

2.3 Objektorientierte Programmierung mit Java

1.1

Hasen als Objekte



Benötigte IDEs:

Greenfoot, BlueJ

Verfasser:

Niko Diamadis

Erstellungs-/ Änderungsdatum

26. Juni 2020

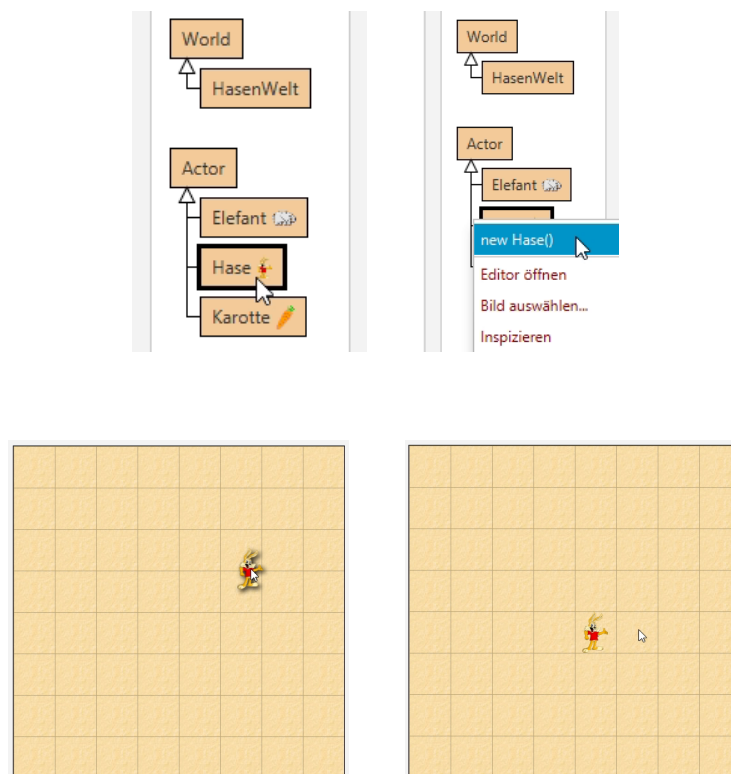
Inhaltsverzeichnis

1	Objekte in Aktion	1
1.1	Objekte und Klassen	1
1.2	Methoden	1
1.3	Parameter & Datentypen	2
1.4	Rückgabewerte	3
1.5	Attribute	4
1.6	Quelltext	5
1.7	Vererbung	5
1.8	Klassendiagramm für Hase	6
2	Fachkonzept - Objekt	7
3	Fachkonzept - Klasse	8
4	Übungen	9
4.1	Geburtstagsrechner	9
4.2	Zusammenfassung der Begriffe	10
4.3	Klassen- und Objektdiagramme	13
4.4	Kreuzworträtsel	14

1 Objekte in Aktion

1.1 Objekte und Klassen

Die gegebenen Anweisungen sollten selbsterklärend sein, hier nur noch eine passende Bildfolge dazu.



1.2 Methoden

Zunächst gehe ich auf die genannten Methoden eines Objektes der Klasse **Hase** ein.

