

Programmiervorkurs

Einführung in Java Tag 1



- Ablauf
- Interne Zahlendarstellung
- Variablen
- Datentypen
- Wertebereiche
- Variablennamen
- Verwendung
 - Deklaration und Wertzuweisung
 - Ausgabe
- Arithmetik
 - Modularechnung
 - In- bzw. Dekrement
 - Integer-Division
 - Verknüpfung von Zeichenketten
- Konvertierung
- Kommentare
- Technologie

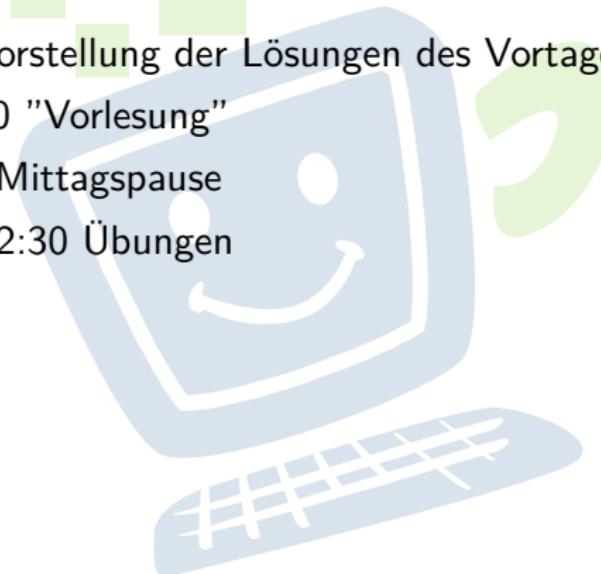
Inhaltsübersicht Vorkurs

- ▶ Tag 1: Zustände, Variablen, Datentypen, Konvertierungen, Arithmetik, Eclipse Livedemo
- ▶ Tag 2: Kommentare, Boolesche Ausdrücke, If-Abfragen, Switch-Case
- ▶ Tag 3: Arrays, (Do-)While-Schleife, For-Schleifen, Weiterführung Debugging
- ▶ Tag 4: (statische) Methoden, KlassenvARIABLEN

Ablauf
Interne Zahlendarstellung
Variablen
Datentypen
Wertebereiche
Variablennamen
Verwendung
Deklaration und Wertzuweisung
Ausgabe
Arithmetik
Modulorechnung
In- bzw. Dekrement
Integer-Division
Verknüpfung von Zeichenketten
Konvertierung
Kommentare
Technologie

Ablauf

- ▶ 09:30 Vorstellung der Lösungen des Vortages
- ▶ ab 10:00 "Vorlesung"
- ▶ 60 min Mittagspause
- ▶ gegen 12:30 Übungen



| Ablauf |
|----------------------------------|
| Interne Zahlendarstellung |
| Variablen |
| Datentypen |
| Wertebereiche |
| Variablennamen |
| Verwendung |
| Deklaration und Wertzuweisung |
| Ausgabe |
| Arithmetik |
| Modularechnung |
| In- bzw. Dekrement |
| Integer-Division |
| Verknüpfung von Zeichenketten |
| Konvertierung |
| Kommentare |
| Technologie |

Zahlen in Binärdarstellung

- ▶ **beware of the: "0"** (Ja, sie ist auch eine Zahl)
- ▶ Dual pro Stelle 2 Zustände
 - ▶ Dezimal pro Stelle 10 Zustände
- ▶ Jede weitere Stelle verdoppelt die Bisher mögliche Anzahl verschiedener Zustände
- ▶ Im Dezimalsystem liegt eine Verzehnfachung vor
 - ▶ $10 = 1 * 10$
 - ▶ $100 = 10 * 10$
 - ▶ $1000 = 100 * \dots$

Ablauf

Interne
Zahlendarstellung

Variablen

Datentypen

Wertebereiche

Variablennamen

Verwendung

Deklaration und
Wertzuweisung

Ausgabe

Arithmetik

Modulorechnung

In- bzw. Dekrement

Integer-Division

Verknüpfung von
Zeichenketten

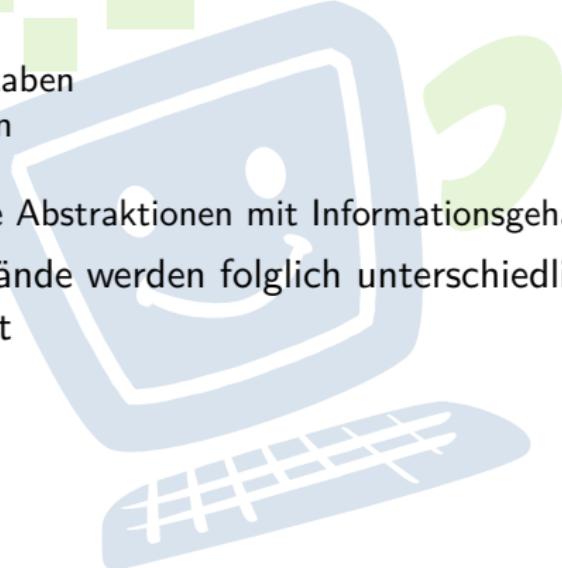
Konvertierung

Kommentare

Technologie

Was haben Zustände mit Zahlen zu tun?

