

GDB Tutorium

Woche 07

Jigao Luo

TUM

5. Dezember 2019

- select, where, from
 - group by, order by
 - Modularisierung mit with (view) \equiv Subquery
 - case ... when ... end
 - Allquantor: Umschreibung mit Existenzquantor oder count(*)
-
- Rekursion: wichtig und klausurrelevant

Auffrischung Möglichkeiten

- Im Buch ist sie recht ausführlich beschrieben.
- Die Vorlesungsfolien von Prof. Kemper.
- Die Musterlösung für Aufgabe 1.
- Die Musterlösungen der vorherigen Blättern.

Der Vorgänger von der Vorlesung 'Der Wiener Kreis ':

```
select vorgaenger
from voraussetzen, vorlesungen v
where nachfolger = v.vorlnr and v.titel = 'Der Wiener Kreis'
```


Der Vor-Vorgänger von der Vorlesung 'Der Wiener Kreis ':

```
select v1.vorgaenger
from voraussetzen v1, voraussetzen v2, vorlesungen v
where v1.nachfolger = v2.vorgaenger
and v2.nachfolger = v.vorlnr and v.titel = 'Der Wiener Kreis'
```

Der Vorgänger von der Vorlesung 'Der Wiener Kreis 'der Tiefe n :

```
select v1.vorgaenger
from voraussetzen v1, ..., voraussetzen vn_minus_1,
voraussetzen vn, vorlesungen v
where v1.nachfolger = v2.vorgaenger and ...
and vn_minus_1.nachfolger = vn.vorgaenger
vn.nachfolger = v.vorltnr and v.titel = 'Der Wiener Kreis'
```


Transitive Hülle ¹: wie Erreichbarkeit in Graph

¹https://en.wikipedia.org/wiki/Transitive_closure 

Der Vorgänger von der Vorlesung 'Der Wiener Kreis 'der Tiefe n :

```
select v1.vorgaenger
from voraussetzen v1, ..., voraussetzen vn_minus_1,
voraussetzen vn, vorlesungen v
where v1.nachfolger = v2.vorgaenger and ...
and vn_minus_1.nachfolger = vn.vorgaenger
vn.nachfolger = v.vorltnr and v.titel = 'Der Wiener Kreis'
```

Transitive Hülle ²: wie Erreichbarkeit in Graph

²https://en.wikipedia.org/wiki/Transitive_closure 

Transitive Hülle der Relation voraussetzen:

```
with recursive TransVorl (Vorg, Nachf) as (  
  select Vorgaenger, Nachfolger from voraussetzen  
  union all  
  select t.Vorg, v.Nachfolger  
  from TransVorl t, voraussetzen v  
  where t.Nachf= v.Vorgaenger)  
  
select t.Vorg  
from TransVorl t, Vorlesungen v  
where v.titel = 'Der Wiener Kreis'  
and v.vorlnr = t.nachf;
```

- 1 Eine temporäre Sicht mit **with**-Klausel, die rekursiv definiert ist.
- 2 Aus der Sicht werden die gewünschten Tupel extrahiert.

```
with recursive TransTable (Att1, Att2, Att3) as (  
  select ... from table <----- Rekursionsanfang  
  union all  
  select ... <----- Rekursionsschritt  
  from TransTable, table  
  where ...)  
  
select ... <----- Ergebnis  
from TransTable t, ...  
where ... ;
```


- Sie kann auch mit **union** funktionieren.
- “Tatsächlich gibt es zwischen **union** und **union all** einen wichtigen Unterschied:
 - Da man mit **union all** keine Duplikatelimination betreibt, ist die Query zwar potentiell schneller, terminiert aber möglicherweise nicht, denn:
 - Rekursion ist im Datenbanksystem iterativ gelöst. Es führt den Rekursionsschritt für alle Tupel aus, die im letzten Schritt neu dazu gekommen sind. Das führt aber zu Problemen, wenn man Schleifen hat:
 - Gehen wir davon aus, dass ein Tupel A bereits in der Ergebnismenge ist, und im Rekursionsschritt ein neues, identisches Tupel A hinzukommt. Wenn man jetzt **union** verwendet hat, wird das Tupel eliminiert und die Rekursion terminiert. Bei **union all** allerdings wird nicht auf Duplikate gecheckt, die Rekursion für dieses neue Tupel erneut aufgerufen und es entsteht (möglicherweise über mehrere Schritte) ein weiteres Tupel A... wir sind in einer Endlosschleife gefangen.
 - Wichtig: Genau überlegen, ob solche Schleifen mit den gegebenen Daten entsehen können und im Zweifelsfall **union** verwenden. ”³

