

Programmieren in Java

Einführung

Prof. Dr. Peter Thiemann

Albert-Ludwigs-Universität Freiburg, Germany

SS 2017

Koordinaten

▶ **Vorlesung:** Mo, 16st-18 Uhr, HS 101-00-026

▶ **Dozent:** Prof. Dr. Peter Thiemann

Gebäude 079, Raum 00-015

Telefon: 0761 203 -8051/-8247

E-mail: thiemann@cs.uni-freiburg.de

Web: <http://www.informatik.uni-freiburg.de/~thiemann>

▶ **Homepage der Vorlesung:**

<http://proglang.informatik.uni-freiburg.de/teaching/java/2017>

Kommunikation

- ▶ Besprechung von Lösungen, Folienvortrag, Aufzeichnung

Kommunikation

- ▶ Besprechung von Lösungen, Folienvortrag, Aufzeichnung
- ▶ Live-Hacking (für Lösungen und on demand)

Kommunikation

- ▶ Besprechung von Lösungen, Folienvortrag, Aufzeichnung
- ▶ Live-Hacking (für Lösungen und on demand)
- ▶ Forum
(<https://daphne.informatik.uni-freiburg.de/forum/viewforum.php?f=689>)

Kommunikation

- ▶ Besprechung von Lösungen, Folienvortrag, Aufzeichnung
- ▶ Live-Hacking (für Lösungen und on demand)
- ▶ Forum
(<https://daphne.informatik.uni-freiburg.de/forum/viewforum.php?f=689>)
- ▶ Subversion (Einreichen von Lösungen)

Ziele aus dem Modulhandbuch

Die Studierenden erlernen objekt-orientierte Programmierkonzepte anhand der Sprache Java. Sie können eigenständig Programme unter Verwendung einer graphischen Entwicklungsumgebung für Java erstellen. Sie sind in der Lage, Java-Bibliotheken in eigenen Programmen einzusetzen.

Inhalt

- ▶ Objekt-orientierte Programmierkonzepte
- ▶ Benutzung der Java Entwicklungswerkzeuge (IntelliJ)
- ▶ Erstellung von Java Programmen
- ▶ Benutzung von Java Bibliotheken
- ▶ Ressourcen zu den Themen Objekt-Orientierung, Java, Software-Entwicklung
- ▶ Eigenständige Recherche

... und was dahinter steckt

- ▶ Programmieren erfordert
 - ▶ Beherrschung einer Programmiersprache
 - ▶ Verständnis des zu lösenden Problems
 - ▶ Kreativität
 - ▶ Arbeitseinsatz
 - ▶ ... und zu Beginn: **Übung**

... und was dahinter steckt

- ▶ Programmieren erfordert
 - ▶ Beherrschung einer Programmiersprache
 - ▶ Verständnis des zu lösenden Problems
 - ▶ Kreativität
 - ▶ Arbeitseinsatz
 - ▶ ... und zu Beginn: **Übung**

- ▶ Ziel der Vorlesung:

Einfache Java-Programme erstellen und
Standardsituationen der Programmierung beherrschen.

Literatur (Design und Programmierung)

- ▶ Felleisen, Findler, Flatt, Gray, Krishnamurthi, Proulx.
How to Design Classes. 2012.
<http://www.ccs.neu.edu/home/matthias/HtDC/htdc.pdf>
- ▶ Matthias Felleisen, Dan Friedman.
A Little Java, A Few Patterns. MIT Press, 1997.
- ▶ Allen Holub.
Holub on Patterns: Learning Design Patterns by Looking at Code.
Apress, 2004.
- ▶ Steve McConnell.
Code Complete: A Practical Handbook of Software Construction, Second Edition. Microsoft Press, 2004.
- ▶ Arnd Poetzsch-Heffter.
Konzepte objektorientierter Programmierung. Mit einer Einführung in Java. 2. Auflage. Springer, 2009.

