

MySQL 빠른 참조

데이터베이스, 테이블, CRUD, 조인, 인덱스, 사용자

연결

커맨드 라인

```
mysql -u root -p
mysql -u user -p -h 127.0.0.1 -P 3306
mysql -u user -p mydb < dump.sql
```

연결 문자열

```
mysql://user:password@host:3306/dbname
mysql -u user -p -e "SELECT VERSION();" 
```

상태 명령어

```
STATUS;
SHOW VARIABLES LIKE 'port';
SHOW PROCESSLIST;
```

데이터베이스 및 테이블

데이터베이스 작업

```
CREATE DATABASE mydb CHARACTER SET utf8mb4;
SHOW DATABASES;
USE mydb;
DROP DATABASE mydb;
```

테이블 작업

```
CREATE TABLE users (
  id INT AUTO_INCREMENT PRIMARY KEY,
  name VARCHAR(100) NOT NULL,
  email VARCHAR(255) UNIQUE,
  created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

테이블 정보

```
SHOW TABLES;
DESCRIBE users;
SHOW CREATE TABLE users;
```

테이블 변경

```
ALTER TABLE users ADD COLUMN age INT;
ALTER TABLE users MODIFY COLUMN name VARCHAR(200);
ALTER TABLE users DROP COLUMN age;
ALTER TABLE users RENAME TO customers;
```

데이터 타입

숫자

INT	4바이트 정수 (-20억 ~ 20억)
BIGINT	8바이트 정수
DECIMAL(p, s)	정밀 숫자 (예: DECIMAL(10,2))
FLOAT / DOUBLE	근사 부동소수점
BOOLEAN	TINYINT(1) 별칭

문자열

VARCHAR(n)	최대 n자의 가변 길이 문자열
TEXT	최대 65 KB 텍스트
MEDIUMTEXT	최대 16 MB 텍스트
CHAR(n)	고정 길이 문자열, 패딩 처리
ENUM('a', 'b')	정의된 집합에서 하나의 값

날짜 및 시간

DATE	YYYY-MM-DD
DATETIME	YYYY-MM-DD HH:MM:SS
TIMESTAMP	UTC 저장 날짜시간
TIME	HH:MM:SS
JSON	기본 JSON 문서 타입

CRUD

삽입

```
INSERT INTO users (name, email)
VALUES ('Alice', 'alice@example.com');
```

```
INSERT INTO users (name, email) VALUES
('Bob', 'bob@example.com'),
('Carol', 'carol@example.com');
```

조회

```
SELECT * FROM users WHERE id = 1;
SELECT name, email FROM users
ORDER BY name LIMIT 10 OFFSET 20;
```

업데이트

```
UPDATE users SET email = 'new@example.com'
WHERE id = 1;
UPDATE users SET active = 0
WHERE last_login < '2025-01-01';
```

삭제

```
DELETE FROM users WHERE id = 1;
TRUNCATE TABLE users; -- fast, resets AUTO_INCREMENT
```

Upsert

```
INSERT INTO users (id, name, email)
VALUES (1, 'Alice', 'a@example.com')
ON DUPLICATE KEY UPDATE
name = VALUES(name), email = VALUES(email);
```

조인

조인 유형

INNER JOIN	양쪽 테이블에서 매칭되는 행
LEFT JOIN	왼쪽 전체 + 오른쪽 매칭
RIGHT JOIN	오른쪽 전체 + 왼쪽 매칭
CROSS JOIN	카르테시안 곱
SELF JOIN	자기 자신과 조인

조인 예시

```
SELECT u.name, o.total
FROM users u
INNER JOIN orders o ON u.id = o.user_id;
```

```
SELECT e.name, m.name AS manager
FROM employees e
LEFT JOIN employees m ON e.manager_id = m.id;
```

인덱스

생성 및 삭제

```
CREATE INDEX idx_name ON users(name);
CREATE UNIQUE INDEX idx_email ON users(email);
CREATE INDEX idx_composite
  ON orders(user_id, created_at);
DROP INDEX idx_name ON users;
```

전문 인덱스

```
ALTER TABLE posts ADD FULLTEXT(title, body);
SELECT * FROM posts
WHERE MATCH(title, body) AGAINST('mysql');
```

인덱스 정보

```
SHOW INDEX FROM users;
EXPLAIN SELECT * FROM users WHERE name = 'Alice';
```

함수

문자열 함수

CONCAT(a, b)	문자열 연결
SUBSTRING(s, pos, len)	부분 문자열 추출
UPPER(s) / LOWER(s)	대/소문자 변환
TRIM(s)	앞뒤 공백 제거
LENGTH(s)	바이트 단위 문자열 길이
REPLACE(s, from, to)	발생 횟수 교체

날짜 함수

NOW()	현재 날짜시간
CURDATE()	현재 날짜
DATE_ADD(d, INTERVAL n DAY)	날짜에 간격 추가
DATEDIFF(d1, d2)	일 단위 차이
DATE_FORMAT(d, fmt)	날짜 형식화 (예: '%Y-%m-%d')

집계 함수

```
SELECT COUNT(*), AVG(price), SUM(qty),
  MIN(price), MAX(price)
FROM products
GROUP BY category HAVING COUNT(*) > 5;
```

사용자 및 권한

사용자 관리

```
CREATE USER 'app'@'localhost'
IDENTIFIED BY 'secret';
ALTER USER 'app'@'localhost'
IDENTIFIED BY 'newsecret';
DROP USER 'app'@'localhost';
```

권한 부여

```
GRANT ALL ON mydb.* TO 'app'@'localhost';
GRANT SELECT, INSERT ON mydb.users
TO 'reader'@'%';
REVOKE INSERT ON mydb.users
FROM 'reader'@'%';
FLUSH PRIVILEGES;
```

권한 확인

```
SHOW GRANTS FOR 'app'@'localhost';
SELECT user, host FROM mysql.user;
```

백업 및 복원

mysqldump

```
mysqldump -u root -p mydb > backup.sql
mysqldump -u root -p --all-databases > all.sql
mysqldump -u root -p mydb users > users.sql
```

복원

```
mysql -u root -p mydb < backup.sql
mysql -u root -p -e "SOURCE /path/backup.sql"
```

MySQL 빠른 참조

바이너리 로그

```
SHOW BINARY LOGS;
SHOW BINLOG EVENTS IN 'binlog.000001';
mysqlbinlog binlog.000001 | mysql -u root -p
```

주요 패턴

페이지네이션

```
SELECT * FROM users
ORDER BY id LIMIT 20 OFFSET 40; -- page 3
```

트랜잭션

```
START TRANSACTION;
UPDATE accounts SET balance = balance - 100
WHERE id = 1;
UPDATE accounts SET balance = balance + 100
WHERE id = 2;
COMMIT; -- or ROLLBACK;
```

조건부 삽입

```
INSERT IGNORE INTO users (email, name)
VALUES ('a@ex.com', 'Alice');
-- skips if email already exists (UNIQUE)
```

변수 및 준비된 구문

```
SET @name = 'Alice';
PREPARE stmt FROM
'SELECT * FROM users WHERE name = ?';
EXECUTE stmt USING @name;
DEALLOCATE PREPARE stmt;
```