Programmiervorkurs

Tag 4

Michael Martel

Ablauf

09:30	Tag 3 Übungsbesprechung
10:00	Vorlesung
12:00	60 Minuten Mittagspause
13:00	Übungen im LI 136 und LI 137

Danach

Inhaltsübersicht Vorkurs

Tag 1	Zustände, Variablen, Datentypen, Konvertierungen, Arithmetik, Eclipse Live-Demo
Tag 2	Kommentare, Boolesche Ausdrücke, If-Abfragen, Switch-Case
Tag 3	Arrays, (Do-)While-Schleife, For-Schleifen, Weiterführung Debugging
Tag 4	**Methoden, Klassen, JavaDoc**

Prozesse digitalisieren

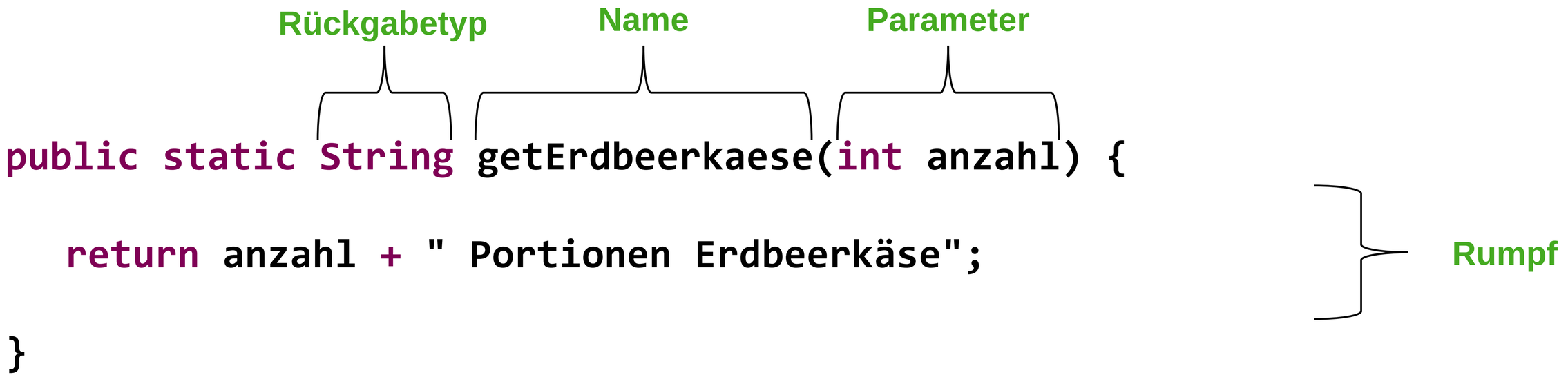
Methoden / Funktionen

Die Begriffe Methode und Funktion werden oft synonym verwendet

Ziel einer Methode / Funktion ist es eine bestimme zugeteilte Aufgabe zu erledigen

```java
public class Beispiel {

    public static void main(String[] args) {
        System.out.println("Hello World");
        double sqrt = Math.sqrt(2.0);
        doMath();
    }

    public static void doMath() {
        int a = 42;
        int b = a * 1337;
        System.out.println(b);
    }

}
```

Methoden / Funktionen



Eine Methode ohne Rückgabewert wird mit dem Schlüsselwort **void** gekennzeichnet!

Übung

Aufgabe:

Implementiere eine Hilfsfunktion um System.out.println(String) zu verkürzen auf println(String)

```
public static void println(String text) {
    System.out.println(text);
}
```

Übung

Aufgabe:

Addiere zwei (ganze | reelle) Zahlen

```
public static int addiere(int a, int b) {
    return a + b;
}

public static double addiere(double a, double b) {
    return a + b;
}
```

Übung

Aufgabe:

