Customer Management Database (With MySQL).

# 1. Create Database

This line creates a new database called `customer\_management`. A database is like a folder that contains various tables where you store information. In this case, the database will store customer-related data.

```sql  
CREATE DATABASE customer\_management;  
USE customer\_management;  
```

# 2. Creating the Customers Table

The `Customers` table stores information about each customer. Here is the breakdown of each field:  
- `CustomerID`: A unique ID for each customer. Automatically generated and cannot be duplicated.  
- `Name`: Stores the customer's name (up to 100 characters).  
- `Email`: Stores the customer's email address (up to 100 characters).  
- `City, State, Country`: These fields store the customer's location (up to 100 characters each).  
- `Phone`: Stores the customer's phone number (up to 30 characters).  
- `Address`: Stores the customer's address (up to 255 characters).

```sql  
CREATE TABLE Customers (  
 CustomerID INT PRIMARY KEY AUTO\_INCREMENT,  
 Name VARCHAR(100),  
 Email VARCHAR(100),  
 City VARCHAR(100),  
 State VARCHAR(100),  
 Country VARCHAR(100),  
 Phone VARCHAR(30),  
 Address VARCHAR(255)  
);  
```

# 3. Creating the Orders Table

The `Orders` table tracks customer orders and links them to the `Customers` table using `CustomerID`.  
- `OrderID`: A unique ID for each order. Automatically generated.  
- `CustomerID`: The ID of the customer placing the order (linked to the `Customers` table).  
- `Order\_Date`: The date the order was made.  
Foreign key ensures each order is tied to an existing customer.

```sql  
CREATE TABLE Orders (  
 OrderID INT PRIMARY KEY AUTO\_INCREMENT,  
 CustomerID INT,  
 Order\_Date DATE,  
 FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID)  
);  
```

# 4. View Tables

After creating the tables, you can check if they exist by running the following command:  
```sql  
SHOW TABLES;  
```

# 5. Setting MySQL Root Password

To secure your MySQL instance, you can set a password for the `root` (administrator) user. This example sets the password to 'root'.  
```sql  
ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql\_native\_password BY 'root';  
FLUSH PRIVILEGES;  
```