➤ **linksDrehen**

- der Hase dreht sich um 90° um die eigene Achse

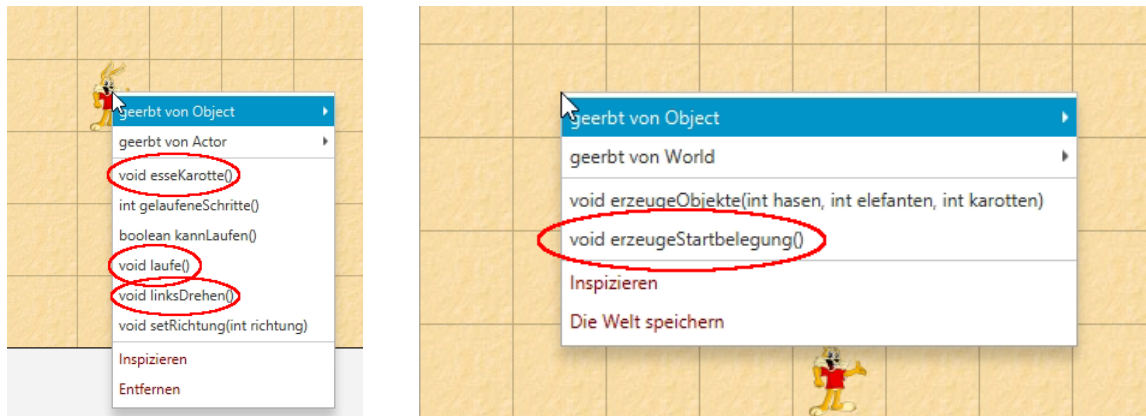
➤ **laufe**

- der Hase geht ein Feld weiter in die Richtung, in welche er gerade schaut

➤ **esseKarotte**

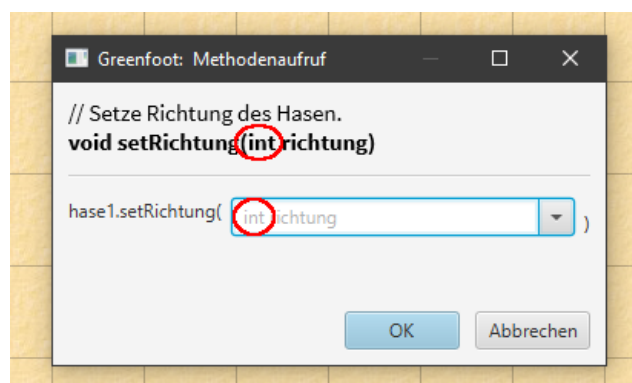
- der Hase frisst eine Karotte, wenn beide Objekte auf einem Feld stehen

Es folgen Bilder zur Erklärung, wie die Methoden von **Hase** und **Hasenwelt** aufgerufen werden.



1.3 Parameter & Datentypen

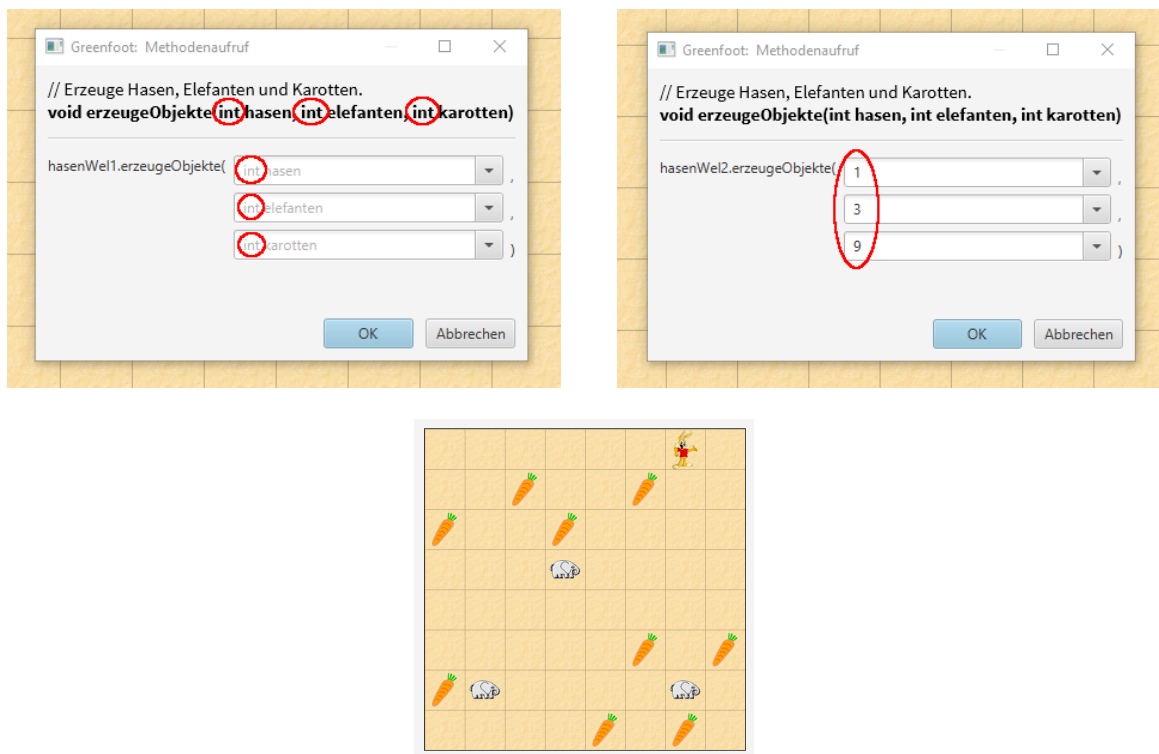
Nach Aufruf der Methode `setRichtung` am **Hasen**-Objekt öffnet sich ein Eingabefenster, um die gerade erlernten Parameter, welche die zu ausführende Methode, falls Parameter vorhanden, benötigt, zu übergeben.



