



Übungen zur Vorlesung  
*Datenbanken und Informationssysteme*  
Wintersemester 2012/2013  
12.12.2012

## 8. Aufgabenblatt: SQL

### Aufgaben, die nicht bewertet werden

#### Übung 1

Grundlage für diese Aufgabe ist der lastfm-Datensatz *lastfm100k*, auf den Sie lesenden Zugriff (Tabelle *lastfm100k*) haben, obwohl die Tabelle im Oracle SQL Developer nicht angezeigt wird. Die Tabelle hat das Schema (*SUB*, *PRE*, *OBJ*), wobei *SUB* die in *lastfm100k* betrachteten *Subjects* sind, die mit gewissen *Objects OBJ* in einer Beziehung *PRE* stehen, die durch ein *Predicate* charakterisiert ist.

- (1) Das Prädikat *userKnows* ist über Benutzern definiert und drückt die Bekanntschaften der einzelnen Benutzer aus. Definieren Sie zunächst eine Sicht *lastfmSymm* über der Tabelle, die *userKnows* symmetrisch abschließt. **Verwenden Sie diese Sicht für die folgenden Teilaufgaben.**
- (2) Berechnen Sie dann für jeden Benutzer *A* basierend auf dem Prädikat *userKnows* die Anzahl Paare (*B, C*), wobei (*A, userKnows, B*), (*A, userKnows, C*) und  $B \neq C$ , d.h. *A* kennt sowohl *B* als auch *C*; wir nennen dies eine offene Dreiecksbeziehung und stellen dies dar wie folgt: (*A | B, C*).
- (3) Berechnen Sie die Anzahl aller unterschiedlichen offenen Dreiecksbeziehungen. Vermeiden Sie Redundanzen der Form (*A | B, C*), (*A | C, B*).
- (4) Berechnen Sie dann alle (geschlossenen) Dreiecksbeziehungen über unterschiedlichen Benutzern der Art: Benutzer *A* kennt Benutzer *B*, Benutzer *B* kennt Benutzer *C*, und Benutzer *C* kennt Benutzer *A*. Bedenken Sie, dass *lastfmSymm* bzgl. *userKnows* symmetrisch abgeschlossen ist und vermeiden Sie somit die Ausgabe redundanter Dreiecksbeziehungen, d.h. *A-B-C-A* und *C-A-B-C* repräsentieren offensichtlich dieselbe Dreiecksbeziehung.
- (5) Berechnen Sie das Verhältnis geschlossener zu offener Dreiecksbeziehungen und interpretieren Sie Ihr Resultat z.B. bzgl. sozialer Aspekte.

### Aufgaben, die bewertet werden

*Hinweis:* Bitte verwenden Sie für die folgenden Aufgaben die "große" Mondial-Datenbank, die in Ihrem Schema verfügbar ist.

#### Übung 2

Die Tabelle *RIVER* repräsentiert neben der Länge eines Flusses ob der Fluss in einen See, ein Meer oder in einen anderen Fluss mündet.

Geben Sie jeweils eine Tabellenbedingung und eine Assertion für die Integritätsbedingung *IB* der Form 'Ein Fluss kann höchstens in ein Gewässer der Art Fluss, See und Meer münden' an. Versuchen Sie, dies auch in Oracle zu implementieren. Wie müssen Sie die Integritätsbedingung ändern, damit die Mondial-Instanz die Bedingung für alle Flüsse außer dem Rhein erfüllt?

### Übung 3

Definieren Sie die folgende Integritätsbedingung "Alle Länder sind Mitglied der Organisation 'World Health Organization'" als Tabellenbedingung zu der Tabelle `is_member`. Wenn Ihnen das nicht in Oracle gelingt, geben sie eine SQL-Anfrage, die die Zeichenkette 'Alles klar, alle Länder sind in der WHO.' liefert, wenn die Integrität erfüllt ist, bzw. andernfalls die Zeichenkette 'Integrität verletzt.'

*Hinweis:* Sie können die Default-Tabelle `DUAL` von Oracle verwenden, die gerade eine Zeile und ein Attribut enthält. Ihre Anfrage kann somit die Struktur "SELECT <hier der Test> FROM DUAL;" haben.

### Übung 4

Überprüfen Sie die Integritätsbedingung "Für die Länder der 'EU' ist die durchschnittliche mit der Bevölkerungszahl gewichtete Inflationsrate kleiner 3%" mittels einer SQL-Anfrage. Wenn die Datenbank die Integritätsbedingung erfüllt, dann soll Ihre Anfrage die Zeichenkette 'Inflationsrate akzeptabel.' liefern. Anderenfalls soll Ihre Anfrage die Fehlernachricht 'Verletzung der Integrität.' liefern.

Abzugeben durch Einwurf in den Briefkasten Raum 01-025 Gebäude 51 bis spätestens 20.12.2012, 12:00 Uhr