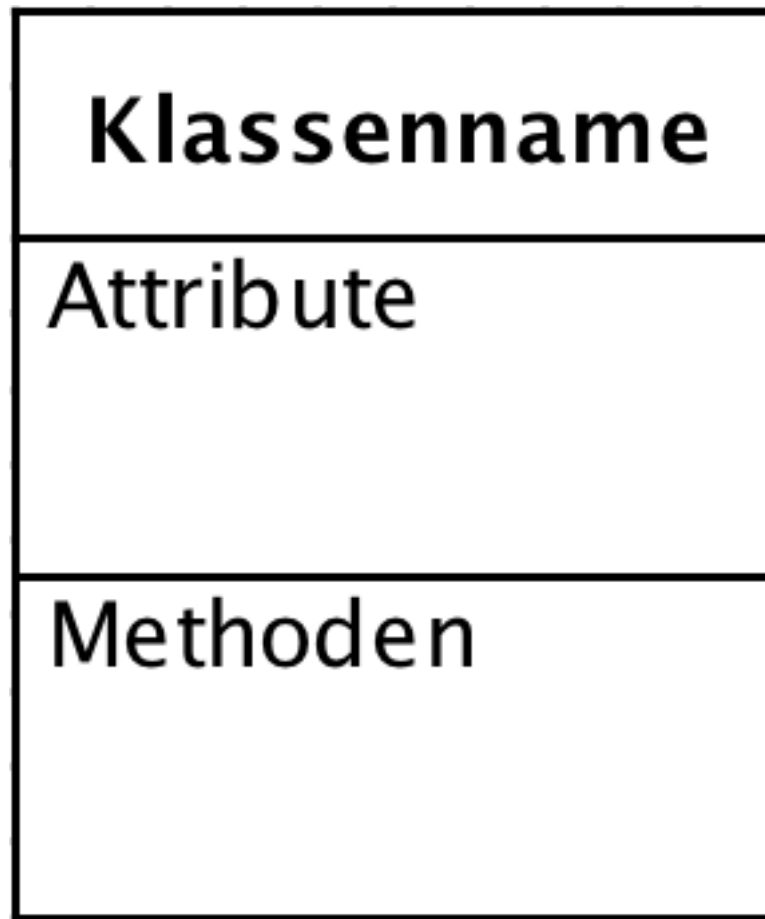


Klassendiagramme in UML

Grundsätzliches Schema

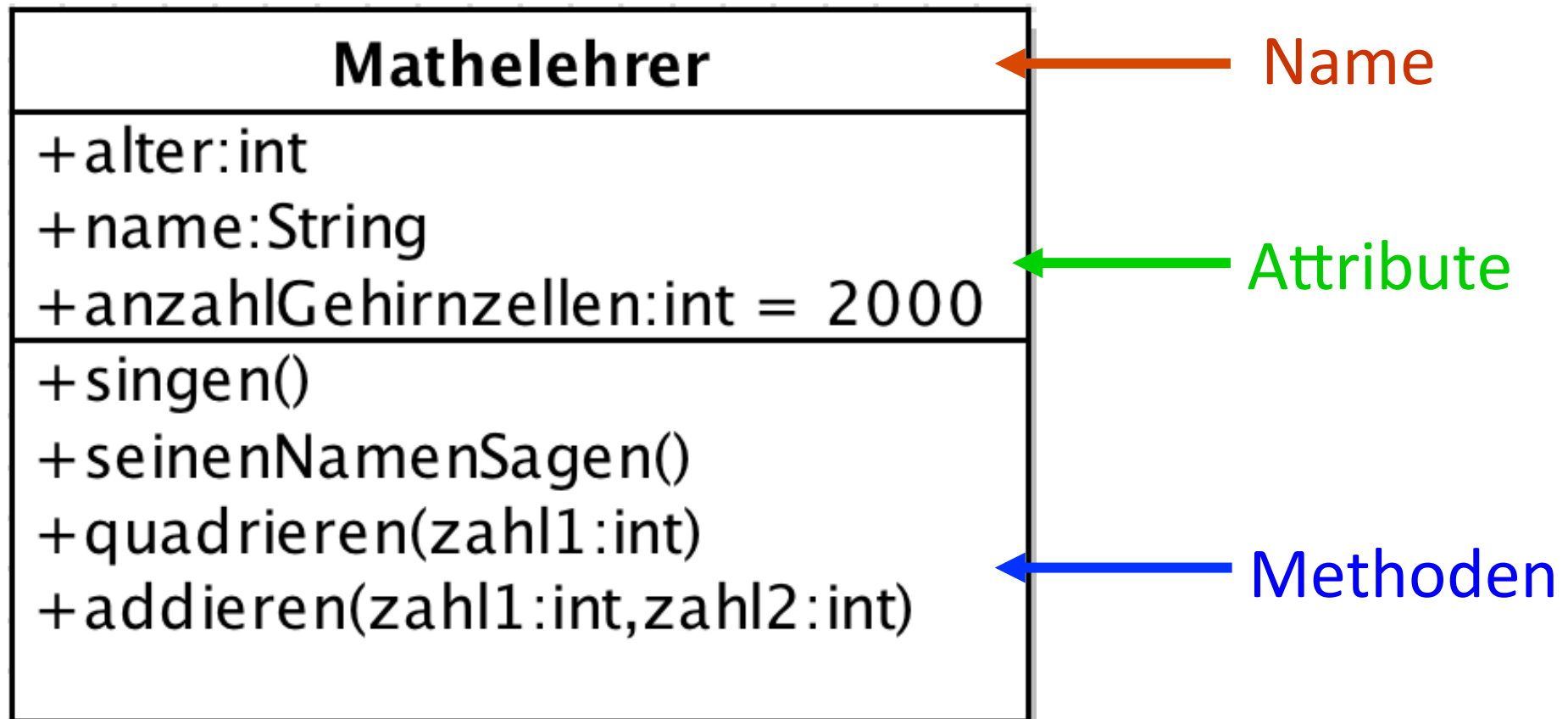


← Name

