

모의해킹 환경 구축 가이드

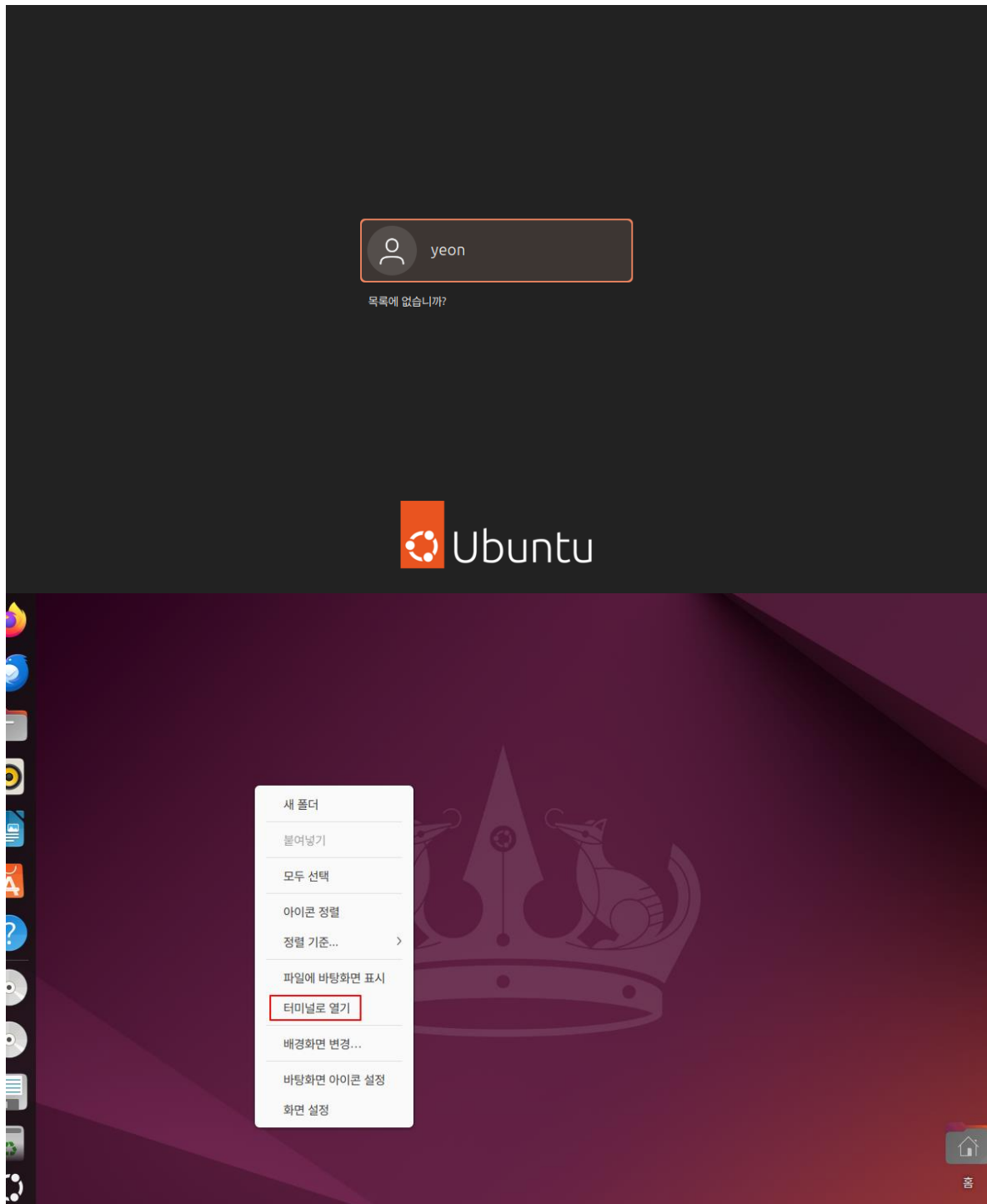
리눅스 환경 구축 가이드

Rookies

E119

1. Ubuntu 초기 사용 환경설정

1.1 SSH 접속



로컬 접속 후 터미널을 열어 진행한다.

```
yeon@yeon-VMware-Virtual-Platform:~$ sudo apt update && sudo apt upgrade
Warning: The unit file, source configuration file or drop-ins of apt-news.service changed on disk.
Run 'systemctl daemon-reload' to reload units.
Warning: The unit file, source configuration file or drop-ins of esm-cache.service changed on disk.
Run 'systemctl daemon-reload' to reload units.
받기:1 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
기존:2 http://kr.archive.ubuntu.com/ubuntu noble InRelease
받기:3 http://kr.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
받기:4 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packages [326 kB]
기존:5 http://kr.archive.ubuntu.com/ubuntu noble-backports InRelease
받기:6 http://security.ubuntu.com/ubuntu noble-security/main i386 Packages [148 kB]
받기:7 http://security.ubuntu.com/ubuntu noble-security/universe i386 Packages [102 kB]
받기:8 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Packages [266 kB]
받기:9 http://kr.archive.ubuntu.com/ubuntu noble/main Translation ko [227 kB]
받기:10 http://kr.archive.ubuntu.com/ubuntu noble/main Translation en [227 kB]

sudo apt update && sudo apt upgrade
```

apt update로 패키지 목록을 최신 상태로 만들고, apt upgrade로 새 버전이 있는 패키지를 업그레이드한다.

```
yeon@yeon-VMware-Virtual-Platform:~$ sudo apt install openssh-server
패키지 목록을 읽는 중입니다... 완료
의존성 트리를 만드는 중입니다... 완료
상태 정보를 읽는 중입니다... 완료
다음의 추가 패키지가 설치될 것입니다 :
  ncurses-term openssh-sftp-server ssh-import-id
제안하는 패키지 :
  molly-guard monkeysphere ssh-askpass
다음 새 패키지를 설치할 것입니다 :
  ncurses-term openssh-server openssh-sftp-server ssh-import-id
0개 업그레이드, 4개 새로 설치, 0개 제거 및 2개 업그레이드 안 함.
