

Java-Crashkurs

0. Einführung | Benno Hölz, Matthias Ruf | 05.09.2023



1. Kurs - Allgemein

Grundinformationen

Grundinformationen zum Kurs

- 6 Termine: Sa 26.10.19 Sa 30.11.19
- 10 16 Uhr
- 5 Themen

Ziel

- Einführung in Java
- Vermittlung der Grundlagen von Programmiersprachen

Inhalt

Aufgeteilt in 4 Themengebiete:

- 1. Datentypen (heute)
- 2. Schleifen
- 3. Objektorientierung
- 4. Methoden

Grundaufbau

Jeder Tag ist in 3 Teile gegliedert:

- Präsentation (ca. 1:30 h)
- Übungsphase (ca. 1:30 h)
- Nachbesprechung (ca. 0:45 h)

2. Java - Allgemein

Java - Heute

Heute hat sich Java als eine der wichtigsten Programmiersprachen durchgesetzt:

- über 9 Mio. hauptberufliche Softwareentwickler
- über 3 Milliarden javafähige Geräte
- über 10 Mio Downloads der Laufzeitumgebung pro Woche

Zahlen von Oracle

Java - Vorteile

Warum Java?

- einfaches objektorientiertes Programmieren
 - besonders geeignet für Applicationen
- viele Geräte/Betriebssysteme unterstützen Java
 - Stichwort: Plattformunabhängig

3. JDK & JRE

JDK & JRE



JDK & JRE

JDK (Java Development Kit)

- Compiler
- wandelt Code in Bytecode-Datei

JRE (Java Runtime Enviroment)

- Laufzeitumgebung
- Interpretation des Bytecode

4. Installation (Windows)

Installation - JDK

Installation:

- 1. Download des JDK & JRE
- 2. Installation des JDK & JRE
- 3. Eintragen des JDK-Path in die Umgebungsvariablen

Installation - JDK (Umgebungsvariable)

Festlegen der Umgebungsvariable:

- 1. Windows Suchleiste öffnen, "Systemumgebungsvariablen bearbeiten" öffnen.
- 2. Auf Reiter "Erweitert" navigieren (siehe Abb. 16)
- 3. In der Box Systemvariablen, die Variable "Path" auswählen. Bearbeiten klicken. (siehe Folie 17)
- 4. Im Fenster (Folie 18) auf "Neu" klicken.
- 5. Pfad zum bin-Verzeichnis als Umgebungsvariable eintippen.

Achtung der Pfad zum bin-Verzeichnis wird vermutlich etwas anderst ausssehen wie in der Abb. auf Folie 19.

Installation - JDK & JRE (Umgebungsvariablen)

Systemeigenschaften				\times
Computername Hardwar	e Erweitert	Computerschutz	Remote	
Sie müssen als Administ durchführen zu können. Leistung	rator angemel	det sein, um diese	Ånderungen	
Visuelle Effekte, Proze Speicher	ssorzeitplanur	ıg, Speichemutzun	ig und virtueller	
			Einstellungen	
Benutzerprofile				
Desktopeinstellungen	bezüglich der	Anmeldung		
			Einstellungen	
Starten und Wiederher	stellen			
Systemstart, Systemfel	nler und Debu	ginformationen		
			Einstellungen	
		Umge	ebungsvariablen	
	C	K Abbree	chen Überneh	nmen

Installation - JDK (Umgebungsvariablen)