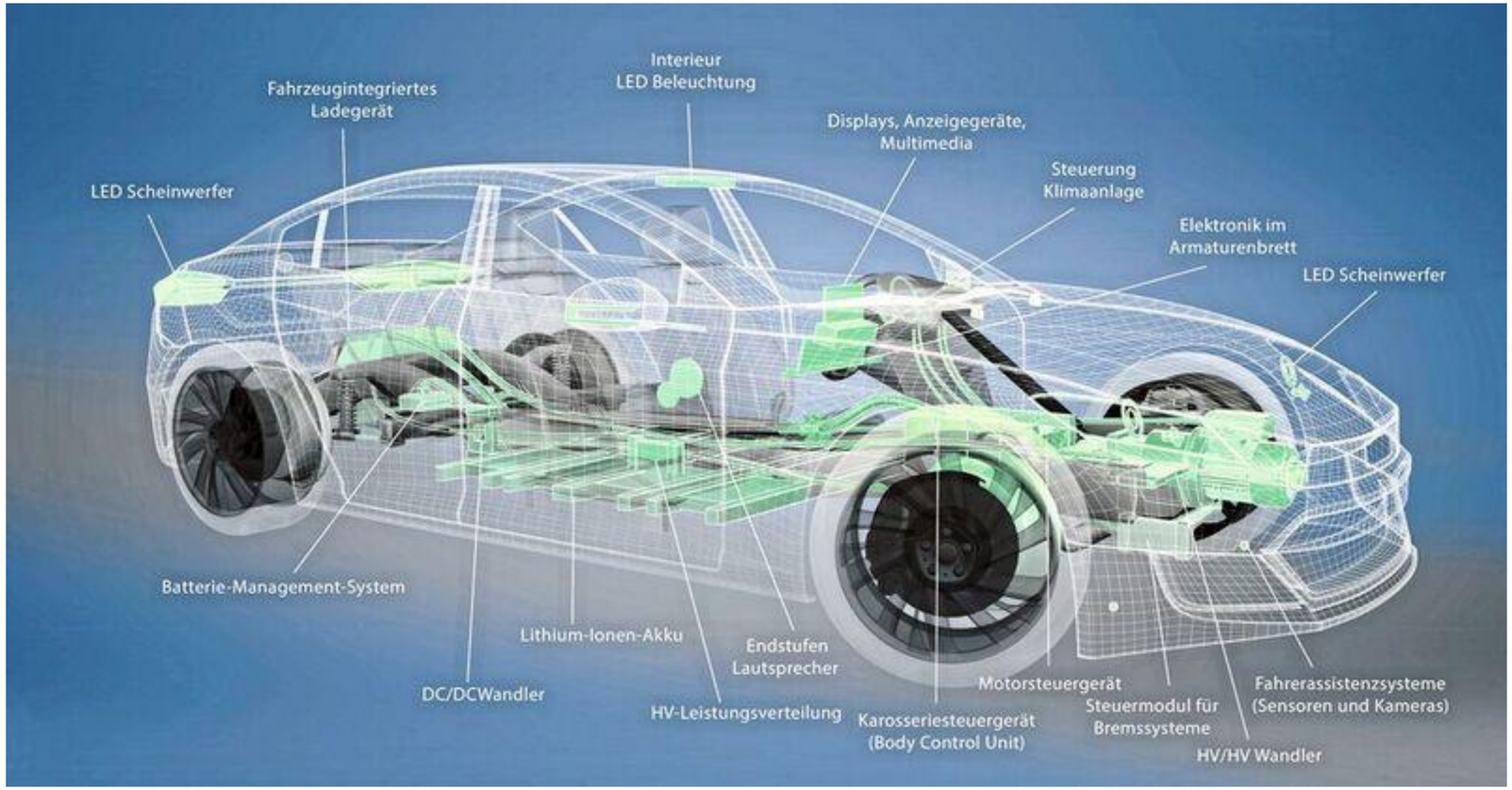
Betrag eines 2D-Vektors $|\vec{a}| = \sqrt{a_x^2 + a_y^2}$ **Math.sqrt()**

```
public static double vektorBetrag(double[ ] vec) {
    return Math.sqrt(vec[0] * vec[0] + vec[1] * vec[1]);
}
```

Methoden / Funktionen

Fragen?

