

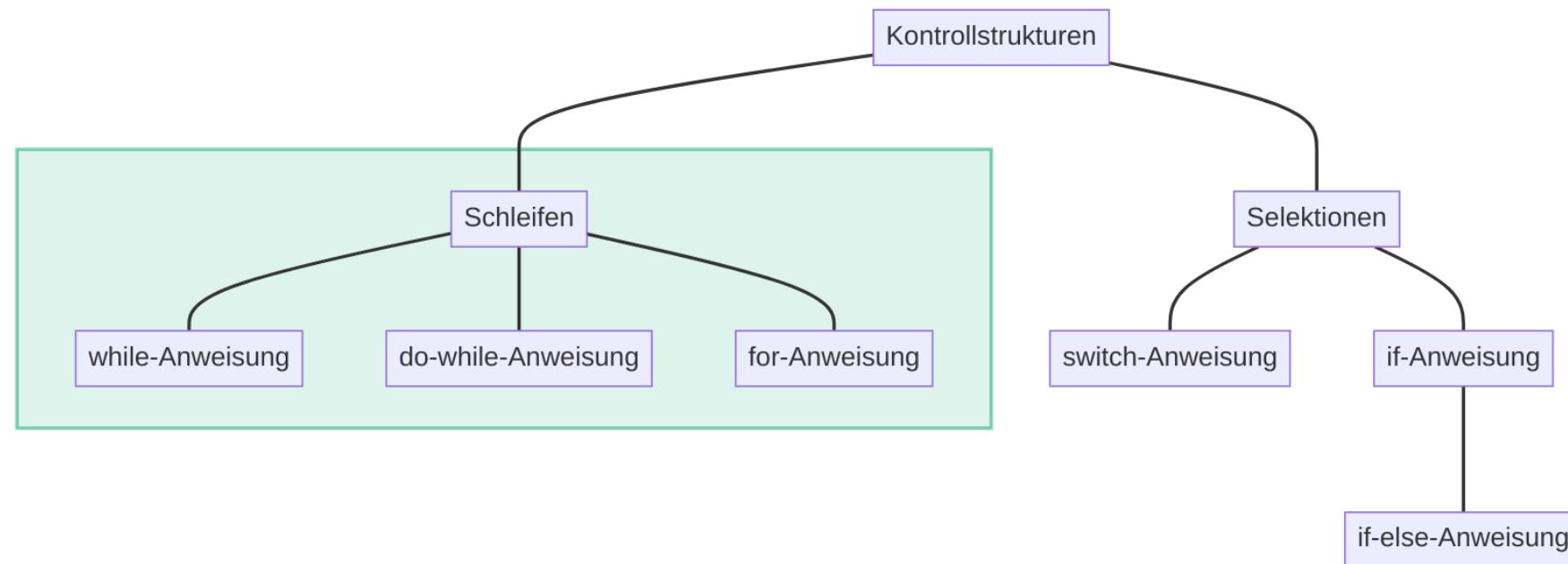


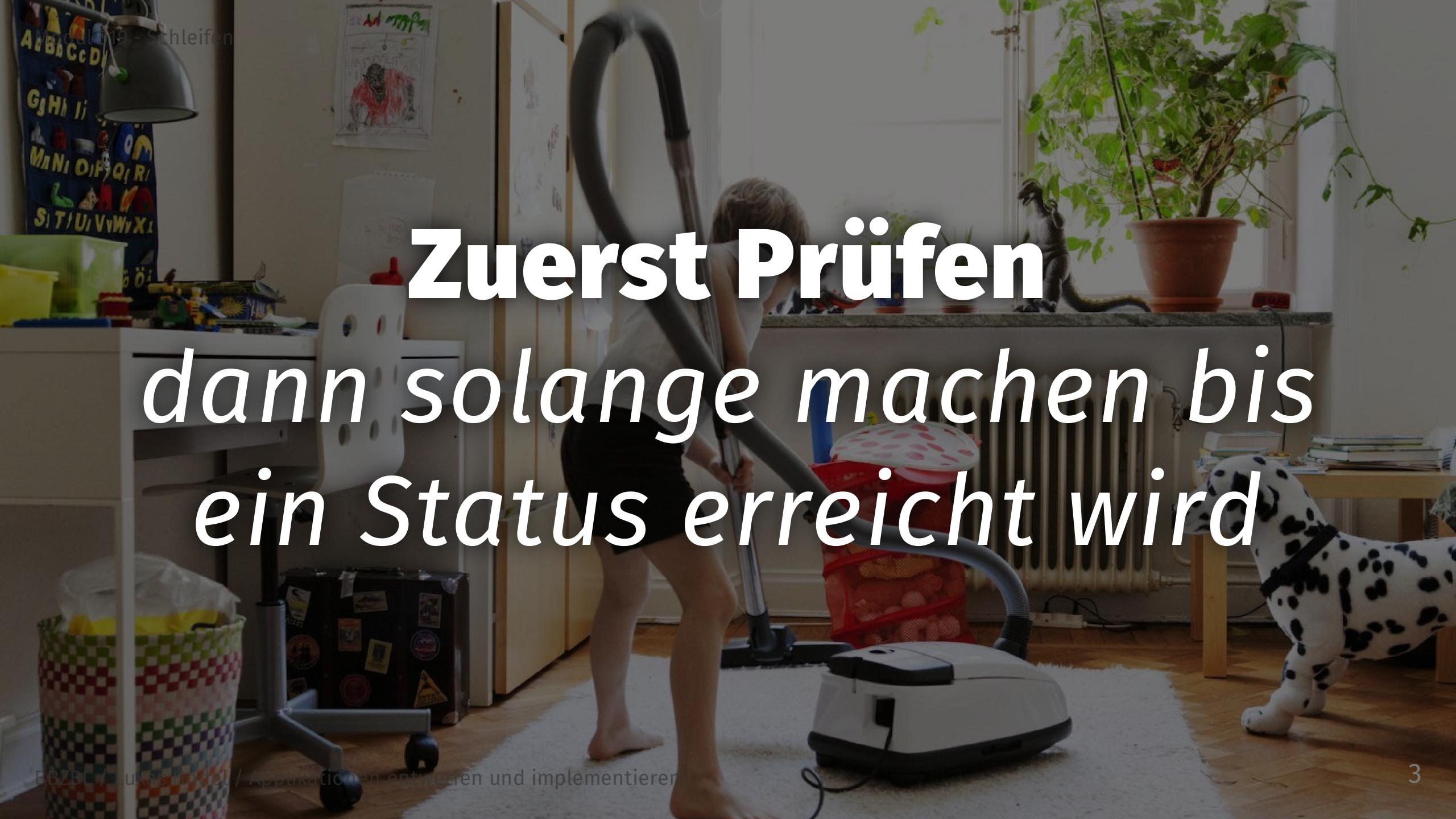
# Schleifen und