832 k바이트 약간의 공간을 받아야 합니다.
```

```
yeon@yeon-VMware-Virtual-Platform:~$ sudo systemctl start ssh
yeon@yeon-VMware-Virtual-Platform:~$ sudo systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/usr/lib/systemd/system/ssh.service; disabled; preset: enabled)
   Active: active (running) since Tue 2024-09-10 14:10:54 KST; 11s ago
 TriggeredBy: ● ssh.socket
     Docs: man:sshd(8)
           man:sshd_config(5)
   Process: 10140 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
    Main PID: 10141 (sshd)
       Tasks: 1 (limit: 4558)
      Memory: 1.2M (peak: 1.5M)
         CPU: 19ms
    CGroup: /system.slice/ssh.service
            └─10141 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

9월 10 14:10:54 yeon-VMware-Virtual-Platform systemd[1]: Starting ssh.service - OpenBSD Secure Shell server:
9월 10 14:10:54 yeon-VMware-Virtual-Platform sshd[10141]: Server listening on :: port 22.
9월 10 14:10:54 yeon-VMware-Virtual-Platform systemd[1]: Started ssh.service - OpenBSD Secure Shell server:
lines 1-17/17 (END)
```

```
sudo apt install openssh-server
```

```
sudo systemctl start ssh
```

```
sudo systemctl status ssh
```

편의성과 보안을 위해서 SSH를 통해 원격으로 서버에 접속할 것이다. 이를 위해 OpenSSH 서버를 설치하고 SSH 서비스를 시작한다. 서비스를 시작했으면 status 명령어로 상태를 확인한다.

```
yeon@yeon-VMware-Virtual-Platform:~/바탕 화면$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group def
1000
    link/ether 00:0c:29:a2:2a:aa brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet 192.168.80.132/24 brd 192.168.80.255 scope global dynamic noprefixroute ens33
        valid_lft 1563sec preferred_lft 1563sec
    inet6 fe80::20c:29ff:fea2:2aaa/64 scope link
        valid_lft forever preferred_lft forever
yeon@yeon-VMware-Virtual-Platform:~/바탕 화면$
```

```
ip a
```

