



Home

 Home

Learning Modules

 Basic Queries

 DDL Commands

 **DML Operations**

 DCL Controls


 TCL Management


 Aggregate Functions


Tools

 SQL Query Editor


 Quick Links

 Documentation

 Tips & Tricks

 Learning Path

MySQL Handbook v2.0

Built with  using Streamlit

+ INSERT Best Practices:

- Use bulk INSERT for multiple rows •
- Specify column names explicitly • Use



MySQL Handbook

Complete Guide to MySQL Database



Data Manipulation Language (DML) ↔

Master the art of data manipulation with INSERT, UPDATE, DELETE, and data transfer operations.

+ INSERT Operations

 UPDATE Operations

 DELETE Operations

 Import/Export

 Practice Lab

+ INSERT Operations

Data Insertion Techniques

Single Row INSERT:

```
INSERT INTO customers (first_name, last_name, email)
VALUES ('John', 'Doe', 'john.doe@email.com');
```

Multiple Rows INSERT:

```
INSERT INTO customers (first_name, last_name, email)
VALUES
    ('Jane', 'Smith', 'jane.smith@email.com'),
    ('Mike', 'Johnson', 'mike.johnson@email.com'),
    ('Sarah', 'Davis', 'sarah.davis@email.com');
```

INSERT with SELECT:


```
INSERT INTO archived_customers (customer_id, full_name, email)
SELECT customer_id,
       CONCAT(first_name, ' ', last_name),
       email
FROM customers
WHERE status = 'inactive';
```

INSERT IGNORE:

```
INSERT IGNORE INTO customers (email, first_name, last_name)
VALUES ('existing@email.com', 'Test', 'User');

-- Prevents duplicate key errors
```

Interactive Data Insertion

Customer Data Entry Form 

First Name:

John

Phone:

+1-555-123-4567

Last Name:

Doe

Status:


active 

Email:

john.doe@example.com

Generate INSERT Statement

Bulk Data Generator 

 **Safety First:** Always backup your data before performing bulk DML operations!