Variable	Wert	
OneDrive	C:\Users\har10\OneDrive	
Path	C:\Program Files\Java\jdk1.8.0 191\bin:C:\Program Files\Java\jre1.8	
TEMP	C:\Users\har10\AppData\Local\Temp	
тмр	C:\Users\har10\AppData\Local\Temp	
stemvariahlen	Neu Bearbeiten Löschen	
stemvariablen Variable	Neu Bearbeiten Löschen	
stemvariablen Variable OS	Neu Bearbeiten Löschen Wert Windows NT	
stemvariablen Variable OS Path	Neu Bearbeiten Löschen Wert Windows, NT C.S/Pogram Files (x89)/Common Files/Oracle/Java/javapath; C.\W/N	
stemvariablen Variable OS Path PATHEXT	Neu Bearbeiten Löschen Wert Windows, NT Clifbragram Files (x66)\Common Files\OracleUsevajjavapath; Cl\WIN Clifbragram Files (x66)\Common Files\OracleUsevajjavapath; Cl\WIN COMP, EXE, BAT, CMD, VBS, VBE, JS, JSE, WSF, WSF, MAC Code, EXE, BAT, CMD, VBS, VBE, JS, JSE, WSF, WSF, MAC	
stemvariablen Variable OS Path PATHEXT PROCESSOR_ARCHITECTURE	Neu Bearbeiten Löschen Wert Windows, NT C.\Program Files\Oracle\Java\javapath_C\WIN C.\Vorgram Files\Oracle\Java\javapath_C\WIN C.OME, EXE, BAT; CMD; VBS; VBE; JS; JSE; WSF; WSF; MSC AMD64 AMD64 AMD64	
stemvariablen Variable OS Path PATHEXT PROCESSOR_ARCHITECTURE PROCESSOR_IDENTIFIER	Neu Bearbeiten Löschen Wert Windows, NT COM; DKE; BAT); CMD; VBS; VBE; US; VB	
stemvariablen Variable OS Path PATHEXT PROCESSOR_ARCHITECTURE PROCESSOR_LEVEL	Neu Bearbeiten Löschen Wert Windows, NT Clifbragam Files (x66)\Common Files\Olascle'Java'javapath, Cl./WIN COM-EXE, BAT, CMD, VBS, VBE, JS, JSE, WSF, WSH, MSC AMD64 Intel64 Family 6 Model 38 Stepping 9, GenuineIntel 6	
sternvariablen OS Path PATHEXT PROCESSOR_ARCHITECTURE PROCESSOR_UEVEL PROCESSOR_UEVEL PROCESSOR_UEVEL	Neu Bearbeiten Löschen Wert Windows, NT C/Wrogram Files (0/socie/Java/javapath; C/W/N C/Wrogram Files (0/socie/Java/javapath; C/W/N C/OME, EXE, BAT, CMD, VBS, VBE, JS, JSE, WSF, WSH, MSC AMD64 Intel64 Family 6 Model 38 Stepping 9, GenuineIntel 6 3a09 3a09 Stepping 9, GenuineIntel 5 5	

Installation - JDK (Umgebungsvariablen)

Umgebungsvariable bearbeiten	×
C:\Program Files (x86)\Common Files\Oracle\Java\javapa	th Neu
%SystemRoot% %SystemRoot%	Bearbeiten
%SystemRoot%\System32\Wbem %SYSTEMROOT%\System32\WindowsPowerShell\v1.0\	Durchsuchen
C:\Program Files (x86)\Windows Live\Shared	Löschen
	Nach oben
	Nach unten
	Text bearbeiten
	OK Abbrechen

Installation - JDK (Umgebungsvariblen)

Umgebungsvariable bearbeiten	×
C:\Program Files (x86)\Common Files\Oracle\Java\javapath	Neu
%SystemRoot%\system32	
%SystemRoot%	Bearbeiten
%SystemRoot%\System32\Wbem	
%SYSTEMROOT%\System32\WindowsPowerShell\v1.0\	Durchsuchen
C:\Program Files\MiKTeX 2.9\miktex\bin\x64\	
C:\Program Files (x86)\Windows Live\Shared	Löschen
%SYSTEMROOT%\System32\OpenSSH\	
C:\Program Files\Java\jdk1.8.0_191\bin	
	Nach oben
	Nach unten
	Text bearbeiten
	OK Abbrechen
L	Abbrechen

Installation - Notepad++ (Windows)

Notepad++:

- Texteditor (unterstützt Syntax zahlreicher Programmiersprachen)
- für Windows-User
- Installation selbsterklärend

Downloadpage von Notepad++: https://notepad-plus-plus.org/download/v7.6.4.html

5. Übersicht - Java-Programme

Übersicht

Zu Beginn versuchen wir zu verstehen wie wir Java-Programmcode...

- 1. abspeichern
- 2. bearbeiten
- 3. kompilieren
- 4. ausführen

Java-Code abspeichern

- Wir verwenden Notepad++ zum bearbeiten von Java-Code
- Erstellen eines *.java Datei funktioniert durch simples Abspeichern einer Datei (in Notepad++) mit der Dateiendung . java.
- In dieser Datei steht später der Quellcode.

