

4. Übung zur Vorlesung “Datenbanken” im Sommersemester 2007

– mit Musterlösungen –

Prof. Dr. Gerd Stumme, Dipl.-Inform. Christoph Schmitz

21. Mai 2007

Aufgabe 1

In SQL gibt es bei Anfragen mittels `SELECT` die Schlüsselwörter `DISTINCT` und `ORDER BY`.

Gibt es Entsprechendes auch in der relationalen Algebra? Wenn ja: wie ist es dort realisiert? Wenn nein: warum nicht?

Beides gibt es in der relationalen Algebra nicht. `DISTINCT` dient zum Filtern von doppelten Tupeln, und `ORDER BY` zum Sortieren. Die relationale Algebra behandelt aber Mengen von Tupeln, die nicht geordnet sind und auch kein Tupel zwei Mal enthalten können.

Aufgabe 2

Formulieren Sie folgende Anfragen in SQL gegen die Uni-Datenbank von Kap. 4, S. 19. Geben Sie an, wie jeweils das Ergebnis lautet.

1. Welcher Professor (Name ist gesucht) arbeitet in Raum 310?

```
SELECT NAME FROM PROFESSOREN WHERE RAUM = 301
```

2. Welche Studenten (Matrnr ist gesucht) hören Vorlesung 5001?

```
SELECT MATRNR FROM HOEREN WHERE VORLNR = 5001
```

3. Wie viele Studenten hören Vorlesung 5001?

```
SELECT COUNT(*) FROM HOEREN WHERE VORLNR = 5001;
```

4. Wie heißen diese Studenten?

```
SELECT NAME FROM STUDENTEN, HOEREN WHERE STUDENTEN.MATRNR = HOEREN.MATRNR  
AND VORLNR = 5001;
```

5. Geben Sie eine Übersicht an, wie viele Vorlesungen es für jede Anzahl an SWS gibt.

```
SELECT SWS, COUNT(*) ANZAHL
FROM VORLESUNGEN
GROUP BY SWS
```

6. Was berechnet folgende Anfrage:

```
SELECT ASSISTENTEN.NAME, FACHGEBIET
FROM ASSISTENTEN, PROFESSOREN
WHERE BOSS = PROFESSOREN.PERSNR
AND PROFESSOREN.RANG = 'C4'
```

Welche Fachgebiete haben die Assistenten von C4-Professoren? Gesucht sind Name des Assistenten und Fachgebiet.

Aufgabe 3

Kennzeichnen Sie in folgender Anfrage die jeweiligen Entsprechungen der Operatoren

- Kreuzprodukt \times
- Projektion π
- Selektion σ
- Umbenennung ρ

der relationalen Algebra:

```
SELECT L.LName as LN, S.SName as SN
FROM Lehrer L, Klasse K, Schueler S
WHERE L.LID = K.LID
AND K.KID = S.KID
AND S.Name < 'B'
```

Kreuzprodukt $L \times K \times S$: ...FROM L, K, S ...

Projektion $\pi_{LN,SN}(\dots)$: SELECT LN, SN FROM ...

Selektion $\sigma_{S.Name < 'B'}$: WHERE ...AND S.Name < 'B'

Umbenennung $\rho_L(\text{Lehrer}), \rho_K(\text{Klasse}), \rho_S(\text{Schueler}), \rho_{LN \leftarrow LName}, \rho_{SN \leftarrow SName}$: Lehrer L, Klasse K, Schueler S; S.LName as LN, S.SName as SN

Aufgabe 4

In Aufgabe 2 vom letzten Blatt wurde ein Ausdruck gesucht, um die Matrikelnummern der Studenten zu wählen, die eine Vorlesung hören, deren Voraussetzung sie nicht gehört haben:

$$\pi_{\text{Matr}\#}(\rho_{\text{Vorl}\# \leftarrow \text{VoraussetzungVorl}\#} \\ (\pi_{\text{Matr}\#, \text{VoraussetzungVorl}\#}(\text{Hören} \bowtie \text{Voraussetzung})) - \text{Hören})$$

- Übersetzen Sie diese Anfrage nach SQL.

```
SELECT MatrNr
FROM Hören H1, Voraussetzen V
WHERE H1.VorlNr = V.VorlNr
AND NOT EXISTS (
  SELECT * FROM Hören H2
  WHERE H2.MatrNr = H1.MatrNr
  AND V.VoraussetzungVorlNr = H2.VorlNr
)
```

oder direkter

```
SELECT MatrNr
FROM (
  (
    SELECT MatrNr, VoraussetzungVorlNr
    FROM Hören, Voraussetzen
    WHERE Hören.VorlNr = Voraussetzen.VorlNr
  ) MINUS (
    SELECT * FROM HOEREN
  )
)
```

Anmerkung: in DB2 muß es statt MINUS heißen: EXCEPT

- Wo findet sich in Ihrem SQL-Statement die Mengensubtraktion “(...)–Hören” wieder?
in AND NOT EXISTS bzw. MINUS