

# LU10d - Anpassen und Erweitern

Das Konzept der Vererbung weist zwei grundlegende Aspekte betreffend der Auswirkungen auf die Programmausgestaltung aus. Eine Unterklasse kann Code erweitern oder Code in angepasster Form ausführen.

## Erweitern

Das Wort „erweitern“ sagt bereits klar aus, was hier Sache ist. Eine Unterklasse erhält in diesem Fall zusätzliche Attribute und Methoden.

## Beispiel

SalaryAccount erweitert BankAccount um die Funktion für den Kontoüberzug. Hier mit dem Attribut `overdraft` und der Property `overdraft` dargestellt. Auf der rechten Seite sehen Sie die resultierende Klasse, bestehend aus den geerbten Elementen und den Erweiterungen in SalaryAccount.

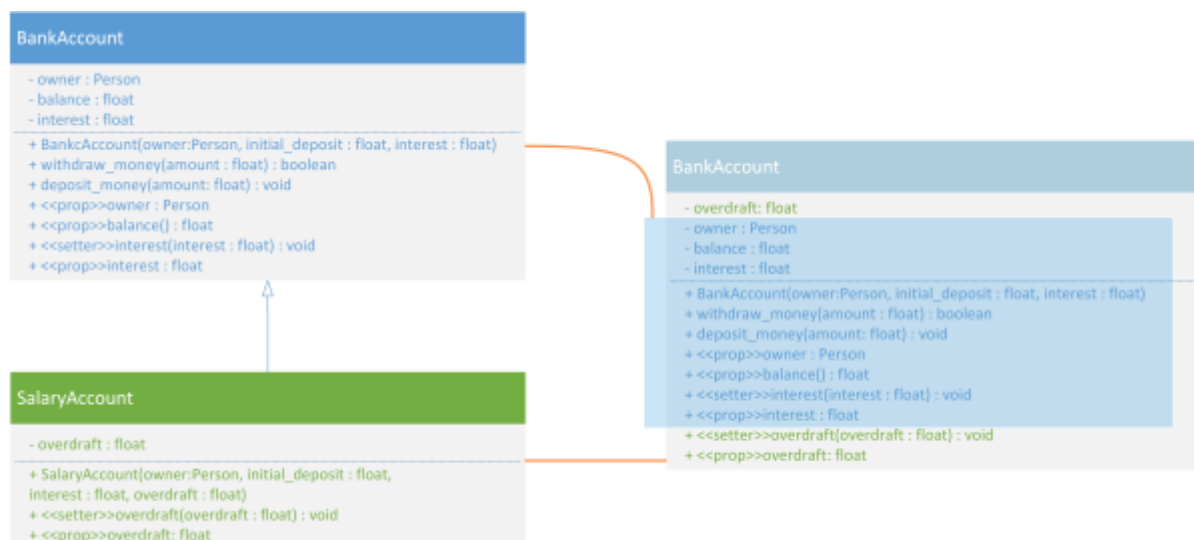


Abb: Elemente einer erweiterten Klasse

Ein Objekt der Klasse SalaryAccount ist somit immer auch ein Objekt der Klasse BankAccount.



Bei der Vererbung weist die abgeleitete Klasse (Unterklasse) immer auch den Typ - die Klasse - der Oberklassen auf.

# Anpassen

Wird in einer Unterklasse eine „Anpassung“ vorgenommen, so heisst dies, dass eine gegebene Funktionalität (eine Methode) anders ausgeführt wird. In der Fachsprache nennt man dies auch „**überschreiben**“ (**overwriting**) einer Methode.

## Beispiel

SalaryAccount überschreibt die Methode `withdraw_money()` aus `BankAccount`. So ist es nun möglich, durch den Aufruf der „gleichen“ Methode - zumindest was den Namen der Methode betrifft - einen anderen Ablauf auszuführen. Während `BankAccount` keine Bezüge zulässt, die zu einem negativen Saldo führen, kann `SalaryAccount` dies tun.



Abb: Visualisierung des Überschreibens in einer abgeleiteten Klasse

M320-LU10



René Probst, bearbeitet durch Marcel Suter

From:  
<https://wiki.bzz.ch/> - **BZZ - Modulwiki**

Permanent link:  
[https://wiki.bzz.ch/modul/m320\\_2024/learningunits/lu10/unterklasse](https://wiki.bzz.ch/modul/m320_2024/learningunits/lu10/unterklasse)

Last update: **2025/03/27 09:14**

