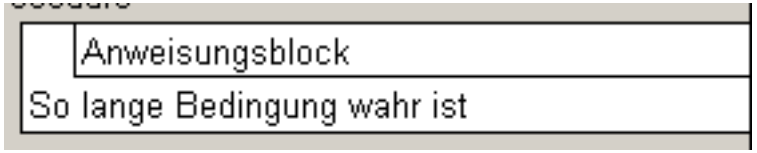
Java Schleifen:
`while / do...while`

kopfgesteuerte Schleife ("while ...")

Bedingung wird im SchleifenKOPF geprüft
*(d.h.: **vor** erstmaliger Ausführung d. Anweisungsblocks)*

fußgesteuerte Schleife ("do .... while")



Bedingung wird im SchleifenFUSS geprüft
*(d.h.: **nach** erstmaliger Ausführung d. Anweisungsblocks)*
→ Fußgesteuerte Schleife wird immer mindestens 1x ausgeführt

Syntax von `while`:

```
while (Bedingung)
{
   Anweisung
}
```

Beispiel: Zahlen von 0 bis 10 ausgeben

```
int zaehler = 0;
while(zaehler <= 10)
   {
      System.out.println(zaehler);
      zaehler++;
   }
```

```
1
2
3
4
5
6
7
8
9
10
```

Syntax von `while`:

```
while (Bedingung)
{
   Anweisung
}
```

Beispiel: Zahlen von 0 bis 10 ausgeben

```
int zaehler = 0;
while(zaehler <= 10)
   { System.out.println(zaehler);
     zaehler++;    }
```

```
1
2
3
4
5
6
7
8
9
10
```

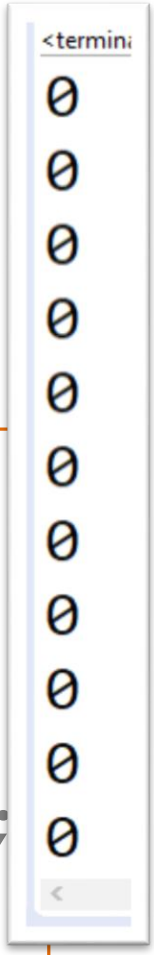
Abbruchbedingung

Syntax von `while`:

```
while (Bedingung)
{
   Anweisung
}
```

Beispiel: Zahlen von 0 bis 10 ausgeben

```
int zaehler = 0;
while(zaehler <= 10)
   { System.out.println(zaehler);
     // zaehler++;   }
```



Häufigster Fehler:
Abbruchbedingung wird nicht erfüllt
→ Programm gerät in Endlosschleife

Syntax von `do ... while`:

```
do
{
   Anweisung
}
while (Bedingung)

// fortfahren, sobald Bedingung falsch ist!
```

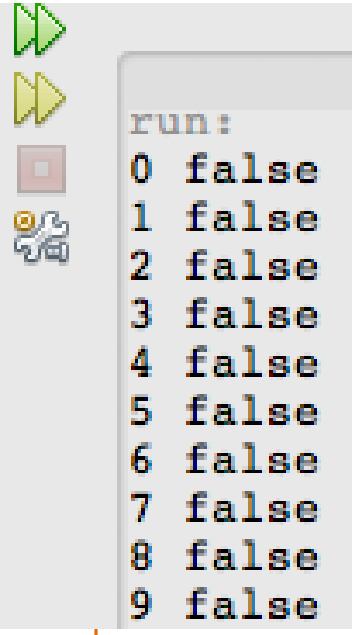
Syntax von `do ... while`:

```
do
{
    Anweisung
}
while (Bedingung)
```

Beispiel: Zahlen von 0 bis 9 ausgeben

```
int zaehler = 0;
do
{
   System.out.println(zaehler);
   zaehler++;
}
while(zaehler < 10)
```

Beispiel: Zahlen von 0 bis 9, Abbruchbedingung mit boolscher Variable




```
int zaehler = 0;
boolean abbruchbedingung = false;

do
{
   System.out.print(zaehler);
   System.out.println(" " + abbruchbedingung);
   zaehler++;
   if (zaehler == 10)
   {
      abbruchbedingung = true;
   }
}
   while (!abbruchbedingung);
   // oder: while (abbruchbedingung == false);
```

Bedingungen verknüpfen

```
// UND-Verknüpfung
while (x > 0 && y > 0)
{
    //
}
```

```
// ODER-Verknüpfung
while (x > 0 || y > 0)
{
    // ...
}
```