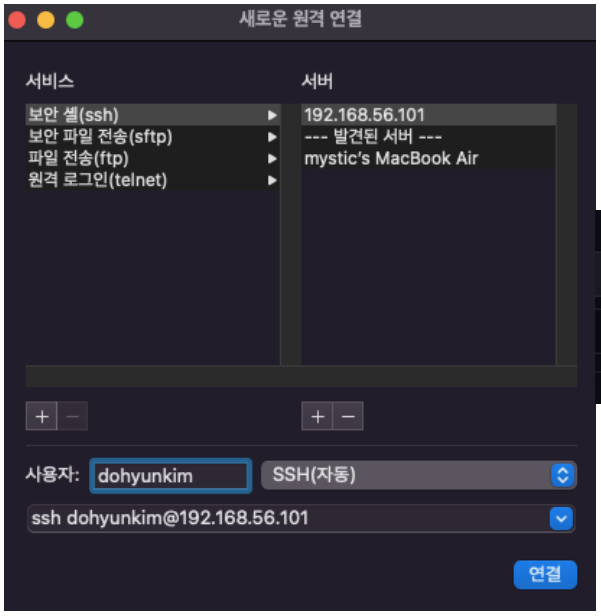
HW2

2019038046 김도현

1. 정적 IP를 이용하여 SSH로 MySQL서버에 접속.




```
The authenticity of host '192.168.56.101 (192.168.56.101)' can't be established.
ED25519 key fingerprint is SHA256:gyg03WkULTvh3BN6Gxbs00AsCnQlCEdL3afwJ0OY7no.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.56.101' (ED25519) to the list of known hosts
.
dohyunkim@192.168.56.101's password:
Last login: Thu Sep 14 14:14:18 2023
[dohyunkim@mysql ~]$
```

2. 요구사항들 설치(GPG-KEY 오류 해결)

```
[dohyunkim@mysql ~]$ sudo yum -y install http://dev.mysql.com/get/mysql57-community-release-el7-11.noarch.rpm
[sudo] dohyunkim의 암호 :
Loaded plugins: fastestmirror
mysql57-community-release-el7-11.noarch.rpm              |  25 kB     00:00
Examining /var/tmp/yum-root-Ogp_fr/mysql57-community-release-el7-11.noarch.rpm:
mysql57-community-release-el7-11.noarch
Marking /var/tmp/yum-root-Ogp_fr/mysql57-community-release-el7-11.noarch.rpm to
be installed
Resolving Dependencies
--> Running transaction check
---> Package mysql57-community-release.noarch 0:el7-11 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package           Arch    Version
                                Repository                                  Size
================================================================================
Installing:
 mysql57-community-release
                   noarch el7-11 /mysql57-community-release-el7-11.noarch  31 k

Transaction Summary
================================================================================
Install  1 Package

Total size: 31 k
Installed size: 31 k
Downloading packages:
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : mysql57-community-release-el7-11.noarch                     1/1
  Verifying  : mysql57-community-release-el7-11.noarch                     1/1

Installed:
  mysql57-community-release.noarch 0:el7-11

Complete!
```

```
[[dohyunkim@mysql ~]$ sudo yum -y install mysql-community-server
Loaded plugins: fastestmirror
Determining fastest mirrors

Public key for mysql-community-server-5.7.43-1.el7.x86_64.rpm is not installed


 Failing package is: mysql-community-server-5.7.43-1.el7.x86_64
 GPG Keys are configured as: file:///etc/pki/rpm-gpg/RPM-GPG-KEY-mysql
  perl-podlators.noarch 0:2.5.1-3.el7
  perl-threads.x86_64 0:1.87-4.el7
  perl-threads-shared.x86_64 0:1.43-6.el7

Replaced:
  mariadb-libs.x86_64 1:5.5.68-1.el7

Complete!
[dohyunkim@mysql ~]$ sudo rpm --import https://repo.mysql.com/RPM-GPG-KEY-mysql-
2022
```

3. mysql접속

```
[[dohyunkim@mysql ~]$ mysql -u root -p
[Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 2
Server version: 5.7.43

Copyright (c) 2000, 2023, 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
.
```

