프로젝트 배포

프로젝트 배포 과정

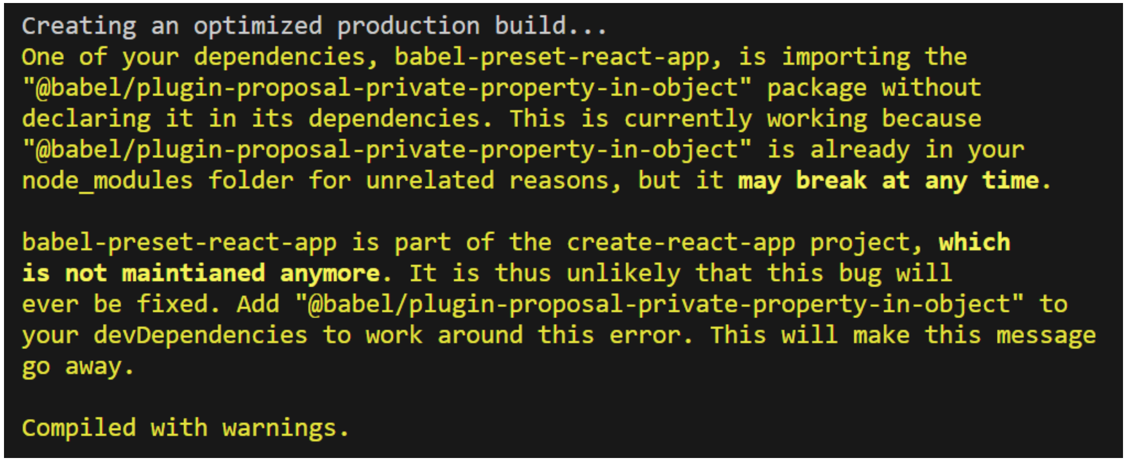
- 프로젝트 배포 과정
    - Django + React
    - 깃 허브에 업로드
    - AWS EC2 서버
        - MySQL 설치
        - Django 배포

1. Django + React

- Django + React
  - 리액트 앱 빌드해서 Django 프로젝트에 넣기
  - 리액트 프로젝트를 빌드하면 build 폴더 생성되고
  - static 폴더 안에 css와 js 폴더 및 파일 생성됨

리액트 프로젝트 빌드 하기 전에

npm install --save-dev @babel/plugin-proposal-private-property-in-object

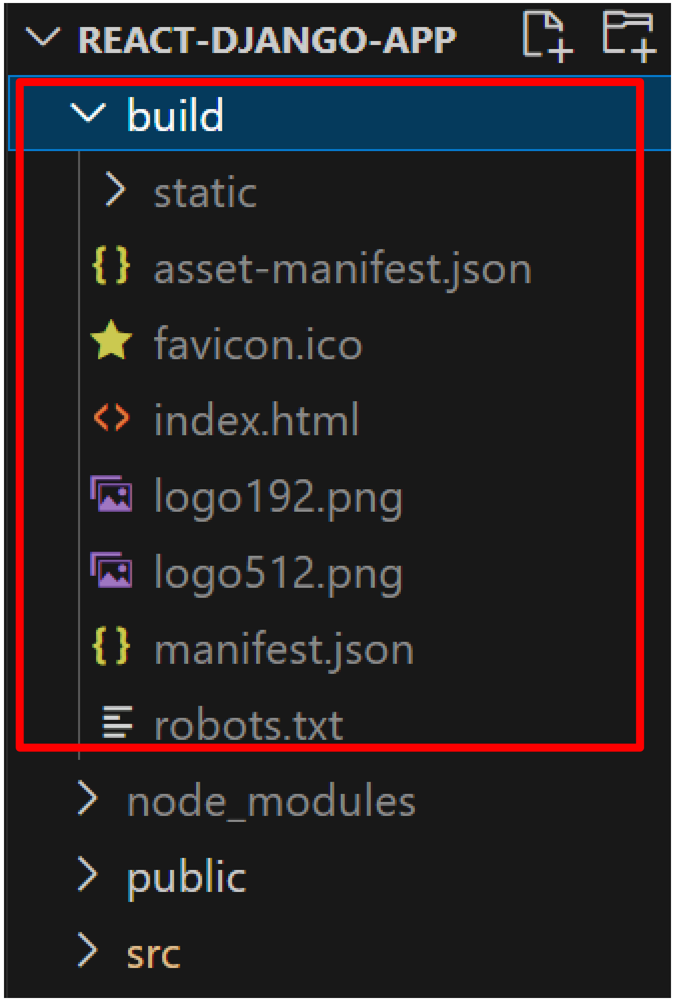


```
Creating an optimized production build...
One of your dependencies, babel-preset-react-app, is importing the
"@babel/plugin-proposal-private-property-in-object" package without
declaring it in its dependencies. This is currently working because
"@babel/plugin-proposal-private-property-in-object" is already in your
node_modules folder for unrelated reasons, but it may break at any time.

babel-preset-react-app is part of the create-react-app project, which
is not maintianed anymore. It is thus unlikely that this bug will
ever be fixed. Add "@babel/plugin-proposal-private-property-in-object" to
your devDependencies to work around this error. This will make this message
go away.

Compiled with warnings.
```

CRA(Create-React-App) 환경 지원하지 않는 라이브러리 설치 시 발생

리액트 앱 빌드

npm run build

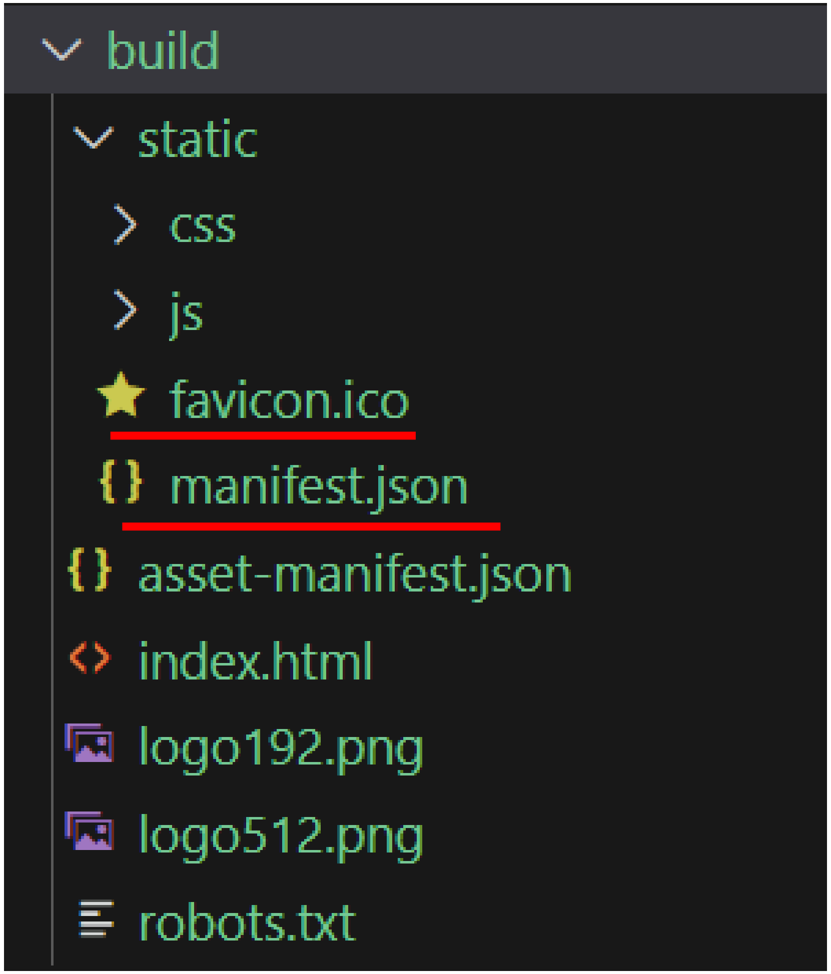


REACT-DJANGO-APP
- build
  - static
  - asset-manifest.json
  - favicon.ico
  - index.html
  - logo192.png
  - logo512.png
  - manifest.json
  - robots.txt
- node_modules
- public
- src

생성됨

build 내용 Django 프로젝트에 넣기

build 폴더 그대로 복사해서 최상위 위치에 저장

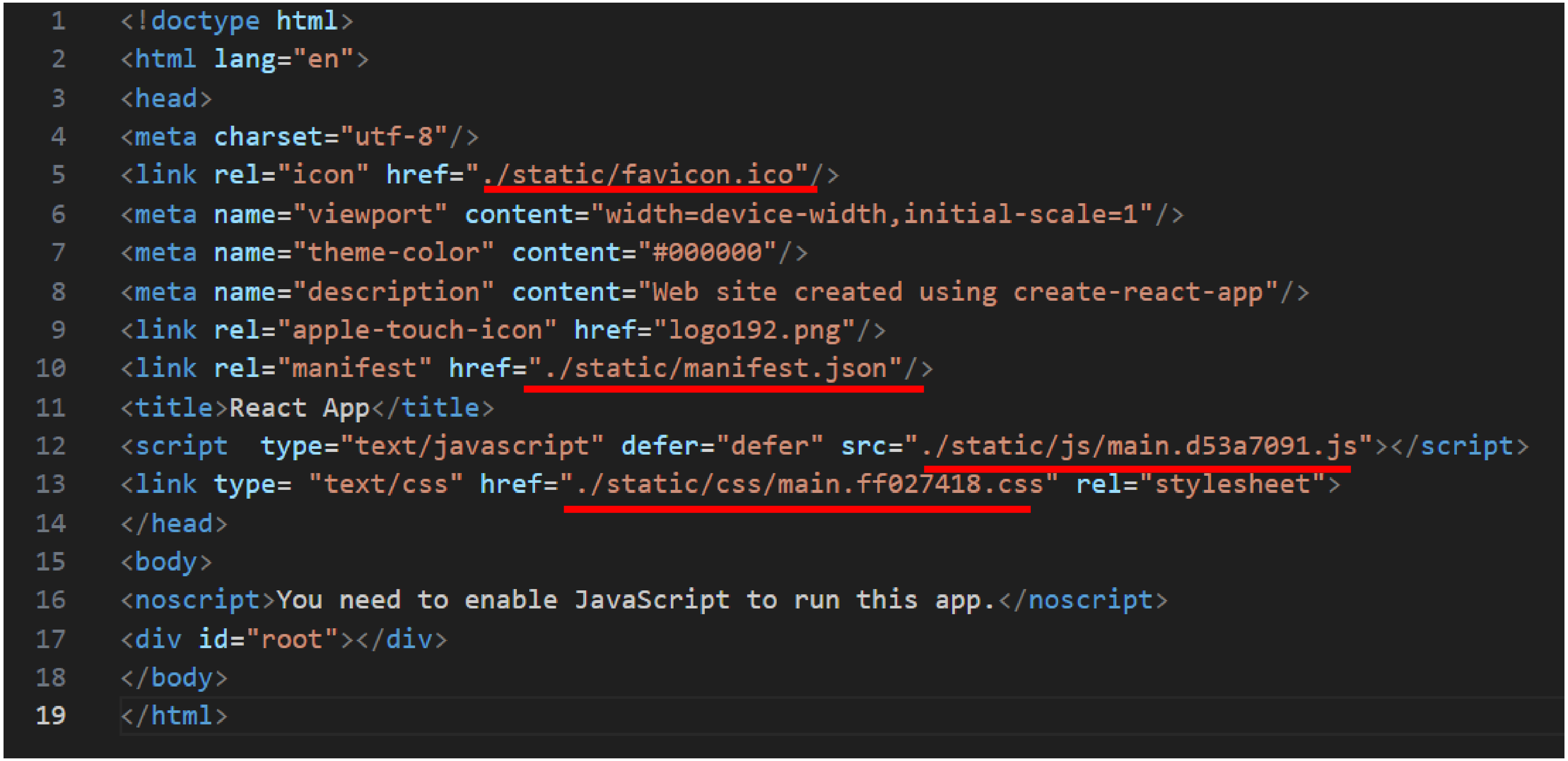


build
static
css
js
favicon.ico
manifest.json
asset-manifest.json
index.html
logo192.png
logo512.png
robots.txt

favicon.ico와 manifest.json을 static 폴더에 넣으면 못찾겠다는 오류 없음

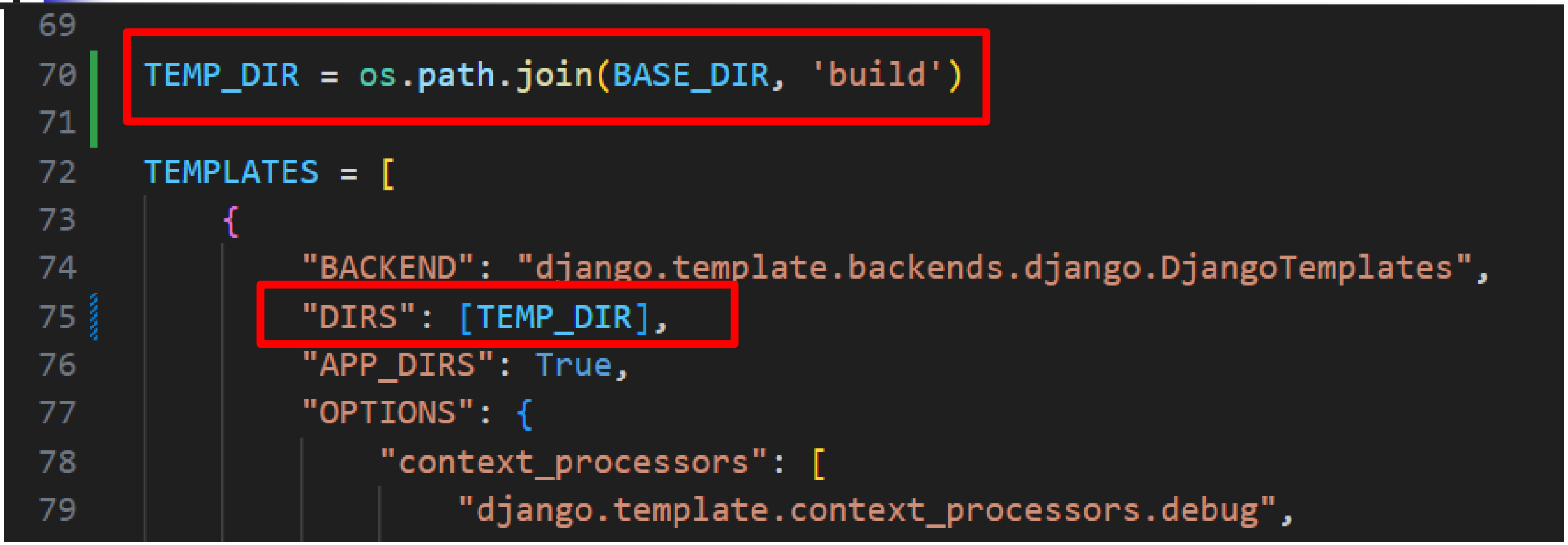
index.html

모든 경로 앞에 .(점) 붙임

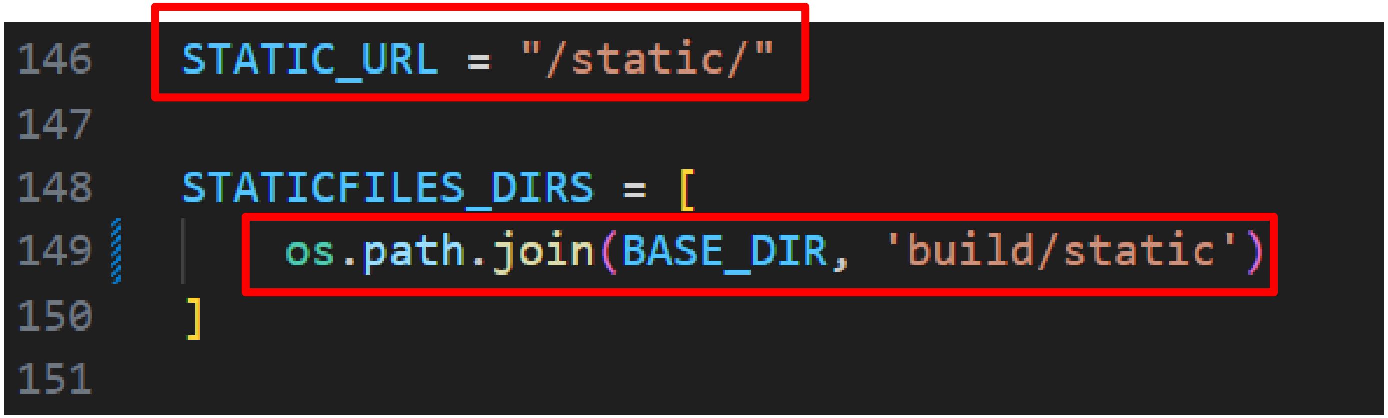


```html
<!doctype html>
<html lang="en">
<head>
<meta charset="utf-8"/>
<link rel="icon" href="./static/favicon.ico"/>
<meta name="viewport" content="width=device-width,initial-scale=1"/>
<meta name="theme-color" content="#000000"/>
<meta name="description" content="Web site created using create-react-app"/>
<link rel="apple-touch-icon" href="logo192.png"/>
<link rel="manifest" href="./static/manifest.json"/>
<title>React App</title>
<script  type="text/javascript" defer="defer" src="./static/js/main.d53a7091.js"></script>
<link type= "text/css" href="./static/css/main.ff027418.css" rel="stylesheet">
</head>
<body>
<noscript>You need to enable JavaScript to run this app.</noscript>
<div id="root"></div>
</body>
</html>
```

template, static 경로 설정 : settings.py

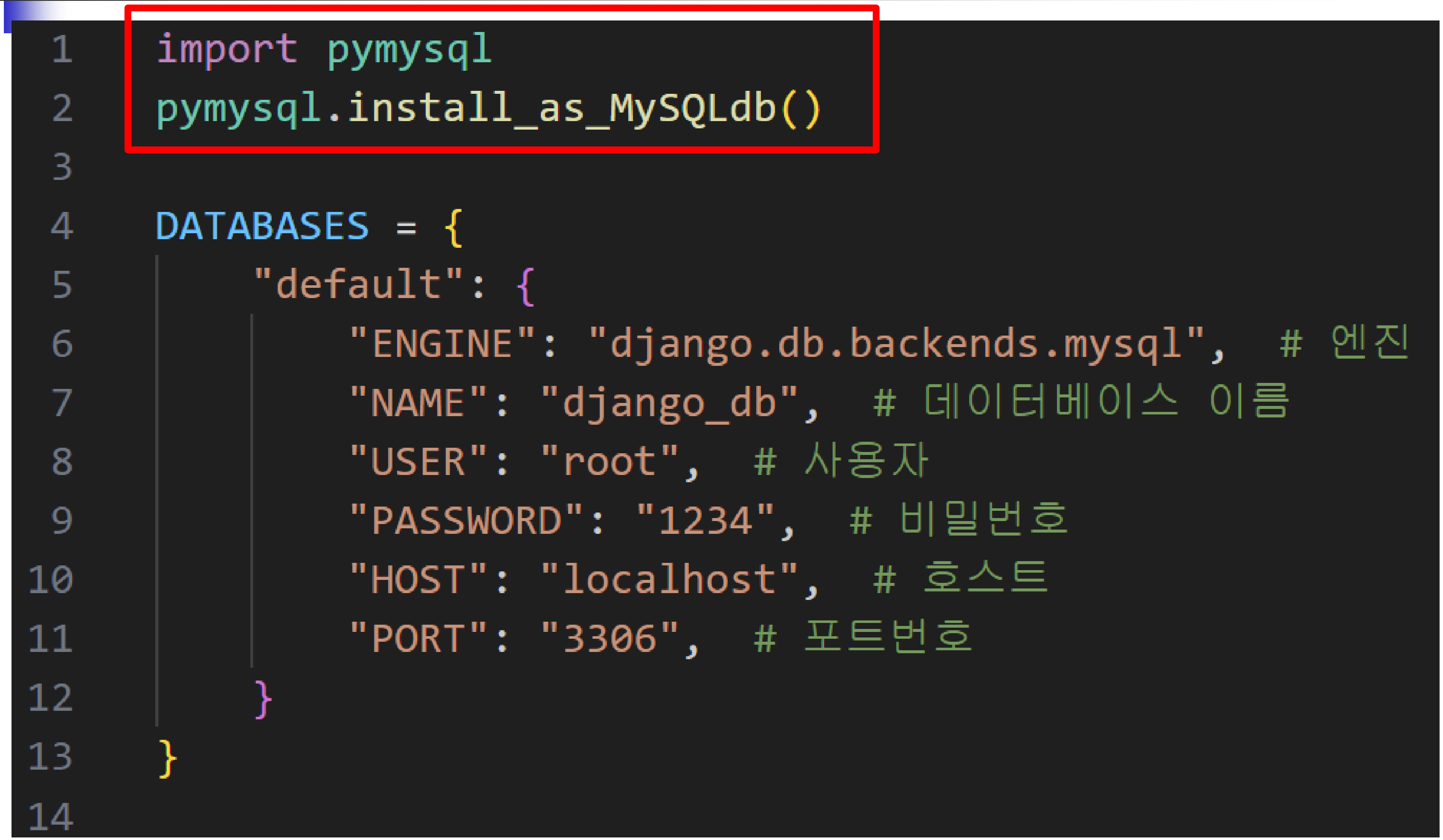


```python

TEMP_DIR = os.path.join(BASE_DIR, 'build')

TEMPLATES = [
    {
        "BACKEND": "django.template.backends.django.DjangoTemplates",
        "DIRS": [TEMP_DIR],
        "APP_DIRS": True,
        "OPTIONS": {
            "context_processors": [
                "django.template.context_processors.debug",
```