Quellcode ist vom Menschen lesbar von der Maschine nicht.

Java-Code bearbeiten / Hello World

Wie alle Programmiersprachen führen auch wir Java mit einem Hello World Programm ein:

```
1 public class HalloWelt{
2     public static void main(String[] args){
3         System.out.println("Hallo Welt");
4     }
5 }
```

Dieses Programm gibt "Hallo Welt" aus.

Java-Code bearbeiten

Klasse

Konvention

- Nur eine Klasse pro *.java Datei
- Klassenname entspricht dem Dateiname, sprich <Klassenname>.java enthält die Klasse <Klassenname>.

Java-Code bearbeiten

main-Methode

Wichtig

- In der main-Methode wird das Programm gestartet.
- Wir werden in den ersten 2 Kapitel nur in dieser Methode unseren Code schreiben.
- Die Begriffe "public", "static", "void", sowie "String" werden später erst erklärt und dürfen erstmal so hingenommen werden.

Compilieren & ausführen



6. Einführung Linux-Shell/PowerShell

Linux-Shell/PowerShell

Linux-Shell

- zur Eingabe von Befehlen
- nur auf Linux verfügbar (mittlerweile auch unter Windows als Subsystem)

PowerShell

- zur Eingabe von Befehlen
- nur auf Windows verfügbar

Unterschiede

- Syntax
- Befehlssatz

Windows PowerShell



Verzeichnis wechseln

• Mit dem Befehl "cd "kann das Verzeichnis gewechselt werden.



weitere Befehle

Für uns wichtige Befehle sind:

Befehl	Bedeutung	
cd	Verzeichnis wechseln	
dir (oder ls)	listet den Inhalt des aktuellen Verzeichisses auf	
javac <datei>.java</datei>	Java-Quellcode kompilieren	
java <datei></datei>	Java-Bytecode interpretieren & ausführen	
help <befehl></befehl>	gibt eine Befehlsbeschreibung aus	

Befehle funktionieren sowohl unter Linux als auch unter Windows, einzig "Is" funktioniert nur unter Linux

Java-Code ausführen



7. Aufgaben

Das Ziel dieser Übung ist es, ein funktionierendes Programm zu erstellen, kompilieren und auszuführen.

Aufgabe 1: Erstellen des Quellcodes

10 Minuten

- 1. Öffne Notepad++ (oder Geany auf Ubuntu)
- 2. Lege eine neue Datei an und speichere sie als HalloWelt.java ab.
- 3. Füge den unten stehenden Quellcode in das Dokument ein.
- 4. Speichere die Datei nochmals ab.

```
1 public class HalloWelt{
2     public static void main(String[] args){
3         System.out.println("Hallo Welt");
4     }
5 }
```

Aufgabe 2: Kompilieren und Ausführen

10 Minuten

- 1. Öffne die PowerShell (oder das Terminal auf Linux)
- 2. Wechsle in das Verzeichnis in dem die HalloWelt.java-Datei aus Aufgabe 1 liegt.
 - → Tipp: Nutze den Befehl cd
- 3. Kompiliere die Datei
 - → Tipp: Nutze den Befehl javac
- Lasse dir den Inhalt des Verzeichnisses ausgeben. Was fällt auf?
 → Tipp: Nutze den Befehl dir (oder 1s auf Linux)
- 5. Führe die Datei aus.
 - → Tipp: Nutze den Befehl java

Aufgabe 3: Quellcode bearbeiten

5 Minuten

- 1. Ändere den Quellcode ab, sodass er deinen Namen ausgibt.
- 2. Ändere den Klassenname ab, sodass er deinem Namen entspricht.

Zusammenfassung

- Programme schreiben wir vorerst nur in der main()-Methode
- System.out.println(<irgendwas>) ist für Ausgaben über die Kommandozeile zuständig
- PowerShell-Befehle:
 - $cd \rightarrow Verzeichnis wechseln$
 - javac <Dateiname>.java \rightarrow kompilieren
 - java <Dateiname> \rightarrow ausführen