

JAVA Programmierung

ECLIPSE & JAVA 8



Themenübersicht

01

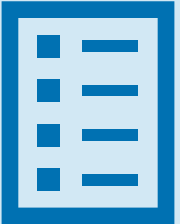
Klassen

1. Generelles
2. Beispiele

02

Aufgabe

1. Aufgabe
2. Lösung





01 Klassen

1. *Generelles*
2. *Beispiele*



1.1

Klassen

Generelles



1.1 Klassen

• Generelles

Java bietet die Möglichkeit der Datenkapselung innerhalb von Klassen. Fast alles was an Quellcode in Java erstellt wird, befindet sich innerhalb solcher Klassen. Der Aufbau folgt immer dem selben Schema.

```
//Modifizier  Schlüsselwort => class      Name  
public      class      FirstClass {  
    // Anweisungsblock innerhalb der geschweiften Klammer  
}
```

Modifizierer, Schlüsselwort „class“, Name der Klasse => dieser beginnt immer mit einem Großbuchstaben und muss im Package einmalig (unique) sein.

Klassen sind ein Grundpfeiler der OOP. Nur über die Klassen lassen sich Objekte erstellen. Zum Erstellen der Objekte wird der Konstruktor verwendet, den wir im nächsten Kapitel kennenlernen werden.

1.1 Klassen

• Generelles

Innerhalb des Anweisungsblock, welcher wie bei Methoden durch geschweifte Klammern gekennzeichnet ist, werden Attribute (Variablen) und Methoden aufgeführt. Attribute und Methoden werden auch als „Member“ einer Klasse bezeichnet.

```
4 public class ErsteKlasse {  
5     //Eigenschaften und Attribute  
6     private int x;  
7  
8     //Methoden, wie Konstruktor, Getter, Setter oder sonstige  
9     public void methode1(){  
10         //Anweisung, was die Methode machen soll  
11     }  
12 }
```

1.1 Klassen

• Generelles

Klassen-spezifische Attribute stellen Eigenschaften dieser Klasse dar.

Methoden steuern das Verhalten von Klassen und Objekten.

Auch der Konstruktor ist eine Methode innerhalb einer Klasse.

Methoden welche nur die Eigenschaften manipulieren, heißen Getter- und Setter-Methoden.

```
private String attribut1;

// getter => gibt ein Attribute zurück
public String getAttribut1() {
    return attribut1;
}
// setter => manipuliert ein Attribute oder instanziiert es
public void setAttribut1(String attribut1) {
    this.attribut1 = attribut1;
}

public void methode() {
    // do something
}
```

1.2 Klassen

Beispiele



1.2 Klassen

- Beispiel

Class

```
1  
2  
3  
4 public class ErsteKlasse {  
5  
6  
7 }  
8
```

Member

```
1  
2  
3  
4 public class ErsteKlasse {  
5     //Eigenschaften und Attribute  
6     private int x;  
7  
8 }
```

1.2 Klassen

- Beispiel

Getter & Setter

```
public class ErsteKlasse {  
    //Eigenschaften und Attribute  
    private int x;  
  
    // getter => gibt den Wert des Member x zurück  
    public int getX() {  
        return x;  
    }  
    // setter => setzt den Wert des Member x  
    public void setX(int x) {  
        this.x = x;  
    }  
}
```

Methoden

```
public class ErsteKlasse {  
    //Eigenschaften und Attribute  
    private int x;  
  
    // getter => gibt den Wert des Member x zurück  
    public int getX() {  
        return x;  
    }  
    // setter => setzt den Wert des Member x  
    public void setX(int x) {  
        this.x = x;  
    }  
    //Methoden, wie Konstruktor, Getter, Setter oder sonstige  
    public void methode1() {  
        //Anweisung, was die Methode machen soll  
    }  
}
```



02 Aufgaben

1. *Aufgabe*
2. *Lösung*



2.1 Aufgaben

Aufgabe



2.1 Aufgabe

Erstellen Sie eine eigene Klasse mit:

- **1 Attribut**
- **2 Methoden => eine ohne
eine mit Return-Wert**
- **Getter- und Setter-Methode
für das Attribut**

2.2 Aufgaben

Lösung



2.2 Lösung

Erstellen Sie eine eigene Klasse mit:

- 1 Attribut
- 2 Methoden => eine ohne
eine mit Return-Wert
- Getter- und Setter-Methode
für das Attribut

```
-  
4 public class ErsteKlasse {  
5  
6     private String name;  
7  
8     public String getX() {  
9         return name;  
0     }  
1  
2     public void setX(String name) {  
3         this.name = name;  
4     }  
5  
6     public void print(){  
7         System.out.println(name);  
8     }  
9  
0     public String getMyName(){  
1         return "Florian";  
2     }  
3 }
```



VIELEN DANK!