- ▶ Ein Computer muss nicht nur zahlen darstellen können
 - ▶ Buchstaben
 - ▶ Zeichen
 - ▶ Bilder
 - ▶ Höhere Abstraktionen mit Informationsgehalt
- ▶ Diese Zustände werden folglich unterschiedlich interpretiert



Ablauf

Interne
Zahlendarstellung

Variablen

Datentypen
Wertebereiche

Variablennamen

Verwendung

Deklaration und
Wertzuweisung
Auszabe

Arithmetik

Modularechnung
In- bzw. Dekrement
Integer-Division
Verknüpfung von
Zeichenketten

Konvertierung

Kommentare

Technologie

ASCII Tabelle

- ▶ Siehe Anhang



Ablauf

Interne
Zahlendarstellung

Variablen

Datentypen
Wertebereiche

Variablennamen

Verwendung

Deklaration und
Wertzuweisung
Ausgabe

Arithmetik

Modularechnung
In- bzw. Dekrement
Integer-Division
Verknüpfung von
Zeichenketten

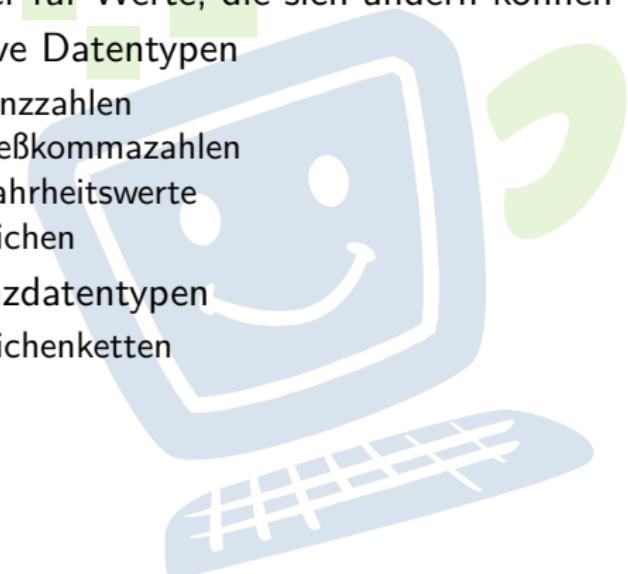
Konvertierung

Kommentare

Technologie

Variablen

- ▶ Speicher für Werte, die sich ändern können
- ▶ Primitive Datentypen
 - ▶ Ganzzahlen
 - ▶ Fließkommazahlen
 - ▶ Wahrheitswerte
 - ▶ Zeichen
- ▶ Referenzdatentypen
 - ▶ Zeichenketten



Ablauf

Interne
Zahlendarstellung

Variablen

Datentypen
Wertebereiche

Variablennamen

Verwendung
Deklaration und
Wertzuweisung
Auszabe

Arithmetik

Modulorechnung
In- bzw. Dekrement
Integer-Division
Verknüpfung von
Zeichenketten

Konvertierung

Kommentare

Technologie

Datentypen

- ▶ Zahlen im Computer sind endlich
- ▶ Ganzzahlen
 - ▶ **byte** (8 Bit / 1 Byte)
 - ▶ **short** (16 Bit / 2 Byte)
 - ▶ **int** (32 Bit / 4 Byte)
 - ▶ **long** (64 Bit / 8 Byte)
- ▶ Kommazahlen (Fließkommazahlen)
 - ▶ **float** (32 Bit / 4 Byte)
 - ▶ **double** (64 Bit / 8 Byte)
- ▶ Unterscheiden sich jeweils nur in ihrem Wertebereich

Ablauf

Interne
Zahlendarstellung

Variablen

Datentypen

Wertebereiche

Variablennamen

Verwendung

Deklaration und
Wertzuweisung

Ausgabe

Arithmetik

Modularechnung
In- bzw. Dekrement
Integer-Division
Verknüpfung von
Zeichenketten

Konvertierung

Kommentare

Technologie

Datentypen

- ▶ Wahrheitswerte
 - ▶ **boolean**
 - ▶ **true** oder **false**
- ▶ 1 Zeichen (keine Zeichenkette)
 - ▶ **char**
 - ▶ 2 Byte lang
 - ▶ Darstellung als 8-Bit ASCII Zeichen
 - ▶ Darstellung als 16-Bit-Unicode-Wert
- ▶ Zeichenketten
 - ▶ **String**
 - ▶ Referenzdatentyp

