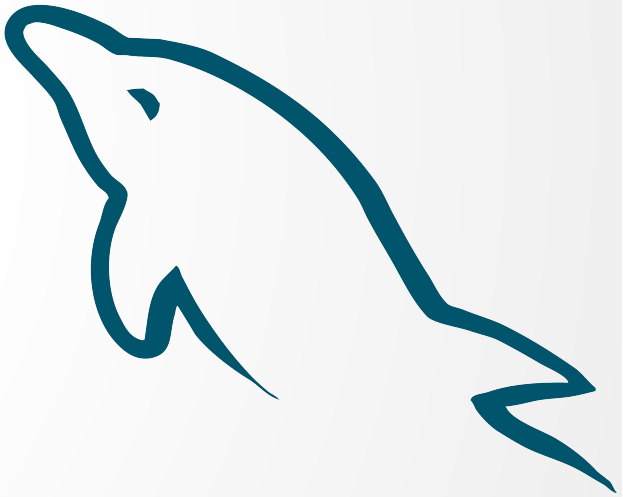
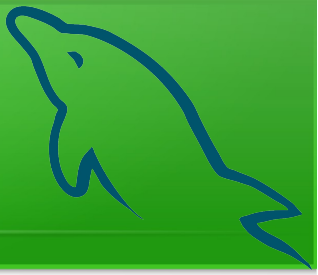
Introduction to MySQL

Raghuram

Free Software Movement Karnataka



Introduction to MySQL

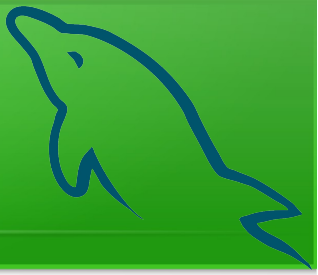


Content

Introduction to Database
Introduction to MYSQL
Database Design
Table – Select, Create
Table – Insert, Update, Delete

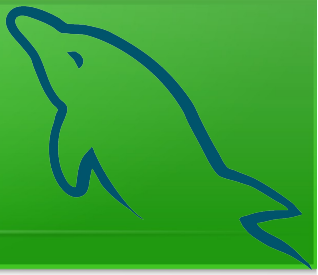
What is Data?

Introduction to MySQL



What is Database?

- Collection of Data
- Organized way of holding data.



What is DBMS?

- DBMS – Software to manage collection of data

Types of DBMS

- Hierarchical databases
- Network databases
- Relational databases
- Object-Oriented Databases

Hierarchical Database

- Commonly used on Mainframe Computers.
- Oldest methods of Organizing data.
- Organized in Pyramid Fashion.
- Top of the pyramid is called the root record.
- A child record always has only one parent record.
- A parent record may have more than one child record.

Hierarchical Database

Advantages

- Accessed and Updated rapidly

Disadvantages

- Each child in the tree may have only one parent.
- Relationships between children are not permitted.
- Rigid Design.
- Adding a new field requires that the entire database be redefined.

Network Database

- Similar to hierarchical databases
- Instead of looking like an upside-down tree, a network database looks more like a cobweb or interconnected network of records.
- Children are called members and parents are called owners.
- Each child or member can have more than one parent (or owner).

Network Database

Advantages

- More flexible

Disadvantages

- Defined in advance.
- Limit to the number of connections.

Relational Database

- Data Organized in Tabular files.
- Related to each other by a common field.
- Key field that uniquely identifies each row.

Relational Database

Advantages

- Relational databases can be used with little or no training.
- Entries can be modified without redefining the entire structure.

Disadvantages

- Searching for data can take more time.

Object Oriented Database

- Used to store data from a variety of media sources.
- Use small, reusable chunks of software called objects.
- Each object consists of two elements:
  1. Data (e.g., sound, video, text, or graphics).
- 2. Instructions, or Methods (what to do with the data).

Object Oriented Database

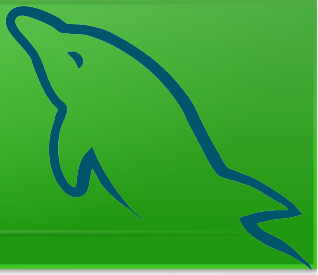
Advantages

- Incredible multimedia capability.
- Used in Healthcare organizations, can store, track, and recall CAT scans, X-rays, electrocardiograms and many other forms of crucial data

Disadvantages

- Costly to develop.

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.

