

Dbms\_lab command query :

mysql -u root

Show databases;

Use database\_name;

Show tables;

Desc tableName;

**CREATE DATABASE** *databasename;*

**DROP DATABASE** *databasename;*

**CREATE TABLE** *table\_name* (

*column1 datatype,*

*column2 datatype,*

*column3 datatype....*

);

**CREATE TABLE** Persons (

ID int **NOT NULL UNIQUE/ ID int PRIMARY KEY,**

LastName varchar(**255**) **NOT NULL,**

FirstName varchar(**255**),

Age int **CHECK** (Age>=**18**),

**PRIMARY KEY** (ID)

);

**CREATE TABLE** Orders (

OrderID int **NOT NULL PRIMARY KEY,**

OrderNumber int **NOT NULL,**

PersonID int **FOREIGN KEY REFERENCES** Persons(PersonID)

);

**CREATE TABLE** *new\_table\_name AS*

**SELECT** *column1, column2,...*

**FROM** *existing\_table\_name*

**WHERE** ....;

**INSERT INTO** *table\_name* values

(.....),

(.....);

**DROP TABLE** *table\_name;*

TRUNCATE/DELETE TABLE *table\_name*;

ALTER TABLE *table\_name*  
ADD *column\_name datatype*;

ALTER TABLE *table\_name*  
DROP COLUMN *column\_name*;

ALTER TABLE *table\_name*  
RENAME COLUMN *old\_name* to *new\_name*;

SELECT \* FROM *table\_name*;  
SELECT *column1, column2, ...*  
FROM *table\_name*;  
SELECT DISTINCT Country FROM Customers;

SELECT *column1, column2, ...*  
FROM *table\_name*  
WHERE *condition*;

SELECT \* FROM Customers  
WHERE Country='Mexico';

SELECT *column1, column2, ...*  
FROM *table\_name*  
ORDER BY *column1, column2, ... ASC|DESC*;

SELECT *column1, column2, ...* FROM *table\_name*  
WHERE *condition1 AND condition2 AND condition3 ...*;

SELECT \* FROM Customers  
WHERE Country = 'Spain' AND CustomerName LIKE 'G%';

SELECT \* FROM Customers  
WHERE CustomerName NOT LIKE 'A%';

SELECT \* FROM Customers  
WHERE NOT Country = 'Spain'

```
SELECT * FROM Customers
WHERE City NOT IN ('Paris', 'London');
```

```
INSERT INTO table_name (column1, column2, column3, ...)
VALUES (value1, value2, value3, ...);
INSERT INTO table_name
VALUES (value1, value2, value3, ...);
```

```
UPDATE table_name SET column1 = value1, column2 = value2, ...
WHERE condition;
DELETE FROM table_name WHERE condition;
DELETE FROM Customers WHERE CustomerName='Alfreds Futterkiste';
DELETE FROM table_name;
```

```
SELECT MIN(column_name)
FROM table_name
WHERE condition;
```

```
SELECT column1, column2, ...
FROM table_name
WHERE columnN LIKE pattern;
```

```
SELECT * FROM Customers
WHERE city LIKE 'L_nd__';
```

```
SELECT * FROM Customers
WHERE city LIKE '%L%';
```

The **IN** operator is a shorthand for multiple **OR** conditions.

```
SELECT column_name(s)
FROM table_name
WHERE column_name IN (value1, value2, ...);
```

```
SELECT * FROM Customers
WHERE Country IN ('Germany', 'France', 'UK');
```

```
SELECT column_name(s)
```

```
FROM table_name  
WHERE column_name BETWEEN value1 AND value2;
```

```
SELECT * FROM Products  
WHERE Price BETWEEN 10 AND 20;
```

Aggregate functions

```
SELECT avg(salary) FROM instructor WHERE dept_name = 'Comp. Sci.;
```

```
SELECT COUNT(distinct dept_name) FROM instructor;
```

```
SELECT COUNT(*) AS num_row FROM instructor;
```

Rename:

```
RENAME table1 to table2;
```

Create trigger:

```
DELIMITER //
```

```
Create Trigger before_insert_occupation
```

```
BEFORE INSERT ON employee FOR EACH ROW
```

```
BEGIN
```

```
IF NEW.occupation = 'Scientist' THEN SET NEW.occupation = 'Doctor';
```

```
END IF;
```

```
END //
```

```
DELIMITER ;
```

Drop trigger [if exists] trigger\_name;

```
DELIMITER //
```

```
Create Trigger after_insert_details
```

```
AFTER INSERT ON student_info FOR EACH ROW  
BEGIN  
INSERT INTO student_detail VALUES (new.stud_id, new.stud_code,  
new.stud_name, new.subject, new.marks, new.phone, CURTIME());  
END //
```

```
DELIMITER $$
```

```
CREATE TRIGGER before_update_salesInfo
```

```
BEFORE UPDATE
```

```
ON sales_info FOR EACH ROW
```

```
BEGIN
```

```
    DECLARE error_msg VARCHAR(255);
```

```
    SET error_msg = ('The new quantity cannot be greater than 2 times the current  
quantity');
```

```
    IF new.quantity > old.quantity * 2 THEN
```

```
        SIGNAL SQLSTATE '45000'
```

```
        SET MESSAGE_TEXT = error_msg;
```

```
    END IF;
```

```
END $$
```

```
DELIMITER $$
```

```
CREATE TRIGGER after_update_studentsInfo
```

```
AFTER UPDATE
```

```
ON students FOR EACH ROW
```

```
BEGIN
```

```
INSERT into students_log VALUES (user() ,  
CONCAT('Update Student Record ', OLD.name, ' Previous Class :',  
      OLD.class, ' Present Class ', NEW.class));  
END $$
```

```
UPDATE sales_info SET quantity = 125 WHERE id = 2;
```

```
DELIMITER $$
```

```
CREATE TRIGGER before_delete_salaries
```

```
BEFORE DELETE
```

```
ON salaries FOR EACH ROW
```

```
BEGIN
```

```
  INSERT INTO salary_archives (emp_num, valid_from, amount)
```

```
  VALUES(OLD. emp_num, OLD.valid_from, OLD.amount);
```

```
END $$
```

```
DELETE FROM salaries WHERE emp_num = 105;
```

```
DELIMITER $$
```

```
CREATE TRIGGER after_delete_salaries
```

```
AFTER DELETE
```

```
ON salaries FOR EACH ROW
```

```
BEGIN
```

```
  UPDATE total_salary_budget SET total_budget = total_budget - old.amount;
```

**END\$\$**

User create:

Create user 'test1'@'localhost' Identified by '1223';

Grant all privileges

On databaseName.tableName

To 'test'@'localhost';

Flush privileges

Drop user 'test1'@'localhost'