Wiederholungen

# Kontrollstrukturen

Vergleichs- und logische Operatoren kommen ebenfalls zum Einsatz, wenn man etwas

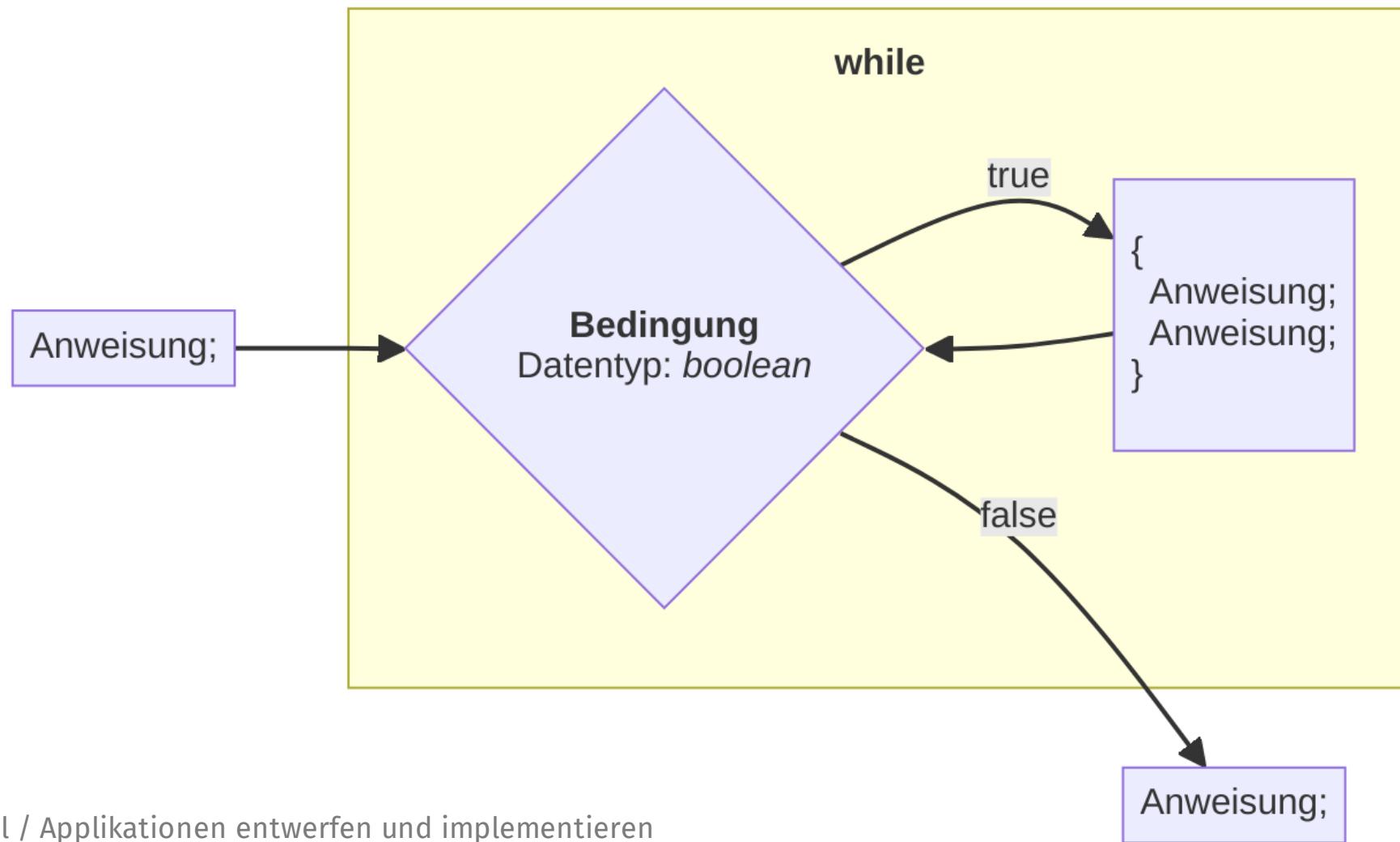
nur **unter einer bestimmten Bedingung wiederholen** soll.