← Liste der Attribute
(Attribute = Eigenschaften)

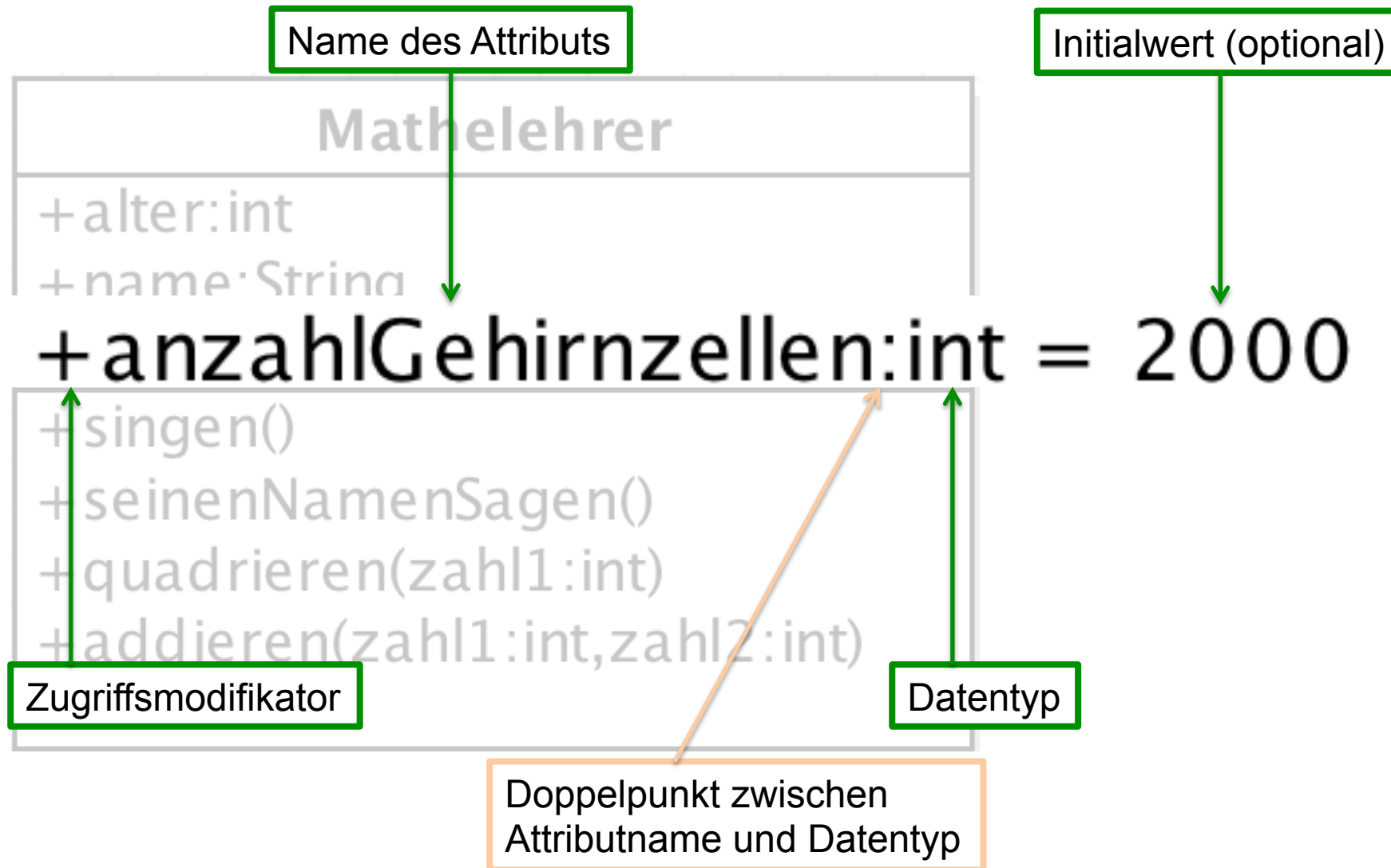
← Methodenliste
(Methoden = Was "kann" ein Objekt)

