

Übung zur Vorlesung
Einsatz und Realisierung von Datenbanksystemen im SoSe15

Moritz Kaufmann (moritz.kaufmann@tum.de)
<http://db.in.tum.de/teaching/ss15/impldb/>

Blatt Nr. 05

Hausaufgabe 1

Wie unterscheidet sich das Konzept eines Objekt im Objekt-Orientierten Modell von dem einer Entität im Entity-Relationship Modell?

Eine Entität ist ein einfacher Datenwert oder auch eine Sammlung von Datenwerten. Ein Objekt hat auch zugeordnete Datenwerte (Zustand) aber zusätzlich noch Methoden (Verhalten) und eine feste Identität. Je nach Implementierung kann der Daten-zugriff und insbesondere auch -Veränderung nur über eine vordefinierte Menge von Methoden erfolgen.

Quelle: <http://codex.cs.yale.edu/avi/db-book/db6/practice-exer-dir/22s.pdf> Aufgabe A22.6

Hausaufgabe 2

Sie müssen als Berater ein Datenbanksystem für einen Kunden auswählen. Entscheiden sie für jeden der folgenden Anwendungsbereiche welche Art von Datenbanksystem am Besten passen würde (Relational, Objekt-Orientierte Datenbank, Objekt-Relationale Datenbanken). Begründen Sie ihre Entscheidung.

1. Eine CAD Anwendung für einen Flugzeugbauer
2. Eine Spendendatenbank um die Spenden an Bundestagsmitglieder zu verfolgen
3. Ein Anwendung zur Unterstützung des Filmmarketings
 1. CAD Anwendung: Ein OODB System scheint hier am geeignetsten. Ein CAD System arbeitet oft mit komplexen Datentypen und ist auch inhärent Objekt-Orientiert. Die hierarchischen Strukturen in Modellen lässt sich besonders gut durch Objekte abbilden.
 2. Spendendatenbank: Ein relationales Datenbanksystem müsste für diesen Fall ausreichend sein. Die erwarteten Datentypen sind einfach strukturiert. Der wichtigste Aspekt wird eine mächtige Anfragesprache sein.
 3. Filmmarketing: In diesem Szenario werden wahrscheinlich oft auch Multimedia oder ähnlich komplexe Datentypen verwendet werden. Der Aufbau der Datenbank wird aber relativ einfach sein und sollte gut in einer Objekt-Relationalen Datenbank umsetzbar sein.

Quelle: <http://codex.cs.yale.edu/avi/db-book/db6/practice-exer-dir/22s.pdf> Aufgabe A22.5

Hausaufgabe 3

Was ist der Unterschied zwischen *extent* und *extends* in ODL?

Mit *extends* wird Vererbung realisiert. Mit diesem Keyword wird angegeben dass die aktuelle Klasse eine Spezialisierung einer anderen Klasse ist. Als Unterklasse hat Sie auch Zugriff auf die Schnittstellen der Oberklasse.

Syntax: **class Assistent extends Angestellter ...**

Mit *extent* wird der Menge aller Objekte die zu dieser Klasse gehören ein Name gegeben. Der *extent* einer Klasse entspricht ungefähr einer Relation in einer Relationalen Datenbank. Diesem Vergleich folgend, wäre dann ein Objekt ein Tupel.

Syntax: **class Assistent (extent Assistenten) ...**

Hausaufgabe 4

Sie wollen ein Onlineverzeichnis für Bars erstellen. Für jede Bar soll das Menü inklusive Preise gespeichert werden. Erstellen Sie ein ODL Schema für die Klassen in Abbildung 1.

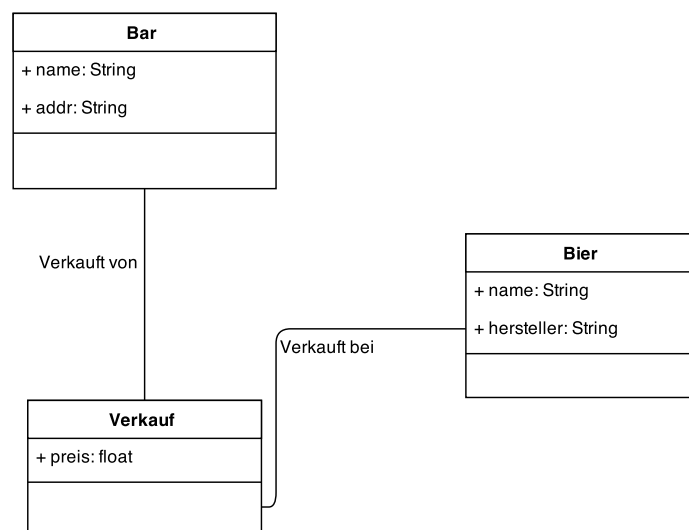


Abbildung 1: Schematisches Klassendiagramm für Bars.

```

class Bar (extent Bars) {
    attribute string name;
    attribute string addr;
    relationship Set(Verkauf) verkaufteBiere inverse Verkauf::bar;
}
class Bier (extent Biere) {
    attribute string name;
    attribute string hersteller;
    relationship Set(Verkauf) verkauftBei inverse Verkauf::bier;
}
class Verkauf (extent Verkaeufe) {
    attribute float preis;
    relationship Bar bar inverse Bar::verkaufteBiere;
    relationship Bier bier inverse Bier::verkauftBei;
}
  
```

Quelle: <http://www.uio.no/studier/emner/matnat/ifi/INF3100/v05/undervisningsmateriale/lysark/OQL.pdf> Seite 9

Hausaufgabe 5

Schreiben sie folgende Anfragen basierend auf dem Schema für das Onlinebarverzeichnis in der OQL (Object Query Language).

1. Geben Sie ein Menü der Bar *Joe's* aus.
2. Finden Sie den Namen und Preis aller Biere bei *Joe's* die mit *B* anfangen.
3. Geben Sie die Liste aller Hersteller von Bier aus, die bei *Joe's* verkauft werden
4. Ermitteln Sie den Durchschnittspreis der Biere in allen Bars
5. Ermitteln Sie die Durchschnittspreise der Biere jeweils pro Bar

1: Variante A

```
SELECT v.bier.name, v.preis
FROM v in Verkaeufer
WHERE v.bar.name = "Joe's"
```

1: Variante B

```
SELECT v.bier.name, v.preis
FROM b in Bars, v in b.verkaufteBiere
WHERE b.name = "Joe's"
```

2:

```
SELECT v.bier.name, v.preis
FROM b in Bars, v in b.verkaufteBiere
WHERE b.name = "Joe's" AND
      v.bier.name LIKE "B*"
```

2: Variante B

```
SELECT v.bier.name, v.preis
FROM v in Verkaeufer
WHERE v.bar.name = "Joe's" AND
      v.bier.name LIKE "B*"
```

3:

```
SELECT DISTINCT v.bier.hersteller
FROM v in Verkaeufer
WHERE v.bar.name = "Joe's"
```

4:

```
AVG(SELECT v.preis
     FROM v in Verkaeufer);
```

5:

```
SELECT bar.name, AVG(preis)
FROM Verkaeufer
GROUP BY bar;
```

Quelle: <http://www.uio.no/studier/emner/matnat/ifi/INF3100/v05/undervisningsmateriale/lysark/OQL.pdf> Seiten >12