

## Definition

Bei der referenziellen Integrität wird sichergestellt, dass für einen Fremdschlüssel nur Werte genommen werden dürfen, die auch als Primärschlüssel in der dazugehörigen Tabelle existieren. Dabei wird sowohl beim Einfügen eines Datensatzes darauf geachtet, dass der referenzierte Wert in der anderen Tabelle vorhanden ist, als auch beim Löschen darauf geachtet wird, dass keine abhängigen Datensätze existieren.

Dies wird im CREATE TABLE Befehl durch die FOREIGN KEY – Klausel erreicht. Die Syntax dazu lautet

## Syntax

```
CREATE TABLE T_Tabellenname (
    Spaltenname1 Datentyp(Größe),
    Spaltenname2 Datentyp(Größe),
    Spaltenname3 Datentyp(Größe),
    ...,
    PRIMARY KEY (Spaltennamen),
    [CONSTRAINT foreignkey_name]
    FOREIGN KEY (Spaltennamen)
    REFERENCES Tabellenname (Spaltennamen)
    [ ON DELETE { NO ACTION | CASCADE | SET NULL | SET DEFAULT } ]
    [ ON UPDATE { NO ACTION | CASCADE | SET NULL | SET DEFAULT } ]
);
```

Mit [ ] sind optionale und mit { } einmalig nutzbare Befehlsteile gekennzeichnet.

Mit den Schlüsselworten ON DELETE und ON UPDATE kann eingestellt werden, wie sich die Attribute der Tabelle verändern, wenn die Ausgangswerte sich ändern. Möglich sind hier NO ACTION, CASCADE, SET NULL oder SET DEFAULT.

Befehl	Aktion
<b>NO ACTION</b>	Der referenzierte Datensatz wird gelöscht. Ungültiger Fremdschlüssel bleibt in der Datenbank. Vorsicht Inkonsistenzen!
<b>CASCADE</b>	Der referenzierte Datensatz wird auch gelöscht. Vorsicht, eventuell gehen Daten verloren!
<b>SET NULL</b>	Fremdschlüsselwert wird auf null gesetzt.
<b>SET DEFAULT</b>	Fremdschlüsselwert wird auf den Standardwert für die Spalte gesetzt. Dies ist datentypabhängig.

Aufgabe: Schreiben Sie die FOREIGN KEY Klausel für die Tabelle T\_Schueler.

T_Lehrer		
P_Lehrer_id	Name	Stundenzahl

T_Schueler				
P_Schueler_id	Vorname	Nachname	Einschulungstage	F_Klassenlehrer