

# 데브옵스(DevOps)

## 1. DevOps / DBMS 1

## ❖ 수업 목표

- DevOps 의 개념과 핵심요소를 설명할 수 있다.
- SQL 사용을 위한 MySQL 환경을 설정할 수 있다.
- SQL 개념과 분류 4가지에 대해 설명할 수 있다.
- 주요 데이터베이스 객체의 문법을 알고 사용할 수 있다.

## ❖ 세부 목표

- 1.1 DevOps 개요
- 1.2 SQL 환경설정
- 1.3 SQL 개요
- 1.4 데이터베이스 객체

# 1. DevOps 개요

## ❖ DevOps 개요

### ■ 데브옵스(DevOps) 소개

- 소프트웨어의 개발(Development)과 운영(Operations)의 합성어
- 소프트웨어 개발자와 정보기술 전문가 간의 소통, 협업 및 통합을 강조하는 개발 환경이나 문화를 의미

### ■ 데브옵스 목적

- 소프트웨어 개발조직과 운영조직간의 상호 의존적 대응
- 조직이 소프트웨어 제품과 서비스를 빠른 시간에 개발 및 배포하는 것

# 1. DevOps 개요

## ❖ DevOps 개요

### ■ 데브옵스 핵심 요소

- CI/CD (지속적 통합 및 지속적 배포): 자동화된 빌드, 테스트, 배포
- IaC (Infrastructure as Code): 코드로 인프라 관리
  - 인프라 : 시스템 구조 및 체계
    - » 기본적으로 개발이나 서비스를 위해 구성된 Network, DB, Server, Cloud 등을 의미
- 컨테이너 (Docker, Kubernetes): 애플리케이션 격리 및 배포
- 모니터링 : 시스템 상태 감시 및 분석

# 1. DevOps 개요

## ❖ DevOps 개요

### ■ 데브옵스 방식

#### • 1. 지속적 통합(CI)

- 자동화된 빌드 및 테스트가 수행 후, 개발자가 코드 변경 사항을 중앙 리포지토리에 정기적으로 병합하는 소프트웨어 개발 방식
- 지속적 통합의 핵심 목표
  - » 버그를 신속하게 찾아 해결
  - » 소프트웨어 품질을 개선
  - » 새로운 소프트웨어 업데이트를 검증 및 릴리스하는 데 걸리는 시간을 단축

#### • 2. 지속적 전달 / 배포 (CD, Continuous Delivery / Deployment)

- 프로덕션에 릴리스 하기 위한 코드 변경이 자동으로 빌드, 테스트 및 준비되는 소프트웨어 개발 방식
- 빌드 단계 이후의 모든 코드 변경 사항을 테스트 및 프로덕션 환경에 배포함으로써 지속적 통합을 확장
- 지속적 전달이 적절하게 구현되면, 개발자는 언제나 즉시 배포 가능
- 표준화된 테스트 프로세스를 통과한 빌드 아티팩트를 보유
  - » 빌드 아티팩트(생성된 산출물) : 컴파일된 코드, 바이너리 파일, 배포 패키지, 로그 파일 등

# 1. DevOps 개요

## ❖ DevOps 개요

### ■ 데브옵스 방식

#### • 3. 마이크로 서비스

- 단일 애플리케이션을 독립적으로 수행되는 작은 서비스의 집합으로 구축하는 설계 접근 방식

#### • 4. 코드형 인프라스트럭처(IaC, Infrastructure as Code)

- 인프라를 코드로 정의하고 이를 자동화하여 서버, 네트워크, 디비 등을 프로비저닝하고 관리하는 접근 방식
  - » #프로비저닝(Provisioning) 의미는 "제공하는것"
- 어떤 종류의 서비스든 사용자의 요구에 맞게 시스템 자체를 제공 하는 것
  - » 제공하는 것은 인프라 자원이나 서비스, 또는 장비

#### • 5. 모니터링 및 로깅

- 시스템 상태(지표와 로그)를 모니터링하여 애플리케이션 및 인프라 성능이 제품의 최종 사용자 경험에 어떤 영향을 미치는지 확인

# 1. DevOps 개요

## ❖ DevOps 개요

### ■ 데브옵스 방식

- 6. 형상(버전) 관리
  - 관리 시스템(Git 등)을 사용하여 개발자들이 작성한 코드를 기록하고 변경내역을 추적하여 협업하며, 문제 발생시 이전 버전으로 롤백하도록 도움
- 7. 협업 도구
  - 팀간 소통과 협력을 위해 이슈 트래킹 시스템(Jira 등)과 대화도구(Slack 등)을 사용
  - 이러한 도구들은 프로젝트 관리, 이슈 추적, 업무 협업(소통,공유)을 용이하게 함

## 2. SQL 환경 설정

### ❖ DBMS 개요

#### ■ DBMS 소개

- 데이터베이스 관리 시스템(Database Management System)
- 데이터를 저장, 관리, 검색, 수정하는 소프트웨어
- 대표적인 DBMS: MySQL, Oracle, PostgreSQL, SQLite, MongoDB

#### ■ 관계형 데이터베이스 vs 비관계형 데이터베이스

- 관계형 데이터베이스 (RDBMS): 테이블 기반 (예: MySQL, Oracle, PostgreSQL, SQLite)
- 비관계형 데이터베이스 (NoSQL): 문서, 키-값, 컬럼 기반 (예: MongoDB, Redis)



## 2. SQL 환경 설정

### ❖ MySQL 개요

#### ■ MySQL 소개

- 관계형 데이터베이스 관리 시스템(RDBMS)
- 오픈 소스이며 다양한 플랫폼에서 사용 가능
- 주요 특징: 빠른 성능, 확장성, 높은 안정성

#### ■ MySQL 설치

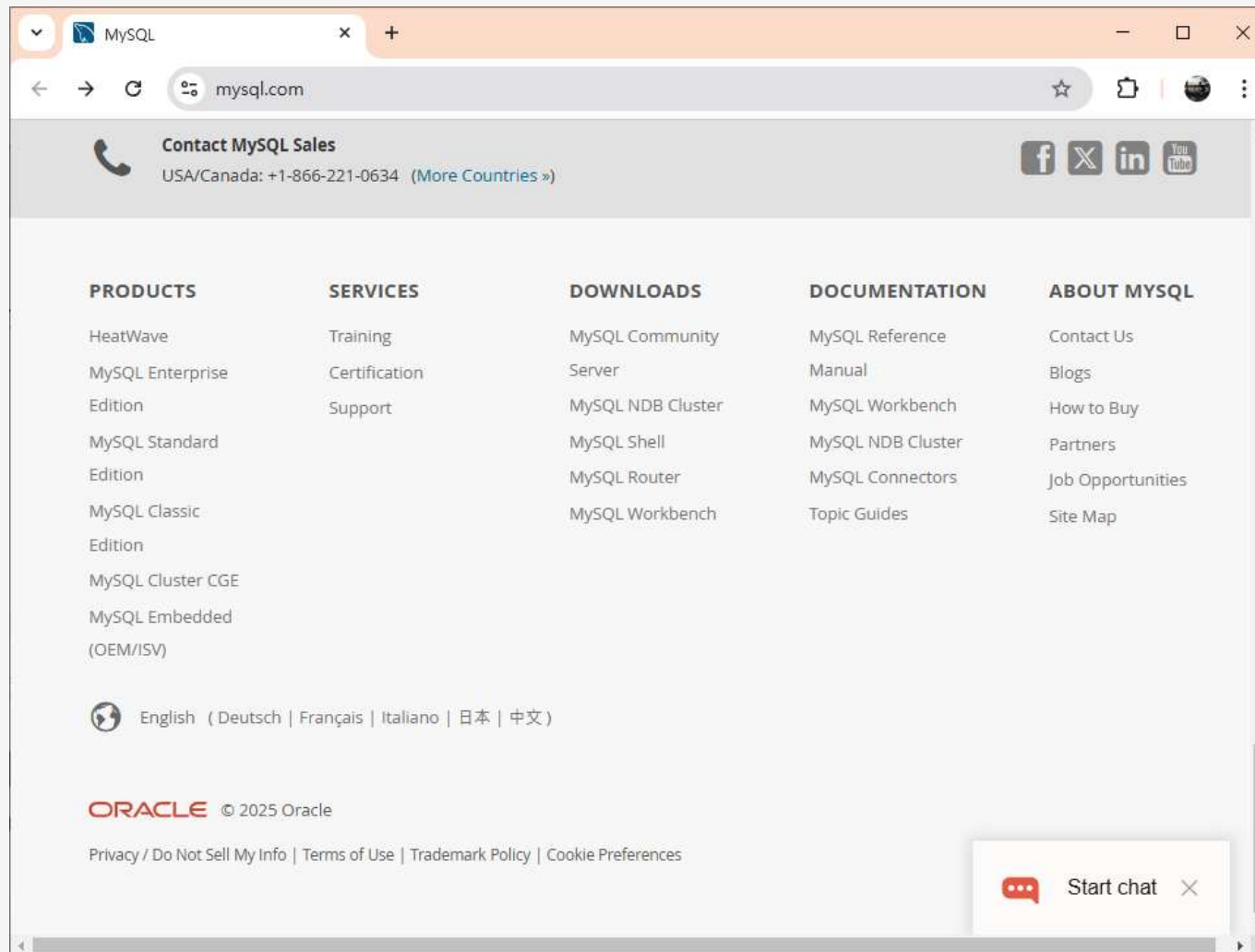
- 홈페이지에서 다운로드
  - <https://www.mysql.com/>



