

Die Online-Befragung zur genutzten alternativen Veranstaltungsform und zur Lehrevaluation ist online. Bitte ausfüllen: <https://forms.gle/8HLbXfVaQLjibSeo5A>. Sie werden eventuell aufgefordert sich bei Google anzumelden, das ist nur notwendig, wenn Sie in der Bearbeitung eine Pause machen wollen und das Teilergebnis zwischenspeichern wollen. Die Befragung endet am 20.12., die Ergebnisse stehen in einem nachfolgenden Fragen&Antworten-Dokument auf der Webseite der Veranstaltung.

Hinweis: Diese Lernnotiz enthält einen sehr sinnvollen Vorschlag um den Lehrstoff der 10. Woche der Veranstaltung (am 19.12.24) zu erlernen. Er ist gegliedert in die generellen Ziele und die Arbeitsschritte. Es ist notwendig, dass Sie die in dieser Lernnotiz genannten Videos bis zum Ende der offiziellen Vorlesungszeit (Do 11:30) durchgearbeitet haben. Zur Vorlesungszeit besteht die Möglichkeit in Zoom Fragen zu stellen und weitergehende Themen zu diskutieren.

<https://hs-osnabrueck.zoom.us/my/kleuker>

Einzelne Termine können kurzfristig per E-Mail vereinbart werden.

Ziele

- Die Pattern Command und Visitor ohne Hilfsmittel erklären und anwenden können.
- GRASP-Pattern mit ihrer Motivation kennen und anwenden können.

Arbeitsschritte

- *Laden Sie sich die folgenden Videos zuerst herunter, wenn Sie die HS-Plattform nutzen und schauen Sie sich diese an. Es ist sinnvoll die Folien danach nochmals durchzugehen.*

Folien 313 – 328: Command

http://kleuker.iui.hs-osnabrueck.de/Videos/OOAD/OOADPattern1_7.mp4 (35:35),
auch <https://youtu.be/jjgmQUmZSi0>

Folien 329 – 333: Visitor

http://kleuker.iui.hs-osnabrueck.de/Videos/OOAD/OOADPattern1_8.mp4 (21:01),
auch <https://youtu.be/uQZGQBON6VE>

Folien 334 – 357: GRASP

http://kleuker.iui.hs-osnabrueck.de/Videos/OOAD/OOADPattern1_9.mp4 (37:24),
auch <https://youtu.be/8ifuDS7gLyw>

- Lesen Sie die Seiten 210-216, dann 183-185. Es gibt leider noch keinen Abschnitt zum Visitor-Pattern. Dokumentieren Sie offene Fragen und schicken Sie sie an den Dozenten.

Wenn gegen Pattern verstoßen wird, müssen gute Gründe vorliegen. Ein solcher guter Grund kann das weit in neueren Bibliotheken verbreitete Method Chaining oder auch Fluent Programming sein. Dabei liefern Command-Methoden, die nur das genutzte Objekte verändern, immer das aktuelle Objekt this als Ergebnis zurück. Dadurch sind Aufrufe möglich, die ähnlich wie `sammlungTueb.filterAlter(80).filterRaucher().filterAlleinstehend()` aussehen können. Diese sind intuitiv lesbar, da sie wie eine in der Programmiersprache spezifizierte Beschreibungssprache aussehen. Dies wird auch interne DSL (Domain Specific Language) genannt.

Die Folien 351-354 zeigen ein solches Beispiel, mit der in der `main()`-Methode markierten Verkettung, die auch durch die gewählte Einrückung noch lesbarer wird.

Der Punkt am Anfang der Zeile ist wichtig, da so sofort klar ist, dass hier kein eigenständiger Befehl steht, sondern die Zeile mit der vorherigen verknüpft gelesen werden muss.

Das Method Chaining wird auch bei dem Builder in den Folien 354-357 genutzt, der vom ersten Aufgabenblatt bekannt sein sollte. So ein Builder erlaubt es komplexe Objekte modular in einzelnen Methodenaufrufen zusammenzubauen. Wenn so ein Builder eingesetzt wird, ist das die einzige Klasse, die Objekte der ge"buildeten" erzeugt.

- Lesen Sie das zur Vorlesung gehörende Fragen-Und-Antworten-Dokument, das meist kurz nach der Vorlesung auf der Veranstaltungsseite in der Nähe dieser Lernnotiz steht.
- Bearbeiten Sie das Quiz unter http://kleuker.iui.hs-osnabrueck.de/quiz/oad10_89366.html und merken Sie sich die oben angegebenen Lösungsbuchstaben.
- Prüfen Sie, ob Sie die angegebenen Lernziele erreicht haben.