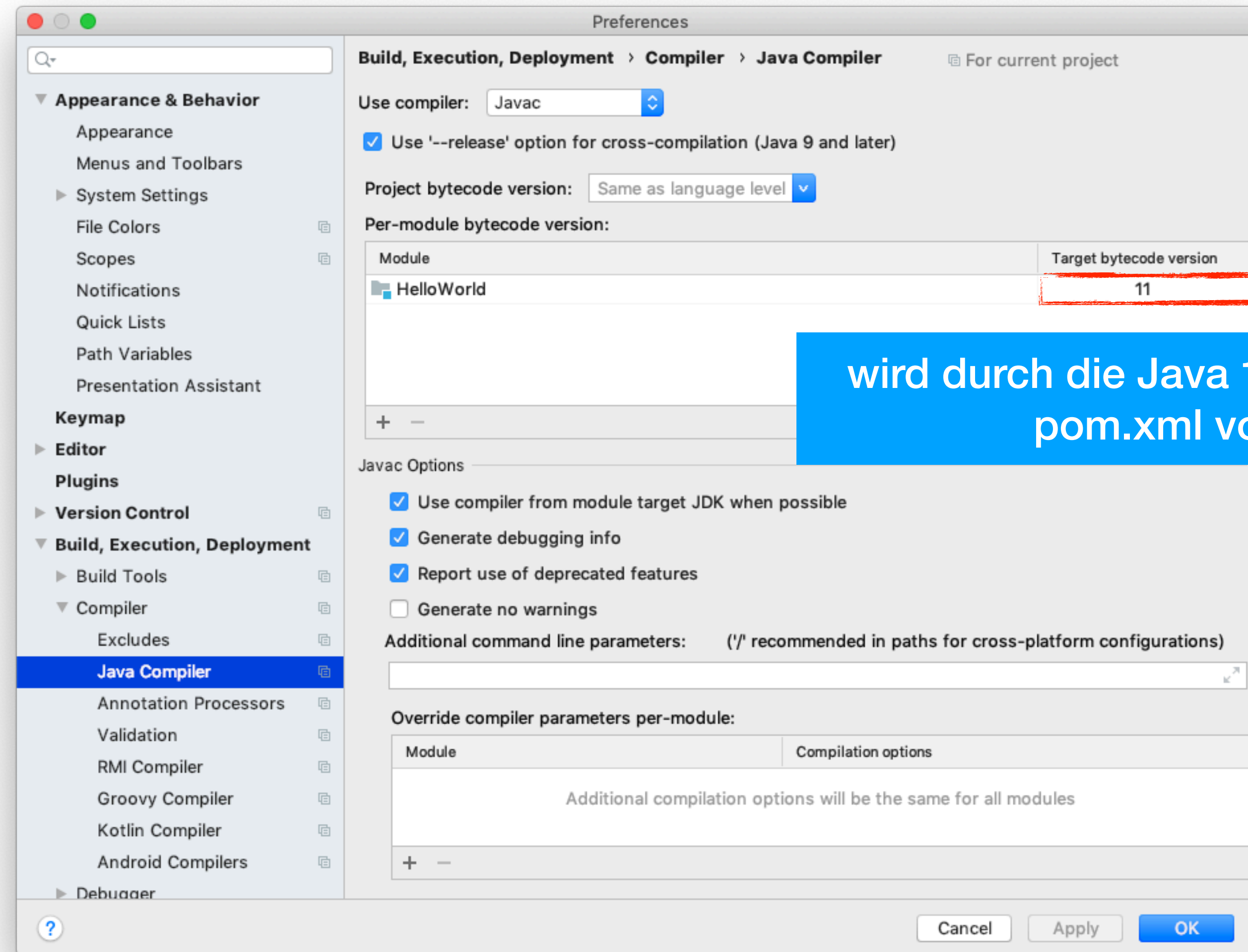
  <build>
    <finalName>helloworld</finalName>
  </build>
</project>
```

junit und hamcrest werden nur für das Testen benötigt und sind hier nur der Vollständigkeit halber eingetragen

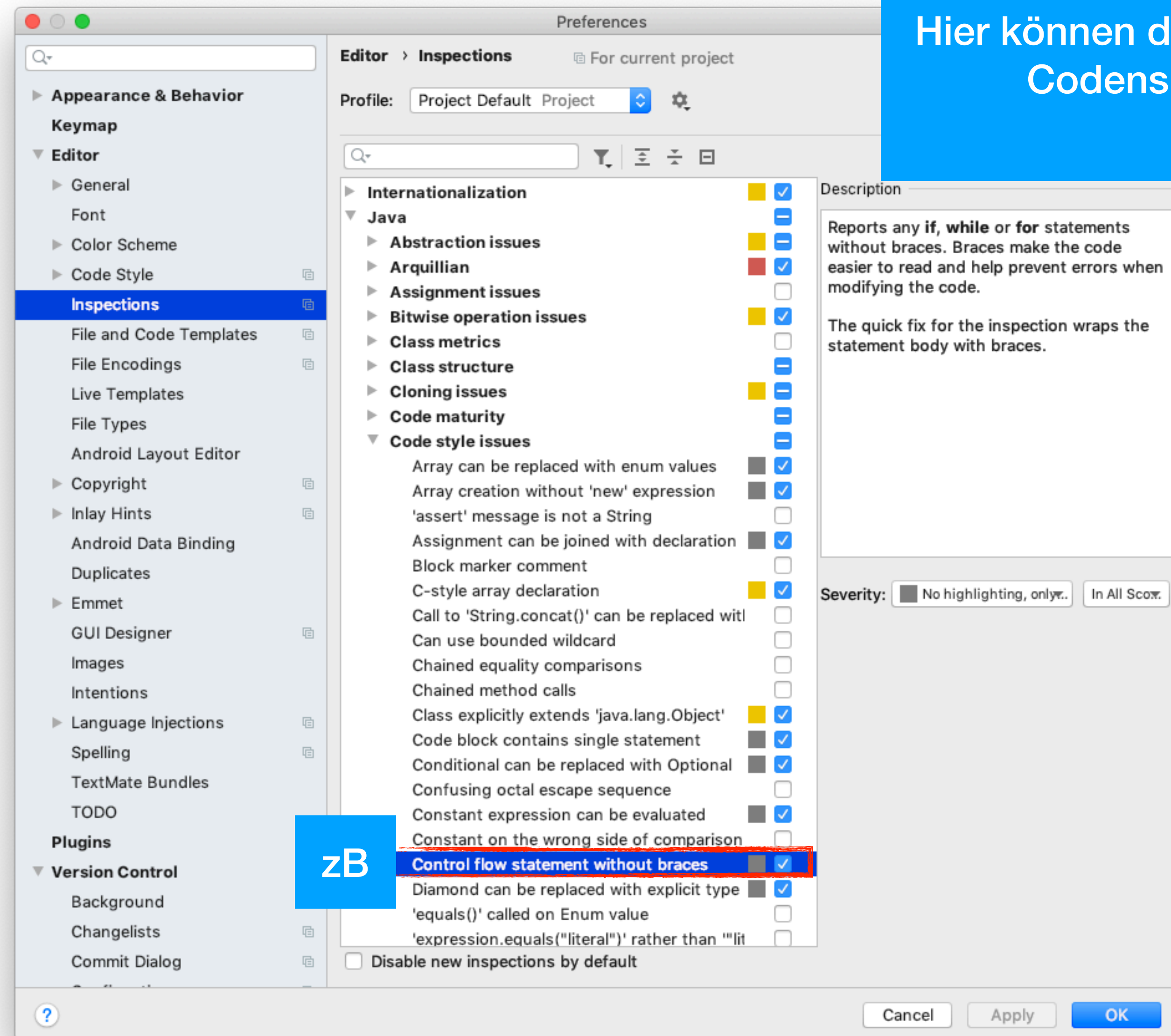
