



Variablen



Was sind Variablen?

Warum ein Typ?

Überblick

Variablen deklarieren und initialisieren

Final Bezeichner

Exkurs: Zeichenketten

Funktionen und Operatoren

Konvertierung

String Konvertierung

Implizite Konvertierung

Anwendungsfälle

Analyse

Explizite Konvertierung

Exkurs: Input Klasse

Fragerunde

Was sind Variablen?



TECHNISCHE
UNIVERSITÄT
DARMSTADT

- ▶ Speicherplatz für Werte
- ▶ Ein Fach im Speicher, in dem man Werte ablegen kann
- ▶ Variable als **alias** für dieses Fach

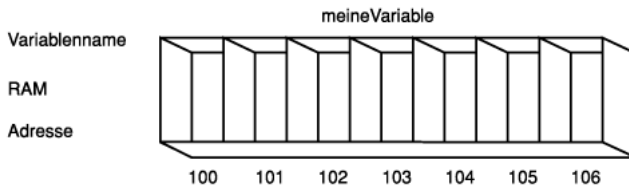


Abbildung : Speichermodell [1]

Warum ein Typ?

- ▶ Fächer im Speicher sind **1 Byte** groß
- ▶ Der Typ gibt an, wie viele *Fächer* reserviert werden sollen

Typ	Bezeichnung	Größe
int	Ganzzahl	32 Bit
float	Fließkommazahl	32 Bit
short	Ganzzahl	16 Bit
...

Achtung

In anderen Hochsprachen, wie beispielsweise **C++**, haben einzelne Variablen auf unterschiedliche Computern andere Größen. In Java gilt dies **nicht!**

Warum ein Typ?

Überblick



TECHNISCHE
UNIVERSITÄT
DARMSTADT

► Primitive Datentypen in Java

Typ	Bezeichnung	Größe	Wertebereich
boolean	Wahrheitswert	1 Bit	true, false
byte	Ganzzahl	8 Bit	−128 bis 127
short	Ganzzahl	16 Bit	−32.768 bis 32.767
int	Ganzzahl	32 Bit	− 2^{31} bis $2^{31} - 1$
long	Ganzzahl	64 Bit	− 2^{63} bis $2^{63} - 1$
float	Fließkommazahl	32 Bit	etwa $-3.4028 \cdot 10^{38}$ bis $3.4028 \cdot 10^{38}$
double	Fließkommazahl	64 Bit	etwa $-1.7977 \cdot 10^{308}$ bis $1.7977 \cdot 10^{308}$
char	Zeichen	16 Bit	Alle Unicode-Zeichen

- ▶ **Typ** Name \Rightarrow *int myAge*;
- ▶ Wird eine Variable deklariert, so wird Speicher allokiert
- ▶ Aussagekräftige Namen für Variablen verwenden
- ▶ Variablennamen müssen mit einem Buchstaben, einem Unterstrich oder \$ beginnen
- ▶ Case sensitive
- ▶ Konventionen für Variablennamen
- ▶ Lieber einen Variablentyp größer nehmen (Ariane)



Variablen deklarieren und initialisieren



TECHNISCHE
UNIVERSITÄT
DARMSTADT

```
int var; //Variable wird deklariert
var = 5; //Variable wird mit dem Wert 5 initialisiert
float floating = 5.5f;
char gender = 'm';
double x,y,height = 12; // x und y bleiben uninitialisiert
```

- ▶ Verwendung des Zuweisungsoperators =
- ▶ Deklaration und Initialisierung gleichzeitig oder getrennt

Achtung

Als Dezimaltrennzeichen wird hier der Punkt verwendet.

Variablen deklarieren und initialisieren



TECHNISCHE
UNIVERSITÄT
DARMSTADT

```
float speed = 12.5 f;  
double maxSpeed = 12.0E2;
```

- ▶ Enthält die Zahl einen Dezimalpunkt oder Exponenten (**E**) wird diese als double interpretiert
- ▶ **f** bzw. **F** anhängen für float Werte

Merke

In Java werden Gleitkommazahlen ohne Angabe von F oder f als double.



- ▶ Konstanten gibt es auch in Java
- ▶ Einmal deklariert und initialisiert können diese nicht mehr verändert werden

```
final int strong = 42;  
strong = 0; // cannot assign a value to final variable strong
```



- ▶ Bisher nur möglich einzelne Zeichen im Datentyp **char** zu Speichern
- ▶ Wie speichert man ganze Zeichenketten?

```
String sentence = "Das_ist_ein_Minkovski_Raum,_manche_Menschen_  
nennen_ihn_auch_Weltraum_-_Prof._Dr._Jürgen_Wambach,_  
Theoretische_Physik,_TU_Darmstadt";
```

Hinweis

Im Gegensatz zu char's werden Strings mit Anführungszeichen eingeschlossen

Achtung

Das 'Keyword' **String** muss groß geschrieben werden!

