

Einstieg Schleifen

Schleifen sind in der Programmierung Strukturen, die dazu dienen einen bestimmten Teil des Programmes/Quellcodes zu wiederholen.

Das Programm läuft also wie Sie es kennen von oben nach unten durch. Absolviert zunächst alle Anweisungen die sich vor der Schleife befinden (im Beispiel Anweisung 1-3), wiederholt dann die Anweisungen innerhalb der Schleife (im Beispiel Anweisung 4+5) so oft wie es von der Schleife vorgesehen ist und nachdem die Schleife wieder verlassen wurde werden die restlichen Anweisungen des Programmes ausgeführt (im Beispiel Anweisung 6+7).

Anweisung 1
Anweisung 2
Anweisung 3
Schleifenbeginn
Anweisung 4
Anweisung 5
Schleifenende
Anweisung 6
Anweisung 7

Es gibt in der Regel drei klassische Arten von Schleifen,

- die zählergesteuerte Schleife (oft auch for-Schleife genannt),
- die kopfgesteuerte Schleife (oft auch while-Schleife genannt) und
- die fußgesteuerte Schleife (oft auch do-while-Schleife genannt).

Sie werden nun zunächst **die zählergesteuerte Schleife** kennenlernen. Diese Schleife kommt vor allem immer dann zum Einsatz, wenn die Anzahl der Durchläufe der Schleife von Anfang an feststeht. Es soll also z.B.

- der Umsatz für alle 12 Monate berechnet werden oder
- die Provision in den 4 Quartalen ausgegeben werden oder
- das kleine Einmaleins der 7 ausgegeben werden oder
- 25 Leerzeilen ausgegeben werden usw.

In **Java** lautet die **Syntax** für die zählergesteuerte Schleife folgendermaßen:

for (Zählervariable=Anfangswert; Zählervariable<=Endwert; Zählervariable++)
{
Anweisungen;
}

Typischerweise wählt man als Zählervariable den Buchstaben `i` oder `lf` für Laufvariable.

Sie können die Variable bereits im Vorfeld mit allen anderen Variablen deklarieren oder aber direkt in der Schleife.

Beispiel 1

Das folgende Beispiel gibt 20 Sterne untereinander aus.

Die Zählervariable `i` wird vor der Schleife deklariert (`int i`) und zu Beginn des ersten Schleifendurchlaufs mit dem Anfangswert initialisiert (`i=1`).

Weil der Endwert natürlich noch nicht erreicht ist, läuft die Schleife zum ersten Mal durch und gibt ein Sternchen aus (`System.out.println("*")`).

Nach jedem Schleifendurchlauf wird die Zählervariable um eins erhöht (`i++`).

Dies passiert so lange (das Erhöhen der Zählervariablen und das Durchlaufen der Schleife) bis die Zählervariable den Endwert erreicht hat (`i<=20`). Sie können es sich auch so vorstellen, dass bei jedem Durchlauf überprüft wird, ob die Bedingung noch zutrifft und solange in diesem Fall `i` kleiner oder gleich 20 ist, wird die Schleife ausgeführt.

Wenn die Zählervariable den Endwert erreicht hat wird die Schleife noch einmal ausgeführt und dann verlassen.

```
public class BeispiellSterne
{
    public static void main (String argv[])
    {
        int i;
        for (i=1; i<=20; i++)
        {
            System.out.println("*");
        }
    }
}
```

Um die in diesem Falle 20 Durchläufe noch deutlicher zu machen, können Sie die Zeile innerhalb der Schleife auch folgendermaßen abändern.

```
System.out.println(i + ". *");
```

Schauen Sie sich in der Beispieldatei an was passiert. Schauen Sie sich auch an was passiert, wenn Sie statt `println` nur `print` verwenden.

```
System.out.print("*");
```

Beispiel 2

Das folgende Beispiel ist ein klein wenig komplexer als das erste. Die Aufgabenstellung lautet: Schreiben Sie ein Programm, das nach Eingabe eines Endwertes die Summe der Quadrate $1^2 + 2^2 + 3^2 + 4^2 + \dots + n^2$ bis zum eingegebenen Endwert (n) berechnet.

Zunächst muss also der Anwender den Endwert eingeben,

```
System.out.print("Bitte geben Sie den Endwert ein: ");  
n = input.nextInt();
```

danach wird der Inhalt der Schleife beginnend mit dem Startwert 1 so oft wiederholt, bis der eingegebene Endwert erreicht wurde. Innerhalb der Schleife wird das Quadrat des jeweiligen Wertes des Zählers (also zunächst 1, dann 2, dann 3 usw.) berechnet und in der Variablen `ergebnis` abgespeichert. Anschließend wird das Ergebnis zur Summe hinzuaddiert (`summe = summe + ergebnis`), denn wir sollen ja am Ende die Summe aller Quadrate ausgeben.

```
for(int i=1; i<=n; i++)  
{  
    ergebnis = i*i;  
    summe = summe + ergebnis;  
}
```

Im Anschluss an die Schleife wird genau diese Summe ausgegeben.

```
System.out.println("Die Summe ist: " + summe);
```

Wichtig ist, dass die Ausgabe nicht IN der Schleife erfolgt, denn dann würde die Summe bei jedem Schleifendurchlauf ausgegeben werden. Probieren Sie es gerne aus!

Außerdem haben Sie vielleicht gesehen, dass in diesem Beispiel die Deklaration der Variablen `i` im Kopf der Schleife erfolgt. Dies hat zu Folge, dass die Variable nur innerhalb der Schleife existiert. Dies hat Vor- und Nachteile, um die wir uns aber später kümmern.

Beispiel 2 komplett mit Import, Scanner und Variablendeklarationen:

import java.util.Scanner;
public class Beispiel2Quadrate
{
public static void main(String[] args)
{
Scanner input = new Scanner(System.in);
int n = 0;
int ergebnis = 0;
int summe = 0;
System.out.print("Bitte geben Sie den Endwert ein: ");
n = input.nextInt();
for(int i=1;i<=n;i++)
{
ergebnis = i*i;
summe = summe + ergebnis;
}
System.out.println("Die Summe ist: " + summe);
}
}