

Informatik Q1 Abels

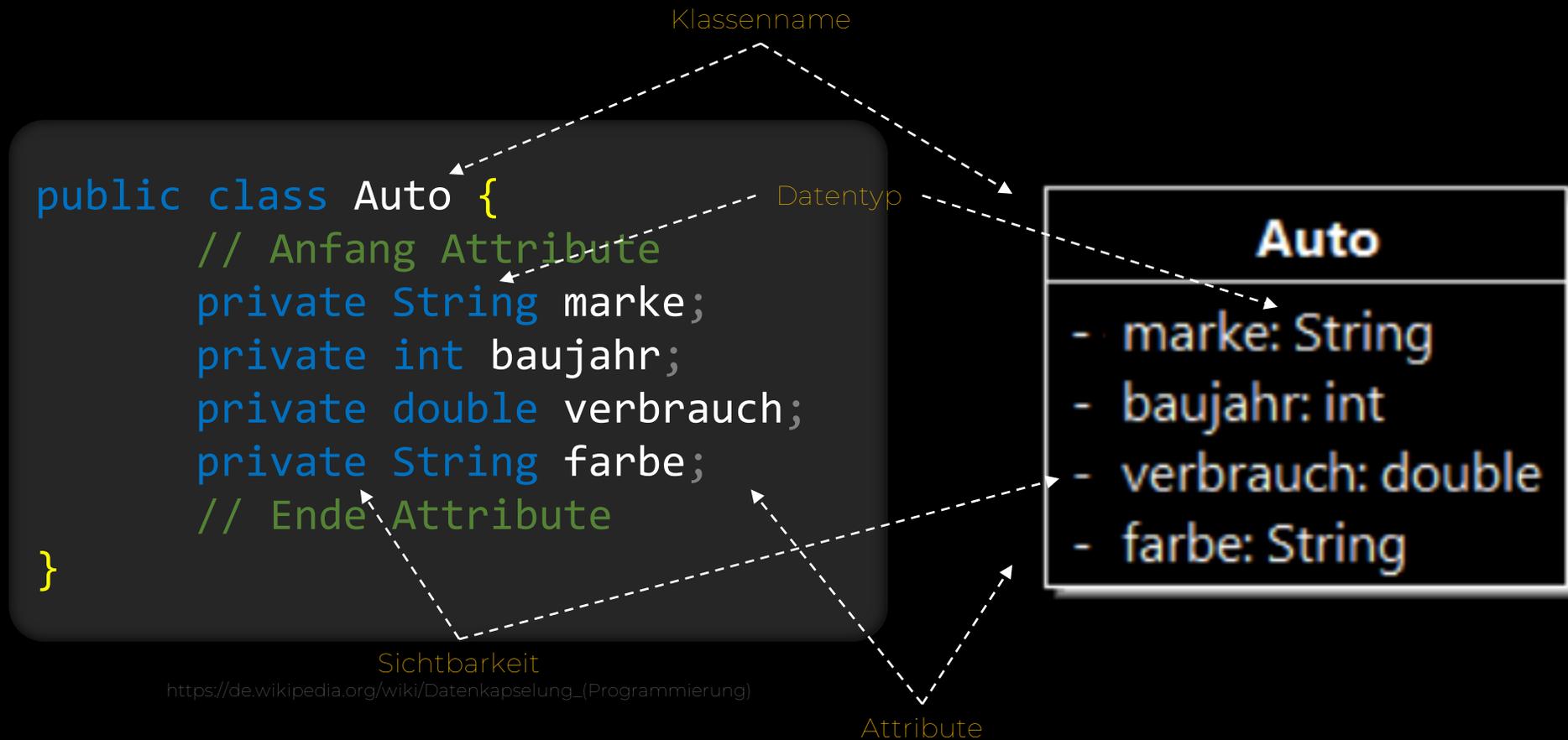


Klasse vs Objekt

Klasse vs Objekt

Klassen sind Vorlagen, aus denen Instanzen genannte Objekte zur Laufzeit erzeugt werden.

