

# Verknüpfen von Tabellen einer Relationalen Datenbank

Michael Dienert

6. November 2013

## 1 Verknüpfung von Tabellen

Um Redundanzfreiheit zu gewähren werden die Tabellen einer relationalen Datenbank normalisiert.

Ein Beispiel: bei einer Kundentabelle werden die Wohnorte der Kunden in einer zweiten Tabelle ausgelagert und über Schlüssel mit den Personen verknüpft.

Möchte man sich nun eine Liste seiner Kunden mit deren Wohnorten ausgeben lassen, muss man die Tabellen `Ort` und `Kunde` wieder verbinden.

Die Verbindung zweier oder mehrerer Tabellen wird *Join* genannt.

Hierfür gibt es in SQL zwei Möglichkeiten, die mit den folgenden, praktischen Übungen untersucht werden sollen.

## 2 Die Übungsdatenbank joins

Um Zugriff auf den Datenbankserver zu haben, muss man sich selbst als Benutzer registrieren. Dazu wechselt man zum Benutzer `postgres`, legt sich selbst dann als berechtigter Benutzer an und wechselt wieder ins eigene Benutzerkonto zurück. Dazu muss man folgende Befehle verwenden (der Postgres-Superuser heisst jetzt `walther`):

```
su postgres
password: postgres
createuser <meinEigenerAnmeldename>
exit
```

Anschliessend soll man eine Datei mit SQL-Befehlen erstellen, die die untenstehenden beiden Relationen enthält. Der Anfang der Datei soll so aussehen:

```
DROP DATABASE joins;
CREATE DATABASE joins;
\connect joins
...
...
```

Der Rest der Datei soll Befehle enthalten, die die beiden Tabellen `relation1` und `relation2` erzeugen und mit Daten füllen.

Die Datei kann dann so in den Datenbankservers eingespielt werden:

```
psql -f joins.sql postgres
```

Um interaktiv mit der soeben erzeugten Datenbank zu arbeiten, muss man den postgres-Client so aufrufen:

```
psql joins
```

relation1		relation2	
id	ziffern	id	buchstaben
20	0	1	A
21	1	2	B
22	2	3	C
...	...	...	...
...	...	...	...
29	9	26	Z

### 3 Aufgaben

Notiere die Antworten auf einem Blatt. Dieses wird eingesammelt und bewertet!

1. Welches Ergebnis liefert folgende SQL-Abfrage:

```
SELECT * from relation1, relation2;
```

2. Recherchiere im Internet den Begriff *Kreuzprodukt* und gib das Ergebnis der Recherche in **eigenen Worten** wieder. Falls der eingereichte Text irgendwie mit Internetquellen korreliert, wird er nicht gewertet!
3. Wie kann man obige Abfrage erweitern, so dass nur die Zeilen ausgegeben werden, bei denen die Primärschlüssel gleich sind?
4. Welches Ergebnis liefert folgende SQL-Abfrage (das Ergebnis ist **verbal** zu beschreiben, nicht mit copy und paste einzufügen!!):

```
SELECT * from relation1 INNER JOIN relation2  
ON relation1.id = relation2.id;
```

5. Was ist der entscheidende Vorteil dieser Abfrage. Denke dabei auch an den Fall, dass viele grosse Tabellen miteinander verknüpft werden müssen.
6. Teste nun diese SQL-Abfragen und beschreibe ebenfalls verbal die Funktion von LEFT JOIN und RIGHT JOIN:

```
SELECT * from relation1 LEFT JOIN relation2 ON relation1.id =  
relation2.id;
```

```
SELECT * from relation1 RIGHT JOIN relation2  
ON relation1.id = relation2.id;
```