Grundsätzliches Schema



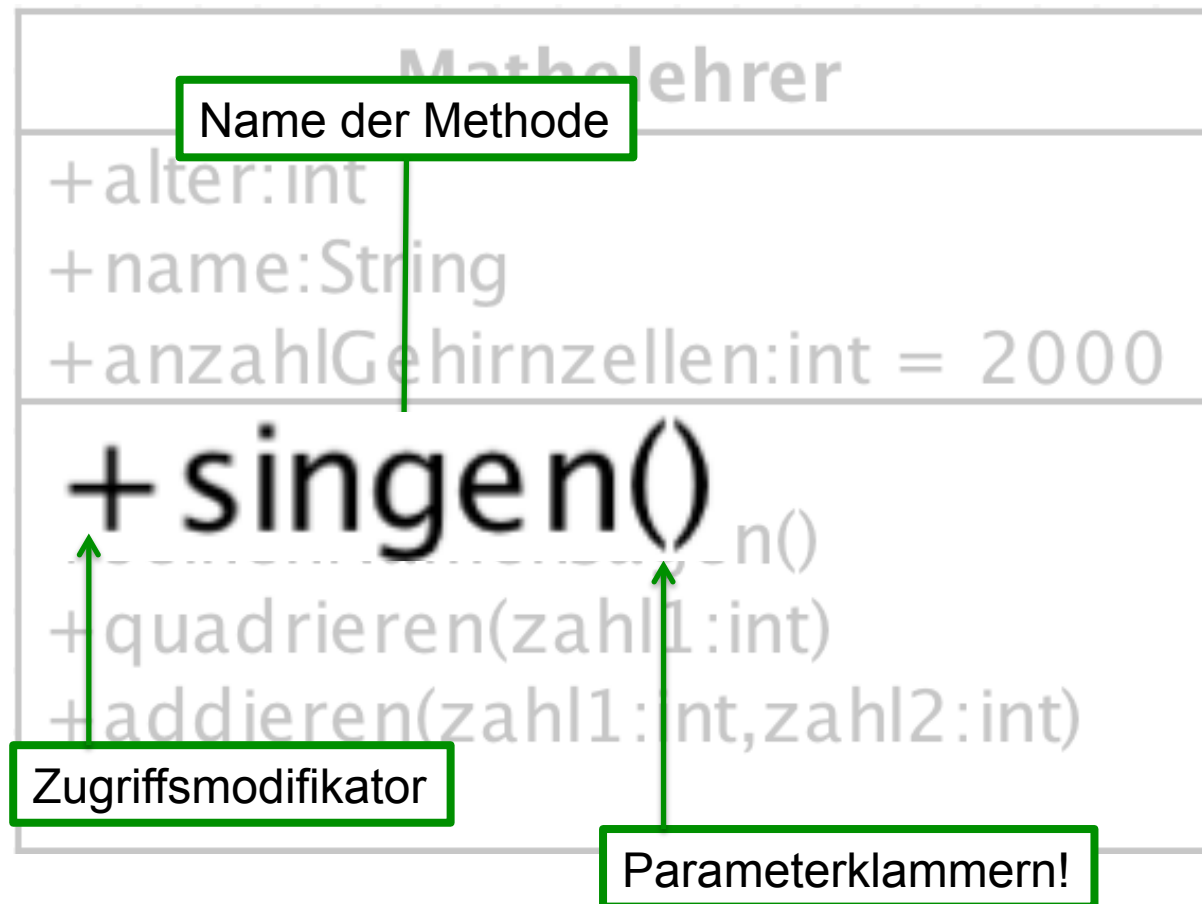
Achtung: Hier fehlen der Einfachheit halber die Rückgabetypen der Methoden!