리눅스의 ip를 확인한다.

```
r2com@DESKTOP-304U2MR MINGW64 ~
$ ssh yeon@192.168.80.132
yeon@192.168.80.132's password:
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-41-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

Applications를 위한 확장된 보안 유지보수 비활성화됨.

0개의 업데이트가 즉시 적용 가능합니다.

ESM Apps을 (를) 활성화해서 미래의 추가적인 업데이트를 받을 수 있습니다.
https://ubuntu.com/esm 을 참조하거나 sudo pro status 를 실행하십시오

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

yeon@yeon-VMware-Virtual-Platform:~$
ssh yeon@192.168.80.132 ('사용자 계정'@'사용자 ip')
```

power shell이나 xshell, git bash, putty 등의 터미널을 사용하여 ssh 접속을 해준다. 비밀번호는 사용자 계정의 비밀번호를 입력한다.

2. 웹 사이트 구축 및 실행

2.1 필수 패키지 설치

```
yeon@yeon-VMware-Virtual-Platform:~$ sudo apt install git docker.io unzip vim mysql-client-core-8.0 openjdk-11-jdk
[sudo] yeon 암호:
패키지 목록을 읽는 중입니다... 완료
의존성 트리를 만드는 중입니다... 완료
상태 정보를 읽는 중입니다... 완료
패키지 unzip는 이미 최신 버전입니다 (6.0-28ubuntu4).
unzip 패키지는 수동설치로 지정합니다.
다음의 추가 패키지가 설치될 것입니다:
  bridge-utils ca-certificates-java containerd docker.io fonts-dejavu-extra git-man java-common libatk-wrapper-java
  libatk-wrapper-java-glib libatk-wrapper-java-jni liberror-perl libice-dev libpthread-stubs0-dev libsm-dev libx11-dev libxau-dev
  libxcb1-dev libxdmcp-dev libxt-dev openjdk-11-jdk-headless openjdk-11-jre openjdk-11-jre-headless pigz runc
  ubuntu-fan vim-runtime x11proto-dev xorg-sgml-doctools xtrans-dev
제안하는 패키지:
  ifupdown aufs-tools btrfs-progs cgroupfs-mount | cgroup-lite debootstrap docker-buildx docker-compose-v2
  docker-doc rinse zfs-fuse | zfsutils git-daemon-run | git-daemon-sysvinit git-doc git-email git-gui gitk gitweb
  git-cvs git-mediawiki git-svn default-jre libice-doc libsm-doc libx11-doc libxcb-doc libxt-doc openjdk-11-demo
  openjdk-11-source visualvm fonts-ipafont-mincho fonts-wqy-microhei | fonts-wqy-zenhei
  fonts-indic ctags vim-doc vim-scripts
다음 새 패키지를 설치할 것입니다:
  bridge-utils ca-certificates-java containerd docker.io fonts-dejavu-extra git git-man java-common
  libatk-wrapper-java libatk-wrapper-java-glib liberror-perl libice-dev libpthread-stubs0-dev libsm-dev libx11-dev
  libxau-dev libxcb1-dev libxdmcp-dev libxt-dev mysql-client-core-8.0 openjdk-11-jdk openjdk-11-jdk-headless
  openjdk-11-jre openjdk-11-jre-headless pigz runc ubuntu-fan vim vim-runtime x11proto-dev xorg-sgml-doctools
  xtrans-dev
0개 업그레이드, 32개 새로 설치, 0개 제거 및 2개 업그레이드 안 함.
215 M바이트 아카이브를 받아야 합니다.
이 작업 후 690 M바이트의 디스크 공간을 더 사용하게 됩니다.
계속 하시겠습니까? [Y/n] y
sudo apt install git docker.io unzip vim mysql-client-core-8.0 openjdk-11-jdk
```