Wie im Bild zu sehen findet man im Eingabefenster den Datentypen des Eingabewertes, der verlangt wird, in diesem Falle vom Typen `int`, also **Integer**¹. Zu bedenken ist jedoch, dass natürlich nicht alle Eingaben sinnvoll sind und verarbeitet werden können.

Selbiges gilt für die Methode `erzeugeObjekte`, nur dass dort drei Eingabewerte vom Typen `int` gefordert werden. Bei dieser Methode z.B. wird mithilfe des Eingabewertes die Anzahl der Objekte, die durch die Methode erstellt werden, festgelegt.

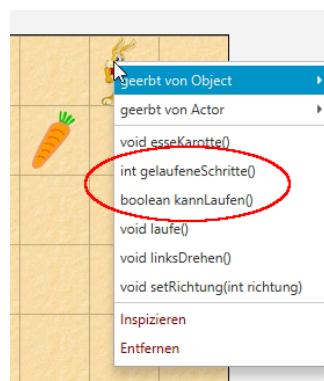
¹ sprich ganzzahlige Werte von -2^{32} ($-2.147.483.648$) bis $2^{32} - 1$ ($2.147.483.647$) inklusive der 0



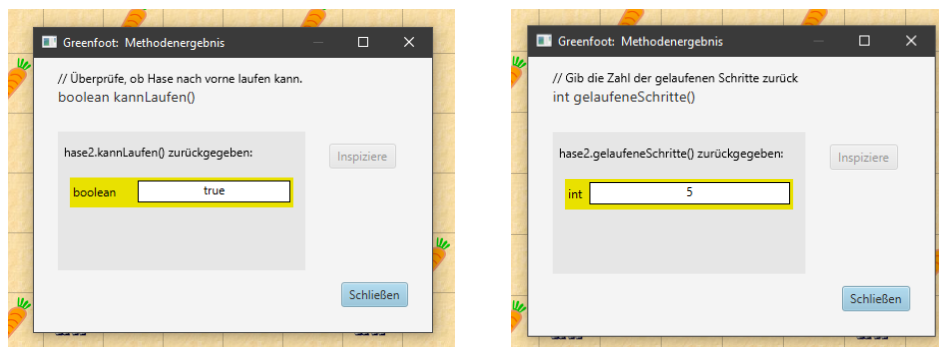
1.4 Rückgabewerte

Nachdem nun auch das Prinzip eines Rückgabewertes einer Methode erklärt worden ist, soll man diese an einem **Hasen**-Objekt erkunden.

Uns stehen bei diesem Beispiel nur zwei Methoden mit Rückgabewert zur Verfügung (`int gelaufeneSchritte` und `boolean kannLaufen`; vor allen anderen steht nämlich `void`, bei diesen wird also kein Wert zurückgegeben).



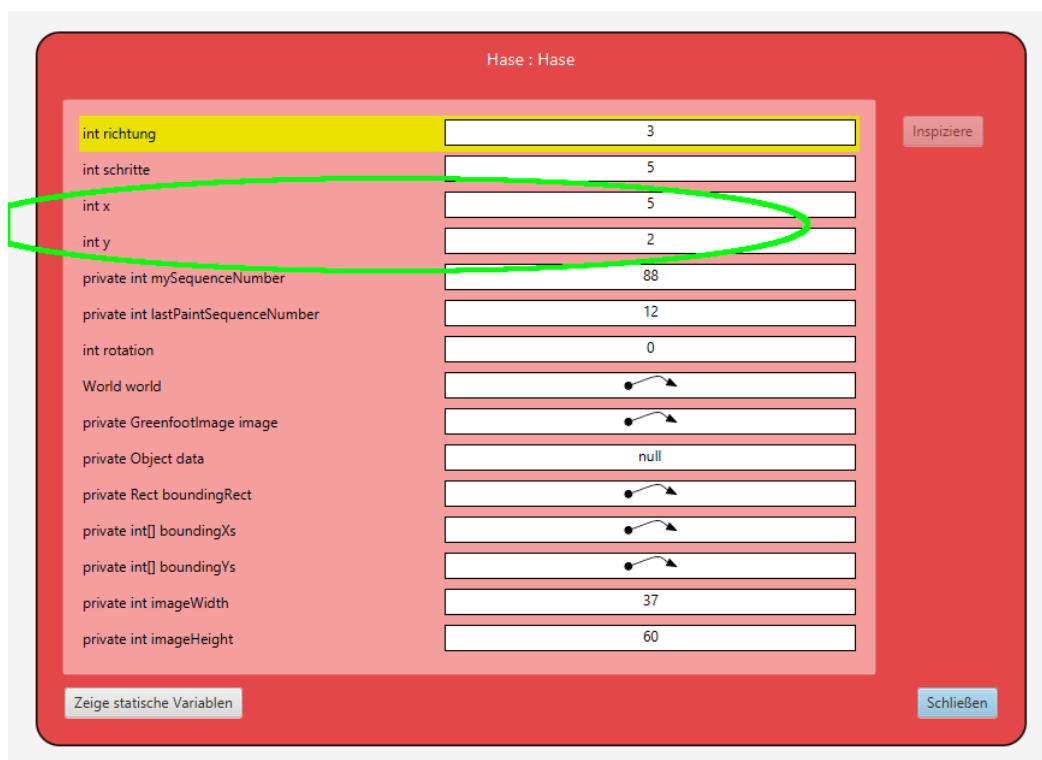
Wenn man also jetzt die zurückgegebenen Werte mit den zugehörigen Signaturen abgleicht, müsste auffallen, dass beide Werte Sinn ergeben.