Attribute



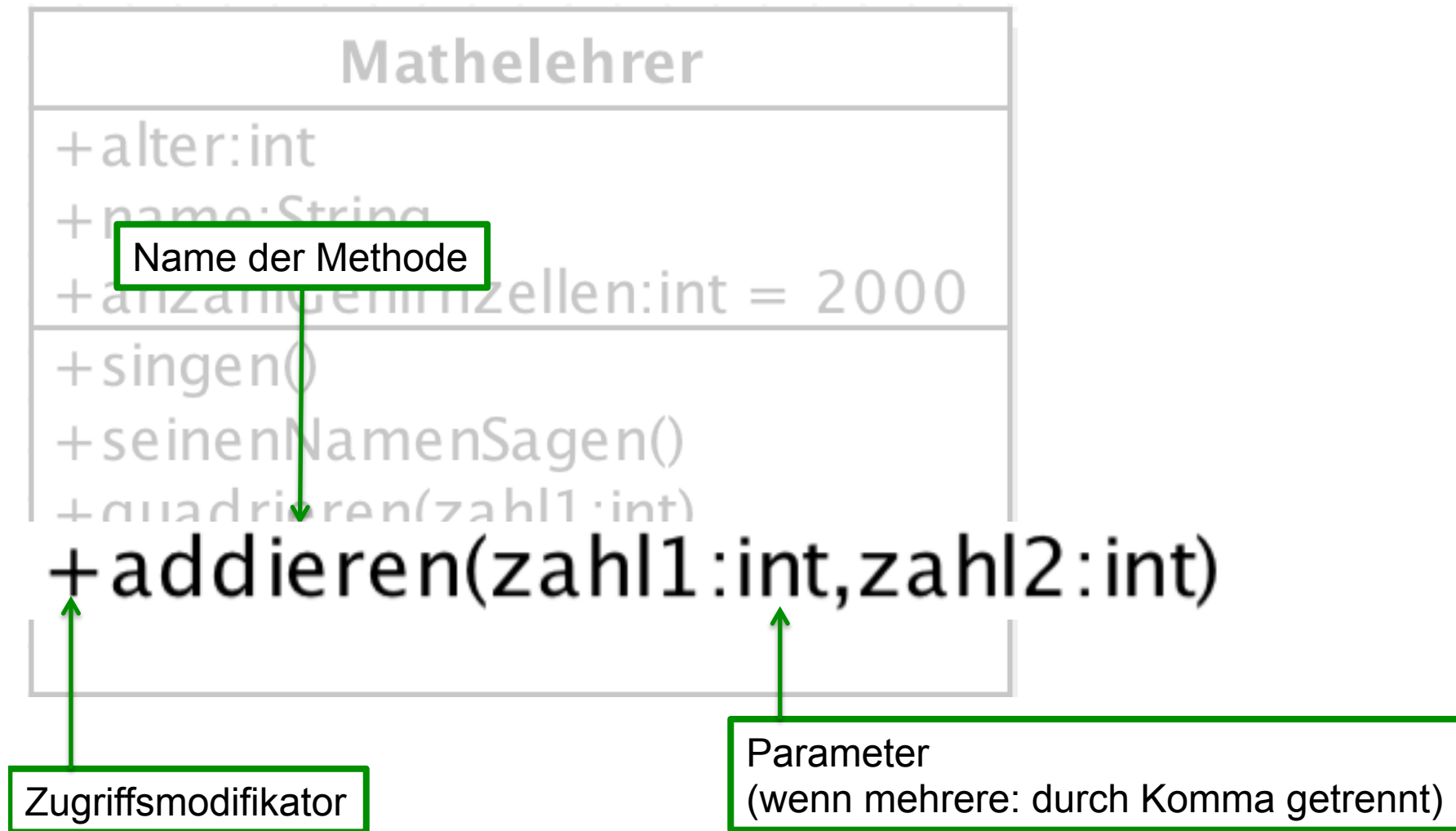
Achtung: Hier fehlen der Einfachheit halber die Rückgabetypen der Methoden!

