

# Objektsammlungen II & Bibliotheksklassen I

## Lernziele

- Sie sammeln Erfahrung mit Objektsammlungen flexibler und fester Grösse.
- Sie setzen bei einfachen Problemstellungen, zu deren Lösung eine oder mehrere Objektsammlungen eingesetzt werden müssen, die jeweils passende Objektsammlung ein.
- Sie können mit (mehrdimensionalen) Arrays umgehen und setzen diese situationsgerecht ein.
- Sie können eine bisher unbekannte Klasse mithilfe der zugehörigen Klassendokumentation in Ihrem Programm gezielt einsetzen.
- Sie können eine Klasse von Grund auf so programmieren, dass Sie eine gegebene Spezifikation erfüllt.

## Aufgabe 1 (auf Papier!)

Der Aufruf `holeDatenreihe()` gibt eine Datenreihe als Text zurück. Der Text besteht aus durch Leerzeichen getrennte Ganzzahlen. Speichern Sie alle diese Datenreihen in einer geeigneten Objektsammlung und geben Sie anschliessend die Zahlen der Datenreihe so aus, dass pro Ausgabezeile genau eine Ganzzahl ausgegeben wird.

**Hinweis:** Die Methoden der Klasse `String` könnten hier hilfreich sein. Sie können eine Zahl rechtsbündig ausgeben mit `System.out.println(String.format("%11d", zahl));`

```
1 package ch.rashanras.helloworld.Prog1Uebung;
2
3 public class Array_Datenreihe {
4
5     1 usage
6     public Array_Datenreihe(){
7
8     }
9
10    public static void main (String[] args){
11        Array_Datenreihe datensammlung = new Array_Datenreihe();
12
13        String[] datenreiheOhneLeerschlag = datensammlung.holeDatenreihe().split( regex: " ");
14
15        for (String ganzeZahl : datenreiheOhneLeerschlag){
16            System.out.println(ganzeZahl);
17        }
18    }
19
20    1 usage
21    public String holeDatenreihe(){
22        String datenreihe = " 1 3 33 882 99 4 93 00 19 3 9 ";
23        return datenreihe;
24    }
25 }
```

## Aufgabe 2

Forken Sie für diese Aufgabe das Projekt [https://github.zhaw.ch/prog1-kurs/04\\_Praktikum-2\\_Messen](https://github.zhaw.ch/prog1-kurs/04_Praktikum-2_Messen)  
Nutzen Sie BlueJ um die eigene Projektkopie auf Ihren Computer zu holen und zu bearbeiten.

In dieser Aufgabe sollen Sie messen, wie lange es im Durchschnitt dauert, eine bestimmte Anzahl Zufallszahlen zu generieren und zu sortieren. Studieren Sie dazu zuerst die Schnittstelle der Klasse Messkonduktor. Die Interna der Klasse brauchen Sie nicht zu interessieren.

a) Die Methode `messungenDurchfuehren()` gibt einen Array zurück. Ist das notwendig und sinnvoll?

Es wird eine bestimmte Anzahl messungen durchgeführt. Deshalb ghat ein Array.

Erweitern Sie anschliessend die Klasse `MessApp` um folgende Funktionalität:

- b) Führen Sie 10 Messreihen à 20 Messungen durch. Die Laufzeiten sollen in einem zweidimensionalen Array gespeichert werden.  
Pro Messung sollen 400'000 Zufallszahlen generiert und sortiert werden.
- c) Berechnen Sie nun anhand der Messergebnisse folgende Mittelwerte und geben Sie diese aus:
  - i) Mittelwert pro Messreihe
  - ii) Mittelwert pro Messung (über alle 1. Messungen pro Messreihe, über alle 2. Messungen pro Messreihe etc.)
- d) Betrachten Sie die Messergebnisse und vergleichen Sie diese mit den Messungen von Kollegen. Haben Sie eine Erklärung für die unterschiedlichen Resultate?



## Aufgabe 3

Forken Sie für diese Aufgabe das Projekt [https://github.zhaw.ch/prog1-kurs/05\\_Praktikum-1\\_Pruefung](https://github.zhaw.ch/prog1-kurs/05_Praktikum-1_Pruefung)  
Nutzen Sie BlueJ um die eigene Projektkopie auf Ihren Computer zu holen und zu bearbeiten.

Die Klasse `Pruefungsergebnis` ist gegeben. Vervollständigen Sie die Klasse `Pruefungsverwaltung` an den bezeichneten Stellen so, dass bei Ausführung des folgenden Codestücks:

```
Pruefungsverwaltung verwaltung = new Pruefungsverwaltung();  
verwaltung.speichereErgebnis(new Pruefungsergebnis("Susi Muster", 5.3333));  
verwaltung.speichereErgebnis(new Pruefungsergebnis("Max Mueller", 3.74));  
verwaltung.speichereErgebnis(new Pruefungsergebnis("Heinz Moser", 4));  
verwaltung.druckeAntworttexte();
```

Folgende Ausgabe (Reihenfolge ist egal) erzeugt wird:

```
Herzliche Gratulation Susi Muster! Sie haben an der Prüfung die Note 5.5
erzielt und haben somit bestanden!
Max Mueller, Sie haben an der Prüfung die Note 3.5 erzielt und sind somit
durchgefallen!
Herzliche Gratulation Heinz Moser! Sie haben an der Prüfung die Note 4.0
erzielt und haben somit bestanden!
```

Sie müssen dabei folgende Überlegungen anstellen:

- Was für eine Datenstruktur verwenden Sie zum Speichern der Prüfungsergebnis-Objekte?

Es ist gute Praxis das Lösen einzelner Probleme in eigene Methoden auszulagern. Ganz getreu dem Grundsatz „teile und herrsche“. Dies hilft Ihnen dabei die Komplexität zu minimieren und macht den Code besser lesbar.

Zum Ausprobieren der Prüfungsverwaltung verwenden Sie die Klasse ZufaellicheNotengebung. Diese muss allerdings noch ergänzt werden, so dass sie folgende Rahmenbedingungen erfüllt:

- Die Methode fuehreAus der Klasse ZufaellicheNotengebung soll für Tests ein Set von Prüfungsergebnissen mit zufällig gewählten Noten erzeugen und anzeigen. Ergänzen Sie die Klasse um den hierfür noch zusätzlich notwendigen Code.
- Die Noten in der Ausgabe müssen auf 0.5 gerundet werden.
- Der Ausgabebetext muss sich auf das Prüfungsergebnis (bestanden, nicht bestanden) und den Kandidaten beziehen.

## Aufgabe 4

Folgende Methode der Klasse Pruefungsverwaltung ist gegeben:

```
private double rundeAufHalbeNote(double note) {
    return Math.round(note * 2) / 2.0;
}
```

Wieso ist es wichtig 2.0 zu schreiben und wieso würde 2 nicht das gewünschte Ergebnis liefern?

note ist zwar double wird aber mit zwei multipliziert und dann auf eine ganze Zahl gerundet. Es entsteht eine ganze Zahl von primitiven Typ long. Wenn man eine ganze Zahl durch eine ganze Zahl teilt, ergibt es einen wieder eine ganze Zahl, bzw dieses mal einen long wert. Wenn am aber 2.0 schriebs entsteht ein double, also mit kommazahlen. Somit können wir .5 Zahlen darstellen.