

MariaDB 설치

MariaDB는 MySQL에서 파생된 오픈소스 RDBMS이다.

오라클이 썬 마이크로 시스템을 2010년에 72억 달러(약 8조)에 인수해가면서 썬 마이크로시스템즈에 속해 있던 MySQL 역시 오라클 것이 됨에 따라 상업적으로 MySQL을 이용할 시 사용료를 내도록 정책을 변경되었다.

이 정책에 반발한 MySQL의 원 개발사의 핵심 창업자중 한 명이었던 몬티 와이드니어스가 2009년 동료들과 나와 MySQL의 소스코드를 기반으로한 오픈소스 RDBMS를 개발한 것이 지금의 MariaDB이다.

MariaDB는 기본적으로 MySQL과 사용방법이 동일하다.

#01. DBMS 설치하기

1) 패키지 업데이트

root 권한을 임대하여 진행한다.

```
$ sudo apt-get update
```

```
leekh@my-server:~$ sudo apt-get update
[sudo] password for leekh:
Hit:1 http://ports.ubuntu.com/ubuntu-ports jammy InRelease
Get:2 http://ports.ubuntu.com/ubuntu-ports jammy-updates InRelease [114 kB]
Get:3 http://ports.ubuntu.com/ubuntu-ports jammy-backports InRelease [99.8 kB]
Get:4 http://ports.ubuntu.com/ubuntu-ports jammy-security InRelease [110 kB]
Get:5 http://ports.ubuntu.com/ubuntu-ports jammy-updates/main arm64 Packages [365 kB]
Get:6 http://ports.ubuntu.com/ubuntu-ports jammy-updates/universe arm64 Packages [164 kB]
Fetched 852 kB in 3s (251 kB/s)
Reading package lists... Done
leekh@my-server:~$
```

2) 프로그램 설치

root 권한을 임대하여 `mariadb-server`를 설치한다.

```
$ sudo apt-get install -y mariadb-server
```

3) 설치된 버전 확인

mariadb이지만 아직까지 명령어들은 mysql을 그대로 따르고 있다. `-V`가 대문자임에 주의

```
$ mysql -V
```

```
leekh@my-server: ~ — 94x3
leekh@my-server:~$ mysql -V
mysql Ver 15.1 Distrib 10.6.7-MariaDB, for debian-linux-gnu (aarch64) using EditLine wrapper
leekh@my-server:~$
```

4) MySQL 보안 설정

```
$ sudo mysql_secure_installation
```

mysql의 현재 root 비밀번호 [엔터입력]

설치 초기 단계이므로 비밀번호가 없다. 바로 엔터를 누른다.

```
leekh@my-server: ~
leekh@my-server:~$ sudo mysql_secure_installation

NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB
SERVERS IN PRODUCTION USE! PLEASE READ EACH STEP CAREFULLY!

In order to log into MariaDB to secure it, we'll need the current
password for the root user. If you've just installed MariaDB, and
haven't set the root password yet, you should just press enter here.

Enter current password for root (enter for none):
```

