Universität Freiburg
Institut für Informatik
Prof. Dr. G. Lausen
Dr.-Ing. T. Hornung

Georges-Köhler Allee, Geb. 51
D-79110 Freiburg
lausen@informatik.uni-freiburg.de
hornungt@informatik.uni-freiburg.de

**Übungen zur Vorlesung**
***Datenbanken und Informationssysteme***
**Wintersemester 2012/2013**
14.11.2012

# 4. Aufgabenblatt: SQL und Relationale Algebra

## Aufgaben, die nicht bewertet werden

### Übung 1

Zeigen Sie oder widerlegen Sie das Assoziativgesetz für:

(a) den natürlichen Verbund, d.h. $R_1 \bowtie (R_2 \bowtie R_3) = (R_1 \bowtie R_2) \bowtie R_3$,

(b) den linken äußeren natürlichen Verbund, d.h. $R_1 \bowtie_{lo} (R_2 \bowtie_{lo} R_3) = (R_1 \bowtie_{lo} R_2) \bowtie_{lo} R_3$.

### Übung 2

Seien gegeben die folgenden Relationsschemata:

```
Produkt(PNr, Gesamtgewicht)
PT(PNr, TNr, Anzahl)
Teil(TNr, Bezeichnung, Gewicht)
```

Das Schema PT spezifiziert, welche Teile in ein jeweiliges Produkt eingehen und in welcher Anzahl dies bzgl. diesem Produkt der Fall ist.

Geben Sie Anfragen in SQL an, die folgende Probleme lösen:

(a) Es sollen alle Produkte bestimmt werden (Ausgabe `PNr`), die aus genau 2 Teilen in beliebiger Anzahl bestehen.

(b) Es sollen alle Teile bestimmt werden (Ausgabe `TNr`), die in kein Produkt eingehen.

(c) Es sollen alle Produkte bestimmt werden (Ausgabe `PNr`), für die das angegebene Gesamtgewicht ungleich der Summe aller Ausdrücke `Anzahl * Gewicht` der in sie eingehenden Teile ist.

### Übung 3

Gegeben sind die beiden Tabellen $R$ und $S$, sowie die Ergebnis-Tabellen $E_1, E_2, E_3$:

$R$ :

| $A$ | $B$ |
|---|---|
| 1 | 1 |
| 2 | 2 |
| 2 | 3 |
| 3 | 2 |
| 4 | 1 |
| 4 | 3 |
| 4 | 4 |
| 5 | 1 |
| 5 | 5 |

$S$ :

| $B$ |
|---|
| 2 |
| 3 |
| 4 |

$E_1$ :

| $A$ |
|---|
| 2 |
| 3 |
| 4 |

$E_2$ :

| $A$ |
|---|
| 2 |
| 2 |
| 3 |
| 4 |
| 4 |

$E_3$ :

| $A$ |
|---|
| 1 |
| 2 |
| 3 |
| 4 |
| 5 |

Ordnen Sie die folgenden SQL-Anweisungen den Ergebnis-Tabellen zu, so dass die jeweiligen SQL-Anweisungen, angewandt auf $R$ und $S$, die entsprechenden Tabellen ergeben.

(1)
```
SELECT  A
FROM R NATURAL JOIN S
```

(2)
```
SELECT A
FROM R NATURAL JOIN S
GROUP BY A
```

(3)
```
SELECT DISTINCT A
FROM R LEFT OUTER JOIN S
ON R.B = S.B
```

(4)
```
SELECT DISTINCT A
FROM R RIGHT OUTER JOIN S
ON R.B = S.B
```

## Aufgaben, die bewertet werden

Laden Sie für folgende Aufgaben das Skript `create_student2.sql` von der Web-Seite herunter und führen es aus.

**Übung 4 (2 Punkte)**
Ermitteln Sie den Namen und die Durchschnittsnote der Studenten, die in Ihren Kursen eine schlechtere Durchschnittsnote als 2 haben (Ausgabe in der Form: Name, Durchschnitt).

**Übung 5 (2 Punkte)**
Ermitteln Sie die Namen der Professoren zusammen mit der Durchschnittsnote über alle Kurse, die sie gelehrt haben (Ausgabe in der Form: Name, Durchschnitt).

**Übung 6 (2 Punkte)**
Geben Sie zu jedem Studenten in der Tabelle Student die belegten Kurse pro Semester an, bei denen anhand der Note zusätzlich folgende Bewertung ausgegeben werden soll sortiert nach Name und Note (Ausgabe in der Form: Name, Kursnummer, Semester, Bewertung):

- Note $< 1.7 \rightarrow$ Sehr gut
- Note $< 2.7 \rightarrow$ Gut
- Note $< 3.7 \rightarrow$ Befriedigend
- Note $< 4.3 \rightarrow$ Ausreichend
- Ansonsten $\rightarrow$ Ungenuegend

(a) Lösen Sie die Aufgabe durch Verwendung des `CASE` Statements in Oracle.

(b) Lösen Sie die Aufgabe durch Verwendung der Oracle built-in `DECODE` Funktion.

Abzugeben durch Einwurf in den Briefkasten Raum 01-025 Gebäude 51 bis spätestens 22.11.2012, 12:00 Uhr.