

Data Access Object (DAO) und Transfer Object (TO) Pattern Aufgabe

Gegeben ist folgendes Datenmodell zur Realisierung eines Hochregallagers, in dem Artikel robotergesteuert automatisch ein- und ausgelagert werden:

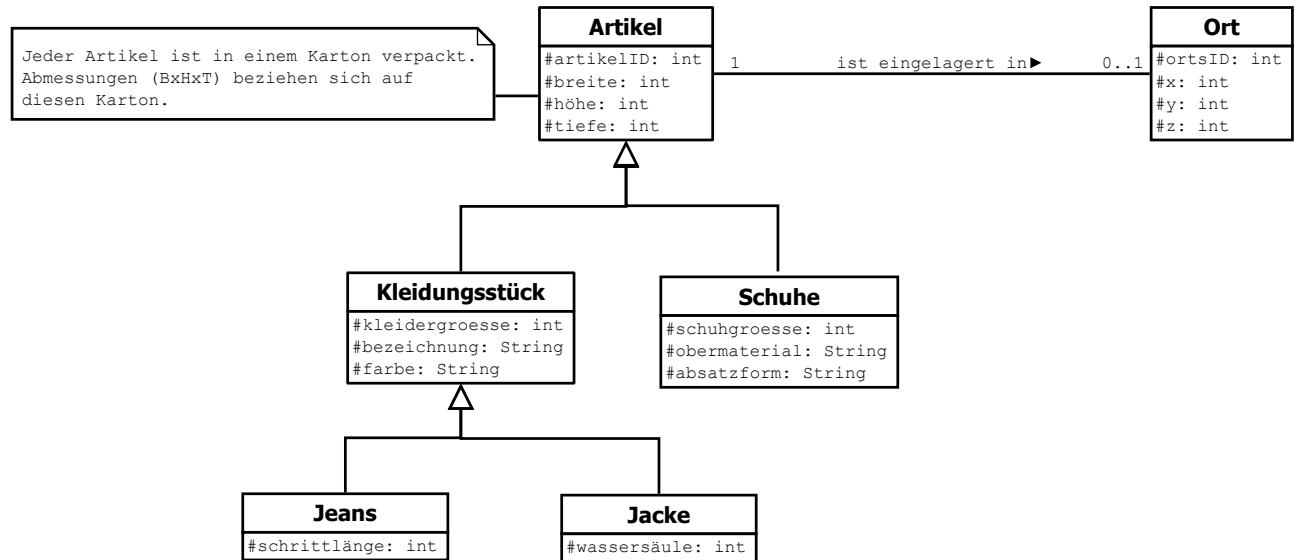


Abbildung 1: Artikel Datenmodell

1. a) Realisieren Sie unter Anwendung des **DAO und TO** Patterns eine persistente Datenhaltung für das Datenmodell in Abbildung 1: Artikel Datenmodell. (Realisieren Sie nicht mehr Patterns d.h. das AbstractFactory Pattern, das auch in [1] im Zusammenhang mit DAO verwendet wird, sollten Sie NICHT implementieren!) Für die Speicherung haben Sie verschiedene Möglichkeiten:

- Datenspeicherung in Datei über den Java Serialisierungsmechanismus **ODER**
- Datenspeicherung in Datei über ein eigenes lesbares Format (z.B. XML) **ODER**
- Datenspeicherung in einer relationalen Datenbank
- und weitere...

Entscheiden Sie sich für **EINE** mögliche Speicherung. Den größten Lerneffekt haben Sie, wenn Sie sich für eine relationale Datenbank entscheiden (je nach Vorkenntnissen bzw. dem aktuellen Stand in der Datenbankvorlesung!) Aufwandsmässig dürfte der Java Serialisierungsmechanismus am einfachsten sein.

Lösungshinweis: damit Objekte „persistent“ identifiziert werden können, benötigen Sie eindeutige Objekt-Ids. (Die Attribute `artikelID` und `ortsID` sind dafür gedacht!)

- b) Erklären Sie, wozu TransferObjects (TO) eingesetzt werden können und unter welchen Umständen diese Sinn machen.

2. In dieser Teilaufgabe realisieren Sie eine Klasse für die „Businessschicht“. In der Businessschicht wären beispielsweise Funktionalitäten, wie z.B. Ermittlung des kürzesten Weges von einem Ort zum nächsten für einen Roboter. Das realisieren wir aus Aufwandsgründen hier nicht. Stattdessen schreiben Sie eine Klasse `LagerSteuerung` mit einer Methode
 - `Ort ermittleEinenFreienOrt()`, die einen Ort ermittelt, an dem kein Artikel eingelagert ist.
3. Zusatzaufgabe: Machen Sie sich klar, zu welchem Sachverhalt zusätzlich zu DAO und TO der Einsatz des AbstractFactory Patterns Ihnen hilft.

Literatur

- [1] <http://corej2eepatterns.com/Patterns2ndEd/DataAccessObject.htm>
- [2] <http://corej2eepatterns.com/Patterns2ndEd/TransferObject.htm>
- [3] Fowler, M.; *Patterns of Enterprise Application Architecture* Addison Wesley 2003