

# Zufallszahlen und Zufallsgenerator

Wir werden bald einen ganzen Schwarm von Fischen steuern wollen. Dabei ist es sehr ungünstig, wenn wir jeden einzelnen Fisch selber platzieren müssen.

Es wäre schön, wenn wir das Auswählen einer Startposition dem Rechner überlassen könnten.

➡ Die Position eines jeden Fisches soll zufällig bestimmt werden.

In Java gibt es dafür eine eigene Bibliotheksklasse, die importiert werden muss. In Processing ist diese Klasse direkt mit eingebunden und die Methode `random` kann ohne weitere Vorarbeit aufgerufen werden.

Methode

`float random(int bereich)` oder  
`float random(float bereich):`

Berechnet bei Aufruf eine Zufallszahl aus dem Intervall `[0;bereich[`

# Zufallszahlen und Zufallsgenerator

Die Methode `random` berechnet bei Aufruf eine Zufallszahl aus dem Intervall `[0;bereich[` . Dabei ist `bereich` der Zahlenwert, der der Methode als Parameterwert übergeben wird.

Achtung: `random` liefert als Ausgabe einen Wert des Datentyps `float`!

Da wir bisher mit ganzen Zahlen (Positionen) arbeiten, müssen wir die berechnete Gleitkommazahl in eine Integerzahl umwandeln.

Dazu braucht man ein sogenanntes **type casting** (Datentypumwandlung).

Die Umwandlung vom Datentyp `float` in den Datentyp `int` ist ohne Probleme möglich.

Dazu wird der gewünschte Datentyp in runden Klammern vor den Aufruf der Methode geschrieben, die den nicht gewünschten (aber umwandelbaren) Datentyp liefert:

Mit dem Aufruf

```
(int) random(6)
```

wird eine ganze Zufallszahl zwischen 0 und 6 bestimmt. Dieser Aufruf darf natürlich nicht als Anweisung verwendet werden, da hier ein Wert zurückgegeben wird, der verarbeitet werden muss.

# Zufallszahlen und Zufallsgenerator

Mit dem Aufruf

```
(int) random(6)
```

wird eine ganze Zufallszahl zwischen 0 und 6 bestimmt. Dieser Aufruf darf natürlich nicht als Anweisung verwendet werden, da hier ein Wert zurückgegeben wird, der verarbeitet werden muss.

Zur Übung lassen wir uns einige Zufallszahlen ausgeben.

## **Arbeitsauftrag:**

Überlegen Sie sich ein Projekt Zufallszahlen, das außer dem Hauptprogramm die Klasse MATH enthält. Eine Methode der Klasse MATH soll bei Aufruf jeweils eine ganze Zufallszahl aus {3; 4; ... ; 10} zu erzeugen, die mit Hilfe der System.out.println-Methode ausgegeben wird.

Überlegen Sie sich, wie Sie im Hauptprogramm die jeweils vorherige Zufallszahl speichern und mit der aktuellen Zufallszahl vergleichen können. Wenn zwei gleiche Zufallszahlen hintereinander kommen, lassen Sie den Text „Dieselbe Zahl berechnet.“ ausgeben. Schaffen Sie es auch, einen passenden Text auszugeben, wenn eine Zahl z.B. viermal hintereinander berechnet wird?