* 23-01-2025
* Basic commands of MySQL

**1. Create a Database**

To create a new database in MySQL, use the CREATE DATABASE statement.

CREATE DATABASE my\_database;

**2. Use a Database**

After creating a database, you can select it to start working within it.

USE my\_database;

**3. Create a Table**

To create a table, specify the table name and its columns along with data types.

CREATE TABLE users (

id INT AUTO\_INCREMENT PRIMARY KEY,

first\_name VARCHAR(50),

last\_name VARCHAR(50),

email VARCHAR(100) UNIQUE

);

**4. Insert Data into a Table**

Use the INSERT INTO statement to add new rows to a table.

INSERT INTO users (first\_name, last\_name, email)

VALUES ('Kashish', 'Choudhary', 'kashish@example.com');

**5. Select Data from a Table**

The SELECT statement retrieves data from a table.

* Retrieve all columns:
* SELECT \* FROM users;
* Retrieve specific columns:
* SELECT first\_name, last\_name FROM users;
* Retrieve data with a condition:
* SELECT \* FROM users WHERE first\_name = 'John';

**6. Update Data in a Table**

The UPDATE statement modifies existing data in a table.

UPDATE users

SET email = 'kashish@newdomain.com'

WHERE id = 1;

**7. Delete Data from a Table**

The DELETE statement removes rows from a table.

DELETE FROM users WHERE id = 1;

**8. Add a New Column to a Table**

Use the ALTER TABLE statement to add a column to an existing table.

ALTER TABLE users ADD age INT;

**9. Drop a Column from a Table**

To remove a column from a table, use the ALTER TABLE statement with DROP COLUMN.

ALTER TABLE users DROP COLUMN age;

**10. Drop a Table**

To delete an entire table and its data, use the DROP TABLE statement.

DROP TABLE users;

**11. Join Tables**

You can join two tables using a common column to retrieve related data.

* **Inner Join**: Returns rows with matching values in both tables.
* SELECT users.first\_name, orders.order\_id
* FROM users
* INNER JOIN orders ON users.id = orders.user\_id;
* **Left Join**: Returns all rows from the left table, and matching rows from the right table.
* SELECT users.first\_name, orders.order\_id
* FROM users
* LEFT JOIN orders ON users.id = orders.user\_id;

**12. Group Data**

Use the GROUP BY statement to group rows that have the same values in specified columns.

SELECT COUNT(\*), age FROM users GROUP BY age;

**13. Aggregate Functions**

* **COUNT**: Returns the number of rows.
* SELECT COUNT(\*) FROM users;
* **AVG**: Returns the average value.
* SELECT AVG(age) FROM users;
* **MAX**: Returns the maximum value.
* SELECT MAX(age) FROM users;
* **MIN**: Returns the minimum value.
* SELECT MIN(age) FROM users;

**14. Create a Foreign Key**

A foreign key is used to link two tables together.

CREATE TABLE orders (

order\_id INT AUTO\_INCREMENT PRIMARY KEY,

user\_id INT,

order\_date DATE,

FOREIGN KEY (user\_id) REFERENCES users(id)

);

**15. Drop a Database**

To remove a database and all its tables, use the DROP DATABASE statement.

DROP DATABASE my\_database;

**16. Show Tables**

To list all the tables in the current database, use the SHOW TABLES statement.

SHOW TABLES;

**17. Show Table Structure**

To view the structure (columns and data types) of a table, use DESCRIBE.

DESCRIBE users;

**18. Rename a Table**

To rename an existing table, use the RENAME TABLE statement.

RENAME TABLE users TO customers;