Ablauf

Interne
Zahlendarstellung

Variablen

Datentypen

Wertebereiche

Variablennamen

Verwendung

Deklaration und
Wertzuweisung

Ausgabe

Arithmetik

Modulorechnung
In- bzw. Dekrement

Integer-Division

Verknüpfung von
Zeichenketten

Konvertierung

Kommentare

Technologie

Variablen - Wertebereiche

| Type | Länge | | Wertebereich |
|---------|-------|-----|---------------------------------------|
| | Byte | Bit | |
| boolean | - | - | true oder false |
| char | 2 | 16 | Unicode Zeichen |
| byte | 1 | 8 | -128 bis 127 |
| short | 2 | 16 | -32768 bis 32767 |
| int | 4 | 32 | -2.147.483.648 bis 2.147.483.647 |
| long | 8 | 64 | -2^{63} bis $2^{63} - 1$ |
| float | 4 | 32 | $\pm 1,4E - 45$ bis $\pm 3,4E + 38$ |
| double | 8 | 64 | $\pm 4,9E - 324$ bis $\pm 1,7E + 308$ |

Ablauf

Interne
Zahlendarstellung

Variablen

Datentypen

Wertebereiche

Variablennamen

Verwendung

Deklaration und
Wertzuweisung

Ausgabe

Arithmetik

Modulorechnung
In- bzw. Dekrement
Integer-Division
Verknüpfung von
Zeichenketten

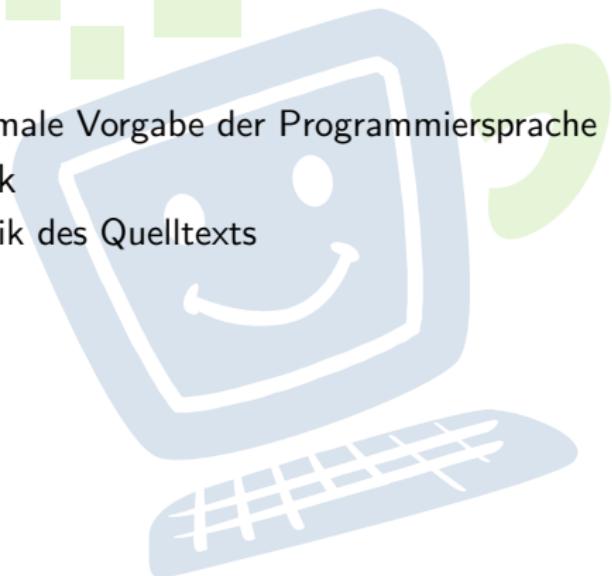
Konvertierung

Kommentare

Technologie

Syntax / Semantik

- ▶ Syntax
 - ▶ Formale Vorgabe der Programmiersprache
- ▶ Semantik
 - ▶ Logik des Quelltextes



Ablauf

Interne
Zahlendarstellung

Variablen

Datentypen

Wertebereiche

Variablennamen

Verwendung

Deklaration und
Wertzuweisung

Ausgabe

Arithmetik

Modularechnung
In- bzw. Dekrement
Integer-Division
Verknüpfung von
Zeichenketten

Konvertierung

Kommentare

Technologie

Variablennamen (Bezeichner)

- ▶ Vorgaben
 - ▶ So MÜSSEN Namen sein, sonst gibt es Compiler-Fehler
 - ▶ Erlaubte Zeichen: Buchstaben, Zahlen und _
 - ▶ Erstes Zeichen darf keine Zahl sein
- ▶ Gesperrte Namen
 - ▶ z.B. **true, false, new**

Ablauf

Interne
Zahlendarstellung

Variablen

Datentypen
Wertebereiche

Variablennamen

Verwendung

Deklaration und
Wertzuweisung
Auszabe

Arithmetik

Modulorechnung
In- bzw. Dekrement
Integer-Division
Verknüpfung von
Zeichenketten

Konvertierung

Kommentare

Technologie

Variablenamen

► Konventionen

- ▶ So SOLLTEN Namen sein, jedoch kompiliert der Quelltext sobald die Formalen Voraussetzungen erfüllt sind
- ▶ sinnvolle, aussagekräftige Namen wählen
- ▶ keine Abkürzungen
- ▶ Substantive
- ▶ Nur lateinische Zeichen, Zahlen und _
 - ▶ KEIN ä, ö, ü, ß, ...
- ▶ Verwendung EINER Sprache, kein Gemisch oder "**Denglisch**"
- ▶ lowerCamelCase-Schreibweise
 - camelCase bedeutet, dass der Bezeichner ohne Trennzeichen wie Leerzeile und Unterstrich angegeben werden und die folgenden Worte groß geschrieben werden.
Die lower Variante beginnt das erste Wort klein geschrieben.

