


KLASSENMODELLIERUNG 1: GRUNDLAGEN

Eigene Klassen mit Variablen und Methoden modellieren, erstellen und verwenden

Struktur und Inhalt des Kurses wurden 2012 von Markus Heckner entwickelt. Im Anschluss haben Alexander Bazo und Christian Wolff Änderungen am Material vorgenommen. Die aktuellen Folien wurden von Alexander Bazo erstellt und können unter der **MIT-Lizenz**  verwendet werden.

AKTUELLER SEMESTERFORTSCHRITT (WOCHE 6)

Kursabschnitt	Themen		
Grundlagen	Einführung	Einfache Programme erstellen	Variablen, Klassen & Objekte
	Kontrollstrukturen & Methoden	Arrays & komplexe Schleife	
Klassenmodellierung	Grundlagen der Klassenmodellierung	Vererbung & Sichtbarkeit	
Interaktive Anwendungen	Event-basierten Programmierung	String- & Textverarbeitung	
Datenstrukturen	Listen, Maps & die Collections	Speicherverwaltung	Umgang mit Dateien
Software Engineering	Planhaftes Vorgehen bei der Softwareentwicklung		
	Qualitätsaspekte von Quellcode		

PINGO-QUIZ



<https://pingo.coactum.de/368355> 

DAS PROGRAMM FÜR HEUTE

- 5-Minuten-Thema: Texte in Variablen speichern (*Strings*)
- Das Prinzip des *Information Hiding*s
- Klassen modellieren, erstellen, verwenden und dokumentieren
- Objekte als Parameter verwenden

TEXT IN VARIABLEN SPEICHERN: STRINGS

EIGENE KLASSEN ENTWERFEN UND VERWENDEN




KLASSEN MIT JAVADOC KOMMENTIEREN UND DOKUMENTIEREN

OBJEKTE ALS PARAMETER

ZUSAMMENFASSUNG

- Strings sind Zeichenketten und können mit anderen Variablen konkateniert (= zusammengefügt, lat. catena = Kette) werden.
- Aus Sicht des Client-Programmierers ist es unerheblich, wie eine Klasse oder Methode implementiert wird.
- Klassen beinhalten typischerweise die folgenden Komponenten: Instanzvariablen, Konstanten, Konstruktoren, Methoden.
- Methoden und Variablen sind als private zu definieren, außer man wünscht explizit Zugriff von außerhalb des Objekts.
- Eigene Klassen sollten gut dokumentiert werden. JavaDoc ermöglicht automatisierte Erstellung von html-Dokumentation eigener Klassen.
- Objekte verhalten sich anders als primitive Datentypen, wenn Sie als Parameter von Methoden verwendet werden.

VIELEN DANK FÜR IHRE AUFMERKSAMKEIT. WENN SIE MÖCHTEN, SEHEN WIR UNS IM ANSCHLUSS IN DER ZENTRALÜBUNG!

Fragen oder Probleme? In allgemeinen Angelegenheiten und bei Fragen zur Vorlesung wenden Sie sich bitte an Alexander Bazo (alexander.bazo@ur.de ). Bei organisatorischen Fragen zu den Studienleistungen und Übungsgruppen schreiben Sie bitte Florin Schwappach (florin.schwappach@ur.de ). Bei inhaltlichen Fragen zu den Übungen, Beispielen und Studienleistungen schreiben Sie uns unter mi.oop@mailman.uni-regensburg.de .

QUELLEN

Eric S. Roberts, *The art and science of Java: an introduction to computer science, New international Edition*, 1. Ausgabe, Pearson, Harlow, UK, 2014.