Methoden



Achtung: Hier fehlen der Einfachheit halber die Rückgabetypen der Methoden!

Methoden mit Parameter

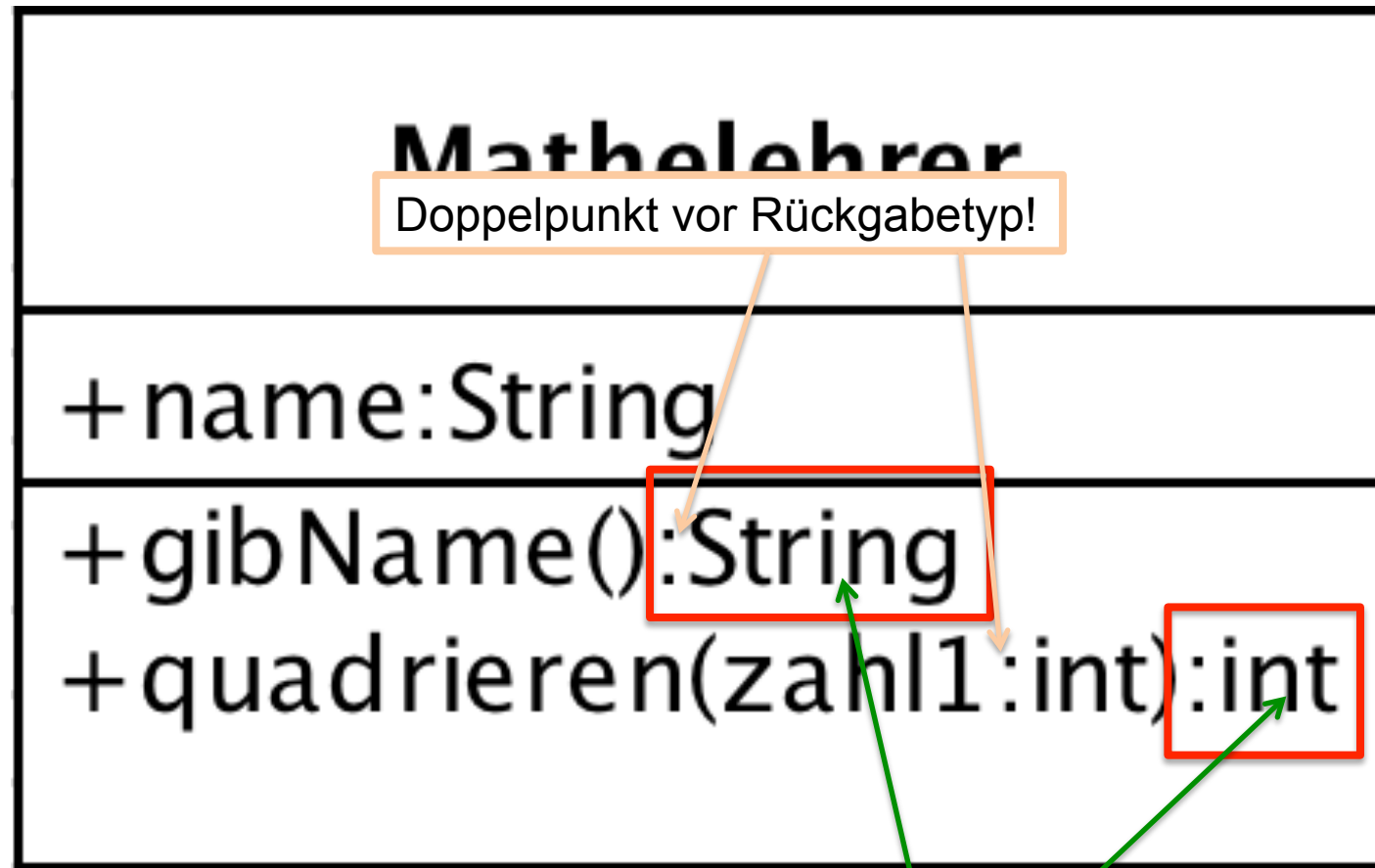


Achtung: Hier fehlen der Einfachheit halber die Rückgabetypen der Methoden!