Zuerst Prüfen  
dann solange machen bis  
ein Status erreicht wird

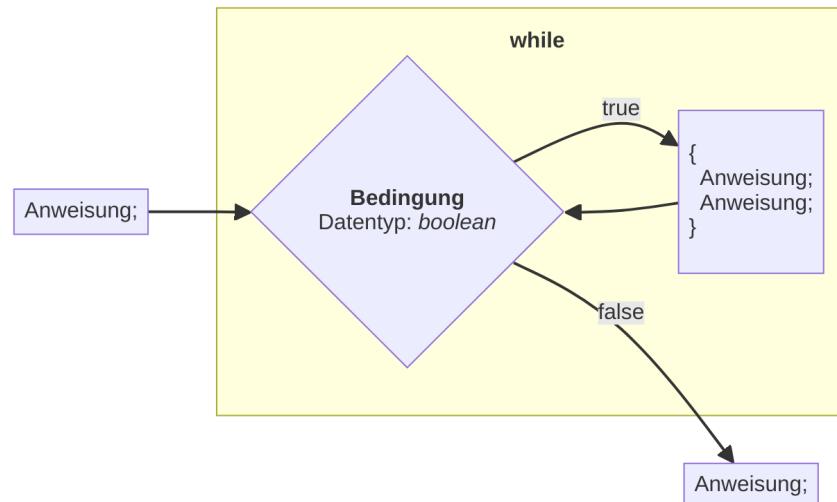
# while - Schleife



# while - Schleife

## Schema

```
while (<Bedingung>) {  
    <Anweisung>;  
    <Anweisung>;  
}
```



## Beispiel

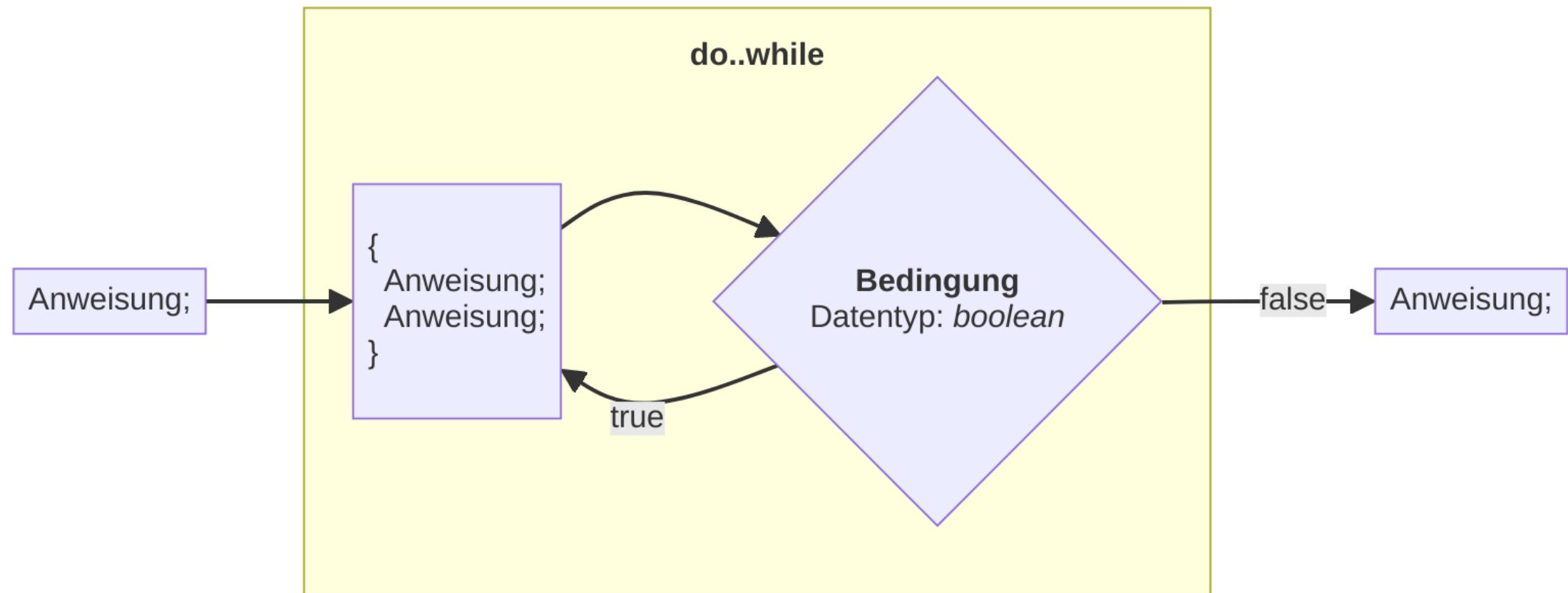
```
int i = 5;  
while (i > 0) {  
    System.out.println("i: " + i);  
    i--;  
}
```

```
// Ausgabe:  
// i: 5  
// i: 4  
// i: 3  
// i: 2  
// i: 1
```

A photograph of a person from the side, wearing a striped shirt, holding a bow and arrow, aiming it towards a blurred archery target in the distance. The background is a dark, out-of-focus green field.

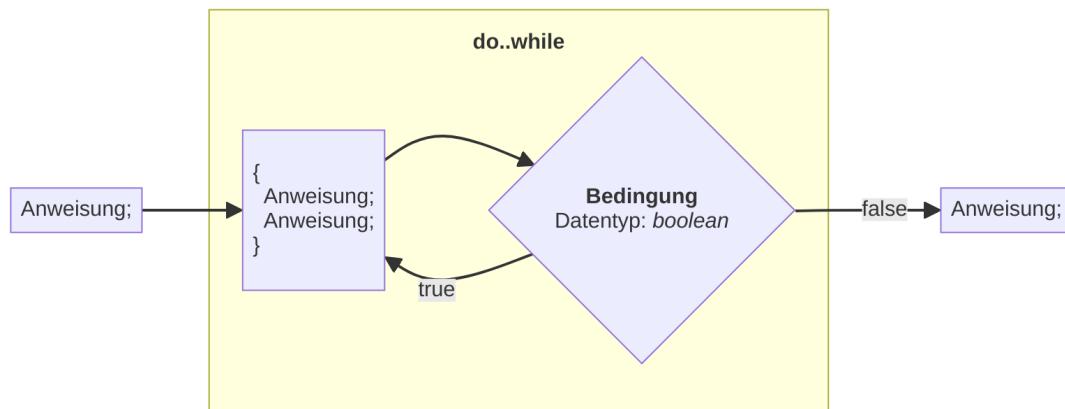
Zuerst Machen  
dann prüfen bis ein Status  
erreicht wird