## 2. SQL 환경 설정

### ■ MySQL 설치

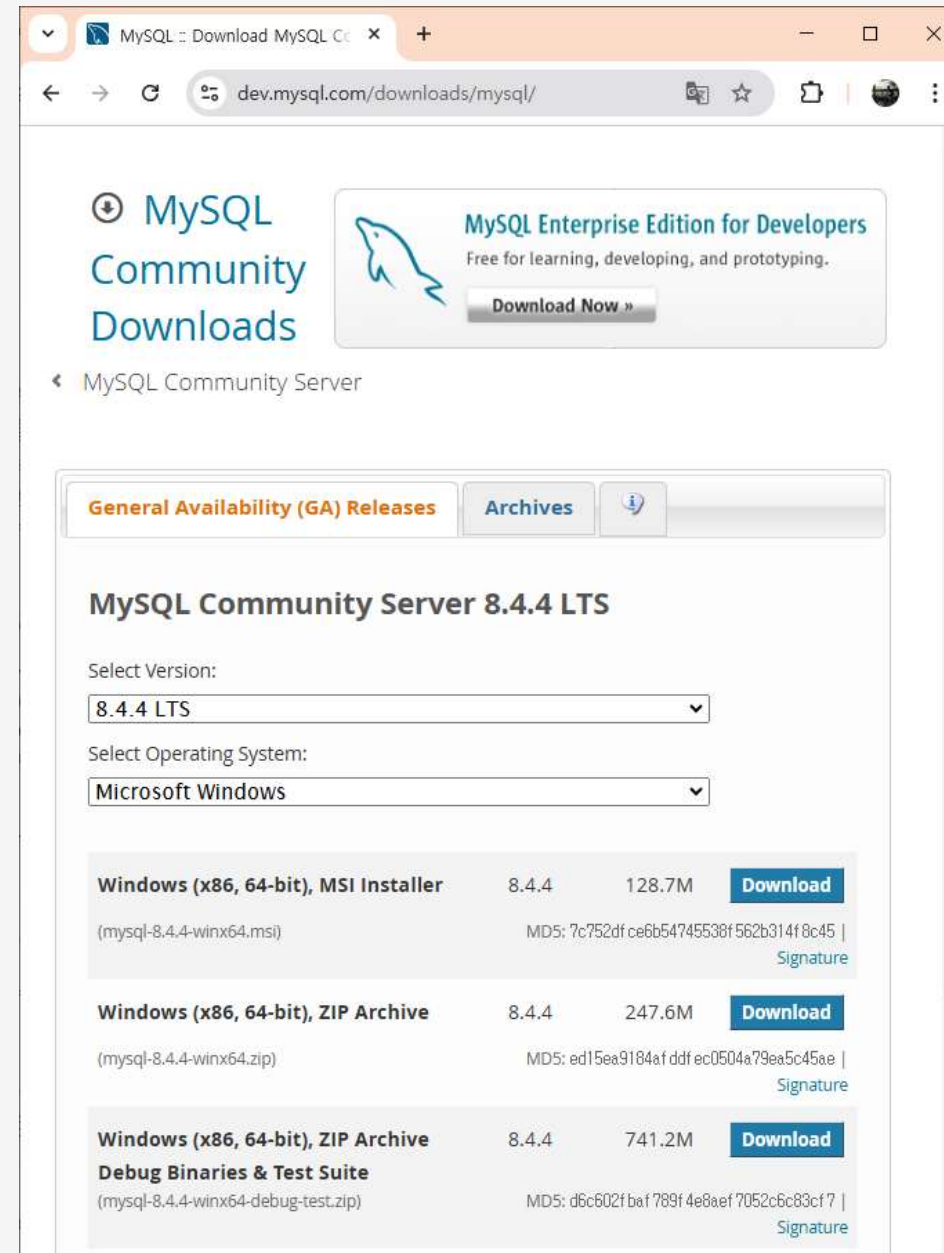
- 홈페이지에서 다운로드
  - 홈페이지 하단 DOWNLOADS 탭의 MySQL Community Server 링크 클릭



## 2. SQL 환경 설정

### ■ MySQL 설치

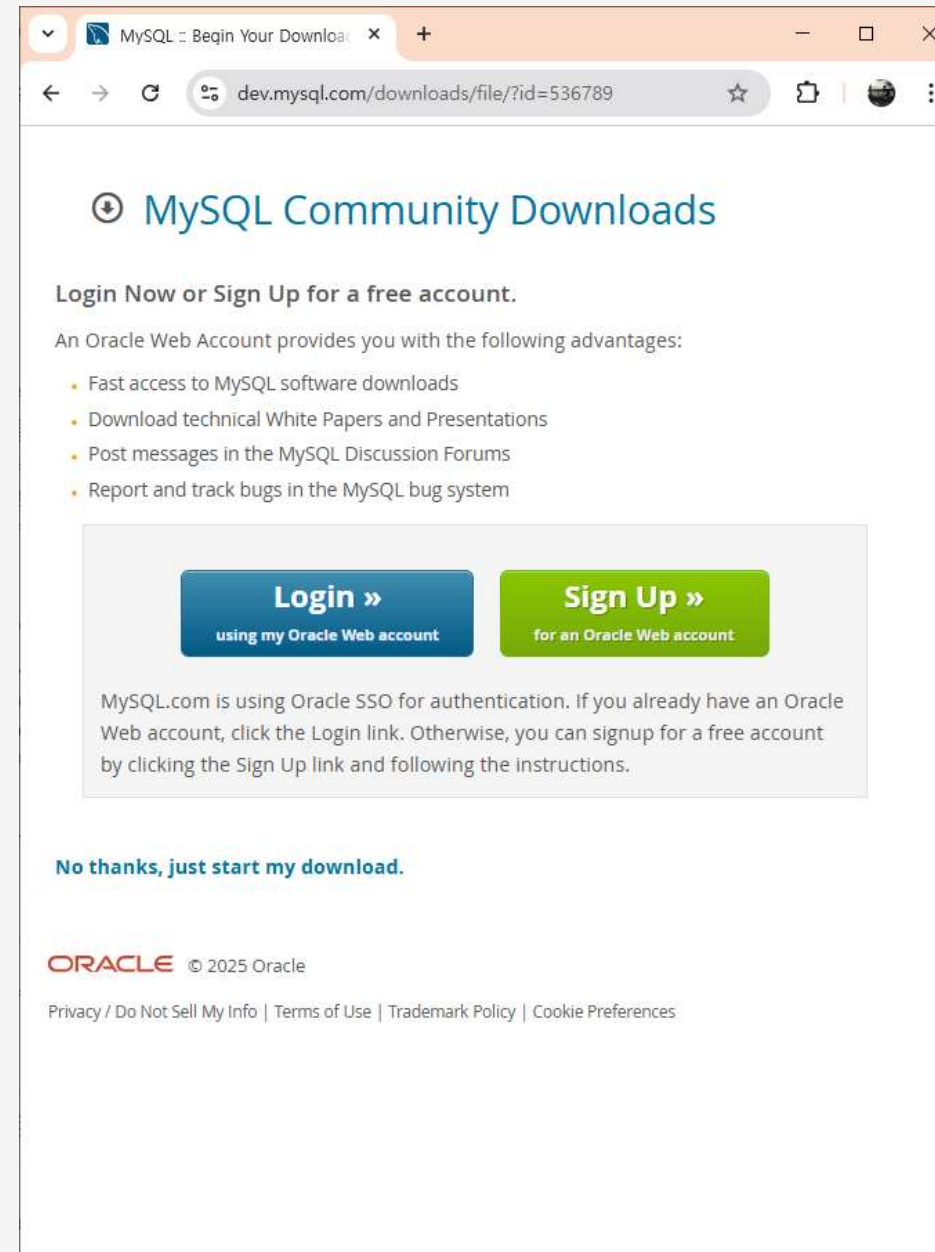
- 홈페이지에서 다운로드
  - 연결된 페이지에서 Select Version 을 LTS(Long Term Support) 버전 선택
- PC 환경을 고려하여 다운로드 버튼 클릭
  - » Windows(x86,64-bit),MSI Installer



