

Kaufhaus reloaded

Eine Kaufhausdatenbank besteht aus folgenden Tabellen:

Artikel

ArtNr	Bezeichnung	Verkaufspreis	Einkaufspreis
95	Kamm	1.25	0.80
97	Kamm	0.99	0.75
507	Seife	3.93	2.45
1056	Zwieback	1.20	0.90
1401	Räucherlachs	4.90	3.60
2045	Herrenhose	37.25	24.45
2046	Herrenhose	20.00	17.00
2340	Sommerkleid	94.60	71.50

Abteilung

Abteilungsname	Stockwerk	Abteilungsleiter
Lebensmittel	I	Josef Kunz
Lebensmittel	EG	Monika Stiehl
Textilien	II	Monika Stiehl

Bestand

Abteilungsname	ArtNr	Vorrat
Lebensmittel	1056	129
Lebensmittel	1401	200
Textilien	2045	14

Hinweis:

Basistabellen (Tabellen, die in der Datenbank definiert sind), werden im Rahmen des Kurses graphisch dadurch gekennzeichnet, dass es eine „nullte“ Spalte gibt, in der der Tabellename angegeben wird. Die Realisierung der Tabellen in der Datenbank enthält diese Spalte natürlich nicht.

1. Geben Sie die SQL-Befehle an, mit der die Tabellenschemata von Artikel und Bestand erzeugt werden können. Wählen Sie dabei geeignete Domänen.

```
1 CREATE TABLE Artikel (  
2   ArtNr INTEGER PRIMARY KEY NOT NULL,  
3   Bezeichnung VARCHAR(100) NOT NULL,  
4   Verkaufspreis FLOAT(2),  
5   Einkaufspreis FLOAT(2)  
6 );  
7  
8 CREATE TABLE Bestand (  
9   Abteilungsname VARCHAR(100) REFERENCES Abteilung(Abteilungsname),  
10  ArtNr INTEGER REFERENCES Artikel(ArtNr),  
11  Vorrat INTEGER,  
12  PRIMARY KEY (Abteilungsname, ArtNr)  
13 );
```

2. Es treten nun nacheinander die folgenden Änderungen auf. Aktualisieren Sie den Tabellenbestand mit den entsprechenden SQL-Befehlen:
 - (a) Ein Sommerkleid mit der Artikelnummer 2341, dem Einkaufspreis 70€ und dem Verkaufspreis 90,75€ wird in das Artikelsortiment aufgenommen.

Musterlösung:

```

1 INSERT INTO Artikel
2 VALUES (2341, 'Sommerkleid', 90.75, 70.00);

```

- (b) Der Artikel mit der Nummer 2341 wird wieder aus dem Sortiment genommen, da er den Qualitätsstandards nicht entsprochen hat.

Musterlösung:

```

1 DELETE FROM Artikel WHERE ArtNr = 2341;

```

- (c) Eine Bürste mit der Artikelnummer 2 wird in das Sortiment aufgenommen. Einkaufs- bzw. Verkaufspreis sind noch nicht festgelegt.

Musterlösung:

```

1 INSERT INTO Artikel (ArtNr, Bezeichnung)
2 VALUES (2, 'Bürste');

```

- (d) Eine Damenhose (Verkaufspreis 89€, Einkaufspreis: 60,50€) wird neu in das Sortiment aufgenommen. Eine Artikelnummer wurde noch nicht festgelegt.

falsch:

Artikelnummer ist noch nicht festgelegt.

```

1 INSERT INTO Artikel
2 VALUES (2342, 'Damenhose', 89.00, 60.50);

```

Musterlösung:

ArtNr ist der Primärschlüssel der Tabelle Artikel. Bei Eingabe eines neuen Datensatzes müssen mindestens die Werte aller Attribute, die zum Primärschlüssel gehören, angegeben werden. Da aber im Fall der Damenhose die Artikelnummer noch nicht festgelegt ist, ist eine Eingabe der Damenhose-Daten in die Tabelle Artikel nicht möglich. Hinweis: Denken Sie also immer daran, dass bei Einfügen von Datensätzen der Primärschlüssel keine NULL-Werte enthalten darf!

- (e) Die Herrenhosen werden aus dem Sortiment genommen und deshalb aus der Tabelle Artikel gelöscht.

```

1 DELETE FROM Bestand WHERE ArtNr = 2045;
2 DELETE FROM Artikel WHERE Bezeichnung = 'Herrenhose';

```

- (f) Die neue Abteilungsleiterin der Lebensmittelabteilung heißt Elvira Sommer.

Musterlösung:

```

1 UPDATE Abteilung
2 SET Abteilungsleiter = 'Elvira Sommer'
3 WHERE Abteilungsnahme = 'Lebensmittel';

```

- (g) Die Abteilung Feinkost hat einen Bestand von 150 Räucherlachspackungen.

falsch:

vergessen Wert in Abteilung einzutragen

```
1 INSERT INTO Bestand VALUES ('Feinkost', 1401, 150);
```

Musterlösung:

Die Attribute ArtNr und Abteilungsname der Tabelle Bestand sind Fremdschlüssel. Ein neuer Datensatz darf in die Tabelle nur eingefügt werden, wenn die Fremdschlüsselwerte in den entsprechenden (Primärschlüssel-)Attribute der referenzierten Tabelle auch existieren. Die Abteilung Feinkost, genauer gesagt den Abteilungsnamen 'Feinkost' gibt es in Abteilung aber noch nicht.

