



## Teil II: Konzepte imperativer Sprachen

Imperative Konzepte und Techniken

Eine Einführung am Beispiel von Java

## II. Konzepte imperativer Sprachen

1. Vorbemerkungen
2. Compilation, Interpretation, virtuelle Rechner
3. Grundlegende Sprachkonstruktionen imperativer Programme
4. Standardbibliotheken: Java-API
5. Auswahl: bedingte Anweisungen
6. Iteration (Zyklen, Schleifen)
7. Methoden (Algorithmen, Funktionen, Prozeduren): Hauptstrukturierungsmerkmal imperativer Programme
8. Ausdrücke, Operatoren, (einfache) Typen
9. Programmierrichtlinien
10. Felder (Arrays)
11. Rekursion, Komplexität von Algorithmen
12. Such- und Sortierverfahren mit Arrays

## 1. Vorbemerkungen

## Schwerpunkte

- Literatur
- Ansatz der Vorlesung
- Kurzcharakteristik zu Java

# Universtitätsbiliothek der HU

## Lehrbuchsammlung (viele Exemplare)

- R. Schiedermeier: Programmieren mit Java II, Pearson, München, 2013
- Sedgewick, Robert: Einführung in die Programmierung mit Java- München, Pearson, 2011.
- u.v.a.

## Freihandbestand (1 – 2 Exemplare)

- 150 unterschiedliche Titel zu Java aus den Jahren 2011-2015

## E-Books

- Goll, Joachim ; Heinisch, Cornelia: Java als erste Programmiersprache: Ein professioneller Einstieg in die Objektorientierung mit Java, Springer Fachmedien Wiesbaden: Wiesbaden ; 2014, <http://dx.doi.org/10.1007/978-3-8348-2270-3>
- Dietmar Ratz ; Jens Scheffler; Detlef Seese: Grundkurs Programmieren in Java, München Carl Hanser Verlag GmbH & Co. KG ; 2014, <http://dx.doi.org/10.3139/9783446441101>
- u.v.a.

# Lehrbücher (1)

- S. Kamin, D. Mickunas, E. Reingold: An Introduction to Computer Science Using Java, 2nd Ed., McGraw Hill, 2002  
*sehr gute Einführung in Konzepte der Programmierung*
- R. Sedgewick, K. Wayne: Introduction to Programming in Java, Addison-Wesley, 2007  
*für Anfänger sehr gut geeignet*
- R. Sedgewick, K. Wayne: Einführung in die Programmierung mit Java. 1. Auflage. Pearson, München, 2011  
*deutsche Übersetzung*
- D. Ratz, J.Scheffler: Grundkurs Programmieren in Java. 7. aktualisierte und erweiterte Auflage. Hanser Verlag, München, Wien, 2014  
*für Anfänger sehr gut geeignet*

# Lehrbücher (2)

- G. Krüger: Handbuch der Java-Programmierung , Addison-Wesley, 7. Auflage, 2011  
*sehr gutes Nachschlagewerk zu Java; ausführliche API-Dokumentation; nicht für Anfänger*
- P. Pepper: Programmieren mit Java, Springer-Verlag, 2007  
*für Anfänger sehr gut geeignet*
- R. Schiedermeier: Programmieren mit Java II, Pearson, München, 2013  
*für Anfänger sehr gut geeignet*

# Vorkenntnisse

**Vorkenntnisse:** Unterschiede sehr groß

- Uni Hamburg (Web-Seite):  
*"Alles Bekannte verlernen:*
  - Dies ist kein Leistungskurs Informatik
  - Dies ist kein Programmierkurs 'Basic für Fortgeschrittene'"
- Wendt (Direktor, Hasso-Plattner-Institut, Uni Potsdam / SAP-Stiftung):  
*"Am liebsten sind mir eigentlich die, die noch nicht programmiert haben. In der Elektrotechnik stellen die Funkamateure auch ein besonderes Problem dar. Die kommen an die Uni und glauben, sie wüssten schon alles ... Programmieren als Heimwerker hat mit Software Engineering nichts zu tun"* (Young Professional 2 / 99).

## Unsere Konzeption

- HU:
  - Schwerpunkt:  
Konzepte von Programmiersprachen,  
Techniken der Softwareentwicklung  
(am Beispiel von Java)
  - Nicht jedes Detail von Java in Vorlesung:  
nicht sinnvoll (umfangreiche API), langweilig  
→ trotzdem: in Vorlesung Java lernen

**Lehrbuch zu Java** besorgen

→ **Ausbildung nicht kurzfristig auslegen,  
sondern für 40 Jahre Berufspraxis**

## Methodik

Programmieren lernen durch

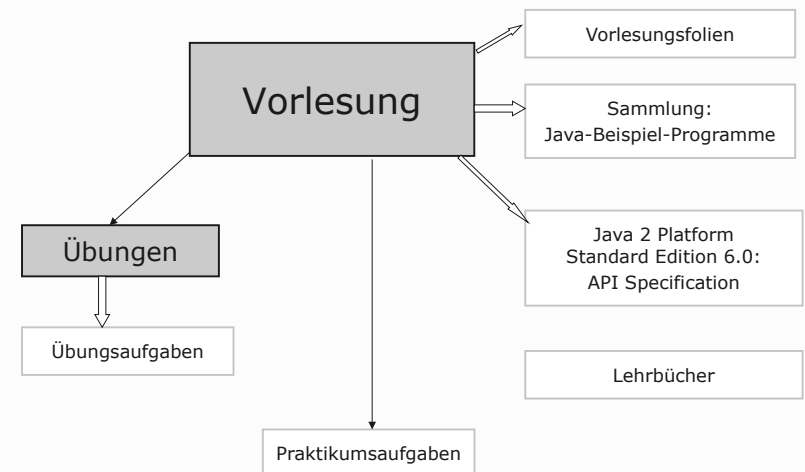
- Studieren "guter" Programme ("learning by reading")  
→ siehe: unsere Java-Beispielsammlung auf der GdP-Webseite
- Selbst programmieren ("learning by doing")

Java-Beispielsammlung  
(siehe GdP-Webseite):  
zu den Vorlesungen mitbringen!

Zwei Varianten:

- Gut lesbare pdf-Version  
(zur Vorlesung: ausgedruckt)
- Programm-Version: abarbeitbar, ausgetestet

## Komponenten der Veranstaltung GdP



## Fragebögen: Kommentare von Studierenden WS 2013/14

Mit Vorkenntnissen	Ohne Vorkenntnisse
<ul style="list-style-type: none"><li>• Vorlesung müsste schneller vorankommen;</li><li>• Bisherige Kenntnisse konnten systematisiert werden;</li><li>• Ich muss nicht zur VL, wenn ich die Thematik beherrsche! Ausgezeichnete Folien.</li></ul>	<ul style="list-style-type: none"><li>• VL-Tempo viel zu hoch;</li><li>• ohne Vorkenntnisse schwer zu folgen;</li><li>• Bequemes Tempo und Schwierigkeitsgrad, man kann den Stoff in der VL auch ohne Vorkenntnisse verstehen</li></ul>

## Java: ein Beispiel für eine Programmiersprache

- Java (1995) (Firma: Sun, jetzt Oracle)
- Eigenschaften von Java:
  - einfach
  - objektorientiert
  - portabel (rechnerunabhängig)
  - Internet-/WWW-tauglich
  - ...
- trotz allem:
  - als 'Erstsprache' (Einstiegssprache) nicht unkompliziert [OO, Bibliotheken]