

Walther- Rathenau- Gewerbeschule Freiburg	<h1>Eine einfache CD-Datenbank</h1> <p>mit postgres und uuids</p>	Fach: SAE	Gruppe:
		28. Oktober 2022	
		Name:	
		Klasse: E2F11t	Seite 1
Punkte: /20		Note:	

## 1 Eine supereinfache Datenbank fuer CDs

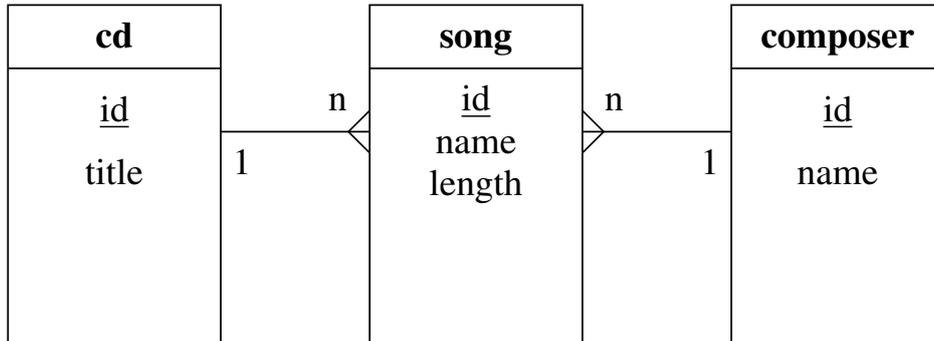


Abbildung 1: Einfache CD-Datenbank

Die Abbildung 1 zeigt das ERM-Modell einer ganz einfachen Datenbank für Musik-CDs. Die Primärschlüssel in dieser Aufgabe sollen zunächst manuell vergeben werden. Als Datentyp wird INTEGER verwendet.

### 1.1 Aufgabe

In diesem ERM sind keine Fremdschlüssel eingetragen. Die Fremdschlüssel ergeben sich eindeutig aus den beiden angegebenen 1-n-Beziehungen.

Trage die fehlenden Fremdschlüssel in das Diagramm ein. Ihre Namen kannst Du frei wählen.

## 2 Datenbank-Skript

Die Syntax und viele Beispiele zu den im Folgenden benötigten SQL-Kommandos finden sich im Handbuch zu *postgres*:

```
https://www.postgresql.org/docs/current/index.html
```

Folgende Kapitel sind wichtig:

- Datenbank löschen: Kapitel 23.5
- Datenbank erstellen: Kapitel 23.2
- mit der Datenbank verbinden:

```
https://www.postgresql.org/docs/15/app-psql.html
#oder
man psql #im terminal eingeben, mit Taste q man-page verlassen.
```

auf der man-page nach \connect suchen

- Tabellen erstellen: Kapitel 2.3, 5.1 und 5.4
- Datensätze eingeben: Kapitel 6.1
- Datensätze löschen: Kapitel 6.3
- Datensätze ändern: Kapitel 6.2
- Wie man eine einfache Abfrage erstellt: Kapitel 2.5

## 2.1 Aufgabe: Tabellen erstellen

Mit einem Texteditor (z.B. gedit) soll ein sql-Skript geschrieben werden, welches drei Tabellen entsprechend der Abb. 1 anlegt. Dabei sollen die Primärschlüssel dem RDBMS mit dem sog. *Constraint PRIMARY KEY* bekannt gemacht werden.

Um auch die *referentielle Integrität* sicherzustellen, sollen die Fremdschlüssel mit dem *FOREIGN KEY-Constraint* mit den entsprechenden Primärschlüsseln verknüpft werden. Es soll noch keine Lösch- und Änderungsweitergabe eingerichtet werden.

### 2.1.1 Hinweis zum sql-Skript

Das erstellte sql-Skript kann mit folgendem Kommando (**aus dem Terminal heraus**) an den Datenbankserver geschickt werden:

```
sudo -u postgres psql < cddb.sql
```

Das Kommando funktioniert nur, wenn die Datei `cddb.sql` auch im aktuellen Verzeichnis liegt.

Mit `pwd` kann man sich im Terminal den Pfad des aktuellen Verzeichnisses ausgeben lassen. **Dorthin muss dann vom Editor aus das sql-Skript gespeichert werden!!**

## 2.2 Aufgabe: Testdatensätze eingeben

Gib in die Tabellen mindestens je 2 Datensätze ein.

## 2.3 Aufgabe: Testen der referentiellen Integrität

Teste, ob man durch Löschen (`DELETE FROM`) oder Ändern (`UPDATE tabelle SET`) von Datensätzen einmal hergestellte Beziehungen zwischen Primär- und Fremdschlüssel aufbrechen kann.

# 3 Lösch- und Änderungsweitergabe

Erweitere die *FOREIGN-KEY-Constraints* um Lösch- und Änderungsweitergaben. Versuche nun, den Primärschlüssel zu ändern. Was passiert mit den Fremdschlüsseln?

Was passiert, wenn man einen Datensatz aus der Tabelle `cd` oder `composer` löscht?

## 4 Für die Schnellen

Wenn man im sql-Skript den Datentyp der Primärschlüssel auf `SERIAL` ändert, werden die Schlüssel automatisch hochgezählt.

Beim Einfügen eines Datensatzes in die Tabelle `cd` oder `composer` darf dann nur der Name oder der Titel eingefügt werden.

Dann wiederum hat man aber das Problem, dass man beim Einfügen eines Datensatzes in die Tabelle `song` nicht weiss, welche Schlüsselwerte automatisch vergeben wurden.

Durch eine Abfrage kann man sich natürlich zuvor die Tabellen `cd` und `composer` anzeigen lassen und die Schlüssel ablesen.

Man kann das aber Automatisieren: mit einer sog. *Unterabfrage* (*subselect*) direkt *innerhalb* der `INSERT`-Anweisung kann man die entsprechenden Primärschlüssel suchen und einfügen.

Die `SELECT`-Anweisung, die anstelle eines festen Werts für den Fremdschlüssel steht, muss dabei in runde Klammern gesetzt werden.

Die `SELECT`-Anweisung benötigt hier einen `WHERE`-Ausdruck. Die Beispiele in Kapitel 2.5 der postgres-Doku reichen hierfür aus.