

ÜBUNGS-BLOCK 7

AUFGABEN



Aufgabe 1:

Gegeben ist folgender Code:

```
Auto[] array = new Auto[3];
// Alle Autos im Array tunen:
for (int i = 1; i <= 3; i++) {
    array[i].tunen();
}
```

Was hältst du davon? Fällt dir irgendetwas auf?

Aufgabe 2:

Gegeben ist der folgende Code:

```
1    int[] b = {5,10,20};
2    int a[] = new int[2]{5,10};
3    b = {10,15,20};
4    a = new int[3];
5    String d = new String[100];
6    double[][] e = new double[5][];
7    double[] f[] = {{10},{50, 20}};
```

Welche dieser Zeilen sind gültig, welche nicht?

Aufgabe 3:

In Video 39 haben wir die Summenfunktion durch Rekursion gelöst. Hier nochmal der Code:

```
static long berechneSumme(int zahl) {
    if(zahl == 1) {
        return 1;
    }
    return zahl + berechneSumme(zahl-1);
}
```

Zu diesem Zeitpunkt konnten wir diese Aufgabe nicht anders lösen. Später in Video-Block 7 haben wir allerdings die Schleifen kennengelernt. Ein alternativer Ansatz zur rekursiven Methode ist die sogenannte iterative Methode. Dabei ruft sich die Methode nicht selber auf - sie wird nur einmal aufgerufen und setzt wiederkehrende Logik in einer Schleife um.

Implementiere die Summenfunktion durch Iteration.

Aufgabe 4:

In einem der vorherigen Übungs-Blöcke gab es eine Aufgabe zur Quersummenberechnung. Die Funktionalität war zu diesem Zeitpunkt allerdings auf ausschließlich zweistellige Zahlen beschränkt.

Mit dem Wissen, dass du inzwischen hast, kannst du eine Quersummenberechnung für eine beliebig große (positive) Eingabe-Zahl vom Datentyp `int` implementieren.

Schreibe eine neue Methode `berechneQuersumme(int)` unter der Bedingung, dass – außer für die Aufsummierung der einzelnen Ziffern – **keinerlei weitere** mathematischen Operationen verwendet werden! Außer dem `“+“`-Operator ist also kein weiterer mathematischer Operator erlaubt, auch nicht `“-“` als Vorzeichen-Operator.

Tipps:

- Denke daran, dass der `“+“`-Operator neben der Addition noch eine weitere Bedeutung hat
- Siehe API-Doc Klassen `„String“` und `„Integer“` – vielleicht findest du etwas nützliches