

Programmieren in Java

Einführung

Peter Thiemann

Albert-Ludwigs-Universität Freiburg, Germany

SS 2013

Organisatorisches

- ▶ **Vorlesung** Di, 14-16 Uhr ct, HS 101-00-026
- ▶ **Dozent** Prof. Dr. Peter Thiemann

Gebäude 079, Raum 00-015

Telefon: 0761 203 -8051/-8247

E-mail: `thiemann at cs uni-freiburg de`

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

- ▶ **Informationen** Homepage der Vorlesung über

<http://proglang.informatik.uni-freiburg.de/teaching/java/2013>

Kommunikation

- ▶ Aufzeichnung

Kommunikation

- ▶ Aufzeichnung
- ▶ Folienvortrag

Kommunikation

- ▶ Aufzeichnung
- ▶ Folienvortrag
- ▶ Live-Demo

Kommunikation

- ▶ Aufzeichnung
- ▶ Folienvortrag
- ▶ Live-Demo
- ▶ Interaktion: Twitter #ufrjava @ProglangUniFr

Kommunikation

- ▶ Aufzeichnung
- ▶ Folienvortrag
- ▶ Live-Demo
- ▶ Interaktion: Twitter #ufrjava @ProglangUniFr
- ▶ Kollaborative Entwicklung: Codeaustausch via pastebin.com

Ziele

Die Studierenden erlernen weiterführende 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 (Eclipse, Ant, Checkstyle)
- ▶ Erstellung von Java Programmen
- ▶ Benutzung von Java Bibliotheken
- ▶ Ressourcen zu den Themen Objekt-Orientierung, Java, Software-Entwicklung
- ▶ (Kommunikationstechniken und Projektorganisation)
- ▶ Eigenständige Recherche

... und was dahinter steckt

- ▶ Programmentwurf ist schwierig und kann geraume Zeit in Anspruch nehmen.
- ▶ Programmentwurf erfordert Kreativität und tiefes Verständnis.

... und was dahinter steckt

- ▶ Programmentwurf ist schwierig und kann geraume Zeit in Anspruch nehmen.
- ▶ Programmentwurf erfordert Kreativität und tiefes Verständnis.
- ▶ Aber:

... und was dahinter steckt

- ▶ Programmentwurf ist schwierig und kann geraume Zeit in Anspruch nehmen.
- ▶ Programmentwurf erfordert Kreativität und tiefes Verständnis.
- ▶ Aber:
- ▶ Es gibt „Standardsituationen“, für die es bewährte Lösungen (sog. Patterns oder Best Practices) gibt.

... und was dahinter steckt

- ▶ Programmentwurf ist schwierig und kann geraume Zeit in Anspruch nehmen.
- ▶ Programmentwurf erfordert Kreativität und tiefes Verständnis.
- ▶ Aber:
- ▶ Es gibt „Standardsituationen“, für die es bewährte Lösungen (sog. Patterns oder Best Practices) gibt.
- ▶ Ziel der Vorlesung ist also:
 - ▶ Sie lernen die Standardsituationen + Lösungen kennen.
 - ▶ Sie verschwenden keine Zeit mit Standardsituationen, sondern können sich auf die Teile konzentrieren, wo Kreativität und Verständnis gefragt sind.

... und was dahinter steckt

- ▶ Programmentwurf ist schwierig und kann geraume Zeit in Anspruch nehmen.
- ▶ Programmentwurf erfordert Kreativität und tiefes Verständnis.
- ▶ Aber:
- ▶ Es gibt „Standardsituationen“, für die es bewährte Lösungen (sog. Patterns oder Best Practices) gibt.
- ▶ Ziel der Vorlesung ist also:
 - ▶ Sie lernen die Standardsituationen + Lösungen kennen.
 - ▶ Sie verschwenden keine Zeit mit Standardsituationen, sondern können sich auf die Teile konzentrieren, wo Kreativität und Verständnis gefragt sind.
- ▶ Chance der Übung
 - ▶ Trainieren der Standardsituationen

Literatur

- ▶ Felleisen, Findler, Flatt, Gray, Krishnamurthi, Proulx.
How to Design Classes. 2008.
<http://www.ccs.neu.edu/home/matthias/HtDC/htdc.pdf>
- ▶ Joshua Bloch.
Effective Java. Prentice Hall, 2001.
- ▶ Peter Sestoft.
Java Precisely, Second Edition. The MIT Press, August 2005.
ISBN 0-262-69325-9.
- ▶ Martin Fowler.
UML Distilled, Third Edition. Addison-Wesley, 2004.
- ▶ Matthias Felleisen, Dan Friedman.
A Little Java, A Few Patterns. MIT Press, 1998.
- ▶ Arnd Poetzsch-Heffter.
Konzepte objektorientierter Programmierung. Mit einer Einführung in Java. Springer, 2000.

Übungen

► **Poolstunden**

Mo 08-10 ct Pool 082-029

Di 16-18 ct Pool 082-029

Di 12-14 ct Pool 082-029

Mi 12-14 ct Pool 082-029

► **Betreuung** Luminous Fennell, Robert Jakob

Gebäude 079, Raum 00-013

Telefon: 0761 203 -8053/-8054

E-mail: {fennell, jakobro} at cs uni-freiburg de

► **Tutoren** Tobias Wieland, Markus Grützner, Stefan Dilger, Sebastian Dufner, Gabriel Kalweit, Nicolas Riesterer

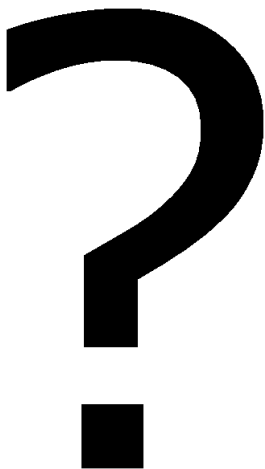
► Teilnahme freiwillig, die Tutoren helfen bei den Aufgaben

► Empfehlung: hingehen, falls ein Blatt mehr als 4-5 Stunden in Anspruch nimmt

Übungsmodus

- ▶ **Neues Aufgabenblatt:** Jeden Dienstag zur Vorlesung
- ▶ **Abgabe:** bis übernächsten Freitag 23:59h, im Übungssystem.
- ▶ Zum **Bestehen des Kurses** sind folgende Punkte zu erfüllen:
 - ▶ Mindestens 40% aller Punkte erreicht.
 - ▶ Für jedes 0 Punkte-Blatt erhöht sich die Bestehensgrenze um 5%. Beispiel: ein 0P-Blatt - mindestens 45%, zwei 0P-Blätter - mindestens 50%, usw.
- ▶ **Kooperation mit anderen Kursteilnehmern:** Ideen austauschen: ja gerne, diskutieren: ja gerne, Code austauschen: verboten, Code aus dem Internet kopieren: verboten.
- ▶ Wir führen eine stichprobenartige Überprüfung nach Plagiaten durch. Alle Beteiligten an einem entdeckten Plagiatsversuch werden von der weiteren Teilnahme ausgeschlossen und der Kurs wird mit nicht bestanden gewertet.

Fragen?



Eingangstest - 30 Min

- ▶ Anonymer Test über Programmierung und Java
- ▶ Kein Einfluß auf Note, Bestehen, etc
- ▶ Schnappschuß des Kenntnisstands
- ▶ Hilfe für uns bei der Themenauswahl
- ▶ Blätter nummeriert
- ▶ **Bitte Nummer merken für Ergebnis und Abschlusstest!**