4. 패스워드 변경

```
mysql> SET GLOBAL validate_password_policy=LOW;
Query OK, 0 rows affected (0.00 sec)

mysql> SET GLOBAL validate_password_length=4;
Query OK, 0 rows affected (0.00 sec)

mysql> SHOW variables LIKE 'validate_password%';
+--------------------------------------+-------+
| Variable_name                        | Value |
+--------------------------------------+-------+
| validate_password_check_user_name    | OFF   |
| validate_password_dictionary_file    |       |
| validate_password_length             | 4     |
| validate_password_mixed_case_count   | 1     |
| validate_password_number_count       | 1     |
| validate_password_policy             | LOW   |
| validate_password_special_char_count | 1     |
+--------------------------------------+-------+
7 rows in set (0.00 sec)
```

```
mysql> ALTER USER 'root'@'localhost' IDENTIFIED BY '0000';
Query OK, 0 rows affected (0.00 sec)

mysql> quit
Bye
[dohyunkim@mysql ~]$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 3
Server version: 5.7.43 MySQL Community Server (GPL)

Copyright (c) 2000, 2023, 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
.
```

5. MySQL 한글환경 설정

```
[dohyunkim@mysql ~]$ cat /etc/my.cnf
# For advice on how to change settings please see
# http://dev.mysql.com/doc/refman/5.7/en/server-configuration-defaults.html

[mysqld]
#
# Remove leading # and set to the amount of RAM for the most important data
# cache in MySQL. Start at 70% of total RAM for dedicated server, else 10%.
# innodb_buffer_pool_size = 128M
#
# Remove leading # to turn on a very important data integrity option: logging
# changes to the binary log between backups.
# log_bin
#
# Remove leading # to set options mainly useful for reporting servers.
# The server defaults are faster for transactions and fast SELECTs.
# Adjust sizes as needed, experiment to find the optimal values.
# join_buffer_size = 128M
# sort_buffer_size = 2M
# read_rnd_buffer_size = 2M
datadir=/var/lib/mysql
socket=/var/lib/mysql/mysql.sock

# Disabling symbolic-links is recommended to prevent assorted security risks
symbolic-links=0

log-error=/var/log/mysqld.log
pid-file=/var/run/mysqld/mysqld.pid

character-set-server = utf8
validate_password_length = 4
validate_password_policy = LOW

[mysqldump]
default-character-set = utf8

[mysql]
default-character-set = utf8

[client]
default-character-set = utf8
```

```
mysql> show variables like 'c%';
+--------------------------+----------------------------+
| Variable_name            | Value                      |
+--------------------------+----------------------------+
| character_set_client     | utf8                       |
| character_set_connection | utf8                       |
| character_set_database   | utf8                       |
| character_set_filesystem | binary                     |
| character_set_results    | utf8                       |
| character_set_server     | utf8                       |
| character_set_system     | utf8                       |
| character_sets_dir       | /usr/share/mysql/charsets/ |
| check_proxy_users        | OFF                        |
| collation_connection     | utf8_general_ci            |
| collation_database       | utf8_general_ci            |
| collation_server         | utf8_general_ci            |
| completion_type          | NO_CHAIN                   |
| concurrent_insert        | AUTO                       |
| connect_timeout          | 10                         |
| core_file                | OFF                        |
+--------------------------+----------------------------+
16 rows in set (0.01 sec)
```

6. Simple Test
- 현재 데이터 베이스들을 보여주고 testdb라는 데이터베이스를 추가 및 dept테이블 생성.

```
mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
4 rows in set (0.00 sec)

mysql> CREATE DATABASE testdb;
Query OK, 1 row affected (0.00 sec)

mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| sys                |
| testdb             |
+--------------------+
5 rows in set (0.00 sec)
```

```
mysql> use testdb;
Database changed
mysql> CREATE TABLE dept(
    -> dno INT NOT NULL,
    -> dname VARCHAR(32) NOT NULL,
    -> PRIMARY KEY (dno)
    -> );
Query OK, 0 rows affected (0.03 sec)

mysql> desc dept;
+-------+-------------+------+-----+---------+-------+
| Field | Type        | Null | Key | Default | Extra |
+-------+-------------+------+-----+---------+-------+
| dno   | int(11)     | NO   | PRI | NULL    |       |
| dname | varchar(32) | NO   |     | NULL    |       |
+-------+-------------+------+-----+---------+-------+
2 rows in set (0.00 sec)
```

