

Einfache Datentypen

((itanius informatik))

Java Foundation Track by Carsten Bokeloh

Einfache oder primitive Datentypen

- Ganzzahlige Datentypen
- Gleitkommatypen
- char
- Zeichenketten
- boolean
- Implizite und explizite Typumwandlungen

Ganzzahlige Datentypen

| Typname | größter Wert | kleinster Wert | Länge |
|---------|---------------------|----------------------|---------|
| byte | 127 | -128 | 8 Bits |
| short | 32767 | -32768 | 16 Bits |
| int | 2147483647 | -21474483648 | 32 Bits |
| long | 9223372036854775807 | -9223372036854775808 | 64 Bits |

Gleitkommatypen

| Typname | größter positiver Wert | Kleinster positiver Wert | Länge |
|---------|-----------------------------|-----------------------------|---------|
| float | 3.402823466385288 6E+038 | 1.401298464324817 1E-045 | 32 Bits |
| double | 1.797693134862315 7E+308 | 4.940656458412465 4E-324 | 64 Bits |
| | | | |
| | | | |

char

- ❖ Enthält einen unicode character
- ❖ Default ist `\u0000`
- ❖ Größe ist 16 bit (2 byte)
- ❖ Beispiel: `'a'` oder `'7'`

Zeichenketten

- ❖ mehrere Zeichen des Typs char können zu einer Zeichenkette zusammengefasst werden. (String)
- ❖ dies entspricht dann aber keinem einfachen Datentyp, sondern es entspricht einem Objekt der Klasse String