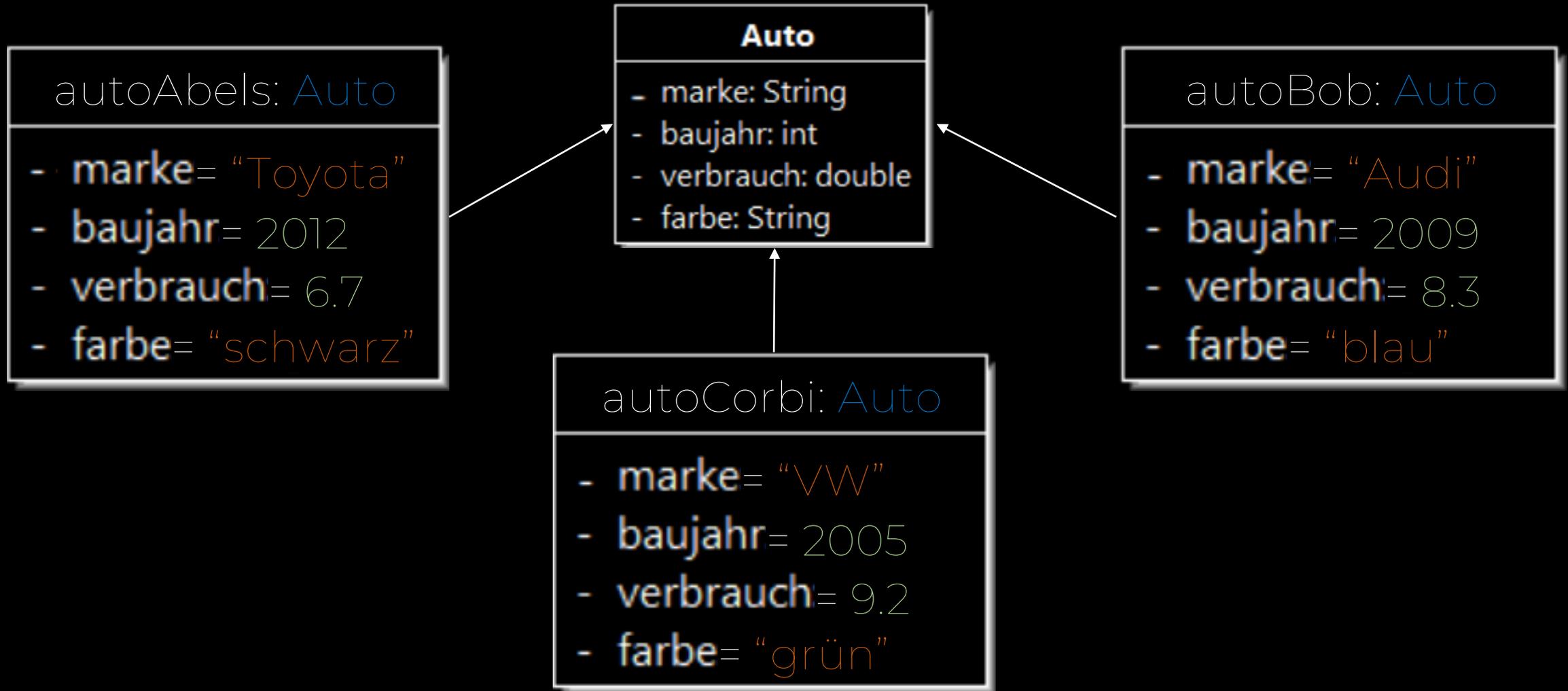
Klasse vs Objekt



Quellcode

UML

Klasse vs Objekt



Konstruktor

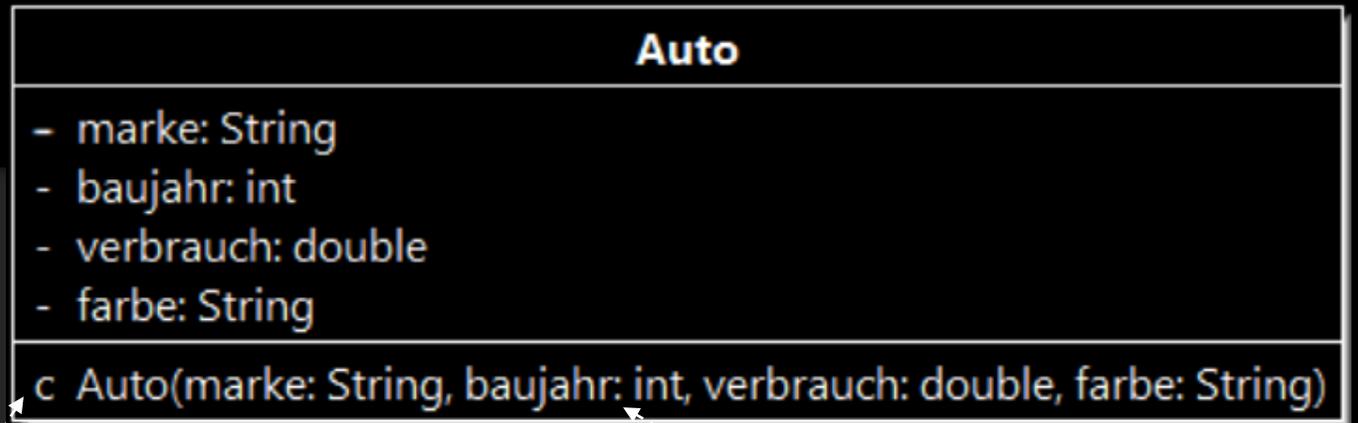
```
public class Auto {  
    // Anfang Attribute  
    private String marke;  
    private int baujahr;  
    private double verbrauch;  
    private String farbe;  
    // Ende Attribute
```

```
    public Auto(String marke, int baujahr, double verbrauch, String farbe) {  
        this.marke = marke;  
        this.baujahr = baujahr;  