# do..while - Schleife



# do .. while - Schema / Beispiel

```
do {
    <Anweisung>;
} while (<Bedingung>);
```



```
String gedanken = "Kaffee";
String eingabe = "";
do {
    if (!eingabe.equals("")) {
        System.out.println("leider falsch :(");
    }
    System.out.println("was denke ich?: ");
    eingabe = StdInput.readString();
} while (gedanken.equals(einbage));
System.out.println("jaaa");

// Ausgabe:
// Was denke ich?
// > quatsch
// leider falsch :(
// Was denke ich?
// > Kaffee
// jaaa
```

do . . while

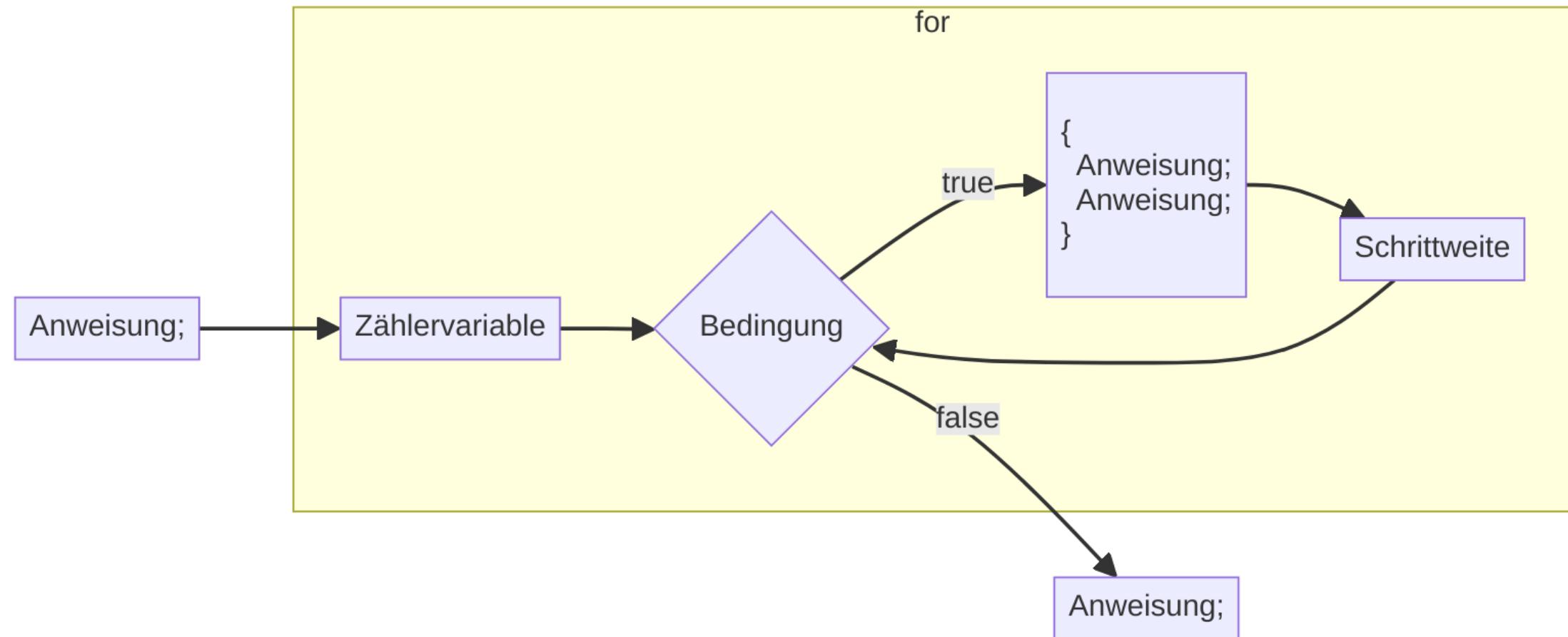
# **Abfrage und Verifikation von Benutzereingaben**



A black and white photograph showing a child's lower body from the waist down. The child is wearing a patterned dress and colorful sneakers. They are standing on a chalk-drawn hopscotch grid on a paved surface. The grid consists of several rectangles outlined in chalk, with some numbers and letters inside them. The child's right foot is positioned on the first square of the grid.

