

Präsenzübungen zur Vorlesung
Objektorientierte Programmierung: Wintersemester 2021/2022
Nr. 8

Aufgabe 8.1: Welcome to the Aperture Science computer-aided enrichment center

Betrachten Sie den folgenden Code zur Berechnung der kubischen Wurzel einer Zahl (die Formel könnte Déjà-vu auslösen):

```
1 public static double cubicRoot(double number, double delta) {
2     return cubicRootHelp(number, delta, number);
3 }
4
5 private static double cubicRootHelp(double number, double
    ↪ delta, double x_n) {
6     double x_nPlusOne = 1.0 / 3.0 * (2 * x_n + number /
    ↪ Math.pow(x_n, 2));
7     if(Math.abs(x_nPlusOne - x_n) < delta){
8         return x_nPlusOne;
9     }
10    return cubicRootHelp(number, delta, x_nPlusOne);
11 }
```

Schreiben Sie eine hinreichende Anzahl an sinnvollen JUnit-Tests für `cubicRoot`, in welchen Sie sowohl positive als auch negative Zahlen testen.

Aufgabe 8.2: It's over 9000!!!!

Legen Sie ein neues Projekt in IntelliJ an und fügen Sie die folgenden Klassen hinzu:

```
1 public class Sayan{
2     private int basePowerLevel
      ↪ = 10;
3     private int transformation
      ↪ = 0;
4     final String name;
5
6     public Sayan(String name){
7         this.name = name;
8     }
9 }

1 public class Namekian{
2     private int basePowerLevel
      ↪ = 8;
3     private int transformation
      ↪ = 0;
4     final String name;
5
6     public Namekian(String
      ↪ name){
7         this.name = name;
8     }
9 }

1 public class Human{
2     private int basePowerLevel = 5;
3     final String name;
4
5     public Human(String name){
6         this.name = name;
7     }
8 }
```

- Legen Sie ein Interface `Fighter` an. Dieses Interface soll über die Methoden `void train(int hours)` und `int getPowerLevel()` verfügen. Implementieren Sie außerdem die Default-Methode `default void gravityTrain(int hours, int gForce)`, in welcher `train` aufgerufen wird, wobei Stunden mit G-Force multipliziert werden sollen.
- Ändern Sie Ihre Klassen dahingehend ab, dass alle 3 das Interface `Fighter` implementieren.

Hinweis: IntelliJ sollte Ihnen mitteilen, dass ein Fehler vorliegt, da Ihre Klassen die benötigten Methoden nicht implementieren. Klicken Sie mit Ihrer Maus auf den Namen Ihrer Klasse im Quellcode und drücken Sie gleichzeitig auf `alt` und `Enter` auf Ihrer Tastatur. Wählen Sie im aufgehenden Menü die Option *Implement Methods*. IntelliJ erzeugt nun sogenannte Stubs für die fehlenden Methoden.

- Implementieren Sie nun die eben eingefügten Methoden.

Ein Mensch sollte vom Training einen Zuwachs von `hours` auf sein `basePowerLevel` erhalten, welcher von `getPowerLevel` zurückgegeben werden sollte.

Ein Namekianer hingegen erhält vom Training das doppelte an `hours` auf sein `basePowerLevel`. Das Powerlevel eines Namekianers berechnet sich aus `basePowerLevel * (transformation * 1.8 + 1)`.

Ein Sayajin erhält sogar das dreifache an `hours` auf sein `basePowerLevel` beim Trainieren. Das Powerlevel eines Sayajin berechnet sich durch `basePowerLevel * (transformation * 2 + 1)`.

- d) Fügen Sie nun der Klasse `Namekian` eine Methode `void namekianAbsorb(int i)` hinzu, die das `transformation` auf den Wert `i` setzt. Erstellen Sie außerdem die Methoden `void supaSayajin(int i)`, welche die `transformation` auf `i` setzt (1-4 ist erlaubt), und `void supaSayajinGoto()`, welche `transformation` auf 666 setzt, in der Klasse `Sayan`. Fügen Sie beiden Klassen eine Methode `void powerdown()` hinzu, welche `transformation` wieder auf 0 setzt.
- e) Testen Sie Ihre Methoden mit JUnit-Tests.
- f) Die Klasse `Sayan` soll nun zusätzlich das Interface `Comparable` implementieren. Implementieren Sie die fehlende Methode dahingehend, dass Sie mithilfe von `getPowerLevel()` nach dem Powerlevel sortieren.
- g) Legen Sie nun 4 verschiedene Instanzen von `Sayan` in einer Main-Methode an: *Son Goku*, *Vegeta*, *Nappa* und *Radditz*. Verändern Sie durch Training oder Transformation die Powerlevel der vier und erstellen Sie ein Array, in welches Sie die vier unsortiert ablegen. Iterieren Sie mit einer For-Schleife über das Array und geben Sie die Namen auf der Konsole aus. Verwenden Sie `Arrays.sort`, um das Array zu sortieren und geben Sie die Namen erneut auf der Konsole aus.