git: 프로젝트 코드를 클론하기 위해 설치

docker.io: Docker를 설치하여 MySQL 컨테이너를 실행하기 위해 사용

unzip: Gradle 압축 파일을 풀기 위해 필요

vim: 텍스트 편집기

mysql-client-core-8.0: MySQL 클라이언트 설치

openjdk-11-jdk: Java 개발 키트 설치, Spring Boot 프로젝트를 실행하기 위해 필요

2.2 Git으로 프로젝트 클론

```
yeon@yeon-VMware-Virtual-Platform:~$ git clone https://github.com/hw20200500/green_airline_project
'green_airline_project'에 복제합니다...
remote: Enumerating objects: 14662, done.
remote: Counting objects: 100% (1283/1283), done.
remote: Compressing objects: 100% (651/651), done.
remote: Total 14662 (delta 535), reused 1158 (delta 484), pack-reused 13379 (from 1)
오브젝트를 받는 중: 100% (14662/14662), 41.98 MiB | 17.51 MiB/s, 완료.
별타를 알아내는 중: 100% (7738/7738), 완료.
yeon@yeon-VMware-Virtual-Platform:~$
git clone https://github.com/hw20200500/green_airline_project
```

원격 GitHub 저장소에서 프로젝트 파일을 로컬 환경으로 복제

2.3 MySQL 컨테이너 실행

```
yeon@yeon-VMware-Virtual-Platform:~$ sudo docker run --name mysql -e MYSQL_ROOT_PASSWORD=1234 -e MYSQL_D
ATABASE=airline_db -d -p 3306:3306 mysql:8.0.32-debian --lower_case_table_names=1
unable to find image 'mysql:8.0.32-debian' locally
8.0.32-debian: Pulling from library/mysql
26c5c85e47da: Pull complete
4a04d5366306: Pull complete
2ba5474d63b9: Pull complete
5c021e7c8f4a: Pull complete
8d3a9a62a81b: Pull complete
f02df4487ae5: Pull complete
3f9945ed2391: Pull complete
fc6089a839b7: Pull complete
c63f5a3e0cd7: Pull complete
19fe4fd79b48: Pull complete
4e3d9ed5bdee: Pull complete
c8a37d470a30: Pull complete
Digest: sha256:e83ca0235010bfdb321e6838db61bdcfcd75376e2ad7344b48f9e6274e439805
Status: Downloaded newer image for mysql:8.0.32-debian
dd8c040a663eff121278fb62c93f9d8bd09184ecf315e5ad43d946aaecbc0779
yeon@yeon-VMware-Virtual-Platform:~$
```

```
sudo docker run --name mysql -e MYSQL_ROOT_PASSWORD=1234 -e
MYSQL_DATABASE=airline_db -d -p 3306:3306 mysql:8.0.32-debian --
lower_case_table_names=1
```

Docker에서 MySQL 8.0.32를 실행한다.

- name mysql: 컨테이너 이름을 mysql로 설정
- e MYSQL_ROOT_PASSWORD=1234: root 사용자 비밀번호를 1234로 설정
- e MYSQL_DATABASE=airline_db: airline_db라는 데이터베이스를 생성
- p 3306:3306: 호스트와 컨테이너의 3306 포트를 연결
- lower_case_table_names=1: 테이블 이름을 소문자로 처리

2.4 Mysql 클라이언트 설치

```
yeon@yeon-VMware-Virtual-Platform:~$ sudo apt install mysql-client-core-8.0
패키지 목록을 읽는 중입니다 ... 완료
의존성 트리를 만드는 중입니다 ... 완료
상태 정보를 읽는 중입니다 ... 완료
패키지 mysql-client-core-8.0는 이미 최신 버전입니다 (8.0.39-0ubuntu0.24.04.2).
0개 업그레이드, 0개 새로 설치, 0개 제거 및 2개 업그레이드 안 함.
yeon@yeon-VMware-Virtual-Platform:~$ |
sudo apt install mysql-client-core-8.0
```

