

Die verschiedenen SQL - Statements

Privileg	Beschreibung
ALL	Alle Privilegien
ALTER	Die Struktur der Tabelle ändern
CREATE	Neue Datenbanken oder Tabellen anlegen
DELETE	Datensätze in einer Tabelle löschen
DROP	Datenbanken oder Tabellen löschen
FILE	Dateien auf dem Server lesen und schreiben
GRANT	Die Rechte von MySQL-Konten ändern
INSERT	Neue Datensätze/Zeilen in eine Tabelle einfügen
SELECT	Daten einer Tabelle auswählen
SHUTDOWN	Den MySQL-Server herunterfahren
UPDATE	Daten aktualisieren/ändern

Datenbanken, Tabellen

Datenbank erzeugen **CREATE database firma;**
 Datenbank benutzen **USE firma;**
 Datenbank löschen **DROP database firma;**
 Tabelle erzeugen **CREATE table personen(name char (30), vorname char (25), ..);**

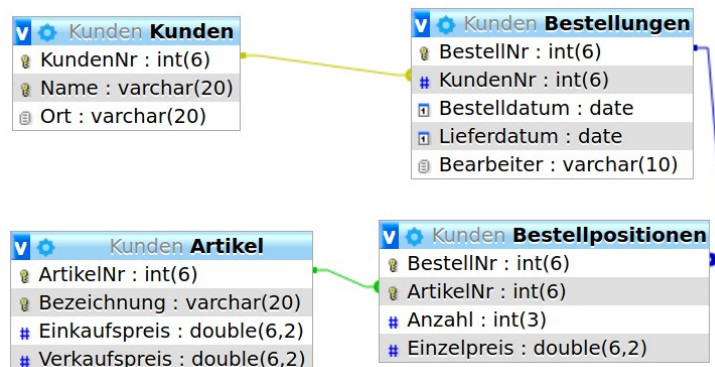
Beispiel:

```
CREATE TABLE Artikel
(ArtikelNr      INT(6) auto_increment,
Bezeichnung    VARCHAR(20) NOT NULL UNIQUE BINARY,
Einkaufspreis  DOUBLE(6,2),
Verkaufspreis  DOUBLE(6,2),
PRIMARY KEY (ArtikelNr));
```

Groß- und Kleinschreibung wird beachtet

```
CREATE TABLE Bestellpositionen
(BestellNr      INT(6),
ArtikelNr       INT(6),
Anzahl          INT(3),
Einzelpreis     DOUBLE(6,2),
PRIMARY KEY (BestellNr, ArtikelNr),
FOREIGN KEY (BestellNr) REFERENCES Bestellungen(BestellNr),
FOREIGN KEY (ArtikelNr) REFERENCES Artikel(ArtikelNr) );
```

Datenbanktyp muss
MyISAM sein!



Struktur ändern

Tabellenname ändern	<i>alter table personen rename person;</i>
Feldname ändern	<i>alter table personen change vorname name char(25);</i>
Feld hinzufügen	<i>alter table personen add gehalt double;</i>
Typ eines Feldes ändern	<i>alter table personen change gehalt gehalt int;</i>
Feld löschen	<i>alter table personen drop gehalt;</i>

Daten einfügen (ein Datensatz)

Datensätze erzeugen (Reihenfolge der Felder wird beachtet)	<i>INSERT personen VALUES ('Maier',21221, '1960-12-11');</i>
Datensätze in bestimmten Feldern erzeugen (Reihenfolge egal)	<i>INSERT personen (name, p_nr) VALUES ('Schmitz', 355690);</i>

Daten einfügen (mehrere Datensätze)

Die Datensätze der Tabelle *MeineKontakte* sollen nach *Kontakte* übertragen werden.

INSERT INTO Kontakte SELECT * FROM MeineKontakte;

Datensätze ändern

Das Feld 'gehalt' auf 3000 setzen, von Tim Müller	UPDATE personen SET gehalt = 3000 WHERE name = 'Müller' AND vorname='Tim';
Das Feld 'gehalt' von Tim Müller um 10% erhöhen	UPDATE personen SET gehalt = gehalt * 1.1 WHERE name = 'Müller' AND vorname='Tim';
Mehrere Änderungen gleichzeitig	UPDATE personen SET gehalt = 3000 , name = 'Mueller' , vorname = 'Timo' WHERE name = 'Müller' AND vorname='Tim';

Das Datum in der gesamten Tabelle verändern:

***UPDATE bestellung SET
bestellung.LIEFERDATUM = date_add(bestellung.LIEFERDATUM, INTERVAL 4 year),
bestellung.BESTELLDATUM = date_add(bestellung.BESTELLDATUM, INTERVAL 4 year)***

Datensätze löschen

Datensätze löschen, wo das Feld 'gehalt' leer ist

DELETE FROM personen WHERE gehalt is null OR gehalt = ' ';

Bestimmte Datensätze löschen

DELETE FROM personen WHERE gehalt = 500;

UNION-Abfragen

Alle Datensätze der Tabellen personen und hobby sollen in einer Tabelle angezeigt werden:

select * from personen union select * from hobby;

Übung

Importieren Sie die Datei **hdd.sql**

festplattenNr	hersteller	typ	gb	preis	artnummer	prod
1	Quantum	Fireball CX	500	58.00	HDA-208	2024-03-14
2	IBM	DJNA 372200	250	49.00	HD-140	2024-03-14
3	Seagate	310232A	800	49.99	HDA-144	2024-03-04
4	Quantum	Fireball Plus	2000	65.00	HDA-163	2024-03-10
5	Fujitsu	MPE 3136	350	38.00	HDA-171	2024-03-02
6	Quantum	310232A	1000	45.00	HDA-208	2024-03-05
7	WD	WD10EZEX	1000	37.66		2024-12-03

Die Datenbank soll über die MySQL-Konsole administriert und abgefragt werden.