        this.verbrauch = verbrauch;  
        this.farbe = farbe;  
    }  
}
```

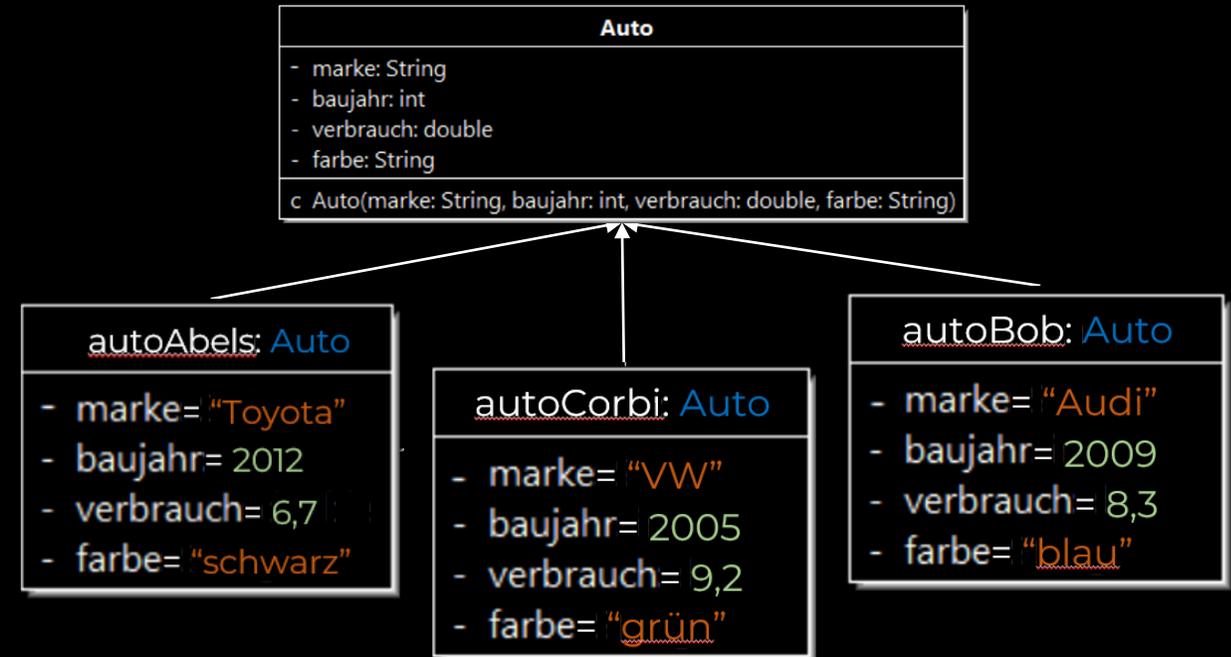
Bezug auf das
aktuelle Objekt

Constructor

Parameter zur Erzeugung
(Instanziierung) eines neuen Objekts



Konstruktor



```
public class Autohaus {
    public static void main(String[] args) {
        Auto autoAbels = new Auto("Toyota", 2012, 6.7, "schwarz");
        Auto autoBob = new Auto("Audi", 2009, 8.3, "blau");
        Auto autoCorbi = new Auto("VW", 2005, 9.2, "grün");
    }
}
```

