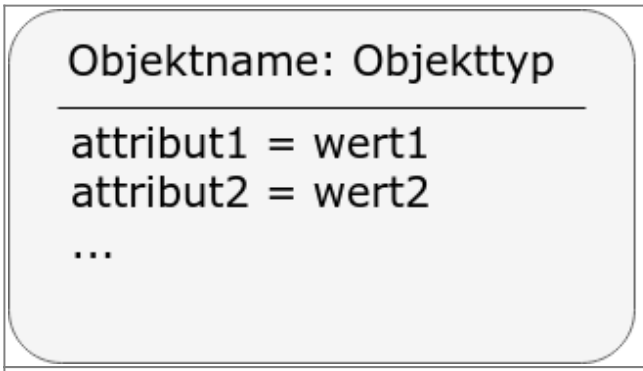
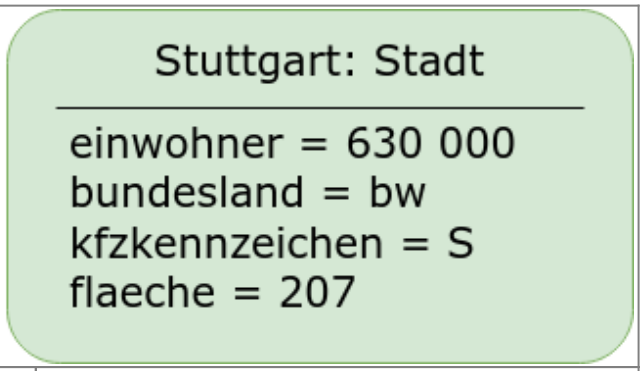
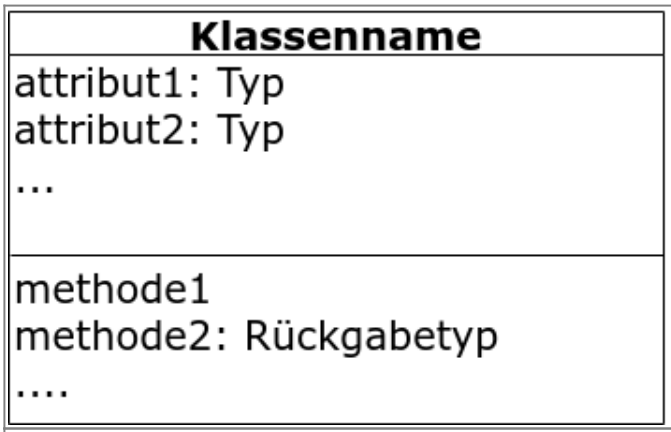
Darstellungselemente der Modellierung

Objektdiagramme

	
Allgemein	Beispiel

- Die Kopfzeile benennt das Objekt und gibt seinen Typ/seine Klasse an
- Attribute sind konkret mit Werten angegeben
- Die Attribute geben den Zustand des Objekts an, wenn man sie verändert, ändert sich der Objektzustand
- Die Methoden der Klasse, aus denen die Objekte instanziiert wurden werden in den Objektdiagrammen nicht angegeben.

Entwurfsdiagramme

	
Allgemein	Beispiel

- Ein Schritt näher an der Programierung, Verallgemeinerung des Objektdiagramms
- Unabhängig von der Programmiersprache
- Wichtige Attribute werden mit "umgangssprachlichem" Typ angegeben
- Wesentliche Methoden werden mit Rückgabetyp angegeben
- *Keine* Setter- und Getter- Methoden, keine Konstruktoren

Implementationsdiagramm

Das Implementationsdiagramm ist auf eine bestimmte Programmiersprache ausgerichtet: Datentypen bei Attributen und Rückgabewerten werden spezifisch für die Zielsprache angegeben. Der Konstruktor

wird passend zur Programmiersprache angegeben, ebenso die Sichtbarkeiten von Attributen und Methoden. Im Implementationsdiagramm werden auch Setter- und Getter-Methoden angegeben.

Man kann sich das Implementationsdiagramm als "Todo-Liste" zur Implementation in einer bestimmten Programmiersprache vorstellen.

