

Hello World

Javakurs 2014, 1. Vorlesung

Sebastian Schuck

basierend auf der Vorlage von
Arne Kappen

wiki.freitagrunde.org

3. März 2014



This work is licensed under the *Creative Commons Attribution-ShareAlike 3.0 License*.

Inhaltsverzeichnis

- 1 Was ist Java?
- 2 Programmieren in Java
 - Werkzeuge
 - Syntax
 - wichtige Datentypen
 - Variablen in Java
 - bedingte Anweisungen



4!

Inhaltsverzeichnis

- 1 Was ist Java?
- 2 Programmieren in Java
 - Werkzeuge
 - Syntax
 - wichtige Datentypen
 - Variablen in Java
 - bedingte Anweisungen



4!

Was ist Java?

- entwickelt von Sun Microsystems (seit 2010 Oracle Corporation)
- relativ junge Programmiersprache
 - **1990** entwickelt, ursprünglich geplante Verwendung für eingebettete Systeme
 - **1996** Version 1.0 wird veröffentlicht
 - ...
 - seit **Mitte 2011**: momentan aktuelle Version, **Java JSE 7**
 - voraussichtlich **18. März 2014**: neue Version **Java JSE 8**

Ziele

- Plattformunabhängigkeit:
JVM - **JavaVirtualMachine**
- einfache, objektorientierte und vertraute Programmiersprache
- robust und sicher
- ...

Vor- und Nachteile

Vorteile

- schneller Einstieg
- umfangreiche Bibliotheken
- streng Typisiert
- Java kümmert sich um Hardware und Speicherverwaltung selbst

Nachteile

- wenig tiefgreifende Kontrolle
- relativ unperformant

Literaturbeispiel

- **Christian Ullenboom** - Java ist auch eine Insel



<http://openbook.galileocomputing.de/javainsel/>

Inhaltsverzeichnis

- 1 Was ist Java?
- 2 Programmieren in Java
 - Werkzeuge
 - Syntax
 - wichtige Datentypen
 - Variablen in Java
 - bedingte Anweisungen



4!

Werkzeuge

(vermutlich) meistbenutzte Entwicklungsumgebungen

- Eclipse - <https://www.eclipse.org/>
- Netbeans - <https://netbeans.org/>

ABER

- zum Erlernen der Programmiersprache wird vorerst das Programmieren mit Hilfe des Texteditors und der Konsole empfohlen
- bei späteren größeren Projekten ist das Nutzen einer Entwicklungsumgebung sinnvoll

Inhaltsverzeichnis

1 Was ist Java?

2 Programmieren in Java

- Werkzeuge
- **Syntax**
- wichtige Datentypen
- Variablen in Java
- bedingte Anweisungen



4!

grundlegende Infos ...

... zur Syntax

- Blöcke werden mit `{ ... }` umschlossen
- Anweisungen werden mit `;` abgeschlossen
- Leerzeichen und Einrückungen werden ignoriert
- Groß- und Kleinschreibung wird unterschieden
- einzeiliger Kommentar mit `//`
- mehrzeiliger Kommentar mit `/* ... */`

... zu den Code-Konventionen

- Klassennamen werden großgeschrieben
- Methoden- und Variablennamen werden klein geschrieben
- zusammengesetzte Wörter in `CamelCaseSchreibweise`
- Einrückungen für eine bessere Lesbarkeit

grundlegender Aufbau eines Programms

Aufbau von HelloWorld.java

```
1 // Klassenname muss dem Dateinamen entsprechen
2 public class HelloWorld {
3
4     // main-Methode ist Einsprungpunkt in das Programm
5     public static void main(String[] args) {
6
7         // hier steht, was das Programm machen soll
8         System.out.println("Hello World");
9     }
10 }
```

Wichtig:

- Dateiname $\hat{=}$ Klassenname.java (hier: HelloWorld.java)
- Programm **compilieren**: javac HelloWorld.java
- Programm **ausführen**: java HelloWorld

Inhaltsverzeichnis

1 Was ist Java?

2 Programmieren in Java

- Werkzeuge
- Syntax
- **wichtige Datentypen**
- Variablen in Java
- bedingte Anweisungen



4!