1.5 Attribute

Das nächste erlernte Konzept ist das der Attribute.

Das geforderte Inspizieren eines Objektes ist ganz einfach durch einen Rechtsklick auf irgendeines der Objekte und das Auswählen von „Inspizieren“ möglich.



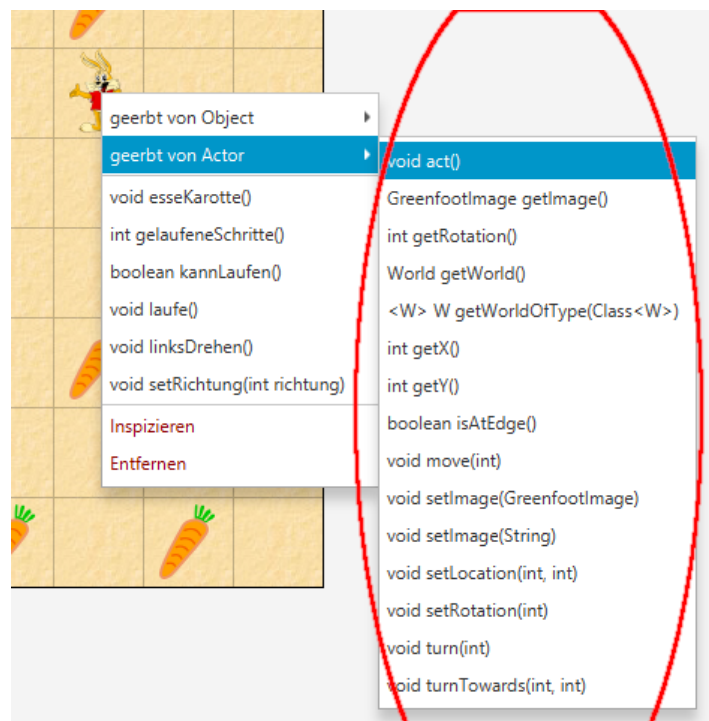
Die als Beispiel gesuchten Attribute für die Position des Objektes sind `int x` und `int y`.

1.6 Quelltext

Zu dieser Aufgabe sind meines Erachtens nach keine Anleitungen und/oder Erläuterungen nötig.
Wenn doch Fragen aufkommen, schreib' einfach an **nikodiamond3@gmail.com**.

1.7 Vererbung

Die markierten Methoden sind die von **Actor** vererbten Methoden, alle, welche unter „geerbt von Actor“ stehen, sind in der Klasse **Hase** selbst definiert.



Analog funktioniert es auch mit der Klasse **Elefant**.

