

Variablen

Lernziele

Variablen

Definitionsregeln

Namenskonvention

Schlüsselwort final

Lernziele

- Variablen mit passenden Datentypen deklarieren und initialisieren können
- Wissen, wann welcher Wertebereich für einen Wert sinnvoll ist
- Die unterschiedlichen Datentypen kennen
- Namensgebung und Namenskonventionen für Variablen kennen
- Wissen, wie man das Schlüsselwort «final» im Zusammenhang mit Variablen braucht und wofür man es braucht

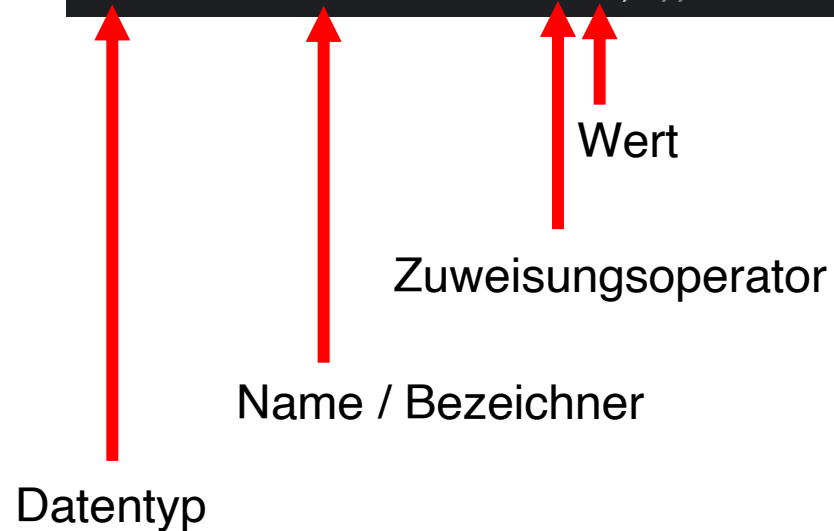
Was sind Variablen?

- Ein Name für einen Speicherbereich, in dem ein Datenobjekt abgelegt wird
 - Platzhalter um Daten zu speichern

Definition

- Jede Variable muss vor Gebrauch definiert werden

```
int meineVariableOhneWert; // Deklaration  
int meineVariableMitWert = 10; // Deklaration + Initialisation
```



Datentypen

- 8 primitive / elementare Datentypen

1 byte = 8 bits

Datentyp	Standartwert	Wertebereich	Grösse in bit
byte	0	-128 bis 127 (inklusive)	8
short	0	-32,768 bis 32,767(inklusive)	16
int	0	-2^{31} bis $2^{31}-1$	32
long	0L	-2^{63} bis $2^{63}-1$	64
float	0.0f	Reicht für ~6-7 nachkommastellen	32
double	0.0d	Reicht für ~15 nachkommastellen	64
boolean	false	true, false	8/16 (meistens 8)
char	'\u0000'	'\u0000' bis '\uffff',	16

int = integer, char = character

Für Interessierte:

[Warum reserviert Java für char 2 bytes?](#)

[Java Unicode System](#)

byte

Datentyp	Standartwert	Wertebereich	Grösse in bit
byte	0	-128 bis 127 (inklusive)	8

```
byte beispielOhneInitialisierung;  
byte beispielEins = -10;  
byte beispielZwei = 105;  
  
// Fehler!  
// "Required Type: byte, Provided: int"  
byte funktioniertNicht = 129;
```

short

Datentyp	Standartwert	Wertebereich	Grösse in bit
short	0	-32,768 bis 32,767(inklusive)	16

```
short beispielOhneInitialisierung;
```

```
short beispielEins = -30000;
```

```
short beispielZwei = 16123;
```

```
// Fehler!
```

```
// "Required Type: short, Provided: int"
```

```
short funktioniertNicht = 32769;
```

int

Datentyp	Standartwert	Wertebereich	Grösse in bit
int	0	-2^{31} bis $2^{31}-1$	32

```
int beispielOhneInitialisierung;  
int beispielEins = -1137402699;  
int beispielZwei = 2147412345;  
  
// Fehler!  
// "Integer number too large"  
int funktioniertNicht = 2147483648; //  $2^{31}-1 = 2147483647$ 
```

```
int oktalZahl = 012;           // Führende "0"  
int hexadezimaleZahl = 0x12;   // Führendes "0x"  
int binaereZahl = 0b1011;      // Führendes "0b"
```