boolean

❖ true or false

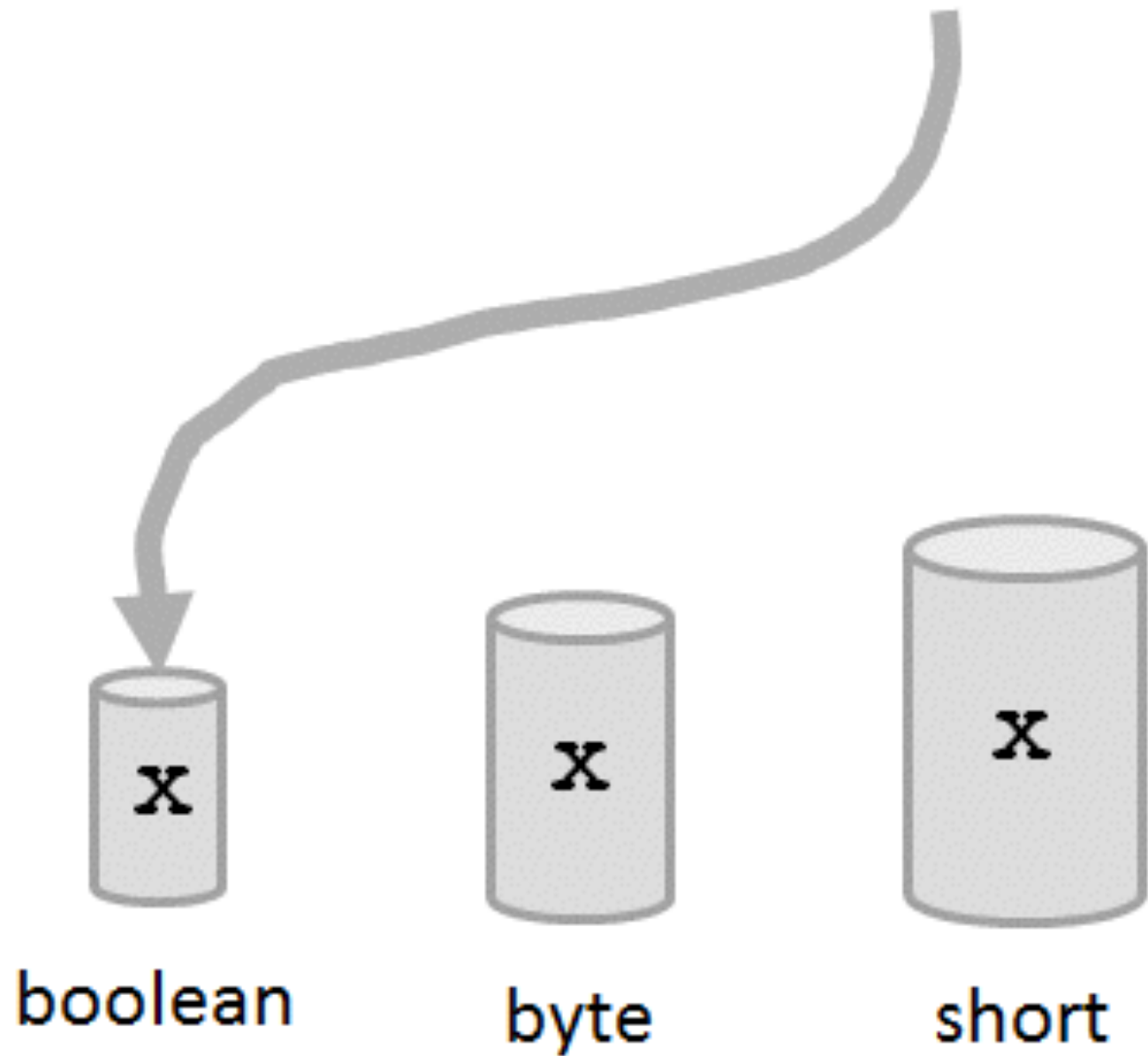
Implizite und explizite Typumwandlungen

- ❖ implizite Typkonvertierungen, wenn einer kleinerer Zahlenbereich in einen größeren abgeleitet wird
- ❖ byte nach short, short nach int, int nach long

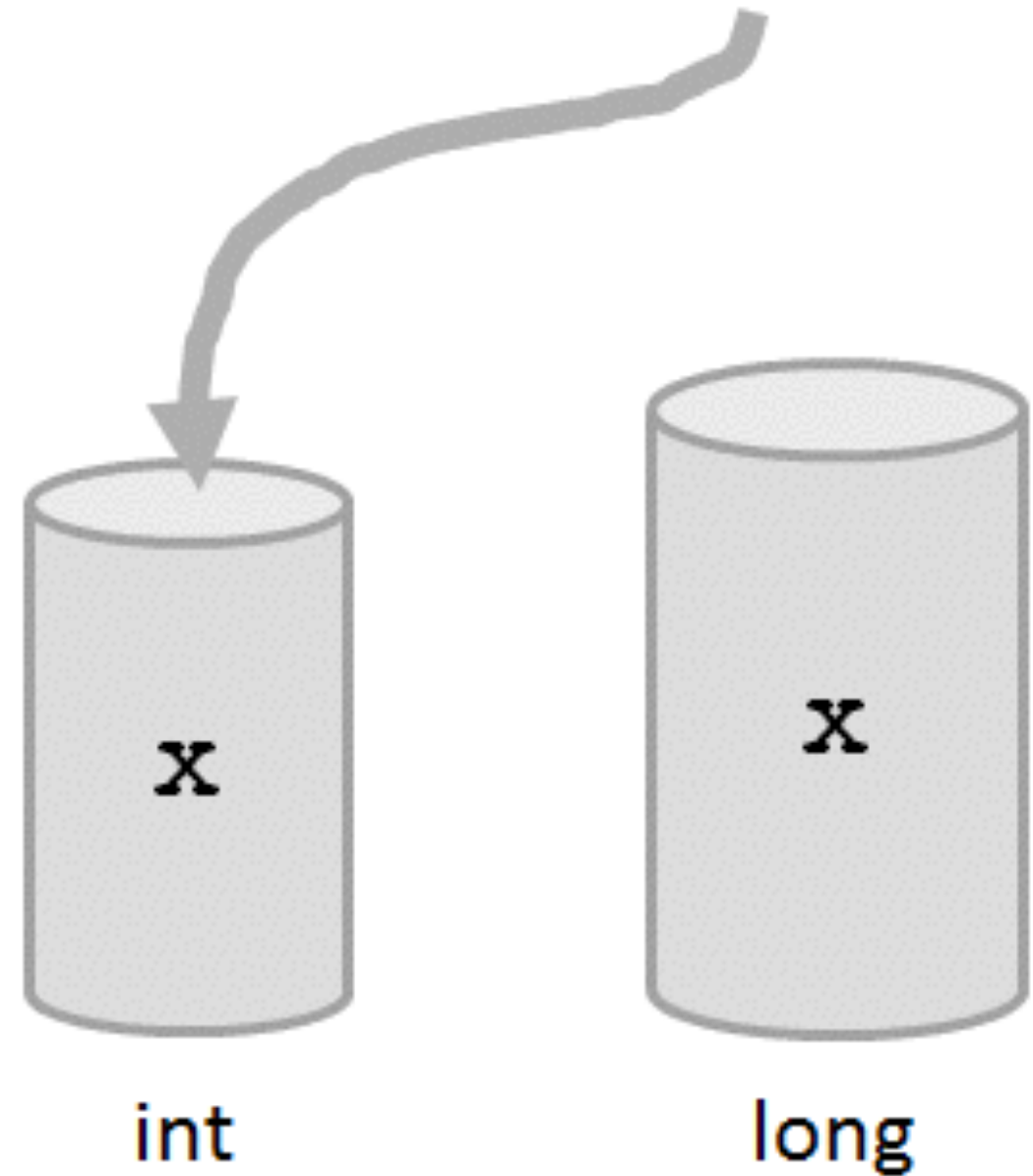
Achtung!!!

