

Java-Online-Gruppentutorium

Givara Ebo

August 10, 2025

Willkommen!

- ▶ Zeitraum: **September – März** (\approx 25 Wochen)
- ▶ Zeitpunkt: **Samstags, 3 Stunden** (z. B. 10:00–13:00)
- ▶ Format: **Live-Online** (Videokonferenz)
- ▶ Gruppengröße: **max. 10 Teilnehmer**
- ▶ Community: **Discord-Server** für Fragen, Code, Q&A

Voraussetzungen & Setup

- ▶ **JDK 17+** (oder aktuell unterstützt)
- ▶ IDE: **IntelliJ IDEA** (+ Java-Extension)
- ▶ **Git** installiert, Accounts bei **GitHub/GitLab**
- ▶ **Discord**-Account
- ▶ Kursmaterial: **Beispielcodes, Vorlagen, Mini-Projekte**
(Repo)

Unterstützung & Arbeitsweise

- ▶ Direkte Hilfe zu **Hausaufgaben, Übungsblättern, Projekten**
- ▶ **Abschlussprojekte** betreuen (Konzept, Technik, Methodik)
- ▶ **Teamarbeit** verbessern: Mini-Projekte, klare Workflows
- ▶ Code-Reviews, Debugging, wöchentliche **Q&A** auf Discord

Woche 1

- ▶ Setup: JDK, IDE, Git; Kurs-Repo klonen
- ▶ Java-Grundlagen I: **Hello World**, Variablen, Datentypen
- ▶ Kontrollstrukturen: if/else, switch, Schleifen
- ▶ Git-Intro: **commit, push, pull**; Branch-Grundlagen
- ▶ **Übung (live)**: Mini-CLI-Programm
(Eingabe/Verarbeitung/Ausgabe)
- ▶ **Hausaufgabe**: Dateien auslesen, bearbeiten und Ausgeben

Woche 2

- ▶ OOP I: Klassen, Objekte, Methoden, Kapselung
- ▶ Pakete, Sichtbarkeiten, toString/equals/hashCode
- ▶ **Maven**-Grundlagen: Projektstruktur, Dependencies, Build
- ▶ Git-Workflow: Branch erstellen, **Pull/Merge Request** anlegen
- ▶ **Übung (live)**: Klassenmodell Student/Kurs/Anmeldung
- ▶ **Hausaufgabe**: kleine Konsolen-App mit Maven bauen

Woche 3

- ▶ OOP II: Vererbung, Polymorphie, Schnittstellen
- ▶ Collections: List, Set, Map; Iteration, Streams (Intro)
- ▶ **Teamarbeit:** Issues, Code-Reviews auf GitHub/GitLab
- ▶ **UML Basics:** Use-Case & Klassendiagramm für das Wochenprojekt
- ▶ **Übung (live):** kleines Team-Feature mit PR/MR und Review
- ▶ **Hausaufgabe:** UML-Klassendiagramm + Implementierung

Woche 4

- ▶ Web & APIs I: HTTP/REST, JSON, Statuscodes
- ▶ **JavaSpark** (SparkJava) – erste GET/POST-Endpoints
- ▶ **Swagger/OpenAPI** – API dokumentieren und testen
- ▶ Client-Server-Basics, Test mit Postman/Insomnia
- ▶ **ER-Modell** (Intro): Entitäten, Beziehungen, Kardinalitäten
- ▶ **Übung (live)**: Mini-API Kurs/Teilnehmer
- ▶ **Hausaufgabe**: Endpoint erweitern + einfache Doku in Swagger