

# Records in Java

## Anwendungsfälle

- DTOs / API- Responses
- Config- Objekte
- Wertobjekte
- EndpointDefinition

## Erweiterungen

- Kompakter Konstruktor
- Validierung
- if (methode.isBlank()) throw...
- Zusätzliche Methoden
- vollstaendigeBeschreibung()

## Vergleich Klasse

- Boilerplate: 22 Zeilen
- equals/ hashCode manuell
- final manuell setzen
- Getter: getName()

## Einführung

- Seit Java 16
- Java 14/ 15 Preview
- Datenklassen

## Syntax

- record Name(Komp1, Komp2) {}
- Komponenten = final Felder
- Getter: name() nicht getName()
- Kanonischer Konstruktor

## Vorteile

- Weniger Boilerplate
- Wertgleichheit
- Unveränderlich
- Semantische Klarheit
- Compiler+ optimiert