

Methoden, Globale Variablen, Schleifen

Methoden:

```
public class MyProgram {

    public static void main(String[] args) {
        System.out.println(mul2(add10(3)));
    }

    public static float mul2(float x) {
        return 2*x;
    }

    public static float add10(float x) {
        return x+10;
    }

}
```

Globale Variablen:

```
public class MyProgram {

    static int anzahl = 0;

    public static void main(String[] args) {
        System.out.println(inc());
        System.out.println(inc());
    }

    public static int inc() {
        return ++anzahl;
    }

}
```

Aufzählschleife:

```
int summe = 0;
for (int i = 1; i <= 10; i++) {
    summe += i;
}
```

Kopfgesteuerte Schleife:

```
int summe = 0;
int i = 1;
while (i <= 10) {
    summe += i;
    i++;
}
```

Fußgesteuerte Schleife:

```
int summe = 0;
int i = 1;
do {
    summe += i;
    i++;
} while (i <= 10);
```

Bsp 1: Methoden

Wie sind Methoden in Java aufgebaut? Was ist eine Methodensignatur, was ein Methodenkopf und was ein Methodenkörper? Geben Sie ein Beispiel.

Bsp 2: Schaltjahr - Methode

Verpacken Sie den Code zur Schaltjahrberechnung aus Blatt 2 Beispiel 4 in eine Methode namens `istSchaltjahr`. Überlegen Sie sich eine sinnvolle Signatur für die Methode.

Bsp 3: Schaltjahr - JavaDoc

Dokumentieren Sie Ihr Schaltjahr-Programm aus Beispiel 2 mit sinnvollen JavaDoc-Kommentaren für die Klasse und die Methode, generieren Sie die zugehörigen HTML-Dokumentationsdateien und sehen Sie sich diese in einem Browser an.

Bsp 4: Schleifen

Welche Arten von Schleifen kennen Sie. Geben Sie Beispiele.

Bsp 5: Primzahl

Lesen Sie eine ganze Zahl von der Tastatur ein und bestimmen Sie ob diese Zahl eine Primzahl ist. Schreiben Sie dazu eine Methode `istPrimzahl`.

Bsp 6: Globale Variable

Was ist eine globale Variable bzw. Klassenvariable? Wie wird eine globale Variable deklariert? Geben Sie ein Beispiel an.

Bsp 7: Scope

Was gibt das folgende Programm aus und warum?

```
public class Scope {
    static int x;

    static void foo(int x) {
        x = 20;
        System.out.println(x); // ?
    }

    public static void main(String[] args) {
        x = 10;
        System.out.println(x); // ?
        foo(x);
        System.out.println(x); // ?
    }
}
```

Bsp 8: Globale Variable vs. Lokale Variable

Was ist der Unterschied zwischen einer globalen und einer lokalen Variable?

Bsp 9: Ziffernsumme

Lesen Sie eine ganze Zahl von der Tastatur ein und bestimmen Sie die Ziffernsumme dieser Zahl. Verwenden Sie dazu eine Methode mit sinnvoller Signatur.

Bsp 10: Zahl umkehren

Lesen Sie eine ganze Zahl von der Tastatur ein und bilden Sie eine neue Zahl durch Umkehren der Reihenfolge der Ziffern. Aus beispielsweise 517 soll die Zahl 715 werden.

Bsp 11: Pi

Bestimmen Sie die Zahl π nach folgender Reihe mit der Genauigkeit ε (Gottfried Wilhelm Leibnitz, 1646-1716).

$$\frac{\pi}{4} = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \frac{1}{9} - \frac{1}{11} + \dots$$

Brechen Sie die Berechnung ab, wenn gilt: $|\pi_{i+1} - \pi_i| \leq \varepsilon$.

π_i ist die i-te Näherung von π , z.B. ist $\pi_3 = 1 - \frac{1}{3} + \frac{1}{5}$.

Bsp 12: Dreieck

Schreiben Sie ein Programm, das ein aus *-Zeichen bestehendes Dreieck ausgibt. Lesen Sie die Anzahl der Zeilen, die für die Generierung des Dreiecks verwendet werden sollen, von der Tastatur ein. Zum Beispiel:

Zeilen = 9

```

*
**
***
****
*****
*****
****
***
**
*
    
```

Zeilen = 10

```

*
**
***
****
*****
*****
*****
*****
****
***
**
*
    
```

Bsp 13: Raute

Schreiben Sie ein Programm, das eine aus *-Zeichen bestehende Raute ausgibt. Die Anzahl der Zeilen sollen von der Tastatur eingelesen werden. Zum Beispiel:

Zeilen = 9

```

  *
 * *
* * *
 * *
  *
    
```

Zeilen = 10

```

  *
 * *
* * *
 * *
  *
    
```

Bsp 14: Taschenrechner

Schreiben Sie ein Programm, das einen Taschenrechner mit den Operationen +, -, *, / und q implementiert. Die folgende Ausgabesequenz zeigt die Verwendung:

```
Wert: 0
Operation: +5
Ergebnis: 5.0
Operation: *3
Ergebnis: 15.0
Operation: /2
Ergebnis: 7.5
Operation: ab
Unbekannte Operation!
Ergebnis: 7.5
Operation: q
Programm beendet!
```