MySQL®

SQL Language

1. DDL(Data Definition Language)
2. DML(Data Manipulation Language)
3. DRL/DQL (Data Retrieval Language/Data Query Language)
4. TCL (Transaction Control Language)
5. DCL (Data Control Language)

SQL Commands

- DDL (Data Definition Language):
  - CREATE, ALTER, DROP, TRUNCATE, RENAME
- DML (Data Manipulation Language):
  - INSERT, UPDATE, DELETE
- DRL/DQL (Data Retrieval /Data Query Language):
  - SELECT
- TCL (Transaction Control Language)
  - COMMIT, ROLLBACK, SAVE POINT
- DCL (Data Control Language)
  - GRANT, REVOKE

Create Database/Schema

- CREATE DATABASE testdb;
- DROP DATABASE testdb;

- CREATE SCHEMA testdb;
- DROP SCHEMA testdb;

- CREATE DATABASE IF NOT EXISTS testdb;

Creating Table

- CREATE TABLE [IF NOT EXISTS] table_name(column_1_definition, column_2_definition, ..., table_constraints)
- Example:
    - USE testdb;
    - CREATE TABLE BOOK(BID INT, BTITLE VARCHAR(100), AUTHOR VARCHAR(30), QUANTITY INT);

Inserting data into table

- INSERT INTO <TABLE NAME> VALUES(VAL1,VAL2,VAL3,....);
- Example:
  - USE testdb;
  - INSERT INTO BOOK VALUES(10, 'My Experiments with Truth', 'MK Gandhi', 3000);
  - INSERT INTO BOOK(AUTHOR,BTITLE,BID,QUANTITY) VALUES('APJ Abdul Kalam','Wings of Fire',20, 2000);
  - INSERT INTO BOOK VALUES(30, 'The Alchemist', 'Paulo Coelho', NULL);

Selecting Rows from a table

- To retrieve rows for all columns or specific columns from the table
  SELECT select_list FROM <table_name>;

- Example:
  - USE classicmodels;
  - SELECT * FROM EMPLOYEES;
  - SELECT LASTNAME, FIRSTNAME, JOBTITLE FROM EMPLOYEES;
  - SELECT EMPLOYEENUMBER EMPID,FIRSTNAME FNAME, JOBTITLE DESIGNATION FROM EMPLOYEES;
  - SELECT EMPLOYEENUMBER+1000 EMPID,FIRSTNAME FNAME, JOBTITLE DESIGNATION FROM EMPLOYEES;

SQL Data types

- Numeric
- String
- Date/time

Numeric Data Types

TINYINT	-128 to 127 normal 0 to 255 UNSIGNED.
SMALLINT	-32768 to 32767 normal 0 to 65535 UNSIGNED.
MEDIUMINT	-8388608 to 8388607 normal 0 to 16777215 UNSIGNED.
INT	-2147483648 to 2147483647 normal 0 to 4294967295 UNSIGNED.
BIGINT	-9223372036854775808 to 9223372036854775807 normal 0 to 18446744073709551615 UNSIGNED.
FLOAT	A small approximate number with a floating decimal point.
DOUBLE(,)	A large number with a floating decimal point.
DECIMAL(,)	A DOUBLE stored as a string , allowing for a fixed decimal point. Choice for storing currency values.

String Data Types

CHAR( )	A fixed section from 0 to 255 characters long.
VARCHAR( )	A variable section from 0 to 65,535 characters long.
BLOB	A string with a maximum length of 65535 characters.
MEDIUMTEXT	A string with a maximum length of 16777215 characters.
MEDIUMBLOB	A string with a maximum length of 16777215 characters.
LONGTEXT	A string with a maximum length of 4294967295 characters.
LONGBLOB	A string with a maximum length of 4294967295 characters.

Date & Time

DATE	YYYY-MM-DD
DATETIME	YYYY-MM-DD HH:MM:SS
TIMESTAMP	YYYYMMDDHHMMSS
TIME	HH:MM:SS
YEAR	YYYY

Where clause

- Used for selecting the rows based on condition (Filtering the rows using where condition)
  SELECT select_list FROM table_name WHERE search_condition;

- Example
  - USE classicmodels;
  - SELECT * FROM EMPLOYEES;
  - SELECT * FROM EMPLOYEES WHERE JOBTITLE = 'Sales Rep';
  - SELECT * FROM EMPLOYEES WHERE OFFICECODE=6;
  - SELECT * FROM CUSTOMERS WHERE CREDITLIMIT>200000;
  - SELECT * FROM CUSTOMERS WHERE CREDITLIMIT<=30000;
  - SELECT * FROM CUSTOMERS WHERE STATE IS NULL;
  - SELECT DISTINCT OFFICECODE FROM EMPLOYEES;
  - SELECT DISTINCT * FROM EMPLOYEES;

Logical Operators (AND, OR, NOT)

- USE classicmodels;
- SELECT * FROM EMPLOYEES;
- SELECT * FROM EMPLOYEES WHERE OFFICECODE=1 AND JOBTITLE='Sales_Rep';
- SELECT * FROM EMPLOYEES WHERE OFFICECODE=1 OR JOBTITLE='Sales_Rep';
- SELECT * FROM EMPLOYEES WHERE NOT LASTNAME='Murphy';

Between & IN Operators

- Between
  - Used to display the rows which falls in the range of values.
    - USE classicmodels;
    - SELECT * FROM CUSTOMERS WHERE CREDITLIMIT BETWEEN 50000 AND 100000;

- Not Between
  - SELECT * FROM CUSTOMERS WHERE CREDITLIMIT NOT BETWEEN 50000 AND 100000;
- IN
  - Return the rows for the values matching in the list
    - SELECT * FROM CUSTOMERS WHERE CREDITLIMIT=11000 OR CREDITLIMIT=21000 OR CREDITLIMIT=84600;
    - SELECT * FROM CUSTOMERS WHERE CREDITLIMIT IN (11000,21000,84600);
- Not In
  - SELECT * FROM CUSTOMERS WHERE CREDITLIMIT NOT IN (11000,21000,84600);

Pattern Matching operators (Wild card characters)

- % matches multiple characters
- _ matches any single character
- SELECT * FROM EMPLOYEES WHERE FIRSTNAME LIKE 'L%';
- SELECT * FROM EMPLOYEES WHERE FIRST_NAME LIKE '%y';
- SELECT * FROM EMPLOYEES WHERE FIRST_NAME LIKE 'A%y';
- SELECT * FROM EMPLOYEES WHERE FIRST_NAME LIKE '%s%';
- SELECT * FROM EMPLOYEES WHERE FIRST_NAME NOT LIKE 'M%';
- SELECT * FROM EMPLOYEES WHERE FIRST_NAME LIKE '%r_';
- SELECT * FROM EMPLOYEES WHERE FIRST_NAME LIKE '_ _ _';

Updating table data

- UPDATE table_name SET column_name1 = expr1, column_name2 = expr2, ... [WHERE condition];
- Example:
  - USE testdb;
  - SET SQL_SAFE_UPDATES = 0;
  - UPDATE BOOK SET QUANTITY=4000 WHERE BID=30;
  - UPDATE BOOK SET AUTHOR='Mahatma Gandhi', QUANTITY=5000 WHERE BID=10;

Deleting data from table

- DELETE FROM <table_name> [WHERE condition];
- Example:
  - USE testdb;
  - DELETE FROM BOOK WHERE BID=30;
  - DELETE FROM BOOK WHERE BID IN (10,20);
  - DELETE FROM BOOK