

LU05b - SQL-DDL: Constraints (Einschränkungen)

Lernziele

Sie lernen in dieser Lektion:

- was **Constraints** (Einschränkungen) sind
- warum sie in Datenbanken wichtig sind
- wie Sie die wichtigsten Constraints direkt beim Erstellen einer Tabelle setzen

Was sind Constraints?

Constraints legen Regeln für Spalten in einer Tabelle fest. Sie sorgen dafür, dass **Daten korrekt, vollständig und widerspruchsfrei** gespeichert werden.

Warum brauchen wir Constraints?

- verhindern fehlerhafte oder unvollständige Einträge
- schützen vor doppelten IDs oder leeren Pflichtfeldern
- stellen sicher, dass Daten logisch und konsistent bleiben

Beispiele aus der Praxis:

- Jeder Schüler/jede Schülerin braucht eine eindeutige Schülernummer (ID)
- Der Name eines Kunden darf nicht leer sein
- Die E-Mail-Adresse eines Benutzers soll nur einmal vorkommen

Constraints helfen also, **Datenqualität** sicherzustellen.

NOT NULL

Mit **NOT NULL** legen Sie fest, dass ein Feld **nicht leer** bleiben darf.

<WRAP box 80% round>

```
CREATE TABLE ort (  
  plz CHAR(5) NOT NULL,  
  ort VARCHAR(50) NOT NULL,  
  vorwahl VARCHAR(12)  
);
```

Wirkung: Versucht jemand, ohne PLZ oder Ort einen Datensatz einzufügen, meldet die Datenbank einen Fehler.

PRIMARY KEY - Eindeutige ID

Der **Primärschlüssel (PRIMARY KEY)** identifiziert jeden Datensatz eindeutig. Jede Tabelle kann **nur einen Primärschlüssel** haben.

```
CREATE TABLE ort (  
  plz CHAR(5) NOT NULL,  
  ort VARCHAR(50) NOT NULL,  
  vorwahl VARCHAR(12),  
  CONSTRAINT pk_plz PRIMARY KEY (plz)  
);
```

Wirkung: Keine zwei Datensätze dürfen dieselbe PLZ haben. Die PLZ darf außerdem nicht leer (NULL) sein.

□ AUTO_INCREMENT - Automatisch hochzählen

Wenn es keine natürliche ID gibt, kann man eine **künstliche ID** verwenden, die automatisch hochzählt.

```
CREATE TABLE kunde (  
  kunden_id INT AUTO_INCREMENT,  
  name VARCHAR(50) NOT NULL,  
  email VARCHAR(100),  
  CONSTRAINT pk_kunde PRIMARY KEY (kunden_id)  
);
```

Wirkung: Die Datenbank vergibt bei jedem neuen Datensatz automatisch eine neue Nummer (1, 2, 3, ...)

□ Übersicht

Constraint	Bedeutung
NOT NULL	Spalte darf nicht leer bleiben
PRIMARY KEY	Eindeutige ID für jeden Datensatz
AUTO_INCREMENT	Automatische laufende Nummer für neue Datensätze

[W3Schools: SQL Constraints](#) [W3Schools: PRIMARY KEY](#) [W3Schools: NOT NULL](#)



From:

<https://wiki.bzz.ch/> - **BZZ - Modulwiki**

Permanent link:

https://wiki.bzz.ch/modul/m290_guko/learningunits/lu05/theorie/b_constraints?rev=1757869037

Last update: **2025/09/14 18:57**