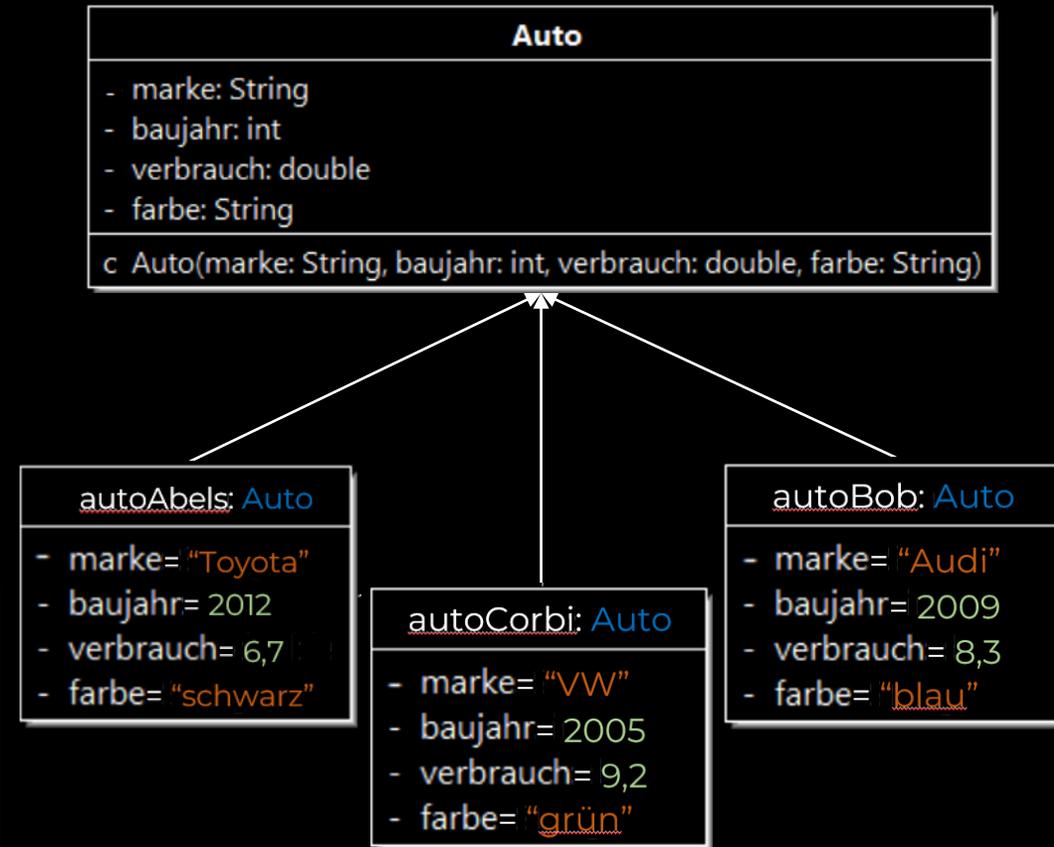
Auto nun als
Datentyp

Aufruf des Constructors mit den
gegebenen Parametern

Konstruktor

```
public class Auto {  
    // Anfang Attribute  
    private String marke;  
    private int baujahr;  
    private double verbrauch;  
    private String farbe;  
    // Ende Attribute  
  
    public Auto(String marke, int baujahr, double verbrauch, String farbe) {  
        this.marke = marke;  
        this.baujahr = baujahr;  
        this.verbrauch = verbrauch;  
        this.farbe = farbe;  
    }  
}
```