Klassen


Interieur
LED Beleuchtung
Fahrzeugintegriertes
Ladegerät
Displays, Anzeigegeräte,
Multimedia
LED Scheinwerfer
Steuerung
Klimaanlage
Elektronik im
Armaturenbrett
LED Scheinwerfer
Batterie-Management-System
Lithium-Ionen-Akku
Endstufen
Lautsprecher
DC/DCWandler
HV-Leistungsverteilung
Karosseriesteuergerät
(Body Control Unit)
Motorsteuergerät
Steuermodul für
Bremssysteme
Fahrerassistenzsysteme
(Sensoren und Kameras)
HV/HV Wandler

Klassen

Kaffeefach

Wassertank

Heizelement

Aufschäumer

Klassen

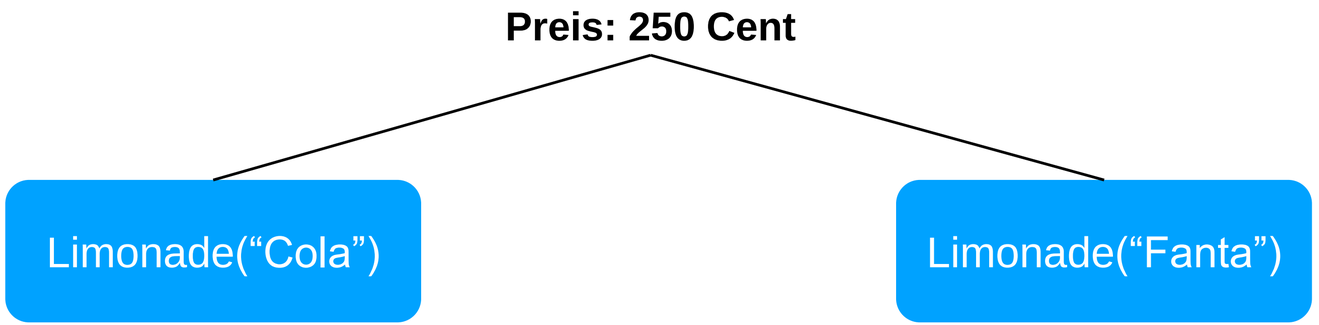
```
public class Kaffeemaschine {
    private Wassertank wassertank;
    private Heizelement heizelement;

    public Kaffeemaschine(Wassertank wassertank, ...) {
        this.wassertank = wassertank;
        // ...
    }

    public Kaffee kocheKaffee(double menge) {
        // ...
        double wasserMenge = wassertank.getWasser(menge);
        // ...
        return kaffee;
    }
}
```

Klassenvariablen

```
public class Kaffeemaschine {
    private Wassertank wassertank;
    private Heizelement heizelement;

    public Kaffeemaschine(Wassertank wassertank, ...) {
        this.wassertank = wassertank;
        // ...
    }

    public Kaffee kocheKaffee(double menge) {
        // ...
        double wasserMenge = wassertank.getWasser(menge);
        // ...
        return kaffee;
    }
}
```

Statisch und Instanz

```
public class Limonade {
    public static int preis = 250;
    public String name;

    public Limonade(String name) {
        this.name = name;
    }
}
```



