

- 2.5.1 Methoden aus dem JDK nutzen
- 2.5.2 Eigene Methoden definieren
- 2.5.3 Nützliche Beispiele aus dem JDK
- 2.5.4 Signatur einer Methode
- 2.5.5 Fehlerbehandlung und Exceptions
- 2.6 Kommentare
- 2.7 Schleifen
 - 2.7.1 Die `for`-Schleife
 - 2.7.2 Die `for-each`-Schleife
 - 2.7.3 Die `while`-Schleife
 - 2.7.4 Die `do-while`-Schleife
- 2.8 Rekapitulation
- 2.9 Weiterführende Dokumentation für nächste Schritte
- 2.10 Aufgaben und Lösungen
 - 2.10.1 Aufgabe 1: Mathematische Berechnungen
 - 2.10.2 Aufgabe 2: Methode und `if`
 - 2.10.3 Aufgabe 3: Selbstabholerrabatt

- 2.10.4 Aufgabe 4: Schleifen mit Berechnungen
- 2.10.5 Aufgabe 5: Schleifen und fixe Schrittweite
- 2.10.6 Aufgabe 6: Schleifen mit variabler Schrittweite
- 2.10.7 Aufgabe 7: Verschachtelte Schleifen – Variante 1
- 2.10.8 Aufgabe 8: Verschachtelte Schleifen – Variante 2
- 2.10.9 Aufgabe 9: Verschachtelte Schleifen – Variante 3

3 Strings

3.1 Schnelleinstieg

- 3.1.1 Gebräuchliche Stringaktionen
- 3.1.2 Suchen und Ersetzen
- 3.1.3 Informationen extrahieren und formatieren

3.2 Nächste Schritte

- 3.2.1 Die Klasse Scanner

3.2.2 Mehrzeilige Strings (Text Blocks)

3.2.3 Strings und `char []`s

3.3 Praxisbeispiel: Text in Title Case wandeln

3.4 Aufgaben und Lösungen

3.4.1 Aufgabe 1: Länge, Zeichen und Enthaltensein

3.4.2 Aufgabe 2: Title Case mit Scanner

3.4.3 Aufgabe 3: Zeichen wiederholen

3.4.4 Aufgabe 4: Vokale raten

3.4.5 Aufgabe 5: String Merge

4 Arrays

4.1 Schnelleinstieg

4.1.1 Gebräuchliche Aktionen

4.1.2 Mehrdimensionale Arrays

4.2 Nächste Schritte

4.2.1 Eindimensionale Arrays

4.2.2 Mehrdimensionale Arrays

4.3 Praxisbeispiel: Flächen füllen

4.4 Aufgaben und Lösungen

4.4.1 Aufgabe 1:

Durcheinanderwürfeln eines Arrays

4.4.2 Aufgabe 2: Arrays kombinieren

4.4.3 Aufgabe 3: Rotation um eine oder mehrere Positionen

4.4.4 Aufgabe 4: Zweidimensionales String-Array ausgeben

4.4.5 Aufgabe 5: Dreieckiges Array: Upside Down

5 Klassen und Objektorientierung

5.1 Schnelleinstieg

5.1.1 Grundlagen zu Klassen und Objekten

5.1.2 Eigenschaften (Attribute)

5.1.3 Verhalten (Methoden)

5.1.4 Objekte vergleichen – die Rolle

von `equals()`

5.2 Nächste Schritte

5.2.1 Klassen ausführbar machen

5.2.2 Imports und Packages

5.2.3 Übergang zum Einsatz einer IDE

5.2.4 Imports und Packages:
Auswirkungen auf unsere Applikation

5.2.5 Verstecken von Informationen

5.3 Vererbung

5.3.1 Basisklassen und abstrakte Basisklassen

5.3.2 Overloading und Overriding

5.4 Die Klasse `Object`

5.4.1 Beispielklasse `Person`

5.4.2 Die Methode `toString()`

5.4.3 Ergänzungen zur Methode `equals(Object)`

5.4.4 Typprüfung mit `instanceof`

5.4.5 Pattern Matching bei