## 2. SQL 환경 설정

### ■ MySQL 설치

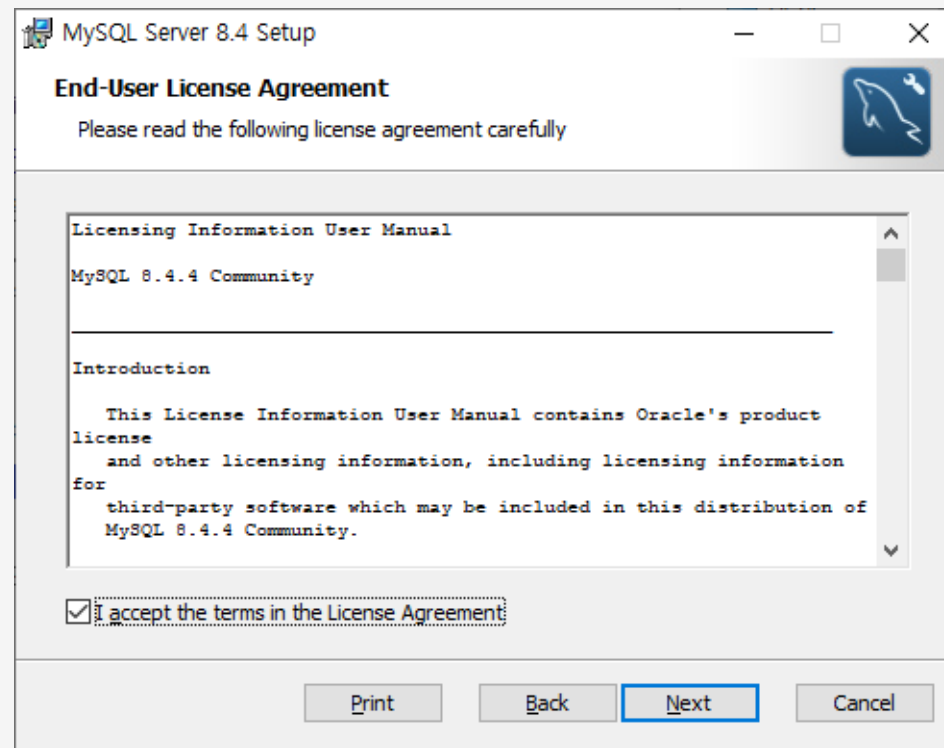
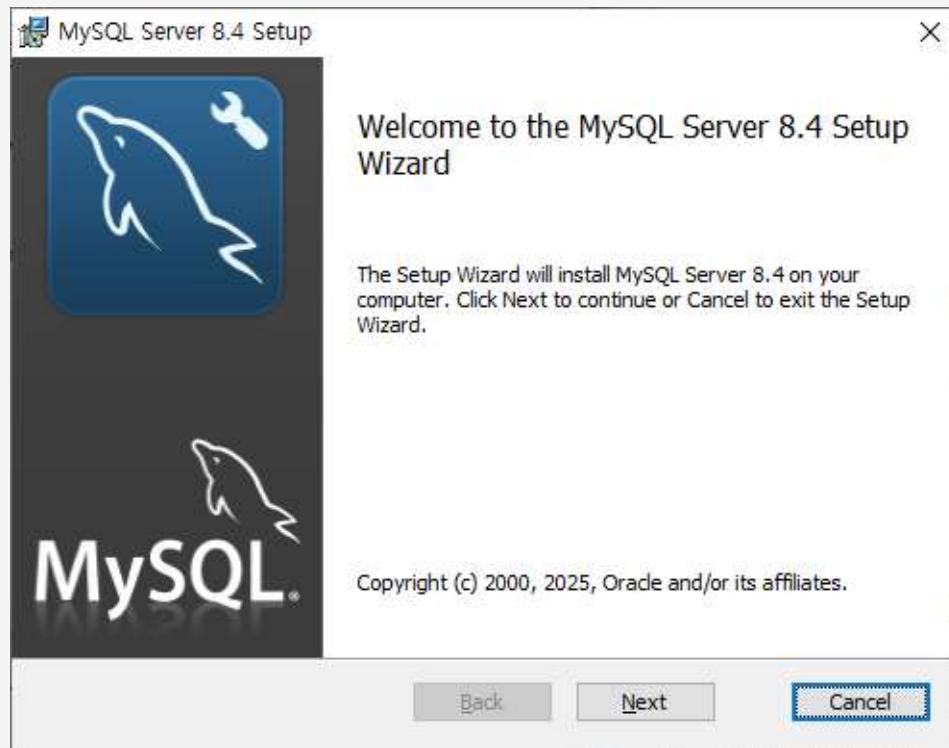
- 홈페이지에서 다운로드
  - 로그인 버튼 아래에 표시된 링크 클릭
    - » No thanks, just start my download.
- 다운로드 수행
  - » mysql-8.4.4-winx64.msi
- 설치 파일 실행



## 2. SQL 환경 설정

### ■ MySQL 설치

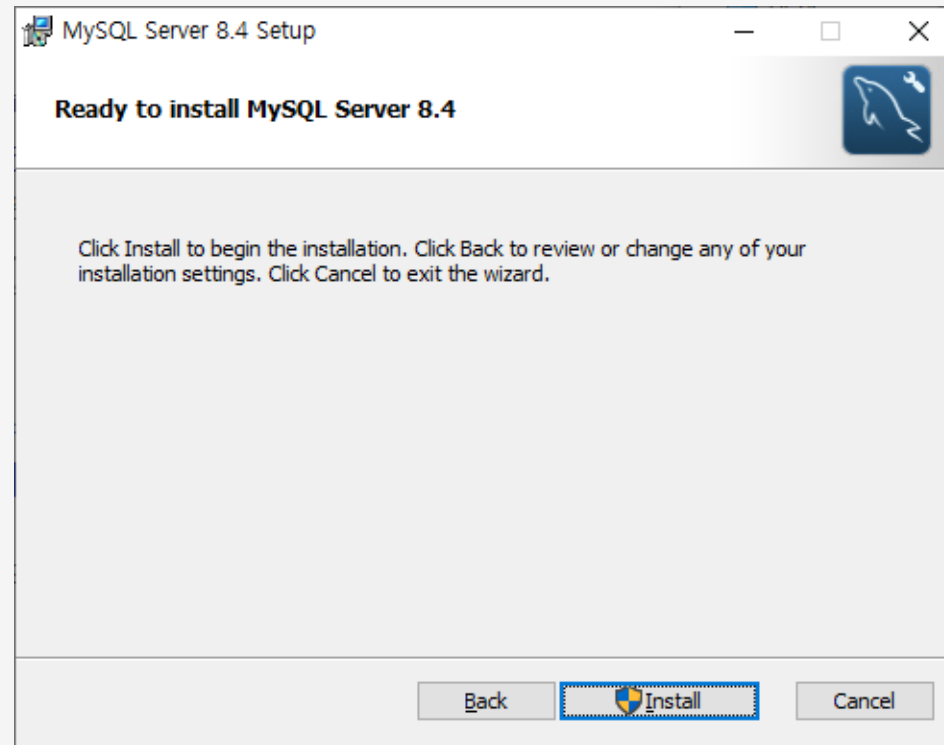
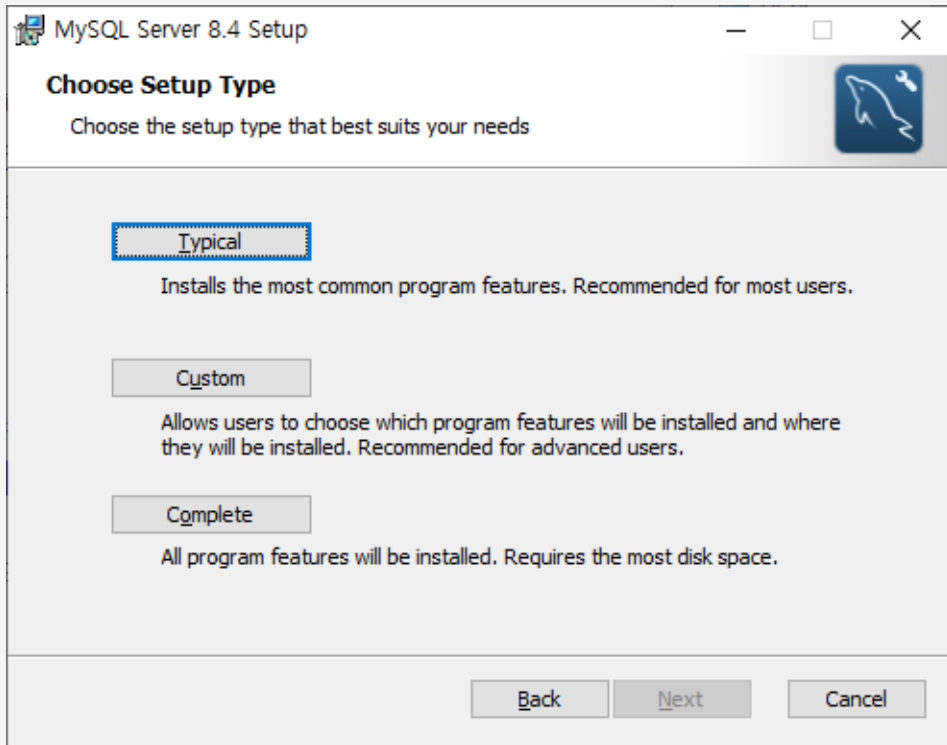
- 설치 파일 실행
  - 다음(Next) 선택
  - 라이선스 동의 체크 => 다음(Next) 선택