Ablauf

Interne
Zahlendarstellung

Variablen

Datentypen
Wertebereiche

Variablenamen

Verwendung

Deklaration und
Wertzuweisung
Auszabe

Arithmetik

Modulorechnung
In- bzw. Dekrement
Integer-Division
Verknüpfung von
Zeichenketten

Konvertierung

Kommentare

Technologie

zur Verwendung



Ablauf

Interne
Zahlendarstellung

Variablen

Datentypen
Wertebereiche

Variablennamen

Verwendung

Deklaration und
Wertzuweisung
Ausgabe

Arithmetik

Modularechnung
In- bzw. Dekrement
Integer-Division
Verknüpfung von
Zeichenketten

Konvertierung

Kommentare

Technologie

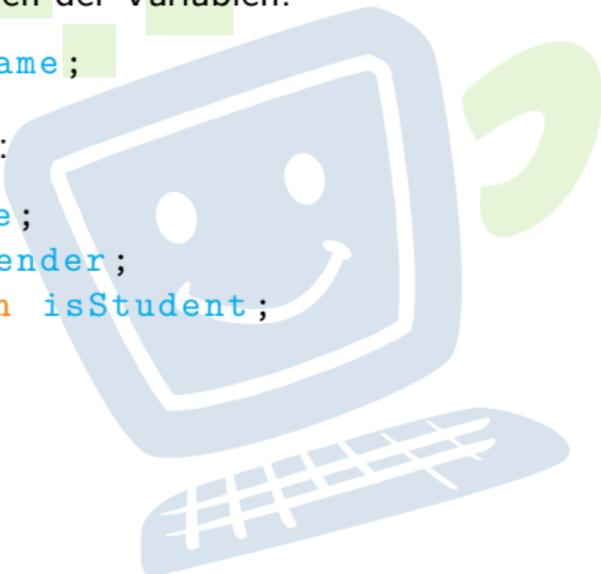
Deklaration

- Deklarieren der Variablen:

```
type name;
```

- Beispiele:

```
int age;  
char gender;  
boolean isStudent;
```



Ablauf

Interne
Zahlendarstellung

Variablen

Datentypen
Wertebereiche

Variablennamen

Verwendung

Deklaration und
Wertzuweisung

Ausgabe

Arithmetik

Modularechnung
In- bzw. Dekrement
Integer-Division
Verknüpfung von
Zeichenketten

Konvertierung

Kommentare

Technologie

Wertzuweisung

- Der Variablen einen Wert zuweisen
Beim ersten Mal spricht man von initialisieren

```
name = wert;
```

- Die Variable muss deklariert worden sein
- Beispiele:

```
int age; age = 20;  
float balance; balance = 4.2f;  
char gender; gender = 'm';  
String name; name = "Douglas Adams";
```

Ablauf

Interne
Zahlendarstellung

Variablen

Datentypen

Wertebereiche

Variablenamen

Verwendung

Deklaration und
Wertzuweisung

Ausgabe

Arithmetik

Modulorechnung

In- bzw. Dekrement

Integer-Division

Verknüpfung von
Zeichenketten

Konvertierung

Kommentare

Technologie

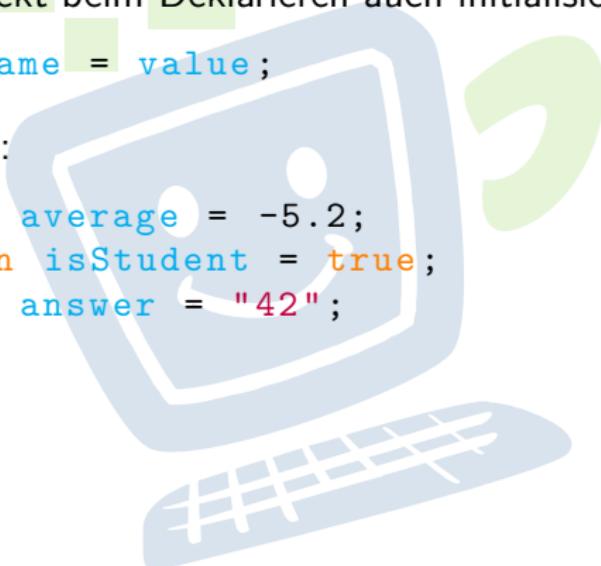
Initialisierung

- Wert direkt beim Deklarieren auch initialisieren:

```
type name = value;
```

- Beispiele:

```
double average = -5.2;
boolean isStudent = true;
String answer = "42";
```



Ablauf

Interne
Zahlendarstellung

Variablen

Datentypen
Wertebereiche

Variablennamen

Verwendung

Deklaration und
Wertzuweisung

Ausgabe

Arithmetik

Modularechnung
In- bzw. Dekrement
Integer-Division
Verknüpfung von
Zeichenketten

Konvertierung

Kommentare

Technologie

Ausgabe

- Sonst würde es nachher ziemlich langweilig

```
System.out.println(ausgabe);  
System.out.print(ausgabe);
```

- Beispiele:

```
System.out.println("Hallo\u00d7Welt");
```

```
String name = "Welt";  
System.out.print("Hallo\u00d7");  
System.out.print(name);  
System.out.println();
```

