\*\*\* mysql.exe \*\*\*  
mysql -u DBUSER -h DBSERVERNAME -p  
  
/\*     Almost every table should have a PRIMARY KEY index, usually as an   
"id" column.  
     If a column is expected to contain unique values, it should have a   
UNIQUE index.  
     If you are going to perform searches on a column often (in the   
WHERE clause), it should have a regular INDEX.  
     If a column is used for a relationship with another table, it   
should be a FOREIGN KEY if possible, or have just a regular index   
otherwise. \*/  
  
------------------ Database & Table Management ------------------  
  
-- Create Database  
CREATE DATABASE my\_db;  
  
-- Create Database with default Character Set and Collation  
CREATE DATABASE my\_db DEFAULT CHARACTER SET utf8 COLLATE utf8\_general\_ci;  
  
-- Delete Database  
DROP DATABASE my\_db;  
  
-- Deletes a Table  
DROP TABLE table\_name;  
  
-- Select a database for use  
USE my\_db  
  
-- Shows all databases  
SHOW DATABASES;  
  
-- Shows all tables in a DB  
SHOW TABLES;  
  
-- Displays table structure  
EXPLAIN table\_name;  
  
-- Shows the ID of last inserted row  
SELECT LAST\_INSERT\_ID();  
  
NOW() -- Returns current date/time  
CURDATE() -- Returns current date  
  
-- Create a table with 3 columns, one of which is an auto incrementing   
primary key:  
CREATE TABLE `users` (  
     `user\_id` INT AUTO\_INCREMENT PRIMARY KEY,  
     `username` VARCHAR(20),  
     `create\_date` DATE  
);  
  
-- Alternate syntax  
CREATE TABLE states (  
     id INT AUTO\_INCREMENT,  
     name VARCHAR(20),  
     PRIMARY KEY (id)  
);  
  
-- Same as above, except 'name' is unique  
     CREATE TABLE states (  
         id INT AUTO\_INCREMENT,  
         name VARCHAR(20),  
         PRIMARY KEY (id),  
         UNIQUE (name)  
     );  
  
-- Same as above, except a different index name is assigned to 'name'  
     CREATE TABLE states (  
         id INT AUTO\_INCREMENT,  
         name VARCHAR(20),  
         PRIMARY KEY (id),  
         UNIQUE state\_name (name)  
     );  
  
-- join\_year is indexed (improving access speed)  
     CREATE TABLE states (  
         id INT AUTO\_INCREMENT,  
         name VARCHAR(20),  
         join\_year INT,  
         PRIMARY KEY (id),  
         UNIQUE (name),  
         INDEX (join\_year)  -- KEY (join\_year) -- has same effect  
     );  
  
-- Add a column to existing table  
ALTER TABLE `users`  
     ADD `email` VARCHAR(100)  
     AFTER `username`;  
  
-- Modify a column in a table  
ALTER TABLE `users`  
     CHANGE `username`  
     `username` VARCHAR(30);  
  
-- Delete a column from table  
ALTER TABLE `users` DROP `email`;  
  
  
------------------ Adding Table Data ------------------  
  
-- Insert new data (row) into table  
INSERT INTO `users` VALUES (  
     NULL,  
     'John Doe',  
     ['jdoe@example.com'](mailto:%27jdoe@example.com%27),  
     CURDATE()  
);  
  
-- Insert new data into table (Alternate syntax 1)  
INSERT INTO `users` SET  
     `username` = 'John Doe',  
     `email` = ['jdoe@example.com'](mailto:%27jdoe@example.com%27),  
     `create\_date` = '2010-10-10';  
  
-- Insert new data into table (Alternate syntax 2)  
INSERT INTO `users` (`email`, `username`, `create\_date`)  
VALUES (['jdoe@example.com'](mailto:%27jdoe@example.com%27), 'John Doe', '2010-10-10');  
  
  
------------------ Update Table Data ------------------  
  
UPDATE `users` SET  
     `email` = ['someone@example.com'](mailto:%27someone@example.com%27),  
     `username` = 'Mike James'  
     WHERE `username` = 'John1 Doe';  
  
UPDATE `users` SET  
     `email` = ['someone@example.com'](mailto:%27someone@example.com%27),  
     `username` = 'Mike James'  
     WHERE `user\_id` = 1;  
  
-- Limit max number of rows to be updated  
UPDATE `users` SET  
     `email` = ['someone@example.com'](mailto:%27someone@example.com%27),  
     `username` = 'Mike Smith'  
     WHERE `username` = 'Mike James'  
     LIMIT 1;  
  
  
------------------ Retreive Table Data ------------------  
  
-- Fetch all columns from the table  
SELECT \* FROM `users`;  
  
-- Fetch all columns from the table, but limit results  
SELECT \* FROM `users` LIMIT 2;  
SELECT \* FROM `users` LIMIT 2 OFFSET 5; -- Fetches from a certain offset  
SELECT \* FROM `users` LIMIT 5, 2; -- Same as above  
  
-- Fetch all columns from the table, ordered by a specific column  
SELECT \* FROM `users` ORDER BY `create\_date` DESC; -- or ASC  
  
SELECT COUNT(\*), `user\_id` from `users` GROUP BY `create\_date`;  
  
-- Fetch only specified columns from the table  
SELECT `username`, `email` FROM `users`;  
  
