

LU05a - SQL-DDL: Datenbanken & Tabellen anlegen und löschen

Lernziele

Sie lernen in dieser Lektion:

- wie Sie **Datenbanken** anlegen und löschen
- wie Sie darin **Tabellen** mit Spalten erstellen
- wie Sie Tabellen auch wieder **löschen**

Diese Befehle gehören zur **Data Definition Language (DDL)** von SQL.

Datenbanken erstellen und löschen

Mit einer **Datenbank** schaffen Sie einen „Ordner“, in dem Sie Tabellen anlegen können. Hinweis: In **MySQL** sind *Database* und *Schema* Synonyme. In WebStorm (Database-Plugin) lautet der Menüpunkt **New → Schema**, er erstellt damit eine MySQL-Datenbank.

Datenbank erstellen:

```
CREATE DATABASE mydb_perfect;
```

*Bisher haben wir das in WebStorm per Rechtsklick auf die Verbindung → **New** → **Schema** gemacht (siehe Screenshot).*



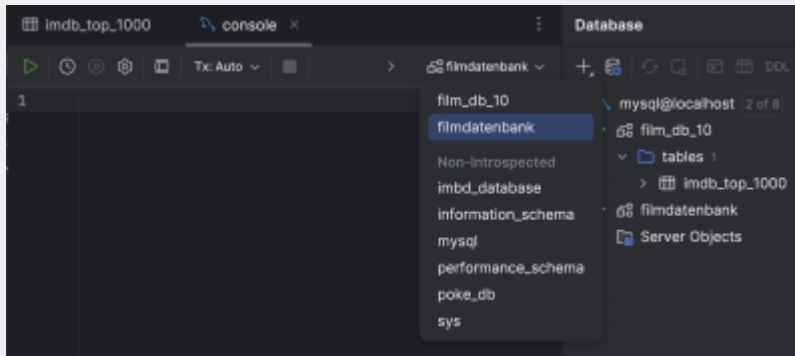
Mit Datenbanken arbeiten (USE) In einem Datenbanksystem (z. B. MySQL) können mehrere **Datenbanken (Schemas)** nebeneinander existieren. Beispiel: Auf demselben MySQL-Server laufen z. B. die Datenbanken *digitech_ch* und *galaxus_ch* - sie liegen im gleichen System, verwalten ihre Daten aber **getrennt**.

Damit man weiss, **in welcher Datenbank man arbeitet**, muss man diese zuerst auswählen. Das geschieht mit dem SQL-Befehl **USE**:

Auf eine Datenbank umschalten (Standard-Datenbank für SQL-Befehle aktiv setzen):

```
USE mydb_perfect;
```

In WebStorm haben wir bisher die Datenbank über das **Drop-Down-Feld <Schema>** oben im Datenbank-Fenster ausgewählt. Der Befehl *USE* macht dasselbe - nur in SQL-Code-Form.



Ab diesem Moment werden alle folgenden SQL-Befehle (z. B. *CREATE TABLE*, *SELECT*, *DROP TABLE*) in dieser Datenbank ausgeführt.

Datenbank löschen:

```
DROP DATABASE mydb_perfect;
```

Achtung: *DROP DATABASE* löscht die **gesamte** Datenbank inkl. **aller** Tabellen und Daten. In produktiven Systemen nur nach Backup verwenden.

Tabellen erstellen

Wie wird nun in eine Datenbank eine Tabelle angelegt? Dazu gibt es in SQL die Anweisung *CREATE*

TABLE, die eine Tabelle mit den dazugehörigen Spalten und deren Datentypen erzeugt. Mit dem Befehl **CREATE TABLE** legen Sie eine Tabelle in der aktuell gewählten Datenbank an.

Allgemeine Syntax:

```
CREATE TABLE tabellenname (  
    spaltenname1 DATENTYP,  
    spaltenname2 DATENTYP,  
    ...  
);
```

Wichtig: nach der letzten Spalte wird kein Kommma mehr geschrieben.