Literatur (Java)

- ▶ Peter Sestoft.
Java Precisely, Second Edition. The MIT Press, August 2005. ISBN 0-262-69325-9.
- ▶ James Gosling, Bill Joy, Guy Steele, Gilad Bracha, Alex Buckley.
The Java Language Specification, Java SE 8 Edition .
Addison-Wesley Professional, 2014.
- ▶ Joshua Bloch.
Effective Java. Prentice Hall, 2001.
- ▶ Martin Fowler.
UML Distilled, Third Edition. Addison-Wesley, 2003.
- ▶ Kirill Fakhroutdinov.
The Unified Modeling Language. <http://www.uml-diagrams.org/>

Übungen

- ▶ in Form von **Programmieraufgaben**
- ▶ bilden die **Prüfungsleistung**
- ▶ **Betreuung:** Luminous Fennell
Gebäude 079, Raum 00-013
Telefon: 0761 203 -8053
E-mail: fennell@informatik.uni-freiburg.de
- ▶ **Tutoren/Korrektoren:** Fabian Wenzelmann, Frank Schüssele Karsten Fix, Megha Elias , Ravisha Gaur, Tim Aicher, Tobias Strickfaden,
- ▶ Wöchentliche **Q&A Session:**
Donnerstag, 18st-19
Gebäude 101, Raum Seminar 00-010/014

Moderiert von Tutoren. Hier können Fragen zu den aktuellen
Übungen besprochen werden.
on-demand

Übungsmodus (1)

`http://proglang.informatik.uni-freiburg.de/teaching/java/2017/`

In der **Vorlesung** (z.B. v02):

- ▶ Besprechung der abgegebenen Übung (w01)
- ▶ Einführung in nächste Übung (w02)
- ▶ (w03 kann schon auf eigene Faust bearbeitet werden)

Übungsmodus (2)

- ▶ **Kooperation:** Ideen austauschen: ✓, diskutieren: ✓, Code austauschen: ✗, Code aus dem Internet kopieren: ✗.
- ▶ **Jede Abgabe muss ausschließlich selbstgeschriebenen Code enthalten.** Wir führen eine stichprobenartige Überprüfung nach Plagiaten durch. Alle Beteiligten an einem entdeckten Plagiatsversuch werden von der weiteren Teilnahme ausgeschlossen und der Kurs wird als nicht bestanden gewertet.

Fragen?



Abgabe der Übungen über Daphne

(test2 durch eigenen Benutzernamen ersetzen, w01 durch aktuelles Paket)

- ▶ <https://daphne.informatik.uni-freiburg.de/ss2017/ProgrammierenJava/>

Abgabe der Übungen über Daphne

(test2 durch eigenen Benutzernamen ersetzen, w01 durch aktuelles Paket)

- ▶ `https://daphne.informatik.uni-freiburg.de/ss2017/ProgrammierenJava/`
- ▶ `> svn checkout https://daphne.informatik.uni-freiburg.de/ss2017/ProgrammierenJava/svn/test2`

Abgabe der Übungen über Daphne

(test2 durch eigenen Benutzernamen ersetzen, w01 durch aktuelles Paket)

- ▶ `https://daphne.informatik.uni-freiburg.de/ss2017/ProgrammierenJava/`
- ▶ `> svn checkout https://daphne.informatik.uni-freiburg.de/ss2017/ProgrammierenJava/svn/test2`
- ▶ Übungs-zip herunterladen und in test2 entpacken

Abgabe der Übungen über Daphne

(test2 durch eigenen Benutzernamen ersetzen, w01 durch aktuelles Paket)

- ▶ `https://daphne.informatik.uni-freiburg.de/ss2017/ProgrammierenJava/`
- ▶ `> svn checkout https://daphne.informatik.uni-freiburg.de/ss2017/ProgrammierenJava/svn/test2`
- ▶ Übungs-zip herunterladen und in test2 entpacken
- ▶ `> svn add w01`

Abgabe der Übungen über Daphne

(test2 durch eigenen Benutzernamen ersetzen, w01 durch aktuelles Paket)

- ▶ `https://daphne.informatik.uni-freiburg.de/ss2017/ProgrammierenJava/`
- ▶ `> svn checkout https://daphne.informatik.uni-freiburg.de/ss2017/ProgrammierenJava/svn/test2`
- ▶ Übungs-zip herunterladen und in test2 entpacken
- ▶ `> svn add w01`
- ▶ `> svn ci -m 'initial commit of w01'`

Abgabe der Übungen über Daphne

(test2 durch eigenen Benutzernamen ersetzen, w01 durch aktuelles Paket)

- ▶ `https://daphne.informatik.uni-freiburg.de/ss2017/ProgrammierenJava/`
- ▶ `> svn checkout https://daphne.informatik.uni-freiburg.de/ss2017/ProgrammierenJava/svn/test2`
- ▶ Übungs-zip herunterladen und in test2 entpacken
- ▶ `> svn add w01`
- ▶ `> svn ci -m 'initial commit of w01'`
- ▶ Aufgabe bearbeiten, (z.B. `whats-your-name`)

