



Universität Freiburg  
Institut für Informatik  
Prof. Dr. Georg Lausen  
Io Taxidou

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

Übungen zur Vorlesung  
**Datenbanken und Informationssysteme**  
**Wintersemester 2017/2018**

Ausgabe: 12.12.2017  
Abgabe: 18.12.2017, 12:00 Uhr

## 10. Aufgabenblatt: OLAP, Zugriffskontrolle & ER-Modell

In ILIAS finden Sie Skripte zur Erstellung der Tabellen `is_Member` und `Province`. Laden Sie die Tabellen in Ihre Datenbank (F5 - als Skript ausführen). Die Schemata der Tabellen sind wie folgt:

<b>is_Member</b>	Organization memberships of countries
<i>Organization:</i>	Name of organization
<i>Country:</i>	Country code
<i>Type:</i>	Status of membership

<b>Province</b>	Information about Provinces
<i>Name:</i>	Name of province
<i>Country:</i>	Country code
<i>Population:</i>	Population of province
<i>Area:</i>	Area of province
<i>Capital:</i>	Capital of province

### Übung 1 (3+2=5 Punkte)

Die Tabelle `CITY` der Mondial-Datenbank enthält die Attribute `LATITUDE` und `LONGITUDE` jeweils vom Typ `NUMBER`. Betrachten Sie die folgende Sicht `CITY_LAT_LON`, die eine zu `CITY` bis auf den Typ der Spalten `LATITUDE` und `LONGITUDE` identische Tabelle definiert. `LATITUDE` und `LONGITUDE` sind in der Sicht vom Typ `NUMBER(3)`, d.h. Nachkommastellen werden abgeschnitten.

```
CREATE OR REPLACE VIEW
CITY_LAT_LON ( NAME , COUNTRY , PROVINCE , POPULATION , LAT , LON ) AS
  SELECT name, country, province, population,
         CAST(TRUNC(latitude) AS NUMBER(3)),
         CAST(TRUNC(longitude) AS NUMBER(3))
  FROM City;
```

- a) Erläutern Sie das Ergebnis der folgenden SQL-Anweisung.  
Geben Sie eine äquivalente Anfrage unter Verwendung von `GROUP BY` ohne `ROLLUP` an.

```

SELECT ci.country, lat, lon, SUM(population), COUNT(*)
FROM   City_Lat_Lon ci, is_member
WHERE  ci.country = is_member.country AND is_member.organization = 'OPEC'
GROUP BY ROLLUP(ci.country, lat, lon)
ORDER BY ci.country, lat, lon ;

```

b) Erläutern Sie das Ergebnis der folgenden SQL-Anweisung.

Eine Anfrage mit CUBE kann in einer äquivalenten Anfrage unter Verwendung von GROUP BY ohne CUBE und ROLLUP analog zu (a) berechnet werden. Geben Sie an, aus wievielen mit UNION ALL verbundenen Teilanfragen eine solche Anfrage im Allgemeinen bestehen würde, wenn der CUBE-Ausdruck  $n$  Spalten hat.

```

SELECT ci.country, lat, lon, SUM(population), COUNT(*)
FROM   City_Lat_Lon ci, is_member
WHERE  ci.country = is_member.country AND is_member.organization = 'OPEC'
GROUP BY CUBE(ci.country, lat, lon)
ORDER BY ci.country, lat, lon ;

```

## Übung 2 (3 Punkte)

Angenommen eine Datenbank habe vier Benutzer ( $A$ ,  $B$ ,  $C$ ,  $D$ ).

Weiterhin sei der Benutzer  $A$  Erzeuger der Tabelle  $T$ . Betrachten Sie folgende Zugriffszuweisungen:

```

-- Benutzer A
GRANT Select, Insert, Update, Delete ON T TO B,C WITH GRANT OPTION ;

-- Benutzer B
GRANT Select ON T TO C WITH GRANT OPTION ;
GRANT Insert ON T TO C ;

-- Benutzer C
GRANT Select, Insert, Update ON T TO D ;

```

Geben Sie anhand der folgenden Tabelle an, welcher Benutzer welche Rechte an der Tabelle  $T$  besitzt:

User	Select	Insert	Update	Delete
A				
B				
C				
D				

(√: Yes, ×: No)

