Dr. Johannes Mayer
René Just



# Design und Architektur von Softwaresystemen (SS 2009)

## 1 Reflection und Annotations

Schreiben Sie ein kleines Programm zur Batchverarbeitung von Methoden. Alle auszuführenden Methoden sollen als Vereinfachnung statisch sein und in der selben Klasse existieren. Der Name dieser Klasse wird als Parameter an Ihr Programm übergeben. Da in der entsprechenden Klasse auch andere Methoden existieren können, sollen die auszuführenden Methoden annotiert werden. Implementieren Sie dazu eine eigene Annotation, die eine ganzzahlige ID und eine Beschreibung enthält.

Ihr Programm soll dann via Reflection die Klasse dynamisch laden und alle Methoden identifizieren, die annotiert sind. Anhand der ID sollen die Methoden sortiert (doppelte IDs sind erlaubt) und in einem Container referenziert (zwischengespeichert) werden. Zum Schluss soll der Container mit einer for-each-Schleife abgearbeitet und die entsprechende Methode zur Ausführung gebracht werden. Schreiben Sie dazu unmittelbar vor der Ausführung die ID und die zugehörige Beschreibung auf die Konsole. Der Inhalt der Methoden kann sich auf eine einfache Konsolenausgabe beschränken.

**Tipp:** Die notwendigen Klassen, Methoden und Typen für die Verwendung von Reflection und Annotations finden Sie in *java.lang.reflect* bzw. *java.lang.annotation*
(http://java.sun.com/javase/6/docs/api/)

## 2 UML Klassendiagramm

Entwerfen Sie als Erweiterung zu Blatt 1 ein UML Klassendiagramm mit hohem Detaillierungsgrad. Berücksichtigen Sie dabei insbesondere:

- Sichtbarkeit
- Assoziationen
- Multiplizitäten
- Navigierbarkeit

Stellen Sie zusätzlich die Kopplungen zwischen Ihren Klassen und Schnittstellen (Schnittstellen-Implementierung) dar.

**Viel Erfolg!**