Unix 인증 방식 적용 여부 [Y입력]

```
leekh@my-server: ~
Switch to unix_socket authentication [Y/n] Y
Enabled successfully!
Reloading privilege tables..
... Success!

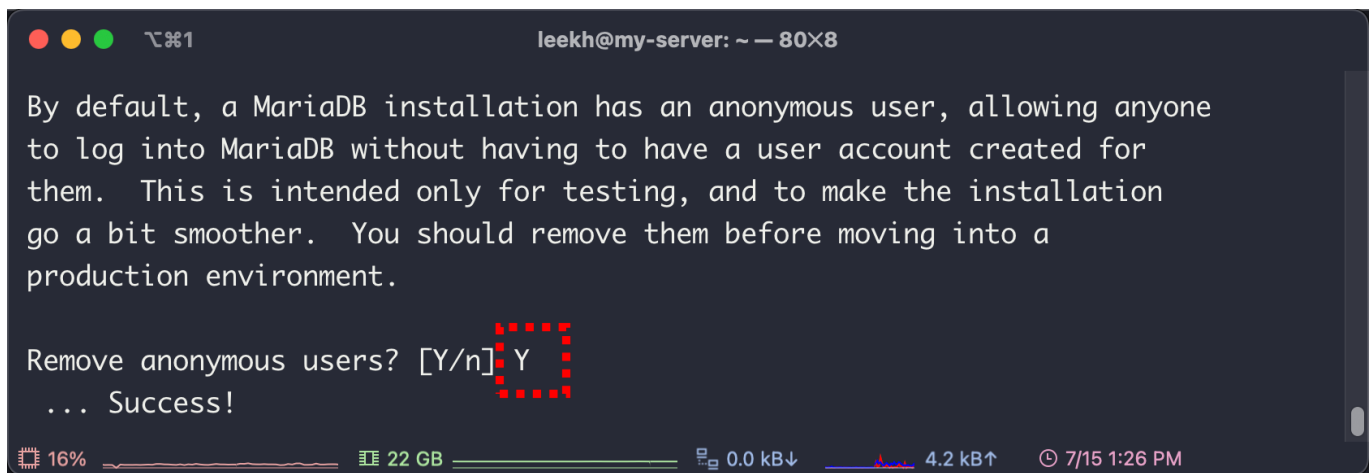
You already have your root account protected, so you can safely answer 'n'.
```

mysql의 root 비밀번호 변경 여부 [Y입력 후 비밀번호 설정]


```

leekh@my-server: ~
Change the root password? [Y/n] Y
New password:
Re-enter new password:
Password updated successfully!
Reloading privilege tables..
... Success!
17% 22 GB 7.2 kB↓ 8.2 kB↑ 7/15 1:25 PM

```

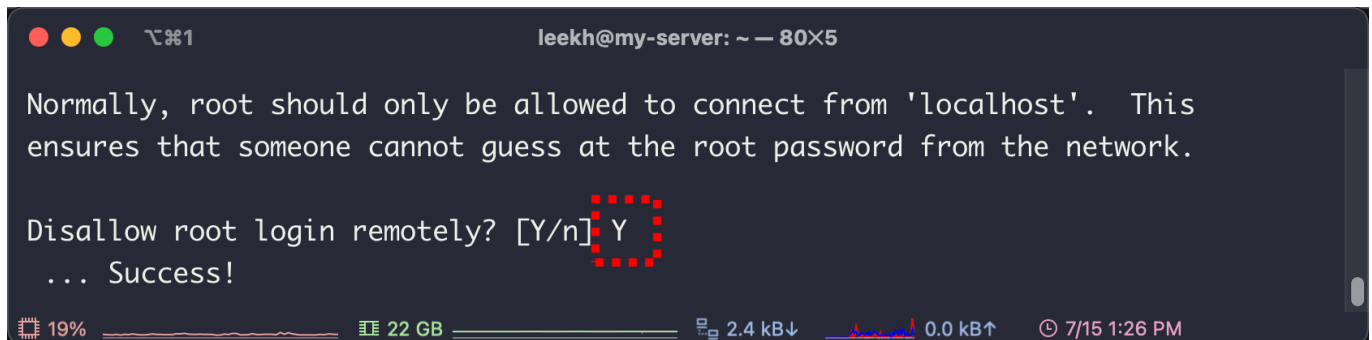
익명 사용자 계정 삭제 여부 [Y입력]


```

leekh@my-server: ~ — 80x8
By default, a MariaDB installation has an anonymous user, allowing anyone
to log into MariaDB without having to have a user account created for
them. This is intended only for testing, and to make the installation
go a bit smoother. You should remove them before moving into a
production environment.

Remove anonymous users? [Y/n] Y
... Success!
16% 22 GB 0.0 kB↓ 4.2 kB↑ 7/15 1:26 PM