# Eine bekannte Reihe/Liste durchlaufen

# for - Schleife

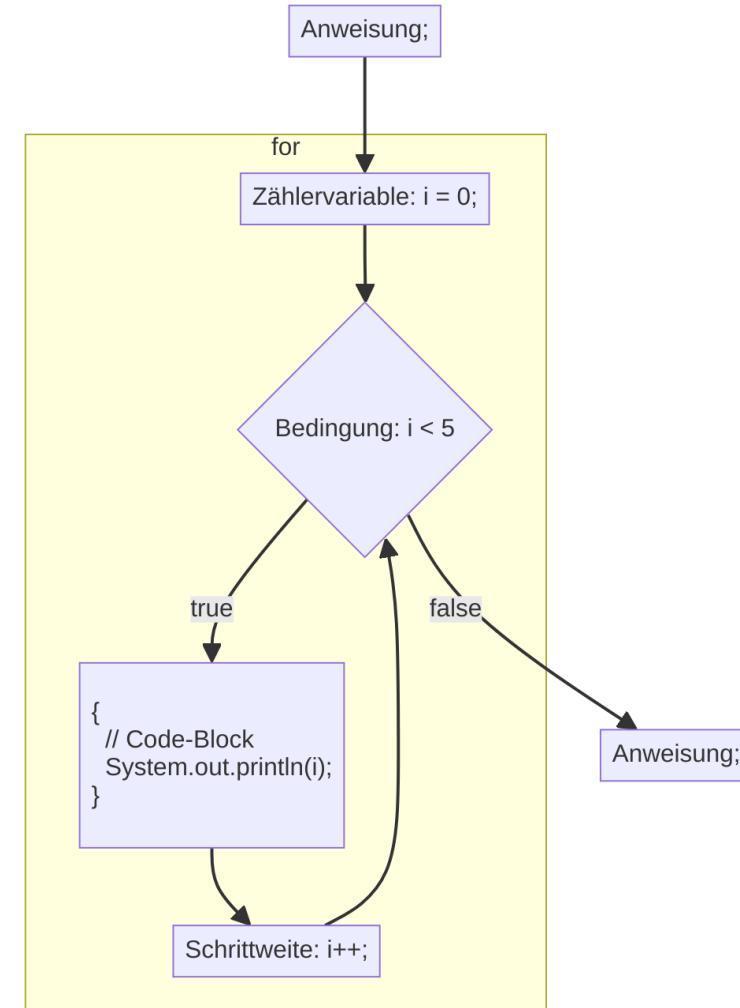


# for - Schema / Beispiel

```
for (
    <Zählervariable>;
    <Bedingung>;
    <Schrittweite>
) {
    <Anweisungen, Sequenz>
}
```