- i. Lösungsmöglichkeit: Die Aktualisierung kann nicht durchgeführt werden.
- ii. Lösungsmöglichkeit: Die entsprechende Abteilung Feinkost wird – natürlich in „Absprache“ mit der Kaufhausleitung – eingeführt und ein dementsprechender Datensatz in Abteilung eingefügt.

```
1 INSERT INTO Abteilung (Abteilungsname) VALUES ('Feinkost');  
2 INSERT INTO Bestand VALUES ('Feinkost', 1401, 150);
```

3. Formulieren Sie folgende Anfragen in SQL:

- (a) Gesucht sind Artikelnummer und Vorrat aller Artikel aus der Textil-Abteilung.

Musterlösung:

```
1 SELECT ArtNr, Vorrat FROM Bestand WHERE Abteilungsname =  
   ↳ 'Textilien';
```

- (b) Gesucht sind alle Informationen über die Abteilungen, die im zweiten Stock platziert sind oder von Frau Stiehl geleitet werden.

Musterlösung:

```
1 SELECT * FROM Abteilung  
2 WHERE Stockwerk = 'II' OR Abteilungsleiter = 'Monika Stiehl';
```

4. Formulieren Sie folgende SQL-Anfragen umgangssprachlich:

- (a) SQL-Anfrage:

```
1 SELECT DISTINCT Abteilungsleiter  
2 FROM Abteilung  
3 WHERE NOT (Abteilungsname = 'Kosmetik');
```

richtig:

Es werden alle Abteilungsleiter aus der Abteilungsleiter der Kosmetik-Abteilung aufgelistet.

Musterlösung:

Gesucht sind die Namen aller Abteilungsleiter mit Ausnahme der Kosmetik-Abteilung. Duplikate sollen eliminiert werden.

(b) SQL-Anfrage:

```
1 SELECT ArtNr
2 FROM Bestand
3 WHERE Abteilungsname = "Lebensmittel" AND Vorrat <= 100;
```

richtig:

Gesucht sind die Artikelnummer aller Artikel von denen 100 oder weniger Bestand vorhanden ist und die aus der Abteilung Lebensmittel stammen.

Musterlösung:

Gesucht sind die Nummern der Artikel, von denen in der Lebensmittelabteilung maximal 100 vorrätig sind.

5. Interpretieren Sie nun die obigen Tabellen als Repräsentationen der drei Relationen Artikel, Abteilung und Bestand. Bestimmen Sie die Ergebnismengen folgender relationaler Ausdrücke:

Artikel

ArtNr	Bezeichnung	Verkaufspreis	Einkaufspreis
95	Kamm	1.25	0.80
97	Kamm	0.99	0.75
507	Seife	3.93	2.45
1056	Zwieback	1.20	0.90
1401	Räucherlachs	4.90	3.60
2045	Herrenhose	37.25	24.45
2046	Herrenhose	20.00	17.00
2340	Sommerkleid	94.60	71.50

Abteilung

Abteilungsname	Stockwerk	Abteilungsleiter
Lebensmittel	I	Josef Kunz
Lebensmittel	EG	Monika Stiehl
Textilien	II	Monika Stiehl

Bestand

Abteilungsname	ArtNr	Vorrat
Lebensmittel	1056	129
Lebensmittel	1401	200
Textilien	2045	14

(a) $\pi_{ArtNr, Bezeichnung}(Artikel)$

Musterlösung:

95	Kamm
97	Kamm
507	Seife
1056	Zwieback
1401	Räucherlachs
2045	Herrenhose
2046	Herrenhose
2340	Sommerkleid

(b) $\pi_{\text{Abteilungsname}}(\text{Bestand})$

Musterlösung:

Lebensmittel
Textilien

(c) $\sigma_{((\text{Vorrat} < 100 \wedge \text{ArtNr} > 1500) \vee \text{ArtNr} < 1100)}(\text{Bestand})$

Musterlösung:

Lebensmittel	1056	129
Textilien	2045	14

(d) $\sigma_{((\text{Vorrat} < 100 \wedge (\text{ArtNr} > 1500 \vee \text{ArtNr} < 1100))}(\text{Bestand})$

Musterlösung:

Textilien	2045	14
-----------	------	----

(e) $\pi_{\text{ArtNr}}(\sigma_{\text{Bezeichnung}=\text{Herrenhose}}(\text{Artikel}))$

Musterlösung:

2045
2046

(f) $\pi_{\text{Abteilungsname}}(\text{Abteilung}) - \pi_{\text{Abteilungsname}}(\text{Bestand})$

Musterlösung:

Kosmetik

(g) $\pi_{\text{Bezeichnung}, \text{Einkaufspreis}}(\sigma_{\text{Einkaufspreis} < 2.50}(\text{Artikel})) \cup$
 $\pi_{\text{Bezeichnung}, \text{Einkaufspreis}}(\sigma_{\text{Einkaufspreis} > 20.00}(\text{Artikel}))$

Musterlösung:

Die letzten Zeile ist nicht in der Musterlösung dabei. Ich glaube aber es müsste so stimmen.

Bezeichnung	Einkaufspreis
Kamm	0.80
Kamm	0.75
Seife	2.45
Zwieback	0.90
Herrenhose	24.45
Sommerkleid	71.50

Hinweis:

Die Aufgabe bezieht sich auf die oben angegebenen Tabellen. Die in Aufgabe 3 durchgeführten Änderungen des Datenbestandes brauchen nicht berücksichtigt werden.