

# Java

Prof. Dr. Stefan Enderle  
NTA Isny

# 3. Weitere Grundlagen

# Kommentare

- 2 Formen:

- „C-Kommentar“:

- `/* bla bla bla bla */`

- „Einzeilen-Kommentar“:

- `// bla bla bla`

- Typische Benutzung:

- Auskommentieren einer oder mehrerer Zeilen:

- `/*  
 bla bla bla  
*/`

- Anhängen eines Kommentars an eine Programmzeile:

- `System.out.print("Hallo"); // Schreib Hallo`

# Kommentare

- Anders herum geht es auch:
  - Auskommentieren einer oder mehrerer Zeilen:

```
//  
// bla bla bla  
//
```
  - Anhängen eines Kommentars an eine Programmzeile:

```
System.out.print("Hallo");    /* Schreib Hallo */
```
- Jede Kommentarform kann die andere auskommentieren:
  - ```
//  
// /* Hier stand schon ein Kommentar */  
//
```
  - ```
/*  
System.out.print("Hallo");    // Schreib Hallo  
*/
```

# Datentypen

- Alle Daten (Variablen und Konstanten) besitzen einen Datentyp, den der Programmierer festlegen muss.
- Der Datentyp legt fest, welche Operationen mit den Daten erlaubt sind und welchen Typ das Ergebnis besitzt.

# Ganze Zahlen

- `int`            4 Byte   -2147483647 bis 2147483647
- `short`           2 Byte   -32768 bis 32767
- `long`            8 Byte   -9\*10<sup>18</sup> bis 9\*10<sup>18</sup>
- `byte`            1 Byte   -128 bis 127
  
- In Java gibt es keinen Typ `unsigned`.

# Gleitkommazahlen

- `float`      4 Byte    $3.4 \cdot 10^{-38}$  bis  $3.4 \cdot 10^{38}$    (7 Stellen)
- `double`      8 Byte    $1.7 \cdot 10^{-308}$  bis  $1.7 \cdot 10^{308}$    (125 Stellen)
  
- Einer float-Zahl muss ein "f" folgen:      `123.4f`
- Folgt kein "f", so wird die Zahl als double interpretiert.  
Alternativ darf ein "d" folgen:  
`123.4`      `123.4d`

# Zeichen

- `char`            2 Byte            `a..z, A..Z, 0..9, !, ?, =, %, ...`
- Zeichen werden in einzelnen Hochkomma eingeschlossen:  
`'a'`            `'4'`            `'?'`
- Das verwendete "Unicode" ist ein Zeichensatz mit 56536 Zeichen, die auch Umlaute und Sonderzeichen sowie japanische und chinesische Zeichen einschließt.

# Wahrheitswerte

- `boolean`    **1 Byte**    `true` **oder** `false`

# Zeichenketten (Strings)

- In C als Array von `char`.
- In C++: `string`
- In Java: `String`

# Variablen und Deklaration

- Bevor eine Variable benutzt werden kann, muss sie *deklariert* werden
- Bei der Deklaration wird der Variablen ein Datentyp fest zugeordnet
- Beispiele:
  - Deklaration von `x` als `int`:  
`int x;`
  - Deklaration von `zins` als `float`:  
`float zins;`
  - Deklaration von `name` als `string`:  
`String name;`

# Zuweisung

- Einer deklarierten Variablen kann ein Wert zugewiesen werden.
- Der Typ des zugewiesenen Wertes muss gleich oder kompatibel sein.
- Zuweisungssymbol ist das Zeichen =
- Beispiele:
  - Integer `x` erhält den Wert 50:

```
int x;  
x = 50;
```
  - Realzahl `zins` wird 372.90:

```
float zins;  
zins = 372.90f;
```
  - Der String `name` soll „Anna“ beinhalten:

```
String name;  
name = "Anna";
```

# Bemerkungen

- Eine Variable muss *vor* dem Gebrauch deklariert werden.
- Eine Variable wird nur 1x deklariert. Danach wird sie beliebig oft verwendet.
- Der deklarierte Typ kann sich nicht mehr ändern.
- **Kurzform:** Deklaration und Wertzuweisung

```
int x = 50;  
float zins = 372.90f;  
String name = "Anna";
```

# Operationen mit Ganzzahlen

- Bei Ganzen Zahlen (`int`):
  - Addition +
  - Subtraktion -
  - Multiplikation \*
  - Ganzzahlige Division /
  - Rest bei Division %
- Ergebnis-Typ von `int`-Operationen ist wieder `int`
- Beispiel:
  - `int a = 5;`
  - `int b = 2;`
  - `int c = a/b; // c ist dann 2 !`

# Gleitkomma-Operationen

- Bei float und double:
  - Addition +
  - Subtraktion -
  - Multiplikation \*
  - Division /
- Resultate von floats sind wieder float, von doubles wieder double, von gemischt dann double
- Beispiel:
  - `float a = 5.2f;`
  - `float b = 2.0f;`
  - `float c = a/b; // c ist dann 2.6`

# Boolsche Operationen

- Bei boolean:
  - Und &&
  - Oder ||
  - Nicht !
- Resultat wieder boolean
- Beispiel:
  - ```
boolean a = true;  
boolean b = false;  
boolean c = a&&b;           // c ist false  
boolean d = a||b;          // d ist true  
boolean e = !a;            // e ist false
```

# Vergleichsoperationen

- Vergleichsoperationen vergleichen zwei Zahlenwerte und liefern als Ergebnis einen `boolean` Wert.

- Operationen: `<` `>` `<=` `>=` `==` `!=`

- Beispiel:

```
- boolean a = 6>5;           // a ist true
```

```
- boolean a = 6<=5;         // a ist false
```

```
- boolean a = 6==5;         // a ist false
```

```
- boolean a = 6!=5;         // a ist true
```

# Konstanten

- Konstanten werden behandelt wie Variablen, ändern aber ihren Wert nicht.
- Konstanten werden durch das Schlüsselwort `final` bezeichnet.
- **Deklaration:**
  - `final float PI = 3.14159265;`
  - `final int MAX_SPEED = 250;`
  - `final String TEXT = "Für immer gleich";`