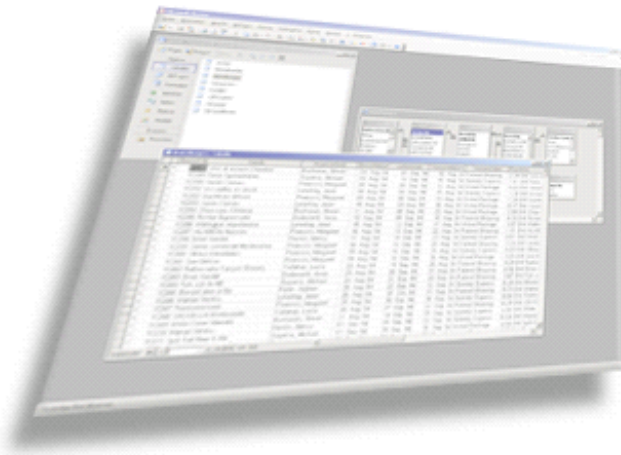


Einführung in Java





Installation I

1.

Java Development Kit herunterladen und installieren: JDK





Arbeiten mit dem Java Editor I

1.

Installation vom Java Editor

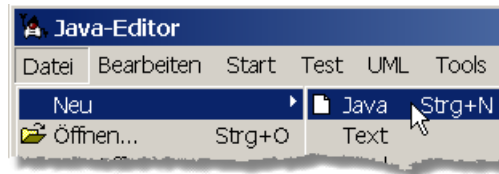




Arbeiten mit dem Java Editor II

2.

Wählen von dem Menüpunkt „Datei“ „Neu“ „Java“ oder Strg+N



3.

Wählen von dem Menüpunkt „Datei“ „Neu“ „Java“ oder Strg+N und Eingeben des gesamten Quellcodes

```

Java-Editor - [C:\Documents and Settings\MrBig\workspace\FirstProgramm\src\Add2Integers.java]
Datei Bearbeiten Start Test UML Tools Fenster Hilfe
Programm Kontrollstrukturen Datentypen AWT Swing 1 Swing 2 Layout
Add2Integers.java*
1 import java.util.Scanner; // program uses class Scanner
2 public class Add2Integers {
3
4     public static void main(String[] args) {
5         // create Scanner to obtain input from command window
6
7         Scanner input = new Scanner( System.in );
8         int number1; // first number to add
9         int number2; // second number to add

```



Kommentare

Einzeilige Kommentare beginnen mit //:

```
// Kommentar
```

**Mehrzeilige Kommentare werden durch /* und*/
eingeschlossen**

```
/* Kommentar1  
Kommentar2 */
```



Mehrzeilige Ausgabe I

print: Zeile ohne Zeilenumbruch
println: Zeile mit Zeilenumbruch

```
public class FirstProgram {  
    /**  
     * @param args  
     */  
    public static void main(String[] args) {  
        // TODO Auto-generated method stub  
        System.out.print( "Welcome to " );  
        System.out.println( "Java Programming!" );  
    }  
}
```

Ausgabe:

```
<terminated> FirstProgram [Java Application] C:\Program File  
Welcome to Java Programming!
```



Mehrzeilige Ausgabe II

/n: Steuerzeichen für Zeilenumbruch

```
public class FirstProgram {  
    /**  
     * @param args  
     */  
    public static void main(String[] args) {  
        // TODO Auto-generated method stub  
        System.out.println( "Welcome\nto\nJava\nProgramming!" );  
    }  
}
```

Ausgabe:

```
<terminated> FirstProgram [Java Application] C:\Program Files\Jav  
Welcome  
to  
Java
```

Escape sequence Description

<code>\n</code>	Newline. Position the screen cursor at the beginning of the next line.
<code>\t</code>	Horizontal tab. Move the screen cursor to the next tab stop.
<code>\r</code>	Carriage return. Position the screen cursor at the beginning of the current line do not advance to the next line. Any characters output after the carriage return overwrite the characters previously output on that line.
<code>\\</code>	Backslash. Used to print a backslash character.
<code>\"</code>	Double quote. Used to print a double-quote



Arithmetische Operatoren

Operation	Operator	Ausdruck
Addition	+	$a+b$
Subtraktion	-	$a-b$
Multiplikation	*	$a*b$
Division	/	a/b
Rest	%	$a\%b$



