

# Kapitel 5 – Methoden: Zusammenfassung + Übungen

---

In diesem Kapitel lernst du, wie man in Java Methoden (Funktionen) definiert und verwendet. Methoden helfen dir, deinen Code zu strukturieren, wiederzuverwenden und lesbarer zu machen.

## Themenübersicht

### 1. Methodenstruktur

Syntax:

Rückgabetyt Methodenname(Parameter) { Anweisungen }

Beispiel: `int quadrat(int x) { return x * x; }`

### 2. Rückgabetypen

Verwende `void`, wenn nichts zurückgegeben werden soll. Sonst z. B. `int`, `double`, `String`, etc.

### 3. Parameter

Methoden können Eingabewerte bekommen: `int addiere(int a, int b)`. Übergabe erfolgt bei Aufruf.

### 4. Methodenaufruf

Einfach per Name aufrufen: z. B. `int ergebnis = addiere(2, 3);`

### 5. Überladen von Methoden

Gleicher Name, unterschiedliche Parameter:

`int add(int a, int b); double add(double a, double b);`

### 6. Statische Methoden

Mit `static` definierte Methoden können ohne Objekt verwendet werden – wie `main` oder `Math.sqrt()`.

### 7. Methoden verschachteln

Methoden können andere Methoden aufrufen, z. B.:

`int erg = quadrat(differenz(5, 2));`

## Übungsaufgaben

1. Definiere eine Methode `int quadrat(int x)`, die das Quadrat einer Zahl zurückgibt.

2. 2. Schreibe eine Methode `void begruessung()`, die 'Hallo Welt' ausgibt.
3. 3. Erstelle eine Methode `double add(double a, double b)`.
4. 4. Rufe in `main()` deine Methode `add()` mit zwei Zahlen auf und gib das Ergebnis aus.
5. 5. Überlade `add()`, sodass es auch zwei `int` addieren kann.
6. 6. Was passiert, wenn eine Methode `return`-los ist, aber du versuchst, ihr Ergebnis zu speichern?
7. 7. Erkläre den Unterschied zwischen `void` und `int` als Rückgabotyp.
8. 8. Schreibe eine Methode `String begruessung(String name)`, die 'Hallo <Name>' zurückgibt.