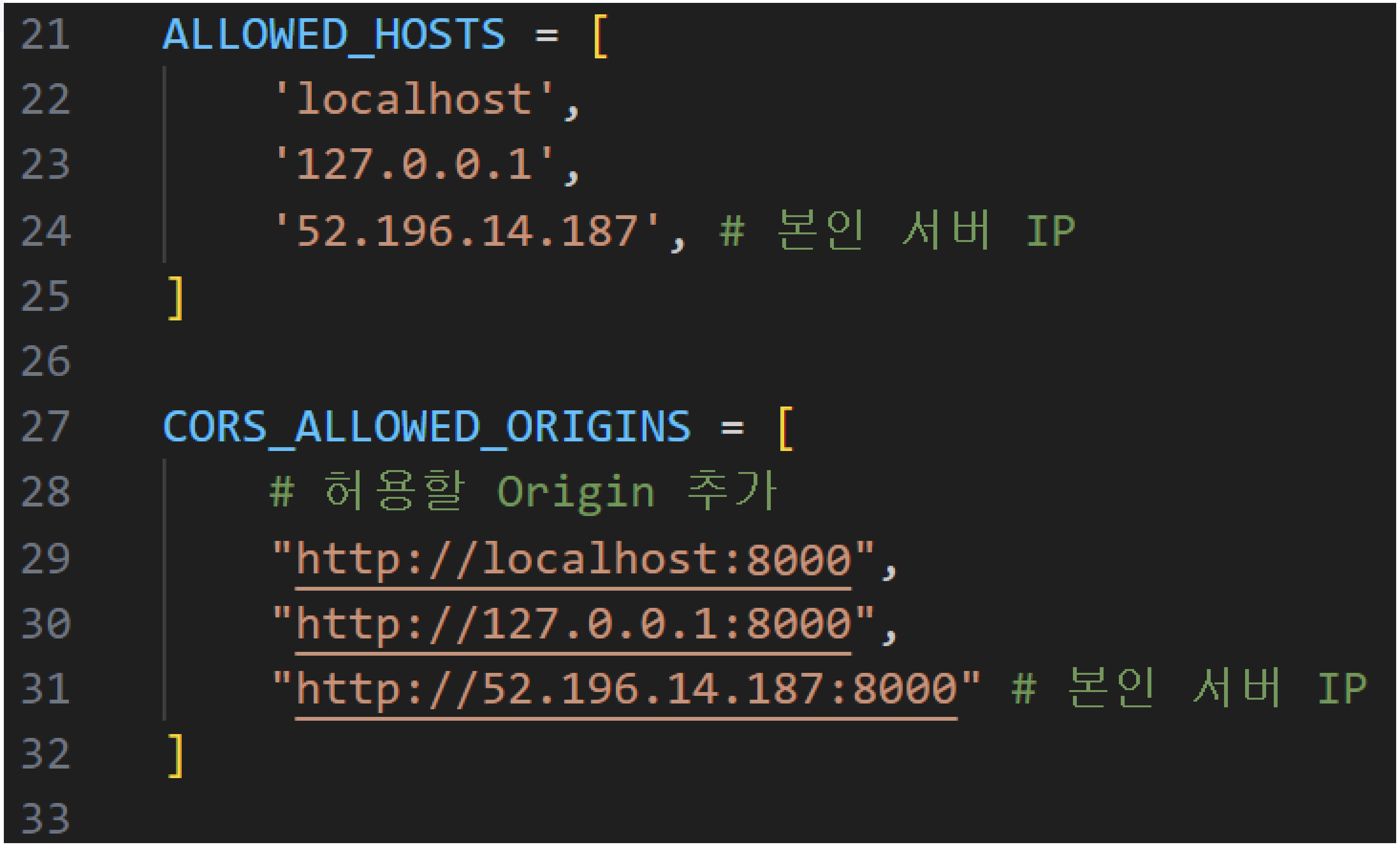
```python
STATIC_URL = "/static/"

STATICFILES_DIRS = [
    os.path.join(BASE_DIR, 'build/static')
]

```

db_settings.py에 추가



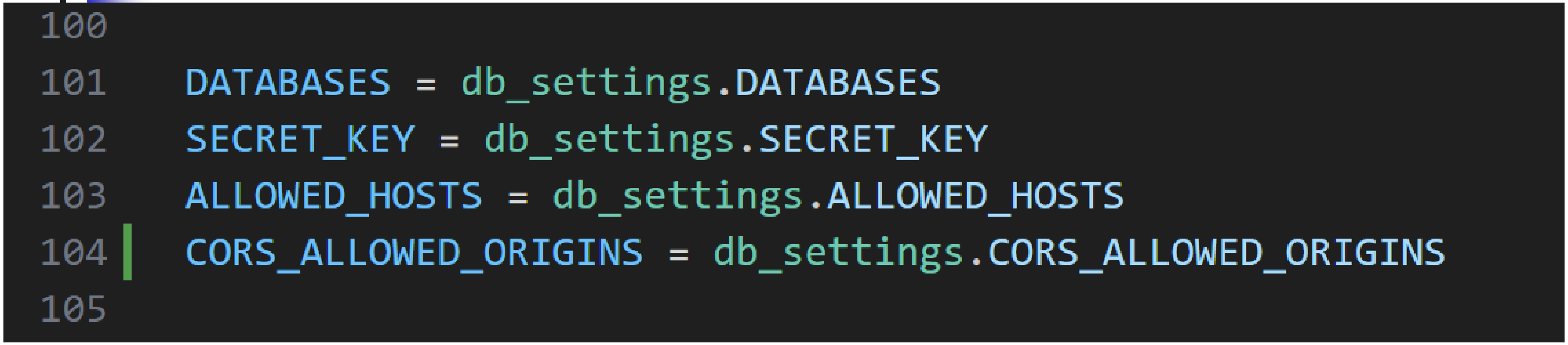
```
import pymysql
pymysql.install_as_MySQLdb()

DATABASES = {
    "default": {
        "ENGINE": "django.db.backends.mysql",  # 엔진
        "NAME": "django_db",  # 데이터베이스 이름
        "USER": "root",  # 사용자
        "PASSWORD": "1234",  # 비밀번호
        "HOST": "localhost",  # 호스트
        "PORT": "3306",  # 포트번호
    }
}
```

db_settings.py에 계속

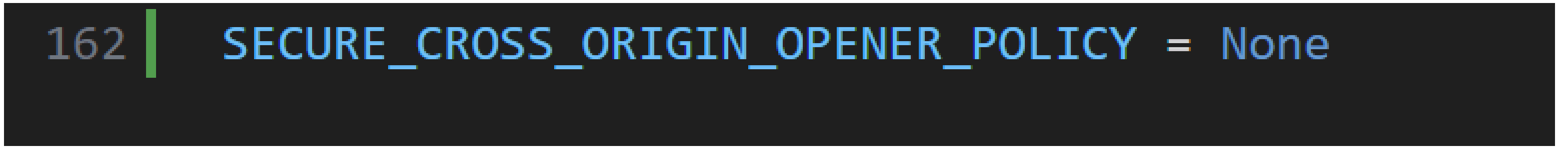


```
ALLOWED_HOSTS = [
    'localhost',
    '127.0.0.1',
    '52.196.14.187', # 본인 서버 IP
]

CORS_ALLOWED_ORIGINS = [
    # 허용할 Origin 추가
    "http://localhost:8000",
    "http://127.0.0.1:8000",
    "http://52.196.14.187:8000" # 본인 서버 IP
]

```

settings.py에 추가



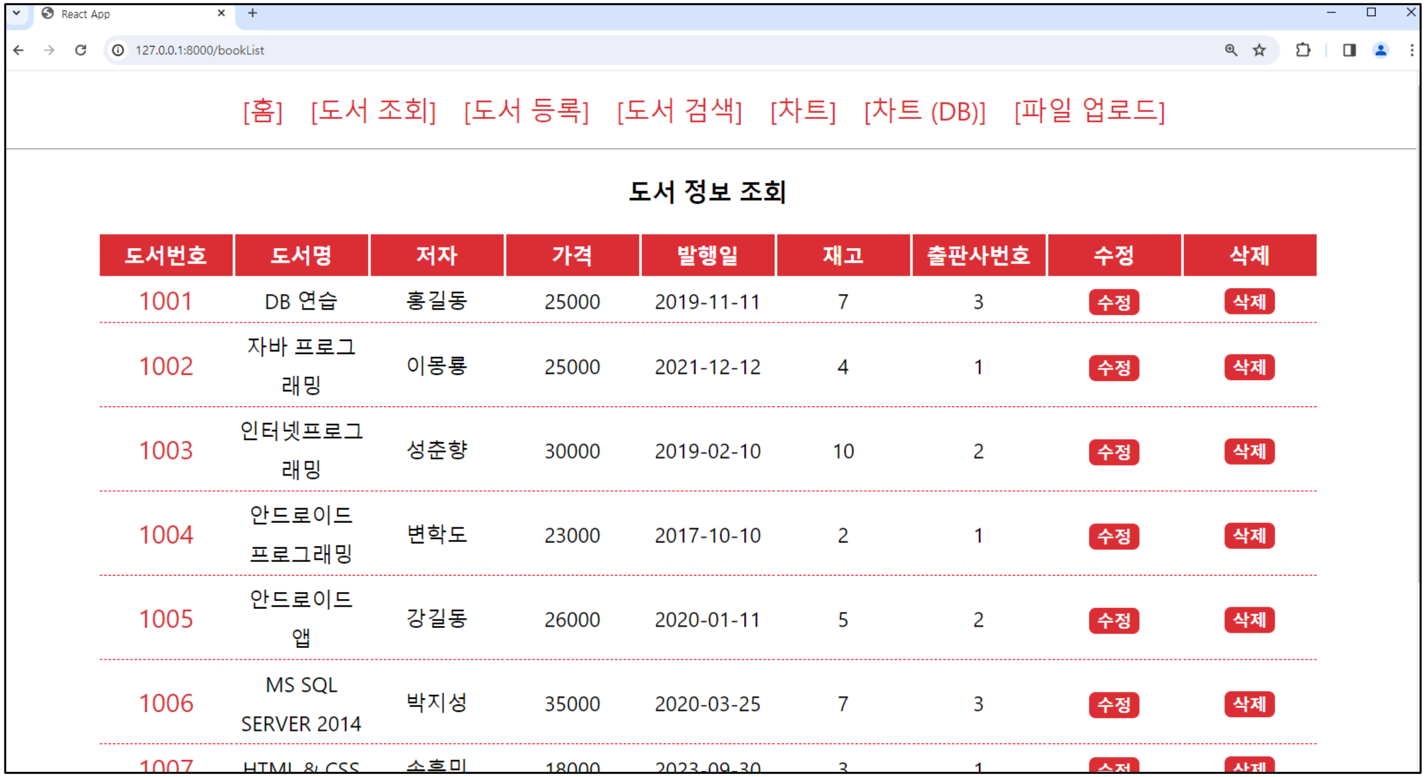
```python

DATABASES = db_settings.DATABASES
SECRET_KEY = db_settings.SECRET_KEY
ALLOWED_HOSTS = db_settings.ALLOWED_HOSTS
CORS_ALLOWED_ORIGINS = db_settings.CORS_ALLOWED_ORIGINS

```



```python
SECURE_CROSS_ORIGIN_OPENER_POLICY = None
```

Django 프로젝트 실행시켜서 확인

http://127.0.0.1:8000/



[홈] [도서 조회] [도서 등록] [도서 검색] [차트] [차트 (DB)] [파일 업로드]

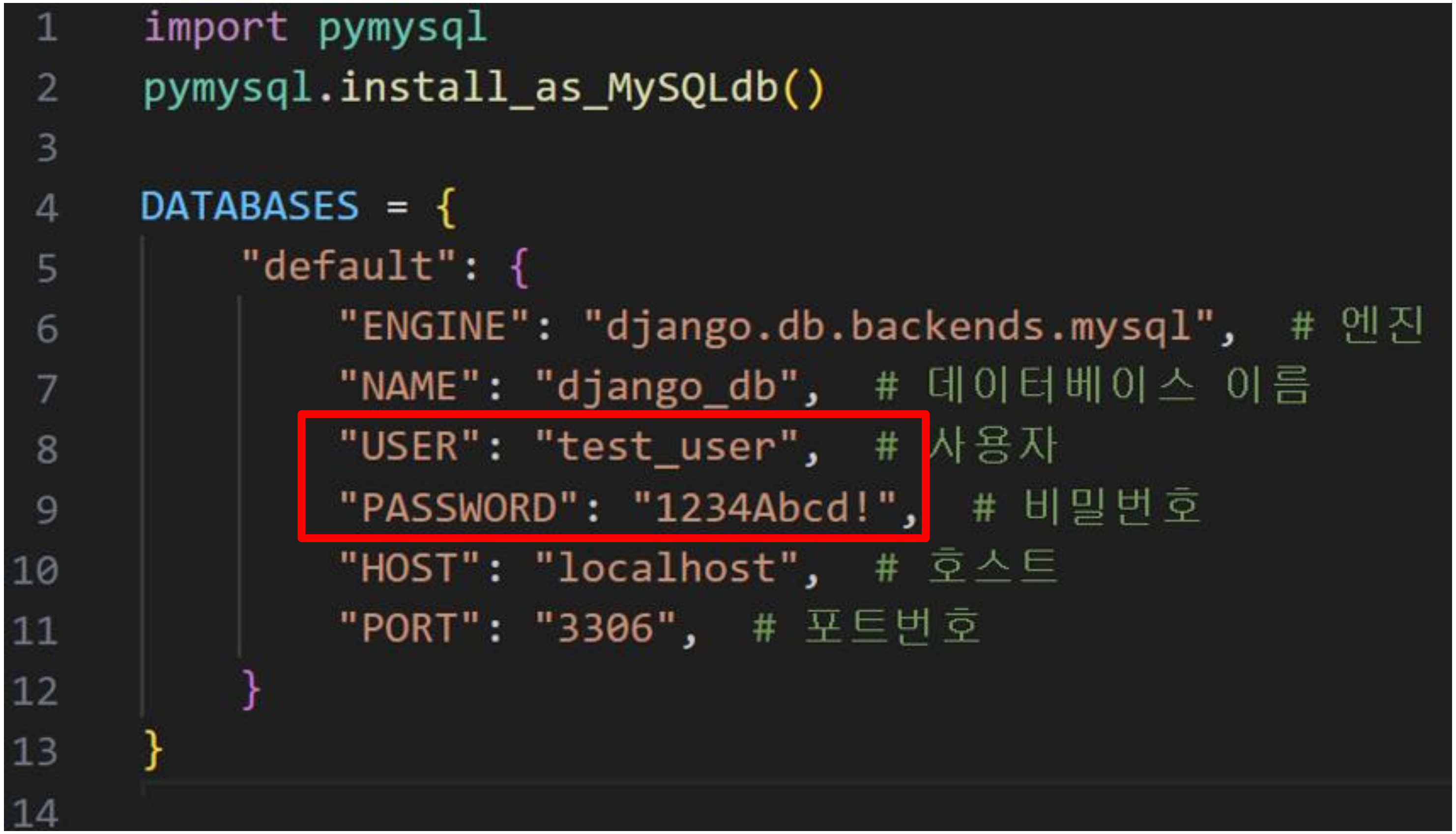
도서 정보 조회

도서번호	도서명	저자	가격	발행일	재고	출판사번호	수정	삭제
1001	DB 연습	홍길동	25000	2019-11-11	7	3	수정	삭제
1002	자바 프로그래밍	이몽룡	25000	2021-12-12	4	1	수정	삭제
1003	인터넷프로그래밍	성춘향	30000	2019-02-10	10	2	수정	삭제
1004	안드로이드 프로그래밍	변학도	23000	2017-10-10	2	1	수정	삭제
1005	안드로이드 앱	강길동	26000	2020-01-11	5	2	수정	삭제
1006	MS SQL SERVER 2014	박지성	35000	2020-03-25	7	3	수정	삭제
1007	HTML & CSS	손흥민	18000	2023-09-30	3	1	수정	삭제

모든 기능 확인

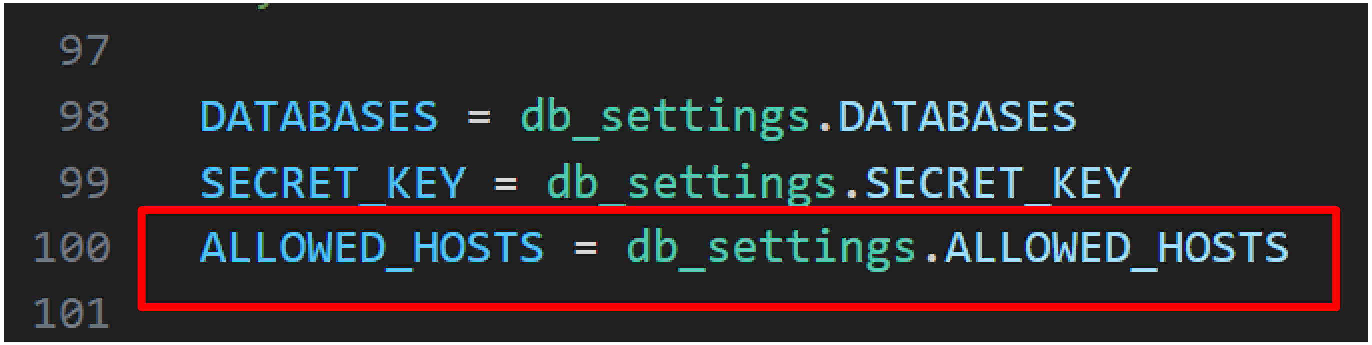
서버에 올리기 위해

db_settings.py 변경



```python
import pymysql
pymysql.install_as_MySQLdb()

DATABASES = {
    "default": {
        "ENGINE": "django.db.backends.mysql",  # 엔진
        "NAME": "django_db",  # 데이터베이스 이름
        "USER": "test_user",  # 사용자
        "PASSWORD": "1234Abcd!",  # 비밀번호
        "HOST": "localhost",  # 호스트
        "PORT": "3306",  # 포트번호
    }
}

```

settings.py

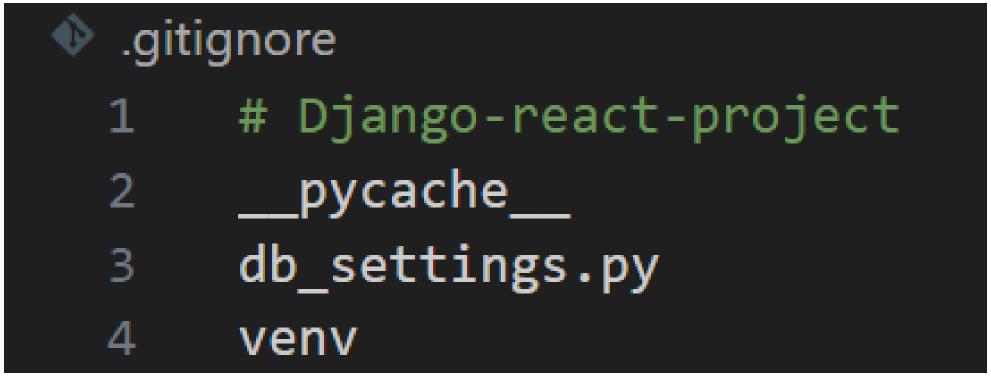


```python
DATABASES = db_settings.DATABASES
SECRET_KEY = db_settings.SECRET_KEY
ALLOWED_HOSTS = db_settings.ALLOWED_HOSTS
```

2. 깃 허브에 업로드

- 깃 허브에 업로드
  - .gitignore 작성
  - requirements.txt 생성
  - 깃 허브에 레포지토리 생성
  - 업로드

.gitignore 생성



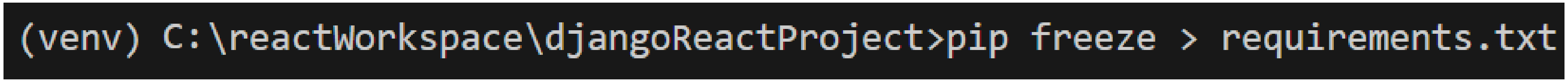
.gitignore

```
# Django-react-project
__pycache__
db_settings.py
venv
```

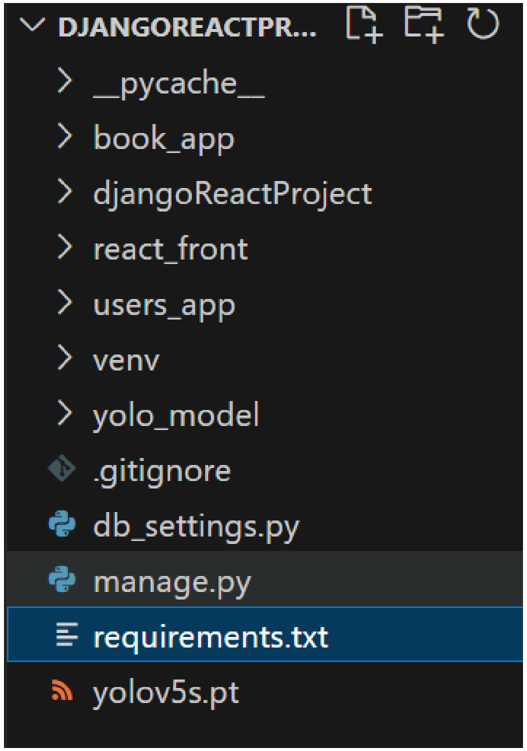
```
# Django-react-project
__pycache__
db_settings.py
venv
```

requirements.txt 생성

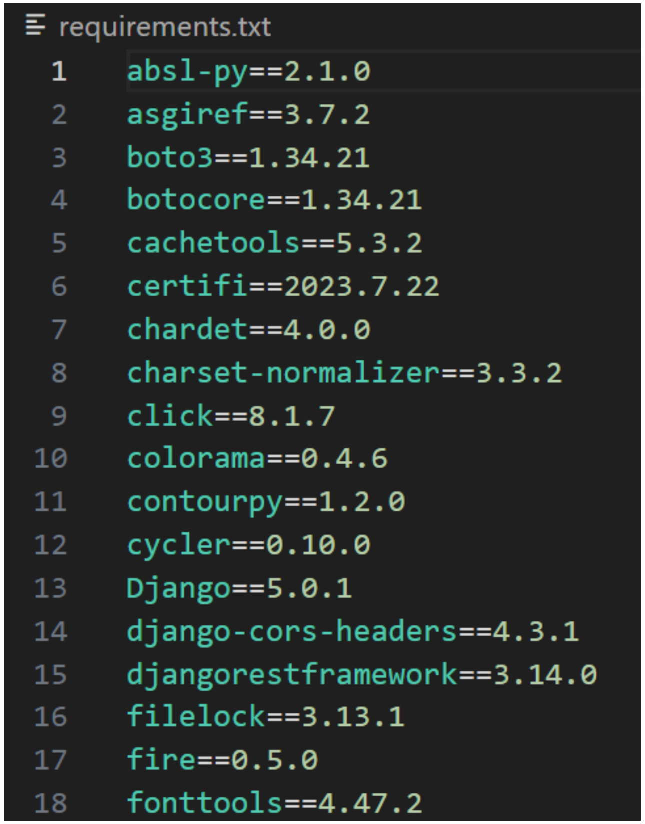
- **pip freeze > requirements.txt**



```
(venv) C:\reactWorkspace\djangoReactProject>pip freeze > requirements.txt
```