Methoden mit Rückgabotyp

Mathelehrer	
+ name:String	
+ gibName():String	
+ quadrieren(zahl1:int):int	

Methoden mit Rückgabotyp



Welchen Datentyp gibt die Methode zurück?
(mit **return**)

Übung 1

Vogel
+ name:String + anzahlFedern:int + farbe:String
+ gibName():String + singen() + fliegen() + picken()

Wie viele Methoden hat diese Klasse?

Wie viele Attribute hat diese Klasse?

Achtung: Hier fehlen der Einfachheit halber die Rückgabetypen der Methoden!

Übung 2: Smartphone

Erstellen Sie ein Klassendiagramm für eine Klasse "Smartphone":

Attribut:

preis (Datentyp: float)

Methode:

klingeln (Parameterklammern nicht vergessen 😊)

Übung 2: Smartphone

Erstellen Sie ein Klassendiagramm für eine Klasse "Smartphone":

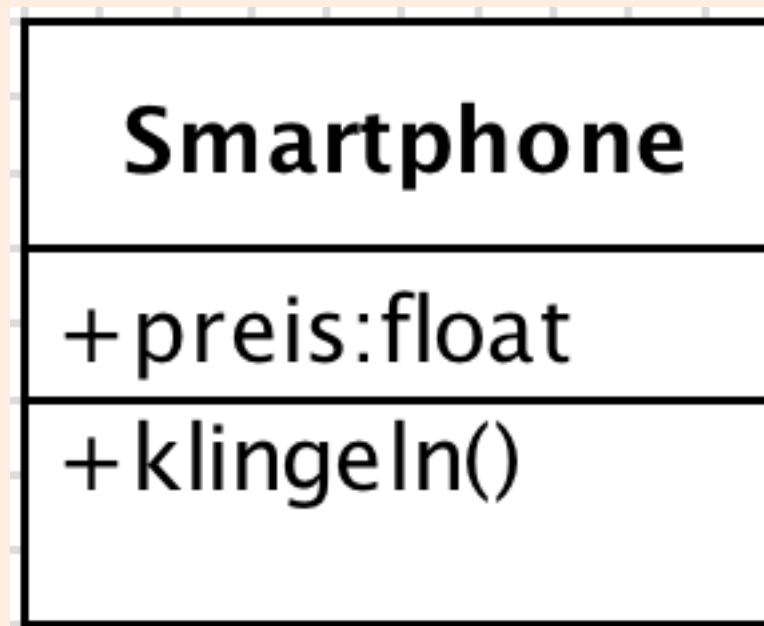
Attribut:

preis (Datentyp: float)

Methode:

klingeln (Parameterklammern nicht vergessen 😊)

Lösung:



Übung 3a: Uhr

Erstellen Sie ein Klassendiagramm für eine Klasse "Uhr":

Attribute:

armbandlaenge (Datentyp: int)

farbe (Datentyp: String)

Methoden:

ticken

aufziehen

weckerStellen

Übung 3a: Uhr

Erstellen Sie ein Klassendiagramm für eine Klasse "Uhr":

Attribute:

armbandlaenge (Datentyp: int)

farbe (Datentyp: String)

Methoden:

ticken

aufziehen

weckerStellen

Lösung:



Übung 3b: Uhr

Erstellen Sie ein Klassendiagramm für eine Klasse "Uhr":

Attribute:

armbandlaenge (Datentyp: int)

farbe (Datentyp: String)

Methoden:

ticken

aufziehen

weckerStellen

Wenn Sie eine Uhr erzeugen, ist sie automatisch rot und das Armband hat die Länge 10.

Die Attribute bekommen also einen Initialwert.

Übung 3b: Uhr

Erstellen Sie ein Klassendiagramm für eine Klasse "Uhr":

Attribute:

armbandlaenge (Datentyp: int)

farbe (Datentyp: String)

Methoden:

ticken

aufziehen

weckerStellen

Wenn Sie eine Uhr erzeugen, ist sie automatisch rot und das Armband hat die Länge 10.

Die Attribute bekommen also einen Initialwert.