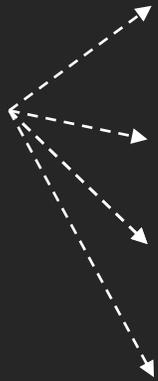
```
public class Autohaus {  
    public void main(String[] args) {  
        Auto autoAbels = new Auto("Toyota", 2012, 6.7, "schwarz");  
        Auto autoBob = new Auto("Audi", 2009, 8.3, "blau");  
        Auto autoCorbi = new Auto("VW", 2005, 9.2, "grün");  
    }  
}
```



get – Methode

```
public class Auto {  
    // Anfang Attribute  
    private String marke;  
    private int baujahr;  
    private double verbrauch;  
    private String farbe;  
    // Ende Attribute  
  
    public Auto(String marke, int baujahr, double verbrauch, String farbe) {  
        this.marke = marke;  
        this.baujahr = baujahr;  
        this.verbrauch = verbrauch;  
        this.farbe = farbe;  
    }  
  
    // Anfang Methoden  
    public String getMarke() {  
        return this.marke;  
    }  
  
    public int getBaujahr() {  
        return this.baujahr;  
    }  
  
    public double getVerbrauch() {  
        return this.verbrauch;  
    }  
  
    public String getFarbe() {  
        return this.farbe;  
    }  
    // Ende Methoden  
}
```

get-Methode



Methoden



Auto	
- marke: String	
- baujahr: int	
- verbrauch: double	
- farbe: String	
c Auto(marke: String, baujahr: int, verbrauch: double, farbe: String)	
+ getMarke(): String	← Rückgabewert
+ getBaujahr(): int	
+ getVerbrauch(): double	
+ getFarbe(): String	

Rückgabewert



get – Methode

```
public class Auto {
    // Anfang Attribute
    private String marke;
    private int baujahr;
    private double verbrauch;
    private String farbe;
    // Ende Attribute

    public Auto(String marke, int baujahr, double verbrauch, String farbe) {
        this.marke = marke;
        this.baujahr = baujahr;
        this.verbrauch = verbrauch;
        this.farbe = farbe;
    }

    // Anfang Methoden
    public String getMarke() {
        return this.marke;
    }

    public int getBaujahr() {
        return this.baujahr;
    }

    public double getVerbrauch() {
        return this.verbrauch;
    }

    public String getFarbe() {
        return this.farbe;
    }
    // Ende Methoden
}
```

```
public class Autohaus {
    public void main(String[] args) {
        Auto autoAbels = new Auto("Toyota", 2012, 6.7, "schwarz");
        Auto autoBob = new Auto("Audi", 2009, 8.3, "blau");
        Auto autoCorbi = new Auto("VW", 2005, 9.2, "grün");

        System.out.println(autoAbels.getMarke());
    }
}
```

set – Methode

```
public class Auto {
    // Anfang Attribute
    private String marke;
    private int baujahr;
    private double verbrauch;
    private String farbe;
    // Ende Attribute

    public Auto(String marke, int baujahr, double verbrauch, String farbe) {
        this.marke = marke;
        this.baujahr = baujahr;
        this.verbrauch = verbrauch;
        this.farbe = farbe;
    }

    // Anfang Methoden
    public String getMarke() {
        return this.marke;
    }

    public int getBaujahr() {
        return this.baujahr;
    }

    public double getVerbrauch() {
        return this.verbrauch;
    }

    public String getFarbe() {
        return this.farbe;
    }
}
```

Auto
- marke: String - baujahr: int - verbrauch: double - farbe: String
c Auto(marke: String, baujahr: int, verbrauch: double, farbe: String) + getMarke(): String + getBaujahr(): int + getVerbrauch(): double + getFarbe(): String + setMarke(marke: String) + setBaujahr(baujahr: int) + setVerbrauch(verbrauch: double) + setFarbe(farbe: String)

set-Methode

```
public void setMarke(String marke) {
    this.marke = marke;
}

public void setBaujahr(int baujahr) {
    this.baujahr = baujahr;
}

public void setVerbrauch(double verbrauch) {
    this.verbrauch = verbrauch;
}

public void setFarbe(String farbe) {
    this.farbe = farbe;
}

// Ende Methoden
}
```