DJANGOREACTPR...
- _pycache_
- book_app
- djangoReactProject
- react_front
- users_app
- venv
- yolo_model
- .gitignore
- db_settings.py
- manage.py
- requirements.txt
- yolov5s.pt



requirements.txt

```
absl-py==2.1.0
asgiref==3.7.2
boto3==1.34.21
botocore==1.34.21
cachetools==5.3.2
certifi==2023.7.22
chardet==4.0.0
charset-normalizer==3.3.2
click==8.1.7
colorama==0.4.6
contourpy==1.2.0
cycler==0.10.0
Django==5.0.1
django-cors-headers==4.3.1
djangorestframework==3.14.0
filelock==3.13.1
fire==0.5.0
fonttools==4.47.2
```

설치해야 할
필수 패키지 목록

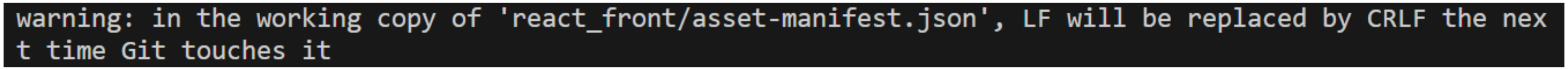
requirements.txt

패키지 버전 변경

new_requirements.txt 파일 내용으로 변경

깃 허브에 업로드

- repository 생성
  - django_react
  - 주소 복사
- Git Bash 사용 : (C:\reactWorkspace\djangoReactProject)
  - git init
  - git add .
  - git commit -m "django-react project"
  - git config --global user.name " 본인 아이디"
  - git config --global user.email 본인이메일
  - git remote add origin "https://주소"
  - git pull origin main
  - git push -u origin +main
- repository 확인



```
warning: in the working copy of 'react_front/asset-manifest.json', LF will be replaced by CRLF the nex
t time Git touches it
```

윈도우
git config core.autocrlf true
(git config --global core.autocrlf true : 시스템 전체)

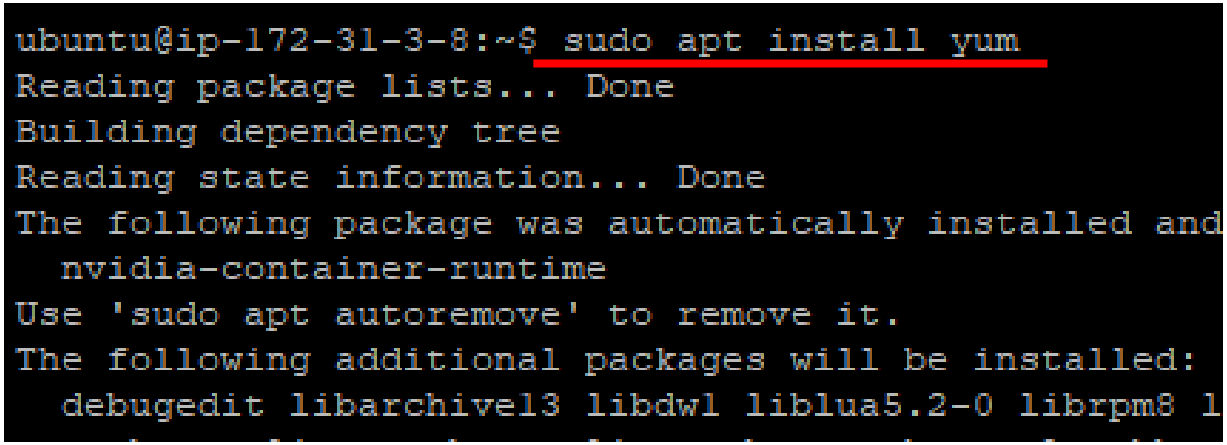
Mac
git config core.autocrlf input

3. MySQL 설치

- 서버에 MySQL 서버 하나만 설치
- 팀원 중 **1명만 설치 진행**
- PuTTy에서 ubuntu로 로그인

yum 설치

sudo apt install yum



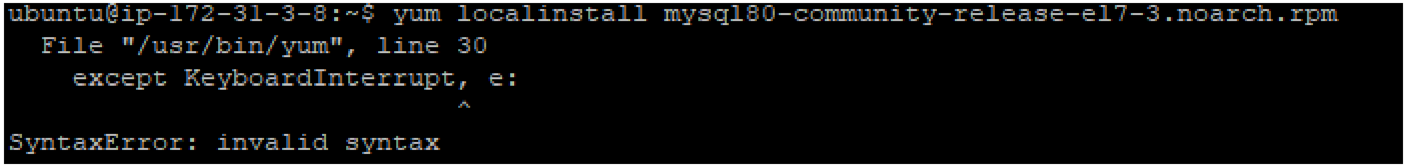
```
ubuntu@ip-172-31-3-8:~$ sudo apt install yum
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following package was automatically installed and
  nvidia-container-runtime
Use 'sudo apt autoremove' to remove it.
The following additional packages will be installed:
  debugedit libarchive13 libdwl liblua5.2-0 librpm8 l
```

sudo (super user do)

- super user 권한 부여

설치

오류



```
ubuntu@ip-172-31-3-8:~$ yum localinstall mysql80-community-release-el7-3.noarch.rpm
  File "/usr/bin/yum", line 30
    except KeyboardInterrupt, e:
                            ^
SyntaxError: invalid syntax
```

sudo vim /usr/bin/yum

첫 줄을 #!/usr/bin/python2 로 변경

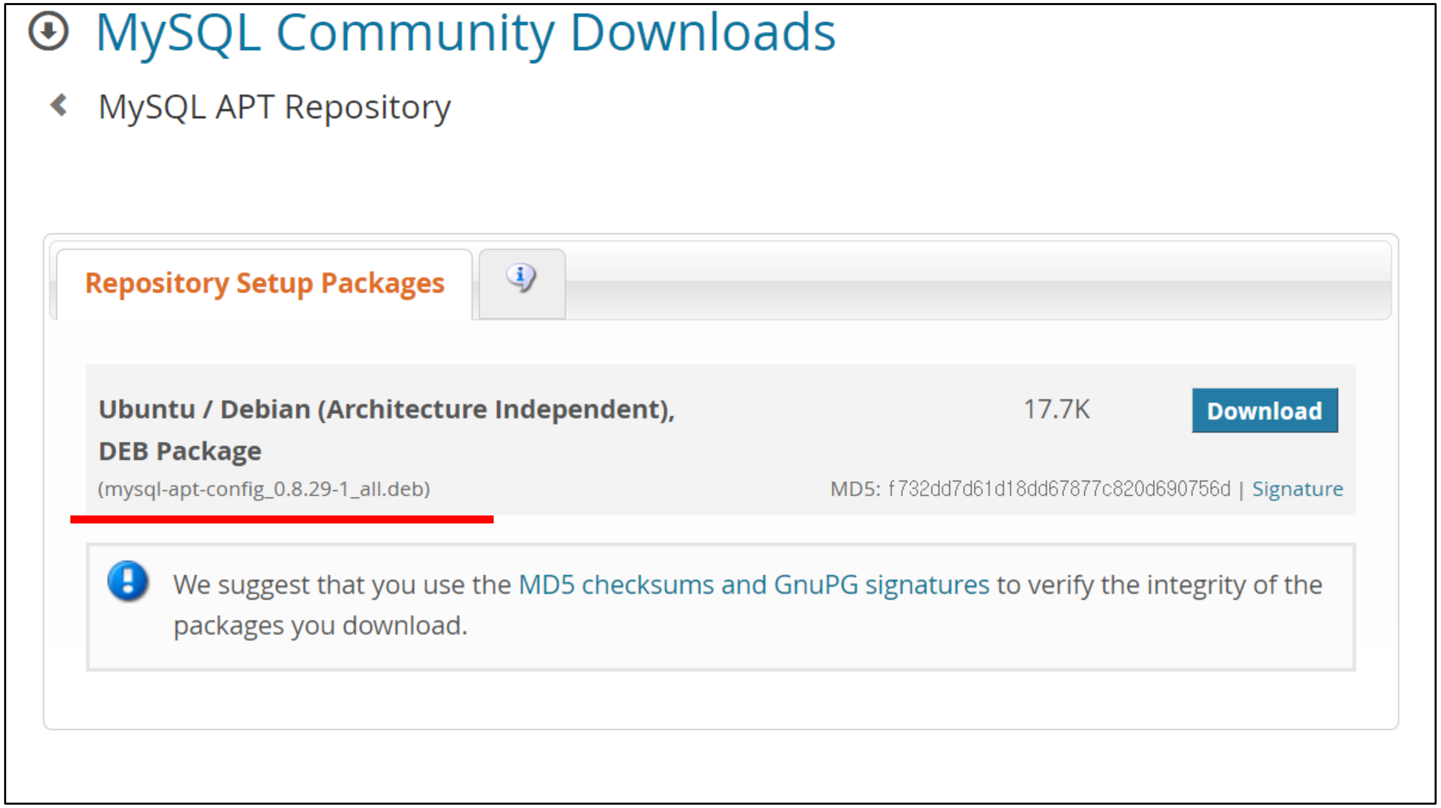
(i 누르고 맨 뒤로 이동해서 2 입력/저장/종료)

- **ESC**
- **:wq!**

다시 실행

다운로드할 Repository 확인만 할 것

https://dev.mysql.com/downloads/repo/apt/



MySQL Community Downloads

MySQL APT Repository

Ubuntu / Debian (Architecture Independent), DEB Package
(mysql-apt-config_0.8.29-1_all.deb)

17.7K

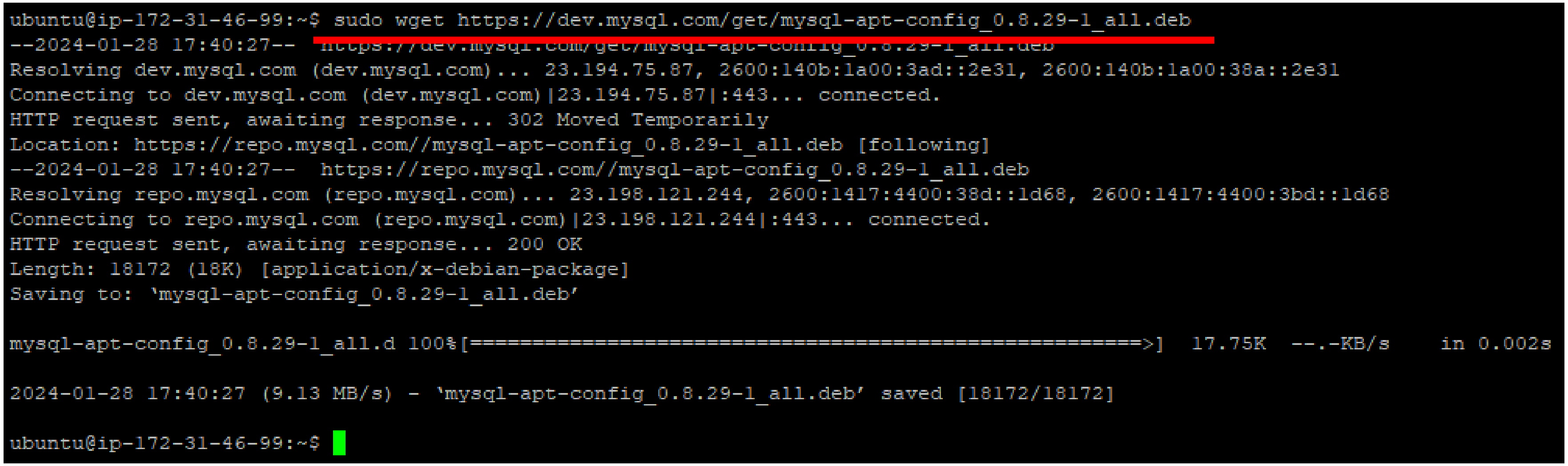
MD5: f732dd7d61d18dd67877c820d690756d | Signature

We suggest that you use the MD5 checksums and GnuPG signatures to verify the integrity of the packages you download.

mysql-apt-config_0.8.29-1_all.deb

다운로드 Repository

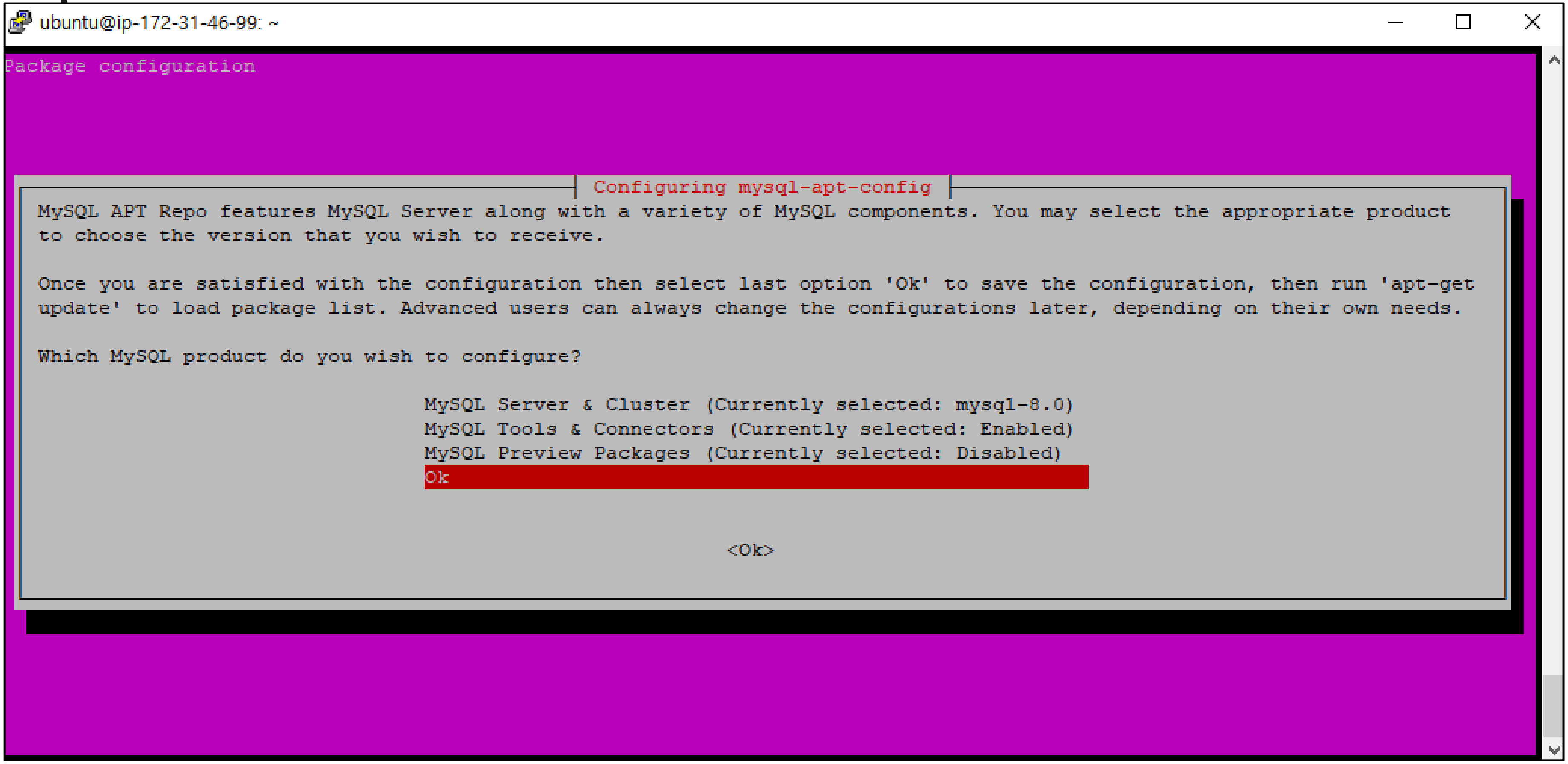
sudo wget https://dev.mysql.com/get/mysql-apt-config_0.8.22-1_all.deb



```
ubuntu@ip-172-31-46-99:~$ sudo wget https://dev.mysql.com/get/mysql-apt-config_0.8.29-1_all.deb
--2024-01-28 17:40:27--  https://dev.mysql.com/get/mysql-apt-config_0.8.29-1_all.deb
Resolving dev.mysql.com (dev.mysql.com)... 23.194.75.87, 2600:140b:1a00:3ad::2e31, 2600:140b:1a00:38a::2e31
Connecting to dev.mysql.com (dev.mysql.com)|23.194.75.87|:443... connected.
HTTP request sent, awaiting response... 302 Moved Temporarily
Location: https://repo.mysql.com//mysql-apt-config_0.8.29-1_all.deb [following]
--2024-01-28 17:40:27--  https://repo.mysql.com//mysql-apt-config_0.8.29-1_all.deb
Resolving repo.mysql.com (repo.mysql.com)... 23.198.121.244, 2600:1417:4400:38d::1d68, 2600:1417:4400:3bd::1d68
Connecting to repo.mysql.com (repo.mysql.com)|23.198.121.244|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 18172 (18K) [application/x-debian-package]
Saving to: 'mysql-apt-config_0.8.29-1_all.deb'

mysql-apt-config_0.8.29-1_all.d 100%[=======================================================>]  17.75K  --.-KB/s    in 0.002s

2024-01-28 17:40:27 (9.13 MB/s) - 'mysql-apt-config_0.8.29-1_all.deb' saved [18172/18172]

ubuntu@ip-172-31-46-99:~$
```

최신 버전 mysql-apt-config_0.8.29-1_all.deb 설치 불가하므로
0.8.22로 버전 낮춰서 다운로드

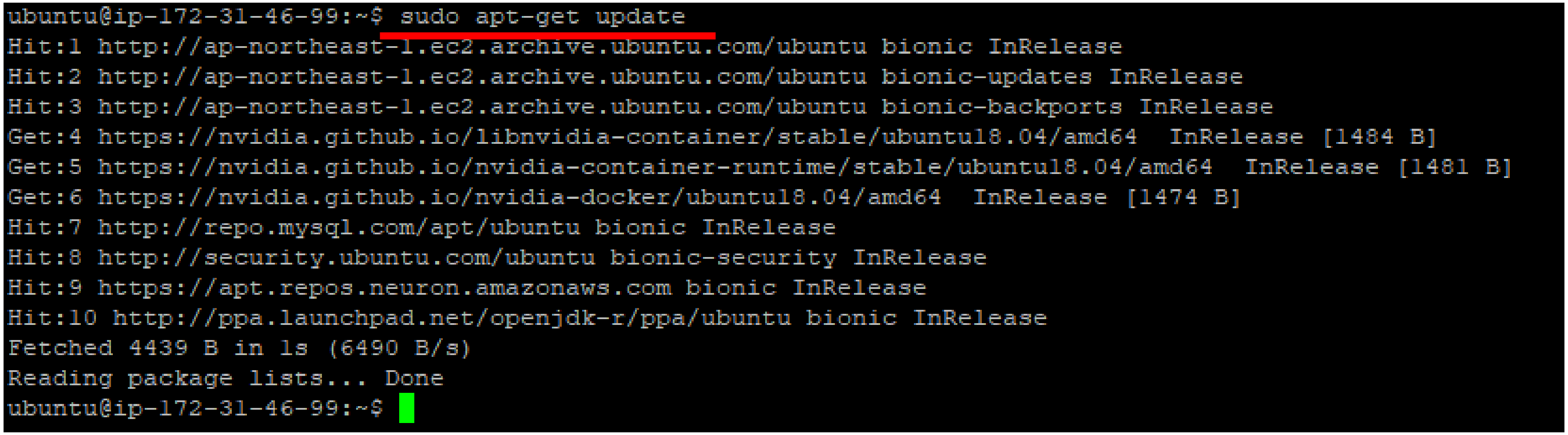
sudo dpkg -i mysql-apt-config_0.8.22-1_all.deb



```
Package configuration

                         Configuring mysql-apt-config
 MySQL APT Repo features MySQL Server along with a variety of MySQL components. You may select the appropriate product
 to choose the version that you wish to receive.

 Once you are satisfied with the configuration then select last option 'Ok' to save the configuration, then run 'apt-get
 update' to load package list. Advanced users can always change the configurations later, depending on their own needs.

 Which MySQL product do you wish to configure?

                                MySQL Server & Cluster (Currently selected: mysql-8.0)
                                MySQL Tools & Connectors (Currently selected: Enabled)
                                MySQL Preview Packages (Currently selected: Disabled)
                                Ok


                                                         <Ok>
```

