

```
if (klassenstufe == 9) {
    fachwahl="IMP";
}
```

Zählschleifen in Java

Aus Scratch oder dem MIT App Inventor kennt ihr Kontrollstrukturen, die ein Programm einfacher machen. Natürlich gibt es diese auch in der Programmiersprache Java und wir können sie in unseren Programmen nutzen.

Wird eine Anweisung wiederholt, nutzen wir eine **Schleife**. Wissen wir, wie oft wiederholt werden soll, so bietet sich eine **Zählschleife** an.

In der Programmiersprache Java sieht der Rahmen einer **Zählschleife** folgendermaßen aus:

```
for (int i=0; i<=2; i++) {
    }
}
```

Es gibt dort eine **Zählvariable** *i*, für die man einen **Startwert**, einen **Endwert** und eine **Schrittweite** angeben kann. Zwischen den geschweiften Klammern steht das, was wiederholt werden soll. Die Zählvariable darf dabei genutzt werden.

In Klammern wird zuerst die Variable *i* deklariert (bekannt gemacht) und initialisiert (sie bekommt den Wert 0 zugewiesen). Das `int` gibt an, dass *i* nur ganze Zahlen als Wert annehmen kann.

Dann wird mit `i <= 2` vorgegeben, dass *i* höchstens 2 sein soll.

Das `i++` könnte auch `i = i + 1` heißen und bedeutet, dass *i* jedes Mal um 1 erhöht wird. Also nimmt die Variable *i* die Werte 0, 1 und 2 in unserem Beispiel an.

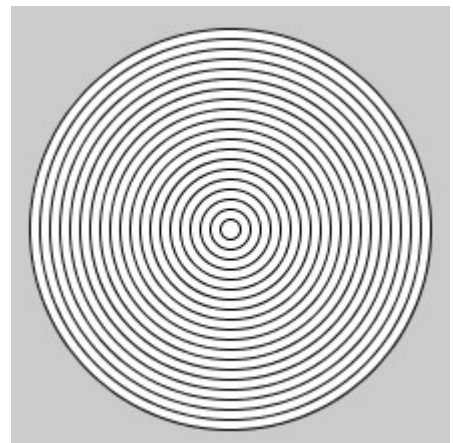
Aufträge:

1. Ergänze in der Vorlage `figurC_mitSchleife` aus dem Tauschverzeichnis die Methode `zeichneBildC()` so, dass zunächst die drei Kreise in das Quadrat gezeichnet werden. Nutze dazu eine Zählschleife.
2. Für die drei Quadrate in dem anderen Kreis kannst du auch eine Schleife verwenden. Suche wieder nach den Regelmäßigkeiten für den Startpunkt der Quadrate und für die Seitenlänge (bzw. Breite) und schreibe dein Programm um. (Bei deiner Lehrerin gibt es **Hilfekarten**, wenn du Hilfe brauchst!)

Weitere Übungen

Nutze für die weiteren Übungen eine neue Datei, in der du die eigenen Methoden schreibst und in der `setup()`-Methode aufrufst.

3. Schreibe eine Methode `zeichneQuadrate()`, die 10 Quadrate gleicher Größe direkt nebeneinander zeichnet.
4. Schreibe eine Methode `zeichneKreise()`, die 20 Kreise ineinander, die immer kleiner werden, zeichnet.



Bilder: Screenshots von Ausführung des Programms „zehn_quadrate“ (Eisenmann)

*** Schreibe die beiden Methoden aus 3. und 4. so, dass man ihnen die Anzahl der Quadrate bzw. Kreise übergeben kann. Hol dir dazu das Infoblatt zu Methoden in Java.

Lösungen

1. Diese Aufgabe wurde schon gemeinsam im Unterricht vorbesprochen. Ergebnis:

```

rect(10, 10, 100, 100);
for (int i=0; i<=2; i++) {
    ellipse(60, 60, 80-i*20, 80-i*20);
}

```

2. In den Hilfekarten steht der ausführliche Lösungsweg. Ergebnis:

```

ellipse(60, 170, 100, 100);
for (int i=0; i<=2; i++){
    rect(30+i*10, 140+i*10, 60-i*20, 60-i*20);
}

```

Gesamtes Programm für Figur C mit Aufruf der Methode:

```

void zeichneBildC() {
    rect(10, 10, 100, 100);
    for (int i=0; i<=2; i++) {
        ellipse(60, 60, 80-i*20, 80-i*20);
    }

    ellipse(60, 170, 100, 100);
    for (int i=0; i<=2; i++) {
        rect(30+i*10, 140+i*10, 60-i*20, 60-i*20);
    }
}

void setup() {
    size(120, 250);
    zeichneBildC();
}

```

3. Mögliches Ergebnis:

```

void zeichneQuadrate() {
    for (int i=0; i<=9; i++) {
        rect(30+i*30, 50, 30, 30);
    }
}

```

4. Mögliches Ergebnis:

```

void zeichneKreise() {
    for (int i=0; i<=19; i++) {
        ellipse(200, 200, 200-i*10, 200-i*10);
    }
}

void setup() {
    size(400, 400);
    zeichneQuadrate();
    zeichneKreise();
}

```

Mögliche Lösungen der Differenzierungsaufgaben:

```
void zeichneQuadrate(int anz) {  
    for (int i=0; i<=anz-1; i++) {  
        rect(30+i*30, 50, 30, 30);  
    }  
}  
  
void zeichneKreise(int anz) {  
    for (int i=0; i<=anz-1; i++) {  
        ellipse(200, 200, 200-i*10, 200-i*10);  
    }  
}  
  
void setup() {  
    size(800, 400);  
    zeichneQuadrate(20);  
    zeichneKreise(10);  
}
```

Natürlich kann man auch noch die Seitenlänge der Quadrate oder den Abstand zwischen den Kreisen übergeben lassen...

```
void zeichneQuadrate(int anz, int s) { // Anzahl der Quadrate und Seitenlänge werden übergeben  
    for (int i=0; i<=anz-1; i++) {  
        rect(30+i*s, 50, s, s);  
    }  
}  
  
void zeichneKreise(int anz, int d) { // Anzahl der Kreise und Abstand werden übergeben  
    for (int i=0; i<=anz-1; i++) {  
        ellipse(200, 200, 200-i*d, 200-i*d);  
    }  
}  
  
void setup() {  
    size(800, 400);  
    zeichneQuadrate(20, 10);  
    zeichneKreise(10, 5);  
}
```