set – Methode

```
public class Auto {
    // Anfang Attribute
    private String marke;
    private int baujahr;
    private double verbrauch;
    private String farbe;
    // Ende Attribute

    public Auto(String marke, int baujahr, double verbrauch, String farbe) {
        this.marke = marke;
        this.baujahr = baujahr;
        this.verbrauch = verbrauch;
        this.farbe = farbe;
    }

    // Anfang Methoden
    public String getMarke() {
        return this.marke;
    }

    public int getBaujahr() {
        return this.baujahr;
    }

    public double getVerbrauch() {
        return this.verbrauch;
    }

    public String getFarbe() {
        return this.farbe;
    }
}
```

```
public void setMarke(String marke) {
    this.marke = marke;
}

public void setBaujahr(int baujahr) {
    this.baujahr = baujahr;
}

public void setVerbrauch(double verbrauch) {
    this.verbrauch = verbrauch;
}

public void setFarbe(String farbe) {
    this.farbe = farbe;
}
// Ende Methoden
}
```

```
public class Autohaus {
    public void main(String[] args) {
        Auto autoAbels = new Auto("Toyota", 2012, 6.7, "schwarz");
        Auto autoBob = new Auto("Audi", 2009, 8.3, "blau");
        Auto autoCorbi = new Auto("VW", 2005, 9.2, "grün");

        System.out.println(autoAbels.getMarke()); ←--- Toyota
        autoAbels.setMarke("VW");
        System.out.println(autoAbels.getMarke()); ←--- VW
    }
}
```



Übung 1

Gegeben sei eine Klasse namens **Bus**. Die Klasse Bus hat Attribute für den Namen des Fahrers und die Zahl der Gäste. Für beide Attribute besitzt sie get-/set-Methoden und einen entsprechenden Konstruktor. Außerdem hat sie zwei Methoden für das Zu- bzw. Aussteigen von **n** Personen.

- a) Entwirf ein UML-Diagramm für die Klasse **Bus**.
- b) Erweitere das UML-Diagramm um zwei Objekte **bus1** und **bus2**.
- c) Implementiere die Klasse **Bus** in Java.
- d) Implementiere die Klasse **Verkehr**, in der die beiden Busse **bus1** und **bus2** mit erfundenen Daten instanziiert werden.
- e) Gib den Fahrer des Busses mit den meisten Gästen in der Console aus.
- f) Befördere 5 Gäste vom Bus mit mehr Gästen zum Bus mit weniger Gästen.



Tagebucheintrag



Klasse



Objekt

Attribut



Konstruktor

get- / set-Methode



Sichtbarkeit / Geheimnisprinzip / Datenkapselung

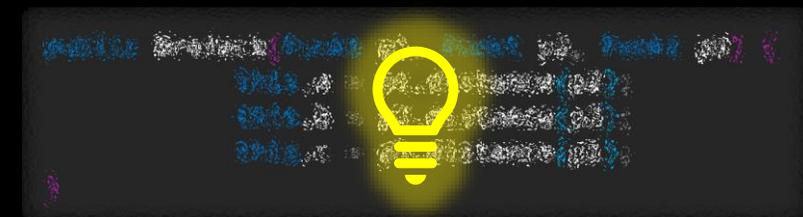


Übung 2

In einem neuen Ordner **Geometrie** wollen wir geometrische Objekte betrachten.

- Erstelle eine Klasse **Dreieck** nach folgendem UML.
- Implementiere die Funktion **public double area()**.
- Erstelle eine Klasse **Main**, in der das Objekt **d1** vom Typ **Dreieck** mit willkürlichen Zahlen instanziiert und dessen Flächeninhalt berechnet sowie ausgegeben wird.
- Erstelle eine Klasse **Punkt** mit den Parametern **double x** und **double y** inklusive Gettern, Settern und einem Konstruktor.
- Erweitere die Klasse **Punkt** um die Funktion **public double distance(Punkt p)**.
- Erstelle in der **Main**-Klasse ein weiteres **Dreieck**-Objekt **d2** durch Angabe dreier **Punkt**-Objekte und gib dessen Flächeninhalt aus. Was musst du dafür noch in der **Dreieck**-Klasse ergänzen?

Dreieck
- a: double - b: double - c: double
c Dreieck(a: double, b: double, c: double) + getA(): double + setA(aNew: double) + getB(): double + setB(bNew: double) + getC(): double + setC(cNew: double) + area(): double





Wochenübung

In einem neuen Ordner **Geometrie** wollen wir verschiedene geometrische Objekte betrachten.

- Erstelle jeweils eine Klasse für die Objekte **Dreieck**, **Zylinder** und **Kreis**.
- Erweitere die Klassen mit den Attributen, die diese Objekte eindeutig bestimmen.
- Erweitere die Klassen mit den jeweiligen get-/set-Methoden und Konstruktoren.
- Erweitere die Klassen mit Methoden zu Berechnung des Flächeninhalts bzw. Volumens.
- Erweitere die Klassen mit weiteren Konstruktoren und Methoden.