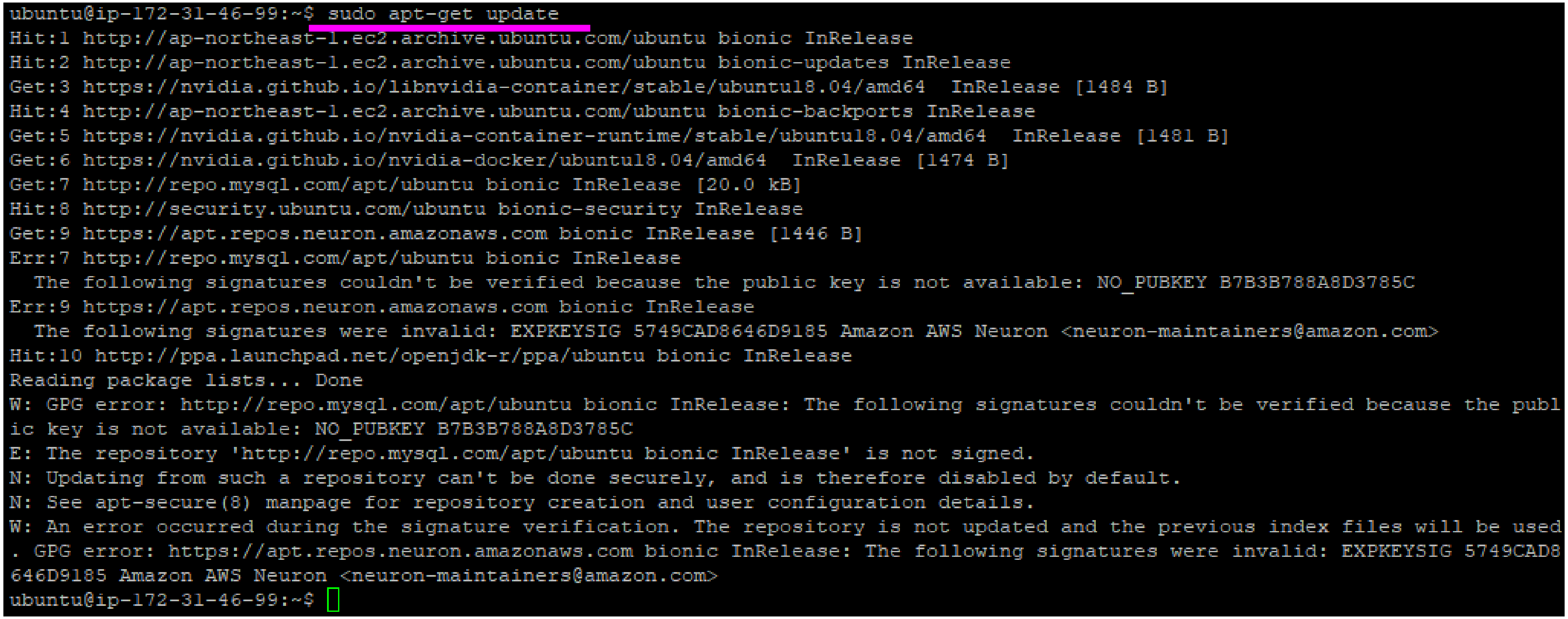
repository 추가

sudo apt-get update



```
ubuntu@ip-172-31-46-99:~$ sudo apt-get update
Hit:1 http://ap-northeast-1.ec2.archive.ubuntu.com/ubuntu bionic InRelease
Hit:2 http://ap-northeast-1.ec2.archive.ubuntu.com/ubuntu bionic-updates InRelease
Hit:3 http://ap-northeast-1.ec2.archive.ubuntu.com/ubuntu bionic-backports InRelease
Get:4 https://nvidia.github.io/libnvidia-container/stable/ubuntu18.04/amd64  InRelease [1484 B]
Get:5 https://nvidia.github.io/nvidia-container-runtime/stable/ubuntu18.04/amd64  InRelease [1481 B]
Get:6 https://nvidia.github.io/nvidia-docker/ubuntu18.04/amd64  InRelease [1474 B]
Hit:7 http://repo.mysql.com/apt/ubuntu bionic InRelease
Hit:8 http://security.ubuntu.com/ubuntu bionic-security InRelease
Hit:9 https://apt.repos.neuron.amazonaws.com bionic InRelease
Hit:10 http://ppa.launchpad.net/openjdk-r/ppa/ubuntu bionic InRelease
Fetched 4439 B in 1s (6490 B/s)
Reading package lists... Done
ubuntu@ip-172-31-46-99:~$
```

publick key 오류 날 경우



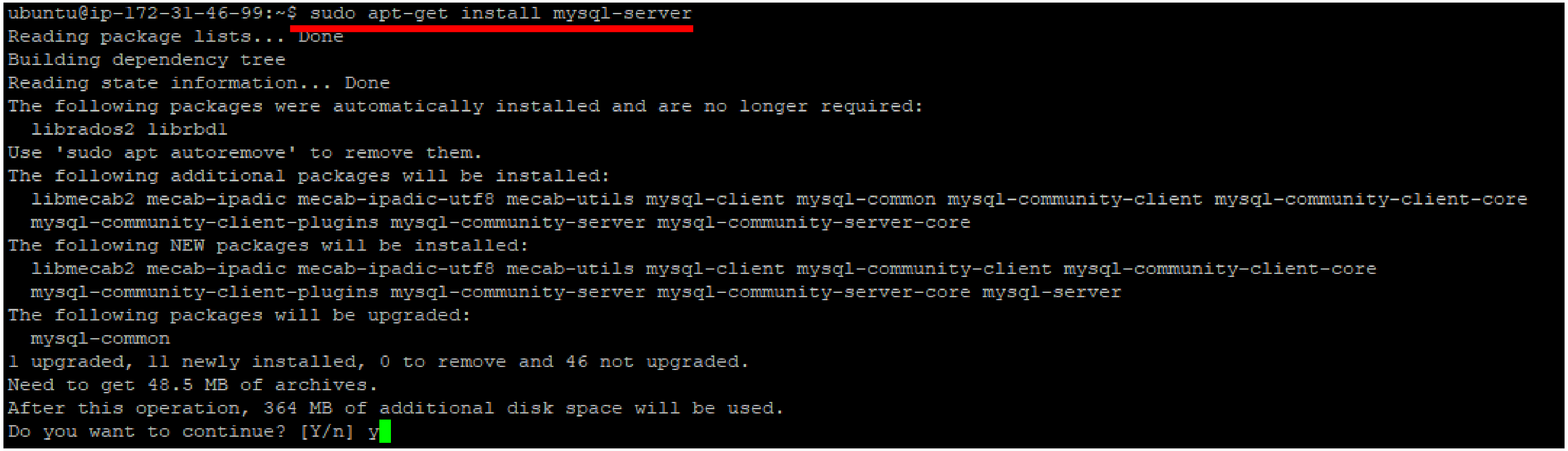
```
ubuntu@ip-172-31-46-99:~$ sudo apt-get update
Hit:1 http://ap-northeast-1.ec2.archive.ubuntu.com/ubuntu bionic InRelease
Hit:2 http://ap-northeast-1.ec2.archive.ubuntu.com/ubuntu bionic-updates InRelease
Get:3 https://nvidia.github.io/libnvidia-container/stable/ubuntu18.04/amd64  InRelease [1484 B]
Hit:4 http://ap-northeast-1.ec2.archive.ubuntu.com/ubuntu bionic-backports InRelease
Get:5 https://nvidia.github.io/nvidia-container-runtime/stable/ubuntu18.04/amd64  InRelease [1481 B]
Get:6 https://nvidia.github.io/nvidia-docker/ubuntu18.04/amd64  InRelease [1474 B]
Get:7 http://repo.mysql.com/apt/ubuntu bionic InRelease [20.0 kB]
Hit:8 http://security.ubuntu.com/ubuntu bionic-security InRelease
Get:9 https://apt.repos.neuron.amazonaws.com bionic InRelease [1446 B]
Err:7 http://repo.mysql.com/apt/ubuntu bionic InRelease
  The following signatures couldn't be verified because the public key is not available: NO_PUBKEY B7B3B788A8D3785C
Err:9 https://apt.repos.neuron.amazonaws.com bionic InRelease
  The following signatures were invalid: EXPKEYSIG 5749CAD8646D9185 Amazon AWS Neuron <neuron-maintainers@amazon.com>
Hit:10 http://ppa.launchpad.net/openjdk-r/ppa/ubuntu bionic InRelease
Reading package lists... Done
W: GPG error: http://repo.mysql.com/apt/ubuntu bionic InRelease: The following signatures couldn't be verified because the publ
ic key is not available: NO_PUBKEY B7B3B788A8D3785C
E: The repository 'http://repo.mysql.com/apt/ubuntu bionic InRelease' is not signed.
N: Updating from such a repository can't be done securely, and is therefore disabled by default.
N: See apt-secure(8) manpage for repository creation and user configuration details.
W: An error occurred during the signature verification. The repository is not updated and the previous index files will be used
. GPG error: https://apt.repos.neuron.amazonaws.com bionic InRelease: The following signatures were invalid: EXPKEYSIG 5749CAD8
646D9185 Amazon AWS Neuron <neuron-maintainers@amazon.com>
ubuntu@ip-172-31-46-99:~$
```

```
wget -qO - https://apt.repos.neuron.amazonaws.com/GPG-PUB-KEY-AMAZON-AWS-NEURON.PUB | sudo apt-key add -
```

우리는 오류 없음

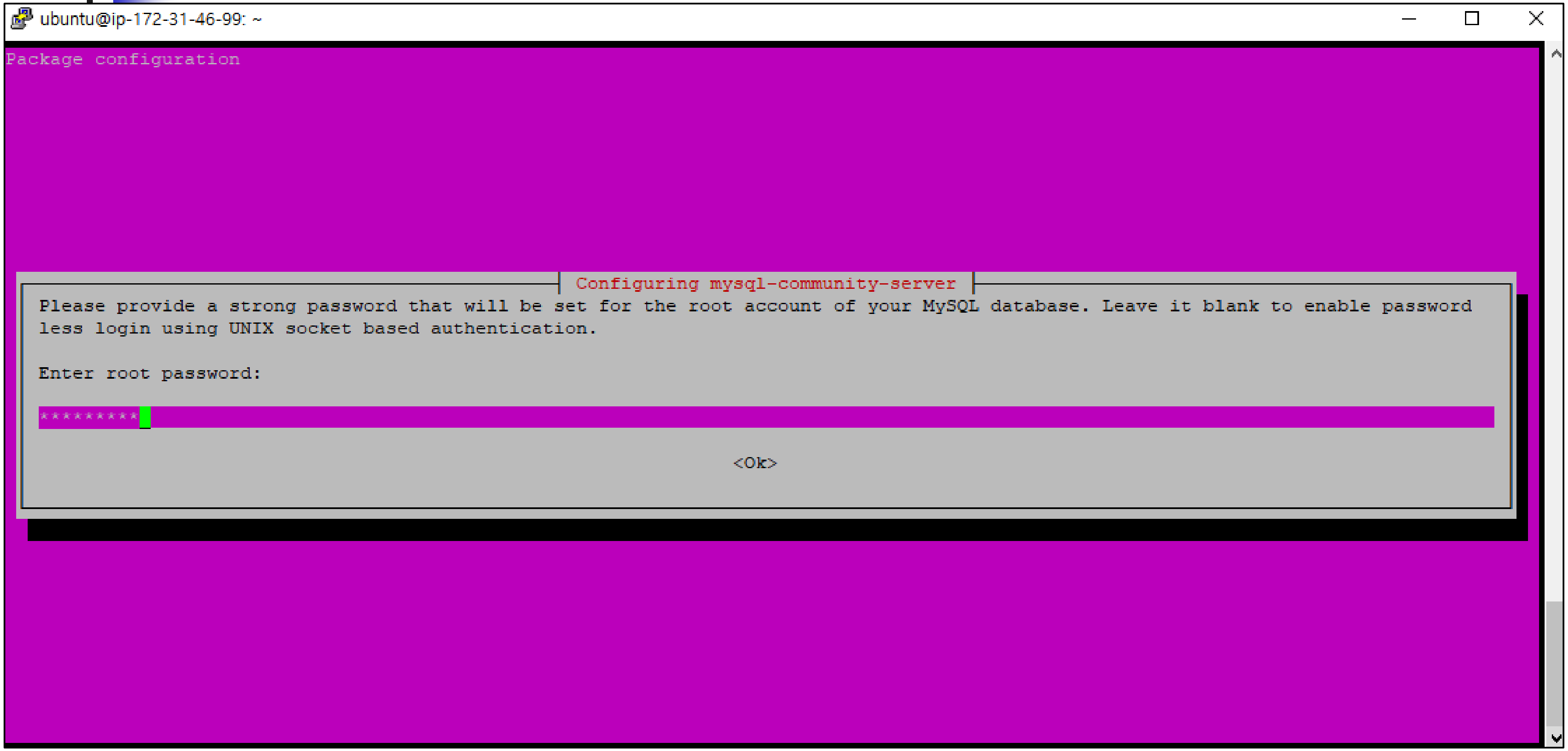
설치

sudo apt-get install mysql-server



```
ubuntu@ip-172-31-46-99:~$ sudo apt-get install mysql-server
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  librados2 librbdl
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  libmecab2 mecab-ipadic mecab-ipadic-utf8 mecab-utils mysql-client mysql-common mysql-community-client mysql-community-client-core
  mysql-community-client-plugins mysql-community-server mysql-community-server-core
The following NEW packages will be installed:
  libmecab2 mecab-ipadic mecab-ipadic-utf8 mecab-utils mysql-client mysql-community-client mysql-community-client-core
  mysql-community-client-plugins mysql-community-server mysql-community-server-core mysql-server
The following packages will be upgraded:
  mysql-common
1 upgraded, 11 newly installed, 0 to remove and 46 not upgraded.
Need to get 48.5 MB of archives.
After this operation, 364 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
```

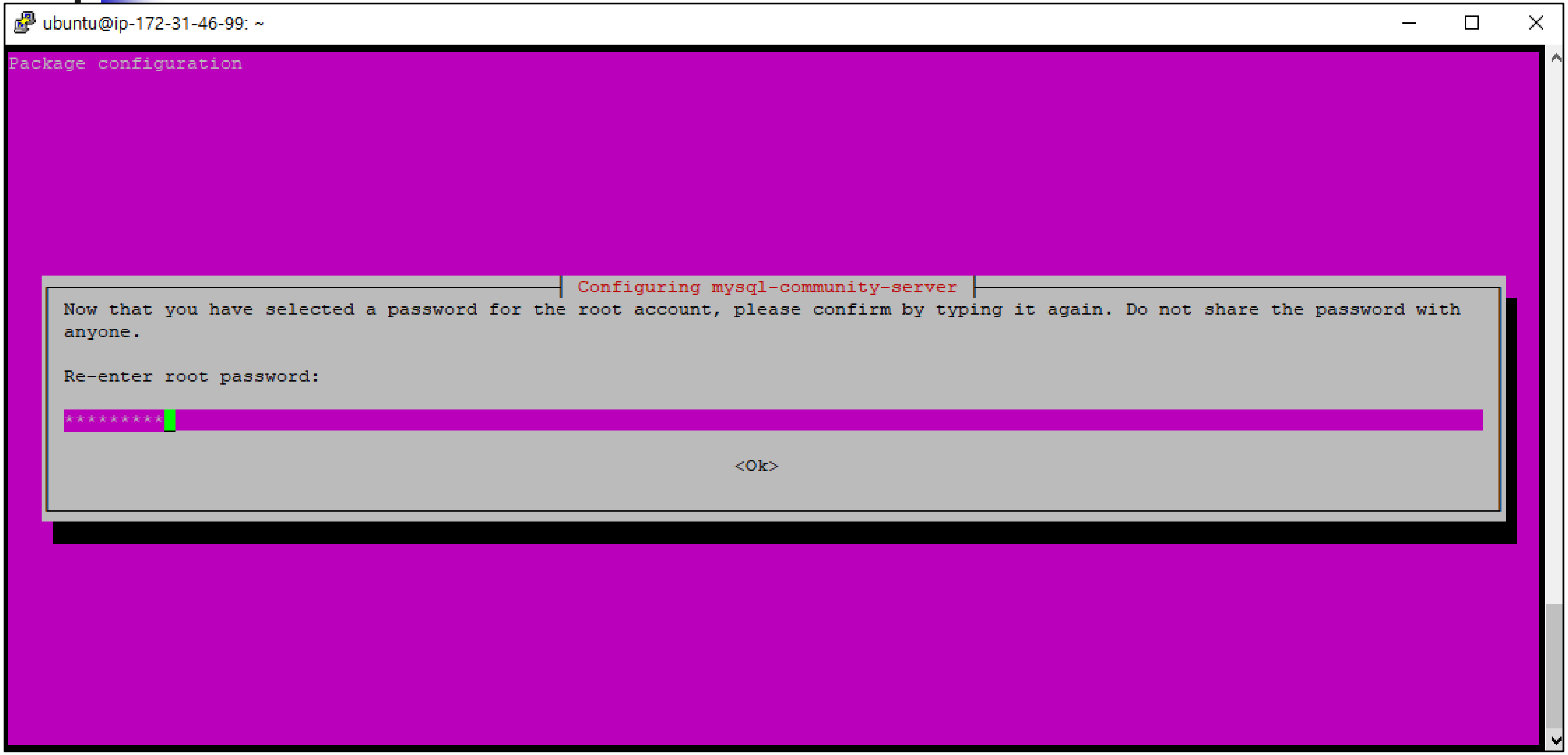
root 비밀번호 입력 : 1234Abcd!



```
Package configuration

                          ┤ Configuring mysql-community-server ├
 Please provide a strong password that will be set for the root account of your MySQL database. Leave it blank to enable password
 less login using UNIX socket based authentication.

 Enter root password:

 *********

                                           <Ok>
```

비밀번호 입력하고 엔터

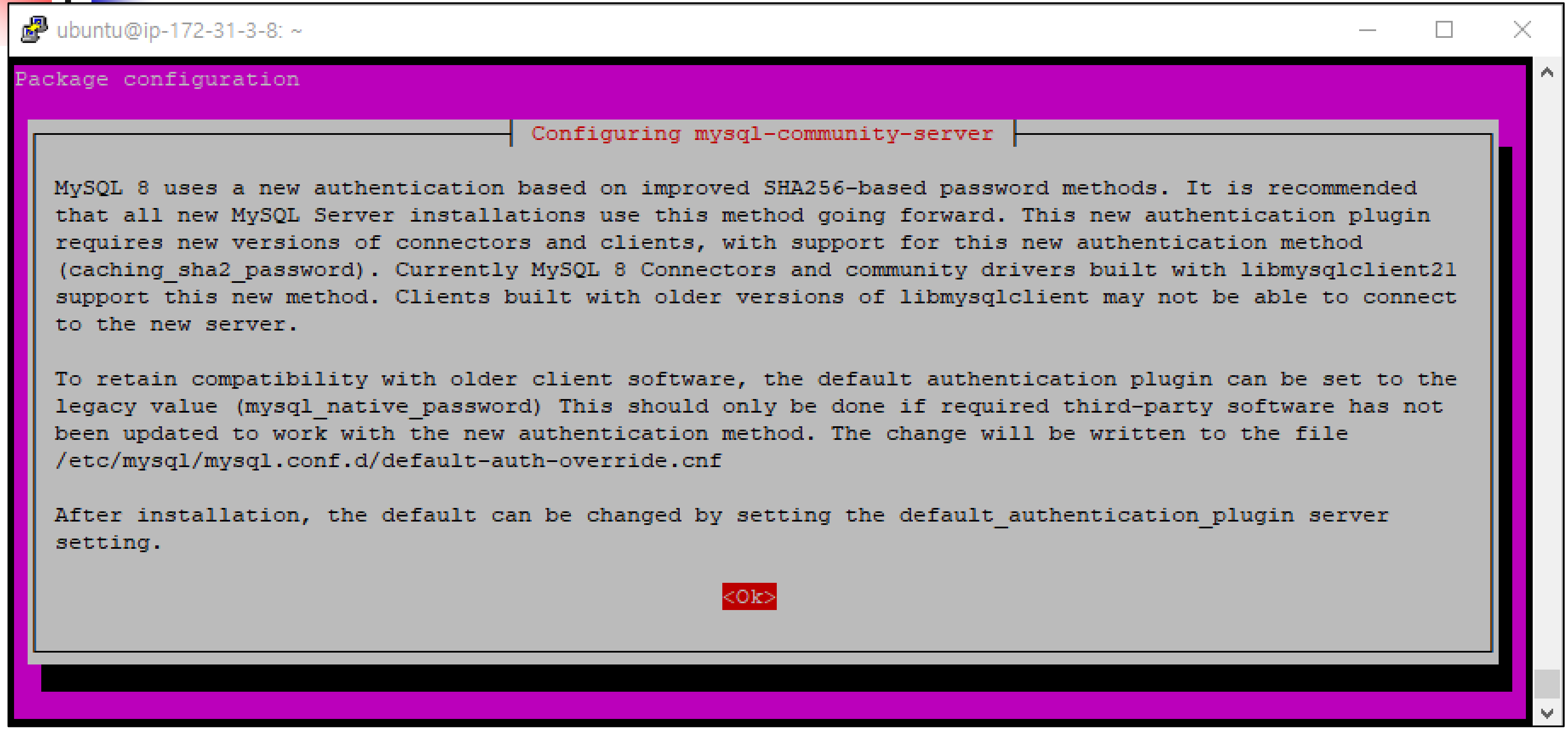
비밀번호 확인 입력 : 1234Abcd!



```
Package configuration

Configuring mysql-community-server

Now that you have selected a password for the root account, please confirm by typing it again. Do not share the password with
anyone.

Re-enter root password:

*********

                              <Ok>
```

