

# Datenbanken

## Proseminar Abgabe 2

### Wintersemester 2013/2014

*Abgabetermin: 7. Dezember 2013, 23:59.*

## 1 Aufgabestellung

Das Ziel dieses zweiten Teiles des Projektes ist es

1. das ER-Modell des ersten Teiles zu korrigieren und zu verfeinern;
2. das ER-Modell in ein relationales Schema umzusetzen und dieses Schema in SQL zu implementieren;
3. die Tabellen in der so entstandenen Datenbank mit Daten zu füllen;
4. die Datenbank wieder vollständig (Schemas und Daten) zu entfernen.

Die einzelnen Punkte werden nachfolgend im Detail beschrieben.

### 1.1 Überarbeitung des ER-Diagramms

Korrigieren und verfeinern Sie, wo nötig, Ihr ER-Modell aus dem ersten Teil des Projektes und zeichnen Sie das ER-Diagramm. Beachten Sie dabei die Rückmeldung, die Sie zu Ihrer ER-Modellierung erhalten haben.

Wo sich aus dem ER-Diagramm Unklarheiten ergeben könnten, können Sie das Diagramm kurz erläutern. Beispiel: “Eine Entität des Typs *Waren* beschreibt eine Art von Ware, z.B. das Buch ‘Herr der Ringe’, nicht die einzelnen Exemplare. Die Anzahl der tatsächlichen Exemplare einer jeden Ware, die auf Lager ist, wird durch das Attribut *Anzahl* beschrieben (z.B. hat das Attribut *Anzahl* den Wert 3 für das Buch ‘Herr der Ringe’, wenn drei Exemplare des Buches im Lager sind).”

**Abgabe:** PDF-Datei (*er.pdf*) mit dem verbesserten ER-Diagramm (Team-Name, sowie Namen und Matrikelnummern aller Mitglieder nicht vergessen).

### 1.2 Übersetzung des ER-Modells

Schreiben Sie ein SQL Skript zur Erstellung der Tabellen (ein SQL Skript ist eine Textdatei mit SQL Befehlen, die durch Strichpunkt getrennt sind).

Ihre Tabellendefinitionen sollte folgende Punkte abdecken:

1. passende Typen für die Attribute definieren und Default-Werte, wo sinnvoll;
2. Primärschlüssel festlegen;
3. Fremdschlüssel-Beziehungen definieren;
4. etwaige andere Integritätsbedingungen auf Attribute (z.B. NULL-Werte verbieten, Wertebereiche angeben) und/oder Tabellen festlegen.

Wo Ihre Wahl (z.B. einer Integritätsbedingung) nicht offensichtlich ist, bitte einen Kommentar hinzufügen. Für einzeilige Kommentare wird der doppelte Bindestrich (`-- Kommentar`) verwendet, für mehrzeilige Kommentare C-artige Bruchstrich-Stern Paare (`/* Kommentar */`).

**Abgabe:** SQL Skript `create.sql`, das alle Tabellen zum ER-Diagramm wie oben beschrieben erstellt.

### 1.3 Daten laden

Laden Sie mithilfe von `INSERT` Befehlen Daten in Ihre Datenbank. Achten Sie dabei darauf, dass durch Fremdschlüsselbedingungen keine Fehler auftreten.

**Abgabe:** SQL Skript `load.sql`, das die vorher erzeugten Tabellen mit Daten füllt.

### 1.4 Tabellen löschen

Löschen Sie alle Tabellen aus Ihrer Datenbank. Achten Sie darauf, dass durch Fremdschlüsselbedingungen keine Fehler auftreten.

**Abgabe:** SQL Skript `clean.sql`, das alle Tabellen löscht, die von `create.sql` erzeugt und von `load.sql` mit Daten gefüllt wurden.

## 2 Zugang zum Datenbankserver

Für das Proseminar wird ein PostgreSQL Datenbankserver zur Verfügung gestellt:

- *Host:* `dumbo.cosy.sbg.ac.at`
- *Port:* 5432
- *Datenbank:* DBWS2013-GruppeA b.z.w. DBWS2013-GruppeB

Benutzername und Passwort sollten Sie via Email bereits erhalten haben. Sie können sich mithilfe des PostgreSQL Clients `psql` wie folgt einloggen:

```
psql -h dumbo.cosy.sbg.ac.at -U <user> -d <datenbank>
```

Eine Starthilfe zur Verwendung des PostgreSQL Clients `psql` finden Sie hier:

<http://www.postgresql.org/docs/9.2/static/tutorial-accessdb.html>

Natürlich können Sie die Übung auch auf Ihrem eigenen PostgreSQL Server durchführen. PostgreSQL steht kostenlos zur Verfügung und ist einfach zu installieren. Sie sollten Ihre Skripts jedoch unbedingt auf dem Server `dumbo.cosy.sbg.ac.at` testen.

### 3 Abgabeformat

Die Abgabe erfolgt via E-Mail an `petra.kirchwegger@sbg.ac.at` mit dem Betreff `PS Datenbanken <teamname>`.

Bitte halten Sie sich an folgendes Format für die Abgabe:

1. PDF des ER-Diagramms und die drei SQL Skripts werden in eine ZIP Datei mit dem Namen `<teamname>_2.zip` gepackt, wobei `<teamname>` durch den Namen Ihres Teams zu ersetzen ist, z.B. `a3_2.zip`, `b4_2.zip`.
2. Die PDF Datei und alle SQL Dateien enthalten einen Kopfteil mit folgenden Informationen (als Kommentar in SQL Skripts):
  - Name des Teams
  - Name und Matrikelnummer aller Mitglieder

Die Einteilung in Teams finden Sie auf der Website des Proseminars.

**Achtung:** Verspätete Abgaben oder Abgaben, die das oben angegebene Format nicht beachten, werden zurückgewiesen und nicht bewertet.

### 4 Bewertungskriterien

Die SQL Skripts werden wie folgt auf einer leeren Datenbank getestet. Stellen Sie sicher, dass alle Ihre SQL Skripts mit Ihrem `<user>` und Ihrer `<datenbank>` (DBWS2013-GruppeA oder DBWS2013-GruppeB) diesen Test ohne Fehler bestehen.

```
psql -h dumbo.cosy.sbg.ac.at -U <user> -d <datenbank> -f create.sql
psql -h dumbo.cosy.sbg.ac.at -U <user> -d <datenbank> -f load.sql
psql -h dumbo.cosy.sbg.ac.at -U <user> -d <datenbank> -f clean.sql
```

Zusätzlich zur Ausführbarkeit der SQL-Skripts fließen folgende Kriterien in die Bewertung ein:

- *Relationale Umsetzung:* Korrektheit der Umsetzung des ER-Modells in Tabellen, d.h. die Tabellen mit den angegebenen Integritätsbedingungen sollen das ER-Modell möglichst getreu wiedergeben. Die Attributtypen sollten sinnvoll und nachvollziehbar sein (im Zweifelsfall bitte kommentieren).
- *Zusätzliche Bedingungen* (über Schlüssel-/Fremdschlüsselbedingungen hinaus) auf Attribute und Tabellen, sofern sinnvoll, werden positiv bewertet.

Details zur Syntax der SQL Befehle in PostgreSQL (z.B. Integritätsbedingungen auf Spalten und Tabellen) finden Sie in der offiziellen Dokumentation:

<http://www.postgresql.org/docs/9.2/interactive/sql.html>