
Aufgaben zur Klausur in *Objektorientierte Datenbanken (WS 2006/2007)*

Zeit: 60 Minuten,

erlaubte Hilfsmittel: keine

Bitte tragen Sie Ihre Antworten auf gesonderten Blättern ein (nicht hier!).

Vergessen Sie nicht, das Deckblatt zu unterschreiben!

Für die Klausur werden insgesamt 30 Bewertungseinheiten (BE) vergeben.

Zum Bestehen benötigen Sie mindestens 15 BE. Viel Erfolg !

1. Aufgabe (4 BE), Thema: ODMG

- a) Erklären Sie den Unterschied zwischen einem Attribut und einer Beziehung in ODMG! Geben Sie an, wie das jeweilige Konzept in Java realisiert wird! (3 BE)
- b) Welche der beiden eben erwähnten Konzepte wird in ODMG mit einer Modellierungsgenauigkeit versehen, die in Java nicht möglich ist? Geben Sie diese an! (2 BE)

2. Aufgabe (5 BE), Thema: JDO-Metadaten

In den JDO-Metadaten finden Sie folgende Einträge:

```
<class name = 'professor'>  
  <field name = 'office'  
    persistence-modifier= 'transactional'  
    embedded = 'true' />  
</class>
```

- a) Geben Sie eine Instanz mit Werten an, die zu diesen Metadaten passen! (1 BE)
- b) Erklären Sie das Konzept, das durch **persistence-modifier** definiert wird und geben Sie an, wie es sich in diesem Beispiel auswirkt! (2 BE)
- c) Erklären Sie das Konzept, das durch **embedded** definiert wird und geben Sie an, wie es sich in diesem Beispiel auswirkt! (2 BE)

3. Aufgabe (3 BE) Thema: Queries

- a) Welchen Vorteil bieten Queries gegenüber dem Extrahieren von Daten über Extents?(1 BE)
- b) Gibt es über die syntaktischen Unterschiede hinaus einen konzeptionellen Unterschied zwischen (Filter-)Queries / Criteria auf der einen Seite und Single-String-Queries auf der anderen Seite bzgl. des Compiler-Einsatzes?
Beantworten Sie die Frage getrennt für JDO und Hibernate! (2 BE)

4. Aufgabe (3 BE) Thema: JDO-Datenidentitätskonzepte

- a) Nennen Sie 3 Eigenschaften, die eine Klasse erfüllen muss, um als ID-Klasse verwendet werden zu können! (2 BE)
- b) Welchen Vorteil hat es, die ID-Klasse vom Anwender festlegen zu lassen und wie nennt man dieses Verfahren in JDO? (1 BE)

5. Aufgabe (4 BE), Thema: JDO-Lebenszykluszustände

- a) Aus welchem Grund wurden in JDO 2.0 die Zustände **detached-clean** und **detached-dirty** eingeführt? (2 BE)
- b) Beschreiben Sie die Unterschied für den Anwender zwischen folgenden Zuständen:
 - i) **hollow**
 - ii) **persistent-nontransactional** (2 BE)

6. Aufgabe (3 BE), Thema: Persistenzmanagement

Sie wollen ein Java-Programm an eine relationale Datenbank anbinden.

- a) Was müssen Sie außer einer Standard-Java Virtual Machine noch bereitstellen, damit die Anbindung funktioniert? (2 BE)
- b) Worin unterstützt Sie Hibernate? (1 BE)

7. Aufgabe (3 BE), Thema: Hibernate vs. JDO

- a) Nennen Sie 2 Vorteile, die das Prinzip der Transparenten Persistenz von Hibernate gegenüber JDO hat! (2 BE)
- b) Welchen Nachteil hat Hibernate gegenüber JDO beim Detached Object Support? (1 BE)

8. Aufgabe (5 BE), Thema: Hibernate-Queries

Die folgende HQL-Query bezieht sich auf das untenstehende Datenmodell.

```
from Customer customer
```

```
left join fetch customer.buys purchase
```

```
where customer.name like '%Müller' and purchase.price > 50
```

- a) Beschreiben Sie, was diese Query bewirkt: Welche Objekte werden aus der Datenbank geladen? Gehen Sie vor allem auf die spezielle Wirkung von **left join fetch** ein! (2 BE)
- b) Formulieren Sie die Frage so um, dass dieselben Kunden geladen werden, aber kein fetch gemacht wird! Welche Objekte werden jetzt nicht mehr oder jetzt zusätzlich geladen? (2 BE)
- c) Welches Konzept ist das Analogon zu **left join fetch** in JDOQL? Worin unterscheidet sich das Konzept von der HQL-Lösung? (Antwort wird nur in Stichpunkten erwartet, Codebeispiel nicht erforderlich) (1 BE)