Exkurs: Zeichenketten

Funktionen und Operatoren



TECHNISCHE
UNIVERSITÄT
DARMSTADT

- ▶ String deklarieren, initialisieren und ausgeben

```
String sentence = "Nicole: Ich will mit Peter spielen!";  
System.out.print(sentence);
```

- ▶ Strings verketteten

```
String subject = "Studenten";  
String predicate = "brauchen";  
String object = "Schlaf";  
String sentence = subject + predicate + object + "!";
```

- ▶ Länge eines Strings

```
int length = sentence.length();
```



► Einzelne Zeichen eines Strings

```
String sentence = "Studenten_haben_keine_Ferien";  
String subString = "" + sentence.charAt(22) + sentence.charAt  
    (11) + sentence.charAt(5) + sentence.charAt(6) + sentence.  
    charAt(11);  
System.out.print(subString);
```

Achtung

In der Informatik fängt man bei 0 an zu zählen

Abschließend

Achtung: Strings sind keine primitiven Datentypen!



```
int counter = 10; // ok
double speed = 12.5; // ok

String number = 10; // ??
int dramaQueen = 21.5; // ??
```

- ▶ Wie speichert man eine Ganzzahl in einem String?
- ▶ Was passiert bei der Zuweisung einer Gleitkommazahl in einen Container für Ganzzahlen?

Konvertierung

String Konvertierung



TECHNISCHE
UNIVERSITÄT
DARMSTADT

- ▶ Einen String mit einer Ganzzahl verketteten

```
int number = 42;  
String senseless = "Die_Zahl_" + number + "_ist_awesome";
```

- ▶ Einen primitiven Datentyp in einen String umwandeln

```
int halfTruth = 21;  
double number = 17.5;  
  
String note = Integer.toString(halfTruth);  
String convert = Double.toString(number);
```

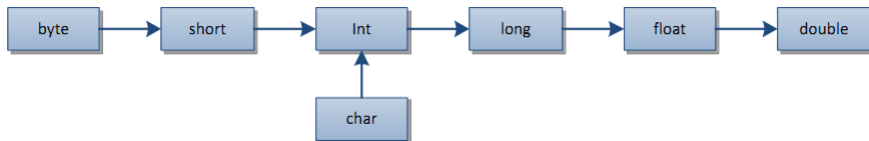
Konvertierung

Implizite Konvertierung



TECHNISCHE
UNIVERSITÄT
DARMSTADT

- ▶ Bereits beim Kompilieren ist Java in der Lage die unterschiedlichen Datentypen unter Berücksichtigung ihres Wertebereichs ineinander umzuwandeln
- ▶ kleiner Datentyp \Rightarrow größerer Datentyp
- ▶ Java folgt dabei dieser Hierarchie:



Hinweis

Boolsche Datentypen lassen sich nicht in einen anderen übertragen



► Ganzzahl Division, **die erste**

```
int a = 8; int b = 3;  
int q = a/b;           // Erwartet: 2.666667  
System.out.print(q); // Ausgabe: 2
```

► Ganzzahl Division, **die zweite**

```
int a = 8; int b = 3;  
float q = a/b;         // Erwartet: 2.666667  
System.out.print(q); // Ausgabe: 2.0
```

► Fließkomma Division

```
float a = 8; float b = 3;  
float q = a/b;         // Erwartet: 2.666667  
System.out.print(q) // Ausgabe 2.666667
```




- Informationsverlust durch Typverengung

```
float a = 8; float b = 3;  
int q = a/b; // Compiler verweigert die Übersetzung  
System.out.print(q);
```

Analyse

- Operator hier Kontextabhängig (Fließkomma Division, Ganzzahl Division)
- Die Genauigkeit des Ergebnisses lässt sich weiter erhöhen, wenn man beispielsweise **double** verwendet.
- Lösung nicht zufriedenstellend. Ansatz ohne Divisor und Dividend zu Ändern?

Konvertierung

Explizite Konvertierung



TECHNISCHE
UNIVERSITÄT
DARMSTADT

- ▶ Eigene Verantwortung der Typumwandlung
- ▶ Expliziter Cast wird durchgeführt durch voranstellen des Zieltyps.

```
int a = 8; int b = 3;  
float q = (float)a/b;  
System.out.print(q); //Ausgabe 2.666667
```



- ▶ Eingabe in der Konsole einlesen und verarbeiten

Beispiele

- ▶ Zeichenkette einlesen

```
System.out.print("Deine_Eingabe: ");  
String eingabe = Input.readString();
```

- ▶ Ganzzahlen einlesen

```
System.out.print("Deine_Glückszahl: ");  
int number = Input.readInt();
```



- ▶ Zeichen einlesen

```
System.out.print("Geschlecht?(m,w): ");  
char gender = Input.readChar();
```

- ▶ Fließkommazahl einlesen

```
System.out.print("Eine Kommazahl: ");  
float number = Input.readFloat();
```

Achtung

Bei der Eingabe der Fließkommazahl das Zeichen , als Dezimalpunkt nutzen!

Exkurs: Input Klasse

Live-Coding



TECHNISCHE
UNIVERSITÄT
DARMSTADT

```
public class Input_Listing {  
  
    public static void main(String[] args) {  
        System.out.println("Zahl_eingeben:");  
        int number = Input.readInt();  
        number = number + 1;  
        System.out.print("Nummer_um_1_inkrementiert:" + number);  
    }  
}
```

Wichtig

Die Datei Input.java muss im gleichen Verzeichnis wie die Quelldatei liegen!



- ▶ Was passiert, wenn man einem int eine reelle Zahl zuweist?(int test = 5.4;)
- ▶ Was ist der Unterschied zwischen int und float?
- ▶ Wodurch zeichnen sich gute und schlechte Variablennamen aus?



- ▶ **[1]** C++ in 21 Tagen
- ▶ **[2]** <http://www.futurama-madhouse.net/fanart/daran.shtml>