A Quick Guide To MySQL Tables & Queries

This is a Quick reference Guide for MySQL 5.x. MySQL is a relational database management system (RDBMS) based on SQL (Structured Query Language). MySQL is available under GNU public license and through subscription to MySQL Network for business applications. It runs on Unix, iMac, and Windows and provides rich API for many programming languages including C, C++, Java, Perl, Python, Ruby and PHP.

Database Queries:

List all databases
```
>SHOW databases;
```

Select the database
>USE <database name>
```
>USE university;
```

Create a database
>CREATE DATABASE <database name>
```
>CREATE DATABASE university;
```

Delete a database
>DROP DATABASE <database name>
```
>DROP DATABASE university;
```

Rename a database
>ALTER DATABASE <database name> RENAME <new database name>
```
>ALTER DATABASE university RENAME faculty
```

Table Queries:

Create a table
>CREATE TABLE <table name> (<field name> <field type> (<field size>), ...)
```
>CREATE TABLE teachers (name varchar(20), age INT(10));
```

List all tables in the database
```
>SHOW tables;
```

Show table format with column names and data types
>DESCRIBE <table name>
```
>DESCRIBE teachers;
```

Modify the structure of table
```
>ALTER TABLE <table name> <alter specifications>
>ALTER TABLE teachers DROP COLUMN salary;
>ALTER TABLE teachers ADD COLUMN salary INT(5);
>ALTER TABLE teachers CHANGE firstName name VARCHAR(20);
```

Delete the table
>DROP TABLE <table name>
```
>DROP TABLE teachers;
```

Retrieving Data:

>SELECT <columns> FROM <tables> WHERE <conditions>

Retrieve from all columns
*>SELECT * FROM <tables>*
```
>SELECT * FROM teachers;
```

Retrieve from selected columns
>SELECT <column 1>, <column 2> FROM <tables>
```
>SELECT id, name FROM teachers;
```

Retrieve from selected tables
>SELECT <columns> FROM <table 1>, <table 2>
```
>SELECT teachers.name, students.name FROM teachers, students;
```

Retrieve unique values
>SELECT DISTINCT <column name> FROM <table>
```
>SELECT DISTINCT name FROM teachers;
```

Retrieve data satisfying a given condition
>SELECT <columns> FROM <tables> WHERE <condition>
```
>SELECT name FROM teachers WHERE age > 35;
```

Retrieve data satisfying multiple conditions
>SELECT <columns> FROM <tables> WHERE <condition> AND <condition>
```
>SELECT name FROM teachers WHERE age > 35 AND gender = 'female';
```

Inserting Data:

> INSERT INTO <table> (<columns>) VALUES (<data>)
```
> INSERT INTO teachers (id, name, age) VALUES (NULL, 'John', '12');
```

Loading Data from Files:

>LOAD DATA LOCAL INFILE '<filename>' INTO TABLE <table>
```
>LOAD DATA LOCAL INFILE 'file.sql' INTO TABLE teachers;
```

This is very convenient to load data directly from files when you have thousands of entries.

Modifying Data:

>UPDATE <table> SET <field1> = <value1> AND <field2> = <value2> WHERE <conditions>
```
>UPDATE teachers SET status = 'enrolled' WHERE fees = 'paid';
```

Deleting Data:

>DELETE FROM <table> WHERE <condition>
```
>DELETE FROM teachers WHERE fees = 'paid';
```

Pattern Matching Examples:

```
>SELECT * FROM teachers WHERE name LIKE 'j%';
```
Wildcard % selects joe, john, jones, etc.

```
> SELECT * FROM teachers WHERE name LIKE '_ _ _';
```
Selects 3 character values.

```
> SELECT * FROM teachers WHERE name REGEXP '^A';
```
Selects all entries beginning with A.

```
> SELECT * FROM teachers WHERE name REGEXP 'p$';
```
Selects all entries ending with p.

[abc]	match a, b, or c
[^abc]	match all expect a, b, or c
[A-Z]	match uppercase
[a-z]	match lowercase
[0-9]	match any digit

*	match zero or more instances
+	match one or more instances
?	match zero or one instance
.	match any single char
^	match the beginning
$	match the end
|	separates alternatives
{n,m}	match at least n times but notmore than m times
{n}	string must occur exactly n times
{n,}	string must occur at least n times

Sorting:

>SELECT <columns> FROM <table> ORDER BY <column> <ASC or DESC>

```
>SELECT * FROM teachers ORDER BY age;
>SELECT * FROM teachers ORDER BY name
DESC;
```

Sorts the query results.

Limiting:

>SELECT <columns> FROM <table> LIMIT <from>, <to>

```
>SELECT * FROM teachers LIMIT 1,5;
```

Limits query results to specific range.

Grouping:

>SELECT <columns> FROM <table> GROUP BY <column>

```
>SELECT name, COUNT(*) FROM faculty GROUP
BY name;
```

GROUP BY is used to group values from a column. You can also perform calculations on that column using count, sum, avg. Count returns the number of rows which are selected.

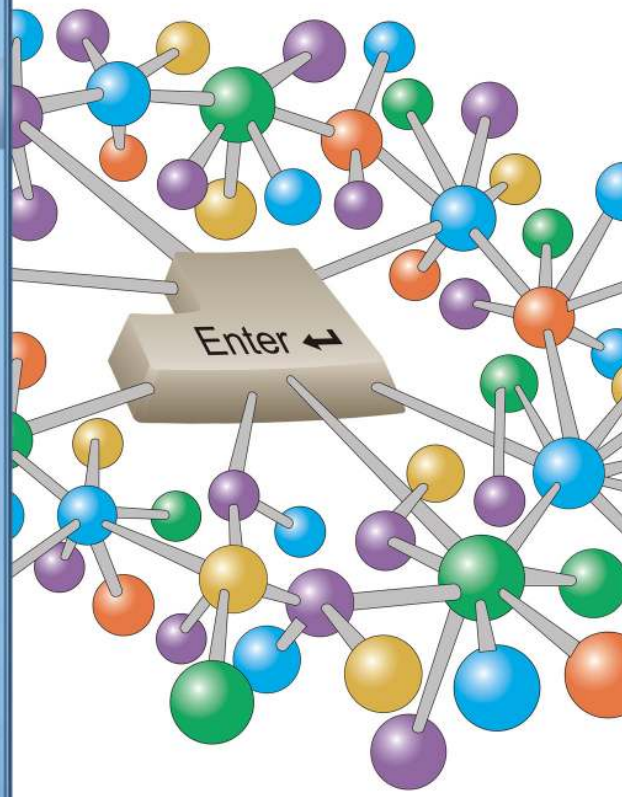
This document was written by Awais Naseem & Nazim Rahman from EMBnet Pakistan Node and being distributed by Education & Training Program Committee of EMBnet.

EMBnet – European Molecular Biology Network – is a bioinformatics support network of bioinformatics support centers situated primarily in Europe.

http://www.embnet.org/

A Quick Guide To MySQL Tables & Queries
First edition © 2010

EMBnet



A Quick Guide

MySQL

Tables & Queries