Ablauf

Interne
Zahlendarstellung

Variablen

Datentypen
Wertebereiche

Variablenamen

Verwendung

Deklaration und
Wertzuweisung

Ausgabe

Arithmetik

Modulorechnung
In- bzw. Dekrement
Integer-Division
Verknüpfung von
Zeichenketten

Konvertierung

Kommentare

Technologie

Arithmetik

| Bezeichnung | Operator | Anwendung |
|----------------|----------|-----------|
| Addition | + | $a + b$ |
| Subtraktion | - | $a - b$ |
| Multiplikation | * | $a * b$ |
| Division | / | a / b |
| Inkrement | ++ | $a++$ |
| Dekrement | -- | $a--$ |
| Modulo | % | $a \% b$ |

Ergebnis kann Variablen zugewiesen werden:

```
int result = 5 + 2;  
double division = 3.5 / (result - 1);
```

Ablauf

Interne
Zahlendarstellung

Variablen

Datentypen
Wertebereiche

Variablenamen

Verwendung

Deklaration und
Wertzuweisung

Ausgabe

Arithmetik

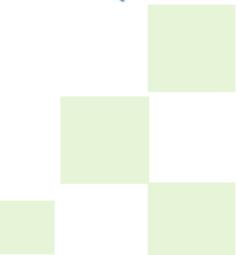
Modulorechnung
In- bzw. Dekrement
Integer-Division
Verknüpfung von
Zeichenketten

Konvertierung

Kommentare

Technologie

Modulo (Restwertbestimmung)



- Das Ergebnis des Modulo ist der Rest der Division:

$$26/5 = 5 \text{ Rest } 1 \Rightarrow 26 \% 5 = 1$$

$$30/2 = 15 \text{ Rest } 0 \Rightarrow 30 \% 2 = 0$$

Ablauf

Interne
Zahlendarstellung

Variablen

Datentypen
Wertebereiche

Variablennamen

Verwendung

Deklaration und
Wertzuweisung

Ausgabe

Arithmetik

Modulorechnung

In- bzw. Dekrement
Integer-Division
Verknüpfung von
Zeichenketten

Konvertierung

Kommentare

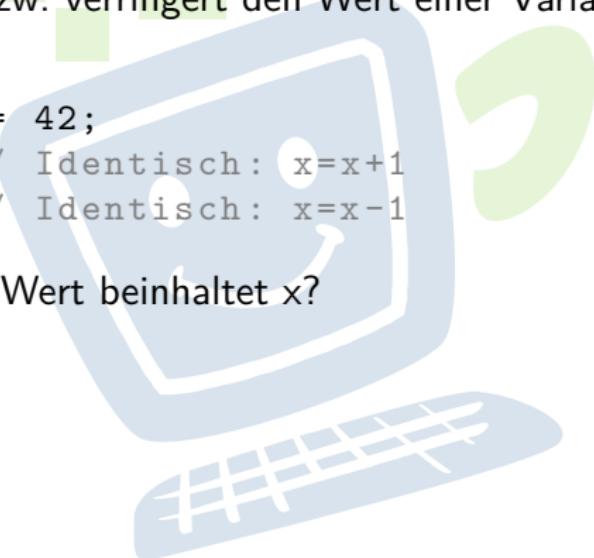
Technologie

In- bzw. Dekrement

- Erhöht bzw. verringert den Wert einer Variablen um 1.

```
int x = 42;  
x++; // Identisch: x=x+1  
x--; // Identisch: x=x-1
```

Welchen Wert beinhaltet x?



Ablauf

Interne
Zahlendarstellung

Variablen

Datentypen
Wertebereiche

Variablenamen

Verwendung

Deklaration und
Wertzuweisung
Ausgabe

Arithmetik

Modularechnung

In- bzw. Dekrement

Integer-Division

Verknüpfung von
Zeichenketten

Konvertierung

Kommentare

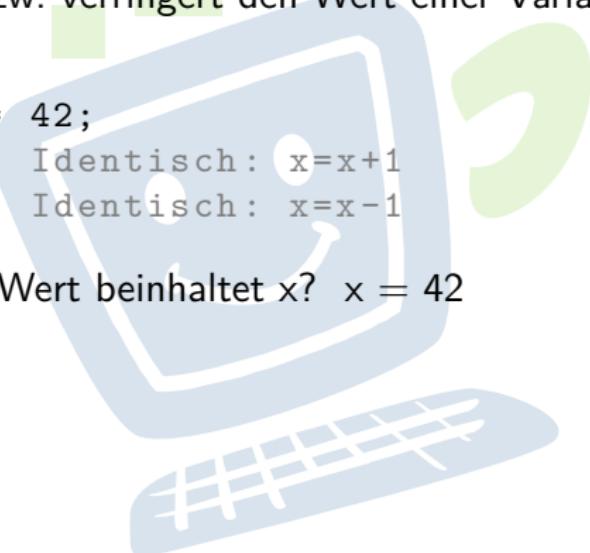
Technologie

In- bzw. Dekrement

- Erhöht bzw. verringert den Wert einer Variablen um 1.

```
int x = 42;  
x++; // Identisch: x=x+1  
x--; // Identisch: x=x-1
```

