

Datenbanken
Wintersemester 2020/21
Prof. Dr. W. May

3. Übungsblatt: SQL

Besprechung voraussichtlich am 12./19.12.2016

Aufgabe 1 (SQL ist relational vollständig) Zeigen Sie, dass SQL *relational vollständig* ist, d.h. zu jedem Ausdruck der relationalen Algebra gibt es einen äquivalenten Ausdruck in SQL.

Aufgabe 2 (Mondial (SQL)) Gegeben sei folgendes Datenbankschema (Auszug aus Mondial)

```
Country(Name, Code, Capital, Province, Area, Population)
Organization(Name, Abbreviation, Established)
ismember(Organization, Country, Type)
```

Formulieren Sie die folgenden Anfragen in SQL:

(in den Teilaufgaben a) - e) brauchen verschiedene Arten von Mitgliedschaften nicht berücksichtigt werden!)

- a) Geben Sie von jeder Organisation die Summe der Einwohner aller Mitgliedsländer absteigend geordnet an.
- b) Welche Länder sind Mitglied in mehr als 60 Organisationen?
- c) Welche Länder mit einer Fläche von mehr als 500000 km² sind Mitglied in mehr als 60 Organisationen?
- d) Welche Länder sind in mindestens einer Organisation Mitglied, in der auch Deutschland ('D') Mitglied ist?
- e) Welche Länder sind in mindestens den Organisationen Mitglied, in denen auch Andorra ('AND') Mitglied ist?
- f) Zeigen Sie, dass es in der Datenbank keine Organisation gibt, in der alle Länder Mitglied sind!

Diese Anfragen können mit der Web-Schnittstelle zur Mondial-DB getestet werden (siehe Vorlesungsseite).

Aufgabe 3 (Gruppierung)

- Die Frage nach der größten Landesfläche in der Mondial-Datenbank lautet

```
SELECT MAX(area)
FROM Country;
```

Zusätzlich soll dazu der Landes-Code ausgegeben werden. Warum ist die folgende SQL-Anfrage fehlerhaft? Geben Sie eine entsprechend korrigierte SQL-Anfrage an.

```
SELECT MAX(area), code
FROM Country;
```

- In der Vorlesung wurde für jedes Land die Bevölkerungszahl der größten Stadt ermittelt. Geben Sie eine Anfrage an, die zusätzlich auch den Namen dieser Stadt ausgibt.

Prüfen Sie das Ergebnis auf Plausibilität. Falls daran etwas unerwartetes ist, begründen Sie dies.

Aufgabe 4 (SQL und Algebra: HAVING) Diese Aufgabe behandelt die `GROUP BY` und `HAVING`-Klauseln von SQL-Anfragen.

- a) Zeigen Sie: Alle SQL-Anfragen können auch ohne Verwendung von `HAVING` ausgedrückt werden. Geben Sie die SQL-Anfragen mit und ohne `HAVING` für die Anfrage “Welche Länder sind Mitglied in mehr als 60 Organisationen (mit Angabe der Anzahl der Mitgliedschaften)?” an.
- b) Algebra: Definieren Sie einen Operator `group-by`, der die aus SQL bekannte Funktionalität von `GROUP BY` hat. Betrachten Sie dabei für die Aggregatfunktionen nur einfache Anwendungen auf Attribute, wie z.B. `max(population)`, nicht aber komplexere Ausdrücke wie `max(population/area)`. Gehen Sie dabei wie bei der Definition der Basisoperatoren vor:
- Welche Parameter müssen dem Operator mitgegeben werden?
 - Welche Signatur hat er?
 - Welche Signatur besitzt die Ergebnisrelation (in Abhängigkeit der Eingaberelation(en))?
 - Wie ist die erhaltene Tupelmenge definiert?
- c) Geben einen Algebra-Baum für die obige Anfrage “Welche Länder sind Mitglied in mehr als 60 Organisationen?” an.

Aufgabe 5 (Duplikate) a) Überlegen Sie sich, welche Gründe es gibt, dass (i) die relationale Algebra keine Duplikate erlaubt, aber (ii) in SQL Duplikate erlaubt sind. (Es gibt jeweils mindestens 2 “gute” Gründe.)

- b) wie kann man in SQL Duplikate aus einer Tabelle entfernen?

Ausserdem:

- Formulieren Sie die Anfragen vom vorigen Blatt auch in SQL, und formulieren Sie die Anfragen an Mondial von diesem Blatt soweit möglich auch in der relationalen Algebra.
- Weitere Aufgaben finden Sie auf dem ersten Übungsblatt des SQL-Praktikums (<http://dbis.informatik.uni-goettingen.de/Teaching/DBP/>)
Dort finden Sie auch detaillierte Folien sowie ein Skript zu SQL ...