

MySQL - Datenmanipulation

Stefan Maihack Dipl. Ing. (FH)

Datum: 6.4.2015

Datenmanipulation

- Wie erzeugt man eine Datenbank?
- Wie entfernt man eine Datenbank?
- Wie erzeugt man Tabellen und deklariert die Spalten?
- Wie entfernt man Tabelleninhalte und wie entfernt man ganze Tabellen?
- Wie beschreibt man eine Tabelle?
- Wie ändert man Spalten und Indizes ändert?

Datenmanipulation

Erzeugen und Löschen einer Datenbank

- Die folgenden Beispiele funktionieren nur innerhalb des Client-Programms „mysql“

- Erzeugen einer Datenbank:

```
CREATE DATABASE <Name der Datenbank>;
```

- Löschen einer Datenbank:

```
DROP DATABASE <Name der Datenbank>;
```

Datenmanipulation

Tabellen manipulieren

- Eine Tabelle erzeugen:

```
CREATE [TEMPORARY] TABLE [IF NOT EXISTS] <Tabellenname>  
[(Erzeugungsdefinition,...)]  
[Tabellenoptionen] [Auswahanweisung]
```

Beispiel:

```
mysql > CREATE TABLE artikel IF NOT EXISTS (  
-> artikel_id INT(9) NOT NULL auto_increment,  
-> headline TEXT NOT NULL,  
-> date_post DATETIME NOT NULL DEFAULT '0000-00-00 00:00:00',  
-> text_body TEXT,  
-> text_summary TEXT,  
-> who_created INT(9) DEFAULT NULL,  
-> email_send INT(1) NOT NULL DEFAULT '0',  
-> date_email DATETIME DEFAULT NULL,  
-> who_approved INT(9) DEFAULT NULL,  
-> pic VARCHAR(255) DEFAULT NULL,  
-> PRIMARY KEY (artikel_id)  
-> );
```

Beispiel: Anlegen einer temporären Tabelle

```
CREATE TEMPORARY TABLE tmp (name VARCHAR(100) NULL);
```

Datenmanipulation

Tabellen manipulieren

- Eine temporäre Tabelle aus einer SELECT-Anweisung heraus anlegen:

```
CREATE TEMPORARY TABLE temp
SELECT abonenten_id, name
FROM abonenten
WHERE name LIKE 'John%';
```

➔ Die temporäre Tabelle bezieht Spaltennamen und Deklarationen aus der SELECT-Abfrage

- Anderer Spaltenname verwenden über Alias:

```
CREATE TEMPORARY TABLE temp
SELECT abonenten_id AS id, name AS vorname
FROM abonenten
WHERE name LIKE 'John%';
```

Datenmanipulation

Tabellen entfernen und ändern

Folgende Aktionen an Tabellen sind möglich

Aktionssyntax	Ausgeführte Aktion
ALTER TABLE <tabellenname> ADD [COLUMN] <Spaltendeklaration> [FIRST AFTER <Spaltenname>]	Fügt eine Spalte in eine Tabelle ein.
ALTER [COLUMN] <Spaltenname> [SET DEFAULT <Literal> DROP DEFAULT]	Spezifiziert den neuen Standardwert für eine Spalte oder entfernt den alten Standardwert.
ALTER TABLE <tabellenname> CHANGE [COLUMN] <Spaltenname> <Spaltendeklaration>	Modifiziert die Spaltendeklaration.
ALTER TABLE <tabellenname> MODIFY [COLUMN] <Spaltendeklaration>	Modifiziert eine Spaltendeklaration, ohne die Spalte umzubenennen.
ALTER TABLE <tabellenname> DROP [COLUMN] <Spaltenname>	Entfernt eine Spalte.
ALTER TABLE <tabellenname> RENAME [AS] <NeuerTabellenname>	Bennent die Tabelle um.
DROP TABLE <Tabellenname>	Löscht die gesamte Tabelle inkl. Inhalt und Definition.

Datenmanipulation

Tabellen befüllen mit INSERT

- Eine Tabelle mit Daten befüllen

```
INSERT INTO <tabellenname>  
    (feld2, feld3, feld4)  
VALUES  
    (,inhalt2', ,inhalt3', ,inhalt4');
```

- Alle Feldinhalte vom Typ String kommen in einfache Anführungsstriche.
Die automatische inkrementierte ID wird nicht berücksichtigt (oder erhält den Wert NULL (ohne Anführungsstriche)).

Datenmanipulation

befüllen mit INSERT

- Tabellen einfügen mit AUTO_INCREMENT
DROP DATABASE db_animals;
CREATE DATABASE db_animals;
USE db_animals;
CREATE TABLE animals (
 id MEDIUMINT NOT NULL AUTO_INCREMENT,
 name CHAR(30) NOT NULL,
 PRIMARY KEY (id)
);

INSERT INTO animals (name) VALUES
 (,dog'),
 (,cat'),
 (,penguin'),
 (,lax'),
 (,whale'),
 (,ostrich');

SELECT * FROM animals;

Id	name
1	dog
2	cat
3	penguin
4	lax
5	whale
6	ostrich

Übung 1

- Legen sie folgende Tabelle „abonnetten“ an. Schreiben sie dazu am besten eine SQL-Loaderdatei

Spaltenname1: abonnetten_id	Type: int(9)	Key: ja (mit auto_increment)
Spaltenname2: name	Type: varchar(100)	Key: nein
Spaltenname3: email	Type: varchar(100)	Key: nein
Spaltenname4: geburtsdatum	Type: datetime	Key: nein

- Einfügen der zusätzlichen Spalte „geschlecht“
- Löschen Sie die Spalte „geschlecht“
- Modifizieren der Spalte „email“ von 100 auf 200 Zeichen
- Lösungen:
- ```
CREATE TABLE geburtsdaten (
 abonnetten_id int(9),
 name VARCHAR(100),
 email VARCHAR(100),
 geburtsdatum datetime,
 PRIMARY KEY(abonnetten_id)
);
```
- ```
ALTER TABLE abonnetten ADD COLUMN geschlecht varchar(10);
```
- ```
ALTER TABLE abonnetten DROP COLUMN geschlecht;
```
- ```
ALTER TABLE abonnetten MODIFY email VARCHAR(200);
```

Übung 2

- Legen Sie die folgende Tabelle in einer Datenbank an.

1 .Spalte: Nummer

2. Spalte: Name

- Fügen Sie folgende Werte ein:

Nummer	Name
1	Meier
2	Müller
3	Brendel
4	Adorf
5	Klein
6	Groß

- Fügen Sie die weitere Spalte „Vorname“ ein und füllen Sie die Spalte mit Werten.
- Schreiben Sie eine SQL-Datei, die automatisch eine Datenbank und eine Tabelle anlegt und diese mit den oben genannten Werte füllt.