Welchen Wert beinhaltet x? x = 42



Ablauf

Interne
Zahlendarstellung

Variablen

Datentypen
Wertebereiche

Variablennamen

Verwendung

Deklaration und
Wertzuweisung
Ausgabe

Arithmetik

Modularechnung

In- bzw. Dekrement

Integer-Division

Verknüpfung von
Zeichenketten

Konvertierung

Kommentare

Technologie

Post- bzw. Präinkrement sind zu beachten

```
int x = 23;  
System.out.println(++x);
```

Wie lautet die Ausgabe und welchen Wert hat x?

```
int y = 23;  
System.out.println(y++);
```

Wie lautet die Ausgabe und welchen Wert hat y?

- Ablauf
- Interne Zahlendarstellung
- Variablen
- Datentypen
- Wertebereiche
- Variablennamen
- Verwendung
 - Deklaration und Wertzuweisung
 - Ausgabe
- Arithmetik
 - Modularechnung
 - In- bzw. Dekrement
 - Integer-Division
 - Verknüpfung von Zeichenketten
- Konvertierung
- Kommentare
- Technologie

Post- bzw. Präinkrement sind zu beachten

```
int x = 23;  
System.out.println(++x);
```

Wie lautet die Ausgabe und welchen Wert hat x?

Ausgabe: **24**; x = 24

```
int y = 23;  
System.out.println(y++);
```

Wie lautet die Ausgabe und welchen Wert hat y?

- Ablauf
- Interne Zahlendarstellung
- Variablen
- Datentypen
- Wertebereiche
- Variablennamen
- Verwendung
- Deklaration und Wertzuweisung
- Ausgabe
- Arithmetik
- Modularechnung
- In- bzw. Dekrement
- Integer-Division
- Verknüpfung von Zeichenketten
- Konvertierung
- Kommentare
- Technologie

Post- bzw. Präinkrement sind zu beachten

```
int x = 23;  
System.out.println(++x);
```

Wie lautet die Ausgabe und welchen Wert hat x?

Ausgabe: **24**; x = 24

```
int y = 23;  
System.out.println(y++);
```

Wie lautet die Ausgabe und welchen Wert hat y?

Ausgabe: **23**; y = 24

Ablauf

Interne
Zahlendarstellung

Variablen

Datentypen
Wertebereiche

Variablennamen

Verwendung

Deklaration und
Wertzuweisung

Ausgabe

Arithmetik

Modularechnung

In- bzw. Dekrement

Integer-Division

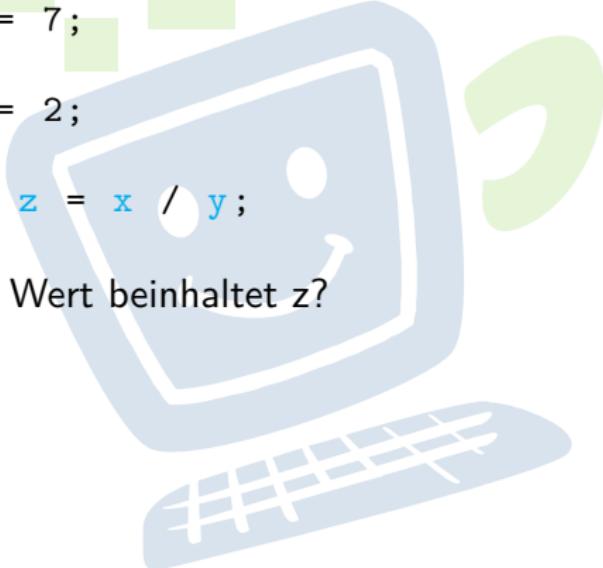
Verknüpfung von
Zeichenketten

Konvertierung

Kommentare

Technologie

Integer-Division

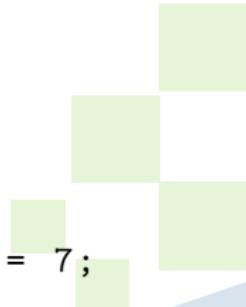


```
int x = 7;  
int y = 2;  
double z = x / y;
```

Welchen Wert beinhaltet z?

- Ablauf
- Interne Zahlendarstellung
- Variablen
- Datentypen
- Wertebereiche
- Variablennamen
- Verwendung
- Deklaration und Wertzuweisung
- Ausgabe
- Arithmetik
- Modularechnung
- In- bzw. Dekrement
- Integer-Division**
- Verknüpfung von Zeichenketten
- Konvertierung
- Kommentare
- Technologie