bei **expliziter** Typumwandlung gehen Informationen verloren,
Compiler macht es aber **nicht** automatisch
(int)3.14 wird zu 3


```
boolean x = true;
```



```
int x = 12;
```



Einfache oder auch primitive Datentypen

- ❖ primitive Datentypen
 - ❖ Die primitiven Typen sind Datentypen wie Zahlen und einzelne Zeichen, die nicht in Objekten verwaltet werden
 - ❖ *boolean(1 Bit), char(16 Bit), byte(8 Bit), short(16 Bit), int(32 Bit), long(64 Bit), float(32 Bit), double(64 Bit)*

Merksatz: bei charmanten Bären sind Ihre leckeren Früchte dahin

Übung

```
double myDouble1 = 2147483642d;  
double myDouble2 = 2147483647d;  
double myDouble3 = 2147483648d;  
    int a = 32769;  
int myInt1, myInt2, myInt3;  
    short myShort;  
    short x = 5, y = 12;  
  
myInt1= (int) myDouble1;  
myInt2= (int) myDouble2;  
myInt3= (int) myDouble3;  
    myShort= (short) a;  
    short z = x + y;
```

Umgang mit einfachen Datentypen

((itanius informatik))

Java Foundation Track by Carsten Bokeloh

Umgang mit einfachen Datentypen

- Variablen
- Operatoren & Ausdrücke
- Zuweisungsoperator
- Vergleichsoperator
- Inkrement und Dekrementoperatoren

Variable

Variablentyp

Ausdruck

final int anzahlTeilnehmer = 10;

Variablenbezeichner

Operatoren

❖ <<OPERATOR>> <<OPERAND>>

❖ **i++**

❖ <<OPERAND>> <<OPERATOR>>

❖ **++i**

❖ <<OPERAND>> <<OPERATOR>> <<OPERAND>>

❖ **i + j**

❖ <<OPERAND>> <<OPERAND>> <<OPERAND>>

❖ **i == 0 ? OK : nicht OK**

Arithmetische Operatoren

- ❖ erwarten **byte, short, int, long, float, double, char**
- ❖ aber: “abc” + “def” ergibt “abcdef”

Vorsicht!!!

- ❖ `short a = 1; short b = 2; short c = a + b;`

Zuweisungsoperatoren

- ❖ `a = 3;`
- ❖ `a = b = c = 5;`
- ❖ `a = a + 1;`
- ❖ `a += 1;`

Vergleichsoperatoren

$>$ größer als

$>=$ größer gleich

$<$ kleiner als

$<=$ kleiner gleich

$==$ gleich

$!=$ ungleich

logische Operatoren

| boolean a | boolean b | ! a | a && b | a b | a ^ b |
|------------------|------------------|------------|-----------------------|---------------|--------------|
| true | true | false | true | true | false |
| true | false | false | false | true | true |
| false | true | true | false | true | true |
| false | false | true | false | False | false |

Inkrement & Dekrement

❖ `a++;`

❖ `a- -;`

❖ `++a;`

❖ `- - a;`

Im Buch gibt es mehr Infos unter Kapitel 2.3

[http://openbook.galileocomputing.de/javainsel/
javainsel_02_003.html#dodtpba664243-0cd9-4f96-b1d7-63063559438b](http://openbook.galileocomputing.de/javainsel/javainsel_02_003.html#dodtpba664243-0cd9-4f96-b1d7-63063559438b)

Übung

- ❖ Welche Statements sind falsch?
- ❖ Ein Programmierer kann keine neuen primitiven Datentypen definieren.
- ❖ Ein Programmierer kann neue Datentypen definieren.
- ❖ Einmal zugewiesen, kann der Wert eines primitiven Datentyps nicht mehr modifiziert werden.
- ❖ Ein Wert kann nicht einem primitiven Datentyp zugewiesen werden.

Übung

```
public class Foo {  
    public static void main(String args[]) {  
        int a = 10;  
        long b = 20;  
        short c = 30;  
        System.out.println(++a + b++ * c);  
    }  
}
```

a: 611

b: 641

c: 930

d: 960

Übung

Schreiben Sie ein JAVA Programm, das eine vorgegebene Zahl von Sekunden in Jahre, Tage, Stunden, Minuten zerlegt:

Das Programm soll für einen Sekundenwert von 158036522

folgendes ausgeben:

158036522 sind:

5 Jahre,

4 Tage,

3 Stunden,

2 Minuten und 2 Sekunden.