**Creating and Altering Tables in MySQL**

In MySQL, you can create and modify database tables using the CREATE TABLE and ALTER TABLE statements. Here's a detailed explanation with examples:

**1. Creating Tables**

The **CREATE TABLE** statement is used to define a new table.

**Syntax**

sql

Copy code

CREATE TABLE table\_name (

column\_name1 data\_type constraints,

column\_name2 data\_type constraints,

...

);

**Example: Create a employees Table**

sql

Copy code

CREATE TABLE employees (

id INT AUTO\_INCREMENT PRIMARY KEY,

name VARCHAR(100) NOT NULL,

department\_id INT,

salary DECIMAL(10, 2),

hire\_date DATE

);

**Explanation:**

* id: An integer column with auto-increment for unique IDs, set as the primary key.
* name: A variable-length string, cannot be NULL.
* department\_id: An integer to link with another table (foreign key).
* salary: A decimal value with up to 10 digits, 2 after the decimal point.
* hire\_date: A date field.

**2. Adding Constraints**

You can add constraints while creating the table to ensure data integrity.

**Common Constraints**

| **Constraint** | **Description** |
| --- | --- |
| PRIMARY KEY | Uniquely identifies a row. |
| NOT NULL | Prevents null values. |
| UNIQUE | Ensures all values in the column are unique. |
| FOREIGN KEY | Establishes a relationship with another table. |
| CHECK | Ensures column values meet a condition (MySQL 8+). |
| DEFAULT | Sets a default value for the column. |

**Example: Add Constraints**

sql

Copy code

CREATE TABLE departments (

id INT AUTO\_INCREMENT PRIMARY KEY,

department\_name VARCHAR(50) NOT NULL UNIQUE,

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP

);

**3. Altering Tables**

The **ALTER TABLE** statement is used to modify an existing table.

**Syntax**

sql

Copy code

ALTER TABLE table\_name

ADD column\_name data\_type constraints;

ALTER TABLE table\_name

DROP COLUMN column\_name;

ALTER TABLE table\_name

MODIFY COLUMN column\_name data\_type constraints;

ALTER TABLE table\_name

ADD CONSTRAINT constraint\_name constraint\_definition;

**Examples of ALTER TABLE**

1. **Add a Column**

sql

Copy code

ALTER TABLE employees

ADD email VARCHAR(100);

1. **Drop a Column**

sql

Copy code

ALTER TABLE employees

DROP COLUMN email;

1. **Modify a Column**

sql

Copy code

ALTER TABLE employees

MODIFY COLUMN salary DECIMAL(12, 2) NOT NULL;

1. **Add a Foreign Key**

sql

Copy code

ALTER TABLE employees

ADD CONSTRAINT fk\_department

FOREIGN KEY (department\_id) REFERENCES departments(id);

**4. Renaming and Dropping Tables**

**Rename a Table**

sql

Copy code

RENAME TABLE employees TO staff;

**Drop a Table**

sql

Copy code

DROP TABLE employees;

**5. Advanced Alterations**

1. **Rename a Column** (MySQL 8+)

sql

Copy code

ALTER TABLE employees

RENAME COLUMN name TO full\_name;

1. **Add Multiple Columns**

sql

Copy code

ALTER TABLE employees

ADD phone\_number VARCHAR(15),

ADD address TEXT;

1. **Change Table Engine**

sql

Copy code

ALTER TABLE employees

ENGINE = InnoDB;

**6. Viewing Table Schema**

To see the structure of a table, use:

sql

Copy code

DESCRIBE employees;

Or, for more detailed information:

sql

Copy code

SHOW CREATE TABLE employees;

**Example Workflow: Create and Modify a Table**

**Step 1: Create a Table**

sql

Copy code

CREATE TABLE products (

product\_id INT AUTO\_INCREMENT PRIMARY KEY,

product\_name VARCHAR(100) NOT NULL,

price DECIMAL(10, 2) NOT NULL,

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP

);

**Step 2: Add a New Column**

sql

Copy code

ALTER TABLE products

ADD stock INT DEFAULT 0;

**Step 3: Modify a Column**

sql

Copy code

ALTER TABLE products

MODIFY COLUMN price DECIMAL(12, 2) NOT NULL;

**Step 4: Add a Unique Constraint**

sql

Copy code

ALTER TABLE products

ADD CONSTRAINT unique\_product\_name UNIQUE (product\_name);