

# Einführung in die Programmierung Übung

André Karge - Wintersemester 18/19

Intelligent  
Software  
Systems



# Übungsablauf

- Java Installation
- erstes Programm schreiben + kompilieren
- IDE installieren
- Java Projekt erstellen
- Programmstrukturen
- Datentypen / Zahlendarstellungen
- Organisatorisches

# Übungsleitung

M. Sc. André Karge

Büro: Digital Bauhaus Lab (DBL) Raum 308

Mail: [andre.karge@uni-weimar.de](mailto:andre.karge@uni-weimar.de)

Webseite:

<https://www.uni-weimar.de/de/medien/professuren/medieninformatik/intelligente-softwareysteme/lehre/>



# Java Installieren

## Windows:


- auf die Webseite von Oracle gehen, Java SDK 8 herunterladen und installieren

## Linux:

- mit dem Paketmanager installieren:
  - > `sudo apt update`
  - > `sudo apt install openjdk-8-jre openjdk-8-jdk openjdk-8-source`

# Java Installieren - Installation prüfen

Windows Konsole öffnen:

-  → cmd

Linux Terminal öffnen:

- Terminal öffnen

Java check:

- > java -version
- > javac -version

# Erstes Java Programm

- Editor öffnen
  - Notepad, Notepad++, Gedit, Visual Studio Code, Sublime Text, Atom, VIM, Emacs, ...
- HelloWorld.java Datei anlegen
- Quellcode schreiben

## *Live Demo*

```
2 public class HelloWorld {
3     public static void main(String[] args) {
4         System.out.println("Hello World");
5     }
6 }
```

# Erstes Java Programm - Kompilieren per Hand

- geschriebenes Programm muss in Bytecode übersetzt werden
- das geschieht mit dem Befehl “*javac*”

*live Demo*

```
→ firstProgram javac HelloWorld.java
```

- neue Datei: HelloWorld.class
- Ausführen von Java Programmen mittels “*java*”

*live Demo*

```
→ firstProgram java HelloWorld  
Hello World  
→ firstProgram
```

# Programmierwerkzeuge

- Integrated Development Environment (IDE)
  - Editor + Compiler + Debugger
- für Java:
  - Eclipse
  - IntelliJ
- <https://www.eclipse.org/>
- Eclipse herunterladen und **Eclipse IDE for Java Developer** installieren

*live Demo*



Quelle:  
<http://www.tutego.de/seminare/java-schulung/eclipse-seminar.html>



Quelle:  
<https://www.zoschke.com/intellij-idea>



# Eclipse


- neues Projekt erstellen
- neue Class-Datei in /src erstellen
- Quellcode schreiben

*live Demo*

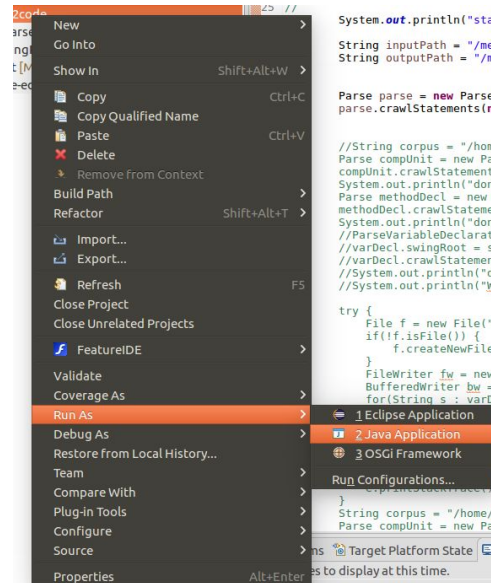


Quelle: <http://www.tutego.de/seminare/java-schulung/eclipse-seminar.html>

# Eclipse - Kompilieren

- 2 Wege:
  - Drücken des grünen Play-Buttons oben im Menü 
  - Rechtsklick auf das Auszuführende Projekt → Run As → Java Application

*live Demo*



# Programmstrukturen

- Jedes Java Programm muss eine Main-Methode implementieren um es ausführen zu können:

```
public class Main {  
    public static void main(String[] args) {  
  
    }  
}
```

- Dateien müssen den selben Namen haben wie die Klassen, die sie Implementieren:
  - Datei *HelloWorld.java* implementiert die Klasse *public class HelloWorld {}*
- Klassenbezeichner werden in der Regel **groß** geschrieben
- Methoden- und Variablenbezeichner werden in der Regel **klein** geschrieben

# Programmstrukturen

- Kommentare
  - Einzeilig mittels: //
  - Mehrzeilig mittels: /\* .... \*/
- Klassen und Methoden haben einen Body der mit geschweiften Klammern geöffnet und geschlossen wird: {}
- Methoden bestehen aus Anweisungen:

```
public class HelloWorld {  
    public static void main(String[] args) {  
        int a = 1;  
        int b = 2;  
        int c = a + b;  
        System.out.println(c);  
    }  
}
```

# Standard Typen

<b>Typ</b>	<b>Beschreibung</b>
byte	8bit Ganzzahl
short	16bit Ganzzahl
int	32bit Ganzzahl
long	64bit Ganzzahl
float	32bit Fließkommazahl
double	64bit Fließkommazahl

# Standard Typen

<b>Typ</b>	<b>Beschreibung</b>
char	16bit Unicode-Zeichen
boolean	1bit Wahrheitswert

# Ausgabe/Eingabe mit der Konsole

- Ausgabe wie schon vorhin gesehen:

```
System.out.println("Hallo");
```

- Eingabe:

*live Demo*

# Info

- In dieser Übung könnt ihr jederzeit Fragen stellen
- Erste Anlaufstelle, falls ihr irgendwo nicht weiter kommt ist das Buch “Java ist auch eine Insel”  
<http://openbook.rheinwerk-verlag.de/javainsel/>
- Tipp: lest das Kapitel 1-2 und versucht die Beispiele nachzubauen



# Teams

- wie schon in der Vorlesung
  - 2-3 Studenten pro Team
  - 8 Gruppen-Slots pro Übungstermin
- Merkt euch eure Teamnummer
- Gebt sie bei jeder Abgabe an

# Abgaben

- Dokumente als .pdf
- Quellcode als .java
- Angabe der Gruppennummer in allen abgegebenen Dateien
  - im Quellcode als Kommentar am Anfang der Datei
- Wenn mehrere Dateien:
  - Komprimieren! (Als zip, rar, tar.gz, etc)
- Benennt dieses Archiv nach folgendem Schema:

abgabe\_programmierung\_uebungXX\_gruppeXX.zip

- Schickt eure Lösungen vor! Ablauf der Frist an [andre.karge@uni-weimar.de](mailto:andre.karge@uni-weimar.de)

# Nächste Woche

- Besprechung des ersten Belegs
- weitere Grundlagen Datentypen, Anweisungen und Methoden

Fragen?