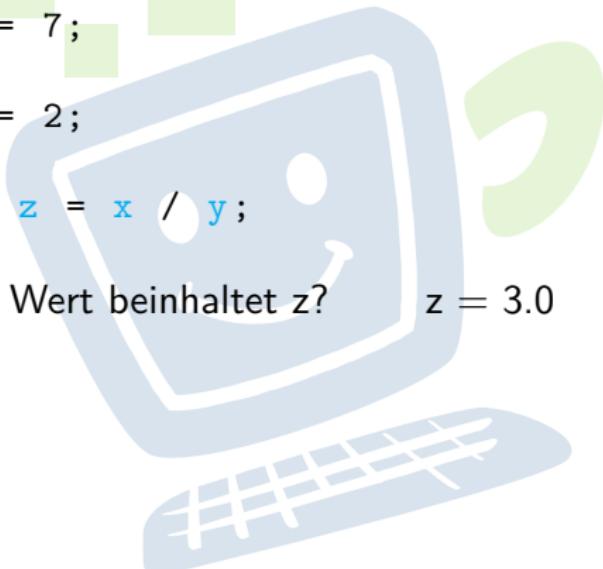
Integer-Division



```
int x = 7;  
  
int y = 2;  
  
double z = x / y;
```

Welchen Wert beinhaltet z?

$z = 3.0$



- Ablauf
- Interne Zahlendarstellung
- Variablen
- Datentypen
- Wertebereiche
- Variablennamen
- Verwendung
 - Deklaration und Wertzuweisung
 - Ausgabe
- Arithmetik
- Modularechnung
- In- bzw. Dekrement
- Integer-Division**
- Verknüpfung von Zeichenketten
- Konvertierung
- Kommentare
- Technologie

Verknüpfung von Zeichenketten

- ▶ Verknüpfung durch den `+`-Operator

```
String name = "Hallo, " + "Welt";
```

- ▶ auch gemischt mit Zahlen möglich

```
int x = 5;
String text = "x hat den Wert " + x;
```

- ▶ Ausgabe:

```
System.out.println("x ist " + x);
System.out.print("Hallo, "
    "Student");
```

Ablauf

Interne
Zahlendarstellung

Variablen

Datentypen
Wertebereiche

Variablenamen

Verwendung

Deklaration und
Wertzuweisung

Ausgabe

Arithmetik

Modularechnung
In- bzw. Dekrement
Integer-Division

Verknüpfung von
Zeichenketten

Konvertierung

Kommentare

Technologie

Explizite Konvertierung

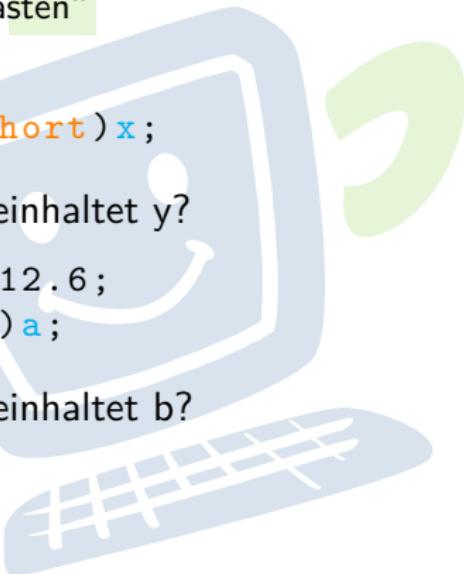
- ▶ Variablenwerte können umgewandelt werden
 - ▶ explizites „Casten“

```
int x = 42;  
short y = (short)x;
```

Welchen Wert beinhaltet y?

```
double a = 512.6;  
int b = (int)a;
```

Welchen Wert beinhaltet b?



Ablauf

Interne
Zahlendarstellung

Variablen

Datentypen
Wertebereiche

Variablennamen

Verwendung

Deklaration und
Wertzuweisung
Auszabe

Arithmetik

Modulorechnung
In- bzw. Dekrement
Integer-Division
Verknüpfung von
Zeichenketten

Konvertierung

Kommentare

Technologie

Explizite Konvertierung

- ▶ Variablenwerte können umgewandelt werden
 - ▶ explizites „Casten“

```
int x = 42;  
short y = (short)x;
```

Welchen Wert beinhaltet y?

```
double a = 512.6;  
int b = (int)a;
```

Welchen Wert beinhaltet b?

y = 42

Ablauf

Interne
Zahlendarstellung

Variablen

Datentypen
Wertebereiche

Variablennamen

Verwendung

Deklaration und
Wertzuweisung
Auszabe

Arithmetik

Modulorechnung
In- bzw. Dekrement
Integer-Division
Verknüpfung von
Zeichenketten

Konvertierung

Kommentare

Technologie

Explizite Konvertierung

- ▶ Variablenwerte können umgewandelt werden
 - ▶ explizites „Casten“

```
int x = 42;  
short y = (short)x;
```

Welchen Wert beinhaltet y?

```
double a = 512.6;  
int b = (int)a;
```

Welchen Wert beinhaltet b?