Anschließend führe Benutzer  $A$  die folgenden Anweisungen aus. Geben Sie den Zustand der Rechte-Tabelle nach jeder REVOKE Anweisung an. Begründen Sie dabei, ob und welche Änderungen sich ergeben haben.

```

-- Benutzer A
REVOKE Select, Insert ON T FROM C CASCADE ;

REVOKE Delete ON T FROM B RESTRICT ;

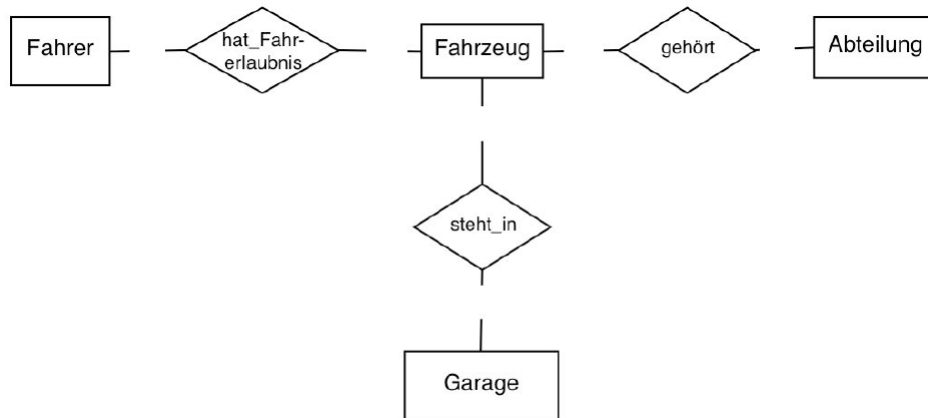
REVOKE GRANT OPTION FOR Select ON T FROM B CASCADE ;

REVOKE Update ON T FROM C RESTRICT ;

```

## Übung 3 (3 Punkte)

Gegeben ist das folgende ER-Modell der Fahrzeugverwaltung einer Firma:



Die Attribute wurden aus Vereinfachungsgründen weggelassen. Es gelten folgende Bedingungen:

- Jedes Fahrzeug gehört zu höchstens einer Abteilung, wobei aber jede Abteilung mindestens ein Fahrzeug hat.
- Für einige Fahrzeuge gibt es eine (fest zugeordnete) Einzelgarage. Jede dieser Garagen ist belegt.
- Für jedes Fahrzeug muss es mindestens drei Personen mit einer entsprechenden Fahrerlaubnis geben.

Ansonsten gibt es keine weiteren Einschränkungen.

Geben Sie die entsprechenden Beziehungskomplexitäten an.

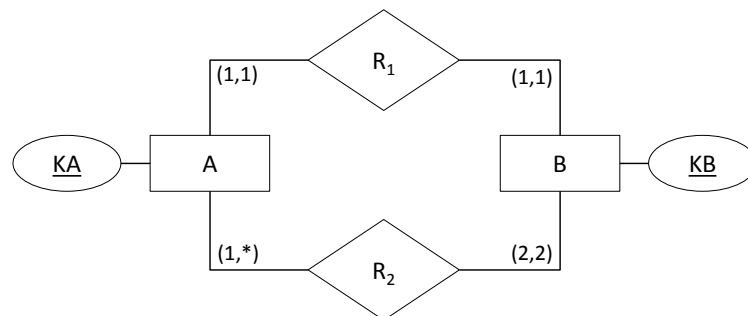
#### Übung 4 (5 Punkte)

Repräsentieren Sie den folgenden Zusammenhang als ER-Modell:

Entwickler programmieren Komponenten, die selbst wiederum aus Komponenten bestehen können. Dazu benötigen Sie eine gewisse Anzahl an Stunden. Dabei darf jede Komponente aber maximal von drei Entwicklern programmiert werden und aus maximal fünf (Sub)Komponenten bestehen. Eine (Sub)Komponente kann aber in beliebig vielen (Super)Komponenten verwendet werden. Entwickler haben einen Namen und ein Gehalt. Komponenten haben eine Bezeichnung und eine Versionsnummer.

#### Übung 5 (2+2=4 Punkte)

Betrachten Sie das folgende ER-Diagramm:



und die folgenden Entitätsmengen:

A	KA	B	KB
	$a_1$		$b_1$
	$a_2$		$b_2$

- Geben Sie Beziehungsmengen für  $R_1$  und  $R_2$  an, so dass die Beziehungskomplexitäten erfüllt sind.
- Angenommen die Beziehungskomplexität  $(1, *)$  ändert sich zu  $(1, 1)$ . Ändern Sie die Entitätsmengen  $A$  und  $B$  so ab, dass die Beziehungskomplexitäten erfüllt werden können oder begründen Sie, wieso das nicht möglich ist. Leere Entitätsmengen sind nicht erlaubt.