-- Fetch records that match the query  
SELECT `user\_id`, `email` FROM `users` WHERE `username` = 'John1 Doe';  
SELECT `user\_id`, `email` FROM `users` WHERE `user\_id` <= 5;  
SELECT `user\_id`, `email` FROM `users` WHERE `create\_date` != CURDATE();  
SELECT `user\_id`, `email` FROM `users` WHERE `user\_id` <= 2 OR   
`create\_date` = CURDATE();  
SELECT `user\_id`, `email` FROM `users` WHERE `create\_date`   
IN('2010-10-10', '2009-10-10'); -- Match multiple values  
SELECT `user\_id`, `email` FROM `users` WHERE `email` LIKE '%doe%'; --   
Wildcart match  
  
-- MIN MAX AVG  
SELECT MIN(`join\_year`), MAX(`join\_year`), AVG(`join\_year`) FROM `states`;  
------------------ Delete Table Data ------------------  
  
DELETE FROM `users` WHERE `username` = "James Doe";  
  
-- Delete all table contents  
DELETE FROM `users`;  
  
-- Same as above, but more efficient. Also resets AUTO\_INCREMENT.  
TRUNCATE TABLE `users`;  
  
\*\*\* mysql.exe \*\*\*  
mysql -u DBUSER -h DBSERVERNAME -p  
  
------------------ Database & Table Management ------------------  
  
-- Create Database  
CREATE DATABASE my\_db;  
  
-- Create Database with default Character Set and Collation  
CREATE DATABASE my\_db DEFAULT CHARACTER SET utf8 COLLATE utf8\_general\_ci;  
  
-- Delete Database  
DROP DATABASE my\_db;  
  
-- Deletes a Table  
DROP TABLE table\_name;  
  
-- Select a database for use  
USE my\_db  
  
-- Shows all tables in a DB  
SHOW TABLES;  
  
-- Displays table structure  
EXPLAIN table\_name;  
  
-- Shows the ID of last inserted row  
SELECT LAST\_INSERT\_ID();  
  
NOW() -- Returns current date/time  
CURDATE() -- Returns current date  
  
-- Create a table with 3 columns, one of which is an auto incrementing   
primary key:  
CREATE TABLE `users` (  
     `user\_id` INT AUTO\_INCREMENT PRIMARY KEY,  
     `username` VARCHAR(20),  
     `create\_date` DATE  
);  
  
-- Add a column to existing table  
ALTER TABLE `users`  
     ADD `email` VARCHAR(100)  
     AFTER `username`;  
  
-- Modify a column in a table  
ALTER TABLE `users`  
     CHANGE `username`  
     `username` VARCHAR(30);  
  
-- Delete a column from table  
ALTER TABLE `users` DROP `email`;  
  
  
------------------ Adding Table Data ------------------  
  
-- Insert new data (row) into table  
INSERT INTO `users` VALUES (  
     NULL,  
     'John Doe',  
     ['jdoe@example.com'](mailto:%27jdoe@example.com%27),  
     CURDATE()  
);  
  
-- Insert new data into table (Alternate syntax 1)  
INSERT INTO `users` SET  
     `username` = 'John Doe',  
     `email` = ['jdoe@example.com'](mailto:%27jdoe@example.com%27),  
     `create\_date` = '2010-10-10';  
  
-- Insert new data into table (Alternate syntax 2)  
INSERT INTO `users` (`email`, `username`, `create\_date`)  
VALUES (['jdoe@example.com'](mailto:%27jdoe@example.com%27), 'John Doe', '2010-10-10');  
  
  
------------------ Update Table Data ------------------  
  
UPDATE `users` SET  
     `email` = ['someone@example.com'](mailto:%27someone@example.com%27),  
     `username` = 'Mike James'  
     WHERE `username` = 'John1 Doe';  
  
UPDATE `users` SET  
     `email` = ['someone@example.com'](mailto:%27someone@example.com%27),  
     `username` = 'Mike James'  
     WHERE `user\_id` = 1;  
  
-- Limit max number of rows to be updated  
UPDATE `users` SET  
     `email` = ['someone@example.com'](mailto:%27someone@example.com%27),  
     `username` = 'Mike Smith'  
     WHERE `username` = 'Mike James'  
     LIMIT 1;  
  
  
------------------ Retreive Table Data ------------------  
  
-- Fetch all columns from the table  
SELECT \* FROM `users`;  
  
-- Fetch all columns from the table, but limit results  
SELECT \* FROM `users` LIMIT 2;  
SELECT \* FROM `users` LIMIT 2 OFFSET 5; -- Fetches from a certain offset  
SELECT \* FROM `users` LIMIT 5, 2; -- Same as above  
  
-- Fetch all columns from the table, ordered by a specific column  
SELECT \* FROM `users` ORDER BY `create\_date` DESC; -- or ASC  
  
-- Fetch only specified columns from the table  
SELECT `username`, `email` FROM `users`;  
  
-- Fetch records that match the query  
SELECT `user\_id`, `email` FROM `users` WHERE `username` = 'John1 Doe';  
SELECT `user\_id`, `email` FROM `users` WHERE `user\_id` <= 5;  
SELECT `user\_id`, `email` FROM `users` WHERE `create\_date` != CURDATE();  
SELECT `user\_id`, `email` FROM `users` WHERE `user\_id` <= 2 OR   
`create\_date` = CURDATE();  
SELECT `user\_id`, `email` FROM `users` WHERE `create\_date`   
IN('2010-10-10', '2009-10-10'); -- Match multiple values  
SELECT `user\_id`, `email` FROM `users` WHERE `email` LIKE '%doe%'; --   
Wildcart match  
  
  
------------------ Delete Table Data ------------------  
  
DELETE FROM `users` WHERE `username` = "James Doe";  
  
-- Delete all table contents  
DELETE FROM `users`;  
  
-- Same as above, but more efficient. Also resets AUTO\_INCREMENT.  
TRUNCATE TABLE `users`;