## 2. SQL 환경 설정

### ■ MySQL 설치

- 설치 파일 실행
  - Typical 선택
  - 설치(Install) 선택 => 중간에 설치를 위한 내용 허용

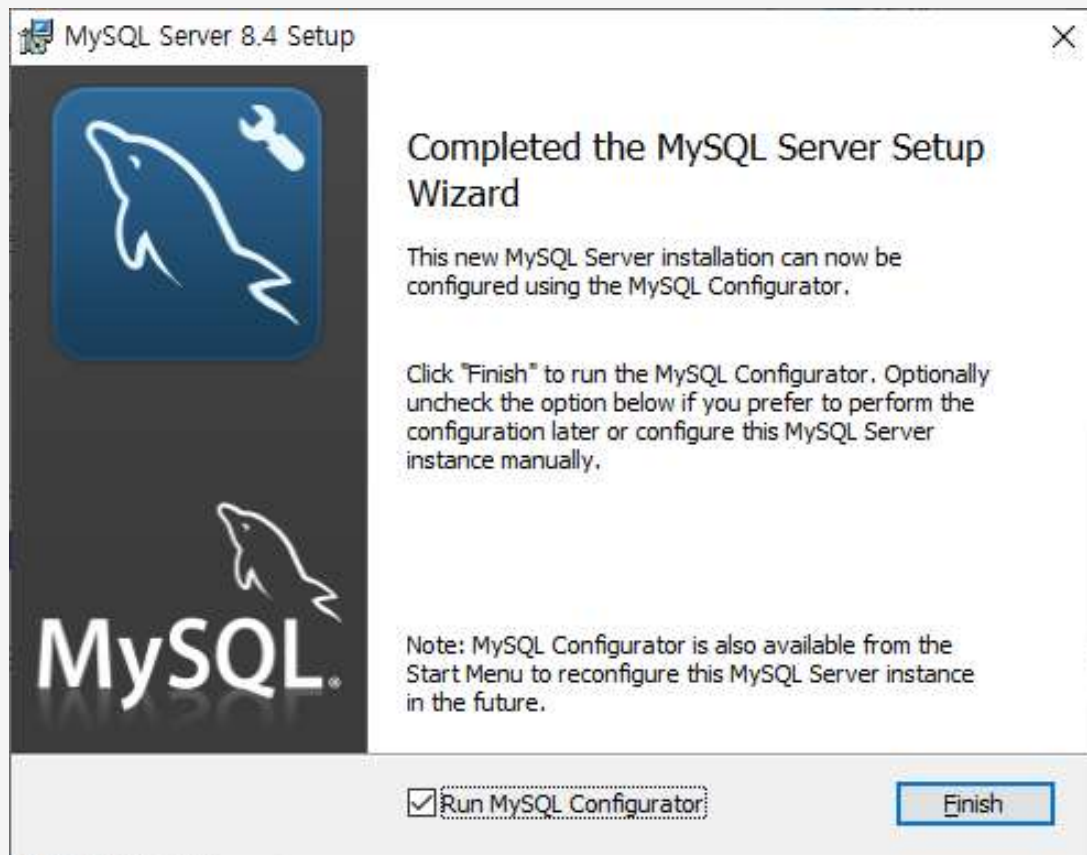


## 2. SQL 환경 설정

### ■ MySQL 설치

- 설치 파일 실행

- Finish 선택 => 중간에 설치를 위한 팝업 창 허용 선택



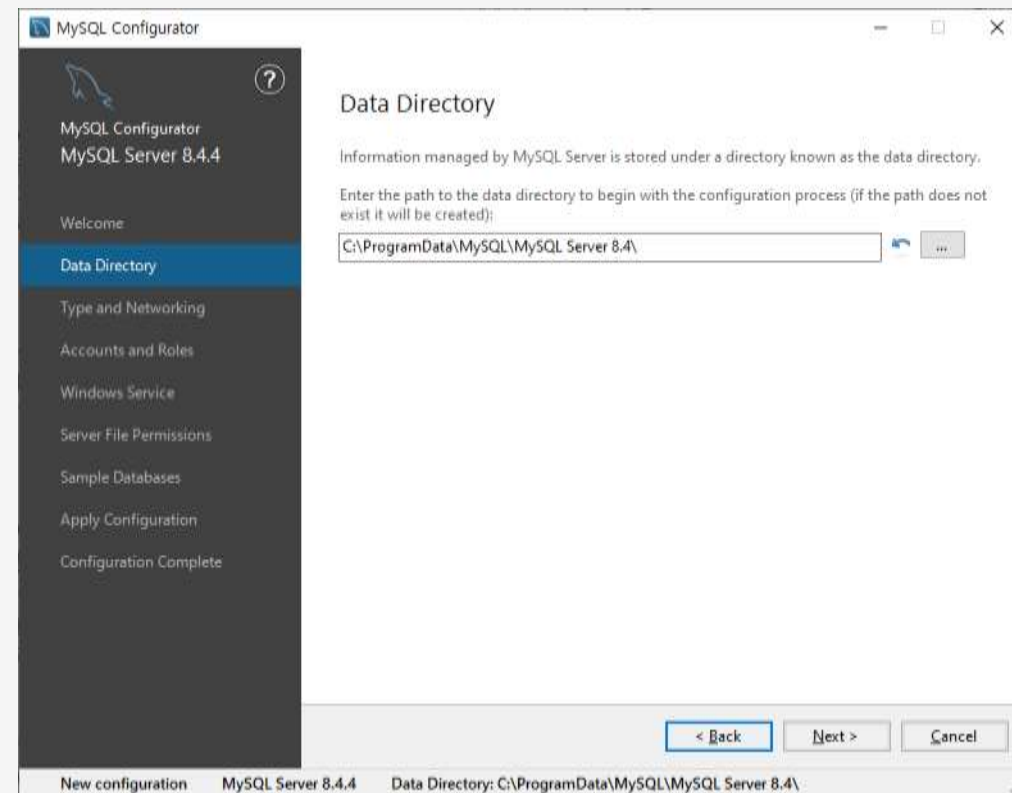
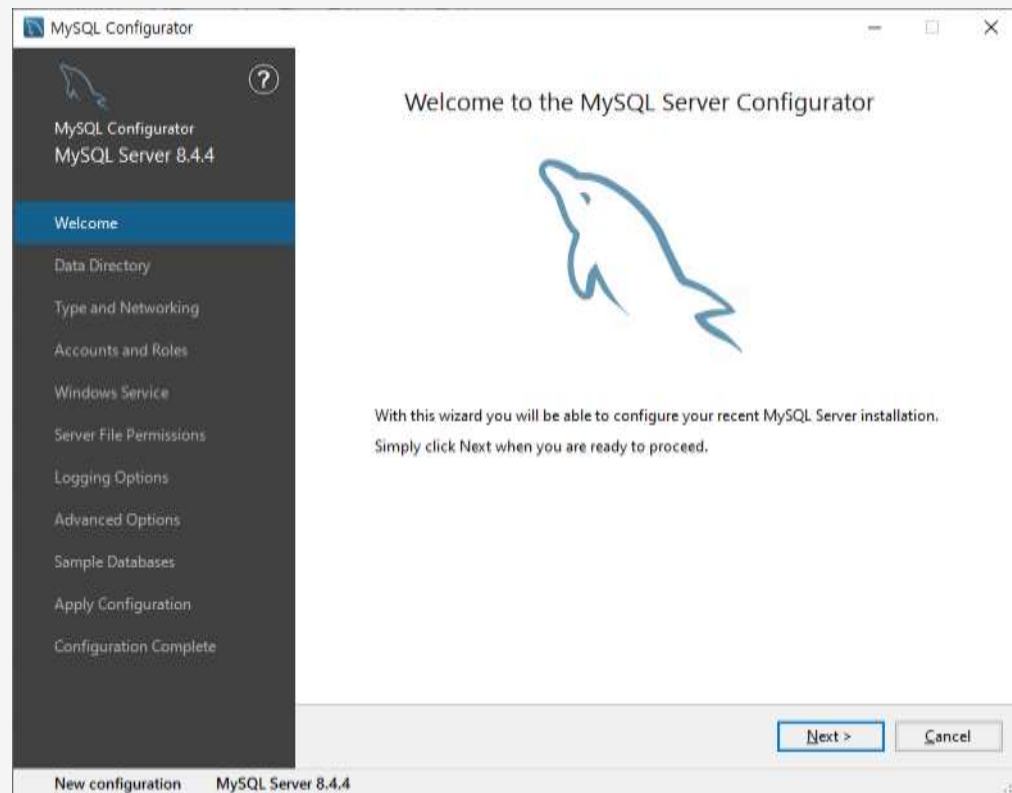


## 2. SQL 환경 설정

### ■ MySQL 설치

#### • MySQL Configurator 실행

- 다음(Next) 선택
- Data Directory 기본 선택 => 다음(Next) 선택
  - » C:\ProgramData\MySQL\MySQL Server 8.4\