```

mysql root의 원격 접속 비활성화 [Y입력]


```

leekh@my-server: ~ — 80x5
Normally, root should only be allowed to connect from 'localhost'. This
ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [Y/n] Y
... Success!
19% 22 GB 2.4 kB↓ 0.0 kB↑ 7/15 1:26 PM

```

테스트 데이터베이스 삭제 여부 [Y입력]

```
leekh@my-server: ~ — 80x9

By default, MariaDB comes with a database named 'test' that anyone can
access. This is also intended only for testing, and should be removed
before moving into a production environment.

Remove test database and access to it? [Y/n] Y
- Dropping test database...
... Success!
- Removing privileges on test database...
... Success!
```

설정 테이블 다시 로드하기 [Y입력]

```
leekh@my-server: ~ — 94x10

Reload privilege tables now? [Y/n] y
... Success!

Cleaning up...

All done! If you've completed all of the above steps, your MariaDB
installation should now be secure.

Thanks for using MariaDB!
leekh@my-server:~$
```

5) MySQL 서비스 시작하기

정상적인 경우 아무런 결과도 표시되지 않는다.

```
sudo systemctl start mariadb
```

6) 정상 가동 여부 확인

```
$ sudo systemctl status mariadb
```

```
leekh@my-server: ~ — 107X13
leekh@my-server:~$ sudo systemctl status mysql
● mariadb.service - MariaDB 10.6.7 database server
   Loaded: loaded (/lib/systemd/system/mariadb.service; enabled; vendor preset: enabled)
   Active: active (running) since Fri 2022-07-15 04:19:02 UTC; 9min ago
     Docs: man:mariadb(8)
           https://mariadb.com/kb/en/library/systemd/
  Process: 2026 ExecStartPre=/usr/bin/install -m 755 -o mysql -g root -d /var/run/mysqld (code=exited, s
  Process: 2027 ExecStartPre=/bin/sh -c systemctl unset-environment _WSREP_START_POSITION (code=exited, >
  Process: 2029 ExecStartPre=/bin/sh -c [ ! -e /usr/bin/galera_recovery ] && VAR= || VAR=`cd /usr/bin/>
  Process: 2071 ExecStartPost=/bin/sh -c systemctl unset-environment _WSREP_START_POSITION (code=exited,>
  Process: 2073 ExecStartPost=/etc/mysql/debian-start (code=exited, status=0/SUCCESS)
 Main PID: 2058 (mariadb)
   Status: "Taking your SQL requests now..."
```

7) 운영체제 부팅시 서비스 자동 시작 등록

```
sudo systemctl enable mariadb
```

8) MySQL 접속 확인하기

```
$ mysql -uroot -p
```

```
leekh@my-server: ~ — 94X11
leekh@my-server:~$ mysql -uroot -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 41
Server version: 10.6.7-MariaDB-2ubuntu1.1 Ubuntu 22.04

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

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

MariaDB [(none)]> 
```

정상 접속이 확인되면 이후 FTP를 통해 백업파일을 서버에 업로드 하고 데이터베이스 복구 및 사용자 계정 생성을 수행합니다.

#03. MySchool Database 복구하기

1. 복구할 데이터베이스 생성

MySQL에 root 계정으로 접속한 상태에서 myschool 데이터베이스를 생성한다.

```
sql> CREATE DATABASE myschool default charset utf8;
```

```

leekh@my-server: ~ — 93x16
MariaDB [(none)]> create database myschool default charset utf8;
Query OK, 1 row affected (0.001 sec)

MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| information_schema |
| myschool |
| mysql |
| performance_schema |
| sys |
+-----+
5 rows in set (0.003 sec)

MariaDB [(none)]>

```

2. 데이터베이스 복구하기

`exit` 명령으로 MySQL과의 접속을 해제하고 업로드 한 백업파일을 복원한다.

