

## Klassenhierarchie §12

Keywords: Vererbung, virtuelle Methoden, Überladen von Methoden, Polymorphismus

Keywords: Basisklasse, abgeleitete Klasse, abstrakte/konkrete Klasse, VMT

Keywords: Basisklassenpointer, `shared_ptr`, `unique_ptr`, dynamic casting

Finales Ziel: STL auf Containern mit polymorphen Elementen.

[v\\_10c\\_shared/](#)

### Intro

Zwei Mechanismen bestimmen das Design einer Klassenhierarchie:

[class\\_0/Intro\\_Klassenhierarchie.pdf](#)

- **IS-A** Relation: von *Basisklasse* abgeleitete Klasse.  
Jede abgeleitete Klasse **ist eine** Basisklasse.  
Jeder Mercedes **ist ein** Auto.
- **HAS-A** Relation: abgeleitete Klasse **hat zusätzliche** Eigenschaften.  
Jeder Mercedes **hat zusätzlich einen Stern** zu den 4 Rädern des Autos.
- Technische Umsetzung mit Klassen A und B zeigen.

[class\\_0/class\\_0\\_annotated.pdf](#)

### Angestelltenhierarchie §12.1

Einführungsbeispiel: Employee, Worker, salesPerson, Manager.

[v\\_9b/](#)

- Entwurf (**IS-A**; **HAS-A**)
- Initialisierung der Basisklasse via *member initialization list* im Konstruktor der abgeleiteten Klasse.
- Zugriffsrechte: private, protected, public
- Überschreiben von Methoden (*virtual method*).
- Polymorphismus mit Referenz `&Employee` in Parameterliste und VMT zur Auswahl der überschriebenen Methode zur Laufzeit.  
Kein Polymorphismus mit Kopie Employee, da erfolgt ein Upcasting (Reduktion auf Eigenschaften der Basisklasse).

## Literatur

- [Haase22] Gundolf Haase: Einführung in die Programmierung mit C++ (2022), *www*<sup>1</sup>.
- [Stroustrup10] Bjarne Stroustrup: Einführung in die Programmierung mit C++. Pearson Studium, München (2010).

---

<sup>1</sup>[http://imsc.uni-graz.at/haasegu/Lectures/Kurs-C/Script/html/script\\_programmieren.pdf](http://imsc.uni-graz.at/haasegu/Lectures/Kurs-C/Script/html/script_programmieren.pdf)