

Objekte

The image shows the BlueJ IDE interface. On the left, there is a sidebar with buttons for 'Neue Klasse...', 'Übersetzen', 'Teamwork', 'Share...', 'Testing', and 'Aufzeichnung'. The main workspace displays a class hierarchy diagram with classes: Leinwand, Kreis, Quadrat, Dreieck, and Person. Dashed lines indicate relationships: Leinwand is associated with Kreis, Quadrat, and Dreieck. Kreis, Quadrat, and Dreieck are subclasses of Person. A window titled 'BlueJ Figuren Demo' is open, showing a green triangle, a blue circle, and a red square. At the bottom, there are three object monitors: 'kreis1: Kreis', 'quadrat1: Quadrat', and 'dreieck1: Dreieck'. The status bar at the bottom left shows 'kreis1 : Kreis'.

Grundbegriffe der OOP:



Klasse

Objekt

Methode

Attribut

Grundbegriffe der OOP

Klasse Eine Vorlage für ein „Objekt“. („Bauplan“ für Objekte)

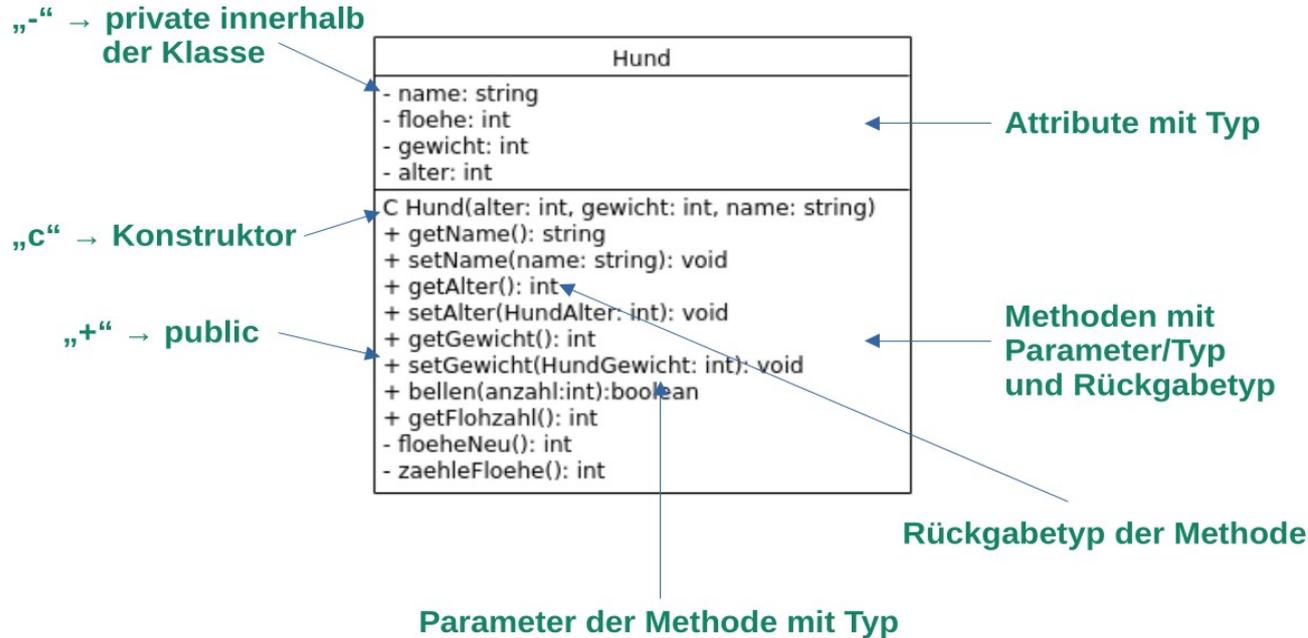
Objekt Gehört zu einer bestimmten Klasse, ein Objekt hat individuelle Eigenschaften. Wird aus einer Klasse „instanziiert“.

