

Objektorientierte Programmierung (OOP)

- OOP ist ein Programmierparadigma, dh. es ist eine Methode Programme zu erstellen.
- Man kann in jeder Programmiersprache objektorientiert programmieren, aber objektorientierte Programmiersprachen erleichtern das objektorientierte Programmieren.
- Ein wichtiges Ziel von OOP ist es, große Programme leichter warten zu können.



Objekt und Klasse

- Ein Objekt besteht aus Daten und Methoden diese Daten zu manipulieren: Den Attributen des Objekts.
- Eine Klasse ist ein Bauplan für ein Objekt.
- Jedes Objekt ist Instanz einer Klasse.
- Jedes Objekt kennt seine Klasse, aber nicht jede Klasse weiß welche Objekte von ihr instanziiert wurden.



Konstruktoren

- Die Laufzeitumgebung einer objektorientierten Programmiersprache verfügt über einen Operator um aus einer Klasse ein Objekt zu erstellen. Dieser Operator heißt Konstruktor.
- In Klassen kann man Initialisierungsmethoden definieren, die gegebenenfalls von dem Konstruktor aufgerufen werden.
- Diese Initialisierungsmethoden werden oft auch als Konstruktoren bezeichnet.

Geheimnisprinzip

- Auf die Daten eines Objekts soll nur das Objekt selbst zugreifen können. Dies nennt man Geheimnisprinzip oder Kapselung.
- Daten, die von außen manipuliert werden sollen, werden dann mit `get` und `set` Methoden versehen.
- Ein Objekt kann so als “black box” verstanden werden.



Vererbung

- Klassen können von einander Attribute erben.
- Methoden können dann von der Superklasse übernommen oder mit einer neuen Implementierung überladen werden.
- Auf diese Weise kann Polymorphie erreicht werden: Methoden gleichen Namens für unterschiedliche Datentypen.



Abstrakte Klassen

- Werden Methoden in einer Klasse zwar deklariert, aber nicht implementiert, so spricht man von einer abstrakten Klasse.
- Von einer abstrakte Klasse kann kein Objekt instanziiert werden.
- Eine Klasse die von einer abstrakten Klasse erbt, erbt die Verpflichtung die offen gebliebenen Methoden zu implementieren.

