

Übung 5

Realisieren Sie die Funktionalität den Zustand der Geschäftslogik zu laden und zu speichern.

Implementieren Sie für die Übung eine main-Methode, die ein befülltes Lager erstellt, dieses speichert und anschließend lädt.

Diese (und nur diese) Abgabe soll als Gradle-Projekt erfolgen.

Änderungen an den Interfaces sind zulässig ab jetzt zulässig. Dabei muss der Funktionsumfang gleich bleiben.

Abgabeanforderungen

Die Abgabe hat als zip-Datei zu erfolgen, die ein lauffähiges IntelliJ-IDEA-Projekt enthält. Sie muss eine (in IntelliJ lesbare) readme-Datei im root des Projektes (neben der iml-Datei) enthalten in der der erreichte Stand (die Nummer) bezüglich des Bewertungsschemas vermerkt ist.

Quellen

Zulässige Quellen sind suchmaschinen-indizierte Internetseiten. Werden mehr als drei zusammenhängende Anweisungen übernommen ist die Quelle in den Kommentaren anzugeben. Ausgeschlossen sind Quellen, die auch als Beleg oder Übungsaufgabe abgegeben werden oder wurden. Zulässig sind außerdem die über moodle bereitgestellten Materialien, diese können für die Übungsaufgaben ohne Quellenangabe verwendet werden.

fakultativ

1. Schreiben in einen Stream mit mindestens 3 Filtern
2. Lesen aus einem Stream mit mindestens 3 Filtern

Bewertungsschema

1 Punkt:

1. zip Archiv
2. IntelliJ-Projekt (kein ~~Gradle~~, Maven o.ä.)
3. readme vorhanden
4. Java8 (language level)
5. kompilierbar
6. main-Methoden nur im default package
7. ausführbar
8. Speichern und Laden der Geschäftslogik

2 Punkte:

9. Trennung zwischen Test- und Produktiv-Code
10. JUnit5 als Testframework
11. Mockito als Mock/Spy-framework
12. keine leeren Test
13. Speichern und Laden der Geschäftslogik mit JOS und JBP
14. Stellvertreter-Tests für das Lesen und Schreiben aus/in Streams
15. Gradle-Projekt

2,5 Punkte:

16. wahlfreies Speichern und Laden von Mediadateien über die Adresse