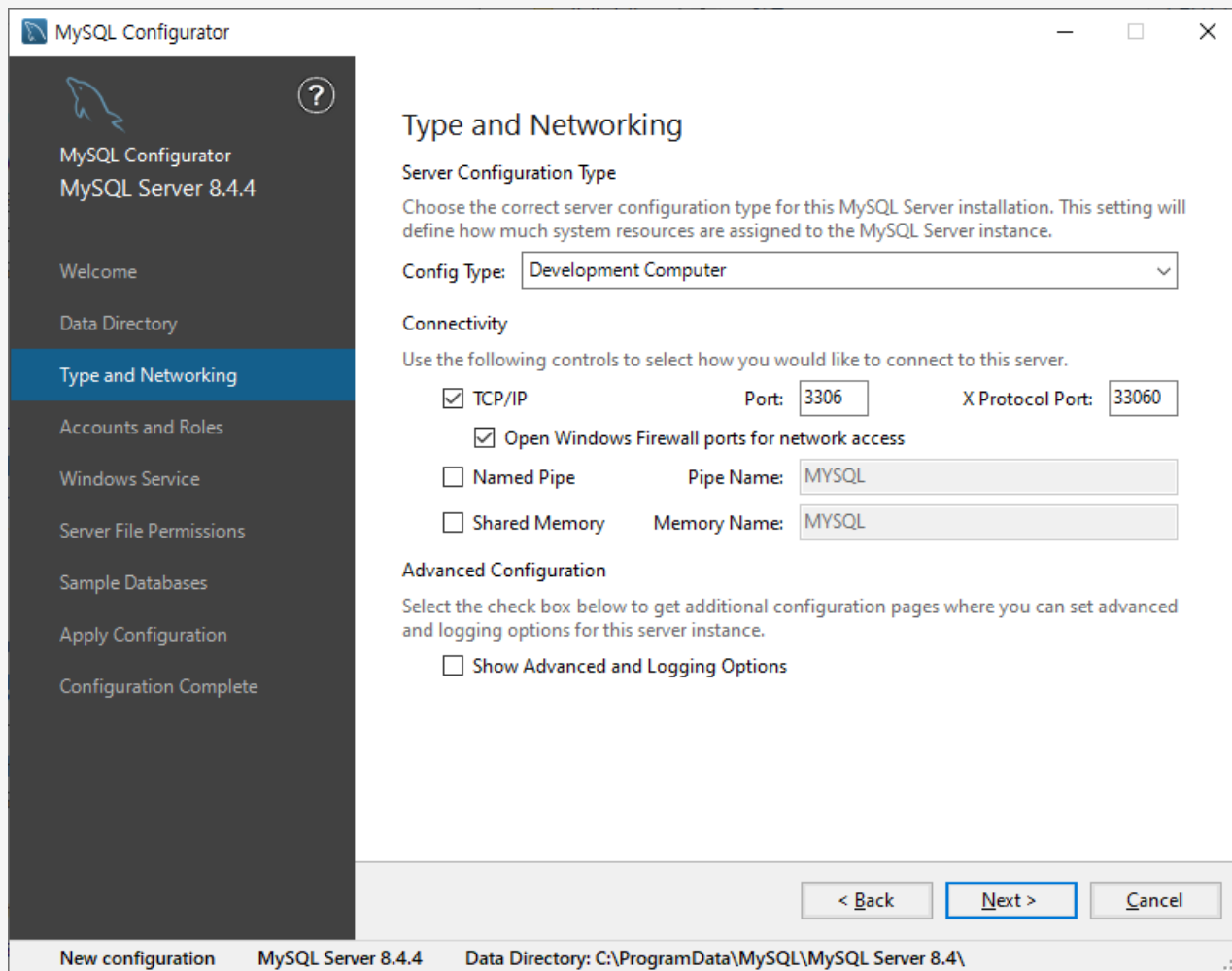
## 2. SQL 환경 설정

### ■ MySQL 설치

- MySQL Configurator 실행

- Type and Networking

- » Port: 3306 확인



## 2. SQL 환경 설정

### ■ MySQL 설치

- MySQL Configurator 실행

- Accounts and Roles

- » MySQL Root Password 입력 => 1234

MySQL Configurator

MySQL Server 8.4.4

Welcome

Data Directory

Type and Networking

**Accounts and Roles**

Windows Service

Server File Permissions

Sample Databases

Apply Configuration

Configuration Complete

### Accounts and Roles

Root Account Password

Enter the password for the root account. Please remember to store this password in a secure place.

MySQL Root Password:

Repeat Password:

Password strength: **Weak**

### MySQL User Accounts

Create MySQL user accounts for your users and applications. Assign a role to the user that consists of a set of privileges.

MySQL User Name	Host	User Role
-----------------	------	-----------

[Add User](#)

[Edit User](#)

[Delete](#)

< Back   Next >   Cancel

New configuration   MySQL Server 8.4.4   Data Directory: C:\ProgramData\MySQL\MySQL Server 8.4\