MySQL 클라이언트를 설치하여 MySQL 서버와 상호작용할 수 있도록 설정

2.5 Java 설치

```
yeon@yeon-VMware-Virtual-Platform:~$ sudo apt install openjdk-11-jdk
패키지 목록을 읽는 중입니다 ... 완료
의존성 트리를 만드는 중입니다 ... 완료
상태 정보를 읽는 중입니다 ... 완료
패키지 openjdk-11-jdk는 이미 최신 버전입니다 (11.0.24+8-1ubuntu3~24.04.1).
0개 업그레이드, 0개 새로 설치, 0개 제거 및 2개 업그레이드 안 함.
yeon@yeon-VMware-Virtual-Platform:~$ |
sudo apt install openjdk-11-jdk
```

OpenJDK 11을 설치하여 프로젝트 실행을 위한 Java 환경을 준비

2.6 Gradle 설치

```
yeon@yeon-VMware-Virtual-Platform:~$ sudo wget https://services.gradle.org/distributions/gradle-7.6.1-bin.zip
--2024-09-10 14:30:07-- https://services.gradle.org/distributions/gradle-7.6.1-bin.zip
services.gradle.org (services.gradle.org) 해석 중 ... 104.16.72.101, 104.16.73.101, 2606:4700::6810:4965, ...
다음으로 연결 중: services.gradle.org (services.gradle.org)|104.16.72.101|:443... 연결했습니다.
HTTP 요청을 보냈습니다. 응답 기다리는 중 ... 301 Moved Permanently
위치: https://github.com/gradle/gradle-distributions/releases/download/v7.6.1/gradle-7.6.1-bin.zip [따라감]
--2024-09-10 14:30:08-- https://github.com/gradle/gradle-distributions/releases/download/v7.6.1/gradle-7.6.1-bin.zip
github.com (github.com) 해석 중 ... 20.200.245.247
다음으로 연결 중: github.com (github.com)|20.200.245.247|:443... 연결했습니다.
HTTP 요청을 보냈습니다. 응답 기다리는 중 ... 302 Found
위치: https://objects.githubusercontent.com/github-production-release-asset-2e65be/696192900/ab0a8664-5692-48c7-a75d-d70ee4c8cf2d?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=releaseassetproduction%2F20240910%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=20240910T053008Z&X-Amz-Expires=300&X-Amz-Signature=4f9073c789cd6efbd0f8e1e9d3a4f2fd5e86134530876528dce0d5f581b39235&X-Amz-SignedHeaders=host&actor_id=0&key_id=0&repo_id=696192900&response-content-disposition=attachment%3B%20filename%3Dgradle-7.6.1-bin.zip&response-content-type=application%2Foctet-stream [따라감]
--2024-09-10 14:30:08-- https://objects.githubusercontent.com/github-production-release-asset-2e65be/696192900/ab0a8664-5692-48c7-a75d-d70ee4c8cf2d?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=releaseassetproduction%2F20240910%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=20240910T053008Z&X-Amz-Expires=300&X-Amz-Signature=4f9073c789cd6efbd0f8e1e9d3a4f2fd5e86134530876528dce0d5f581b39235&X-Amz-SignedHeaders=host&actor_id=0&key_id=0&repo_id=696192900&response-content-disposition=attachment%3B%20filename%3Dgradle-7.6.1-bin.zip&response-content-type=application%2Foctet-stream
objects.githubusercontent.com (objects.githubusercontent.com) 해석 중 ... 185.199.108.133, 185.199.110.133, 185.199.111.133, ...
다음으로 연결 중: objects.githubusercontent.com (objects.githubusercontent.com)|185.199.108.133|:443...
```

```
yeon@yeon-VMware-Virtual-Platform:~$ sudo unzip -d /opt/gradle /tmp/gradle-7.6.1-bin.zip
Archive: /tmp/gradle-7.6.1-bin.zip
  creating: /opt/gradle/gradle-7.6.1/
  inflating: /opt/gradle/gradle-7.6.1/LICENSE
  inflating: /opt/gradle/gradle-7.6.1/NOTICE
  inflating: /opt/gradle/gradle-7.6.1/README
  creating: /opt/gradle/gradle-7.6.1/init.d/
  inflating: /opt/gradle/gradle-7.6.1/init.d/readme.txt
  creating: /opt/gradle/gradle-7.6.1/bin/
  inflating: /opt/gradle/gradle-7.6.1/bin/gradle
  inflating: /opt/gradle/gradle-7.6.1/bin/gradle.bat
  creating: /opt/gradle/gradle-7.6.1/lib/
  inflating: /opt/gradle/gradle-7.6.1/lib/gradle-runtime-api-info-7.6.1.jar
  inflating: /opt/gradle/gradle-7.6.1/lib/gradle-installation-beacon-7.6.1.jar
  inflating: /opt/gradle/gradle-7.6.1/lib/gradle-api-metadata-7.6.1.jar
  inflating: /opt/gradle/gradle-7.6.1/lib/gradle-launcher-7.6.1.jar
  inflating: /opt/gradle/gradle-7.6.1/lib/gradle-kotlin-dsl-7.6.1.jar
  inflating: /opt/gradle/gradle-7.6.1/lib/gradle-build-events-7.6.1.jar
  inflating: /opt/gradle/gradle-7.6.1/lib/gradle-tooling-api-7.6.1.jar
  inflating: /opt/gradle/gradle-7.6.1/lib/gradle-core-7.6.1.jar
  inflating: /opt/gradle/gradle-7.6.1/lib/gradle-jvm-services-7.6.1.jar
  inflating: /opt/gradle/gradle-7.6.1/lib/gradle-file-collections-7.6.1.jar
  inflating: /opt/gradle/gradle-7.6.1/lib/gradle-execution-7.6.1.jar
  inflating: /opt/gradle/gradle-7.6.1/lib/gradle-model-groovy-7.6.1.jar
  inflating: /opt/gradle/gradle-7.6.1/lib/gradle-model-core-7.6.1.jar
  inflating: /opt/gradle/gradle-7.6.1/lib/gradle-build-cache-7.6.1.jar
  inflating: /opt/gradle/gradle-7.6.1/lib/gradle-core-api-7.6.1.jar
  inflating: /opt/gradle/gradle-7.6.1/lib/gradle-persistent-cache-7.6.1.jar
  inflating: /opt/gradle/gradle-7.6.1/lib/gradle-resources-7.6.1.jar
```

```
yeon@yeon-VMware-Virtual-Platform:~$ sudo ln -s /opt/gradle/gradle-7.6.1 /opt/gradle/latest
sudo wget https://services.gradle.org/distributions/gradle-7.6.1-bin.zip -P /tmp
sudo unzip -d /opt/gradle /tmp/gradle-7.6.1-bin.zip
sudo ln -s /opt/gradle/gradle-7.6.1 /opt/gradle/latest
```