Einfache Datentypen

Datentyp	Wertbereich	Art
boolean	true oder false	
byte	$-2^7 \dots 2^7 - 1$	ganze Zahlen
short	$-2^{15} \dots 2^{15} - 1$	ganze Zahlen
int	$-2^{31} \dots 2^{31} - 1$	ganze Zahlen
long	$-2^{63} \dots 2^{63} - 1$	ganze Zahlen
double	$-10^{38} \dots 10^{38}$	Dezimalzahlen
float	$-10^{308} \dots 10^{308}$	Dezimalzahlen
char	1 Zeichen	Zeichen

Deklaration und eine Zuweisung dürfen in einer Zeile erfolgen!

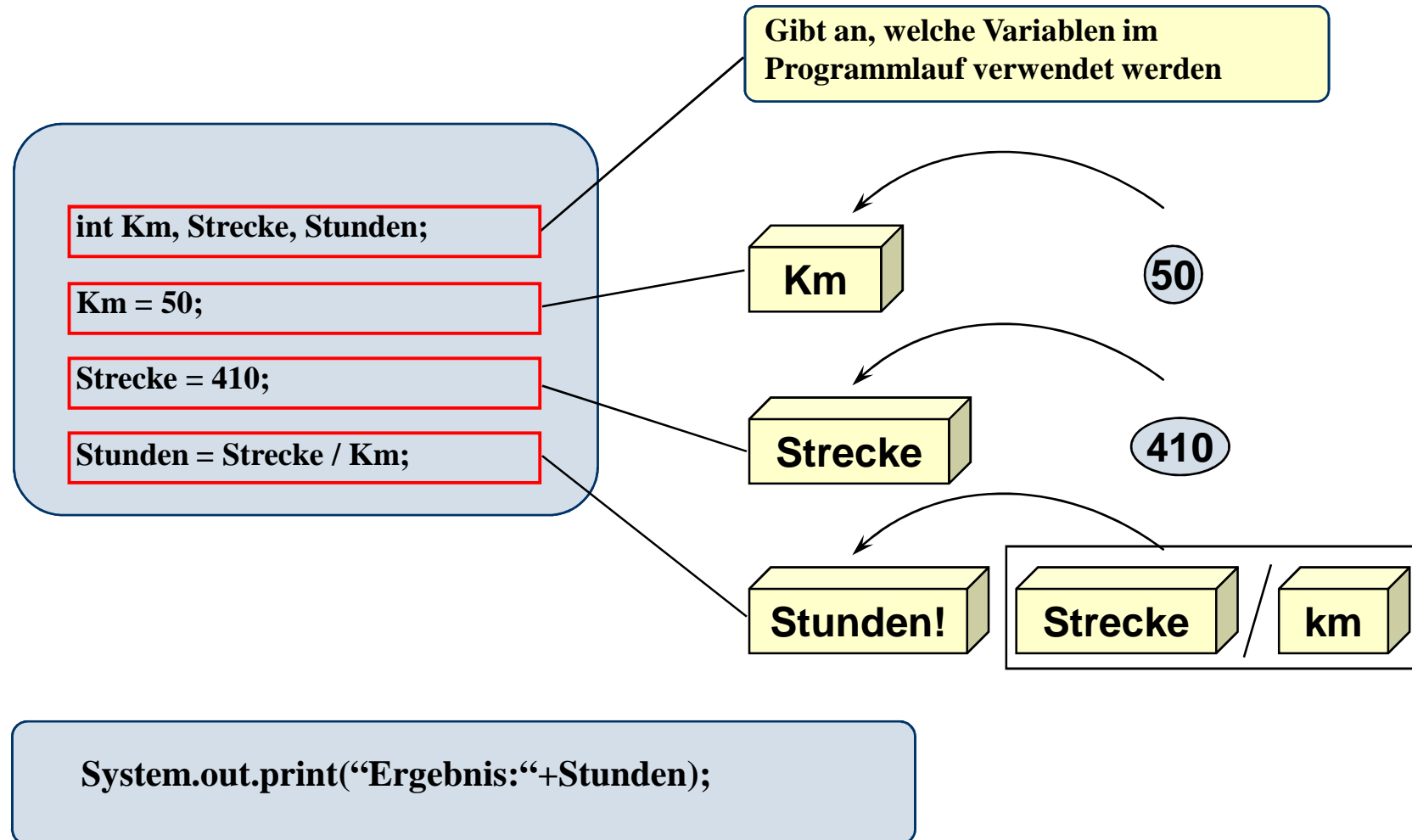
```
boolean result = true;  
char capitalC = 'C';  
byte b = 100;  
short s = 10000;  
int i = 100000;
```

```
int decVal = 26;  
int hexVal = 0x1a;
```

```
double d1 = 123.4;  
double d2 = 1.234e2;  
float f1 = 123.4f;
```