Tipp: `int grosseZahl = 1_234_567_789;`

long

Datentyp	Standartwert	Wertebereich	Grösse in bit
long	0L	-2^{63} bis $2^{63}-1$	64

```
long beispielOhneInitialisierung;  
long beispielEins = -8223371234554775807L;  
long beispielZwei = 8223371234554775807L;  
  
// Fehler!  
// "Long number too large"  
long funktioniertNicht = 9223372036854775808L; //  $2^{63}-1 = 9223372036854775807L$ 
```

float

Datentyp	Standartwert	Wertebereich	Grösse in bit
float	0.0f	Reicht für ~6-7 nachkommastellen	32

```
float beispielOhneInitialisierung;
```

```
float beispielEins = -1.337f;
```

```
float beispielZwei = 21231.12356f;
```

```
// Achtung!
```

```
// float ist nach ~6-7 Nachkommastellen ungenau, siehe letzte Stelle(n)
```

```
float ungenauEins = 1.12345675f; // = ~1.1234567
```

```
float ungenauZwei = 1.12345679f; // = ~1.1234568
```

```
float wissenschaftlicheSchreibweise = 32e4f; // 320000.0
```

<https://0.300000000000000004.com/>

double

Datentyp	Standartwert	Wertebereich	Grösse in bit
double	0.0d	Reicht für ~15 nachkommastellen	64

```
double beispielOhneInitialisierung;  
double beispielEins = -1.123456789012345d;  
double beispielZwei = 21231.12356789012345d;  
  
// Achtung!  
// Double ist nach ~15 Nachkommastellen ungenau, siehe letzte Stelle(n)  
double ungenauEins = 1.1234567890123455d; // = ~1.1234567890123455  
double ungenauZwei = 1.1234567890123459d; // = ~1.123456789012346
```

boolean

Datentyp	Standartwert	Wertebereich	Grösse in bit
boolean	false	true, false	8/16 (meistens 8)

```
boolean beispielOhneInitialisierung;  
boolean beispielEins = false;  
boolean beispielZwei = true;
```

char

Datentyp	Standartwert	Wertebereich	Grösse in bit
char	'\u0000'	'\u0000' bis '\uffff',	16

```
char beispielOhneInitialisierung;  
char beispielEins = '\u003D'; // Unicode escape sequence '\u003D' can be replaced with '='  
char beispielZwei = 61; // '='  
char beispielDrei = 'A';
```

- https://en.wikipedia.org/wiki/List_of_Unicode_characters#Basic_Latin

- Tipp: Sucht nach «=»

U+003D	=	61	075	Equal sign	0030
--------	---	----	-----	------------	------

- Escape-Sequenzen
 - \n = Zeilenumbruch, \\ = Slash etc..

Vereinfachte Definition

- Variablen können gleich mehrfach...