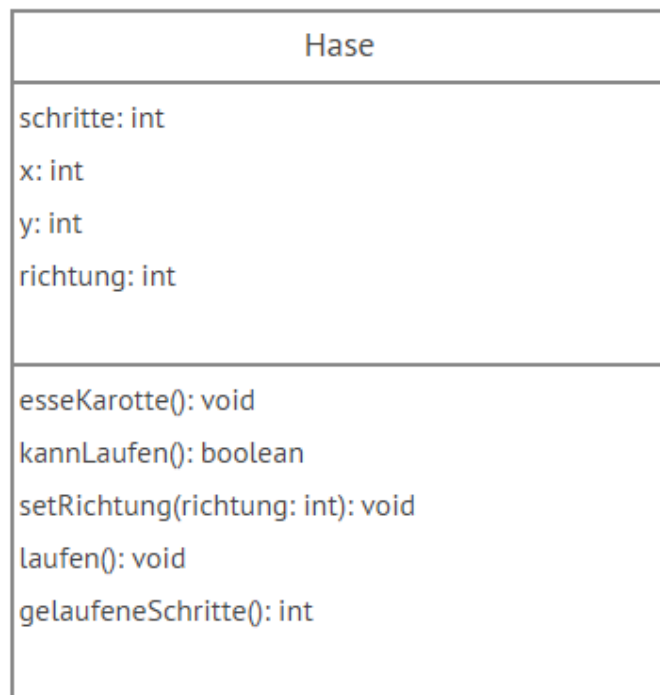
1.8 Klassendiagramm für Hase

Ganz oben im Klassendiagramm steht der Klassenname.

Darunter lassen sich die Attributnamen und dahinter deren Datentyp finden.

Ganz unten findet man die Methoden und angehängt deren Rückgabotyp. Die Parameter werden direkt hinter dem Methodennamen in Klammern angegeben (auch hier zuerst Variablenamen und dann der Datentyp).

Anschließend folgen noch die geforderten Ergänzungen im Klassendiagramm.



2 Fachkonzept - Objekt

Zu dieser Seite sind meines Erachtens nach keine Anleitungen und/oder Erläuterungen nötig.

Wenn doch Fragen aufkommen, schreib' einfach an **nikodiamond3@gmail.com**.

3 Fachkonzept - Klasse

Zu dieser Seite sind meines Erachtens nach keine Anleitungen und/oder Erläuterungen nötig.

Wenn doch Fragen aufkommen, schreib' einfach an **nikodiamond3@gmail.com**.

4 Übungen

4.1 Geburtstagsrechner

Hier folgt die Struktur der Beispielsklasse **Person**.

➤ Attribute

- tag vom Datentypen **int**
- monat vom Datentypen **int**
- jahr vom Datentypen **int**
- name vom Datentypen **String**

➤ Methoden

- **getAlter** mit **int** als Rückgabewert und ohne Parameter
- **getTageBisGeburtstag** mit **int** als Rückgabewert und ohne Parameter
- **setGeburtstag** ohne Rückgabewert und mit den Parametern **int t, int m, int y**
- **istVolljaehrig** mit **boolean** als Rückgabewert und ohne Parameter
- **setName** ohne Rückgabewert und mit dem Parameter **String name**
- **druckeInfo** ohne Rückgabewert und ohne Parameter

Alfred: Person
tag: 4 monat: 8 jahr: 2002 name: Alfred

Im Screenshot wurden folgende Schritte ausgeführt:

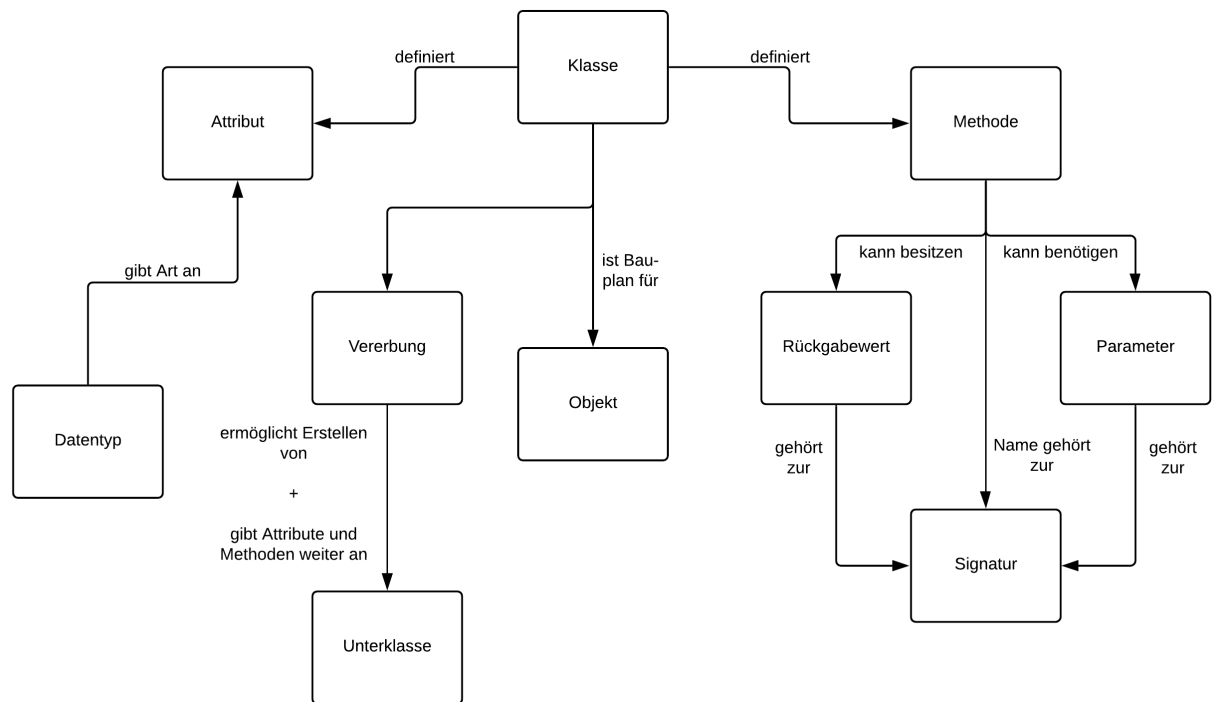
- Erstellen zwei neuer Objekte der Klasse `Person` mit den Namen `a` und `b`
- Inspizieren des Objektes `a`
- Aufrufen der `setName`-Methode an `a` mit dem Parameter `name = "Anna"`
- Aufrufen der `setGeburtsTag`-Methode an `a` mit den Parametern `geburtsTag = 10`,
`geburtsMonat = 4` und `geburtsJahr = 2000`
- Aufrufen der `getAlter`-Methode an `a`
- Aufrufen der `istVolljaehrig`-Methode an `a`
- Aufrufen der `getTageBisGeburtsTag`-Methode an `a`
- Aufrufen der `druckeInfo`-Methode an `a`

4.2 Zusammenfassung der Begriffe

Begriff	Definition
Klasse	Eine Klasse ist ein Bauplan für Objekte. Dieser Bauplan legt genau fest, welche Attribute die zu konstruierenden Objekte haben sollen und welche Methoden sie ausführen können sollen.
Objekt	Ein Objekt ist eine Einheit, die Daten mit Hilfe von Attributen verwalten und Operationen zur Verarbeitung der verwalteten Daten mit Hilfe von Methoden ausführen kann.
Attribut	Attribute sind - an Objekte gebundene - Variablen zur Verwaltung von Daten. Diese entsprechen in der Regel den Eigenschaften der betreffenden Objekte.
Methode	Methoden sind - an Objekte gebundene - Fähigkeiten zur Verarbeitung von Daten. Diese Methoden werden ausgeführt, wenn das betreffende Objekt veranlasst wird, eine bestimmte Operation auszuführen.

Parameter	Parameter sind Variablen, die dazu dienen, einer Methode beim Aufruf bestimmte Informationen zu übergeben. ¹
Rückgabewert	Rückgabewerte sind diejenigen Werte, welche von Methoden als Antwort geliefert werden.
Datentyp	Ein Datentyp gibt an, welche Art von Information ein Attribut hält.
Signatur	Eine Signatur einer Methode gibt den Rückgabewert, den Namen und die Parameter einer Methode an.
Vererbung	Die Vererbung ist eine Vorgehensweise, um von einer Klasse ausgehend eine Unterklasse definieren zu können, welche alle Attribute und Methoden der Überklasse übernimmt und auch abrufen kann.
Klassendiagramm	Ein Klassendiagramm ist ein Hilfsmittel, um Klassen inklusive ihrer Attribute und Methoden und deren Beziehungen einfach und übersichtlich darzustellen. Meist sind diese Diagramme in der Unified Modelling Language (UML) geschrieben.
Objektdiagramm	Ein Objektdiagramm ist ähnlich aufgebaut wie ein Klassendiagramm, nur dass der Name des beschriebenen Objektes dargestellt wird, anstatt der Datentypen Werte eingesetzt sind und meist die Methoden weggelassen werden, da diese bereits im Klassendiagramm wiederzufinden sind.

¹ Parameter, Informatik, <https://www.u-helmich.de/inf/BlueJ/lexikon/M-R/Parameter.html>, (Abgerufen: 24.02.20, 00:22)



Den Screencast lasse ich aus, das kannst du ja selber erledigen, wenn du es noch machen möchtest.

4.3 Klassen- und Objektdiagramme



4.4 Kreuzworträtsel