Abgabe der Übungen über Daphne

(test2 durch eigenen Benutzernamen ersetzen, w01 durch aktuelles Paket)

- ▶ `https://daphne.informatik.uni-freiburg.de/ss2017/ProgrammierenJava/`
- ▶ `> svn checkout https://daphne.informatik.uni-freiburg.de/ss2017/ProgrammierenJava/svn/test2`
- ▶ Übungs-zip herunterladen und in test2 entpacken
- ▶ `> svn add w01`
- ▶ `> svn ci -m 'initial commit of w01'`
- ▶ Aufgabe bearbeiten, (z.B. `whats-your-name`)
- ▶ `> svn add --force w01/whats-your-name/src`

Abgabe der Übungen über Daphne

(test2 durch eigenen Benutzernamen ersetzen, w01 durch aktuelles Paket)

- ▶ `https://daphne.informatik.uni-freiburg.de/ss2017/ProgrammierenJava/`
- ▶ `> svn checkout https://daphne.informatik.uni-freiburg.de/ss2017/ProgrammierenJava/svn/test2`
- ▶ Übungs-zip herunterladen und in test2 entpacken
- ▶ `> svn add w01`
- ▶ `> svn ci -m 'initial commit of w01'`
- ▶ Aufgabe bearbeiten, (z.B. `whats-your-name`)
- ▶ `> svn add --force w01/whats-your-name/src`
- ▶ `> svn ci -m 'solution for w01/whats-your-name'`

Abgabe der Übungen über Daphne

(test2 durch eigenen Benutzernamen ersetzen, w01 durch aktuelles Paket)

- ▶ `https://daphne.informatik.uni-freiburg.de/ss2017/ProgrammierenJava/`
- ▶ `> svn checkout https://daphne.informatik.uni-freiburg.de/ss2017/ProgrammierenJava/svn/test2`
- ▶ Übungs-zip herunterladen und in test2 entpacken
- ▶ `> svn add w01`
- ▶ `> svn ci -m 'initial commit of w01'`
- ▶ Aufgabe bearbeiten, (z.B. whats-your-name)
- ▶ `> svn add --force w01/whats-your-name/src`
- ▶ `> svn ci -m 'solution for w01/whats-your-name'`
- ▶ Resultat auf Jenkins prüfen

Abgabe der Übungen über Daphne

(test2 durch eigenen Benutzernamen ersetzen, w01 durch aktuelles Paket)

- ▶ `https://daphne.informatik.uni-freiburg.de/ss2017/ProgrammierenJava/`
- ▶ `> svn checkout https://daphne.informatik.uni-freiburg.de/ss2017/ProgrammierenJava/svn/test2`
- ▶ Übungs-zip herunterladen und in test2 entpacken
- ▶ `> svn add w01`
- ▶ `> svn ci -m 'initial commit of w01'`
- ▶ Aufgabe bearbeiten, (z.B. whats-your-name)
- ▶ `> svn add --force w01/whats-your-name/src`
- ▶ `> svn ci -m 'solution for w01/whats-your-name'`
- ▶ Resultat auf Jenkins prüfen

Abgabe der Übungen über Daphne

(test2 durch eigenen Benutzernamen ersetzen, w01 durch aktuelles Paket)

- ▶ `https://daphne.informatik.uni-freiburg.de/ss2017/ProgrammierenJava/`
- ▶ `> svn checkout https://daphne.informatik.uni-freiburg.de/ss2017/ProgrammierenJava/svn/test2`
- ▶ Übungs-zip herunterladen und in test2 entpacken
- ▶ `> svn add w01`
- ▶ `> svn ci -m 'initial commit of w01'`
- ▶ Aufgabe bearbeiten, (z.B. `whats-your-name`)
- ▶ `> svn add --force w01/whats-your-name/src`
- ▶ `> svn ci -m 'solution for w01/whats-your-name'`
- ▶ Resultat auf Jenkins prüfen

Falls Jenkins mit Finished: FAILURE endet, bitte im Forum melden
(das ist ein Fehler im System)

Get Started!

1. Bei Daphne für “ProgrammierenJava” anmelden
<https://daphne.informatik.uni-freiburg.de/ss2017/ProgrammierenJava/>
2. Übungen **w01** und **w02** herunterladen
3. Bei Problemen in die Q&A Session gehen oder ins Forum
4. Ende der Woche wird Ihnen ein Tutor zugeteilt

Fragen?