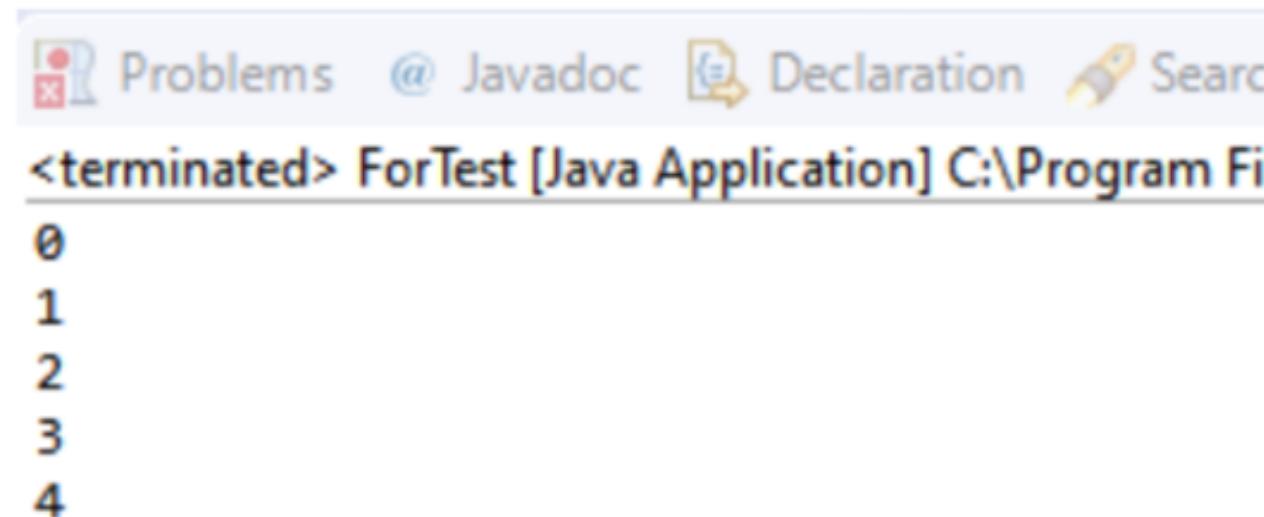
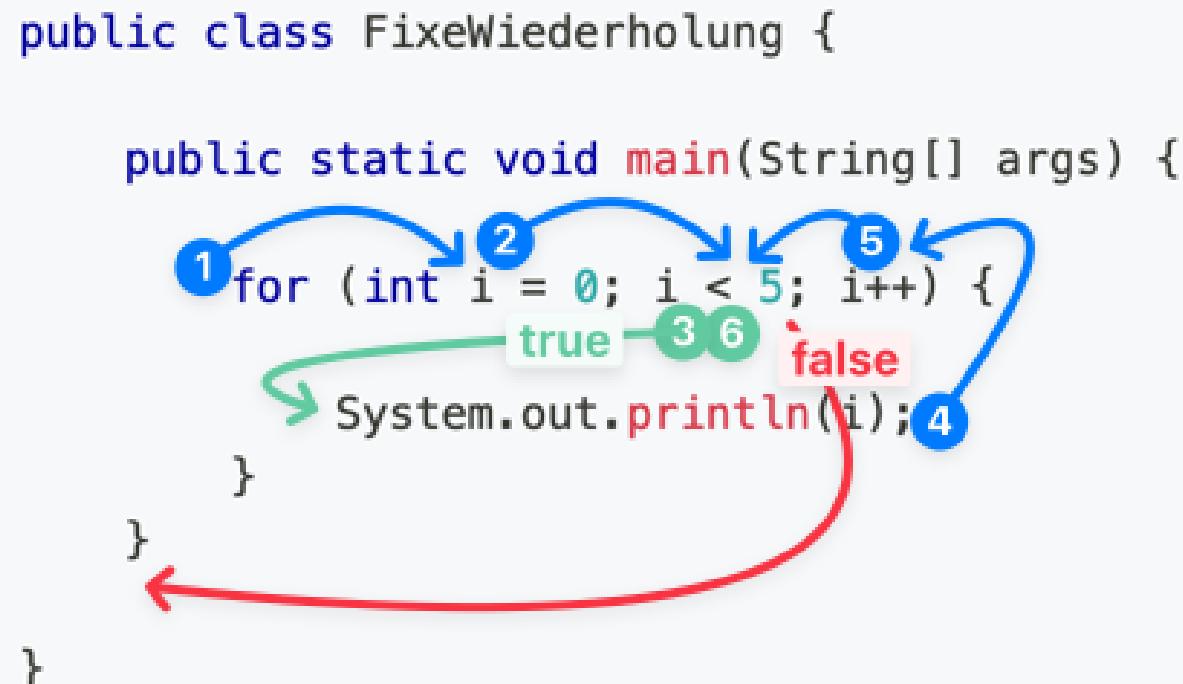
Beispiel:

```
for (int i = 0; i < 5; i++) {
    System.out.println(i);
}
```



# for - Ablauf und Ausgabe

```
public class FixeWiederholung {  
  
    public static void main(String[] args) {  
  
        1 for (int i = 0; i < 5; i++) {  
            2 true  
            3 System.out.println(i);  
            4 false  
        }  
    }  
}
```



```
0  
1  
2  
3  
4
```

# for vs while für Aufzählungen

for:

```
for (int i = 0; i < 5; i++) {  
    System.out.println(i);  
}
```

- ✓ Variable `i` nur innerhalb der Schlaufe sichtbar
- ✓ kürzer

while:

```
int i = 0;  
while (i < 5) {  
    System.out.println(i);  
    i++;  
}
```

- 😱 Variable `i` wird **ausserhalb der Schleife** initialisiert