Gradle 7.6.1을 다운로드하고 /opt/gradle 경로에 압축을 푼다. ln -s 명령어를 사용하여 /opt/gradle/latest로 심볼릭 링크(윈도우의 바로가기와 비슷한 개념)를 만든다.

2.7 Gradle 환경 설정

```
yeon@yeon-VMware-Virtual-Platform:~$ sudo nano /etc/profile.d/gradle.sh
```

```
yeon@yeon-VMware-Virtual-Platform: ~
GNU nano 7.2 /etc/profile.d/gradle.sh *
export GRADLE_HOME=/opt/gradle/latest
export PATH=${GRADLE_HOME}/bin:${PATH}

AG 도움말  AO 기록 저장  AW 위치 찾기  AK 잘라내기  AT 실행  AC 위치  M-U
AX 나가기  AR 파일 읽기  AN 바꾸기  AU 붙여넣기  AJ 정렬  AD 지정  M-E
```

```
yeon@yeon-VMware-Virtual-Platform:~$ sudo nano /etc/profile.d/gradle.sh
yeon@yeon-VMware-Virtual-Platform:~$ source /etc/profile.d/gradle.sh
yeon@yeon-VMware-Virtual-Platform:~$
```

```
sudo nano /etc/profile.d/gradle.sh
export GRADLE_HOME=/opt/gradle/latest
export PATH=${GRADLE_HOME}/bin:${PATH}
source /etc/profile.d/gradle.sh
```

Gradle 경로를 환경 변수에 추가하기 위해 /etc/profile.d/gradle.sh 파일을 생성하고 해당 내용을 추가한다. nano 편집기로 파일 내용을 입력 후 source 명령으로 변경된 환경 변수를 적용한다.