복원할 파일이 FTP를 통해 사전에 서버에 업로드 되어 있어야 한다.

```
$ mysql -uroot -p DATABASE이름 < 백업파일경로
```

```

leekh@my-server: ~
leekh@my-server:~$ pwd
/home/leekh
leekh@my-server:~$ ls
myschool-dump.sql
leekh@my-server:~$ mysql -uroot -p myschool < ~/myschool-dump.sql
Enter password:
leekh@my-server:~$

```

#04. 계정 설정

1. 사용자 계정 생성하기

다시 mysql에 root로 로그인 한 후 사용자 계정을 생성한다.

```
sql> create user '아이디'@'접근허용호스트' identified by '비밀번호';
```

접근허용 호스트 구분

호스트	설명
localhost	해당 사용자는 반드시 리눅스에 원격접속을 한 상태에서만 MySQL의 이용이 가능하다.
특정 IP 주소	리눅스로의 원격 접속을 거치지 않고 지정된 IP를 사용하는 컴퓨터를 통해서 mysql로의 직접 접속이 가능하다.
%	로컬, 원격 구분 없이 어디서나 mysql로 접속이 가능하다

원격지에서 mysql로의 직접 접속이 가능하도록 지정된 경우 리눅스에서 mysql이 사용중인 포트번호에 대한 방화벽을 오픈해야 접속이 가능합니다.

```
leekh@my-server: ~
MariaDB [(none)]> create user 'myschool'@'%' identified by '123qwe!@#';
Query OK, 0 rows affected (0.004 sec)

MariaDB [(none)]>
```

2. 데이터베이스에 대한 사용자 권한 부여하기

```
sql> grant all privileges on 데이터베이스이름.* to '아이디'@'접근허용호스트';
```

```
leekh@my-server: ~
MariaDB [(none)]> grant all privileges on myschool.* to 'myschool'@'%;
Query OK, 0 rows affected (0.011 sec)

MariaDB [(none)]>
```

3. 데이터베이스에 대한 사용자 권한 제거하기

사용자로부터 권한을 회수하는 일은 거의 발생하지 않지만 필요하다면 아래의 명령을 사용한다.

```
sql> revoke all on 데이터베이스이름.* from '아이디'@'접근허용호스트';
```

4. 생성된 계정으로 mysql 로그인 확인

exit 명령으로 MySQL과의 접속을 해제하고 새로 생성한 아이디를 사용해 MySQL에 접근해 본다.

```

leekh@my-server: ~$ mysql -umyschool -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 38
Server version: 10.6.7-MariaDB-2ubuntu1.1 Ubuntu 22.04

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

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

MariaDB [(none)]> use myschool;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
MariaDB [myschool]> select * from department;
+-----+-----+-----+
| deptno | dname                | loc    |
+-----+-----+-----+
|      101 | 컴퓨터공학과         | 1호관  |
|      102 | 멀티미디어학과     | 2호관  |
|      201 | 전자공학과           | 3호관  |
|      202 | 기계공학과           | 4호관  |
+-----+-----+-----+
4 rows in set (0.002 sec)

MariaDB [myschool]>

```

#05. MySQL 외부접속 설정

1. 포트번호 변경

일반적인 실무 시스템에서는 보안상의 이유로 MySQL의 외부접속을 허용하지 않지만 여기서는 실습용 환경이므로 외부 접속을 허용하도록 설정을 진행한다.

기본 포트는 일반적으로 알려져 있기 때문에 보안에 불리하므로 임의의 포트로 변경해야 한다.

