

Wiederholungen in Java

Manchmal kommt es vor, dass man bestimmte Befehle mehrfach ausführen möchte oder einen Befehl solange ausführen möchte bis eine bestimmte Bedingungen erfüllt bzw. nicht mehr erfüllt ist.

» siehe Skript zu Bedingungen und Vergleichen

Bedingte Wiederholung mit while

Nicht immer weiß man vorher (also beim Programmieren), wie oft eine Wiederholung durchlaufen werden soll. Sondern man möchte bestimmten Code **solange wiederholen, bis ein bestimmtes Ereignis eintritt**.

Beispiel:

```
while( ! mario.beruehrt(pilz) ) {  
    mario.schrittNachVorne();  
}  
mario.hebeAuf(pilz);
```



Der Aufbau für die bedingte Wiederholung wird mit dem Schlüsselwort **while** eingeleitet, gefolgt von einem **runden Klammerpaar** und einem **geschweiften Klammerpaar**.

```
while( Bedingung ) {  
    // Befehle für die Wiederholung  
}  
// Hier geht es weiter
```

Solange die **Bedingung zu true ausgewertet** wird, wird weiter der Code in der while-Schleife wiederholt.

Sobald die **Bedingung zu false ausgewertet** wird, wird der Code an dieser Stelle ausgeführt.

Eine besondere Schleife ist die **while(true){ ... }** Schleife. Da hier keine echte Bedingung angegeben wird, sondern die Bedingung prinzipiell true ist, läuft diese Schleife so lange, bis das Programm geschlossen wird.

Wiederholung mit fester Anzahl mit for

Für die Kontrollstruktur der Zählschleife wird durch das Schlüsselwort **for** eingeleitet, gefolgt von einem **runden Klammerpaar** und einem **geschweiften Klammerpaar**.

```
for( int i = startwert; i < maximalwert; i = i + 1 ){  
    // Befehle für die Wiederholung  
}
```

In den runden Klammern stehen 3 Informationen (**mit Strichpunkt getrennt**):

Zählvariable und Startwert

int i = 0 Die lokale (Zähl-)Variable heißt i und startet mit dem Wert 0.

int i = 10 Die lokale (Zähl-)Variable heißt i und startet mit dem Wert 10.

Bedingung für das Ausführen der Befehle

Die Befehle in den geschweiften Klammern werden nur ausgeführt, wenn diese Bedingung **true** ist.

i < 5 Intervall von i von 0 bis 4 (nur sinnvoll beim Erhöhen der Zählvariablen)

i >= 0 Intervall von i von 10 bis 1 (nur sinnvoll beim Verringern der Zählvariablen)

Wert der Zählvariablen ändern

Wenn die Befehle in den geschweiften Klammern einmal ausgeführt wurden, dann muss der Wert der Zählvariablen verändert werden.

i = i + 1 Wert um 1 erhöhen (jede Zahl von 0 bis 4 = 5 Wiederholungen)

i = i - 2 Wert um 2 verringern (jede zweite Zahl von 10 bis 0 = 6 Wiederholungen)

Nutzung der Zählvariable

Auf den ersten Blick erschließt sich nicht, warum in Java die Zählwiederholung mit Startwert, Endwert und Schrittweite so kompliziert aufgebaut ist, denn man könnte sie ja auch einfach folgendermaßen definieren: Wiederhole X mal folgenden Befehl.

Der Sinn hierbei liegt darin, dass man die **lokale (Zähl-)Variable i** für viele Berechnungen oder Anwendungsfälle **nutzen kann**.

Beispiel: Zähle die ersten 100 Zahlen zusammen.

```
public int addiereDieErsten100Zahlen() {  
    int ergebnis;  
    ergebnis = 0;  
    for( int i = 1; i <= 100; i++) {  
        ergebnis += i;  
    }  
    return ergebnis;  
}
```

» Weitere Anwendungsbeispiele folgen in der 10. Jahrgangsstufe