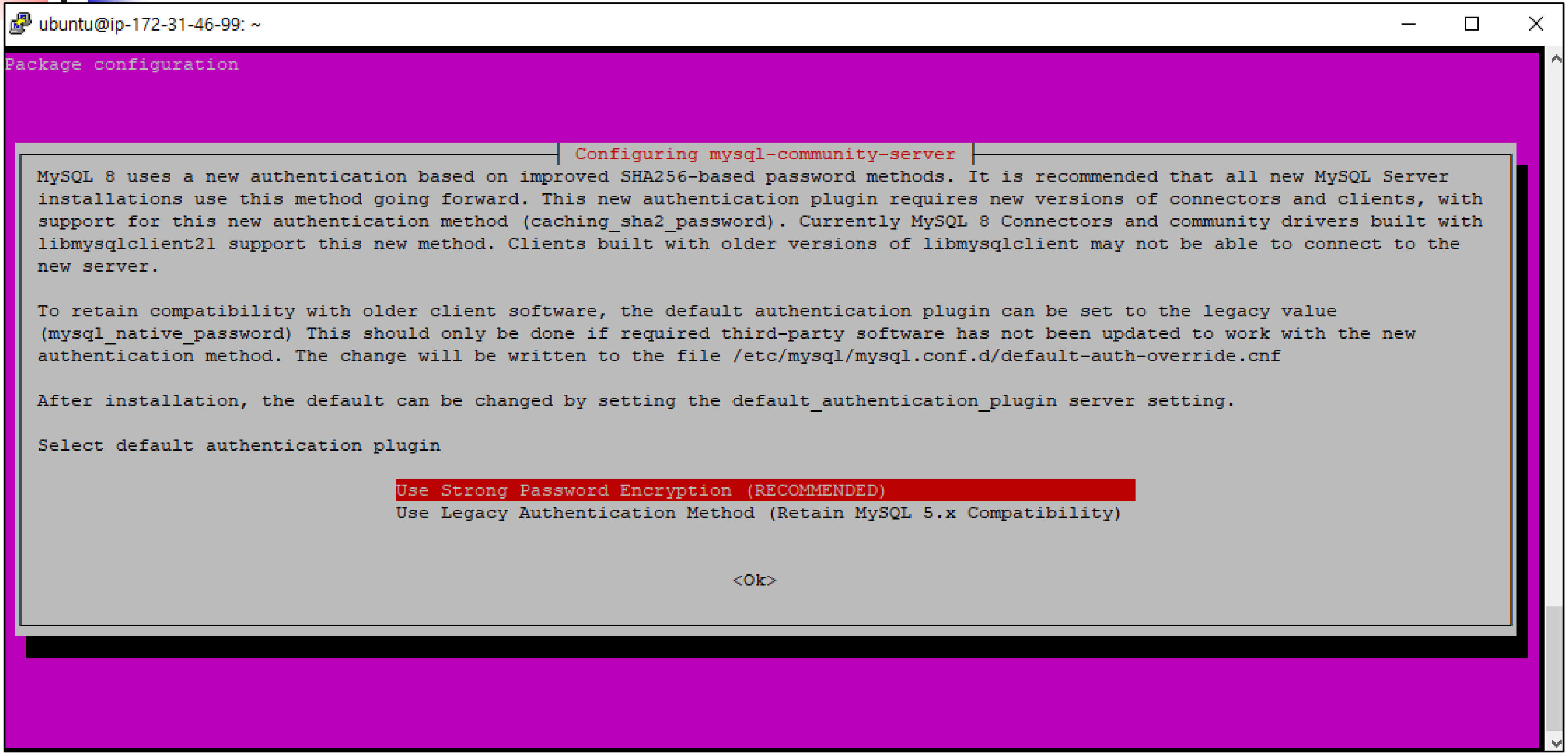
비밀번호 입력하고 엔터

엔터

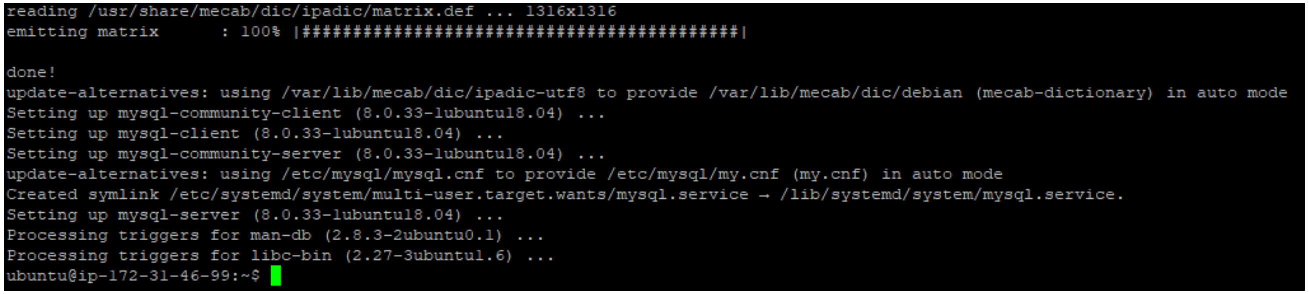


```
Package configuration

Configuring mysql-community-server

MySQL 8 uses a new authentication based on improved SHA256-based password methods. It is recommended
that all new MySQL Server installations use this method going forward. This new authentication plugin
requires new versions of connectors and clients, with support for this new authentication method
(caching_sha2_password). Currently MySQL 8 Connectors and community drivers built with libmysqlclient21
support this new method. Clients built with older versions of libmysqlclient may not be able to connect
to the new server.

To retain compatibility with older client software, the default authentication plugin can be set to the
legacy value (mysql_native_password) This should only be done if required third-party software has not
been updated to work with the new authentication method. The change will be written to the file
/etc/mysql/mysql.conf.d/default-auth-override.cnf

After installation, the default can be changed by setting the default_authentication_plugin server
setting.

<Ok>
```

엔터



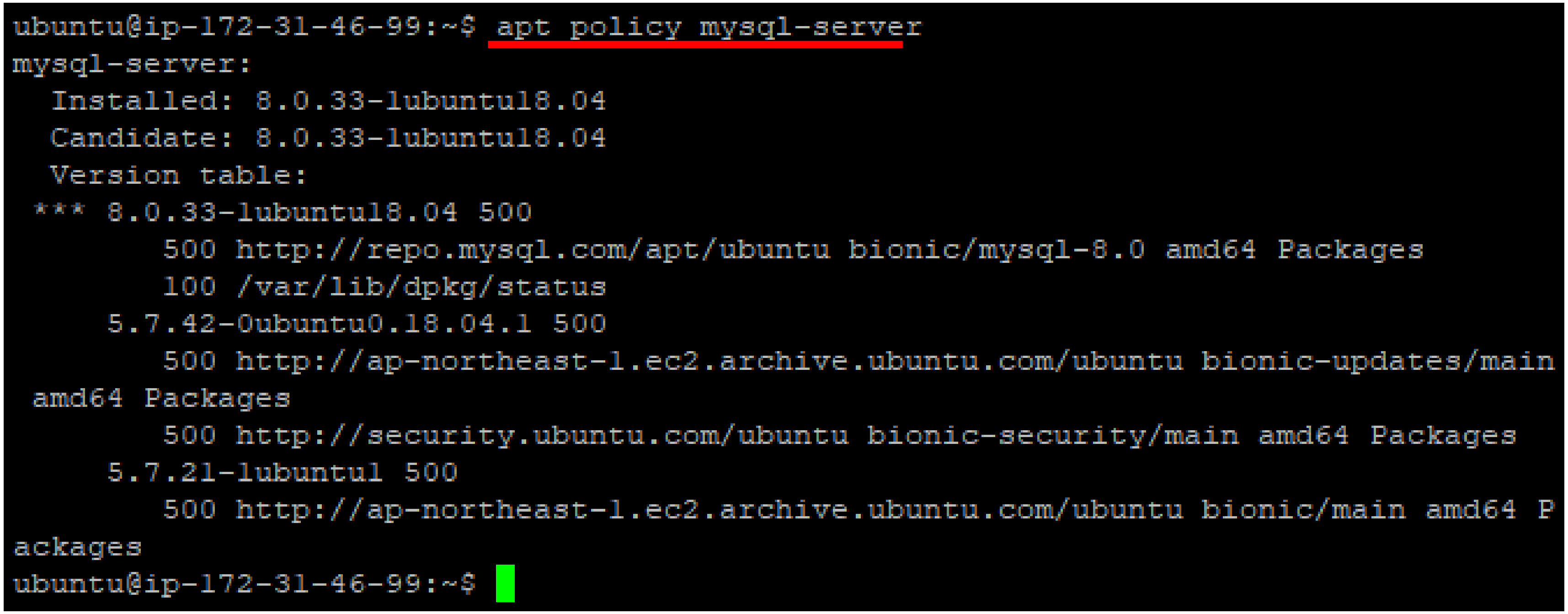
```
Package configuration

                    Configuring mysql-community-server
 MySQL 8 uses a new authentication based on improved SHA256-based password methods. It is recommended that all new MySQL Server
 installations use this method going forward. This new authentication plugin requires new versions of connectors and clients, with
 support for this new authentication method (caching_sha2_password). Currently MySQL 8 Connectors and community drivers built with
 libmysqlclient21 support this new method. Clients built with older versions of libmysqlclient may not be able to connect to the
 new server.

 To retain compatibility with older client software, the default authentication plugin can be set to the legacy value
 (mysql_native_password) This should only be done if required third-party software has not been updated to work with the new
 authentication method. The change will be written to the file /etc/mysql/mysql.conf.d/default-auth-override.cnf

 After installation, the default can be changed by setting the default_authentication_plugin server setting.

 Select default authentication plugin

                                Use Strong Password Encryption (RECOMMENDED)
                                Use Legacy Authentication Method (Retain MySQL 5.x Compatibility)


                                                   <Ok>
```



```
reading /usr/share/mecab/dic/ipadic/matrix.def ... 1316x1316
emitting matrix      : 100% |###########################################|

done!
update-alternatives: using /var/lib/mecab/dic/ipadic-utf8 to provide /var/lib/mecab/dic/debian (mecab-dictionary) in auto mode
Setting up mysql-community-client (8.0.33-1ubuntu18.04) ...
Setting up mysql-client (8.0.33-1ubuntu18.04) ...
Setting up mysql-community-server (8.0.33-1ubuntu18.04) ...
update-alternatives: using /etc/mysql/mysql.cnf to provide /etc/mysql/my.cnf (my.cnf) in auto mode
Created symlink /etc/systemd/system/multi-user.target.wants/mysql.service → /lib/systemd/system/mysql.service.
Setting up mysql-server (8.0.33-1ubuntu18.04) ...
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...
Processing triggers for libc-bin (2.27-3ubuntu1.6) ...
ubuntu@ip-172-31-46-99:~$
```

버전 확인

apt policy mysql-server

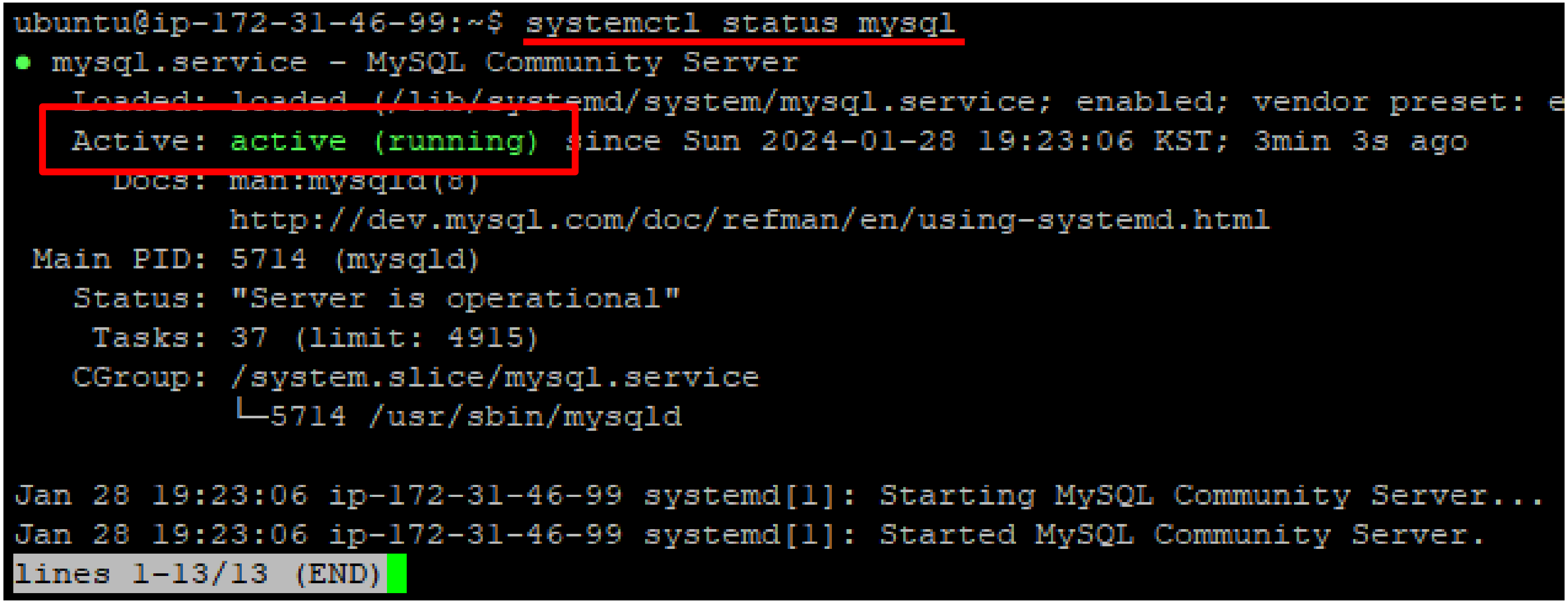


```
ubuntu@ip-172-31-46-99:~$ apt policy mysql-server
mysql-server:
  Installed: 8.0.33-1ubuntu18.04
  Candidate: 8.0.33-1ubuntu18.04
  Version table:
 *** 8.0.33-1ubuntu18.04 500
        500 http://repo.mysql.com/apt/ubuntu bionic/mysql-8.0 amd64 Packages
        100 /var/lib/dpkg/status
     5.7.42-0ubuntu0.18.04.1 500
        500 http://ap-northeast-1.ec2.archive.ubuntu.com/ubuntu bionic-updates/main
 amd64 Packages
        500 http://security.ubuntu.com/ubuntu bionic-security/main amd64 Packages
     5.7.21-1ubuntu1 500
        500 http://ap-northeast-1.ec2.archive.ubuntu.com/ubuntu bionic/main amd64 P
ackages
ubuntu@ip-172-31-46-99:~$ 
```

8.033 설치됨

systemctl status mysql

실행 확인 : active (running)

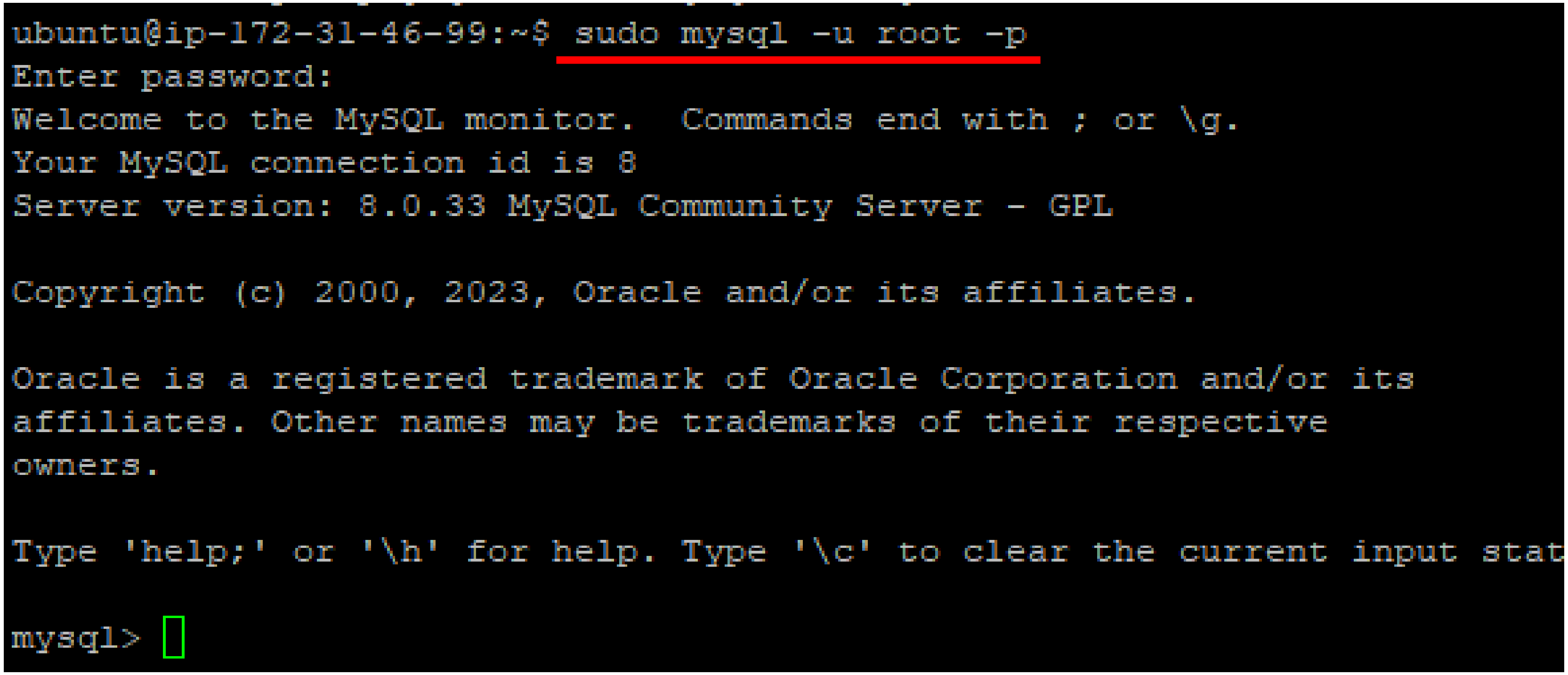


```
ubuntu@ip-172-31-46-99:~$ systemctl status mysql
● mysql.service - MySQL Community Server
   Loaded: loaded (/lib/systemd/system/mysql.service; enabled; vendor preset: e
   Active: active (running) since Sun 2024-01-28 19:23:06 KST; 3min 3s ago
     Docs: man:mysqld(8)
           http://dev.mysql.com/doc/refman/en/using-systemd.html
 Main PID: 5714 (mysqld)
   Status: "Server is operational"
    Tasks: 37 (limit: 4915)
   CGroup: /system.slice/mysql.service
           └─5714 /usr/sbin/mysqld

Jan 28 19:23:06 ip-172-31-46-99 systemd[1]: Starting MySQL Community Server...
Jan 28 19:23:06 ip-172-31-46-99 systemd[1]: Started MySQL Community Server.
lines 1-13/13 (END)
```

Ctrl + C

root 계정으로 접속

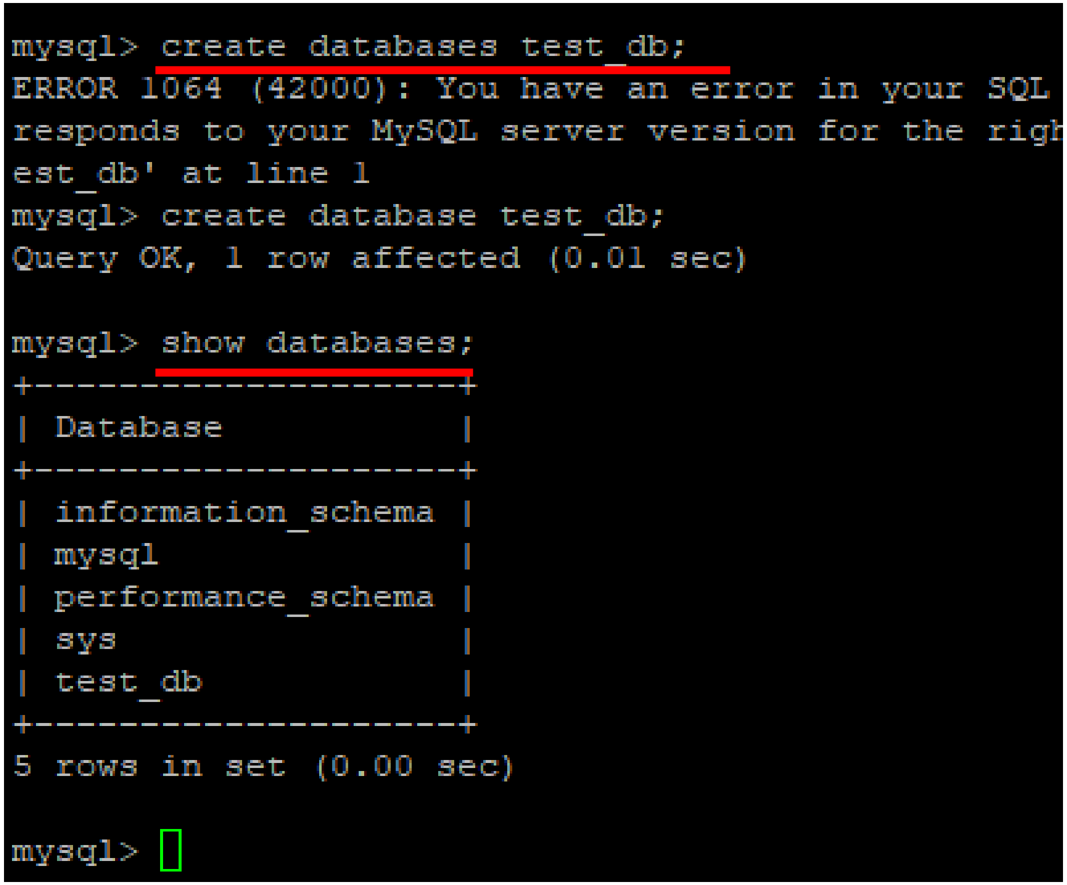
sudo mysql –u root –p
1234Abcd!



```
ubuntu@ip-172-31-46-99:~$ sudo mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 8.0.33 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 stat

mysql>
```

데이터베이스 생성

create database test_db;
show databases;



```
mysql> create databases test_db;
ERROR 1064 (42000): You have an error in your SQL
responds to your MySQL server version for the righ
est_db' at line 1
mysql> create database test_db;
Query OK, 1 row affected (0.01 sec)

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

mysql> 
```

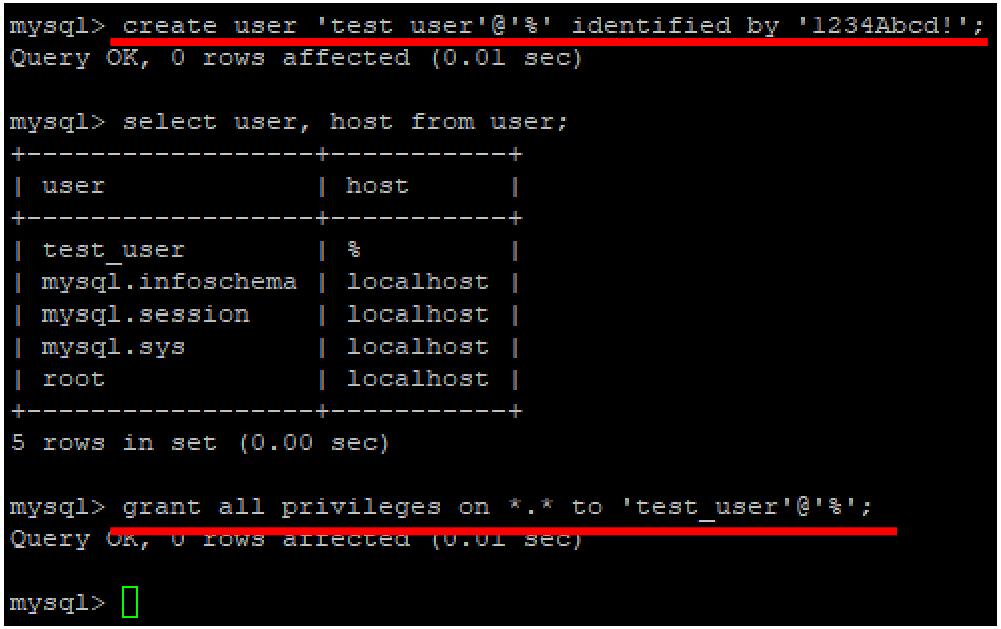
계정 생성 및 권한 부여

계정 생성

create user 'test_user'@'%' identified by '1234Abcd!’;