설정파일 열기

```
$ sudo vi /etc/mysql/mariadb.conf.d/50-server.cnf
```

설정파일에 포트번호 옵션 추가

[mysqld] 섹션에 `port = 포트번호`를 추가하고 파일을 저장한 후 vi를 종료한다.


```

leekh@my-server: ~
[mysqld]

#
# * Basic Settings
#
#user                = mysql
pid-file             = /run/mysqld/mysqld.pid
basedir              = /usr
#datadir              = /var/lib/mysql
#tmpdir               = /tmp
port = 9903

-- INSERT --
14,1 7%
19% 28 GB 12 kB↓ 13 kB↑ 7/20 8:09 PM

```

`bind-address`라는 설정항목을 찾아 `0.0.0.0`으로 값을 수정한다.

```

leekh@my-server: ~
# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
#bind-address        = 127.0.0.1
bind-address = 0.0.0.0

#
# * Fine Tuning
-- INSERT --
29,23 22%
14% 30 GB 0.0 kB↓ 1.0 kB↑ 7/20 8:25 PM

```

서비스 재시작

```
$ sudo systemctl restart mariadb
```

2. 방화벽에 포트번호 추가

설정한 포트번호를 방화벽 허용 목록에 추가하고 방화벽을 다시 로드한다.

```
$ sudo ufw allow 9903/tcp
$ sudo ufw reload
```

```
leekh@my-server: ~$ sudo ufw allow 9903/tcp
Rule added
Rule added (v6)
leekh@my-server:~$ sudo ufw reload
Firewall reloaded
leekh@my-server:~$
```

3. 외부 컴퓨터에서 접속 확인

Window, Mac 등의 호스트 컴퓨터에서 MySQL에 접속을 시도한다.

```
$ mysql -h리눅스아이피 -P포트번호 -u계정명 -p
```

```
/Applications/MAMP/Library/bin/mysql -h10.211.55.8 -P9903 -umyschool -p
→ mysql -h10.211.55.8 -P9903 -umyschool -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 31
Server version: 5.5.5-10.6.7-MariaDB-2ubuntu1.1 Ubuntu 22.04

Copyright (c) 2000, 2021, 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>
```

#06. MariaDB 삭제

1. 서비스 중지

```
sudo systemctl stop mariadb
```

2. 서비스 자동시작 등록 해제

```
sudo systemctl disable mariadb
```

3. MariaDB 완전 삭제

```
$ sudo apt-get purge -y mariadb*
```

4. 사이드 패키지 제거

```
$ sudo apt autoremove -y
```

5. MySQL 관련 파일들 리스트 확인

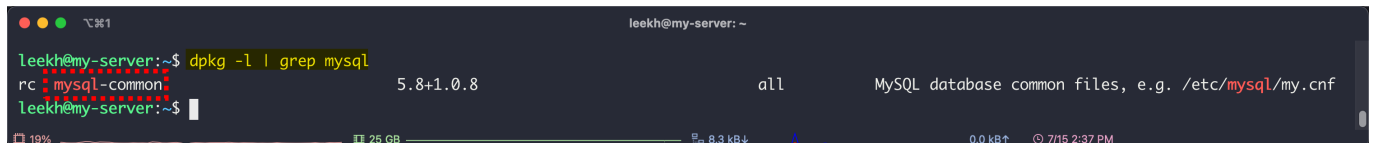
아래 명령을 수행하여 출력되는 항목이 없어야 한다.

```
$ dpkg -l | grep mariadb
$ dpkg -l | grep mysql
```

만약 출력되는 파일목록이 있다면 아래 명령으로 잔존 파일들을 삭제한다.

```
$ sudo apt-get purge -y 파일명
```

예를 들어 아래와 같이 **mysql-common**이라는 항목이 출력될 경우,



```
leekh@my-server:~$ dpkg -l | grep mysql
rc mysql-common 5.8+1.0.8 all MySQL database common files, e.g. /etc/mysql/my.cnf
leekh@my-server:~$
```

아래의 명령으로 삭제한다

```
$ sudo apt-get purge -y mysql-common
```

6. 재부팅

삭제를 완료하기 위해서는 시스템 재부팅이 필요하다.

```
$ sudo reboot
```