

접속 및 데이터베이스 관리 (Connection & DB)

MySQL은 널리 사용되는 오픈소스 관계형 데이터베이스 관리 시스템(RDBMS)입니다.

- 터미널 접속: `mysql -u username -p -h hostname -P 3306`
- 데이터베이스 목록 확인: `SHOW DATABASES;`
- 데이터베이스 생성/삭제: `CREATE DATABASE mydb CHARACTER SET utf8mb4; / DROP DATABASE mydb;`
- 데이터베이스 선택: `USE mydb;`
- 테이블 목록/구조 확인: `SHOW TABLES; / DESCRIBE users; (또는 EXPLAIN users;)`

데이터 정의어 (DDL)

테이블 생성 (CREATE TABLE)

```
CREATE TABLE users (
  id INT AUTO_INCREMENT PRIMARY KEY,
  username VARCHAR(50) NOT NULL UNIQUE,
  email VARCHAR(100) NOT NULL,
  age TINYINT UNSIGNED DEFAULT 0,
  created_at DATETIME DEFAULT
CURRENT_TIMESTAMP,
  INDEX idx_email (email)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

테이블 수정 (ALTER TABLE)

- 컬럼 추가: `ALTER TABLE users ADD COLUMN phone VARCHAR(20) AFTER email;`
- 컬럼 수정: `ALTER TABLE users MODIFY COLUMN username VARCHAR(100) NOT NULL;`
- 컬럼 삭제: `ALTER TABLE users DROP COLUMN phone;`
- 인덱스 추가/삭제: `CREATE INDEX idx_user_age ON users(age); / DROP INDEX idx_user_age ON users;`

데이터 조작어 (DML)

INSERT (데이터 삽입)

```
INSERT INTO users (username, email, age)
VALUES ('alice', 'alice@example.com', 25),
       ('bob', 'bob@example.com', 30);
```

```
-- 중복 키 업데이트 (Upsert)
INSERT INTO users (id, username, email)
VALUES (1, 'alice', 'a@a.com')
ON DUPLICATE KEY UPDATE email =
VALUES(email);
```

SELECT (조회 및 필터링)

```
SELECT id, username, age FROM users
WHERE age >= 20 AND email LIKE
'%@gmail.com'
ORDER BY age DESC, created_at ASC
LIMIT 10 OFFSET 20; -- 3번째 페이지
(21~30번째)
```

UPDATE & DELETE

- `UPDATE users SET age = age + 1, email = 'new@a.com' WHERE id = 1;`
- `DELETE FROM users WHERE age < 18;`
- `TRUNCATE TABLE logs;` (테이블 데이터 전체 고속 초기화)

조인과 집계 (Joins & Aggregation)

JOIN 연산

- **INNER JOIN:** 양쪽 테이블 모두 일치하는 행 반환

```
SELECT u.username, p.title
FROM users u
INNER JOIN posts p ON u.id =
p.user_id;
```
- **LEFT JOIN:** 왼쪽 테이블 기준 전체 행 + 오른쪽 일치 행

```
SELECT u.username, COUNT(p.id) AS
post_count
FROM users u
LEFT JOIN posts p ON u.id = p.user_id
GROUP BY u.id, u.username
HAVING post_count >= 5;
```

주요 집계 및 내장 함수

- 집계: `COUNT(*)`, `SUM(val)`, `AVG(val)`, `MIN(val)`, `MAX(val)`
- 문자열: `CONCAT(a, ' ', b)`, `SUBSTRING(str, pos, len)`, `LENGTH(str)`

- 날짜/시간: `NOW()`, `CURDATE()`, `DATE_ADD(NOW(), INTERVAL 7 DAY)`, `DATE_FORMAT(NOW(), '%Y-%m-%d')`
- 조건 제어: `COALESCE(val, '기본값')`, `IF(cond, true_val, false_val)`
- CASE문:

```
SELECT username,
CASE
  WHEN age >= 65 THEN 'Senior'
  WHEN age >= 20 THEN 'Adult'
  ELSE 'Youth'
END AS age_group
FROM users;
```

트랜잭션 및 잠금 (Transactions)

```
START TRANSACTION; -- 또는 BEGIN;
UPDATE accounts SET balance = balance -
100 WHERE id = 1;
UPDATE accounts SET balance = balance +
100 WHERE id = 2;
COMMIT; -- 변경사항 확정 (취소 시
ROLLBACK;)
```

사용자 및 권한 관리 (User & Grants)

- 사용자 생성: `CREATE USER 'dev'@'%' IDENTIFIED BY 'password123';`
- 권한 부여: `GRANT ALL PRIVILEGES ON mydb.* TO 'dev'@'%';`
- 권한 회수/적용: `REVOKE ALL ON mydb.* FROM 'dev'@'%'; / FLUSH PRIVILEGES;`

백업 및 복구 (mysqldump)

- 백업: `mysqldump -u root -p mydb > backup.sql`
- 복구: `mysql -u root -p mydb < backup.sql`