2.8 애플리케이션 설정 (application.yml 파일 수정)

```
yeon@yeon-VMware-Virtual-Platform:~$ cd green_airline_project/
yeon@yeon-VMware-Virtual-Platform:~/green_airline_project$ nano src/main/resources/application.yml
yeon@yeon-VMware-Virtual-Platform:~/green_airline_project$
```

```
cd green_airline_project/
nano src/main/resources/application.yml (nano 대신 vim을 사용해도 무관)
```

green_airline_project 디렉토리로 이동하여 application.yml 파일에서 데이터베이스 연결 정보를 수정한다.


```

suffix: .jsp # 확장자
servlet:
  multipart:
    resolve-lazily: 'true'
    max-file-size: 20MB #최대 20MB()
datasource: # DB 설정 (MYSQL으로 마이그레이션하기) ***
url: jdbc:mysql://127.0.0.1:3306/airline_db?serverTimezone=Asia/Seoul
driver-class-name: com.mysql.cj.jdbc.Driver
username: root # 유저 이름
password: 1234 # 비밀번호 (비워두면 비밀번호 X)
sql:
init: # 초기 sql 파일 읽기

```

2.9 MySQL에 테이블 및 데이터 삽입

```

yeon@yeon-VMware-Virtual-Platform:~/green_airline_project$ mysql -h 127.0.0.1 -u root -p airline_db < sr
c/main/resources/db/table.sql
Enter password:
yeon@yeon-VMware-Virtual-Platform:~/green_airline_project$ mysql -h 127.0.0.1 -u root -p airline_db < sr
c/main/resources/db/data.sql
Enter password:
yeon@yeon-VMware-Virtual-Platform:~/green_airline_project$ |
mysql -h 127.0.0.1 -u root -p airline_db < src/main/resources/db/table.sql
mysql -h 127.0.0.1 -u root -p airline_db < src/main/resources/db/data.sql

```

table.sql과 data.sql 파일을 MySQL에 삽입하여 데이터베이스에 테이블을 생성하고 데이터를 삽입한다.

2.10 Gradle 빌드

```

yeon@yeon-VMware-Virtual-Platform:~/green_airline_project$ gradle clean build

Welcome to Gradle 7.6.1!

Here are the highlights of this release:
- Added support for Java 19.
- Introduced `--rerun` flag for individual task rerun.
- Improved dependency block for test suites to be strongly typed.
- Added a pluggable system for Java toolchains provisioning.

For more details see https://docs.gradle.org/7.6.1/release-notes.html

Starting a Gradle Daemon (subsequent builds will be faster)

> Task :test
2024-09-10 14:40:58.717 INFO 12235 --- [ionShutdownHook] com.zaxxer.hikari.HikariDat
ariPool-1 - Shutdown initiated...
2024-09-10 14:40:58.750 INFO 12235 --- [ionShutdownHook] com.zaxxer.hikari.HikariDat
ariPool-1 - Shutdown completed.

BUILD SUCCESSFUL in 1m 47s
8 actionable tasks: 8 executed
yeon@yeon-VMware-Virtual-Platform:~/green_airline_project$ |
gradle clean build

```

프로젝트를 빌드한다. clean 명령어는 이전 빌드 파일을 제거한 후 새로 빌드를 수행한다는 의미다.