Statisch und Instanz

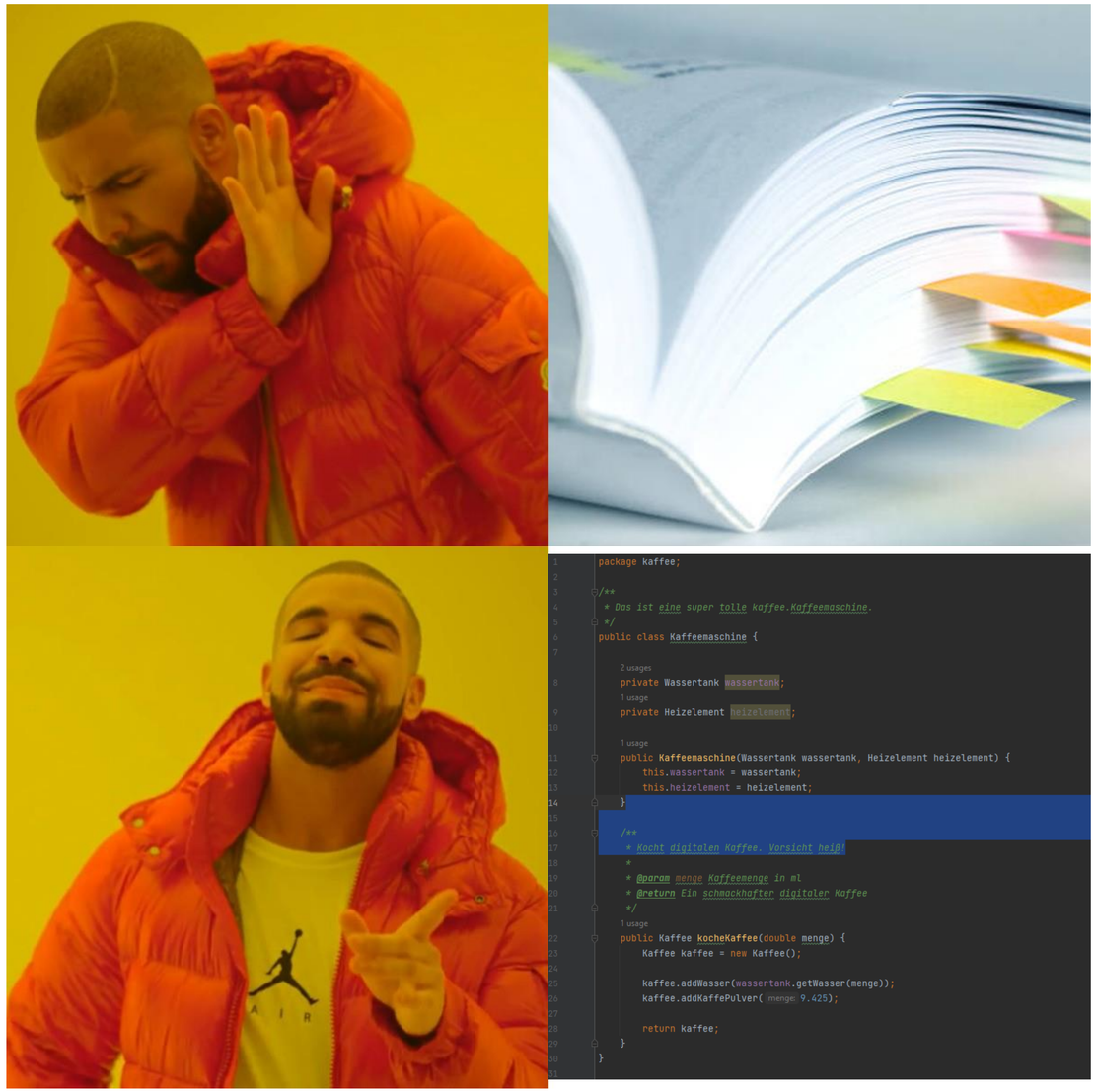
```
public static void main(String[] args) {
    Limonade cola = new Limonade("Cola");
    Limonade fanta = new Limonade("Fanta");

    System.out.println(cola.name); // Cola
    System.out.println(fanta.name); // Fanta

    System.out.println(Limonade.preis); // 250
}
```

Beispiel Applikation

Klassen und Klassenvariablen

Fragen?