Gehen Sie über die Eingabeaufforderung in den 'bin-Ordner' von MySQL und melden Sie sich auf den MySQL-Server als Administrator mit folgenden Verbindungsparametern an: **mysql -h localhost -u root -p** → Enter; bei der Passwortabfrage Enter.

cd.. → in den übergeordneten Ordner wechseln

cd xampp → in den untergeordneten Ordner *xampp* wechseln

```
C:\>cd xampp
```

```
C:\xampp>cd mysql
```

```
C:\xampp\mysql>cd bin
```

```
C:\xampp\mysql\bin>mysql -h localhost -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with
```

Hinweis: Wenn eine Datenbank mit SQL-Befehlen bearbeitet werden soll, muss zuvor mit dem **use**-Befehl die entsprechende Datenbank ausgewählt werden.

1) Werte ausgeben

- Lassen Sie alle Festplatten mit allen Angaben anzeigen, deren Typbezeichnung mit 'Fire' beginnen.
- Lassen Sie alle Festplatten vom Herstellers Quantum, mit allen Angaben anzeigen.
- Fügen Sie einen 8. Datensatz in die obige Tabelle ein s.u..

'WD', 'WD Blue 3D', 250, 49.00, 'HD-022', '2024-03-19'

- Lassen Sie alle Festplatten mit allen Angaben anzeigen mit einer Kapazität von mehr als 250 GB, die weniger als 85 € kosten, mit allen Angaben, nach GB absteigend sortiert

2) Werte in der Tabelle ändern

- a) Alle Festplatten des Herstellers Seagate sollen um 25 Euro teurer werden.
- b) Der Hersteller Fujitsu hat seine Festplatte MPE 3136 ersetzt durch die Festplatte MPE 3139 mit einer Kapazität von 500 GB. Führen Sie die notwendige Änderung durch.
- c) Alle Festplatten des Herstellers Quantum sollen um 8% teurer werden.
- d) Der Hersteller IBM hat seine Produktion eingestellt. Entfernen Sie die betreffenden Festplatten aus dem Angebot.
- e) Löschen Sie alle versehentlich eingetragenen Datensätze. Das Kriterium für einen solchen Datensatz soll sein: Der Datensatz hat keine Artikelnummer.

3) Neue Tabelle

- a) Erstellen Sie die neue Tabelle *hersteller* mit den Feldern:
HerstellerNr, Name, Ort in allen Feldern muss beim Einfügen eines neuen Datensatzes ein Wert eingetragen werden. HerstellerNr soll das Primärschlüsselfeld und AUTO_INCREMENT sein.
Erstellen Sie das SQL_-Statement.
- b) Fügen Sie die Datensätze von 2 Herstellern in die neue Tabelle ein (der Ort ist beliebig).
Erstellen Sie die SQL-Statements.

4) Transaktionen

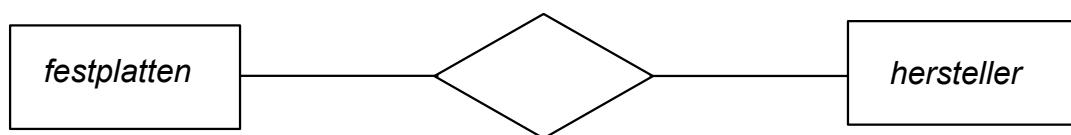
Bei Datenbanktransaktionen geht es darum, dass mehrere Operationen entweder vollständig und fehlerfrei ausgeführt werden oder gar nicht. Auf keinen Fall darf nur ein Teil einer Transaktion durchgeführt werden, also z. B. die ersten beiden von insgesamt fünf Kommandos.

```
1 BEGIN;  
2 insert into hersteller VALUES (NULL, 'IBM', 'Boston');  
3 insert into hersteller VALUES (NULL, 'WD', 'London');  
4 COMMIT;
```

COMMIT schließt die Transaktion ab.

- a) Fügen Sie die restlichen Hersteller aus der Tabelle *festplatten* in die neue Tabelle über eine Transaktion ein.
Erstellen Sie das SQL-Statement.

5) Tabellenverknüpfung



- a) Benennen Sie das Sekundärschlüsselfeld der Tabelle *festplatten* genauso wie das Primärschlüsselfeld der Tabelle *hersteller*.
- b) Fügen Sie mit einer Transaktion den passenden Wert in das Sekundärschlüsselfeld

```
1 BEGIN;  
2 SELECT @ID:= hersteller.HerstellerNr FROM hersteller WHERE hersteller.Name='WD';  
3 UPDATE festplatten SET festplatten.hersteller = @ID WHERE festplatten.hersteller = 'WD';  
4 COMMIT;
```

In der Variablen ID wird der Primärschlüssel eines Herstellers (aus der Tabelle *hersteller*) zwischen gespeichert.

- c) Passen Sie den Datentyp des Sekundärschlüsselfeldes an.