Durch die Java 11 - Properties in der pom.xml werden Einstellungen in der IDE für Java 11 durchgeführt (leider nicht durchgängig)

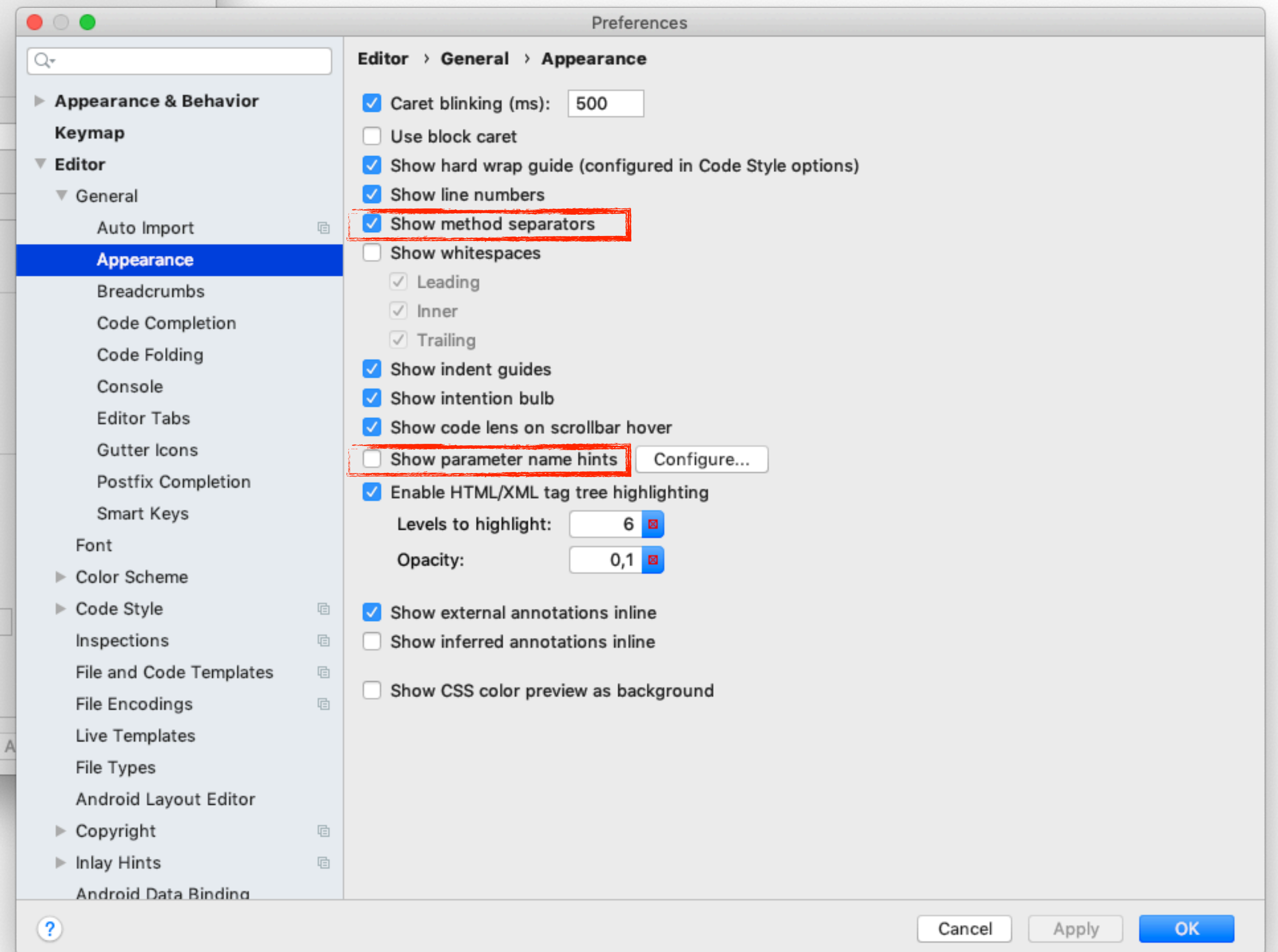
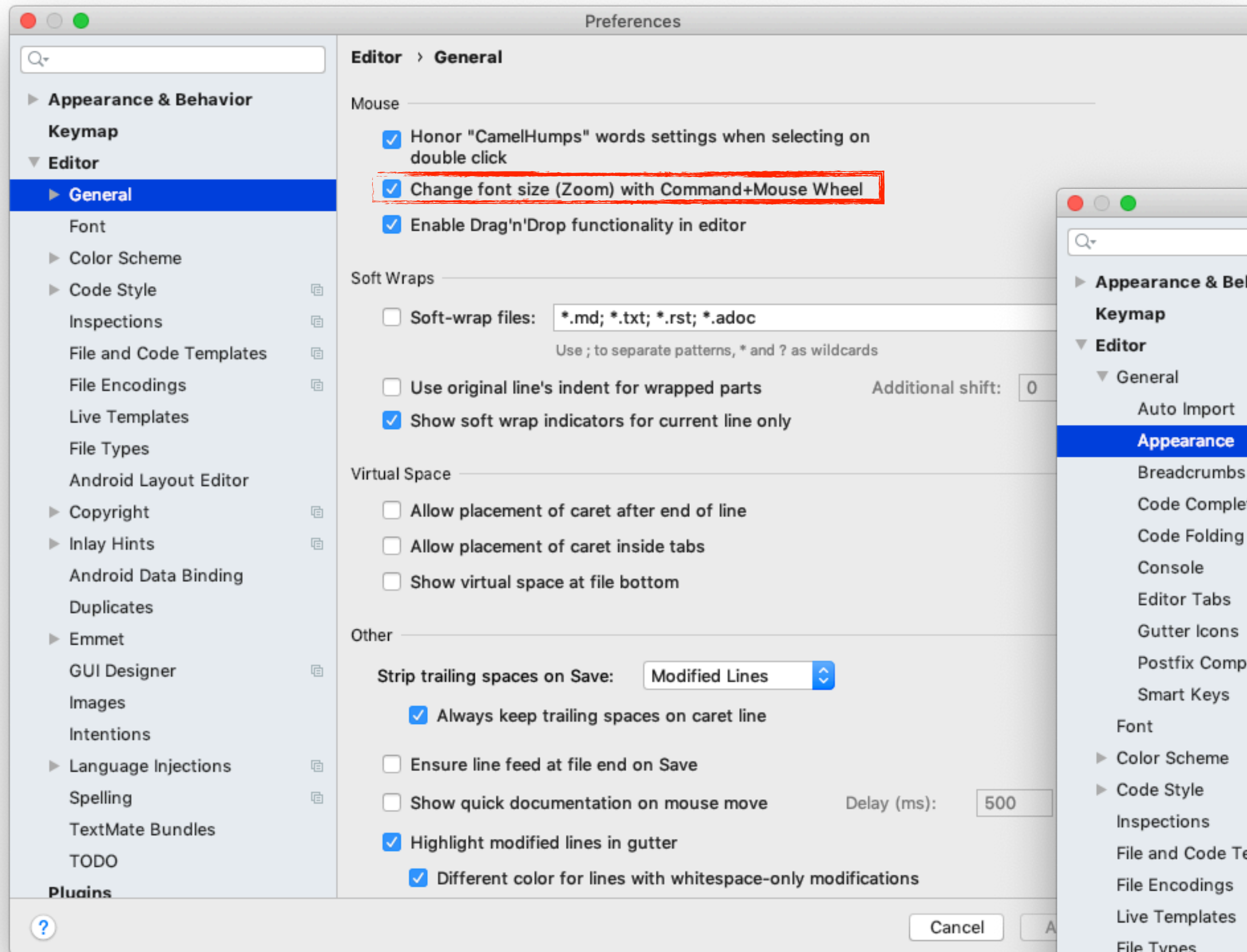
The screenshot shows the IntelliJ IDEA Project Structure dialog for a project named 'HelloWorld'. The 'Project SDK' is set to '11 (java version "11.0.2")'. The 'Project language level' is set to '11 - Local variable syntax for lambda parameters'. The 'Project compiler output' is also visible. In the bottom right, the 'Project Structure' tool window shows the 'SDKs' section with '11' selected. The 'JDK home path' is set to '/Library/Java/JavaVirtualMachines/jdk-11.0.2.jdk'. The 'Classpath' section shows the path to the JDK 11.0.2 content.

Kontrollieren in Preferences



Hier können die Hinweise während des Codens eingestellt werden





HelloWorld.java

