



## Übung zur Vorlesung *Grundlagen: Datenbanken* im WS13/14

Henrik Mühe (muehe@in.tum.de)

<http://www-db.in.tum.de/teaching/ws1314/dbsys/exercises/>

### Blatt Nr. 5

Tool zum Üben der relationalen Algebra <http://www-db.in.tum.de/~muehe/ira/>.

SQL: <http://www-db.in.tum.de/~muehe/sql/>.

### Hausaufgabe 1

Finden Sie alle *Studenten*, die alle *Vorlesungen* gehört haben, die von Sokrates gelesen wurden. Formulieren Sie die Anfrage

- in der Relationenalgebra,
- im relationalen Tupelkalkül und
- im relationalen Domänenkalkül.

#### Formulierung in relationaler Algebra

1. Wir ermitteln die von Sokrates gelesenen *Vorlesungen*.
2. Durch Anwendung des Divisionsoperators bekommen wir die Matrikelnummern der *Studenten*, die alle von Sokrates gehaltenen *Vorlesungen* gehört haben.
3. Aus *Studenten* ermitteln wir zusätzlich die Namen der *Studenten*.

$$R := \Pi_{\text{MatrNr, Name}}(\text{Studenten} \bowtie (\text{hören} \div \Pi_{\text{VorlNr}}(\text{Vorlesungen} \bowtie_{\text{gelesenVon} = \text{PersNr}(\sigma_{\text{Name} = \text{'Sokrates'}}(\text{Professoren}))))))$$

#### Formulierung im Tupelkalkül

1. Wir ermitteln mittels des Allquantors alle *Vorlesungen*, die von Sokrates gelesen werden.
2. Wir fordern, dass ein *Student* alle diese *Vorlesungen* gehört hat (Existenzquantor).

$$\{s \mid s \in \text{Studenten} \wedge \forall v \in \text{Vorlesungen} \\ (\exists p \in \text{Professoren}(v.\text{gelesenVon} = p.\text{PersNr} \wedge p.\text{Name} = \text{'Sokrates'}) \\ \Rightarrow \exists h \in \text{hören}(s.\text{MatrNr} = h.\text{MatrNr} \wedge v.\text{VorlNr} = h.\text{VorlNr}))\}$$

#### Formulierung im Domänenkalkül

Das Vorgehen ist analog zu dem beim relationalen Tupelkalkül.

$$\{[m,n] \mid \exists s([m,n,s] \in \text{Studenten} \\ \wedge \forall v,t,s,w,g( \\ ([v,t,s,w,g] \in \text{Vorlesungen} \wedge \exists rg,ra([g,\text{'Sokrates'},rg,ra] \in \text{Professoren})) \\ \Rightarrow \\ ([m,v] \in \text{hören}) \\ ) \\ )\}$$

## Hausaufgabe 2

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

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

- Ermitteln Sie die Goldmedaillengewinner in relationaler Algebra. (Eine Goldmedaille bekommen alle, für die gilt: es gibt niemand besseren (also mit mehr Punkten).)
- Ermitteln Sie die Silbermedaillengewinner im Tupelkalkül. (Eine Silbermedaille bekommen alle, für die gilt: es gibt genau eine/nbessere/n.)

$(Zehnkampf \setminus (\Pi_{Name,Punkte}(\rho_{z1}Zehnkampf \bowtie_{z2.Punkte > Punkte} \rho_{z2}Zehnkampf)))$

$\{k \mid k \in Z \wedge \exists k_{gold} \in Z ($   
 $k_{gold}.P > k.P \wedge \forall k_{andere} \in Z ($   
 $k_{andere}.P \geq k_{gold}.P \Rightarrow k_{andere}.name = k_{gold}.name) \wedge$   
 $\neg \exists k_{zwischen} \in Z (k_{zwischen}.P > k.P \wedge k_{zwischen}.P < k_{gold}.P))\}$

## Hausaufgabe 3

Formulieren Sie folgende Anfragen auf dem bekannten Universitätsschema in SQL:

- Finden Sie die *Studenten*, die Sokrates aus *Vorlesung(en)* kennen.
  - Finden Sie die *Studenten*, die *Vorlesungen* hören, die auch Fichte hört.
  - Finden Sie die *Assistenten* von *Professoren*, die den *Studenten* Fichte unterrichtet haben – z.B. als potentielle Betreuer seiner Diplomarbeit.
  - Geben Sie die Namen der *Professoren* an, die Xenokrates aus *Vorlesungen* kennt.
  - Welche *Vorlesungen* werden von *Studenten* im Grundstudium (1.-4. Semester) gehört? Geben Sie die Titel dieser *Vorlesungen* an.
- Finden Sie die *Studenten*, die Sokrates aus *Vorlesung(en)* kennen.

```
select s.Name, s.MatrNr
from Studenten s, hoeren h, Vorlesungen v,
     Professoren p
where s.MatrNr = h.MatrNr
     and h.VorlNr = v.VorlNr
     and v.gelesenVon = p.PersNr
     and p.Name = 'Sokrates';
```

DISTINCT wäre nett, um Duplikate zu unterdrücken ist aber nicht explizit in der Aufgabe gefordert.

- (b) Finden Sie die *Studenten*, die *Vorlesungen* hören, die auch Fichte hört.

```
select distinct s1.Name, s1.MatrNr
from Studenten s1, Studenten s2, hoeren h1, hoeren h2
where s1.MatrNr = h1.MatrNr
      and s1.MatrNr != s2.MatrNr
      and s2.MatrNr = h2.MatrNr
      and h1.VorlNr = h2.VorlNr
      and s2.Name = 'Fichte';
```

- (c) Finden Sie die *Assistenten* von *Professoren*, die den *Studenten* Fichte unterrichtet haben – z.B. als potentielle Betreuer seiner Diplomarbeit.

```
select a.Name, a.PersNr
from Assistenten a, Professoren p, Vorlesungen v,
      hoeren h, Studenten s
where a.Boss = p.PersNr
      and p.PersNr = v.gelesenVon
      and v.VorlNr = h.VorlNr
      and h.MatrNr = s.MatrNr
      and s.Name = 'Fichte';
```

- (d) Geben Sie die Namen der *Professoren* an, die Xenokrates aus *Vorlesungen* kennt.

```
select p.PersNr, p.Name
from Professoren p, hoeren h, Vorlesungen v, Studenten
      s
where p.PersNr = v.gelesenVon
      and v.VorlNr = h.VorlNr
      and h.MatrNr = s.MatrNr
      and s.Name = 'Xenokrates';
```

- (e) Welche *Vorlesungen* werden von *Studenten* im Grundstudium (1.-4. Semester) gehört? Geben Sie die Titel dieser *Vorlesungen* an.

```
select v.Titel
from Vorlesungen v, hoeren h, Studenten s
where v.VorlNr = h.VorlNr
      and h.MatrNr = s.MatrNr
      and s.Semester between 1 and 4;
```