- 생성한 dept테이블에 데이터들을 삽입, 삽입된 데이터들을 확인한 후 테이블과 testdb삭제

```
mysql> INSERT INTO dept VALUES(1, 'Computer Science');
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO dept VALUES(2, 'Computer Engineering');
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO dept VALUES(3, 'Mathematics');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO dept VALUES(4, '소프트웨어학과');
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO dept VALUES(5, '전자공학과');
Query OK, 1 row affected (0.00 sec)

mysql> SELECT * from dept;
+-----+----------------------+
| dno | dname                |
+-----+----------------------+
|   1 | Computer Science     |
|   2 | Computer Engineering |
|   3 | Mathematics          |
|   4 | 소프트웨어학과       |
|   5 | 전자공학과           |
+-----+----------------------+
5 rows in set (0.00 sec)
```

```
mysql> drop table dept;
Query OK, 0 rows affected (0.02 sec)

mysql> show tables;
Empty set (0.00 sec)

mysql> drop database testdb;
Query OK, 0 rows affected (0.00 sec)

mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
4 rows in set (0.00 sec)
```

7. MySQL의 information_schema, mysql, performance_schema, sys 데이터베이스들의 역할
- information_schema: 시스템의 카탈로그 데이터들을 갖고 있는 데이터베이스로 데이터베이스들의 테이블이나 칼럼같은 데이터들에 접근하는 것에 도움이 되는 데이터베이스 같습니다.
- mysql: MySQL 서버의 사용자나 권한에 관련된 정보들을 포함하고 있습니다. 권한이나 암호, 인증같은 것들에 도움을 주는 데이터베이스 같습니다.
- performance_schema: MySQL서버의 성능과 관련되어진 정보들을 모니터링 하는 데이터베이스 입니다.
- sys: MySQL 서버의 성능 모니터링과 최적화를 위한 performance_shema를 도와주는 데이터베이스 입니다.

8. 추가적인 MySQL관련 명령어를 테스트
- dummyDatabase와 dummyTable을 생성

```
mysql> CREATE DATABASE dummyDatabase;
Query OK, 1 row affected (0.00 sec)

mysql> use dummyDatabase;
Database changed
mysql> CREATE TABLE dummyTable(
    -> dno INT NOT NULL,
    -> dname VARCHAR(32) NOT NULL,
    -> PRIMARY KEY (dno)
    -> );
Query OK, 0 rows affected (0.03 sec)
```

- Update 명령어 테스트

```
mysql> INSERT INTO dummyTable VALUES(0, 'dummyOne');
Query OK, 1 row affected (0.00 sec)

mysql> UPDATE dummyTable SET dno = 4 WHERE dname = 'dummyOne';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

```
mysql> SELECT * from dummyTable;
+-----+----------+
| dno | dname    |
+-----+----------+
|   4 | dummyOne |
+-----+----------+
1 row in set (0.00 sec)
```

- Where 명령어와 Order명령어 테스트

```
mysql> INSERT INTO dummyTable VALUES(0, 'dummyTwo');
Query OK, 1 row affected (0.01 sec)

mysql> SELECT * FROM dummyTable WHERE dno > -1;
+-----+----------+
| dno | dname    |
+-----+----------+
|   0 | dummyTwo |
|   4 | dummyOne |
+-----+----------+
2 rows in set (0.00 sec)

mysql> UPDATE dummyTable SET dno = 7 WHERE dname = 'dummyTwo';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> SELECT * FROM dummyTable ORDER BY dno ASC;
+-----+----------+
| dno | dname    |
+-----+----------+
|   4 | dummyOne |
|   7 | dummyTwo |
+-----+----------+
2 rows in set (0.00 sec)
```