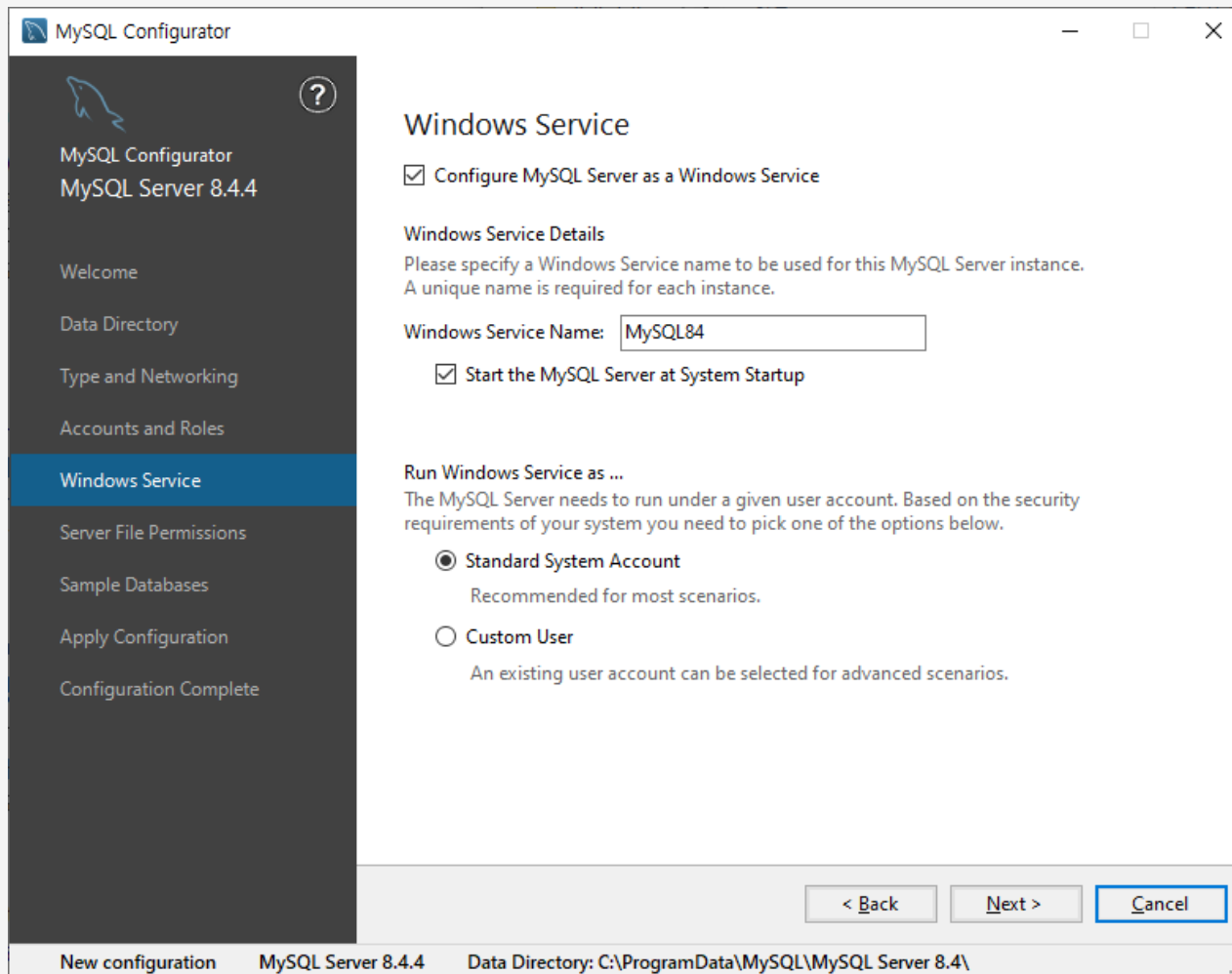
## 2. SQL 환경 설정

### ■ MySQL 설치

- MySQL Configurator 실행

- Windows Service

» Windows Service Name 확인 => MySQL84



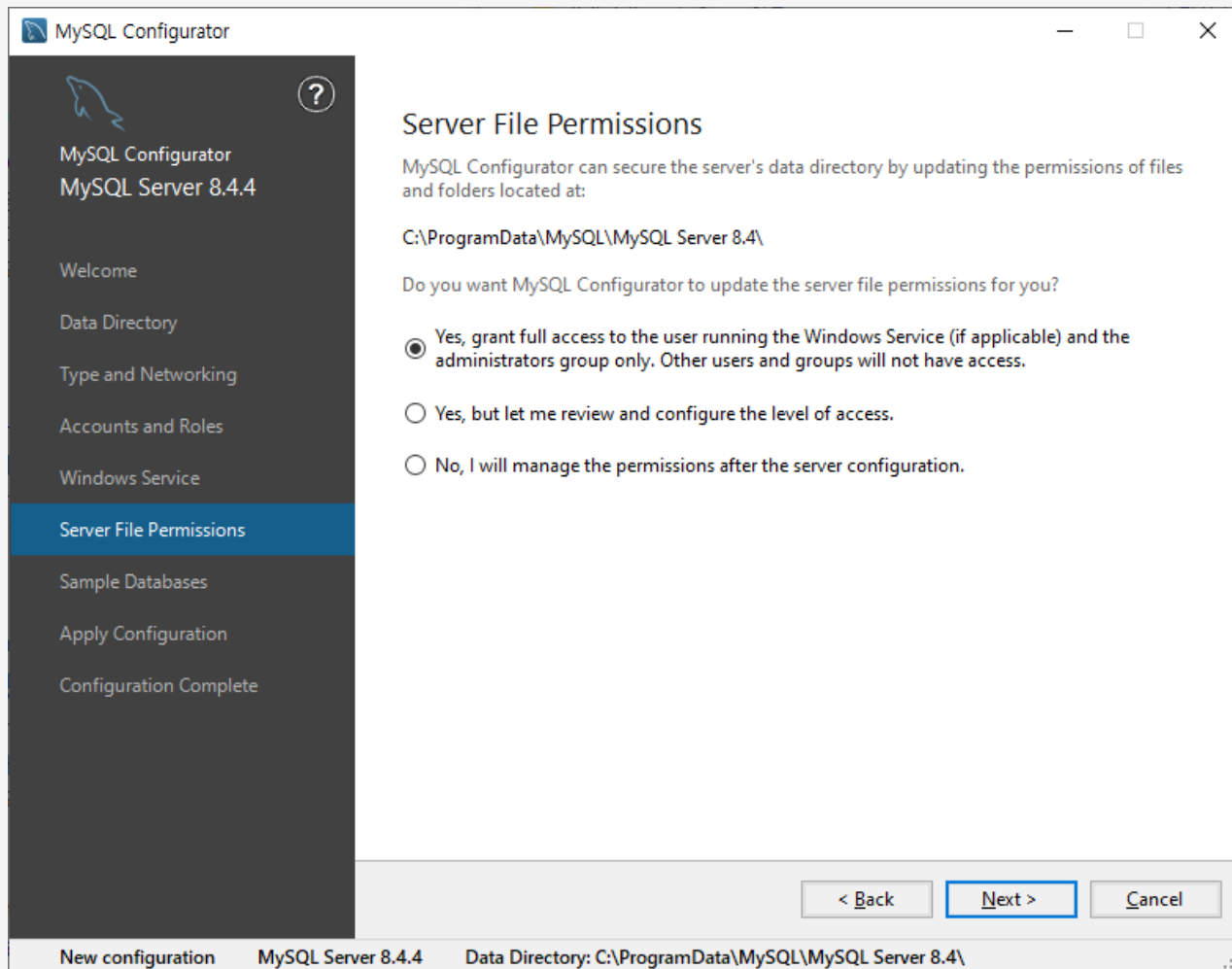
## 2. SQL 환경 설정

### ■ MySQL 설치

#### • MySQL Configurator 실행

##### • Server File Permissions

» Windows 서비스 실행 사용자 및 관리자 그룹에만 접근 권한 부여 => 다음(Next) 선택



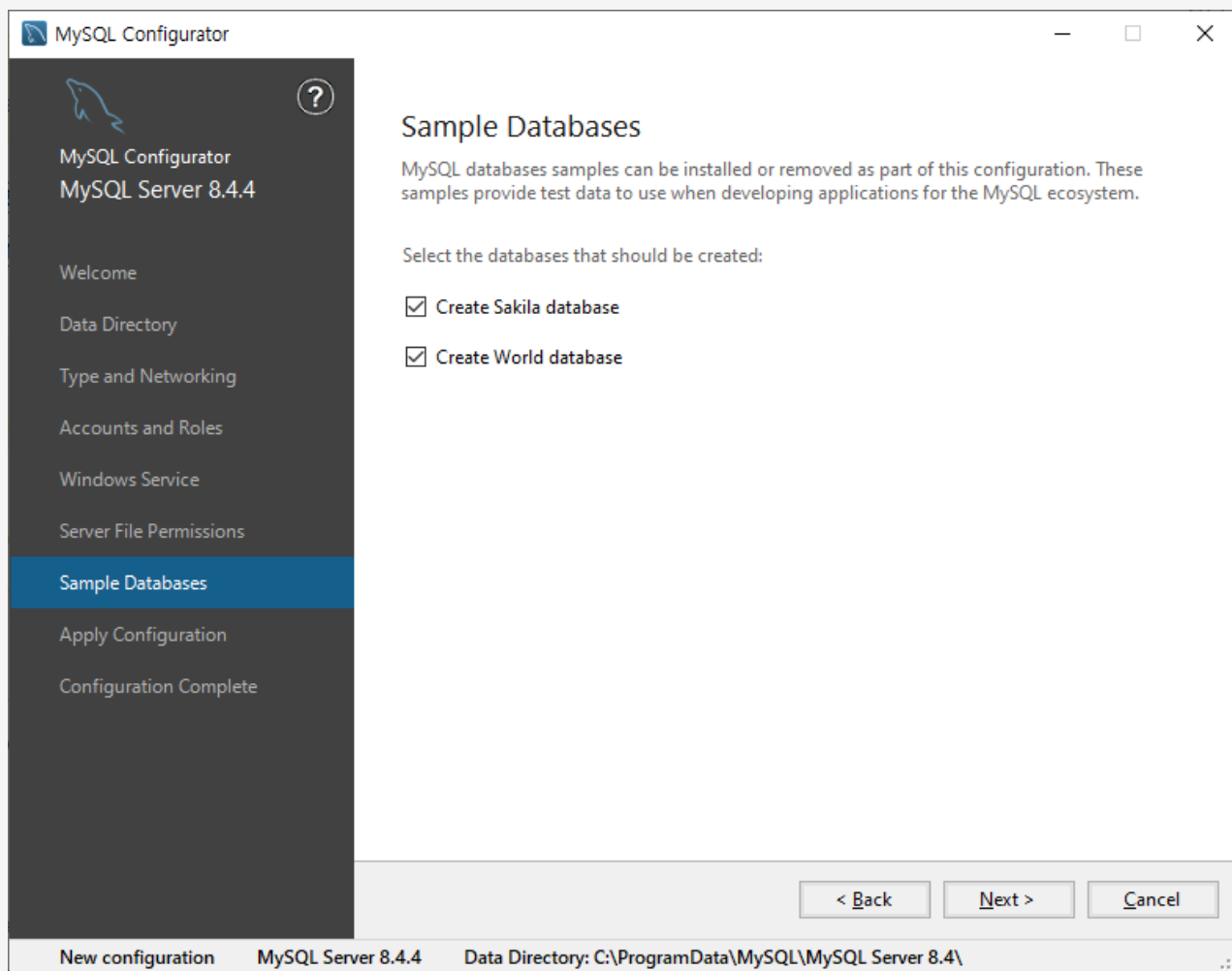
## 2. SQL 환경 설정

### ■ MySQL 설치

- MySQL Configurator 실행

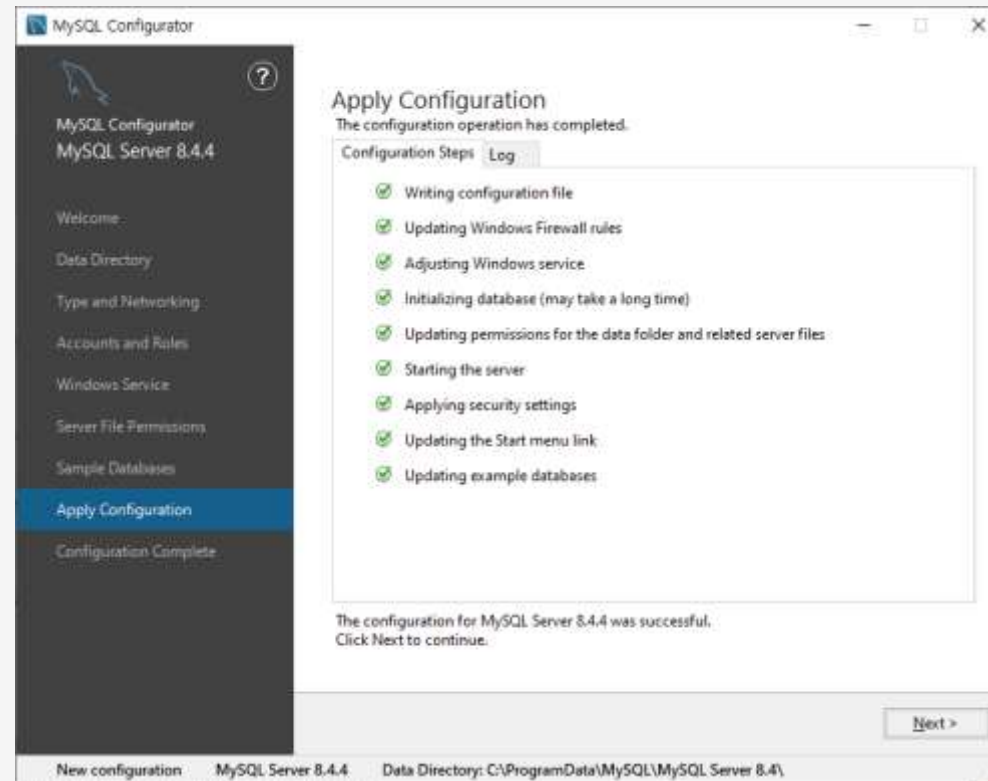
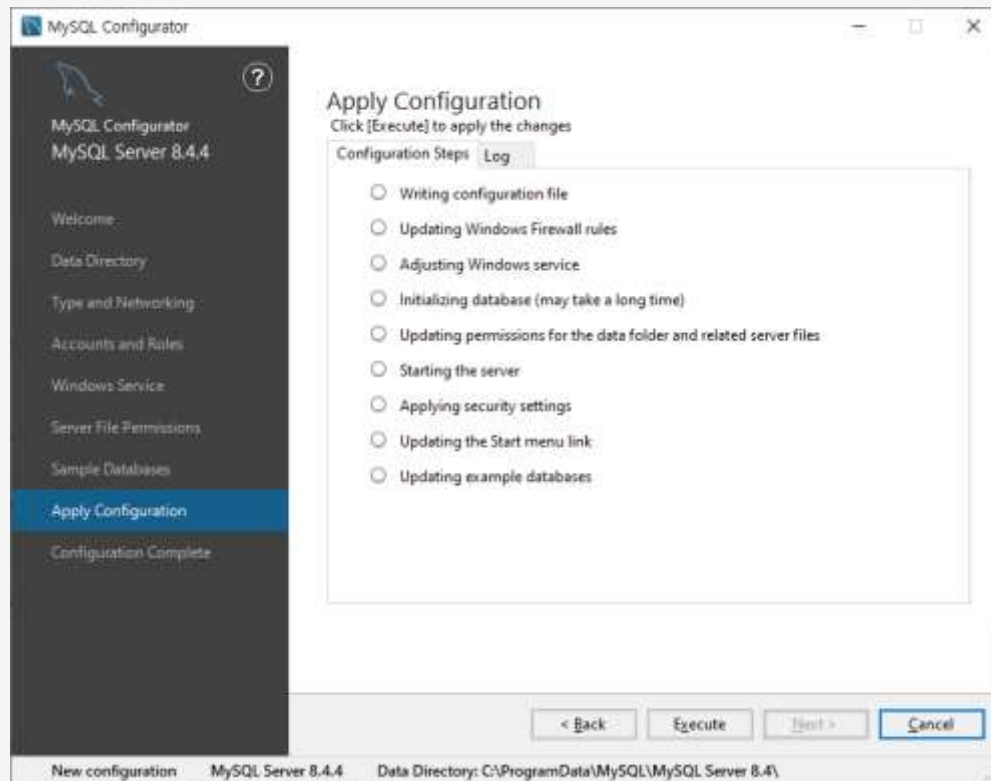
- Sample Databases

