

ADAP CW#12

Projektdaten

Projekt name: 3D-Printing
Projekt repository: <https://github.com/ModischFabrications/wahlzeit>
This week's tag: adap-hw11 on master
Homework diff: <https://github.com/ModischFabrications/wahlzeit/compare/adap-hw10...adap-hw11>
CI: <https://travis-ci.org/ModischFabrications/wahlzeit>
Docker Hub Repo: <https://hub.docker.com/r/modischfabrications/wahlzeit>

Hausaufgaben

Implementation

Neue Klassen (falls vorhanden)

Print, PrintManager, PrintType (+ Tests + TestSuite)

Veränderte Klassen (falls vorhanden)

PrintPhoto

Erklärung

Anfangen habe ich diese Woche mit der Erzeugung der Stubs für die neuen Klassen. Alle neu erstellten Klassen wurden dabei im package “org.wahlzeit.model.printphoto” gebündelt. Dann habe ich Stück für Stück die Methoden implementiert, dabei habe ich mich vor allem auf das Klassendiagramm der Aufgabenstellung bezogen. Initial habe ich dabei wie in der Vorlesung beschrieben den Type zu einer Factory gemacht, im zweiten Überarbeiten mich dann aber dazu entschieden den Manager dafür zu nutzen.

Dabei habe ich erstmal falsch das Beispiel der Vorlesung nachimplementiert, habe mich dann aber für eine andere, einfachere Implementation entschieden. Ich habe auch das Anlegen von Subtypen mit Konstruktorparametern implementiert, um eine intuitiv nutzbare Vererbung nachzubilden.

Parallel dazu habe ich Tests geschrieben, die die neuen Funktionalitäten validieren.

Ich habe alle Klassen möglichst Immutable und “safe-by-design” gehalten, daher werden alle verpflichtenden Attribute im Konstruktor gesetzt.

Zuletzt habe ich alle Änderungen gepusht und den letzten Commit mit dem oben genannten Tag für die Abgabe markiert.

Fragen

How did you deal with type hierarchies?

Ich habe wie in der Vorlesung beschrieben eine flache Hierarchie implementiert, die die Inheritance erst zur Laufzeit modelliert. Dabei erlaube ich das Übergeben von Supertypen als Konstruktorparametern, um einen Inheritance-ähnlichen Syntax zu erlauben.

Die geforderte Methode “isSubtype(Of)” habe ich dabei umbenannt, um eine eindeutigere Bedeutung zu geben. In dieser Methode wird rekursiv der Vererbungsbaum durchfahren, um eine Antwort zu finden. Aktuell kann dabei eine Endlosschleife entstehen, die Lösung wären zB. Tests beim Setzen des Supertyps.