

MYSQL

Raghuram.N

Free Software Movement Karnataka

www.fsmk.org

Content

- Introduction to Database
- Introduction to MYSQL
- Database Design
- Table – Select, Create
- Table – Insert, Update, Delete
- PHP – MYSQL Connection
- Access and Design Table

Database

What is Database?

- Collection of Data
- Organized way of holding data.
- DBMS – Software to manage collection of data
- **RDBMS – Relational Database Management**
- Related data stored in table
- Easy to access complex information

Introduction to MYSQL

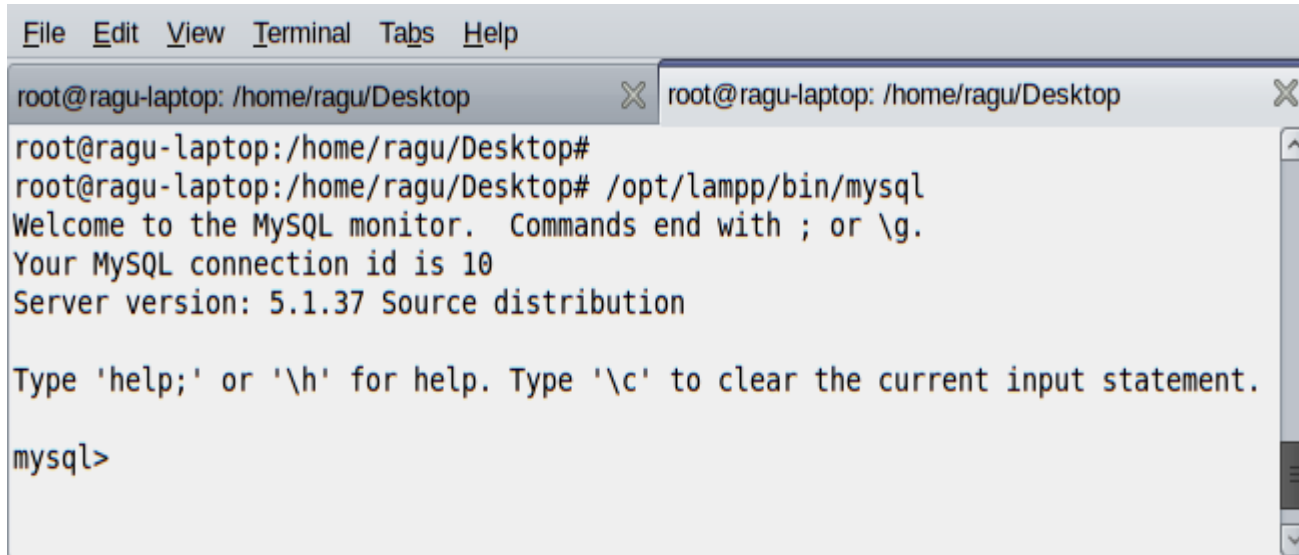
Why MYSQL?

- Well suited for Web Application
- Cost Effective:**
- Free Software – GPL – GNU Public License
- Fast and Secure**
- Extremely fast for small to medium sized database
- Written in C
- Continuous Improvement:**
- Frequent updates are being released by community. It supports sub quires and stored procedures.
- Free from Bugs:**

Introduction to MYSQL

- Download xamp server - xampp-linux-1.7.2.tar.gz
- Extract this file into **/opt/**

Start Mysql on Linus:



The screenshot shows a terminal window with a menu bar (File, Edit, View, Terminal, Tabs, Help) and two tabs. The active tab is titled 'root@ragu-laptop: /home/ragu/Desktop'. The terminal content shows the user running the command `/opt/lampp/bin/mysql`, which opens the MySQL monitor. The output includes a welcome message, connection ID 10, and server version 5.1.37. The prompt `mysql>` is visible at the bottom.

```
File Edit View Terminal Tabs Help
root@ragu-laptop: /home/ragu/Desktop
root@ragu-laptop: /home/ragu/Desktop
root@ragu-laptop:/home/ragu/Desktop#
root@ragu-laptop:/home/ragu/Desktop# /opt/lampp/bin/mysql
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 10
Server version: 5.1.37 Source distribution

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

Introduction to SQL

Introduction to SQL:

- SQL – Structured Query Language

SQL Guidelines:

- Statements are Case Insensitive
- Statement can be entered on one or more lines
- Clauses (WHERE) are usually entered in separate lines for readability and ease of editing

Database Design

Create Database:

CREATE used to create a database

Syntax:

```
CREATE DATABASE db_name;
```

Eg:

```
CREATE DATABASE myFirstDb;
```

Display Available Database:

SHOW is used to display all database

Syntax:

```
SHOW DATABASES;
```

Select a Database:

USE is used to display all database

Syntax:

```
USE DATABASE db_name;
```

Eg:

