

MySQL Command Categories: DDL, DML, DRL, TCL, DCL

Overview

This document explains the five major categories of SQL commands in MySQL—Data Definition Language (DDL), Data Manipulation Language (DML), Data Retrieval Language (DRL), Transaction Control Language (TCL), and Data Control Language (DCL). For each category, you'll find definitions, common commands, usage scenarios, and practical examples.

1) DDL — Data Definition Language

- Purpose: Define and modify schema objects (databases, tables, indexes, views).
- Characteristics: Autocommit by default; changes are structural and often irreversible without backups.

Common DDL Commands

- CREATE: Create database objects (CREATE DATABASE, CREATE TABLE, CREATE VIEW, CREATE INDEX).
- ALTER: Modify existing objects (add/drop columns, change data types, rename).
- DROP: Remove objects (tables, views, databases).
- TRUNCATE: Remove all rows in a table quickly (resets AUTO_INCREMENT).
- RENAME: Rename tables or other objects.

Usage Examples

```
-- Create a database and a table
CREATE DATABASE shop;
USE shop;

CREATE TABLE products (
    product_id INT PRIMARY KEY AUTO_INCREMENT,
    name VARCHAR(100) NOT NULL,
    price DECIMAL(10,2) NOT NULL,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);

-- Add a column
ALTER TABLE products ADD COLUMN stock INT DEFAULT 0;

-- Change a column type
ALTER TABLE products MODIFY price DECIMAL(12,2);

-- Create an index
CREATE INDEX idx_products_name ON products(name);

-- Truncate and drop
TRUNCATE TABLE products;
DROP TABLE products;
```

2) DML — Data Manipulation Language

- Purpose: Insert, update, delete, and merge data within tables.
- Characteristics: Row-level data changes; can be transactional when inside explicit transactions.

Common DML Commands

- INSERT (and INSERT ... SELECT): Add new rows.
- UPDATE: Modify existing rows.
- DELETE: Remove rows based on conditions.
- REPLACE: Insert or replace a row by primary/unique key (MySQL-specific).

Usage Examples

```
USE shop;

-- Insert rows
INSERT INTO products(name, price, stock) VALUES
('Laptop', 79999.00, 10),
('Mouse', 599.00, 100);

-- Update rows
UPDATE products SET stock = stock - 1 WHERE name = 'Laptop';

-- Delete rows
DELETE FROM products WHERE price < 500;

-- Insert from another query
INSERT INTO products(name, price, stock)
SELECT name, price, 0 FROM archived_products;

-- Replace a row (MySQL)
REPLACE INTO products(product_id, name, price, stock)
VALUES (1, 'Laptop', 74999.00, 9);
```

3) DRL — Data Retrieval Language (SELECT)

- Purpose: Query and retrieve data. Often referred to as part of DML in some literature, but commonly singled out as DRL.
- Characteristics: Non-destructive; supports filtering, sorting, grouping, joins, and window functions.

Common SELECT Patterns

- Basic SELECT with WHERE and ORDER BY.
- Aggregations with GROUP BY and HAVING.
- JOINS: INNER, LEFT, RIGHT, CROSS.
- Subqueries and Common Table Expressions (CTEs with WITH, MySQL 8.0+).
- Window functions (MySQL 8.0+): ROW_NUMBER(), RANK(), SUM() OVER (...).

Usage Examples

```

USE shop;

-- Basic select
SELECT product_id, name, price FROM products
WHERE price BETWEEN 1000 AND 100000
ORDER BY price DESC;

-- Aggregation
SELECT name, SUM(stock) AS total_stock
FROM products
GROUP BY name
HAVING SUM(stock) > 5;

-- Join with categories
SELECT p.name, c.category_name
FROM products p
INNER JOIN categories c ON p.category_id = c.id;

-- CTE (MySQL 8.0+)
WITH high_value AS (
    SELECT * FROM products WHERE price > 50000
)
SELECT name FROM high_value;

-- Window function (MySQL 8.0+)
SELECT name, price,
       RANK() OVER (ORDER BY price DESC) AS price_rank
FROM products;

```

4) TCL — Transaction Control Language

- Purpose: Manage transactions to ensure ACID properties (Atomicity, Consistency, Isolation, Durability).
- Characteristics: Works with transactional storage engines (e.g., InnoDB).

Common TCL Commands

- START TRANSACTION / BEGIN: Begin a transaction.
- COMMIT: Persist all changes made during the transaction.
- ROLLBACK: Undo changes made during the transaction.
- SAVEPOINT and ROLLBACK TO SAVEPOINT: Partial rollbacks within a transaction.
- SET TRANSACTION: Set isolation level and access mode.

Usage Examples

```

USE shop;

-- Transfer stock from one product to another atomically
START TRANSACTION;
UPDATE products SET stock = stock - 5 WHERE product_id = 1;
UPDATE products SET stock = stock + 5 WHERE product_id = 2;
-- Validate business rule
SELECT stock FROM products WHERE product_id = 1;
-- If stock < 0 then ROLLBACK, else COMMIT
COMMIT;

```

```

-- Savepoints example
START TRANSACTION;
UPDATE products SET price = price * 0.9; -- 10% discount
SAVEPOINT after_discount;
UPDATE products SET price = price * 0.95; -- additional 5%
ROLLBACK TO SAVEPOINT after_discount; -- cancel 5% but keep 10%
COMMIT;

-- Isolation level
SET TRANSACTION ISOLATION LEVEL READ COMMITTED;

```

5) DCL — Data Control Language

- Purpose: Control access and permissions to database objects.
- Characteristics: Security-focused; changes are typically autocommitted and audited.

Common DCL Commands (MySQL)

- CREATE USER / ALTER USER / DROP USER: Manage accounts.
- GRANT: Assign privileges/roles to users.
- REVOKE: Remove privileges.
- SET PASSWORD: Change user password (or ALTER USER).

Usage Examples

```

-- Create users
CREATE USER 'reporter'@'%' IDENTIFIED BY 'StrongP@ss!';
CREATE USER 'app_user'@'localhost' IDENTIFIED WITH
mysql_native_password BY 'AnotherStr0ng!';

-- Grant privileges
GRANT SELECT ON shop.* TO 'reporter'@'%';
GRANT SELECT, INSERT, UPDATE, DELETE ON shop.* TO
'app_user'@'localhost';

-- Revoke a privilege
REVOKE DELETE ON shop.* FROM 'app_user'@'localhost';

-- Drop a user
DROP USER 'reporter'@'%';

```

Best Practices & Notes

- Use InnoDB for transactional workloads to leverage TCL properly.
- Prefer explicit transactions for multi-step DML operations; avoid mixing DDL in the same transaction.
- Back up schema before destructive DDL (DROP/TRUNCATE).
- Use least-privilege principle with DCL (grant only what is necessary).
- Version-control schema changes with migration tools (e.g., Liquibase, Flyway).
- Validate MySQL version features (e.g., CTEs and window functions require MySQL 8.0+).

Where Do We Use Each Category?

- DDL: During initial database design, feature rollouts, and schema migrations.
- DML: In application code and scripts that create/update/delete operational data.
- DRL (SELECT): Reporting, analytics, dashboards, and application reads.
- TCL: Ensuring data consistency across multi-step operations (payments, inventory updates).
- DCL: User management, security hardening, and audit configurations.