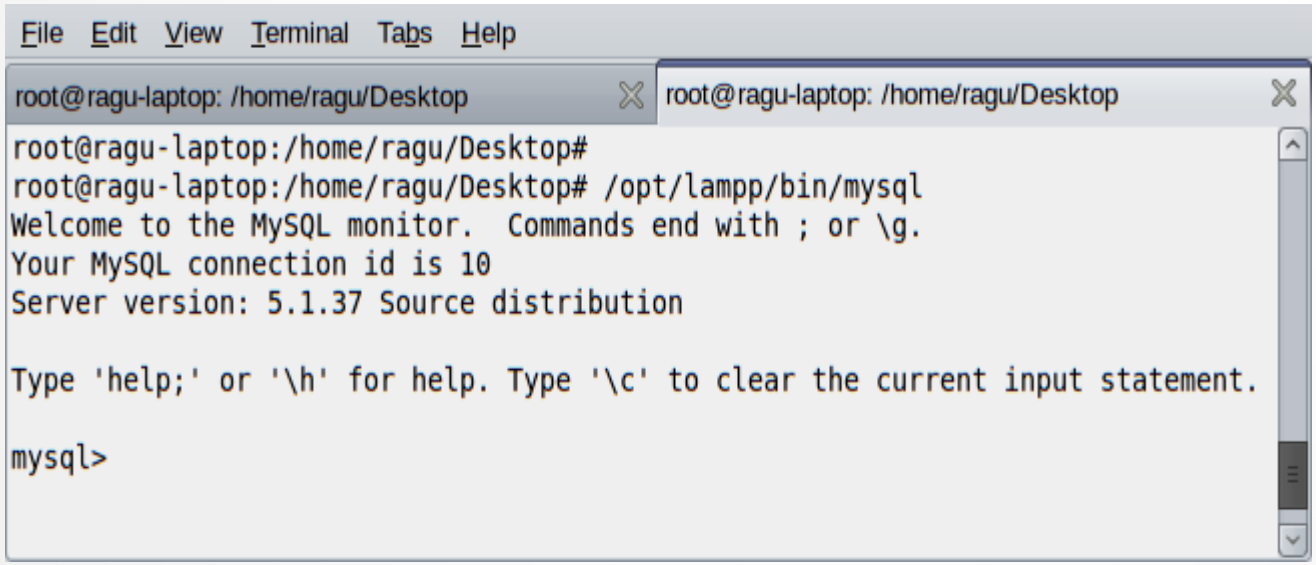
Free from Bugs:

Introduction to MySQL

Install MySQL

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

Start Mysql on Linus:



Introduction to MySQL

Introduction to SQL

- SQL – Structured Query Language

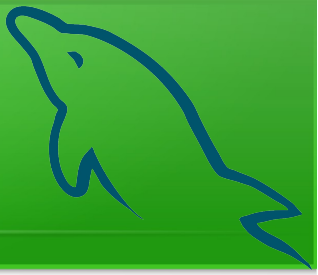
SQL Guidelines:

- 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

Types of SQL

→ Data Definition Language(DDL)
→ Data Manipulation Language(DML)
→ Data Control Language(DCL)
→ Transaction Control Language(TCL)

Introduction to MySQL



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 db_name;
Eg:
USE classicmodels;

Data Definition Language

- Used to define the database object, structure or schema.

Examples

- **CREATE** - to create objects in the database.
- **ALTER** - alters the structure of the database.
- **DROP** - delete objects from the database.
- **TRUNCATE** - remove all records from a table, including all spaces allocated for the records are removed.

Data Manipulation Language

- Used for managing data within schema objects.

Examples

- **SELECT** - Retrieve data from the a database.
- **INSERT** - Insert data into a table.
- **UPDATE** - Updates existing data within a table.
- **DELETE** - Deletes records from a table, the space for the records remain.

Data Control Language

- Used to give access to other users.

Examples

- **GRANT** - Gives user's access privileges to database.
- **REVOKE** - Withdraw access privileges given with the GRANT command.

Transaction Control Language

- Used to manage the changes made by DML statements.
- Allows statements to be grouped together into logical transactions.