Klassenname	Schueler
- attribut1: typ - attribut2: typ ...	- vname: String - nname: String - alter: int - versetzt: boolean - klasse: String
© Konstruktor(param: typ, param: typ, ...) + methode1(param: typ): void + methode2(param: typ): rueckgabetyp ... + getAttribut1(): rueckgabetyp + setAttribut1(param: typ): void ...	© Schueler(vn: String, nn: String, alter: int, kl String, ver: boolean) + getAlter(): int + setAlter(nalter: int): void
Allgemein	Beispiel

Assoziationen

Im objektorientierten Entwurf einer Miniwelt spielen Beziehungen zwischen den Klassen des Modells eine große Rolle - solche Beziehungen zwischen den Klassen nennt man **Assoziationen**.

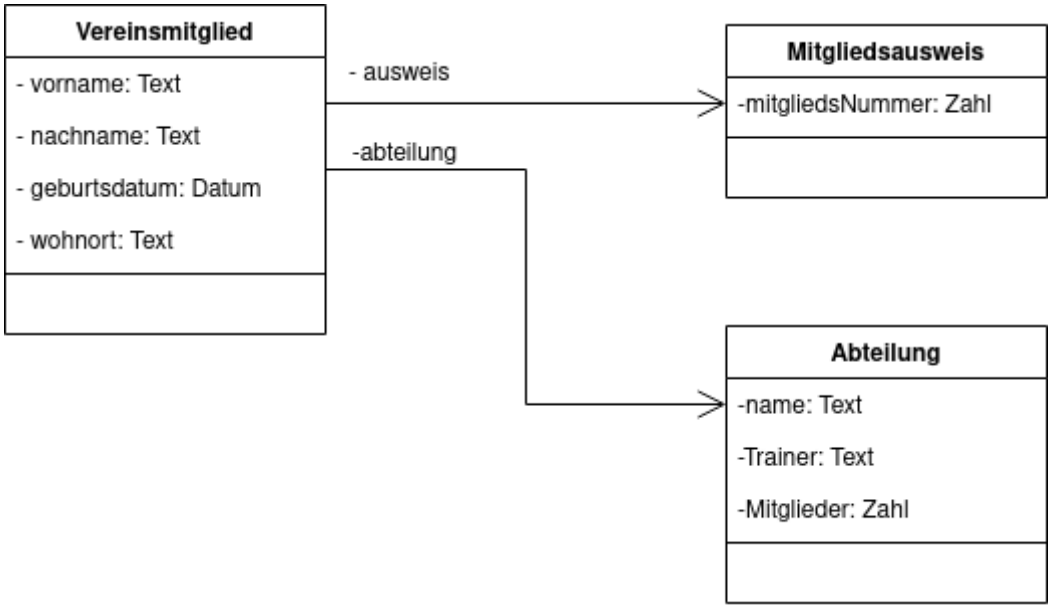
- Eine Assoziation wird häufig über ein **Attribut** in der assoziierenden Klasse realisiert.
- Die Beziehungen können jedoch auch über eine Steuerklasse (Assoziationsklasse) realisiert werden, welche die Beziehungen zwischen den Klassen des Modells verwaltet.

Darstellungsmöglichkeiten im UML Diagramm:

Möglichkeit A

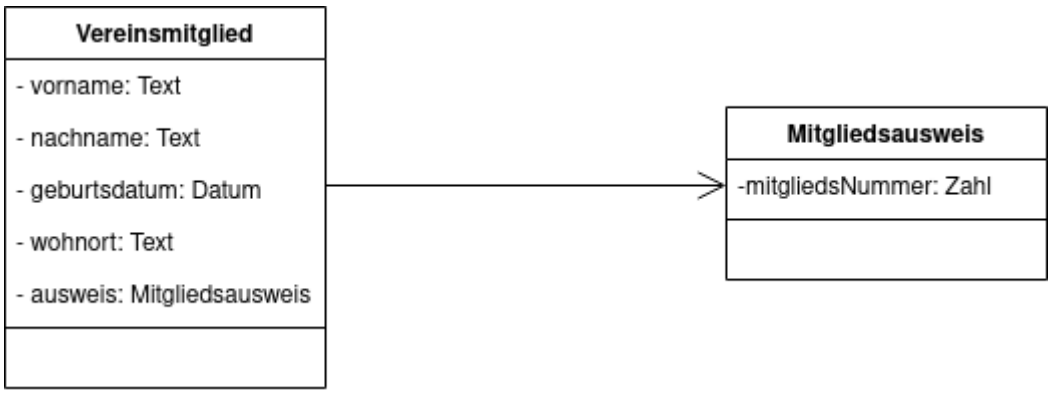
Attribute am Assoziationspfeil: Hier werden die Attribute, welche die Verbindung zur assoziierten KLasse herstellen, an den Pfeil geschrieben und in der Liste der Klassenattribute weggelassen. Man sieht sofort, welöches Attribut die Beziehung herstellt, der Typ des Attributs ist stets die assoziierte Klassen.