모든 데이터베이스이 모든 테이블에 대한 권한 부여

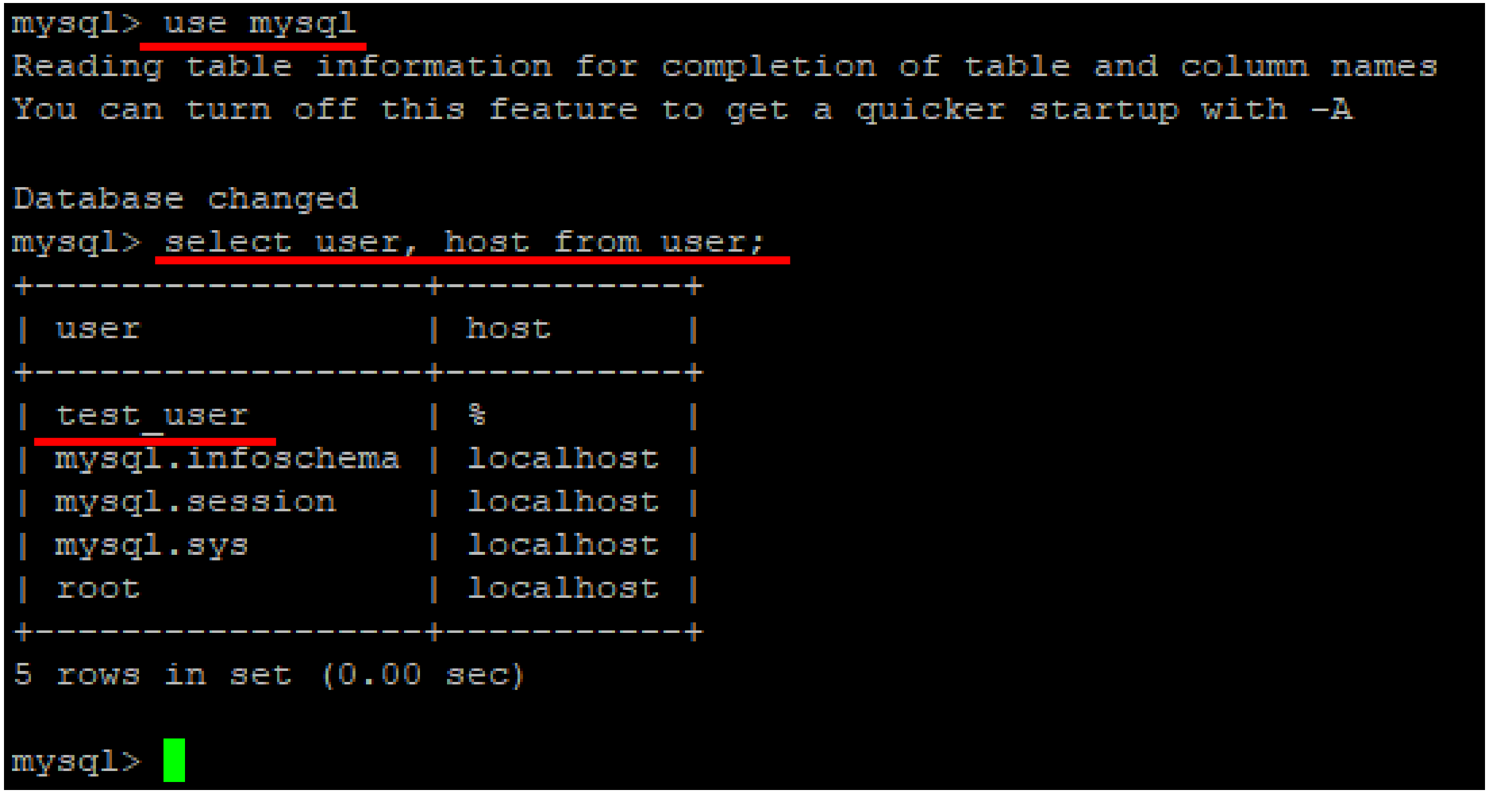
grant all privileges on *.* to 'test_user'@'%';



```
mysql> create user 'test_user'@'%' identified by '1234Abcd!';
Query OK, 0 rows affected (0.01 sec)

mysql> select user, host from user;
+------------------+-----------+
| user             | host      |
+------------------+-----------+
| test_user        | %         |
| mysql.infoschema | localhost |
| mysql.session    | localhost |
| mysql.sys        | localhost |
| root             | localhost |
+------------------+-----------+
5 rows in set (0.00 sec)

mysql> grant all privileges on *.* to 'test_user'@'%';
Query OK, 0 rows affected (0.01 sec)

mysql> 
```

생성된 계정 확인

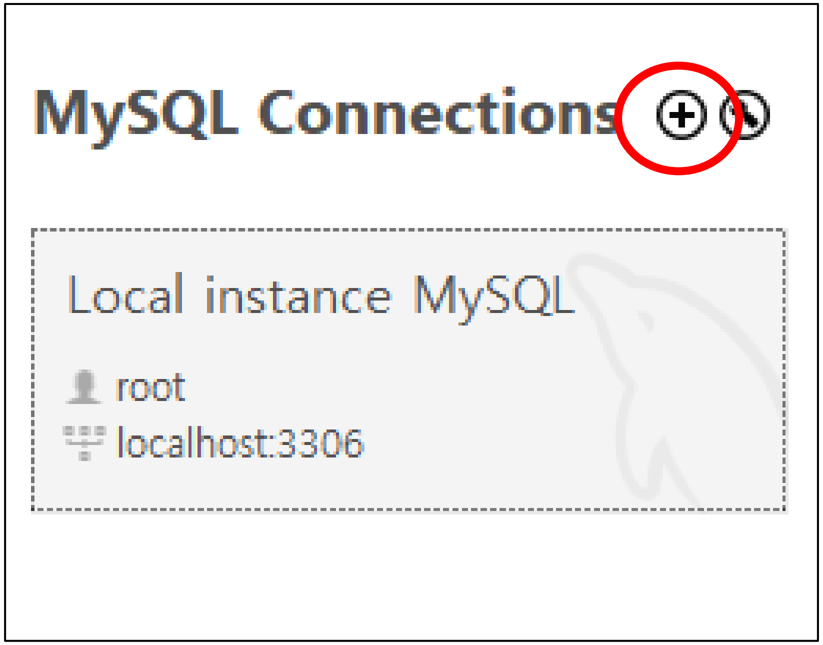
use mysql;
select user, host from user;



```
mysql> use mysql
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
mysql> select user, host from user;
+------------------+-----------+
| user             | host      |
+------------------+-----------+
| test_user        | %         |
| mysql.infoschema | localhost |
| mysql.session    | localhost |
| mysql.sys        | localhost |
| root             | localhost |
+------------------+-----------+
5 rows in set (0.00 sec)

mysql>
```

내 컴퓨터의 MySQL Workbench에서 접속

새 접속 생성



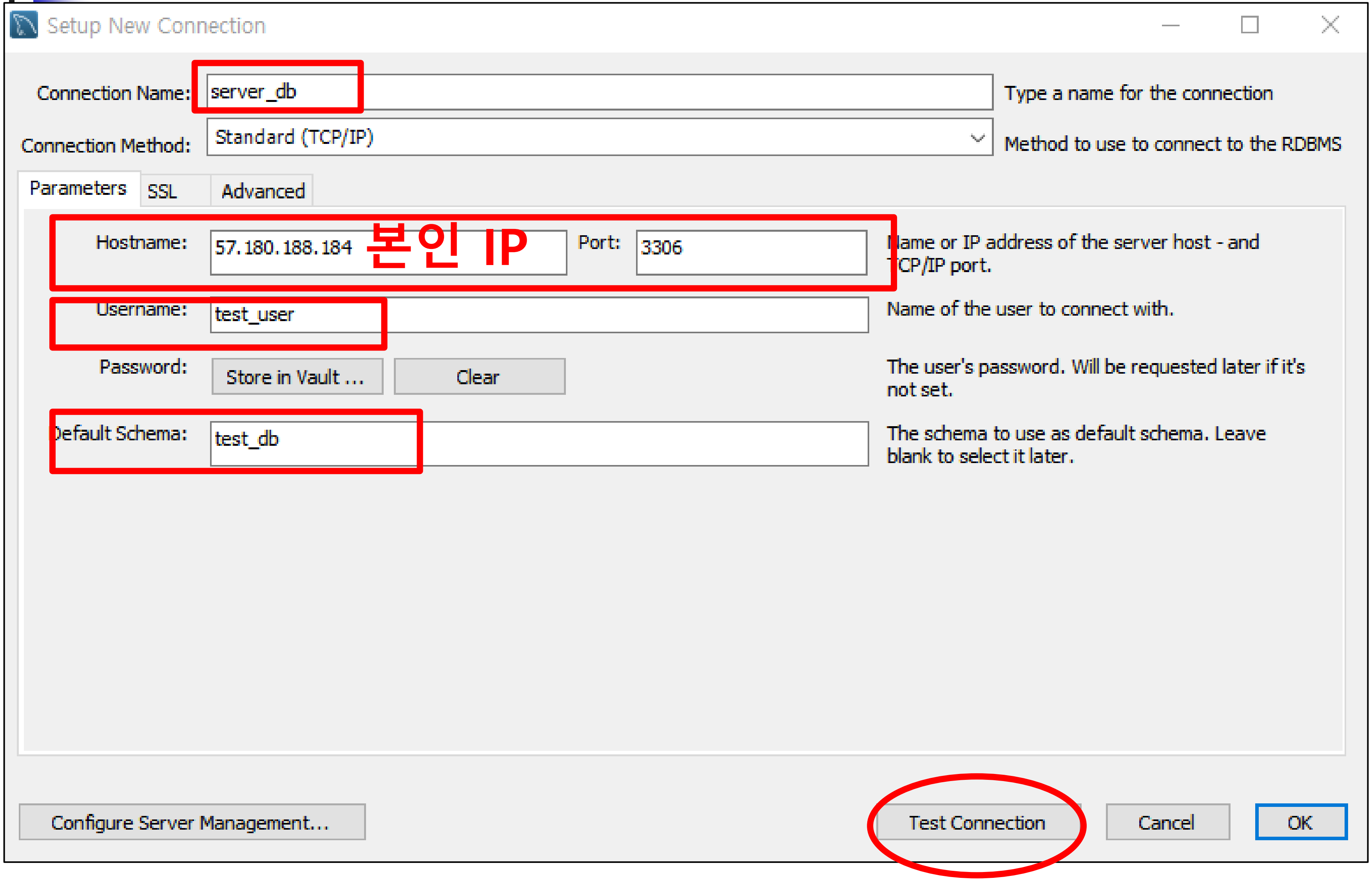
MySQL Connections ⊕

Local instance MySQL

root

localhost:3306

서버에 접속



Setup New Connection

Connection Name: server_db

Type a name for the connection

Connection Method: Standard (TCP/IP)

Method to use to connect to the RDBMS

Parameters | SSL | Advanced

Hostname: 57.180.188.184 Port: 3306

본인 IP

Name or IP address of the server host - and TCP/IP port.

Username: test_user

Name of the user to connect with.

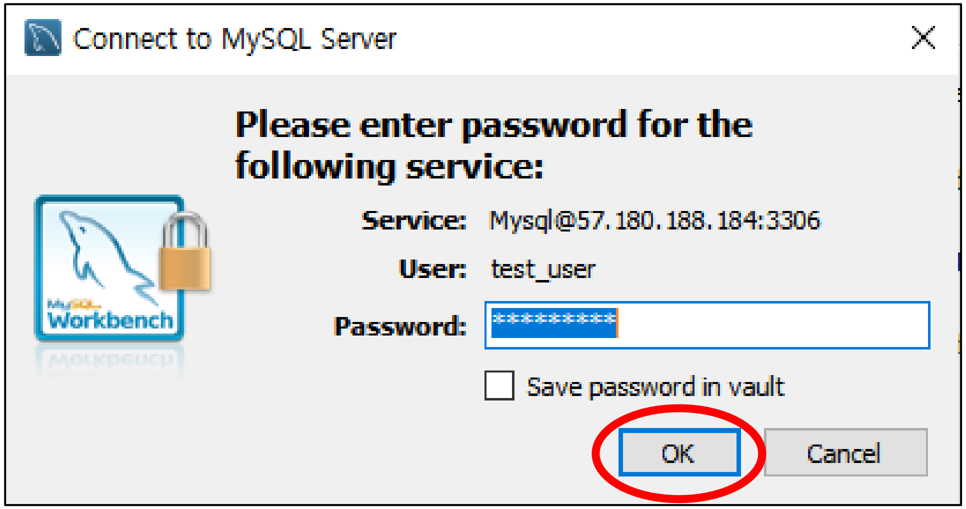
Password: Store in Vault ... Clear

The user's password. Will be requested later if it's not set.

Default Schema: test_db

The schema to use as default schema. Leave blank to select it later.

비밀번호 : 1234Abcd!



Connect to MySQL Server

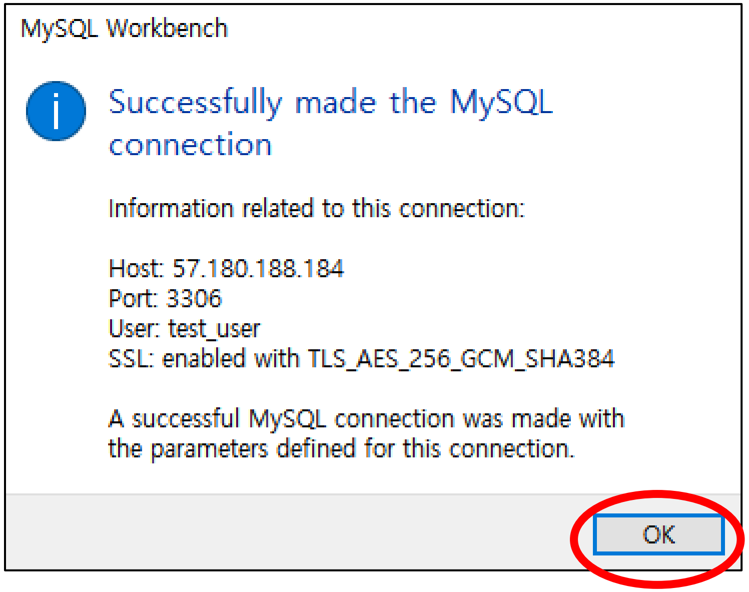
Please enter password for the following service:

Service: Mysql@57.180.188.184:3306

User: test_user

Password: **********

□ Save password in vault



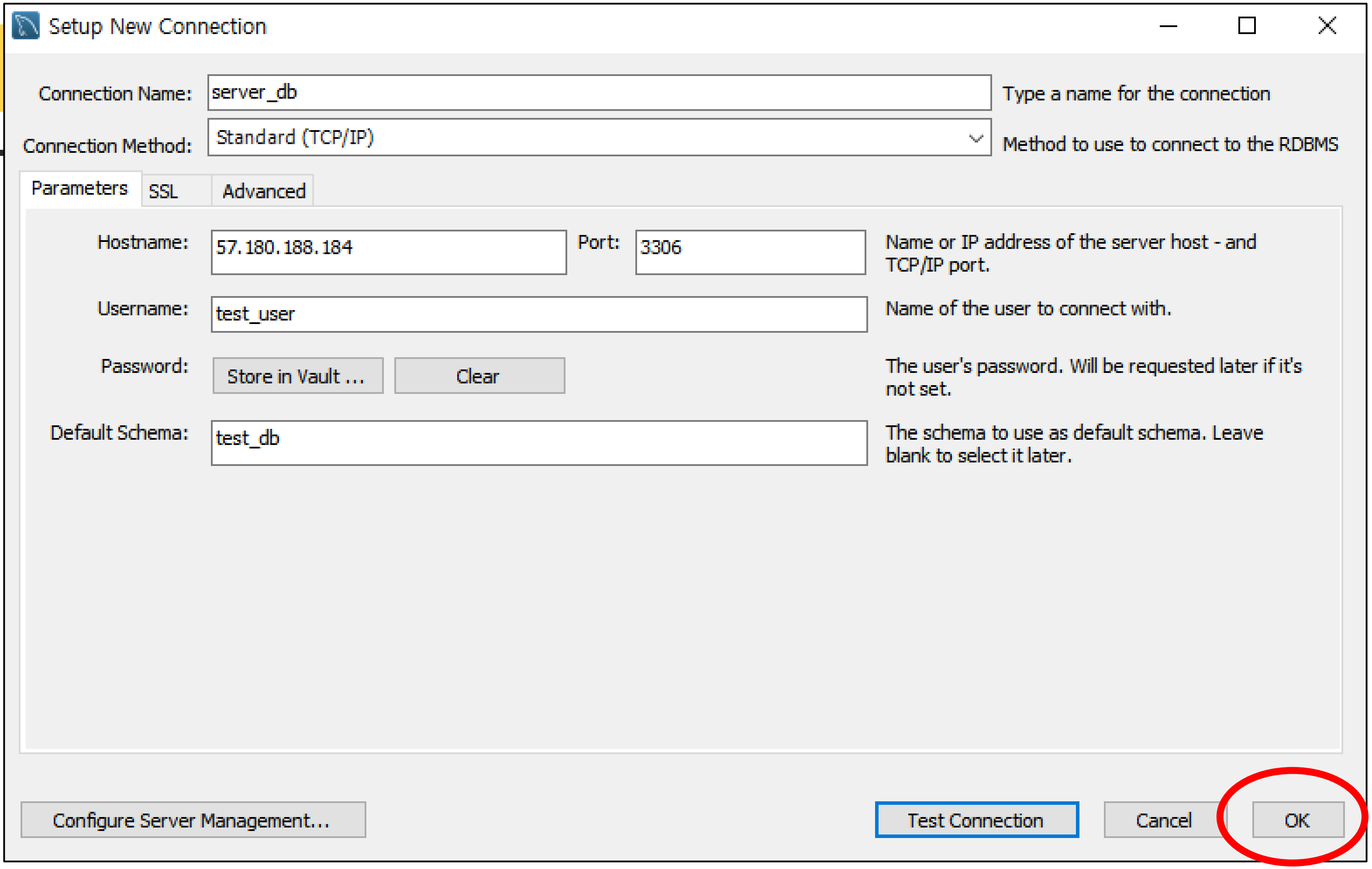
MySQL Workbench

Successfully made the MySQL connection

Information related to this connection:

Host: 57.180.188.184
Port: 3306
User: test_user
SSL: enabled with TLS_AES_256_GCM_SHA384

A successful MySQL connection was made with the parameters defined for this connection.



Setup New Connection

Connection Name: server_db — Type a name for the connection

Connection Method: Standard (TCP/IP) — Method to use to connect to the RDBMS

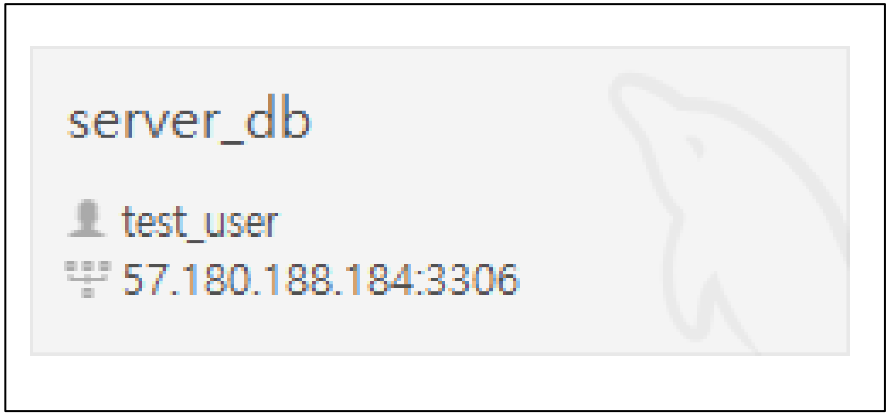
Hostname: 57.180.188.184 Port: 3306 — Name or IP address of the server host - and TCP/IP port.

Username: test_user — Name of the user to connect with.

Password: — The user's password. Will be requested later if it's not set.

Default Schema: test_db — The schema to use as default schema. Leave blank to select it later.

