Cheatography

MySQL Cheat Sheet

by Sunnii via cheatography.com/174791/cs/36720/

SELECT

Alles auswählen	**SELECT * FROM** *table_name* **WHERE** *column1* = *value1*;
Spezifische Spalten auswählen	**SELECT** *column1*, *column2*, ... **FROM** *table_name* **WHERE** *column1* = *value1*

WHERE

Spezifische Condition	**SELECT** *column1*, *column2* **FROM** *tablename* **WHERE** *condition1* **AND** *condition2* **OR** *condition3*
Filtert nach Values, die in der Spalte enthalten sind	**SELECT** *column1* **FROM** *tablename* **WHERE** *column1* **IN**(*value1*, *value2*, *value3*)

WHERE IS

Select Werte Null	**SELECT * FROM** *table_name* **WHERE** *column1* **IS** NULL
Select Werte, die nicht NULL sind	**SELECT * FROM** *table_name* **WHERE** *column1* **IS NOT** NULL

Alias AS

Alias für Spaltennamen	**SELECT** *column_name* **AS** *alias_name* **FROM** *table_name*;
Alias für Tabellenname	**SELECT** *column_name* **FROM** *table_name* **AS** *alias_name*;

CONCAT

Zusammensetzen von Werten	**SELECT CONCAT**(*column1*, *value*, ...) **FROM** *table_name* **WHERE** *condition*

Beispiel:
SELECT CONCAT(`lastname`, '', '', `firstname`) AS Fullname FROM `students` WHERE 1 ORDER BY Fullname ASC;

MIN

Kleinster Wert	**SELECT MIN**(*column1*) **FROM** *table_name*

MAX

Größter Wert	**SELECT MAX**(*column1*) **FROM** *table_name*

AVG

Durschnittswert	**SELECT AVG**(*column1*) **FROM** *table_name*

COUNT

Anzahl alles	**SELECT COUNT**(*) **FROM** *table_name* **WHERE** *condition*;
Anzahl spezifische Spalte	**SELECT COUNT**(*column_name*) **FROM** *table_name* **WHERE** *condition*;
Gib Spalten und Anzahl aus	**SELECT** *column1*, **COUNT**(*column1*) **FROM** *table_name* **WHERE** *condition* **GROUP BY** *column1*;

DISTINCT

Keine doppelten Werte für alle Einträge	**Select Distinct** * **FROM** *tablename*
Keine doppelten Werte aus column1	**Select Distinct** *column1* **FROM** *tablename*
Zähle unterschiedliche Werte aus column1	**Select COUNT(Distinct** *column1*) **FROM** *tablename*

GROUP BY / ORDER BY

Gruppiere und ordne Werte	**SELECT** *column_name* **FROM** *table_name* **WHERE** *condition* **GROUP BY** *column_name* **ORDER BY** *column_name* **ASC**|**DESC**;

LIKE

Filter nach Pattern	**SELECT * FROM** *table_name* **WHERE** *column3* **LIKE** *pattern*;
Filter nach Beginn eines Values	**SELECT * FROM** *table_name* **WHERE** *column1* **LIKE** "*value%*";
Filter nach Beginn eines Values mit einem weiteren Zeichen	**SELECT * FROM** *table_name* **WHERE** *column1* **LIKE** "*value_*";

Platzhalter `%` steht für 0, 1 oder mehrere Werte
Platzhalter _ steht für 1 Wert

HAVING

Nachgelagerte Condition	**SELECT** *column_name* **FROM** *table_name* **WHERE** *condition* **GROUP BY** *column_name* **HAVING** *condition*;

LIMIT

Einschränkung der angezeigten Werte	**SELECT** *column_name* **FROM** *table_name* **WHERE** *condition* **LIMIT** *number_skipped*, *number_shown*;

EXISTS

Prüft Existenz und gibt wahr/falsch zurück	**SELECT** *column_name* **FROM** *table_name* **WHERE EXISTS** (**SELECT** *column_name* **FROM** *table_name* **WHERE** *condition*);
	SELECT *column_name* **FROM** *table_name* **WHERE NOT EXISTS** (**SELECT** *column_name* **FROM** *table_name* **WHERE** *condition*);

INSERT INTO

Werte einfügen	**INSERT INTO** *table_name* (*column1*, *column2*, *column3*, ...) **VALUES** (*value1*, *value2*, *value3*, ...);

*

UPDATE

Update Tabelle	**UPDATE** *table_name* **SET** *column1* = *value1*, *column2* = *value2*, ... **WHERE** *condition*;

DELETE

Alles löschen	**DELETE FROM** *table_name*;
Löschen mit Condition	**DELETE FROM** *table_name* **WHERE** *condition*;

Häufig verwendete Datentypen

VARCHAR(M)	String (0 - 255)	L + 1 Bytes
TINYINT	Integer (-128 - 127)	1 Byte
INT	Integer (-2147-483648 to 2147483647)	4 Bytes
BIGINT	Integer (-9223372-036854775808 to 922337203685-4775807)	8 Bytes
BLOB	String (0-65535)	L + 2 Bytes

L: byte length of the string
M represents the declared column length in characters
unsigned: nur positive Zahlen, dafür größere Zahlen möglich