

Das diesem Dokument zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 16OH21005 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor/bei der Autorin.

GEFÖRDERT VOM



Bundesministerium
für Bildung
und Forschung



Warum objektorientierte Programmierung?

Grundlagen der Programmierung

In diesem Abschnitt wollen wir uns mit der Frage beschäftigen:
Warum ist objektorientierte Programmierung sinnvoll?

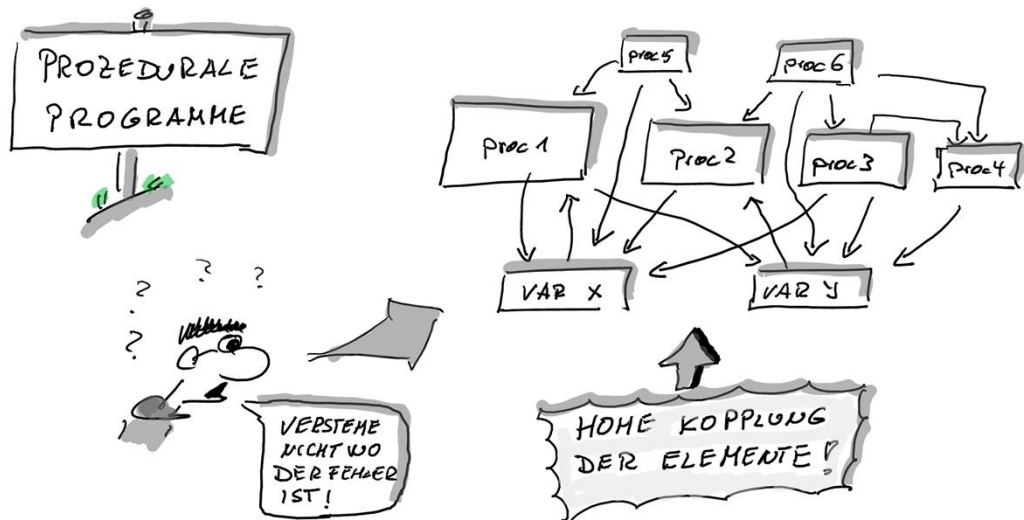
Es gilt also zu ergründen:

Worin der Vorteil der objektorientierten Programmierung liegt, gegenüber der prozeduralen Programmierung?

- ➔ Was ist das Problem bei prozeduraler Programmierung?
- ➔ Wie ist das in OOP gelöst?

Hierzu sehen wir uns zunächst an, was das Problem bei der prozeduralen Programmierung ist.

Anschließend schauen wir uns an, wie dieses bei der objektorientierten Programmierung gelöst wird.



Wie der Namen so sagt, bestehen bei der prozeduralen Programmierung die Programme aus Funktionen bzw. Prozeduren und Variablen.

Meist sind viele Variable global zugreifbar. Dies ist in der Abbildung dargestellt. Aus verschiedenen Prozeduren heraus (z.B. proc, pro2, proc3) wird schreibend auf die Variablen (hier Variable X und Variable Y) zugegriffen.

Hierbei verliert man schnell den Überblick.

Fragen wie:

- Welche Prozedur greift zu welchem Zeitpunkt auf welche Variable zu?
- Warum hat der Inhalt einer Variable zu dem Zeitpunkt t1 den Wert X?

Ursache hierfür ist, die **hohe Kopplung** der einzelnen Prozeduren.

Mit Kopplung ist hier gemeint, dass zwischen den Prozeduren durch die Variablen eine starke Abhängigkeit besteht.

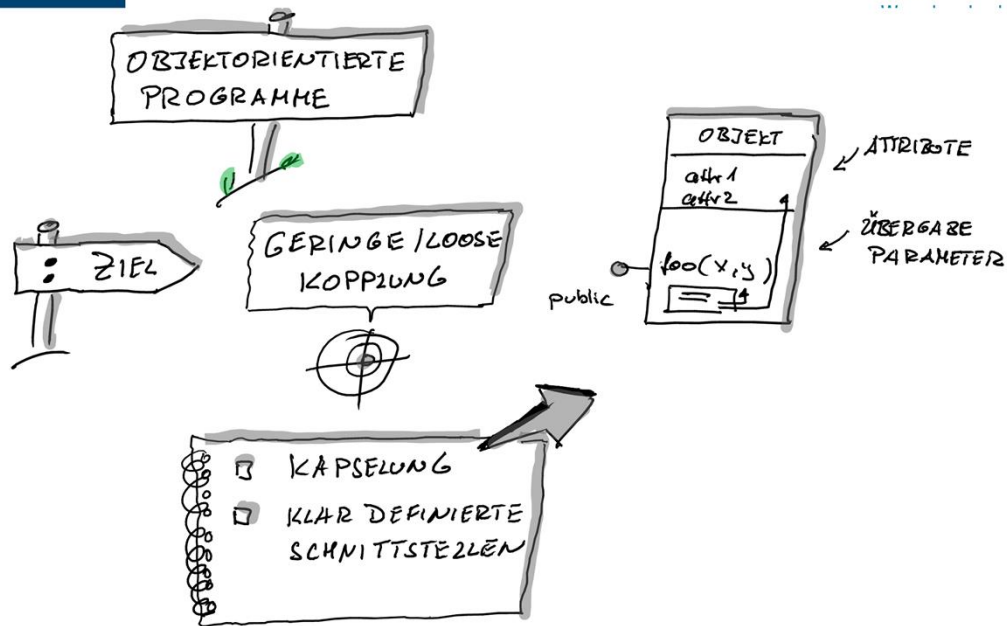
Um Programme

- verständlicher
- wartbarer

- und änderungsfreundlicher

zu machen, ist es notwendig, dass eine Prozedur möglichst keine oder nur geringe Abhängigkeiten besitzt.

→ Ziel ist also eine möglichst geringe Kopplung. (engl. Loos Coupling)



18. Januar 2017

Modul 8338
Grundlagen der Programmierung

- 4 -

Ein Hauptziel der objektorientierten Programmierung ist eine möglichst lose Kopplung.

Hierfür werden bei der objektorientierten Programmierung Klassen definiert. Diese können als Datentyp aufgefasst werden. Instanzen der Klassen nennt man Objekte.

Die Objekte verfügen über eine klar definierte Schnittstelle. Bei der Erstellung einer Klasse wird festgelegt, welche Methoden und Attribute außerhalb der Klasse sichtbar sind.

→ vgl. Sichtbarkeit in Java durch public, protected, private.

Somit wird durch die Möglichkeit

- der Kapselung
 - und der klar definierten Schnittstelle
- eine geringere Kopplung erreicht.

Eine Kopplung über globale Variablen lässt sich somit vermeiden

Bei OOP kommunizieren also Objekte über fest definierte Schnittstellen miteinander.

➔ Nachteile von prozeduralen Programmen

- Unklare Schnittstellen
- Hohe Kopplung der einzelnen Komponenten
- Schwer verständliches Verhalten
- Hoher Wartungsaufwand
- Geringe Wiederverwendbarkeit der Komponenten

➔ Vorteile der OO-Programmierung

- Kapselung durch Klassen
- Klare Schnittstellen durch Sichtbarkeitsregeln
- Geringe Kopplung
- Methoden einer Klassen haben eine hohe Bindung