Transitive Hülle der Relation voraussetzen with Tiefe-Anzahl:

Transitive Hülle der Relation voraussetzen with Tiefe-Anzahl:

```
with recursive TransVorl (Vorg, Nachf, Tiefe) as (  
  select Vorgaenger, Nachfolger, 1 from voraussetzen  
  union all  
  select t.Vorg, v.Nachfolger, t.Tiefe + 1  
  from TransVorl t, voraussetzen v  
  where t.Nachf= v.Vorgaenger)  
  
select *  
from TransVorl t;
```

- $\alpha \rightarrow \beta$: α bestimmt β .
- Formal:
 - Table Schema R
 - $\alpha \subseteq R, \beta \subseteq R$
 - $\alpha \rightarrow \beta \equiv \forall r, s \in R : r.\alpha = s.\alpha \Rightarrow r.\beta = s.\beta$

- R ist eine Table mit mehrere Attributen.
- α, β : als Untermenge von R , Mengen von Attributen von R
- r, s : Tuepls aus Table R

- Mehrwertige Abhängigkeiten \twoheadrightarrow : MVDs sind eine Verallgemeinerung von FDs, d.h. jede FD ist eine MVD (aber nicht unbedingt umgekehrt)

- Super-Schlüssel: $\alpha \subseteq R$ ist ein Super-Schlüssel (Superkey) $\Rightarrow \alpha \rightarrow R$.

α kann das ganze Tupel bestimmen (alle Attributen).

- Kandidatenschlüssel: $\alpha \rightarrow R \wedge \alpha$ nicht mehr minimiert

- 1NF: alle Attribute nur atomare Werte (keine Menge) annehmen.
- 2NF: in 1NF + jedes Nichtschlüsselattribut (NSA) voll funktional von jedem Schlüssel abhängt
- 3NF: in 2NF + für jede FD $\alpha \rightarrow \beta$ mindestens eine der folgenden Eigenschaften gilt:
 - $\alpha \rightarrow \beta$ ist trivial, d.h. $\beta \subseteq \alpha$
 - α ist ein Superschlüssel
 - Jedes Attribut in β ist in einem Schlüssel enthalten
- BCNF: in 3NF + für jede FD $\alpha \rightarrow \beta$ mindestens eine der folgenden Eigenschaften gilt:
 - $\alpha \rightarrow \beta$ ist trivial, d.h. $\beta \subseteq \alpha$
 - α ist ein Superschlüssel
- 4NF: in BCNF + für jede MVD $\alpha \twoheadrightarrow \beta$ mindestens eine der folgenden Eigenschaften gilt:
 - $\alpha \twoheadrightarrow \beta$ ist trivial, d.h. $\beta \subseteq \alpha$ ODER $\alpha \cup \beta = R$
 - α ist ein Superschlüssel

Gegeben sei die Relation Fahrplan, die strukturell dem folgenden Beispiel gleicht:

a) Geben Sie eine Anfrage an, welche für alle Stationen ermittelt, welche anderen Stationen erreicht werden können. Beachten Sie, dass nur tatsächlich mögliche Verbindungen ausgegeben werden sollen, d.h. die Abfahrt an einer Haltestelle darf nicht vor der Ankunft liegen.

Gegeben sei die Relation Fahrplan, die strukturell dem folgenden Beispiel gleicht:

b) Erweitern Sie ihre Anfrage aus Teilaufgabe a), sodass zusätzlich die summierte Fahrtzeit und Wartezeit sowie die gesamte Reisezeit ausgegeben wird. Die Fahrtzeit ist dabei nur die Zeit, in der man sich in einem Verkehrsmittel befindet. Die Wartezeit ist die Zeit, die bei einem Umstieg zwischen Ankunft des alten und Abfahrt des neuen Verkehrsmittels vergeht. Die Reisezeit ist die Zeit zwischen Abfahrt des ersten und Ankunft des letzten Verkehrsmittels.

Gegeben sei die Relation Fahrplan, die strukturell dem folgenden Beispiel gleicht:

c) Erweitern Sie ihre Anfrage aus Teilaufgabe a) oder b) nochmals und geben Sie die Anzahl der Umstiege für jede Verbindung aus.