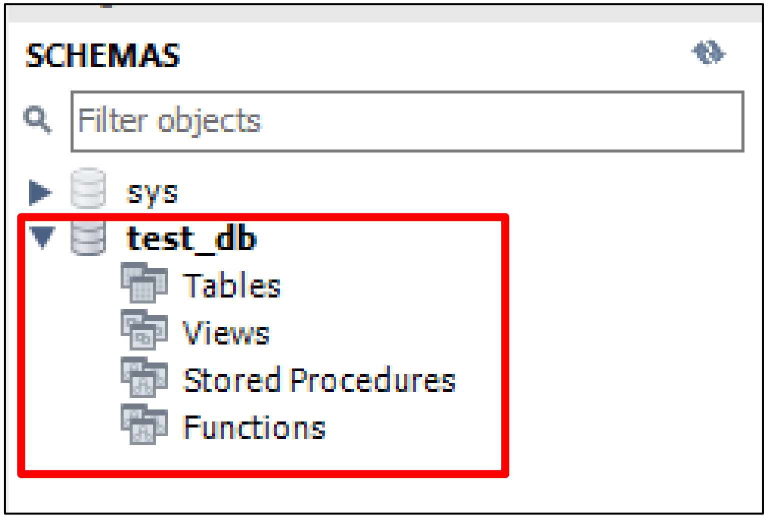
데이터베이스 확인



server_db

test_user

57.180.188.184:3306



SCHEMAS

sys

test_db

Tables

Views

Stored Procedures

Functions

로컬의 django_db import

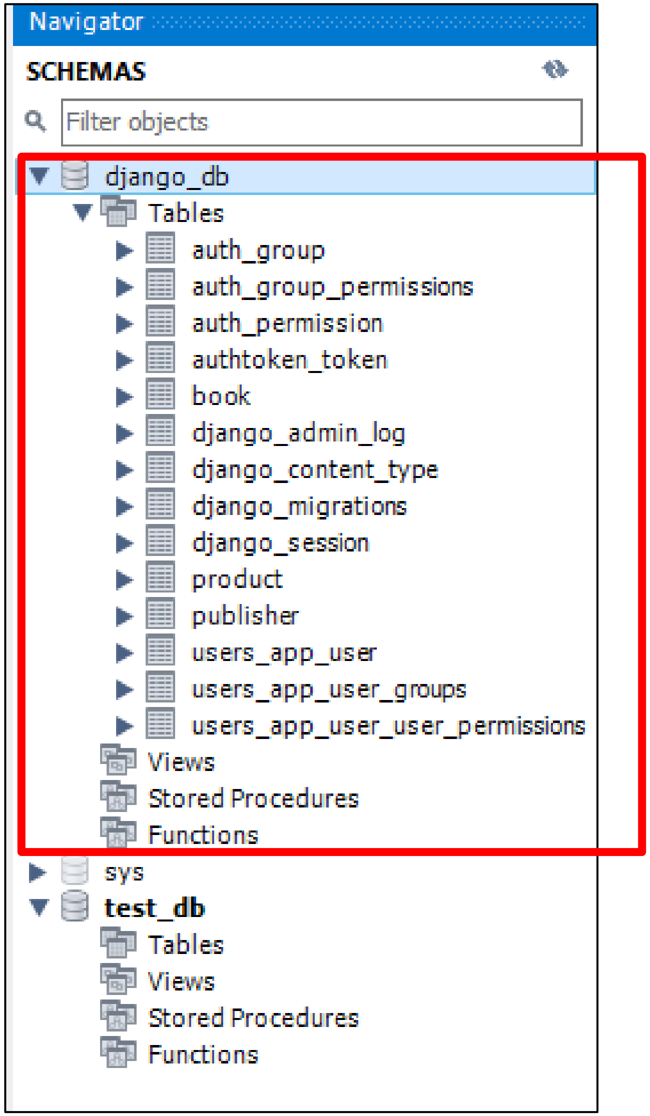
Export 한 django_db.sql 파일로 import

Server / Data Export

dango_db.sql 파일 열고 인코딩 수정

- 메모장에서
- utf8mb4_0900_ai_c 찾아서
- utf8mb4_general_ci 로 변경

Server / Data Import



Navigator

SCHEMAS

Filter objects

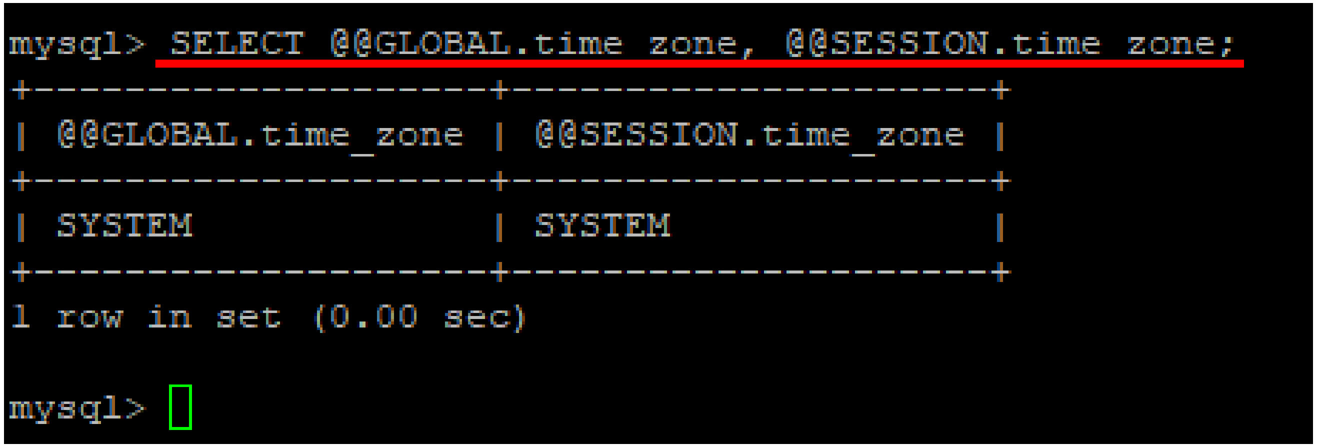
- django_db
  - Tables
    - auth_group
    - auth_group_permissions
    - auth_permission
    - authtoken_token
    - book
    - django_admin_log
    - django_content_type
    - django_migrations
    - django_session
    - product
    - publisher
    - users_app_user
    - users_app_user_groups
    - users_app_user_user_permissions
  - Views
  - Stored Procedures
  - Functions
- sys
- test_db
  - Tables
  - Views
  - Stored Procedures
  - Functions

새로고침하고 확인

MySQL 타임존 설정

현재 설정된 타임존 확인

SELECT @@GLOBAL.time_zone, @@SESSION.time_zone;

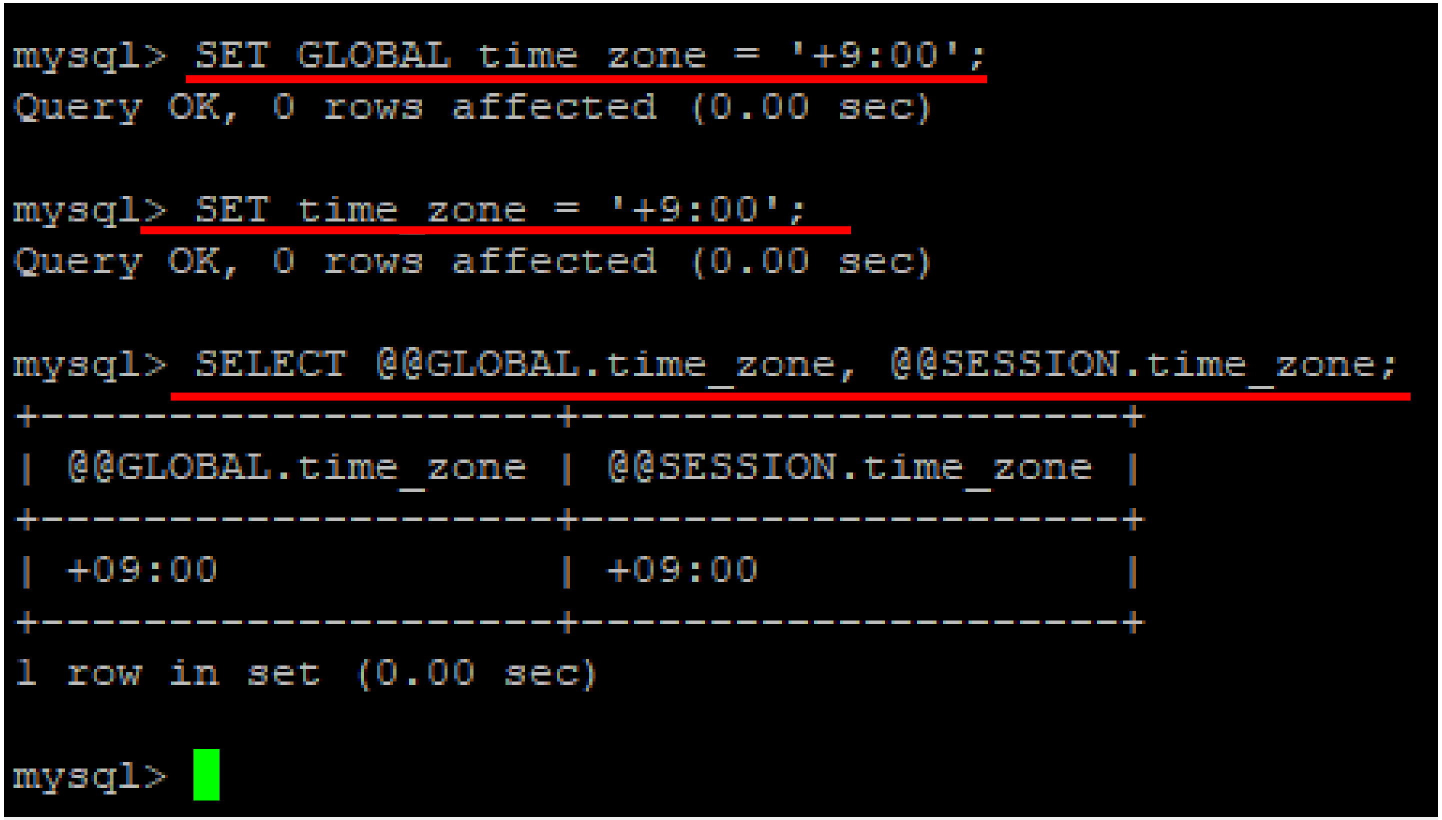


```
mysql> SELECT @@GLOBAL.time_zone, @@SESSION.time_zone;
+--------------------+---------------------+
| @@GLOBAL.time_zone | @@SESSION.time_zone |
+--------------------+---------------------+
| SYSTEM             | SYSTEM              |
+--------------------+---------------------+
1 row in set (0.00 sec)

mysql> 
```

MySQL 타임존 설정

SET GLOBAL time_zone = '+9:00’;
SET time_zone = '+9:00';

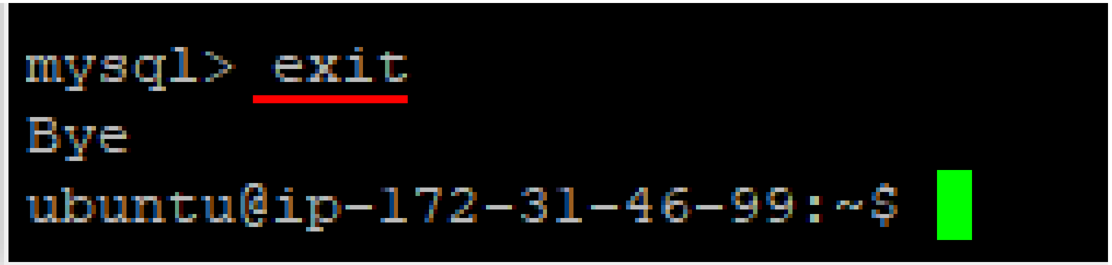
SELECT @@GLOBAL.time_zone, @@SESSION.time_zone;



```
mysql> SET GLOBAL time_zone = '+9:00';
Query OK, 0 rows affected (0.00 sec)

mysql> SET time_zone = '+9:00';
Query OK, 0 rows affected (0.00 sec)

mysql> SELECT @@GLOBAL.time_zone, @@SESSION.time_zone;
+--------------------+---------------------+
| @@GLOBAL.time_zone | @@SESSION.time_zone |
+--------------------+---------------------+
| +09:00             | +09:00              |
+--------------------+---------------------+
1 row in set (0.00 sec)

mysql> 
```

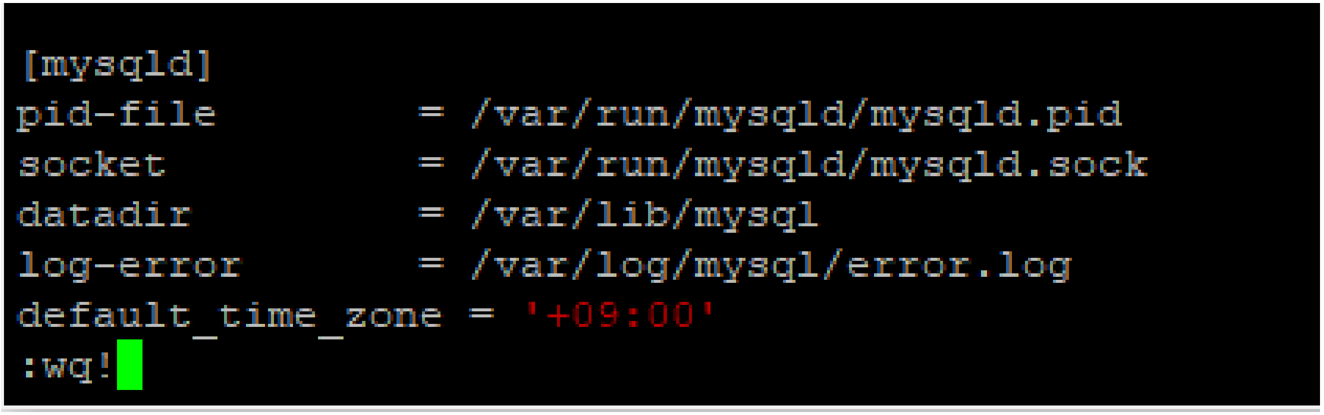
my.cnf 설정 파일에 디폴트 타임존 설정

exit



```
mysql> exit
Bye
ubuntu@ip-172-31-46-99:~$
```

sudo vi /etc/mysql/mysql.conf.d/mysqld.cnf
맨 뒤로 이동해서 i 누르고 입력 후
ESC
:wq!



```
[mysqld]
pid-file        = /var/run/mysqld/mysqld.pid
socket          = /var/run/mysqld/mysqld.sock
datadir         = /var/lib/mysql
log-error       = /var/log/mysql/error.log
default_time_zone = '+09:00'
:wq!
```

vi : visual editor
vim : improved vi

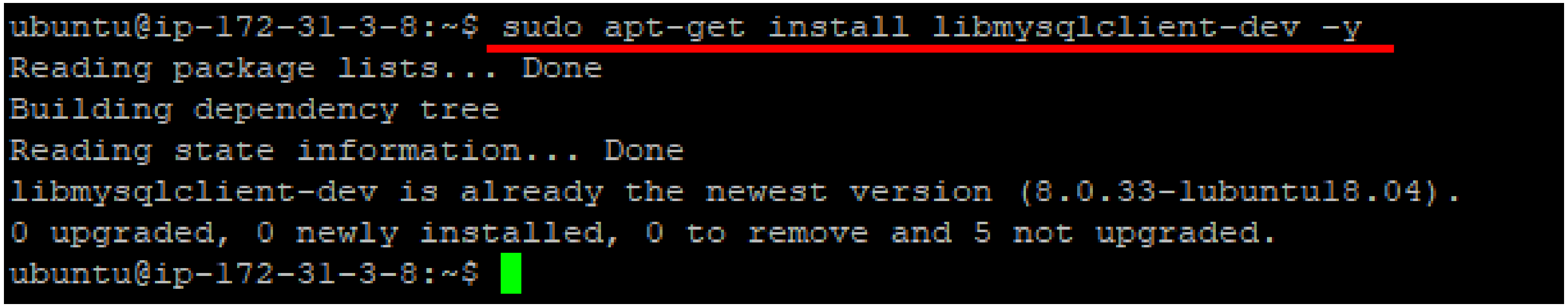
MySQL 서버 재시작

```
sudo systemctl restart mysql
exit
```

libmysqlclient-dev 설치

- 파이썬에서 mysqlclient 모듈 사용하기 위해
- libmysqlclient-dev 설치

sudo apt-get install libmysqlclient-dev -y

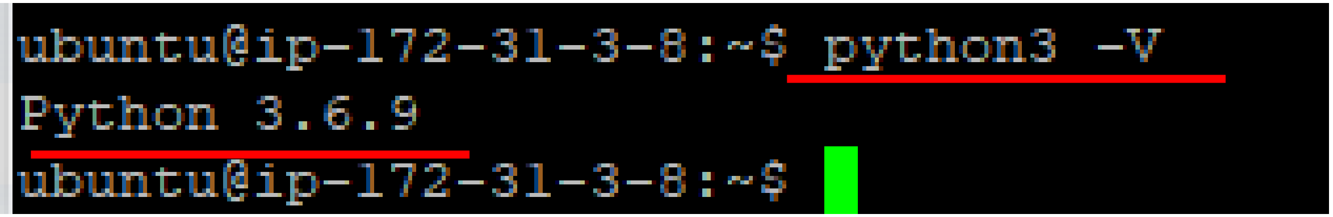


```
ubuntu@ip-172-31-3-8:~$ sudo apt-get install libmysqlclient-dev -y
Reading package lists... Done
Building dependency tree
Reading state information... Done
libmysqlclient-dev is already the newest version (8.0.33-1ubuntu18.04).
0 upgraded, 0 newly installed, 0 to remove and 5 not upgraded.
ubuntu@ip-172-31-3-8:~$
```

python3 버전 변경

현재 python3 버전 확인

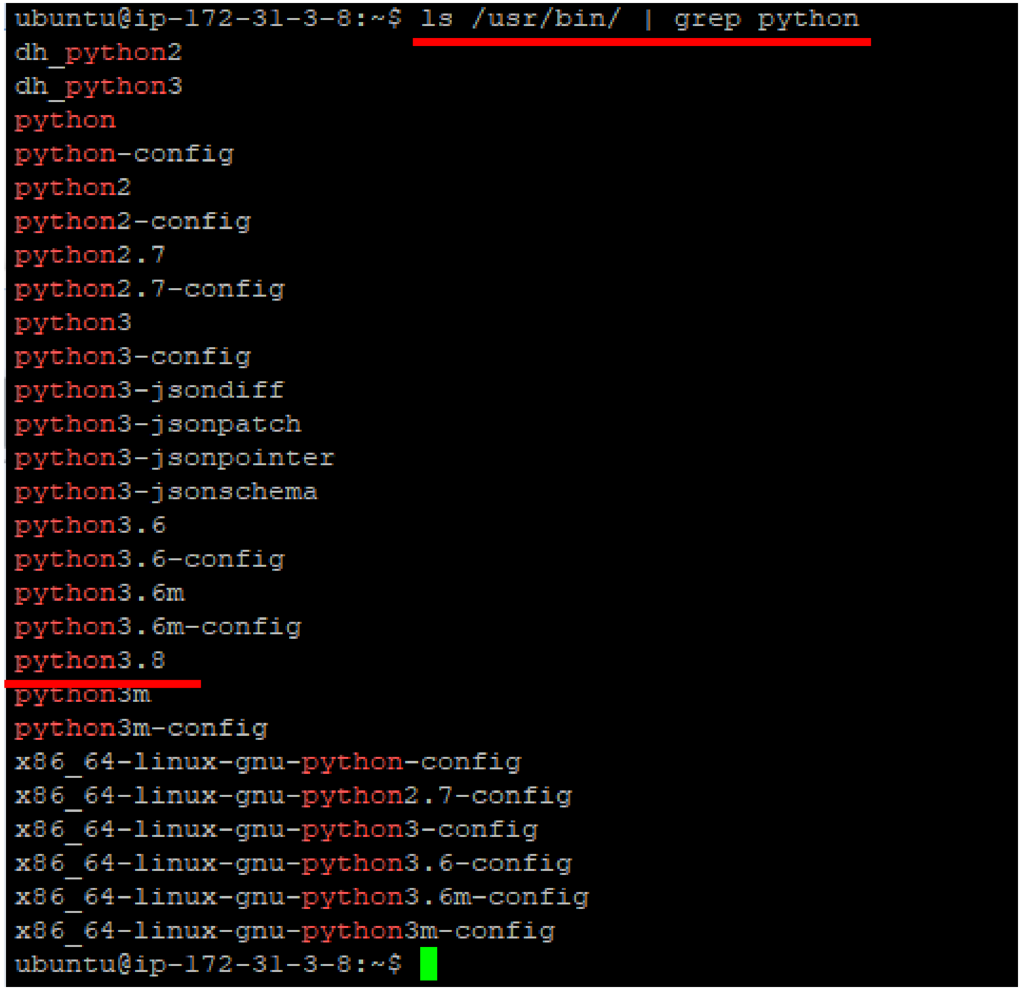
python3 -V



```
ubuntu@ip-172-31-3-8:~$ python3 -V
Python 3.6.9
ubuntu@ip-172-31-3-8:~$ 
```

python 다른 버전 확인

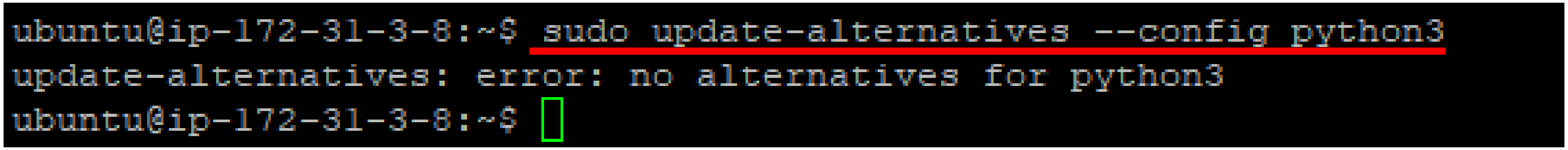
ls /usr/bin/ | grep python



```
ubuntu@ip-172-31-3-8:~$ ls /usr/bin/ | grep python
dh_python2
dh_python3
python
python-config
python2
python2-config
python2.7
python2.7-config
python3
python3-config
python3-jsondiff
python3-jsonpatch
python3-jsonpointer
python3-jsonschema
python3.6
python3.6-config
python3.6m
python3.6m-config
python3.8
python3m
python3m-config
x86_64-linux-gnu-python-config
x86_64-linux-gnu-python2.7-config
x86_64-linux-gnu-python3-config
x86_64-linux-gnu-python3.6-config
x86_64-linux-gnu-python3.6m-config
x86_64-linux-gnu-python3m-config
ubuntu@ip-172-31-3-8:~$
```

alternatives 확인

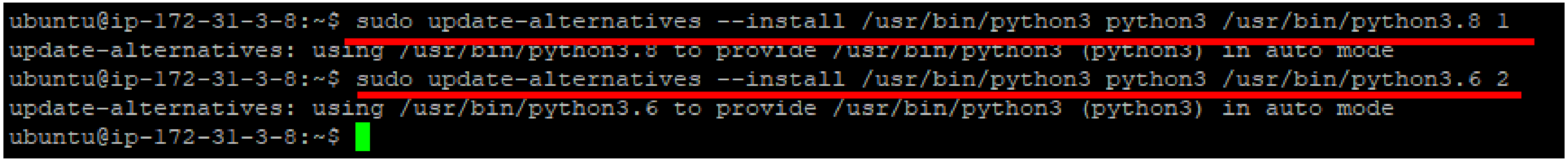
sudo update-alternatives --config python3



```
ubuntu@ip-172-31-3-8:~$ sudo update-alternatives --config python3
update-alternatives: error: no alternatives for python3
ubuntu@ip-172-31-3-8:~$ 
```

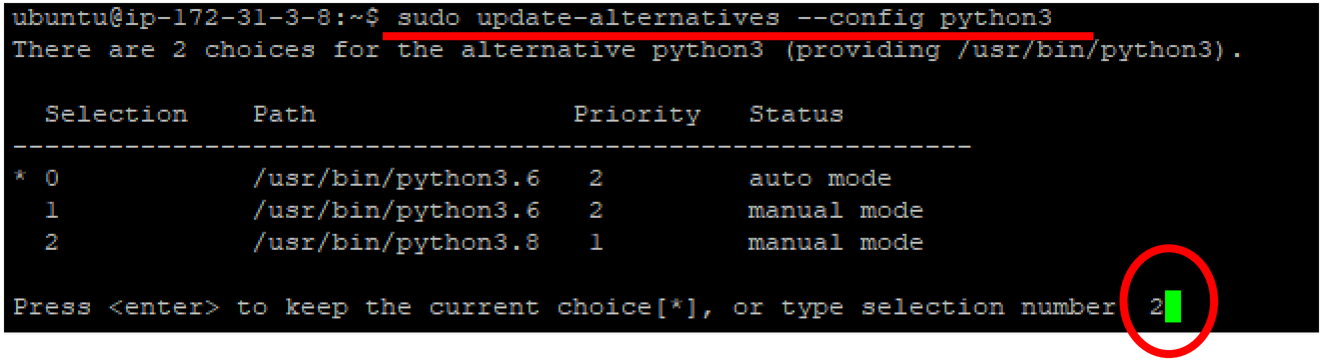
python3.8을 우선순위 1로
python 3.6을 우선순위 2로 등록

sudo update-alternatives --install /usr/bin/python3 python3 /usr/bin/python3.8 1
sudo update-alternatives --install /usr/bin/python3 python3 /usr/bin/python3.6 2



```
ubuntu@ip-172-31-3-8:~$ sudo update-alternatives --install /usr/bin/python3 python3 /usr/bin/python3.8 1
update-alternatives: using /usr/bin/python3.8 to provide /usr/bin/python3 (python3) in auto mode
ubuntu@ip-172-31-3-8:~$ sudo update-alternatives --install /usr/bin/python3 python3 /usr/bin/python3.6 2
update-alternatives: using /usr/bin/python3.6 to provide /usr/bin/python3 (python3) in auto mode
ubuntu@ip-172-31-3-8:~$
```