» Create Sakila database 와 Create World database 체크박스 모두 선택 => 다음(Next) 선택



## 2. SQL 환경 설정

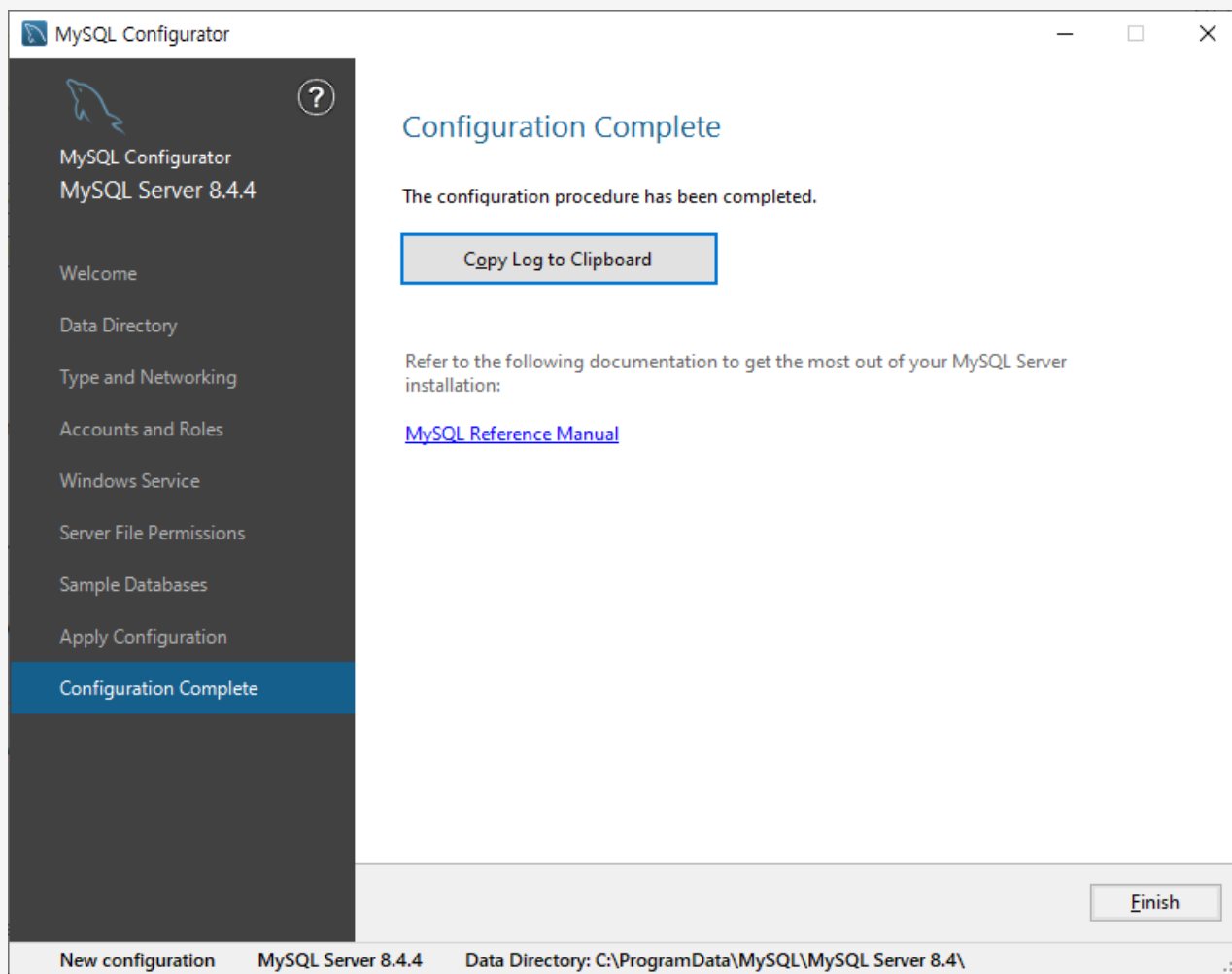
- MySQL 설치
  - MySQL Configurator 실행
    - Apply Configuration
      - » Execute 선택
      - » 다음(Next) 선택



## 2. SQL 환경 설정

### ■ MySQL 설치

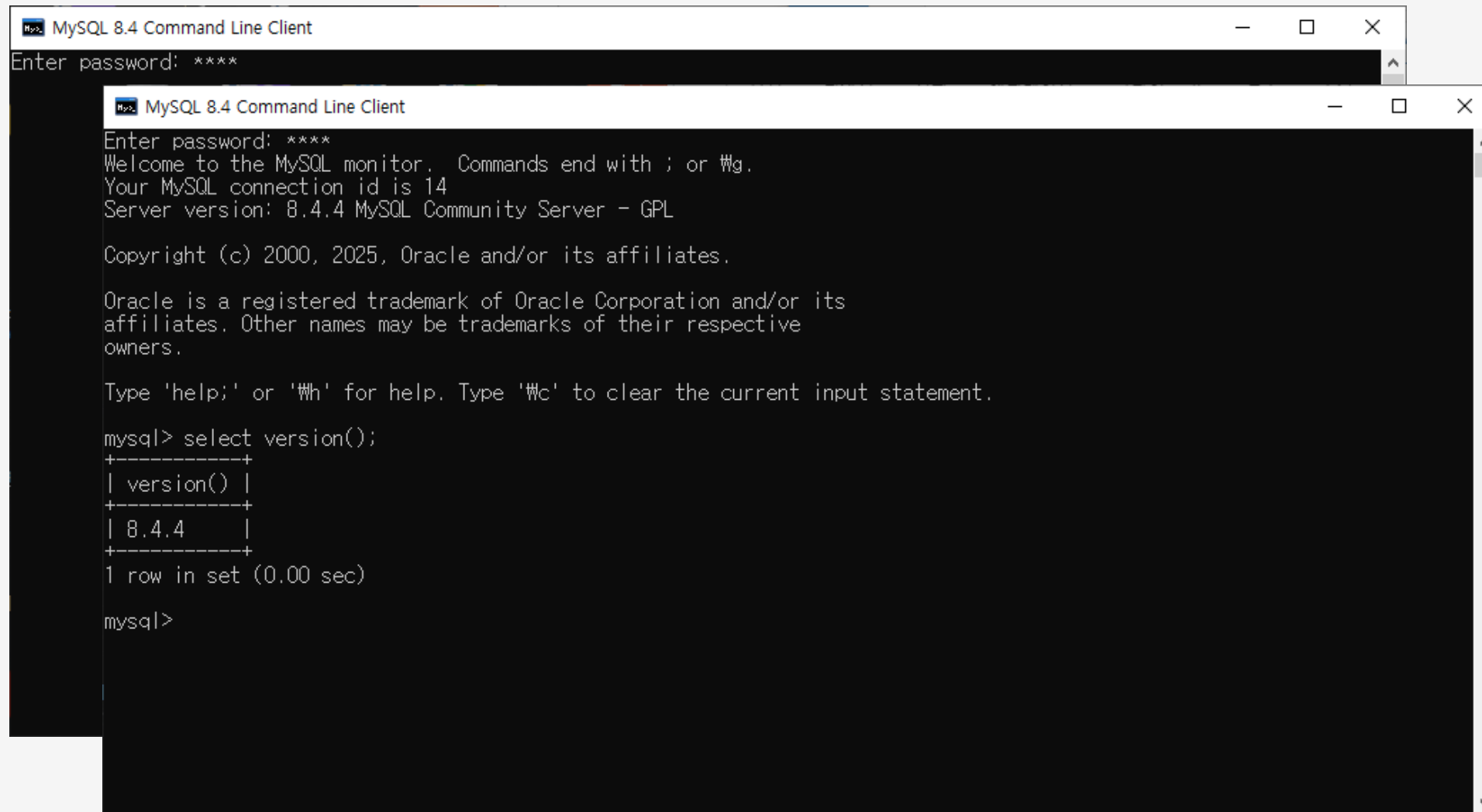
- MySQL Configurator 실행
  - Configuration complete
    - » 정상 설치 완료 확인 => 종료(Finish) 선택



