



## Beziehungen in SQL

### Primärschlüssel

Ein einzelnes Attribut wird zum Primärschlüssel (PK  $\hat{=}$  Primary Key) erklärt, in dem dieses Attribut bei der Erstellung der Tabelle den Zusatz "Primary Key" erhält.

```
CREATE TABLE Tabellenname (  
  Feld1 Datentyp Primary Key,  
  Feld2 Datentyp ,  
  ...  
)
```

```
CREATE TABLE RECHNUNGEN (  
  ID Counter Primary Key  
  NAME TEXT(30) NOT NULL,  
  BETRAG CURRENCY NOT NULL  
)
```

Zusammengesetzte Primärschlüssel werden wie folgt erstellt:

```
CREATE TABLE Tabellenname (  
  Feld1 Datentyp [ NOT NULL ],  
  Feld2 Datentyp [ NOT NULL ],  
  ....  
  Primary Key(Feld1, Feld2...)  
)
```

```
CREATE TABLE RECHNUNGEN (  
  NAME TEXT(30) NOT NULL,  
  VNAME TEXT(30) NOT NULL,  
  BETRAG CURRENCY NOT NULL,  
  Primary Key(NAME, VNAME)  
)
```

### Fremdschlüssel

Ein Fremdschlüssel (FK  $\hat{=}$  Foreign Key) wird in einer Tabelle wie folgt eingerichtet:

```
CREATE TABLE Tabellenname (  
  Feld1 Datentyp Primary Key,  
  Feld2 Datentyp,  
  Foreign Key(Feld2) References FremdTabelle  
)
```

FK-Attribut in  
Tabelle

```
CREATE TABLE RECHNUNGEN (  
  ID Counter Primary Key  
  NAME TEXT(30) NOT NULL,  
  BETRAG CURRENCY NOT NULL,  
  Foreign Key(NAME) References Adressen  
)
```

Fremdtabelle

Wird in der referenzierten Tabelle (im Beispiel: Adressen) ein zusammengesetzter Primärschlüssel verwendet, müssen in der referenzierenden Tabelle (im Beispiel: Rechnungen) die gleichen Attribute (andere Attributbezeichnungen sind natürlich möglich) verwendet werden. Hinter "Foreign Key" wird nun eine Aufzählung verwendet.

### Beispiel

```
Foreign Key(FK_NAME, FK_VNAME) References Adressen
```



B	Lehrgang: Datenbanken	Arbeitsblatt Nr. 12
S	Thema: Beziehungen in SQL	Datum:
G	Name:	Seite 2 von 2

## Einrichten von Beziehungen in Tabellen

Für die Einrichtung einer Beziehung zwischen zwei Tabellen, muss in einer Tabelle (mindestens) ein Attribut zum Primärschlüssel erklärt werden. In der zweiten Tabelle wird (mindestens) ein Attribut zum Fremdschlüssel erklärt.

Die Datentypen der als PK und FK verwendeten Attribute müssen identisch sein!

## Ändern von Schlüsseln

In Tabellen können auch nachträglich Primär- bzw. Fremdschlüssel hinzugefügt<sup>1</sup> werden. Hierzu wird der Befehl **"Alter Table"** verwendet.

```
ALTER TABLE TabellenName ADD
    PRIMARY KEY (Feldnamensliste)

ALTER TABLE TabellenName ADD
    FOREIGN KEY (Feldnamensliste) REFERENCES Fremdtabelle
```

## Übungen

Für einen Buchhändler ist eine Datenbank zu modellieren und in ein Tabelledesign zu überführen. Bei der Besprechung mit dem Buchhändler wird folgender Ausschnitt der Realität festgehalten:

Es erscheinen Bücher, die von Autoren geschrieben werden. Ein Autor kann mehrere Bücher geschrieben haben. Aber jedes Buch hat genau einen Autor. Weiterhin haben Bücher einen Titel, eine ISBN-Nummer und einen Preis. Zu den Autoren sollen deren Vornamen und Namen sowie deren Geburtstag gespeichert werden. Die Bücher werden durch einen Verlag herausgegeben. Ein Verlag hat eine Firmennamen und eine Anschrift. In einem Verlag erscheinen mehrere Bücher; allerdings erscheint jedes Buch nur in genau einem Verlag.

1. Erstellen Sie ein ER-Diagramm zu der obigen Realitätsbeschreibung.
2. Überführen Sie das ER-Diagramm in ein Tabellendesign für das Relationale Datenbankmodell und erzeugen Sie die Tabellen und die erforderlichen Beziehungen mittels SQL.
3. Der Buchhändler wünscht nachträglich eine Ergänzung. Zu jedem Buch soll die Information gespeichert werden, ob es sich um ein Taschenbuch oder um eine gebundene Ausgabe handelt. Erstellen Sie daher die Tabelle "Ausgabe(ID, Bezeichnung)" mit den Datentypen Counter und Text(30). Ergänzen Sie die Relation "Buch" um ein Fremdschlüsselattribut namens Ausgabe, das eine Beziehung zur Tabelle "Ausgabe" herstellt.

<sup>1</sup> Je nach verwendetem DBMS können Schlüssel auch mit **DROP** anstelle von **ADD** nachträglich entfernt werden