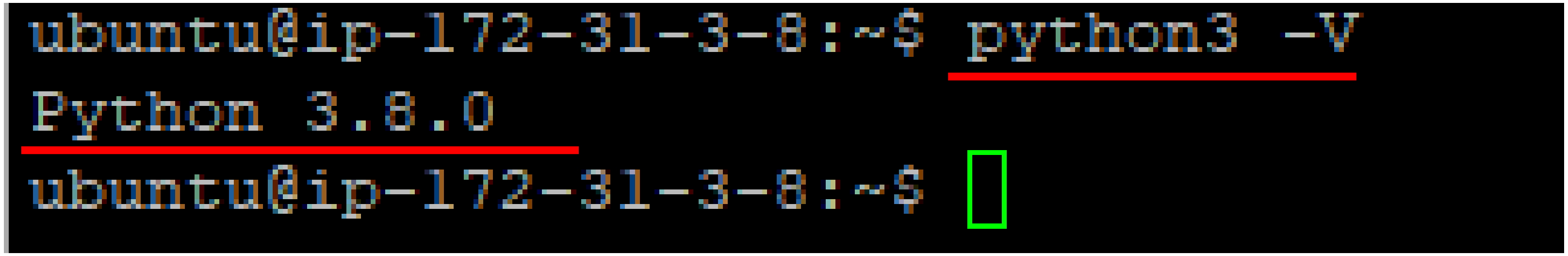
python3.8 선택
sudo update-alternatives --config python
2 선택



```
ubuntu@ip-172-31-3-8:~$ sudo update-alternatives --config python3
There are 2 choices for the alternative python3 (providing /usr/bin/python3).

  Selection    Path                Priority   Status
------------------------------------------------------------
* 0            /usr/bin/python3.6   2         auto mode
  1            /usr/bin/python3.6   2         manual mode
  2            /usr/bin/python3.8   1         manual mode

Press <enter> to keep the current choice[*], or type selection number: 2
```

파이썬 버전 확인

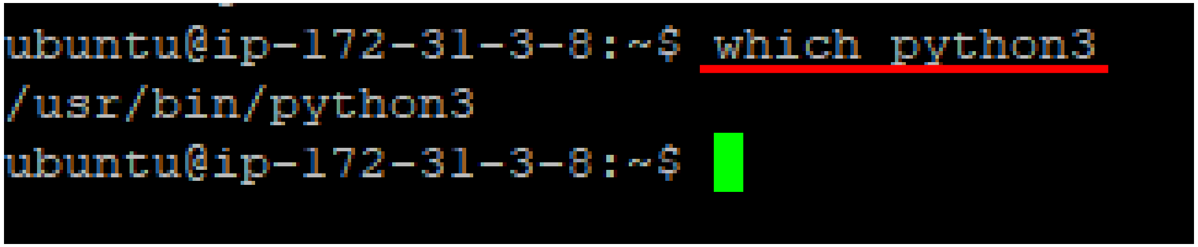
python3 -V



```
ubuntu@ip-172-31-3-8:~$ python3 -V
Python 3.8.0
ubuntu@ip-172-31-3-8:~$ 
```

환경변수 설정

which python3



```
ubuntu@ip-172-31-3-8:~$ which python3
/usr/bin/python3
ubuntu@ip-172-31-3-8:~$
```

vi ~/.bashrc

맨 아래로 이동해서 한 줄 띄고 i 누르고 입력

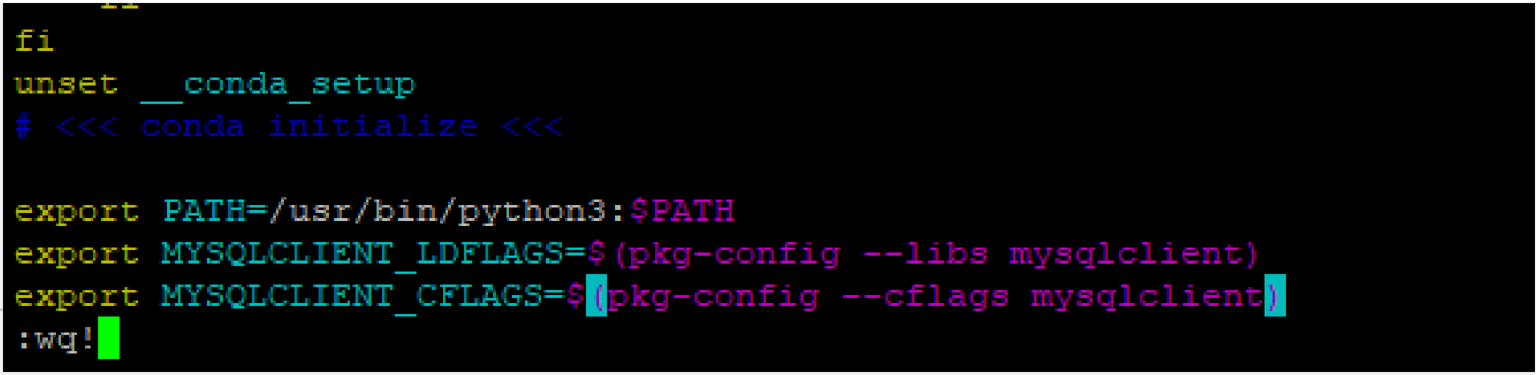
export PATH=/usr/bin/python3:$PATH

export MYSQLCLIENT_LDFLAGS=$(pkg-config --libs mysqlclient)

export MYSQLCLIENT_CFLAGS=$(pkg-config --cflags mysqlclient)

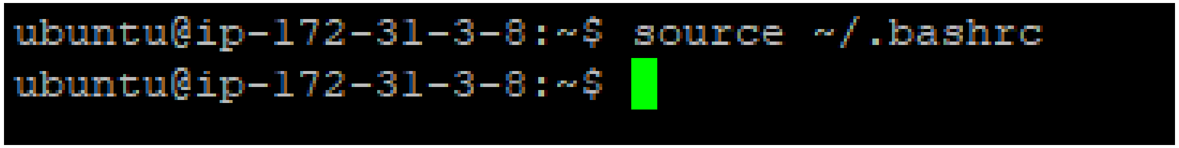
ESC

:wq!



```
fi
unset __conda_setup
# <<< conda initialize <<<

export PATH=/usr/bin/python3:$PATH
export MYSQLCLIENT_LDFLAGS=$(pkg-config --libs mysqlclient)
export MYSQLCLIENT_CFLAGS=$(pkg-config --cflags mysqlclient)
:wq!
```

source ~/.bashrc (변경 사항 반영)



```
ubuntu@ip-172-31-3-8:~$ source ~/.bashrc
ubuntu@ip-172-31-3-8:~$ 
```

venv 가상환경 생성하기 위한 패키지 설치

```
sudo apt-get install python3.8-venv

sudo apt-get install python3.8-dev
```

ubuntu 계정 종료

```
exit
```

4. 서버에 업로드

- Putty로 접속하고
- pwd로 현재 위치 확인하고 : /home/lab01
- 현재 위치에서 깃 복제
  - git clone https://github.com/xxxx(본인 주소)
- 디렉토리명 변경 : 본인 이니셜 붙임
  - ls로 django_react 디렉토리 확인
  - mv django_react django_react_hkd
  - cd django_react_hkd 하고 내용 확인

db_settings.py 파일 생성

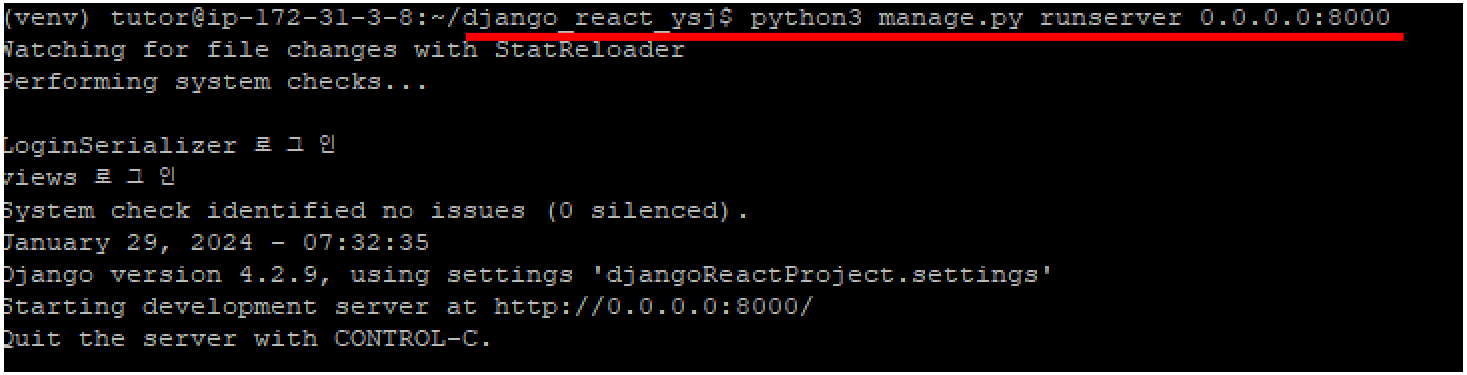
- django_react_hkd에 db_settings.py 파일 생성
- db_settings.py에서 USER와 PASSWORD 변경
  - “USER” : “test_db”
  - “PASSWORD” : “1234Abcd!”
- vi db_settings.py 해서 파일 열고 내용 복사/붙여넣기

가상환경 생성 / 필수 패키지 설치

- django_react_hkd 디렉토리 안에서
  - 가상환경 생성
    - python3 –m venv venv
    - 활성화 : source venv/bin/activate
  - 필수 패키지 설치
    - pip3 install --upgrade pip
    - pip3 install --upgrade setuptools
    - pip3 install -r requirements.txt
    - pip3 install cryptography

실행 확인

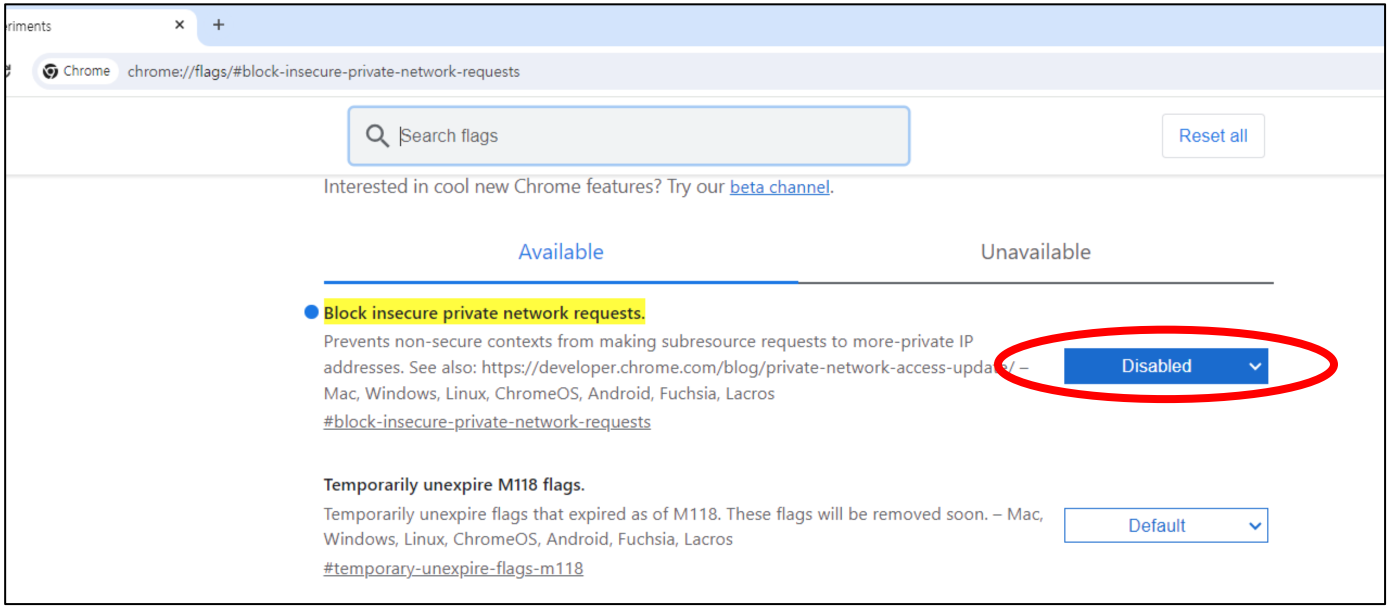
- django 서버 실행
  - python3 manage.py runserver 0.0.0.0:8000



```
(venv) tutor@ip-172-31-3-8:~/django_react_ysj$ python3 manage.py runserver 0.0.0.0:8000
Watching for file changes with StatReloader
Performing system checks...

LoginSerializer 로 그 인
views 로 그 인
System check identified no issues (0 silenced).
January 29, 2024 - 07:32:35
Django version 4.2.9, using settings 'djangoReactProject.settings'
Starting development server at http://0.0.0.0:8000/
Quit the server with CONTROL-C.
```

CORS 오류 처리

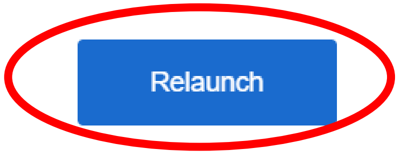
chrome://flags/#block-insecure-private-network-requests



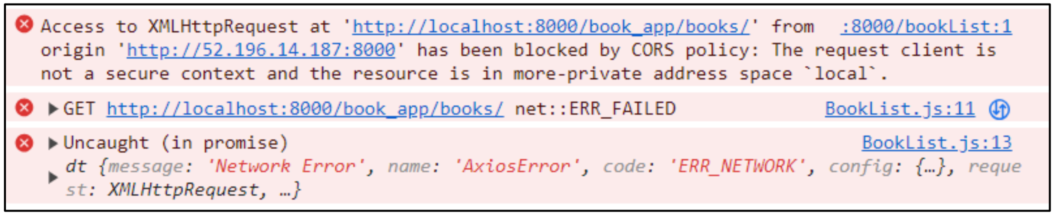
Interested in cool new Chrome features? Try our beta channel.

Block insecure private network requests.
Prevents non-secure contexts from making subresource requests to more-private IP addresses. See also: https://developer.chrome.com/blog/private-network-access-update/ – Mac, Windows, Linux, ChromeOS, Android, Fuchsia, Lacros
#block-insecure-private-network-requests
Disabled

Temporarily unexpire M118 flags.
Temporarily unexpire flags that expired as of M118. These flags will be removed soon. – Mac, Windows, Linux, ChromeOS, Android, Fuchsia, Lacros
#temporary-unexpire-flags-m118
Default



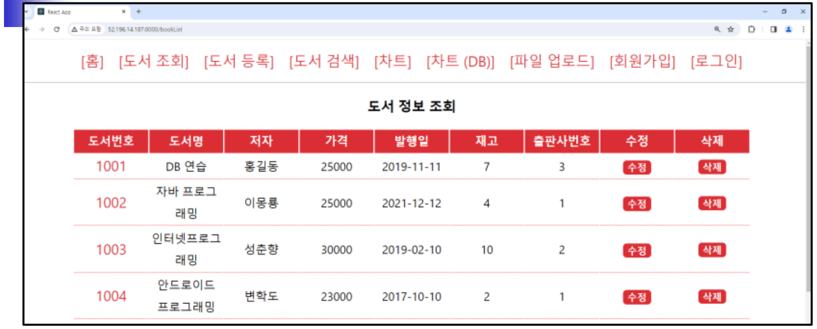
Disabled 설정하지 않으면 오류 발생



```
Access to XMLHttpRequest at 'http://localhost:8000/book_app/books/' from   :8000/bookList:1
origin 'http://52.196.14.187:8000' has been blocked by CORS policy: The request client is
not a secure context and the resource is in more-private address space `local`.
▶GET http://localhost:8000/book_app/books/ net::ERR_FAILED              BookList.js:11
▶Uncaught (in promise)                                                      BookList.js:13
 dt {message: 'Network Error', name: 'AxiosError', code: 'ERR_NETWORK', config: {…}, reque
 st: XMLHttpRequest, …}
```

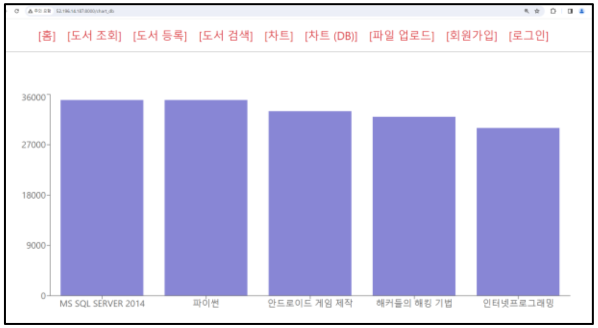
또는
https 전환하면
오류 없이 실행

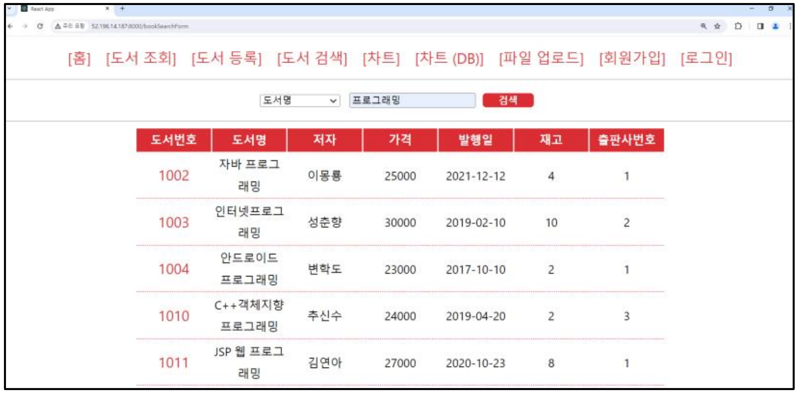
결과 확인


[홈] [도서 조회] [도서 등록] [도서 검색] [차트] [차트 (DB)] [파일 업로드] [회원가입] [로그인]

도서 정보 조회

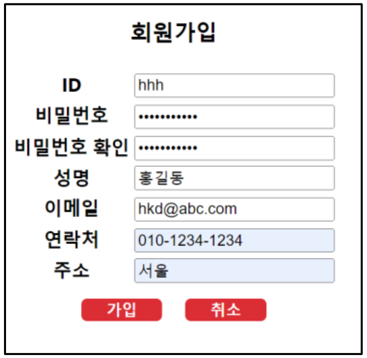
도서번호	도서명	저자	가격	발행일	재고	출판사번호	수정	삭제
1001	DB 연습	홍길동	25000	2019-11-11	7	3	수정	삭제
1002	자바 프로그래밍	이몽룡	25000	2021-12-12	4	1	수정	삭제
1003	인터넷프로그래밍	성춘향	30000	2019-02-10	10	2	수정	삭제
1004	안드로이드 프로그래밍	변학도	23000	2017-10-10	2	1	수정	삭제




[홈] [도서 조회] [도서 등록] [도서 검색] [차트] [차트 (DB)] [파일 업로드] [회원가입] [로그인]
도서명
검색

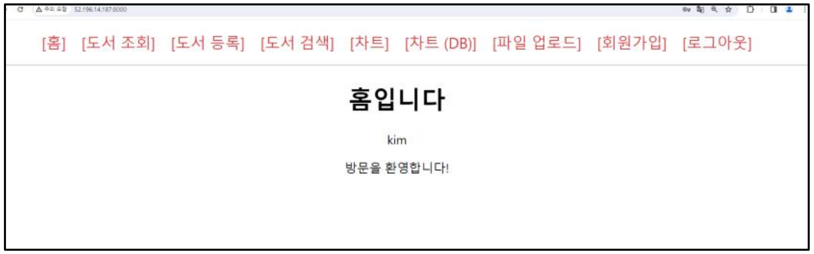
프로그래밍

도서번호	도서명	저자	가격	발행일	재고	출판사번호
1002	자바 프로그래밍	이몽룡	25000	2021-12-12	4	1
1003	인터넷프로그래밍	성춘향	30000	2019-02-10	10	2
1004	안드로이드 프로그래밍	변학도	23000	2017-10-10	2	1
1010	C++객체지향 프로그래밍	추신수	24000	2019-04-20	2	3
1011	JSP 웹 프로그래밍	김연아	27000	2020-10-23	8	1



회원가입

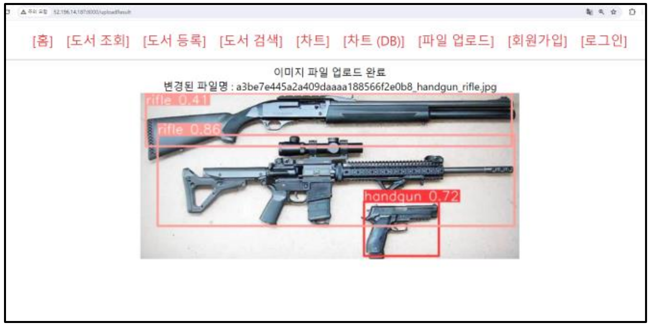
ID	hhh
비밀번호	•••••••••••
비밀번호 확인	•••••••••••
성명	홍길동
이메일	hkd@abc.com
연락처	010-1234-1234
주소	서울


[홈] [도서 조회] [도서 등록] [도서 검색] [차트] [차트 (DB)] [파일 업로드] [회원가입] [로그아웃]

홈입니다

kim

방문을 환영합니다!


[홈] [도서 조회] [도서 등록] [도서 검색] [차트] [차트 (DB)] [파일 업로드] [회원가입] [로그인]

이미지 파일 업로드 완료

변경된 파일명 : a3be7e445a2a409daaaa188566f2e0b8_handgun_rifle.jpg

rifle 0.41
rifle 0.86
handgun 0.72

모든 기능 다 잘 됨