Lösung:

Uhr
+armbandlaenge:int = 10 +farbe:String = "rot"
+ticken() +aufziehen() +weckerStellen()

Übung 4: Schuh

Erstellen Sie ein Klassendiagramm für eine Klasse "Schuh":

Attribute:

farbe (Datentyp: String, Initialwert: "weiß")

groesse (Datentyp: int)

bezeichnung (Datentyp: String)

Methoden:

anziehen

putzen

Übung 4: Schuh

Erstellen Sie ein Klassendiagramm für eine Klasse "Schuh":

Attribute:

farbe (Datentyp: String, Initialwert: "weiß")

groesse (Datentyp: int)

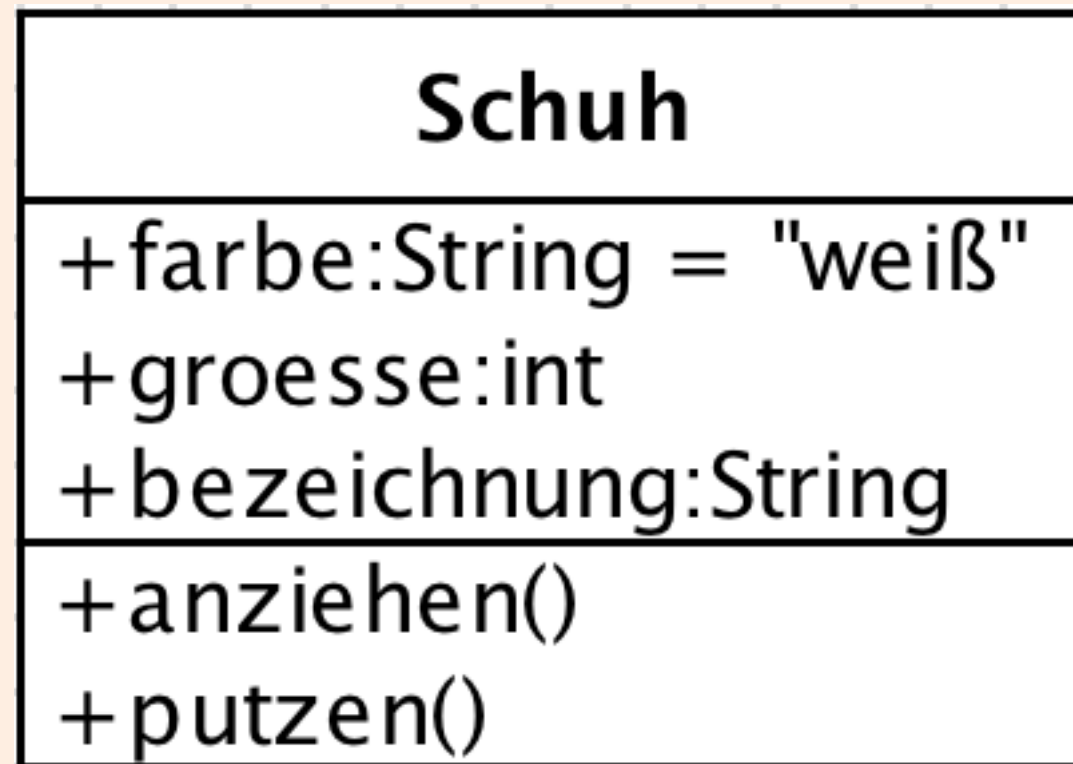
bezeichnung (Datentyp: String)

Methoden:

anziehen

putzen

Lösung:



Parameter

Die Methode „bekommt“ einen Wert, mit dem sie arbeiten kann.

`noteEintragen()` welche??

`noteEintragen(1.75)`

UML-Diagramm:

noteEintragen(**note:float**)

Parameter

Parameter

Die Methode „bekommt“ einen Wert, mit dem sie arbeiten kann.

lottozahlenEintragen() welche??

```
lottozahlenEintragen(1,2,7,13,19,33)
```

UML-Diagramm:

```
lottozahlenEintragen(zahl1:int, zahl2:int, zahl3:int, zahl4:int, zahl5:int, zahl6:int)
```

oder

lottozahlenEintragen(z1:int, z2:int, z3:int, z4:int, z5:int, z6:int)

Übung 5: Mathelehrer

Erstellen Sie ein Klassendiagramm für eine Klasse "Mathelehrer":

Attribute:

alter (Datentyp: int, Initialwert: 30)

Methoden:

veraendereAlter (bekommt Parameter namens *neuesAlter*, Datentyp int)

addiere (bekommt zwei Parameter namens *zahl1*, *zahl2*, beides int)