Gegeben sei die Relation Fahrplan, die strukturell dem folgenden Beispiel gleicht:

d) Finden Sie die “guten” Verbindungen, um von Fröttmaning pünktlich zur Vorlesung “Grundlagen: Datenbanken” um 10:30 Uhr zu kommen. Verwenden Sie dazu Ihre Anfrage aus Teilaufgabe c). Eine Verbindung ist “gut”, wenn sie spätestens um 10:30 in “Garching, Forschungszentrum” ist und es keine andere Verbindung gibt, die später abfährt aber noch rechtzeitig eintrifft, deren Reisezeit geringer ist und bei der man weniger Umstiege hat.

Gegeben sei eine Relation

$R: \{[A : \text{integer}, B : \text{integer}, C : \text{integer}, D : \text{integer}, E : \text{integer}]\}$,

die schon sehr viele Daten enthält (Millionen Tupel). Sie “vermuten”, dass folgendes gilt:

a) AB ist ein Superschlüssel der Relation

Formulieren Sie SQL-Anfragen, die Ihre Vermutungen bestätigen oder widerlegen.

Gegeben sei eine Relation

$R: \{[A : \text{integer}, B : \text{integer}, C : \text{integer}, D : \text{integer}, E : \text{integer}]\}$,

die schon sehr viele Daten enthält (Millionen Tupel). Sie “vermuten”, dass folgendes gilt:

a) AB ist ein Superschlüssel der Relation

Formulieren Sie SQL-Anfragen, die Ihre Vermutungen bestätigen oder widerlegen.

Wann ist der Schlüsseleigenschaft verletzt?

Gegeben sei eine Relation

$R: \{[A : \text{integer}, B : \text{integer}, C : \text{integer}, D : \text{integer}, E : \text{integer}]\}$,

die schon sehr viele Daten enthält (Millionen Tupel). Sie “vermuten”, dass folgendes gilt:

a) AB ist ein Superschlüssel der Relation

Formulieren Sie SQL-Anfragen, die Ihre Vermutungen bestätigen oder widerlegen.

Wann ist der Schlüsseleigenschaft verletzt?

Mindestens zwei Tupel (Zeilen) mit den gleichen Werten für A und B.

Gegeben sei eine Relation

$R: \{[A : \text{integer}, B : \text{integer}, C : \text{integer}, D : \text{integer}, E : \text{integer}]\}$,
die schon sehr viele Daten enthält (Millionen Tupel). Sie “vermuten”,
dass folgendes gilt:

b) $DE \rightarrow B$

Formulieren Sie SQL-Anfragen, die Ihre Vermutungen bestätigen oder widerlegen.

Gegeben sei eine Relation

$R: \{[A : \text{integer}, B : \text{integer}, C : \text{integer}, D : \text{integer}, E : \text{integer}]\}$,
die schon sehr viele Daten enthält (Millionen Tupel). Sie “vermuten”,
dass folgendes gilt:

b) $DE \rightarrow B$

Formulieren Sie SQL-Anfragen, die Ihre Vermutungen bestätigen oder widerlegen.

Wann ist der FD-Eigenschaft verletzt?

Gegeben sei eine Relation

$R: \{[A : \text{integer}, B : \text{integer}, C : \text{integer}, D : \text{integer}, E : \text{integer}]\}$,
die schon sehr viele Daten enthält (Millionen Tupel). Sie “vermuten”,
dass folgendes gilt:

b) $DE \rightarrow B$

Formulieren Sie SQL-Anfragen, die Ihre Vermutungen bestätigen oder widerlegen.

Wann ist der FD-Eigenschaft verletzt?

Mindestens zwei Tupel (Zeilen) mit dem ungleichen Wert von B, aber
mit den gleichen Werten für D und E.

Betrachten Sie das Relationenschema

PunkteListe: {Name, Aufgabe, Max, Erzielt, KlausurSumme, KNote, Bonus, GNote}

1. Bestimmen Sie die geltenden FDs.
2. Bestimmen Sie die Kandidatenschlüssel.

Geben Sie für jede der Normalformen 1NF, 2NF, 3NF, BCNF, 4NF jeweils eine Relation mit FDs an, sodass die Relation in der gewünschten Normalform ist (und in keiner höheren).

Für alle Normalformen betrachten wir die Relation $R = \{A, B, C, D\}$.

Die höheren-Eigenschaften zu verletzen.

