

Objektorientierte Programmierung in Java

Vorlesung 3 - Klassen und Objekte

Emily Lucia Antosch

HAW Hamburg

09.10.2024

Inhaltsverzeichnis

1. Einleitung	3
2. Klassen und Objekte	7
3. License Notice	11

1. Einleitung

1.1 Wo sind wir gerade?

- Zuletzt haben wir uns mit den imperativen Konzepten der Programmiersprache Java beschäftigt.
- Sie können nun
 - ▶ einfache Datentypen in Java verwenden,
 - ▶ den Programmfluss mit Kontrollstrukturen und Schleifen steuern und
 - ▶ Datentypen konvertieren.
- Heute geht es um **Klassen und Objekte**.

1.1 Wo sind wir gerade?

1. Imperative Konzepte
2. **Klassen und Objekte**
3. Klassenbibliothek
4. Vererbung
5. Schnittstellen
6. Graphische Oberflächen
7. Ausnahmebehandlung
8. Eingaben und Ausgaben
9. Multithreading (Parallel Computing)

1.2 Das Ziel dieses Kapitels

- Sie implementieren Klassen und Objekte in Java, um reale Dinge abzubilden.
- Sie erzeugen Objekte einer Klasse und ändern deren Zustand über Operationen.
- Sie wenden zusätzliche Programmierrichtlinien an, um die Qualität und die Wartbarkeit Ihres Codes zu verbessern.

2. Klassen und Objekte

! Merke

- Eine **Klasse** ist ein Bauplan für Objekte. Sie enthält
 - ▶ **Attribute** (Datenfelder) und
 - ▶ **Methoden** (Operationen).
- Zusammen heißen Attribute und Methoden **Members**.

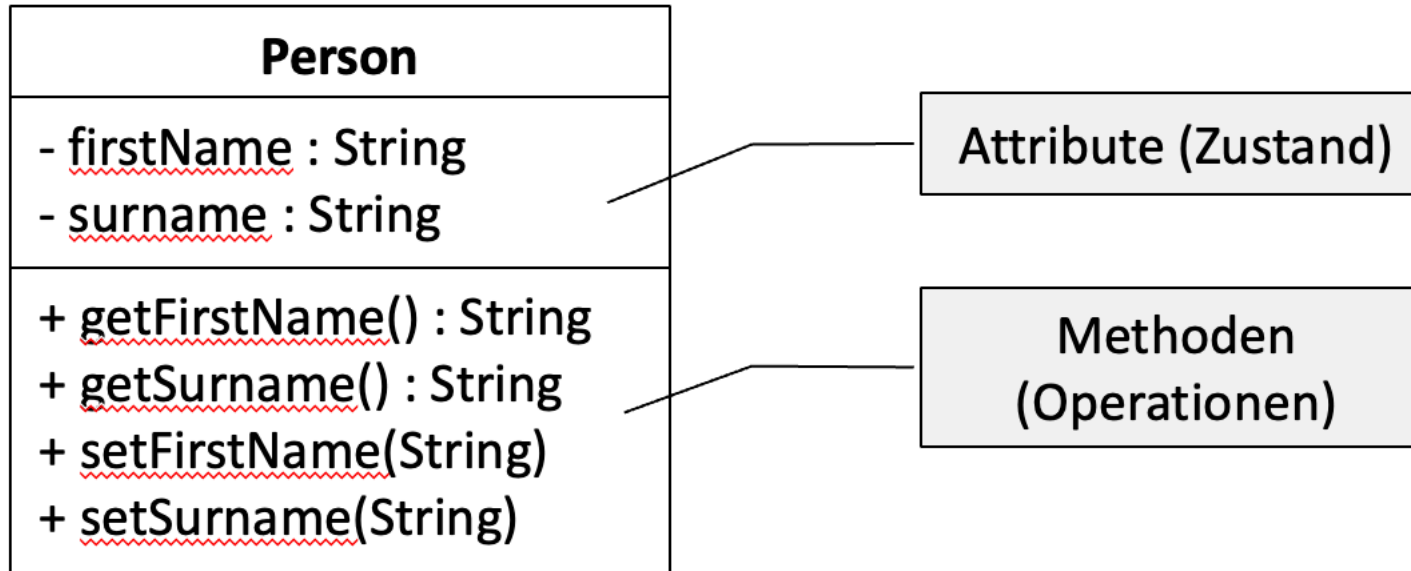


Abbildung 1: UML-Notation einer Klasse Person

! Merke

- Zur Laufzeit im Speicher erzeugter Datensatz einer Klasse
- Variablen beschreiben **Zustand** des Objekts
- Methoden beschreiben **Fähigkeiten** des Objekts
- Bezeichnungen für Variablen: **Attribute**, **Objektvariablen**, **Instanzvariablen**

3. License Notice

3.1 Attribution

- This work is shared under the CC BY-NC-SA 4.0 License and the respective Public License
- <https://creativecommons.org/licenses/by-nc-sa/4.0/>
- This work is based off of the work Prof. Dr. Marc Hensel.
- Some of the images and texts, as well as the layout were changed.
- The base material was supplied in private, therefore the link to the source cannot be shared with the audience.