



Übung zur Vorlesung *Grundlagen: Datenbanken im WS16/17*

Harald Lang, Linnea Passing (gdb@in.tum.de)

<http://www-db.in.tum.de/teaching/ws1617/grundlagen/>

Blatt Nr. 05

Tool zum Üben der relationalen Algebra:

<http://db.in.tum.de/people/sites/muehe/ira/>

Tool zum Üben von SQL-Anfragen:

<http://hyper-db.com/interface.html>

Hausaufgabe 1

Beantworten Sie mittels relationaler Algebra:

- Geben Sie einen Ausdruck an, der die Relation \neg hoeren erzeugt. Diese enthält für jeden Studenten und jede Vorlesung, die der Student **nicht** hört einen Eintrag mit Matrikelnummer und Vorlesungsnummer.
- Finden Sie alle Studenten, die keine Vorlesung hören. Geben Sie zwei verschiedene Lösungen an.

Hausaufgabe 2

„Bekanntheitsgrad“: Formulieren Sie eine SQL-Anfrage, um den Bekanntheitsgrad von Studenten zu ermitteln. Gehen Sie dabei davon aus, dass Studenten sich aus gemeinsam besuchten Vorlesungen kennen. Sortieren Sie das Ergebnis absteigend nach Bekanntheitsgrad!

Hausaufgabe 3

„Fleißige Studenten“: Formulieren Sie eine SQL-Anfrage, um die Studenten zu ermitteln, die mehr SWS belegt haben als der Durchschnitt. Berücksichtigen Sie dabei auch Totalverweigerer, die gar keine Vorlesungen hören.

Hausaufgabe 4

Gegeben sei ein erweitertes Universitätsschema mit den folgenden zusätzlichen Relationen *StudentenGF* und *ProfessorenF*:

StudentenGF : {[MatrNr : integer, Name : varchar(20), Semester : integer,
Geschlecht : char, FakName : varchar(20)]}

ProfessorenF : {[PersNr : integer, Name : varchar(20), Rang : char(2),
Raum : integer, FakName : varchar(20)]}

Die erweiterten Tabellen sind auch in der Webschnittstelle angelegt.

- Ermitteln Sie den Männeranteil an den verschiedenen Fakultäten in SQL!
- Ermitteln Sie in SQL die Studenten, die alle Vorlesungen ihrer Fakultät hören. Geben Sie zwei Lösungen an, höchstens eine davon darf auf Abzählen basieren.

Hausaufgabe 5

Gegeben sei die folgende Relation *Zehnkampf* mit Athletennamen und den von ihnen erreichten Punkten im Zehnkampf:

Name	Punkte
Eaton	8869
Suarez	8523
Behrenbruch	8126
Hardee	8671
...	...

- a) Ermitteln Sie die Goldmedaillengewinner in SQL. (Eine Goldmedaille bekommen alle Athleten, für die gilt: es gibt niemand besseren (also mit mehr Punkten).)
- b) Ermitteln Sie die Silbermedaillengewinner in SQL. (Eine Silbermedaille bekommen alle, für die gilt: es gibt genau eine/n bessere/n.)

HINWEIS: Beachten Sie, dass die Relation **Zehnkampf** in der oben genannten Webschnittstelle nicht existiert. Verwenden Sie die folgende Syntax um temporäre Relationen zu erzeugen:

```
with zehnkampf(name,punkte) as (
  values
    ('Eaton', 8869),
    ('Suarez', 8523),
    ('Behrenbruch', 8126),
    ('Hardee', 8671),
    ('Sebrle', 8869)
)
select * from zehnkampf order by punkte desc
```