Methode Steuert das Verhalten von Objekten

Attribut Speichert den Zustand eines individuellen Objekts

Grundbegriffe der OOP

Darstellung: UML Diagramm



Methoden und Parameter

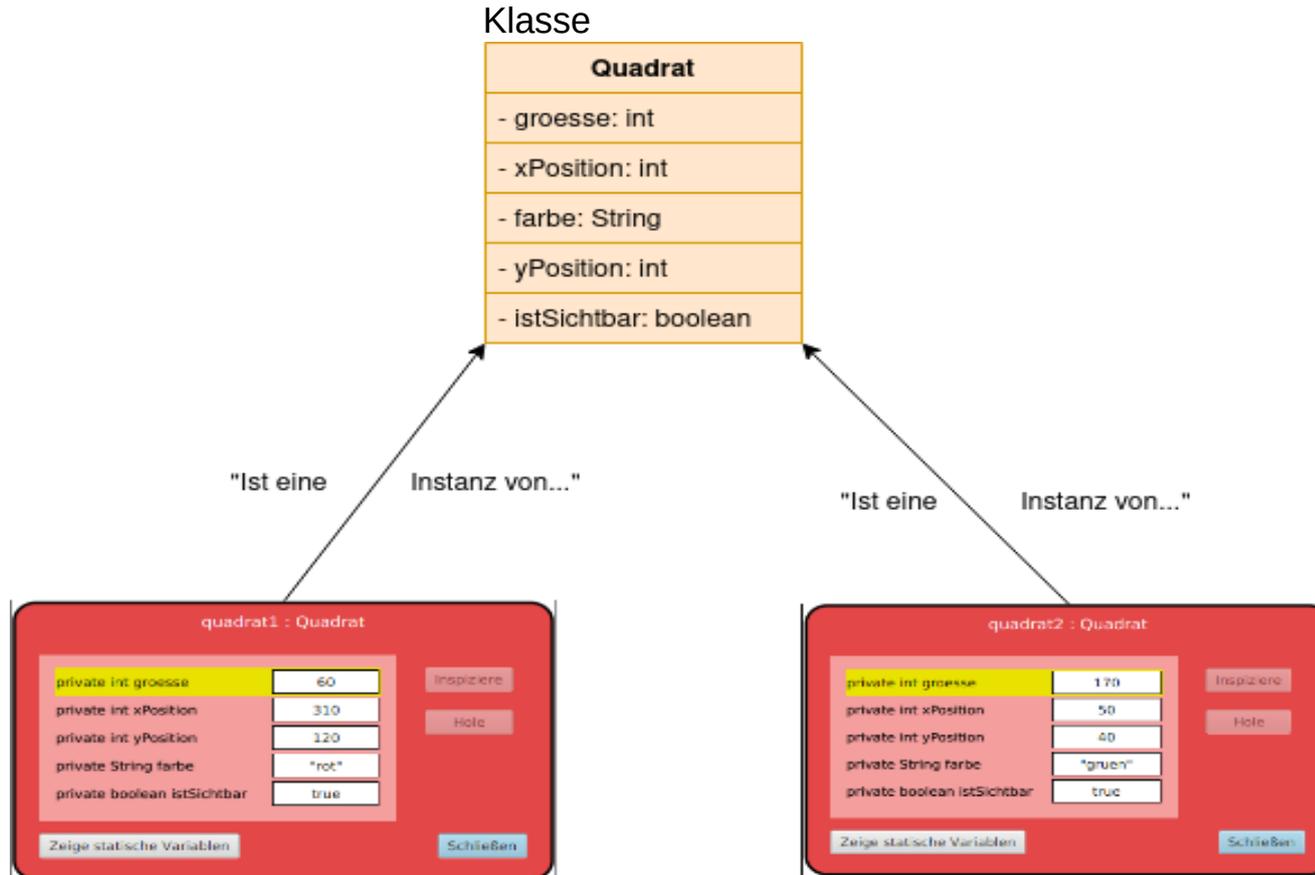
Das **Verhalten** und der **Zustand** von Objekten kann durch die **Methoden** eines Objekts beeinflusst werden.

Methoden haben häufig **Parameter**, mit denen man beim Aufruf einer Methode zusätzliche Information übermitteln kann. Je nach Parameter kann ein Methodenaufruf das Verhalten oder den Zustand des Objekts verschieden beeinflussen.

Aus einer **Klasse** können viele verschiedene Objekte erzeugt werden („instanziiert“). Man spricht auch von „**Instanzen einer Klasse**“. Ein Objekt ist also eine „Instanz einer Klasse“.

Die **Klasse** legt fest, welche **Attribute (Datenfelder)** ein Objekt hat, jedes Objekt kann dort aber **eigene Werte** speichern

Objektzustand



Objekte/Instanzen der Klasse

Methodensignatur

- Name
- Anzahl, Typ und Reihenfolge der Parameter
- Typ des Rückgabewerts

```
public void printFunktionswert(int y)
```

```
public void printFunktionswert(double y, int n)
```

```
public String printFunktionswert(double y, int n)
```

3 unterschiedliche Funktionen!