

Arrays: Übungen 1

Gegeben ist eine Klasse "**messreihe1**" mit einigen Methoden. Bei der Erzeugung einer Instanz des Typs Messreihe wird ein Array mit zufällig generierten "Messwerten" vom Typ `double` erzeugt. bearbeite die folgenden Aufgaben.¹⁾

- [BlueJ Projekt Arrays](#)

Aufgaben:

- **(1)** Erprobe die Methode `gibGewicht()`. Wie muss sie aufgerufen werden? Klappt der Aufruf immer? Was wird in der Abfrage in Zeile 31 geprüft? Was versucht man hier abzufangen? Erläutere diese Prüfabfrage im Detail.
- **(2)** Vervollständige die Methode `setzeAn5und9` zum Setzen eines neuen Wertes für die Elemente `gewicht[5]` und `gewicht[9]` diesem Array.
- **(3)** Schreibe eine Methode zum Setzen eines neuen Wertes für ein Element dieses Arrays mit wählbarem Index. Überprüfe dabei, ob der übergebene Index sich im erlaubten Bereich befindet und gib zurück, ob die Operation erfolgreich war.
- **(4)** Ermittle das Durchschnittsgewicht der gesamten Messreihe. Notiere zuerst deine Idee und setze sie dann in Quelltext um. Warum sollte dies eine eigenständige Funktion(Methode) werden?
- **(5)** Schreibe eine Methode, die je eine Integer Zahl als Start- (s) und Endindex (e) erhält und damit den Durchschnitt aller Werte mit Indizes (i) zwischen s und e ermittelt.

Lösungsvorschläge

Aufgabe 1: Das ist ein klassischer "Getter": `gibGewicht(index)` gibt das Gewicht des am Index `index` stehenden Arrayelements zurück. Die Überprüfungen sollen sicherstellen, dass man nicht den zulässige Wertebereich für den Index verlässt, als "Fehlermeldung" wird `-8.888` zurückgegeben.

Hier ein paar Vorschläge zur Lösung der Aufgaben. Die Methoden können meist noch verbessert werden, die Lösungen machen keine Eingabeüberprüfung.

```
/**
 * Setze ein Element des Arrays auf einen gegebenen Wert.
 * Gib zurück, ob die Operation erfolgreich war.
 *
 *
 * @param index Index des zu setzenden Elements
 * @param wert Wert, der an diesem Index gesetzt werden soll
 * @return boolean Erfolgreich oder nicht
 */
public boolean setzeAnIndex(int index, double wert) {

    if(index < 0 || index > anzahl) return false;

    gewicht[index] = wert;
    return true;
}
```

```
}  
/**  
 * Aufgabe 4  
 *  
 * @return Durchschnittsgewicht  
 */  
public double getAverage()  
{  
    double sum = 0;  
  
    for (int i=0; i<anzahl;i++) {  
        sum = sum + gewicht[i];  
    }  
    return sum/anzahl;  
}  
  
/**  
 * Aufgabe 5  
 *  
 * @param start Startindex (inklusive dieses Elements)  
 * @param end Endindex (inklusive dieses Elements)  
 * @return Durchschnittsgewicht  
 */  
public double getAverageFromTo(int start, int end)  
{  
    // Hier gibt es zahlreiche Verbesserungsmöglichkeiten  
    double sum = 0;  
  
    for (int i=start; i<=end;i++) {  
        sum = sum + gewicht[i];  
    }  
    return sum/(end-start+1);  
}
```

1)

Diese Übungen stehen unter einer CC-BY-SA Lizenz, sie wurden erstellt in enger Anlehnung an das Material der ZPG BW/Heußler

From:
<https://www.info-bw.de/> -

Permanent link:
<https://www.info-bw.de/faecher:informatik:oberstufe:java:algorithmen:arrays:uebungen1:start>

Last update: **10.10.2023 13:05**

