

Praktikum "SEP: Java-Programmierung"

SS 2019

Organisatorisches

Karlheinz Friedberger und Thomas Bunk

- ▶ Anmeldung zur Veranstaltung im UniWorX
(erledigt via Zentralanmeldung)
- ▶ Wichtige Informationen auf Vorlesungs-Webseite
(im UniWorX verlinkt)
- ▶ Unbedingt regelmäßig Vorlesungs-Webseite besuchen!
- ▶ Wichtige Infos per Mail

- ▶ Vorlesung, Aufgabenstellung und Besprechung:
 - ▶ Dienstag 10:15 – 11:45 Uhr
 - ▶ Geschw.-Scholl-Pl. 1, B 011

- ▶ Übungstermine (nur während der Gruppenphase, ab Juni!):
 - ▶ Mo 16-18 Uhr c.t., Oettingenstr. 67, C 003
 - ▶ Mi 14-16 Uhr c.t., Oettingenstr. 67, 065
 - ▶ Mi 16-18 Uhr c.t., Oettingenstr. 67, 065
 - ▶ Do 12-14 Uhr c.t., Oettingenstr. 67, 065
 - ▶ Do 16-18 Uhr c.t., Oettingenstr. 67, C 003

- ▶ Personen
 - ▶ Karlheinz Friedberger
 - ▶ Thomas Bunk

Erste Aufgabe für euch:
Baseline für Antworten im Plenum

- ▶ gängige Algorithmen und Datenstrukturen
- ▶ Objekt-orientiertes Programmieren
(Vererbung, Information Hiding,...)
- ▶ Design-Pattern
- ▶ Nebenläufige Programmierung
- ▶ Testen und Debuggen

- ▶ Umsetzung in Java

Empfohlene Voraussetzungen I

- ▶ Fundierte Erfahrung mit Java
 - ▶ Udacity Course: Java Programming Basics
<https://tinyurl.com/ud4c1tyJava>
 - ▶ Buch: Christian Ullendbloom. Java ist auch eine Insel
<http://openbook.rheinwerk-verlag.de/javainsel/>

- ▶ Kenntnisse von VCS, bei uns: Git(Lab)
 - ▶ Simple words for a GitLab Newbie:
<https://tinyurl.com/g1t14bNewbie>
 - ▶ Udacity Course: Version Control with Git
<https://tinyurl.com/ud4c1tyGit>
 - ▶ Buch: Scott Chacon, Ben Straub. Pro Git
<https://tinyurl.com/pr0G1t>

Empfohlene Voraussetzungen II

- ▶ Nutzung der Kommandozeile
 - ▶ Udacity Course: Shell Workshop
<https://tinyurl.com/ud4c1tyShell>
 - ▶ Buch: William Shotts. The Linux Command Line
<https://tinyurl.com/l1nuxLC3>

- ▶ Design-Patterns und Code-Smells
 - ▶ Buch: Gang of Four. Design Patterns: Elements of Reusable Object-Oriented Software.
 - ▶ Website: z. B. Source Making
<https://sourcemaking.com/>

- ▶ Theoretische Grundlagen der Programmierung und von Algorithmen und Datenstrukturen
- ▶ Entsprechende Vorlesungen, siehe Veranstaltungswebseite für detaillierte Liste

- ▶ Jedes Mal: Aufgabenbesprechung
- ▶ Kurze Einführungen:
 - ▶ Codestyle
 - ▶ Module/Kapselung/Objektorientiertes Arbeiten in Java
 - ▶ Git(Lab)?
 - ▶ Evtl. Nutzung von Analyse-Tools/Gitlab CI
 - ▶ Wünsche?

- ▶ Anhand von 4 Programmieraufgaben
 - ▶ 3 Einzelabgaben
 - ▶ 1 Teamabgabe (5–7 Personen)
- ▶ Codebewertung nach folgenden Kriterien:
 - ▶ Funktionalität
 - ▶ Struktur
 - ▶ Layout (Kommentare u. Formatierung)
- ▶ Zusätzlich bei letzter Aufgabe: Bewertung der Teamarbeit
- ▶ jeweils **eine** Deadline, keine Nachholtermine

- ▶ Disclaimer ausfüllen!

- ▶ Aufgaben von uns
- ▶ Lösungen von euch
- ▶ Abgabe über den Praktomat
- ▶ Fragen und Besprechung in der „Vorlesung“

- ▶ <https://praktomat.sosy.ifi.lmu.de>
- ▶ Registrierung mit LMU-Email-Adresse (name@campus.lmu.de)
 - ▶ Anmeldung demnächst möglich
 - ▶ Deadline: 30.04. (nächster Dienstag)
- ▶ Aufgabenstellung und Abgabe im Praktomat
- ▶ Automatische Tests bei Abgabe:
 - Compiler, CheckStyle,
 - Plagiatstest, öffentliche und geheime Tests, ...
- ▶ Mehrfache Abgaben vor Deadline möglich, nur letzte Abgabe wird bewertet

	Aufgabe	Zeitraum	#Wochen
1.	Trie	22.04. – 14.05.	3
2.	TBD	14.05. – 04.06.	3
3.	TBD	04.06. – 25.06.	3
4.	TBD	25.06. – 30.07.	5

- ▶ Aufgabenstellungen werden zum jeweiligen Starttermin veröffentlicht.
- ▶ Abgaben vor jeweiliger Deadline sind möglich!
- ▶ Aufgaben 1–3 in Einzelarbeit.
- ▶ Aufgabe 4 in Teamarbeit.