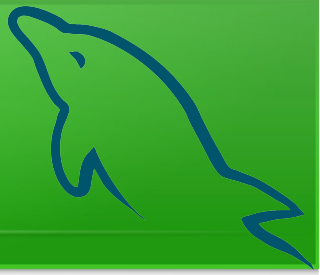
Examples

- **COMMIT** - save work done.
- **ROLLBACK** - Restore database to original since the last COMMIT.
- **SAVEPOINT** - Identify a point in a transaction to which you can later roll back.
- **SET TRANSACTION** - Change transaction options like isolation level and what rollback segment to use

String Column Types

- Char
- Varchar
- tinytext/tinyblob
- text/blob
- mediumtext/mediumblob
- longtext/longblob
- Enum
- Set

Introduction to MySQL



Create a Table:

CREATE is used to create a table

Syntax:

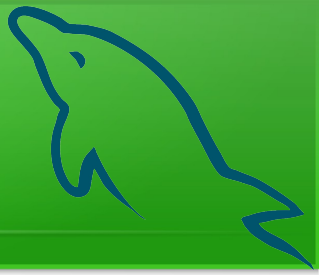
CREATE TABLE table_name (Column1 datatype,Column2 datatype,Column3 datatype);

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);

Introduction to MySQL



ALTER

- Used to change table name
- Add new column or modify existing column.

Change Table Name:

ALTER TABLE table_name RENAME new_table_name;

Eg:

ALTER TABLE studInfo RENAME stud_Info;

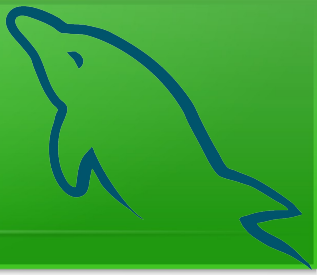
Adding Column:

ALTER TABLE table_name ADD COLUMN column_name column_type;

Eg:

ALTER TABLE studInfo ADD COLUMN previous_institute VARCHAR(50);

Introduction to MySQL



DROP

- Used to remove Database objects like table, view, column, stored procedure.

Drop Column

ALTER TABLE table_name DROP COLUMN col_name;

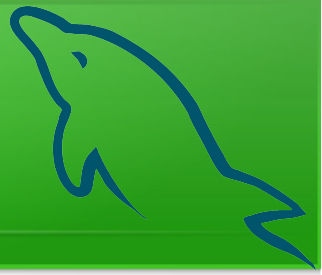
Drop Table

DROP TABLE table_name;

Drop Database

DROP DATABASE db_name;

Introduction to MySQL



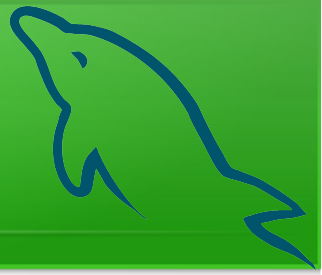
Display Available Tables:
SHOW is used to display all Tables
Syntax:
SHOW TABLES;

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

Introduction to MySQL



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

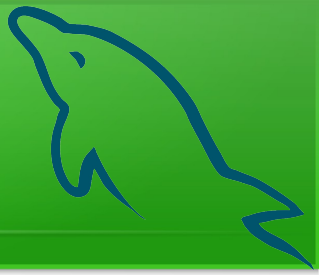
INSERT INTO table_name (Column1,Column2,Column4) VALUES (val1,val2, val4); //Enter value for specified columns

Eg:

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

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

Introduction to MySQL

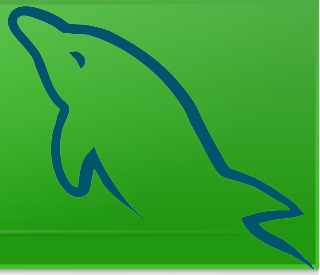


Update Table Data:
UPDATE is used to update/modify exists data.

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;

Introduction to MySQL

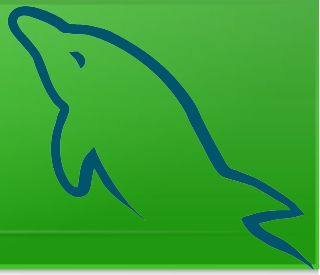


Delete Table Data:
DELETE is used to delete data.

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;

Introduction to MySQL



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 or records.

Syntax

SELECT * FROM table_name LIMIT no;

Eg:

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

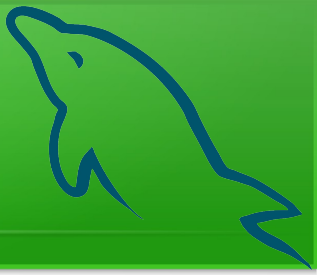
Arithmetic Operator

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

Introduction to MySQL



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

Eg:

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

Introduction to Workbench
&
PhpMyAdmin