



Übung zur Vorlesung *Grundlagen: Datenbanken* im WS15/16

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

<http://www-db.in.tum.de/teaching/ws1516/grundlagen/>

Blatt Nr. 04

Tool zum Üben von SQL-Anfragen: <http://hyper-db.com/interface.html>.

Hausaufgabe 1

Formulieren Sie die folgenden Anfragen auf dem bekannten Universitätsschema im relationalen **Tupelkalkül** und im relationalen **Domänenkalkül**:

- Geben Sie alle *Vorlesungen* an, die der *Student* Xenokrates gehört hat.
- Geben Sie die Titel der direkten Voraussetzungen für die *Vorlesung* Wissenschaftstheorie an.
- Geben Sie Paare von *Studenten*(-Namen) an, die sich aus der *Vorlesung* Grundzüge kennen.

Lösung:

- Geben Sie alle *Vorlesungen* an, die der *Student* Xenokrates gehört hat.

Formulierung im Tupelkalkül

$$\{v \mid v \in \text{Vorlesungen} \wedge \exists h \in \text{hören}(v.\text{VorlNr} = h.\text{VorlNr} \wedge \exists s \in \text{Studenten}(s.\text{MatrNr} = h.\text{MatrNr} \wedge s.\text{Name} = \text{'Xenokrates'}))\}$$

Formulierung im Domänenkalkül

$$\{[v,t] \mid \exists s,g([v,t,s,g] \in \text{Vorlesungen} \wedge \exists m([m,v] \in \text{hören} \wedge \exists \text{sem}([m,\text{'Xenokrates'},\text{sem}] \in \text{Studenten})))\}$$

- Geben Sie die Titel der direkten Voraussetzungen für die *Vorlesung* Wissenschaftstheorie an.

Formulierung im Tupelkalkül

$$\{[v.\text{Titel}] \mid v \in \text{Vorlesungen} \wedge \exists \text{vor} \in \text{voraussetzen}(v.\text{VorlNr} = \text{vor.Vorgänger} \wedge \exists v2 \in \text{Vorlesungen}(v2.\text{VorlNr} = \text{vor.Nachfolger} \wedge v2.\text{Titel} = \text{'Wissenschaftstheorie'}))\}$$

Formulierung im Domänenkalkül

$$\{[t] \mid \exists v,s,g([v,t,s,g] \in \text{Vorlesungen} \wedge \exists v2([v,v2] \in \text{voraussetzen} \wedge \exists s2,g2([v2,\text{'Wissenschaftstheorie'},s2,g2] \in \text{Vorlesungen})))\}$$

- Geben Sie Paare von *Studenten*(-Namen) an, die sich aus der *Vorlesung* Grundzüge kennen.

Formulierung im Tupelkalkül

$$\{[s1.Name, s2.Name] \mid s1, s2 \in \text{Studenten} \wedge \exists h1, h2 \in \text{hören} \\ (s1.MatrNr = h1.MatrNr \wedge s2.MatrNr = h2.MatrNr \wedge \\ h1.VorlNr = h2.VorlNr \wedge s1.MatrNr \neq s2.MatrNr \wedge \\ \exists v \in \text{Vorlesungen}(h1.VorlNr = v.VorlNr \wedge v.Titel = \text{'Grundzüge'}))\}$$

Formulierung im Domänenkalkül

$$\{[n1, n2] \mid \exists m1, m2, s1, s2 (m1 \neq m2 \wedge [m1, n1, s1] \in \text{Studenten} \\ \wedge [m2, n2, s2] \in \text{Studenten} \wedge \exists v ([m1, v] \in \text{hören} \\ \wedge [m2, v] \in \text{hören} \wedge \exists s, g ([v, \text{'Grundzüge'}, s, g] \in \text{Vorlesungen})))\}$$

Hausaufgabe 2

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.

Lösung:

Formulierung in relationaler Algebra

1. Wir ermitteln die von Sokrates gelesenen *Vorlesungen* (nur die Vorlesungsnummern in diesem Fall):

$$V := \Pi_{\text{VorlNr}}(\text{Vorlesungen} \bowtie_{\text{gelesenVon} = \text{PersNr}}(\sigma_{\text{Name} = \text{'Sokrates'}}(\text{Professoren})))$$

2. Durch Anwendung des Divisionsoperators bekommen wir die Matrikelnummern der *Studenten*, die alle von Sokrates gehaltenen *Vorlesungen* gehört haben:

$$M := \text{hören} \div V$$

Wichtig: Für die relationale Division muss sichergestellt sein, dass das Schema von V eine Teilmenge des Schemas von *hören* ist: $\text{sch}(V) \subset \text{sch}(\text{hören})$

3. Aus *Studenten* ermitteln wir zusätzlich die Namen der *Studenten*:

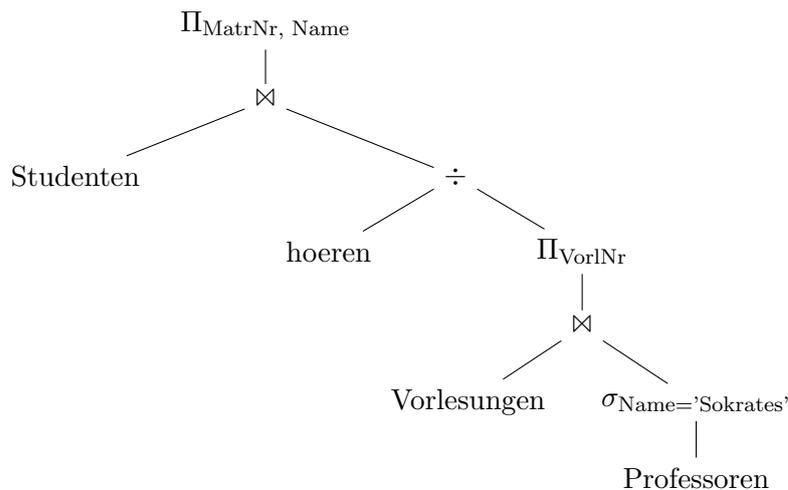
$$S := \Pi_{\text{MatrNr}, \text{Name}}(\text{Studenten} \bowtie M)$$

Zusammengefasst ergibt sich dann folgender Ausdruck:

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

Hinweis: Die Zusammenfassung ist optional. Relationale Ausdrücke können wie oben gezeigt *modularisiert* werden indem Teilausdrücke (und die dadurch definierten Relationen) an Variablen zugewiesen werden.

In Operatorbaumdarstellung:



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.

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

Hausaufgabe 3

Formulieren Sie folgende Anfragen auf dem bekannten Universitätsschema in SQL:

- (a) Finden Sie die *Studenten*, die Sokrates aus *Vorlesung(en)* kennen.
- (b) Finden Sie die *Studenten*, die *Vorlesungen* hören, die auch Fichte hört.
- (c) Finden Sie die *Assistenten* von *Professoren*, die den *Studenten* Fichte unterrichtet haben – z.B. als potentielle Betreuer seiner Diplomarbeit.
- (d) Geben Sie die Namen der *Professoren* an, die Xenokrates aus *Vorlesungen* kennt.

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

Lösung:

- (a) 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;
```