Wichtige Namensregeln:

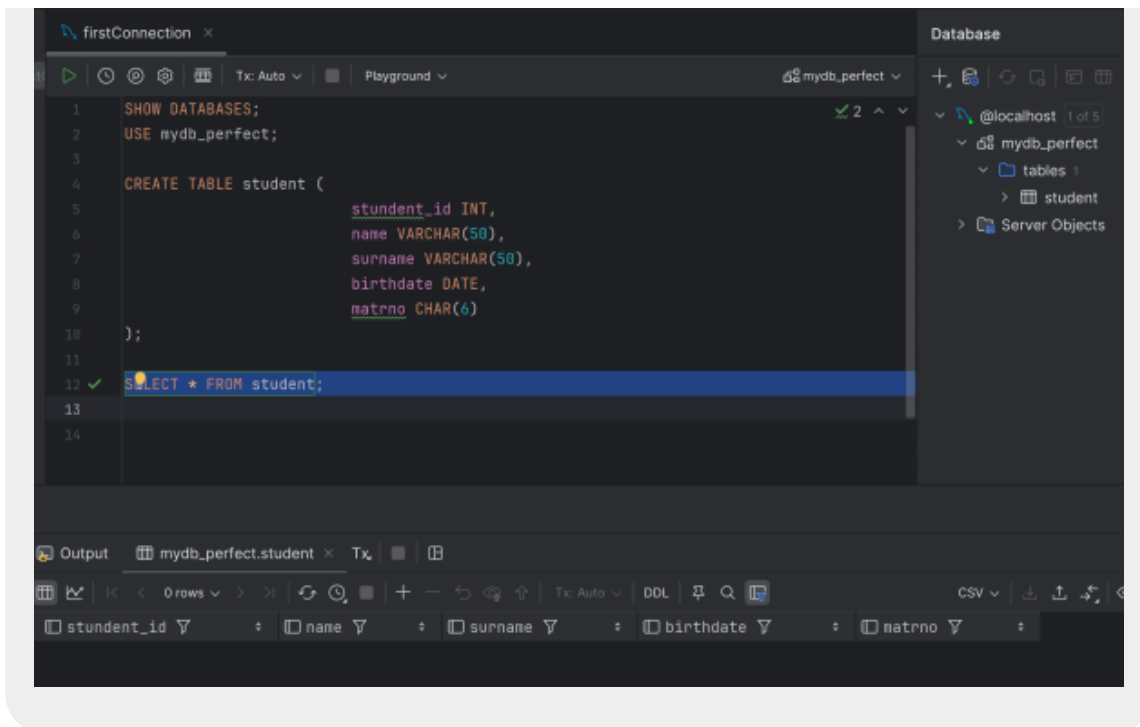
- Keine Leerzeichen, Umlaute oder Sonderzeichen
- Nur Kleinbuchstaben
- Tabellennamen müssen eindeutig sein

Beispiel: Tabelle Student

Wir wollen eine Tabelle **student** (Lernende:r) erstellen, die eine ID, den Vornamen, den Nachnamen und das Geburtsdatum speichert:

```
CREATE TABLE student (  
    student_id INT,  
    name VARCHAR(50),  
    surname VARCHAR(50),  
    birthdate DATE  
);
```

Nach Ausführung dieses DDL-Befehls würde das Ergebnis ähnlich wie in der folgenden Abbildung dargestellt aussehen.



Erklärung:

- *INT* → ganze Zahl (1..50 etc)
- *VARCHAR(50)* → Zeichenkette variabler Länge (max. 50)
- *DATE* → Datum in YYYY-MM-DD

Tabellen löschen

Mit dem Befehl **DROP TABLE** löschen Sie eine komplette Tabelle (inklusive aller Daten):

```
DROP TABLE student;
```



Achtung: Dieser Befehl kann **nicht rückgängig gemacht** werden und dieser Befehl löscht auch alle Daten aus der Tabelle!

Merksatz

SQL-Befehl	Bedeutung
CREATE DATABASE	Neue Datenbank erstellen
USE DATABASE	Datenbank „aktiv“ schalten
DROP DATABASE	Datenbank löschen

SQL-Befehl	Bedeutung
CREATE TABLE	Neue Tabelle in der aktiven Datenbank erstellen
DROP TABLE	Tabelle aus der aktiven Datenbank löschen



From:

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

Permanent link:

https://wiki.bzz.ch/modul/m290_guko/learningunits/lu05/theorie/a_ddl?rev=1758189041

Last update: **2025/09/18 11:50**

