



Universität Freiburg
Institut für Informatik
Prof. Dr. G. Lausen
Alexander Schätzle
Martin Przyjacił-Zablocki

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

Übungen zur Vorlesung
Datenbanken und Informationssysteme
Wintersemester 2013/2014
27.11.2013

6. Aufgabenblatt: SQL - Prozeduren, Funktionen & Integrität

Aufgaben, die nicht bewertet werden

Übung 1

In der Vorlesung wurde gezeigt, wie mit einer Funktion zu einem gegebenen Land alle erreichbaren Länder berechnet werden können (Kapitel 3.13, Seite 5). Die direkte Rückgabe einer Tabelle aus einer Funktion wird von Oracle jedoch nicht unterstützt. Betrachten Sie daher die folgende Prozedur *ErreichbarVon*, die das gleiche Ergebnis berechnet.

```
1  -- symmetric VIEW of table Borders
2  CREATE VIEW SymBorders AS
3  SELECT country1, country2 FROM borders
4  UNION
5  SELECT country2 AS country1, country1 AS country2 FROM borders;
6
7  -- first create the result table
8  CREATE TABLE Erreichbar(
9      nach CHAR(2),
10     anzahl NUMBER );
11
12 -- then create the procedure
13 CREATE OR REPLACE
14 PROCEDURE ErreichbarVon(von CHAR) AS
15     alt INTEGER; neu INTEGER;
16 BEGIN
17     DELETE FROM Erreichbar;
18     INSERT INTO Erreichbar
19         SELECT sb.country2 AS nach, 1 AS anzahl
20         FROM SymBorders sb WHERE sb.country1 = von;
21     alt := 0;
22     SELECT COUNT(*) INTO neu FROM Erreichbar;
23     WHILE (alt <> neu) LOOP
24         alt := neu;
25         INSERT INTO Erreichbar
26             SELECT DISTINCT sb.country2 AS nach, (e.anzahl + 1) AS anzahl
27             FROM Erreichbar e, SymBorders sb
28             WHERE e.nach = sb.country1 AND sb.country2 <> von
29                 AND sb.country2 NOT IN (SELECT nach FROM Erreichbar);
30         SELECT COUNT(*) INTO neu FROM Erreichbar;
31     END LOOP;
32 END;
```

Die Prozedur wird mittels des Kommandos `EXEC ErreichbarVon('...')`; ausgeführt, nachdem Sie kompiliert wurde. Das Ergebnis kann anschließend der Tabelle `Erreichbar` entnommen werden, die vor dem Erstellen der Prozedur angelegt werden muss.

a) Angenommen der Inhalt der Tabelle `Borders` sei wie folgt:

<i>country1</i>	<i>country2</i>
0	1
1	2
2	3
3	4
1	4
4	5

Protokollieren Sie die beim Aufruf von `ErreichbarVon('0')` zu beobachtenden Änderungen der Tabelle `Erreichbar` nach jeder Iteration.

b) Angenommen, Sie wollen zu jedem erreichbaren Land auch den konkreten Weg in Form der Folge der Grenzübergänge ermitteln. Wie können Sie dies durch Änderung der Prozedur `ErreichbarVon` erreichen? Sie können die üblichen built-in Funktionen für Zeichenketten verwenden und die Tabelle `Erreichbar` um eine geeignete Spalte erweitern.

Übung 2

Erklären Sie in Ihren eigenen Worten, was man bei Relationalen Datenbanken unter den Begriffen *Referentielle Integrität* und *Referentielle Aktionen* versteht. Verdeutlichen Sie Ihre Ausführungen anhand eines oder mehrerer selbst gewählter Beispiele. Verwenden Sie dabei nicht die Beispiele aus der Vorlesung (Studentenverwaltung, Mondial).

Übung 3

Betrachten Sie die folgende Tabellendefinition

```
CREATE TABLE T (
  A NUMBER,
  B NUMBER );
```

und die folgenden beiden Integritätsbedingungen:

- (1) *Zwei Tupel einer Instanz von T, die den gleichen A-Wert haben, haben auch den gleichen B-Wert.*
- (2) *Die Summe aller Werte von A ist mindestens so groß wie die Summe aller Werte von B.*

- a) Definieren diese Bedingungen jeweils als Tabellenbedingungen.
- b) Definieren diese Bedingungen jeweils als Assertions.

Hinweis: Oracle unterstützt keine Subanfragen in Tabellenbedingungen und auch keine Assertions, wie sie im SQL-92 Standard definiert werden. Sie können diese Aufgabe daher leider nicht mit der von uns bereitgestellten Datenbank testen.

Aufgaben, die bewertet werden (20 Punkte)

Übung 4 (4 Punkte)

Erstellen Sie eine Prozedur `SpokenLanguage(lang CHAR)`, die beim Aufruf von `EXEC SpokenLanguage('...')`; eine Tabelle mit allen Ländern liefert, in denen die Sprache (`lang`) gesprochen wird. Die Tabelle soll aufsteigend nach Ländername sortiert sein. Erstellen Sie zunächst eine geeignete Tabelle, die das Ergebnis der Prozedur aufnehmen soll.

Übung 5 (3+4 Punkte)

- a) Erstellen Sie eine Funktion `NumberOfBiggerCities(lcode CHAR, minPop NUMBER)`, die für ein gegebenes Land (`lcode`) die Anzahl der Städte des Landes zurückgibt, deren Einwohnerzahl größer als ein vorgegebener Wert (`minPop`) ist. Sie können Ihre Funktion mit folgendem Aufruf testen (siehe Hinweis):

```
SELECT NumberOfBiggerCities('de', 100000) FROM Dual;
```

- b) Erstellen Sie eine Funktion `BigCapital(lcode CHAR)`, die für ein gegebenes Land (`lcode`) den Namen der Hauptstadt zurückgibt, falls die Hauptstadt die meisten Einwohner aller Städte des Landes hat, und `null` andernfalls. Sie können Ihre Funktion mit folgendem Aufruf testen (siehe Hinweis):

```
SELECT BigCapital('de') FROM Dual; -- should return 'Berlin'
```

```
SELECT BigCapital('au') FROM Dual; -- should return 'null'
```

Hinweis: Die SQL Syntax von Oracle erfordert immer die Angabe einer `FROM` Klausel aber nicht alle Anfragen benötigen die Angabe einer Tabelle. In solchen Fällen kann die Tabelle `Dual` verwendet werden, die nur eine Spalte mit dem Namen `Dummy` mit einem einzigen Wert (`x`) enthält.

Übung 6 (2+3+4 Punkte)

Betrachten Sie die folgende Integritätsbedingung für die Mondial-Datenbank:

Die Summe der Einwohner aller Städte eines Landes darf höchstens so groß sein wie die Einwohnerzahl des entsprechenden Landes.

- a) Definieren Sie diese Bedingung als Tabellenbedingung und diskutieren Sie mögliche Einschränkungen.
b) Definieren Sie diese Bedingung als Assertion.
c) Definieren Sie Trigger zur Gewährleistung der Integritätsbedingung unabhängig von (a) und (b). Welche Operationen auf welchen Tabellen sind kritisch? Es genügt, wenn Sie sich bei der Definition der Trigger auf UPDATE Operationen beschränken.

Hinweis: Lösen Sie die Aufgabe unter Verwendung des SQL-92 Standards, wie er auch in den Beispielen der Vorlesung verwendet wird.

Abzugeben durch Einwurf in den Briefkasten Raum 01-025 Gebäude 51 bis spätestens 02.12.2013, 12:00 Uhr.