y = 42

b = 512

Ablauf

Interne
Zahlendarstellung

Variablen

Datentypen
Wertebereiche

Variablennamen

Verwendung

Deklaration und
Wertzuweisung
Auszabe

Arithmetik

Modulorechnung
In- bzw. Dekrement
Integer-Division
Verknüpfung von
Zeichenketten

Konvertierung

Kommentare

Technologie

Implizite Konvertierung

Einige Typen können ohne Probleme in andere umgewandelt werden

byte → **short** → **int** → **long** → **float** → **double**

```
int x = 42;  
float y = (float)x;
```

ist äquivalent zu:

```
int x = 42;  
float y = x;
```

Ablauf

Interne
Zahlendarstellung

Variablen

Datentypen
Wertebereiche

Variablenamen

Verwendung

Deklaration und
Wertzuweisung
Auszabe

Arithmetik

Modulorechnung
In- bzw. Dekrement
Integer-Division
Verknüpfung von
Zeichenketten

Konvertierung

Kommentare

Technologie

Zurück zum Divisionsproblem

```
int x = 7;
```

```
int y = 2;
```

```
double z = x / y;
```

- ▶ Bei Rechnungen wird in den bestmöglichen Typen gecastet
byte → short → int → long → float → double
- ▶ So funktioniert es:

```
double z = (double) x / y;
```

Ablauf

Interne
Zahlendarstellung

Variablen

Datentypen
Wertebereiche

Variablenamen

Verwendung

Deklaration und
Wertzuweisung
Ausgabe

Arithmetik

Modulorechnung
In- bzw. Dekrement
Integer-Division
Verknüpfung von
Zeichenketten

Konvertierung

Kommentare

Technologie

Kommentare

- ▶ Wird verwendet um Code von der Verwendung auszunehmen oder Kommentare zu hinterlassen.
Wenn wir euch auffordern etwas auszukommentieren reden wir hiervon.
- ▶ Mehrzeilige Kommentare:

```
/*
 * Dashier ist alles Kommentar.
 * int number;
 * char Buchstabe;
 */
```

- ▶ Einzeilige Kommentare:

```
int number; //Hier beginnt der
Kommentar.
```

- Ablauf
- Interne Zahlendarstellung
- Variablen
- Datentypen
- Wertebereiche
- Variablennamen
- Verwendung
- Deklaration und Wertzuweisung
- Ausgabe
- Arithmetik
- Modulorechnung
- In- bzw. Dekrement
- Integer-Division
- Verknüpfung von Zeichenketten
- Konvertierung
- Kommentare
- Technologie

Kommentare

- ▶ Wird verwendet um Code von der Verwendung auszunehmen oder Kommentare zu hinterlassen.
Wenn wir euch auffordern etwas auszukommentieren reden wir hiervon.
- ▶ Mehrzeilige Kommentare:

```
/*
 * Dashier ist alles Kommentar .
 * int number;
 * char Buchstabe;
 */
```

- ▶ Einzeilige Kommentare:

```
int number; //Hier beginnt der
Kommentar .
```

| |
|----------------------------------|
| Ablauf |
| Interne Zahlendarstellung |
| Variablen |
| Datentypen |
| Wertebereiche |
| Variablennamen |
| Verwendung |
| Deklaration und Wertzuweisung |
| Ausgabe |
| Arithmetik |
| Modulorechnung |
| In- bzw. Dekrement |
| Integer-Division |
| Verknüpfung von Zeichenketten |
| Konvertierung |
| Kommentare |
| Technologie |

Technologie

Quellcode (Java)



Ablauf

Interne
Zahlendarstellung

Variablen

Datentypen
Wertebereiche

Variablennamen

Verwendung

Deklaration und
Wertzuweisung
Ausgabe

Arithmetik

Modularechnung
In- bzw. Dekrement
Integer-Division
Verknüpfung von
Zeichenketten

Konvertierung

Kommentare

Technologie

Technologie

Quellcode (Java)

Compiler

Zwischencode (Bytecode)



Ablauf

Interne
Zahlendarstellung

Variablen

Datentypen
Wertebereiche

Variablennamen

Verwendung

Deklaration und
Wertzuweisung
Ausgabe

Arithmetik

Modularechnung
In- bzw. Dekrement
Integer-Division
Verknüpfung von
Zeichenketten

Konvertierung

Kommentare

Technologie

Technologie

Quellcode (Java)

Compiler

Zwischencode (Bytecode)

Laufzeitumgebung (JVM)

Maschinencode

Ablauf

Interne
Zahlendarstellung

Variablen

Datentypen
Wertebereiche

Variablenamen

Verwendung

Deklaration und
Wertzuweisung
Auszabe

Arithmetik

Modularechnung
In- bzw. Dekrement
Integer-Division
Verknüpfung von
Zeichenketten

Konvertierung

Kommentare

Technologie

Eclipse Einführung

- ▶ Live Demo



Ablauf

Interne
Zahlendarstellung

Variablen

Datentypen
Wertebereiche

Variablennamen

Verwendung

Deklaration und
Wertzuweisung
Ausgabe

Arithmetik

Modularechnung
In- bzw. Dekrement
Integer-Division
Verknüpfung von
Zeichenketten

Konvertierung

Kommentare

Technologie