Wenn es mehrere Assoziationen gibt sieht dieses Darstellung folgendermaßen aus:

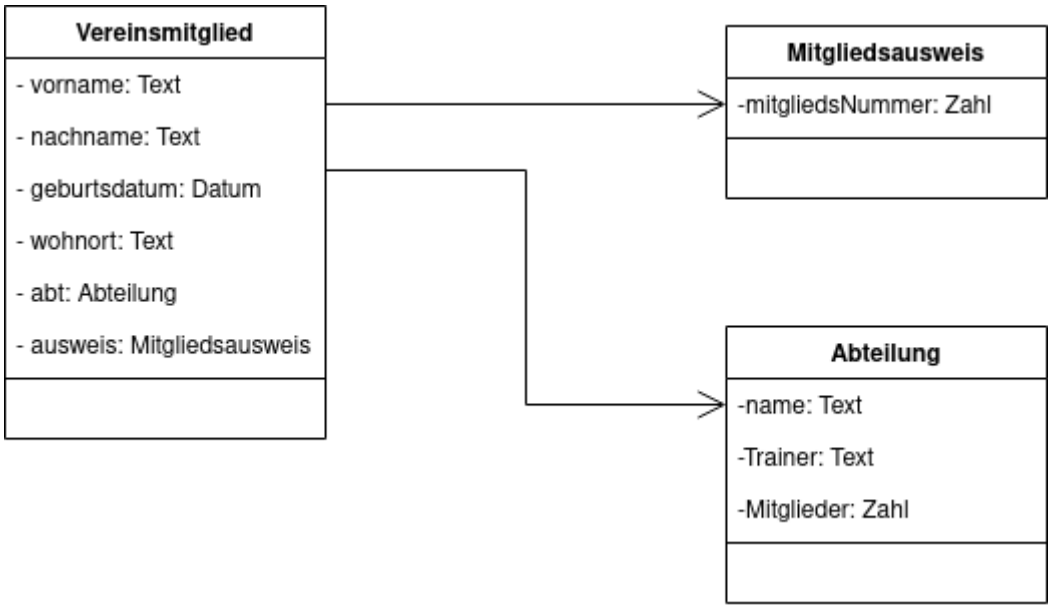


Möglichkeit B

Der Assoziationspfeil verbindet lediglich die Klassen, dem Pfeil kann man nicht ansehen, welches Attribut die Verbindung realisiert. Das erkennt man erst, wenn man die Typen der Attribute genau betrachtet.



Bei mehreren Assoziationen ergibt sich das folgende Bild:



Multiplizitäten

Um auf einen Blick erkennen zu können, ob eine Assoziation einzelne Objekte einer Klasse mit der Steuerklasse verbindet oder mehrere Objekte derselben Art, wie es bei Sammlungen und ADTs der Fall ist, bringt man an der Pfeilspietze des Assoziationspfeils für gewöhnlich noch eine Markierung für die **Multiplizität** des Assoziation an:

Multiplizität	**Erläuterung**
1	Ein Objekt der Klasse assoziiert genau ein Objekt der assoziierten (anderen) Klasse
0..1	Ein Objekt der Klasse assoziiert eines oder keines der assoziierten Objekte
0..*	Ein Objekt der Klasse assoziiert beliebig viele (oder keines) der andern Objekte
n..*	Ein Objekt der Klasse assoziiert beliebig viele, aber mindestens n der andern Objekte

From:
https://info-bw.de/ -

Permanent link:
https://info-bw.de/faecher:informatik:oberstufe:modellierung:uml:start

Last update: **02.02.2022 12:29**