Übung 5: Mathelehrer

Erstellen Sie ein Klassendiagramm für eine Klasse "Mathelehrer":

Attribute:

alter (Datentyp: int, Initialwert: 30)

Methoden:

veraendereAlter (bekommt Parameter namens *neuesAlter*, Datentyp int)

addiere (bekommt zwei Parameter namens *zahl1*, *zahl2*, beides int)

Lösung:

Mathelehrer
+alter:int = 30
+veraendereAlter(neuesAlter:int) +addiere(zahl1:int, zahl2:int)

Übung 6: Uhr mit Parameter

Erstellen Sie ein Klassendiagramm für eine Klasse "Uhr":

Attribute:

stunde (Datentyp: int, Initialwert: 0)

minute (Datentyp: int, Initialwert: 0)

Methoden:

aufziehen(bekommt Parameter namens *anzahlUmdrehungen*, Datentyp int)

uhrStellen (bekommt zwei Parameter namens *std*, *min*, beides int)

Übung 6: Uhr mit Parameter

Erstellen Sie ein Klassendiagramm für eine Klasse "Uhr":

Attribute:

stunde (Datentyp: int, Initialwert: 0)

minute (Datentyp: int, Initialwert: 0)

Methoden:

aufziehen(bekommt Parameter namens *anzahlUmdrehungen*, Datentyp int)

uhrStellen (bekommt zwei Parameter namens *std*, *min*, beides int)

Lösung:

Uhr
+ stunde:int = 0 + minute:int = 0
+ aufziehen(anzahlUmdrehungen:int) + uhrStellen(std:int, min:int)

Zusammenfassung:

Darstellung:

Attribute

attributname:datentyp

alter:int

name:String

Methoden

methodenname(parameter):rückgabetyt

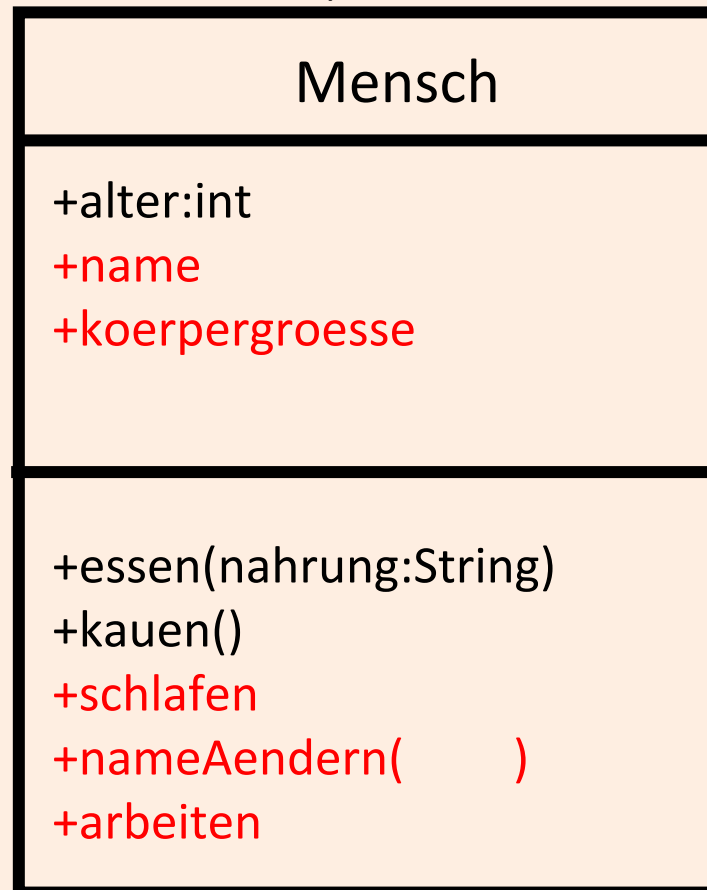
singen()

nameAendern (neuerName:String)

Achtung: Der Einfachheit halber lassen wir den Rückgabetyt erst mal weg.

UML-Klassendiagramm: Attribute, Operationen

Ergänzen Sie das Klassendiagramm in den rot markierten Zeilen.
(Die Zeilen in schwarzer Schrift sind korrekt!)



Achtung: Hier fehlen der Einfachheit halber die Rückgabetypen der Methoden!