Typ	Größe	Wertebereich	Beispiel
Ganze Zahlen			
byte	8 bit	$-128 \dots 127$	0
short	16 bit	$-32768 \dots 32767$	1234
int	32 bit	$\sim -2 \cdot 10^9 \dots 2 \cdot 10^9$	12345
long	64 bit	$\sim 10^{19} \dots 10^{19}$	123456L
Gleitkommazahlen			
float	32 bit	$\sim \pm 10^{37}$	1.5f
double	64 bit	$\sim \pm 10^{307}$	1.5
Wahrheitswert			
boolean	8bit	{true, false}	true
Zeichen			
char	16 bit	Unicode Zeichen	'c'
Zeichenkette			
String			„Hello World“

Inhaltsverzeichnis

1 Was ist Java?

2 Programmieren in Java

- Werkzeuge
- Syntax
- wichtige Datentypen
- **Variablen in Java**
- bedingte Anweisungen



4!

Variablen

Deklaration und Zuweisung:

- Deklaration einer Variablen:
 ⟨Datentyp⟩⟨Bezeichner⟩; **int** zahl;
- Wertzuweisung:
 ⟨Bezeichner⟩ = ⟨Wert⟩; zahl = 42;
- beides in einer Zeile:
 ⟨Datentyp⟩⟨Bezeichner⟩ = ⟨Wert⟩; **int** zahl = 42;

Wichtig

- Java ist streng typisiert: der Typ des Wertes muss zum Typen der Variablen passen!
 - Das Speichern eines int-Wertes in einer double-Variablen ist möglich.
 - Der Typ wird automatisch umgewandelt.
 - ABER: welcher Wert steht in d? **double** d = 3/2;

numerische Operatoren

+	Addition
-	Subtraktion
*	Multiplikation
/	Division
%	Modulo (Rest der Division)
++	Inkrement
+=	Inkrement und Zuweisung
--	Dekrement
-=	Dekrement und Zuweisung
(...)	zusätzlich: Klammerung

logische Operatoren

==	Gleichheit
!=	Ungleichheit
<	kleiner als
<=	kleiner gleich
>	größer als
>=	größer gleich
!	nicht
&&	bedingtes UND
	bedingtes ODER

Das Ergebnis logischer Operatoren ist immer ein **boolean**-Wert.

Sonderfall: String (Erklärung später)

einString.equals(andererString) – Gleichheit

!einString.equals(andererString) – Ungleichheit

Inhaltsverzeichnis

- 1 Was ist Java?
- 2 Programmieren in Java
 - Werkzeuge
 - Syntax
 - wichtige Datentypen
 - Variablen in Java
 - **bedingte Anweisungen**



4!

bedingte Anweisung

bedingte Anweisung

```
if(condition){  
    // Anweisungen werden ausgeführt, wenn condition true  
} else {  
    // Anweisungen werden ausgeführt, wenn condition false  
}
```

condition

- muss ein boolescher Ausdruck sein
- kann mit Hilfe der booleschen Operatoren gebildet werden

if- bzw. else-Block

- kann beliebig viele Anweisungen enthalten
- wenn der jeweilige Block durch { ... } umschlossen wird

Beispielaufgabe: HelloJavaKurs.java

Aufbau von HelloJavaKurs.java

```
1 // HelloWorld mit bedingter Anweisung
2 public class HelloJavaKurs {
3
4     public static void main(String[] args) {
5         // Anzahl der Teilnehmer in der ersten Vorlesung
6         // wird zur Demonstration variiert
7         int teilnehmer = 0;
8
9         // Begrueßung abhaengig von der Anzahl der Teilnehmer
10        if (teilnehmer < 0){
11            System.err.println(" Eine negative Teilnehmerzahl... Ja klar!");
12        } else if (teilnehmer == 0) {
13            System.out.println(" Hallo einsamer Basti!");
14        } else if (teilnehmer == 1){
15            System.out.println(" Hallo Du JavaKurs-Teilnehmer!");
16        } else {
17            System.out.println(" Hallo Ihr JavaKurs-Teilnehmer!");
18        }
19    }
20 }
```