

SQL-Befehlsübersicht

DDL (Data Definition Language)

Datenbanken anlegen, löschen, auswählen:

SQL-Befehl	Beschreibung	Beispiel
<pre>CREATE DATABASE db_name</pre>	Erstelle die Datenbank db_name	<pre>CREATE DATABASE autovermietung</pre>
<pre>DROP DATABASE db_name</pre>	Lösche die Datenbank db_name	<pre>DROP DATABASE autovermietung</pre>
<pre>USE db_name</pre>	Auswahl der Datenbank, auf welche die folgenden SQL-Befehle ausgeführt werden sollen	<pre>USE autovermietung</pre>

Tabellen und Indizes erzeugen, ändern und löschen:

SQL-Befehl	Beschreibung	Beispiel
<pre>CREATE TABLE IF NOT EXISTS tb_name</pre>	Erstelle die Tabelle tb_name falls diese noch nicht existiert	<pre>CREATE TABLE kunden (kd_id INT NOT NULL AUTO_INCREMENT PRIMARY KEY, kd_name VARCHAR(30) NOT NULL);</pre>
<pre>CREATE INDEX ... ON ... CREATE UNIQUE INDEX ... ON ...</pre>	Erstelle einen eindeutigen Index	<pre>CREATE INDEX kunden_name ON kunde (kd_name);</pre>

<pre>ALTER TABLE ... ADD <field> ADD INDEX ADD PRIMARY KEY ADD UNIQUE RENAME AS MODIFY CHANGE DROP COLUMN</pre>	<p>Ändere die Struktur der Tabelle, etwa durch Hinzufügen eines Index (z.B. zum Durchsuchen), Primärschlüssels, usw.</p>	<pre>ALTER TABLE kunden ADD INDEX kunden_name (kd_name); ALTER TABLE kunden RENAME AS kunde; ALTER TABLE kunden MODIFY kd_name varchar(50); ALTER TABLE kunden CHANGE kd_nname name varchar(30); ALTER TABLE kunde ADD mobile varchar(20);</pre>
<pre>DROP TABLE ...</pre>	<p>Lösche Tabelle aus DB</p>	<pre>DROP TABLE kunden;</pre>
<pre>DROP INDEX ...</pre>	<p>Entferne den Index</p>	<pre>DROP INDEX kunden_name;</pre>

DML (Data Manipulation Language)

SQL-Befehl	Beschreibung	Beispiel
<pre>INSERT INTO ... VALUES ...</pre>	<p>Einfügen eines oder mehrerer Datensätze. Die Reihenfolge der Werte richtet sich nach der Reihenfolge der angegebenen Attribute</p>	<pre>INSERT INTO kunden (kd_id, kd_name, kd_vname) VALUE (NULL, "Mustermann", "Max"), (NULL, "Musterfrau", "Frieda");</pre>
<pre>UPDATE ... SET ... WHERE ...</pre>	<p>Ändern von Werten eines oder mehrerer Datensätze</p>	<pre>UPDATE kunden SET kd_nname = "Mustermann" WHERE kd_id=42;</pre>

<pre>DELETE ... FROM ...</pre>	Löschen eines Datensatzes	<pre>DELETE FROM kunden WHERE kd_name = "Mustermann";</pre>
--------------------------------	---------------------------	---

DQL (Data Query Language)

SQL-Befehl	Beschreibung	Beispiel
<pre>SELECT ... FROM ... WHERE ... GROUP BY ... ORDER BY ... HAVING ... LIMIT ...</pre>		<pre>SELECT * FROM kunden; SELECT kd_ort, COUNT(kd_ort) FROM kunden GROUP BY kd_ort; SELECT COUNT(*) FROM kunden WHERE kd_ort = "Hof"; SELECT * FROM kunden ORDER BY kd_name;</pre>
<pre>SELECT ... AS ...</pre>	Dem Feldnamen (Attribut) wird ein anderer Name gegeben (= Alias)	<pre>SELECT kd_id AS Kundennummer FROM kunden;</pre>

Funktionen

SQL-Befehl	Beschreibung	Beispiel
<pre>COUNT(ausdruck)</pre>	Anzahl aller Datensätze	<pre>SELECT COUNT(*) FROM kunden;</pre>
<pre>SUM(ausdruck)</pre>	Summe	<pre>SELECT SUM(po_preis) FROM posten;</pre>

MAX(ausdruck)	Maximum	SELECT MAX(po_preis) FROM posten;
MIN(ausdruck)	Minimum	
AVG(ausdruck)	Durchschnitt	
ABS(ausdruck)	Betrag	
NOW()	Gibt das aktuelle Datum mit Uhrzeit im Format "JJJJ-MM-TT hh:mm:ss" zurück	
YEAR(date) MONTH(date) DAY(date) HOUR(date) MINUTE(date) SECOND(date)	Geben die aktuellen Zeit-Werte dem Namen nach zurück	SELECT * FROM kunden WHERE YEAR(gebdat) > 2000
DATEDIFF(datum1, datum2)	Berechnet die Differenz der Tage datum1 - datum2 mit Vorzeichen	

Informationen aus verschiedenen Tabelle holen (JOIN)

SQL-Befehl	Beschreibung	Beispiel
------------	--------------	----------

<pre>SELECT ... FROM ... JOIN ... ON ...</pre>	<p>Schnittmenge, d.h. nur Kunden mit zugehörigen Aufträgen</p>	<pre>SELECT * FROM kunden JOIN auftraege ON kd_id = au_kd_id;</pre>
<pre>LEFT JOIN ... ON ...</pre>	<p>Alle Kunden (auch die ohne Auftrag!) und zugehörige Aufträge</p>	<pre>SELECT * FROM kunden LEFT JOIN auftraege ON kd_id = au_kd_id;</pre>

Operatoren

SQL-Befehl	Beschreibung	Beispiel
<pre>>, <, >=, =, !=</pre>	<p>Boolsche Operatoren zum Vergleich zweier Werte</p>	<pre>SELECT * FROM kunden WHERE kd_or t = "Hof";</pre>
<pre>IN, NOT IN</pre>	<p>Prüfung, ob sich ein Attribut innerhalb einer Untermenge befindet. Häufig verwendet bei Sub-Selects</p>	<pre>SELECT * FROM kunden WHERE kd_id IN (1, 3, 5);</pre>
<pre>BETWEEN ... AND ...</pre>	<p>Prüfung eines Attributs, ob es sich zwischen den beiden Werten befindet</p>	<pre>SELECT * FROM kunden WHERE kd_id BETWEEN 10 AND 20;</pre>
<pre>LIKE</pre>	<p>Prüfung auf ähnliche Strings. Mit dem Platzhalter % werden im Beispiel Musterfrau und Mustermann erkannt</p>	<pre>SELECT * FROM kunden WHERE kd_nname LIKE "Muster%";</pre>
<pre>AND, OR, NOT</pre>	<p>Verknüpfung von boolschen Ausdrücken über UND, ODER und NICHT</p>	<pre>SELECT * FROM kunden WHERE kd_name LIKE "%ust%" AND kd_ka_id = 1;</pre>

Revision #17

Created 5 June 2024 19:20:57 by Niklas

Updated 18 June 2024 19:09:05 by Niklas