## 2. SQL 환경 설정

### ■ MySQL 실행

- **Command Line Client 실행 => password 1234 입력**
  - **select version(); 입력 후, MySQL 버전 출력 확인**



```
MySQL 8.4 Command Line Client
Enter password: ****

MySQL 8.4 Command Line Client
Enter password: ****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 14
Server version: 8.4.4 MySQL Community Server - GPL

Copyright (c) 2000, 2025, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> select version();
+-----+
| version() |
+-----+
| 8.4.4      |
+-----+
1 row in set (0.00 sec)

mysql>
```



### ❖ SQL 개요

#### ■ SQL 소개

- 구조화된 질의 언어(Structured Query Language)
- 관계형 데이터베이스(RDBMS)를 관리하기 위한 언어
- 데이터 조회, 삽입, 수정, 삭제 등의 기능 수행

#### ■ SQL 분류

- DDL (Data Definition Language) : 데이터 정의 언어
  - 데이터베이스의 구조 관련 작업 수행
- DML (Data Manipulation Language) : 데이터 조작 언어
  - 데이터 조작을 위한 조회, 삽입, 수정, 삭제 작업 수행
- DCL (Data Control Language) : 데이터 제어 언어
  - 데이터베이스의 보안 및 권한 관리와 관련된 작업 수행
- TCL (Transaction Control Language) : 트랜잭션 제어 언어
  - 트랜잭션의 변경 사항의 영구적 저장 및 데이터베이스에 적용 작업 수행
  - DCL로 분류하는 경우도 있음

### ❖ SQL 개요

#### ■ 주요 SQL 명령어

- DDL

- CREATE : 테이블이나 인덱스, 뷰 등 데이터베이스 객체를 생성
- ALTER : 기 생성된 데이터베이스 객체를 수정
- DROP : 생성된 데이터베이스 객체를 영구히 삭제

- DML

- SELECT : 테이블이나 뷰에 있는 데이터를 조회
- INSERT : 데이터를 신규로 생성
- UPDATE : 기 생성된 데이터를 수정
- DELETE : 데이터 삭제

- DCL

- GRANT : 권한을 정의
- REVOKE : 권한을 삭제

- TCL

- COMMIT : 변경된 데이터를 최종 적용
- ROLLBACK : 변경된 데이터를 적용하지 않고 이전 상태로 되돌림

## 4. 데이터베이스 객체

### ❖ 데이터베이스 객체 개요

#### ■ 데이터베이스 객체 소개

- 데이터베이스를 구성하는 기본 요소들로, SQL을 사용하여 조작 가능

#### ■ 주요 객체:

- 데이터베이스(Database)
  - 데이터를 논리적으로 그룹화하는 컨테이너 역할을 수행
  - 여러 개의 테이블, 뷰, 프로시저 등과 같은 객체들을 포함 가능
- 테이블(Table): 데이터를 저장하는 기본 구조
- 뷰(View): 가상 테이블로 특정 데이터를 쉽게 조회할 수 있음
- 인덱스(Index): 검색 속도를 향상시키는 데이터 구조
- 제약조건(Constraints): 데이터가 특정 조건을 만족하도록 강제
- 함수(Functions): 저장된 프로그램으로, 특정 작업을 수행하고 값을 반환

## 4. 데이터베이스 객체

### ❖ 데이터베이스 생성

- 데이터를 관리하기 위한 시스템, 데이터베이스를 생성
- 생성 구문
  - **CREATE DATABASE** 데이터베이스명;
- 데이터베이스 보기
  - **show databases;**
- 선택 구문
  - **USE** 데이터베이스명;
- 삭제 구문
  - **DROP DATABASE** 데이터베이스명;

## 4. 데이터베이스 객체

### ❖ 테이블(Table)

#### ■ 데이터베이스 생성 예제

- `CREATE DATABASE ExampleDB;`
- `USE ExampleDB;`

#### ■ 선택 예제

- `USE ExampleDB;`

#### ■ 삭제 예제

- `DROP DATABASE ExampleDB;`

## 4. 데이터베이스 객체

### ❖ 테이블(Table)

- 데이터를 저장하는 가장 기본적인 데이터 베이스 객체
- 행(Row)과 열(Column)으로 구성된 2차원 형태
- SQL의 실행 대상인 객체

#### ■ 생성 구문

- **CREATE TABLE** 테이블명 (
  - 컬럼명1    컬럼1\_데이터타입    [제약조건],
  - 컬럼명2    컬럼2\_데이터타입    [제약조건],
  - ...
- );

#### ■ 삭제 구문

- **DROP TABLE** 테이블명;

## 4. 데이터베이스 객체

### ❖ 테이블(Table)

#### ■ 테이블 생성 규칙

- 테이블명, 컬럼명의 최대 길이는 64자
- 테이블명, 컬럼명으로 예약어 사용 불가
- 테이블명, 컬럼명은 문자, 숫자, '\_', '\$ ' 문자 사용가능
- 테이블명, 컬럼명의 첫 글자는 문자 또는 \_ 만
- 한 테이블에 만들 수 있는 컬럼은 최대 4096개
  - 단, 테이블 최대 크기와 컬럼 크기에 따라 제약이 있음

## 4. 데이터베이스 객체

### ❖ 테이블(Table)

#### ■ 주요 데이터 타입

- 문자 데이터 타입
  - CHAR(n): 고정 길이 문자열 저장
    - » 최대 길이 n 지정, 예: 국가 코드 등
  - VARCHAR(n): 가변 길이 문자열 저장
    - » 최대 길이 n 지정, 예: 이름, 주소 등
  - TEXT: 대량의 문자열 데이터 저장
    - » 예: 설명, 본문 내용 등
- 숫자 데이터 타입
  - INT: 정수형 데이터 저장
    - » 예: 나이, 개수 등
  - DECIMAL(m,d): 정밀한 소수점을 포함한 숫자 저장
    - » m: 전체 자리수, d: 소수점 이하 자리수
- 날짜 데이터 타입
  - DATE: 날짜 데이터 저장
    - » YYYY-MM-DD 형식
  - DATETIME: 날짜와 시간을 함께 저장
    - » YYYY-MM-DD HH:MM:SS 형식



## 4. 데이터베이스 객체

### ❖ 테이블(Table)

#### ■ NULL

- 데이터가 없음을 의미, 컬럼 속성에 명시, 디폴트는 NULL 허용

#### ■ 제약 조건

- NOT NULL: 해당 컬럼이 NULL 값을 가질 수 없음
- UNIQUE: 컬럼 값이 테이블 내에서 중복되지 않도록 제한
- PRIMARY KEY: 기본 키로 설정하며, 중복 및 NULL 값을 허용하지 않음
- FOREIGN KEY: 다른 테이블의 기본 키를 참조하는 외래 키 설정
- CHECK: 특정 조건을 만족하는 값만 입력 가능 (MySQL 8.0 이상 지원)
- DEFAULT: 특정 컬럼에 기본값 설정
- AUTO\_INCREMENT: 자동 증가 값 부여 (기본 키와 함께 사용)

## 4. 데이터베이스 객체

### ❖ 테이블(Table)

#### ■ 테이블 생성 예제

- **CREATE TABLE employees (**
- **employee\_id INT PRIMARY KEY AUTO\_INCREMENT,**
- **name VARCHAR(50) NOT NULL,**
- **department\_name VARCHAR(50),**
- **salary DECIMAL(10,2) CHECK (salary > 0),**
- **hiredate DATE DEFAULT (CURRENT\_DATE)**
- **);**

#### ■ 테이블 목록 확인

- **show tables;**

## 4. 데이터베이스 객체

### ❖ 테이블(Table)

- 테이블 구조 확인
  - `describe employees;`
- 테이블 삭제 예제
  - `DROP TABLE employees;`

## 4. 데이터베이스 객체

### ❖ 뷰(View)

- 하나 이상의 테이블에서 특정 컬럼이나 조건을 바탕으로 데이터를 회조회 할 수 있는 가상 테이블
- 생성 구문
  - **CREATE VIEW 뷰명 AS**
  - **SELECT 컬럼명 FROM 테이블명 WHERE 조건;**
- 삭제 구문
  - **DROP VIEW 테이블명;**

## 4. 데이터베이스 객체

### ❖ 뷰(View)

#### ■ 뷰 생성 예제

- **CREATE VIEW developer\_view AS**
- **SELECT name, position, salary**
- **FROM employees**
- **WHERE position = '개발자';**

#### ■ 뷰 구조 확인

- **describe developer\_view;**

#### ■ 뷰 삭제 예제

- **DROP VIEW developer\_view;**

## ❖ 과제

- 1. MySQL 환경 설정 과정을 재 확인하기
- 2. SQL 개념과 분류에 대해 설명하기
- 3. 데이터 베이스의 주요 객체에 대해 설명하기
- 4. 데이터 베이스 객체인 테이블과 제약조건을 사용하여 코드 작성하기

## ❖ 다음 수업 내용

- DBMS 2
  - SQL 문장
  - 조인
  - 파이썬 DB 연동