```
USE DATABASE myFirstDb;
```

Arithmetic Operator

Operator	Description	Eg
+	Addition	SELECT 3+5;
-	Minus	SELECT 5-3;
*	Multiplication	SELECT 5*3;
/	Division	SELECT 5/3;
DIV	Division	SELECT 5 DIV 3

Math Operator

Operator	Description	Eg
ABS()	Returns absolute value	SELECT ABS(2);=>2 SELECT ABS(-2)=>2
CEIL()	Return the smallest integer value not less than arg	SELECT CEIL(1.25) => 2
EXP()	Raise to the power of arg	SELECT EXP(2) => 7.38
FLOOR()	Return the largest integer value not greater than the arg	SELECT FLOOR(1.23) => 2
MOD()	Returns remainder	SELECT MOD(29,9) => 2 SELECT 29 % 2 => 2
OCT()	Return an octal representation of a decimal	SELECT OCT('2') => 50
PI()	Return the value of pi	SELECT PI() => 3.141
POW()	Return the arg raised to the specified power	SELECT POW(5,2) => 25
POWER()	Return the arg raised to the specified power	SELECT POWER(5,2) => 25
RAND()	Return random floating no	SELECT RAND();
ROUND()	Return the round value of arg	SELECT ROUND(1.2) => 1 SELECT ROUND(1.6) => 2
SQRT()	Return the square root of arg	SELECT SQRT(4) => 2
TRUNCATE()	Truncate to specified no of decimal places	SELECT TRUNCATE(1.2332,1) => 1.2

SELECT

Display Available Tables:

SHOW is used to display all Tables

Syntax:

```
SHOW TABLES;
```

Select Data from Table:

SELECT is used to display table's record

Syntax:

```
SELECT * FROM table_name; //Selects all row and column
```

```
SELECT Column1,Column2 FROM table_name; //Selects specified column and all row of table
```

```
SELECT Column1,Column2 FROM table_name WHERE Column1=value;
```

Eg:

```
SELECT * FROM studInfo;
```

```
SELECT RollNo, StudentName FROM studInfo;
```

```
SELECT * FROM studInfo WHERE RollNo=1001; //Selects all column where rollno is equal to 1001
```

CREATE

Create a Table:

CREATE is used to create a table

Syntax:

```
CREATE TABLE table_name (Column1  
    datatype, Column2 datatype, Column3 datatype);
```

Eg:

```
CREATE TABLE studInfo(RollNo int, StudentName  
varchar(30), StudentDept varchar(20), Year int);
```

```
CREATE TABLE studInfo(RollNo int not null primary key  
auto_increment, StudentName varchar(30), StudentDept  
varchar(20), Year int);
```

INSERT

Insert Data into Table:

INSERT is used to enter/add a record into table

Syntax:

INSERT INTO table_name VALUES (val1, val2, val3); //Enter value for all columns

INSERT INTO table_name (Column1, Column2, Column4) VALUES (val1, val2, val4); //Enter value for specified columns

Eg:

INSERT INTO studInfo VALUES (1001, 'Raghu', 'CSE', 2004);

INSERT INTO studInfo (RollNo, StudentName) VALUES (1002, 'Ram');

UPDATE and DELETE

Update Table Data:

UPDATE is used to update/modify exists data.

Syntax:

UPDATE table_name SET column1='value'; //Update value of column1 of all row
UPDATE table_name SET column2='value' WHERE column1=value; //Update value of column2 with conditions

Eg:

```
UPDATE studInfo SET StuDept='CSE';  
UPDATE studInfo SET StuDept='IT' WHERE RollNo=1002;
```

Delete Table Data:

DELETE is used to delete data.

Syntax:

DELETE FROM table_name; //Deletes all row
DELETE FROM table_name WHERE column2='value'; //Delete specified row

Eg:

```
DELETE FROM studInfo;  
DELETE FROM studInfo WHERE RollNo=1002;
```

Limiting Result

Limiting Result:

It provides following Clause to limit result.

- ORDER BY
- LIMIT
- WHERE
- Comparison Operators
- AND, OR, LIKE, BETWEEN

ORDER BY

Display result by Ascending or Descending order.

Syntax

```
SELECT * FROM table_name ORDER BY Column ASC/DESC;
```

Eg:

```
SELECT * FROM studInfo ORDER BY RollNo ASC/DESC;
```

LIMIT

Display limited no of records.

Syntax

```
SELECT * FROM table_name LIMIT no;
```

Eg:

```
SELECT * FROM table_name LIMIT 2; //Display 2 records
```

Comparison Operator

Operator	Description
=	Equal to
!=	Not Equal to
<	Less than
<=	Less than or equal to
>	Greater Than
>=	Greater than or equal to

Eg:

```
SELECT * FROM studInfo WHERE RollNo>1001 AND RollNo<1003;
```

PHP - MYSQL

Create Connection with MYSQL:

mysql_connect() function establish connection with MYSQL.

Syntax

```
mysql_connect(servername,username,password);
```

Parameter	Description
Servername	Specifies the server to connect to. Default value localhost
Username	Specifies the username to log in with. Default value root.
Password	Specifies the password to log in with.

Eg:

```
$con=mysql_connect("localhost","root","");
```

Close Connection:

mysql_close() is used to close the connection

Syntax

```
mysql_close($con);
```


PHP - MYSQL

Select Database:

mysql_select_db() function selects a database.

Syntax:

```
mysql_select_db(DatabaseName,Connection Variable);
```

Eg:

```
mysql_select_db("my_db",$con);
```

Execute MYSQL Query from PHP:

mysql_query() function executes mysql query.

mysql_num_rows() function returns no of row.

Syntax

```
mysql_query(strQuery,Connection Variable);
```

Eg:

```
$r=mysql_query("SELECT * FROM studInfo",$con);
```

Thank you