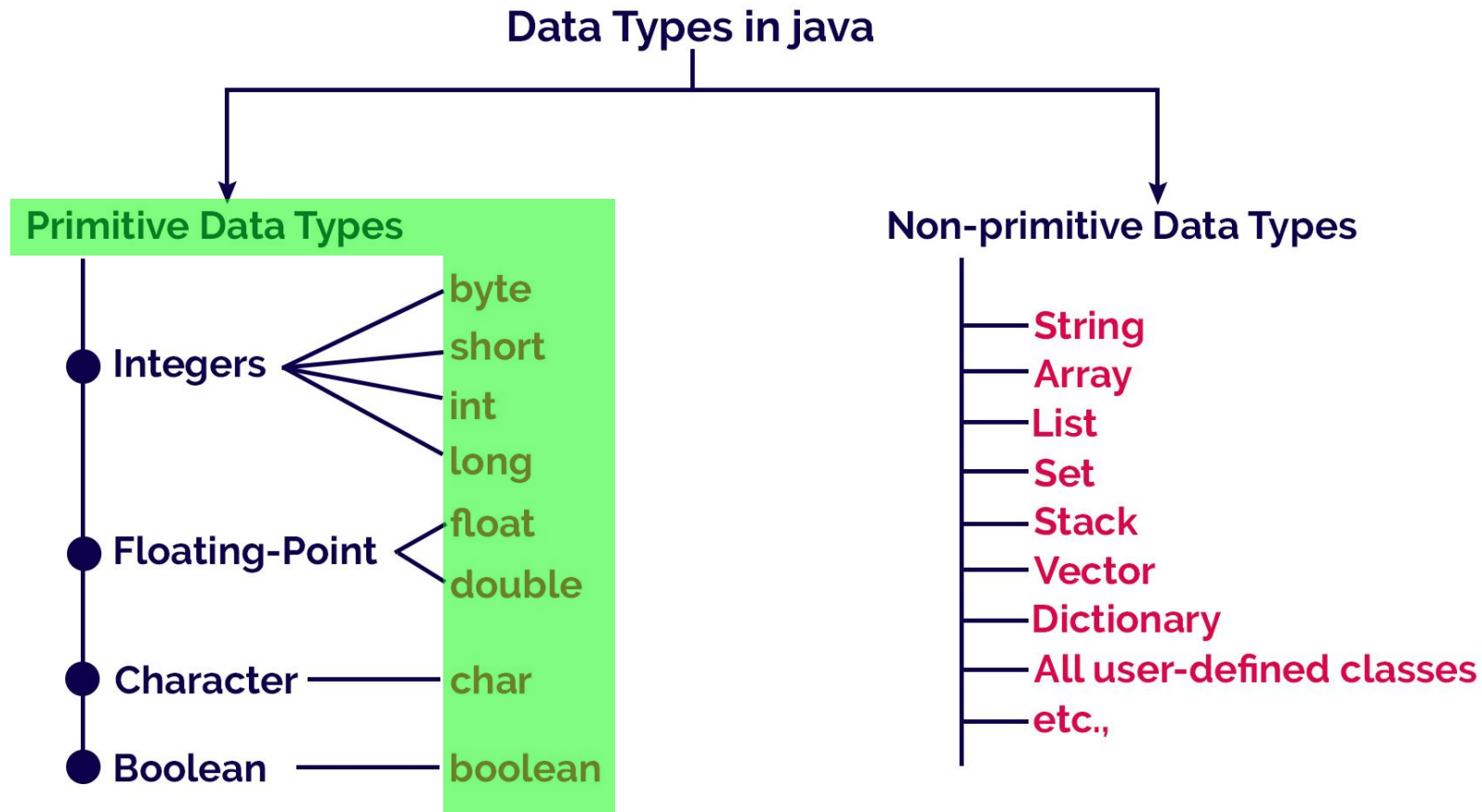
- ...deklariert werden: `int x, y, z;`

- ...deklariert und initialisiert werden: `int a = 10, b = 5, c = 3;`

- Oder: `int d, e, f;`
`d = 7;`
`e = 2;`
`f = 1;`

- Oder: `int g, h, i;`
`g = h = i = 42;`

Datentypen



Namensgebung Regeln

- Für die Namensgebung ist folgendes erlaubt:
 - A-Z, a-z
 - 0-9
 - _ (Unterstrich)
 - \$ (Dollarzeichen)
 - Arbeitet nicht mit dem Unicode Zeichensatz. Ich rate von der Verwendung von den Zeichen ab, da dies unerwartete Resultate/Fehler erzeugen kann
- Das erste Zeichen muss ein Buchstabe oder ein Unterstrich sein
- Schlüsselwörter von Java dürfen nicht verwendet werden
 - new, for, if, int, enum, float, while, try,
 - https://docs.oracle.com/javase/tutorial/java/nutsandbolts/_keywords.html

Namenskonventionen für Variablen

- Ein Variablenname sollte
 - Mit einem Kleinbuchstaben anfangen
 - wetter, id, name, vorname
 - Sollte nicht mit den Spezialzeichen anfangen
 - **nicht:** _wetter, _id...
 - Bei mehreren Wörtern dem lowerCamelCase Prinzip folgen
 - meineVariable, firstName, lastName, temperatureInCelsius...
 - Einzelne Buchstaben **vermeiden**
 - **nicht:** g, h, e, k,
 - Kann aber durchaus Sinn machen, zb: x,y,z für Achsen in einem 3D-Raum

Tipp

Schreibt euren Code / eure Kommentare auf Englisch

1. Ihr lernt passiv neue, englische, Wörter
2. Übung macht den Meister
3. Ihr könnt den Code mit Freunden oder im Internet teilen. Bei Fragen (online) wird Englisch (quasi) vorausgesetzt
4. Gute Namensgebung von Variablen kann schwierig sein;
Auch hier: Übung macht den Meister

```
// Beispiel 1
```

```
int numberOfStudentsWhoHaveCompletedTheCourseSuccessfully; // Descriptive, but horrible to read
```

```
int graduates; // Still descriptive, but (way) better!
```

```
// Beispiel 2
```

```
int numberOfDaysWithContinuousRainfall;
```

```
int consecutiveRainyDays;
```

Erstes Schlüsselwort (Keyword)

- Wird **vor** dem Datentyp der Variable gesetzt
- Wenn wir eine Variable mit dem Schlüsselwort «**final**» markieren, kann der Wert nur einmal gesetzt werden
 - Daher: Auch nicht mehr nachträglich geändert werden!
- Final-variablen werden meistens ausdrücklich aus Grossbuchstaben geschrieben

```
final float PI = 3.14159f;
```

```
final int ERDOBERFLAECHE_IN_KM = 510100933;
```

Aufgaben

- Schaut euch den Link bei der Slide «char» an:
 - Den Wikipedia-link: Was ist der Unicode für den Buchstaben «X»?
- Variablen

Fragen?