```
package at.htl.helloworld;

import java.io.IOException;
import java.util.Scanner;

/**
 *
 * @author teacher
 */
public class HelloWorld {

    /**
     * @param args the command line arguments
     * @throws java.io.IOException
     */
    public static void main(String[] args) throws IOException {
        System.out.print("Please enter your name: ");

        Scanner s = new Scanner(System.in);
        String name = s.nextLine();

        boolean nameIsValid = name.length() > 0;

        if (nameIsValid)
            System.out.println("Hello " + name);
        else
            System.out.println("Hello World");
    }
}
```


Fakers

Ancient
Animal
Address
App
Artist
Avatar
Back To The Future
Aviation
Beer
Book
Bool
Business
ChuckNorris
Cat
Code
Color
Commerce
Company
Crypto
DateAndTime
Demographic
Dog
DragonBall
Dune
Educator
Esports
File
Finance
Food
Friends
FunnyName
GameOfThrones
Hacker
HarryPotter
Hipster
HitchhikersGuideToTheGalaxy
Hobbit
HowIMetYourMother
IdNumber
Internet
Job
LeagueOfLegends
Lebowski
LordOfTheRings
Lorem
Matz
Music
Name
Nation
Number
Options
Overwatch
PhoneNumber
Pokemon
PrincessBride
RelationshipTerms
RickAndMorty
Robin

Java-Faker

@Test

```
void javaFaker() {  
    Faker faker = new Faker();  
  
    String name = faker.name().fullName(); // Miss Samanta Schmidt  
    String firstName = faker.name().firstName(); // Emory  
    String lastName = faker.name().lastName(); // Barton  
  
    String streetAddress = faker.address().streetAddress(); // 60018 Sawayn Brooks Suite 449  
  
    System.out.println(name);  
    System.out.println(firstName);  
    System.out.println(lastName);  
    System.out.println(streetAddress);  
}
```

<https://github.com/DiUS/java-faker>

Tolle Library zum Erstellen
von Testdaten

Ed Gerlach
Kyla
Bartoletti
52085 Christian Radial

Ressourcen

- Ullenboom, C.; Java ist auch eine Insel, 14. Auflage, Rheinwerk Verlag Bonn 2019
- Ullenboom, C.; Java ist auch eine Insel, 12. Auflage, <http://openbook.rheinwerk-verlag.de/javainsel/> (frei verfügbar)
- Java Notes for Professionals book, <https://goalkicker.com/JavaBook/>, download am 26. August 2019 (frei verfügbar)
- Präsentationen der Oracle Academy (Java, SQL, PL/SQL, ...) <https://en.ppt-online.org/search?text=oracle&cat=&lang=en>
- Oracle Java Tutorial, <https://docs.oracle.com/javase/tutorial/>



Noch
Fragen?

HTL LE  NDING

Schön, hier zu lernen