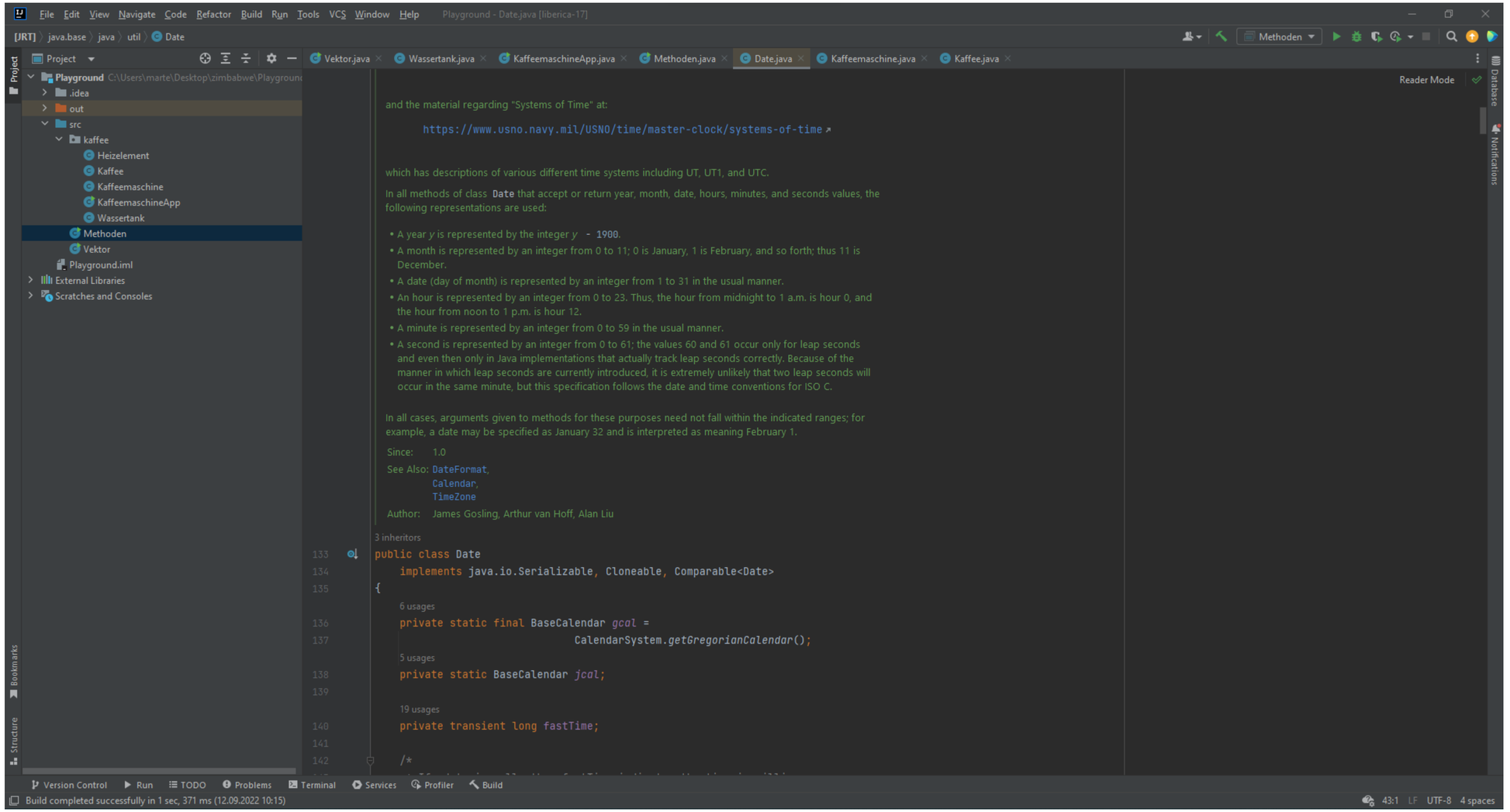
JavaDoc


package kaffee;
/**
* Das ist eine super tolle Kaffee.Kaffeemaschine.
*/
public class Kaffeemaschine {
2 usages
private Wassertank wassertank;
1 usage
private Heizelement heizelement;
1 usage
public Kaffeemaschine(Wassertank wassertank, Heizelement heizelement) {
this.wassertank = wassertank;
this.heizelement = heizelement;
}
/**
* Kocht digitalen Kaffee. Vorsicht heiß!
*
* @param menge Kaffeemenge in ml
* @return Ein schmackhafter digitaler Kaffee
*/
1 usage
public Kaffee kocheKaffee(double menge) {
Kaffee kaffee = new Kaffee();
kaffee.addWasser(wassertank.getWasser(menge));
kaffee.addKaffePulver(menge: 9.425);
return kaffee;
}
}

JavaDoc

```java
/**
 * Das ist eine Kaffeemaschine.
 */
public class Kaffeemaschine {

    /**
     * Kocht digitalen Kaffee. Vorsicht heiß!
     *
     * @param menge Kaffeemenge in ml
     * @return Ein schmackhafter digitaler Kaffee
     */
    public Kaffee kocheKaffee(double menge) {
        // ...
        return kaffee;
    }
}
```

JavaDoc reale Beispiele



Ende! Fragen?