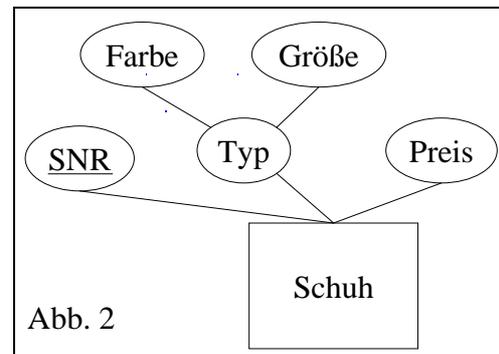


Datenbanksysteme 2 -- WS 2006/07 -- Übungsblatt 5

1. Aufgabe (SQL:1999: DISTINCT-Typ, ROW-Typ)

Definieren Sie eine SQL:1999-Tabelle *Schuhe* zur Abspeicherung der in Abb. 2 in ER-Notation spezifizierten Daten zur Verwaltung eines Schuh-lagers. Verwenden Sie hierbei für das zusammengesetzte Attribut *Typ* einen ROW-Typ, für das Attribut *Größe* einen DISTINCT-Typ. Formulieren Sie dann auf Ihrer Tabelle eine SQL:1999-Anfrage, die Ihnen die Schuhnummern (SNR) für alle Schuhe mit der Farbe "Braun" und der Größe > 43 liefert.



2. Aufgabe (SQL:1999: Tabellendefinitionen, Anfragen)

Für eine Universitätsanwendung seien u.a. folgende SQL:1999-Typen definiert:

CREATE TYPE PersonT

(Name VARCHAR (40),
 Fak REF (FakultaetT));

CREATE TYPE StudentT UNDER PersonT

(MatNr INT,
 Hauptfach VARCHAR (40),
 Nebenfach VARCHAR (40),
 Vordiplom BOOLEAN);

CREATE TYPE ProfT UNDER PersonT

(Buro BuroT,
 Besoldungsstufe CHAR(2),
 DRTitel VARCHAR(20),
 Forschungsgebiete VARCHAR(20) ARRAY[5]);

CREATE TYPE BuroT

(Gebaeude VARCHAR(40),
 Stockwerk INTEGER,
 Bueronummer INTEGER,
 Telefonnummer INTEGER);

CREATE TYPE FakultaetT

(FName VARCHAR (40),
 Dekan REF (ProfT),
 Professoren REF(ProfT) ARRAY[50],
 Studenten REF(StudentT)
 ARRAY[3000]);

a) Definieren Sie die zur Speicherung von Fakultäten, Studenten und Professoren benötigten Tabellen in SQL:1999 (ohne Tabelle zur Abspeicherung von *BuroT*-Instanzen).

b) Definieren Sie zu dem Typ *FakultaetT* eine Methode

countStudents (vd *BOOLEAN*) *RETURNS INTEGER*

die zu einer gegebenen Fakultät (d.h. zu dem Fakultätsobjekt, auf dem sie aufgerufen wird), die Anzahl aller Studenten mit Vordiplom (vd = TRUE) bzw. ohne Vordiplom liefert.

c) Schreiben Sie einen Konstruktor für *StudentT*. Dieser soll neben der Attributinitialisierung das neu generierte Objekt in das array *Studenten* des entsprechenden Fakultätsobjekts eintragen.

d) Geben Sie SQL:1999-Anfragen (ohne Join) zur Beantwortung folgender Fragen an:

1. In welchen Fakultäten sind Informatik-Studenten (Haupt-/Nebenfach) immatrikuliert?
2. Wie heisst der Dekan der Fakultät, in der der Student mit Matrikelnummer 234567 immatrikuliert ist?
3. In welchen Fakultäten sind Studenten mit Namen "Schneider" immatrikuliert?
4. Finden Sie alle Professoren mit Name und Telefonnummer, die Dekan einer Fakultät sind und die Besoldungsstufe C4 haben.

e) Definieren Sie in SQL eine Tabelle Dekane vom Typ ProfT. Schreiben Sie in SQL einen Ausdruck, der den Dekan der Fakultät 'Informatik' in die Tabelle Dekane einträgt.

3. Aufgabe (Multimengen und Tupel)

Gegeben sei das folgende NF²-Schema einer Abteilungsdatenbank (vgl. Übungsblatt 4).

Abteilung								
Abtnr	MgrNr	Budget	Ausstattung		Mitarbeiter			
			Anzahl	Typ	MaNr	Funktion	Projekt	
							PNr	PName

a) Schreiben Sie eine DDL-Anweisung, welche das Schema in einer SQL-Tabelle umsetzt. Nutzen Sie dafür die Tupel- und Multimengentypkonstruktoren von SQL:2003.

b) Geben Sie SQL:2003-Anfragen zur Beantwortung folgender Anfragen an:

1. Welche Abteilungen beschäftigen Mitarbeiter mit der Funktion eines Programmierers?
2. Welche Projekte werden in mehr als einer Abteilung bearbeitet?

4. Aufgabe (Rekursion in SQL:1999)

Die Tabellen TEIL (TNR, Bez, Mat, Bestand) und STRUKTUR (OTNR, UTNR, Anzahl) enthalten die Informationen über den Lagerbestand und die Struktur von Produkten.

TEIL			
TNR	Bezeichnung	Material	Bestand
A	Getriebe		10
B	Gehäuse	Alu	0
C	Welle	Stahl	100
D	Schraube	Stahl	200
E	Kugellager	Stahl	50
F	Scheibe	Blei	0
G	Schraube	Chrom	100

STRUKTUR		
OTNR	UTNR	Anzahl
A	B	1
A	C	5
A	D	8
B	D	4
B	G	2
C	D	4
C	E	2

a) Geben Sie mit Hilfe von RECURSIVE-Konstrukten alle Teile aus, in die Kugellager E als Unterteile eingehen. Ihre Lösung soll für beliebige Schachtelungstiefen gelten.

b) Modifizieren Sie Ihre Lösung dahingehend, dass Sie (größere STRUKTUR-Tabelle angenommen) nur Oberteile von Kugellager mit einer max. Schachtelungstiefe = 4 ausgeben (d.h. Teilhierarchie zwischen Oberteil und Kugellager E hat nicht mehr als 3 andere Teile).