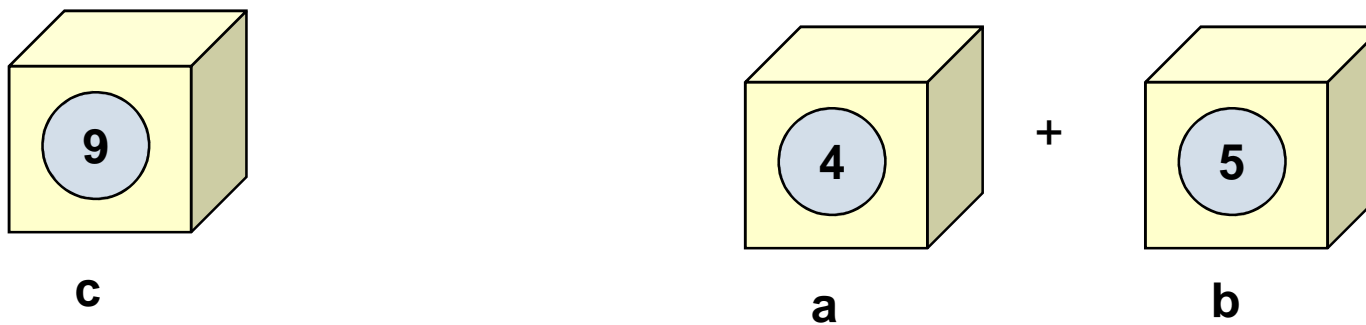
Beispiel zu Variablen I





Beispiel zu Variablen II

```
a=4;  
b=5;  
c=a + b;
```



Die Summe der Inhalte der Variablen a und b werden der Variable c zugeordnet!



Beispiel zu Variablen III

	a	b	c	
<code>int a, b, c;</code>	0	0	0	Var. festlegung(deklaration)
<code>a = 3;</code>	3	0	0	a wird 3 zugewiesen
<code>b = 4;</code>	3	4	0	b wird 4 zugewiesen
<code>c = a + b;</code>	3	4	7	c wird 3 + 4 zugewiesen
<code>a = a * c;</code>	21	4	7	a wird 3 * 7 zugewiesen
<code>system.out.print(a - b);</code>	21	4	17	Ausgabe
<code>b = b*b;</code>	21	16	7	b wird 4*4 zugewiesen



Regeln für professionelle Programmierung

Pro Zeile eine Variable deklarieren

Aussagekräftige Variablennamen: Selbstdokumentierende Programme

Variablen beginnen mit Kleinbuchstaben und alle anderen Wörter mit Großbuchstaben z.B. firstName

Kommentare und Leerzeilen helfen, das Programm besser lesbar zu machen

Geben Sie beim Erstellen zunächst { und } ein und füllen anschließend den Quellcode zwischen die Klammern ein

Tab= 3 Leerzeichen



Debuggen

```
1 public class ...
2
3
4 /**
5  * @param args
6  */
7 public static void main(String[] args) {
8     float länge,breite, fensterfläche;
9     länge = 16;
10    breite = 14;
11    fensterfläche= länge*breite/10;
12    System.out.printf( "Erg: %f", fensterfläche);
13 }
14
```

Eine Anweisung ausführen

Haltepunkt setzen oder löschen

Inhalte von Variablen anzeigen

Aktuelle Zeile, die als nächstes ausgeführt wird

“Haltepunkt”

**Überwachte Ausdrücke:
Zufügen mit Doppelklick!**

Durch Klick erzeugen

Automatisch werden die Var. angezeigt