(My)SQL Cheat Sheet

Here are the most commonly used SQL commands and the most commonly used options for each. There are many more commands and options than listed here. In other words, the syntaxes as I have listed them are far from complete. See the links at the bottom for more complete syntaxes and more commands.

Common MySQL Column Types

Purpose	Data Type	Example
Integers	int(*M*)	int(5)
Floating-point (real) numbers	float(*M*,*D*)	float(12,3)
Double-precision floating-point	double(*M*,*D*)	double(20,3)
Dates and times	timestamp(*M*)	timestamp(8) (for YYYYMMDD) timestamp(12) (for YYYYMMDDHHMMSS)
Fixed-length strings	char(*M*)	char(10)
Variable-length strings	varchar(*M*)	varchar(20)
A large amount of text	blob	blob
Values chosen from a list	enum('value1',value2',...)	enum('apples','oranges','bananas')

M is maximum to display, and *D* is precision to the right of the decimal.

MySQL Mathematical Functions

What	How
Count rows per group	COUNT(*column* | *)
Average value of group	AVG(*column*)
Minumum value of group	MIN(*column*)
Maximum value of group	MAX(*column*)
Sum values in a group	SUM(*column*)
Absolute value	abs(*number*)
Rounding numbers	round(*number*)
Largest integer not greater	floor(*number*)
Smallest integer not smaller	ceiling(*number*)
Square root	sqrt(*number*)
*n*th power	pow(*base*,*exponent*)
random number *n*, 0<*n* < 1	rand()
sin (similar cos, etc.)	sin(*number*)

MySQL String Functions

What	How
Compare strings	strcmp(*string1*,*string2*)
Convert to lower case	lower(*string*)
Convert to upper case	upper(*string*)
Left-trim whitespace (similar right)	ltrim(*string*)
Substring of string	substring(*string*,*index1*,*index2*)
Encrypt password	password(*string*)
Encode string	encode(*string*,*key*)
Decode string	decode(*string*,*key*)
Get date	curdate()
Get time	curtime()
Extract day name from date string	dayname(*string*)
Extract day number from date string	dayofweek(*string*)
Extract month from date string	monthname(*string*)

Basic MySQL Commands

What	How	Example(s)
List all databases	SHOW DATABASES;	SHOW DATABASES;
Create database	CREATE DATABASE *database*;	CREATE DATABASE PhoneDB;
Use a database	USE *database*;	USE PhonDB;
List tables in the database	SHOW TABLES;	SHOW TABLES;
Show the structure of a table	DESCRIBE *table*; SHOW COLUMNS FROM *table*;	DESCRIBE Animals; SHOW COLUMNS FROM Animals;
Delete a database (*Careful!*)	DROP DATABASE *database*;	DROP DATABASE PhoneDB;

SQL Commands: Modifying

What	How	Example(s)
Create table	CREATE TABLE *table* (*column1 type* [[NOT] NULL] [AUTO_INCREMENT], *column2 type* [[NOT] NULL] [AUTO_INCREMENT], ... *other options*, PRIMARY KEY (*column(s)*));	CREATE TABLE Students (LastName varchar(30) NOT NULL, FirstName varchar(30) NOT NULL, StudentID int NOT NULL, Major varchar(20), Dorm varchar(20), PRIMARY KEY (StudentID));
Insert data	INSERT INTO *table* VALUES (*list of values*); INSERT INTO *table* SET *column1=value1*, *column2=value2*, ... *columnk=valuek*; INSERT INTO *table* (*column1*,*column2*,...) VALUES (*value1*,*value2*...);	INSERT INTO Students VALUES ('Smith','John',123456789,'Math','Selleck'); INSERT INTO Students SET FirstName='John', LastName='Smith', StudentID=123456789, Major='Math'; INSERT INTO Students (StudentID,FirstName,LastName) VALUES (123456789,'John','Smith');
Insert/Select	INSERT INTO *table* (*column1*,*column2*,...) *SELECT statement*; (See below)	INSERT INTO Students (StudentID,FirstName,LastName) SELECT StudentID,FirstName,LastName FROM OtherStudentTable; WHERE LastName like '%son';
Delete data	DELETE FROM *table* [WHERE *condition(s)*]; (Omit **WHERE** to delete all data)	DELETE FROM Students WHERE LastName='Smith'; DELETE FROM Students WHERE LastName like '%Smith%'; AND FirstName='John'; DELETE FROM Students;
Updating Data	UPDATE *table* SET *column1=value1*, *column2=value2*, ... *columnk=valuek* [WHERE *condition(s)*];	UPDATE Students SET LastName='Jones' WHERE StudentID=987654321; UPDATE Students SET LastName='Jones', Major='Theatre' WHERE StudentID=987654321 OR (MAJOR='Art' AND FirstName='Pete');
Insert column	ALTER TABLE *table* ADD COLUMN	ALTER TABLE Students ADD COLUMN

	column type options;	Hometown varchar(20);
Delete column	ALTER TABLE *table* DROP COLUMN *column*;	ALTER TABLE Students DROP COLUMN Dorm;
Delete table (*Careful!*)	DROP TABLE [IF EXISTS] *table*;	DROP TABLE Animals;

SQL Commands: Querying

What	**How**	**Example(s)**
All columns	SELECT * FROM *table*;	SELECT * FROM Students;
Some columns	SELECT *column1*,*column2*,... FROM *table*;	SELECT LastName, FirstName FROM Students;
Some rows/ columns	SELECT *column1*,*column2*,... FROM *table* [WHERE *condition(s)*];	SELECT LastName,FirstName FROM Students WHERE StudentID LIKE '%123%';
No Repeats	SELECT [DISTINCT] *column(s)* FROM *table*;	SELECT DISTINCT LastName FROM Students;
Ordering	SELECT *column1*,*column2*,... FROM *table* [ORDER BY *column(s)* [DESC]];	SELECT LastName,FirstName FROM Students ORDER BY LastName, FirstName DESC;
Column Aliases	SELECT *column1* [AS *alias1*], *column2* [AS *alias2*], ... FROM *table1*;	SELECT LastName,FirstName AS First FROM Students;
Grouping	SELECT *column1*,*column2*,... FROM *table* [GROUP BY *column(s)*];	SELECT LastName,COUNT(*) FROM Students GROUP BY LastName;
Group Filtering	SELECT *column1*,*column2*,... FROM *table* [GROUP BY *column(s)*] [HAVING *condition(s)*];	SELECT LastName,COUNT(*) FROM Students GROUP BY LastName HAVING LastName like '%son';
Joins	SELECT *column1*,*column2*,... FROM *table1*,*table2*,... [WHERE *condition(s)*];	SELECT LastName,Points FROM Students,Assignments WHERE AssignmentID=12 AND Students.StudentID=Assignments.StudentID;
Table Aliases	SELECT *column1*,*column2*,... FROM *table1* [*alias1*], *table2* [*alias2*],... [WHERE *condition(s)*];	SELECT LastName,Points FROM Students S,Assignments A WHERE S.StudentID=A.StudentID AND A.AssignmentID=12;
Everything	SELECT [DISTINCT] *column1* [AS *alias1*], *column2* [AS *alias2*], ... FROM *table1* [*alias1*], *table2* [*alias2*],... [WHERE *condition(s)*] [GROUP BY *column(s)*] [HAVING *condition(s)*] [ORDER BY *column(s)* [DESC]];	SELECT Points, COUNT(*) AS Cnt FROM Students S,Assignments A WHERE S.StudentID=A.StudentID AND A.AssignmentID=12 GROUP BY Points HAVING Points > 10 ORDER BY Cnt, Points DESC;