



**Festlegen eines Fremdschlüssels**

Der Ort in der Tabelle kunden wird durch das Feld Ortplz eindeutig bestimmt. Die Postleitzahl Ortplz kann in der Tabelle orte nachgeschlagen werden. Um dies dem Datenbanksystem anzuzeigen, wird in der Tabelle kunden ein Fremdschlüssel deklariert, der auf den Primärschlüssel der Tabelle orte verweist (referenziert).

Diese Beschränkung (CONSTRAINT) definiert diesen Fremdschlüssel (FOREIGN KEY) mit Verweis (REFERENCES) auf das Feld 'orte'('Ortplz'). Dadurch wird die Tabelle kunden zur Child-Tabelle, die Tabelle orte zur Parents-Tabelle.

Das Feld Ortplz in der Tabelle kunden kann nur noch Werte annehmen, die in der Parents-Tabelle orte im Feld Ortplz bereits angelegt sind.

**Beispiel:**

Geben sie in die Tabelle kunden die folgenden Werte ein:

Kundenname      Mayer  
Kundenvorname    Hans  
Ortplz            89077.

**Ergebnis:**

Das Datenbanksystem bricht die Aktion mit einer Fehlermeldung ab:



Die Ursache liegt in einer Verletzung der referentiellen Integrität.

Zunächst muss in der Parents-Tabelle orte der Wert 89077 eingetragen und einem Ort-namen zugeordnet werden. Erst dann kann in der Child-Tabelle kunden der Wert 89077 als Ortplz verwendet werden.

Diese Regel der Reihenfolge ist beim Befüllen der Tabellen mit Daten unbedingt zu beachten.

Die Option ON UPDATE CASCADE bewirkt, dass eine Änderung (ON UPDATE) des Primärschlüsselfeldes in der Parents-Tabelle orte weitergegeben wird in die untergeordnete Tabelle kunden. Wird also eine Postleitzahl in der Tabelle orte geändert, so wird diese Änderung auch in der Tabelle kunden durchgeführt. Ob dies sinnvoll ist muss der Designer der Datenbank bereits beim Entwurf berücksichtigen.

Fremdschlüssel dienen der referentiellen Integrität: die Datensätze in verschiedenen Tabellen müssen dauerhaft zueinander passen.

Einem Kunden muss genau eine Postleitzahl zugewiesen werden. Diese Postleitzahl muss in der Datenbank existieren.

Mögliche CONSTRAINT-Bedingungen für Fremdschlüsselfelder sind:

NO ACTION	Alle Änderungen werden verweigert.
CASCADE	Eine Änderung des Wertes in der Primärtabelle wird an die Detailtabelle weitergegeben.
RESTRICT	Eine Änderung wird sowohl in der Primärtabelle als auch in der Detailtabelle verweigert.
SET NULL	Eine Änderung oder Löschung des Wertes in der Primärtabelle bewirkt, dass der Wert in der Detailtabelle auf NULL gesetzt wird.
SET DEFAULT	Eine Änderung oder Löschung des Wertes in der Primärtabelle bewirkt, dass der Wert in der Detailtabelle auf einen Standardwert gesetzt wird.

Diese Bedingungen können sowohl für Änderungsversuche (ON UPDATE) der Parents-Werte als auch für Löscheversuche (ON DELETE) festgelegt werden.

**Beispiel:**

```
CONSTRAINT 'kunden_fk_1' FOREIGN KEY ('Ortplz')
REFERENCES 'orte'('Ortplz')
ON UPDATE CASCADE
ON DELETE RESTRICT)
```

**Ergebnis:**

Das Datenbanksystem gibt eine Änderung (ON UPDATE) eines Wertes Ortplz in der Tabelle orte an alle betroffenen Felder der Tabelle kunden weiter.

Soll ein Wert Ortplz in der Tabelle orte gelöscht werden (ON DELETE) und es existiert in der Tabelle kunden ein Datensatz mit diesem Wert, so wird der Löschvorgang sowohl für die Parents-Tabelle als auch für die Child-Tabelle verweigert.

**Aufgabe:**

Bewerten Sie die Gefahr der folgenden Fremdschlüssel-Deklaration. Was geschieht, wenn versehentlich die Postleitzahl 89077 in der Parents-Tabelle gelöscht wird.

```
CONSTRAINT 'kunden_fk_1' FOREIGN KEY ('Ortplz')
REFERENCES 'orte'('Ortplz')
ON UPDATE CASCADE
ON DELETE CASCADE)
```