2.11 Spring Boot 애플리케이션 실행

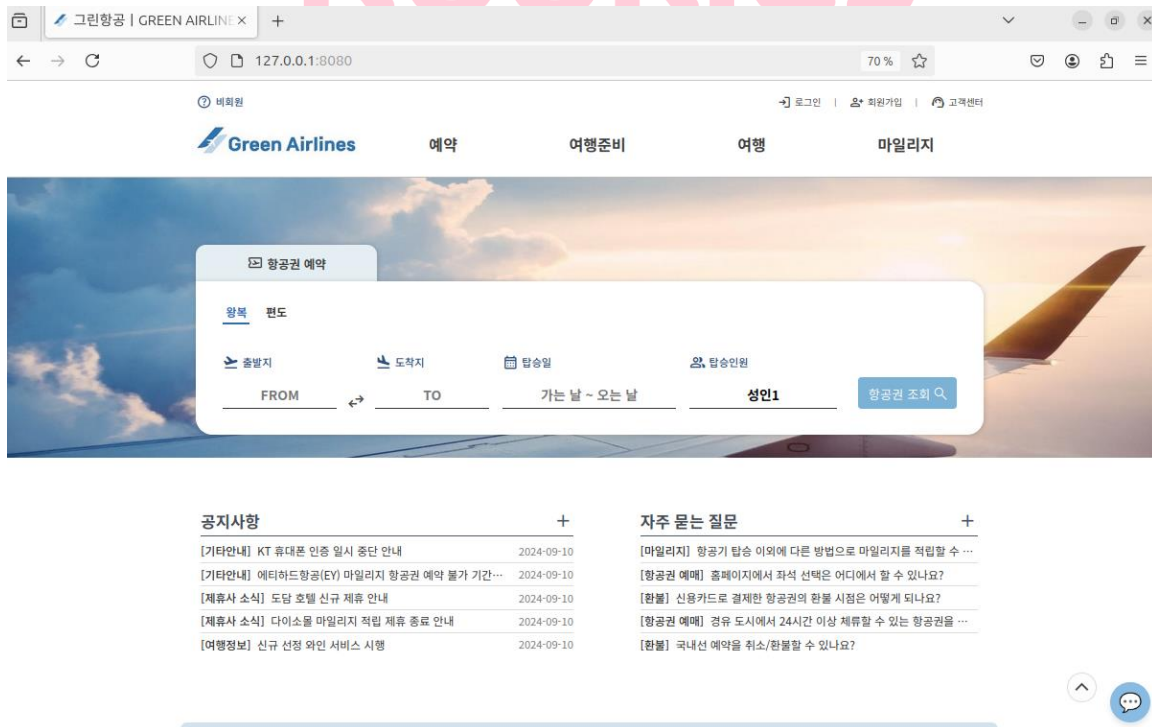
```
yeon@yeon-VMware-Virtual-Platform:~/green_airline_project$ sudo java -jar build/libs/green_airline_project-0.0.1-SNAPSHOT.jar
```



```
2024-09-10 14:42:19.971 INFO 12287 --- [main] c.green.airline.GreenAirlineApplication : Starting GreenAirlineApplication using Java 11.0.24 on yeon-VMware-Virtual-Platform with PID 12287 (/home/yeon/green_airline_project/build/libs/green_airline_project-0.0.1-SNAPSHOT.jar started by root in /home/yeon/green_airline_project)
2024-09-10 14:42:19.983 INFO 12287 --- [main] c.green.airline.GreenAirlineApplication : No active profile set, falling back to 1 default profile: "default"
2024-09-10 14:42:23.308 INFO 12287 --- [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port(s): 8080 (http)
2024-09-10 14:42:23.335 INFO 12287 --- [main] o.apache.catalina.core.StandardService : Starting service [Tomcat]
2024-09-10 14:42:23.335 INFO 12287 --- [main] org.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/9.0.74]
2024-09-10 14:42:24.292 INFO 12287 --- [main] org.apache.jasper.servlet.TldScanner : At least one JAR was scanned for TLDs yet contained no TLDs. Enable debug logging for this logger for a complete list of JARs that were scanned but no TLDs were found in them. Skipping unneeded JARs during scanning can improve startup time and JSP compilation time.
2024-09-10 14:42:24.564 INFO 12287 --- [main] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplicationContext
2024-09-10 14:42:24.564 INFO 12287 --- [main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 4417 ms
2024-09-10 14:42:24.745 INFO 12287 --- [main] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Starting...
2024-09-10 14:42:25.426 INFO 12287 --- [main] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Start completed.
```

```
sudo java -jar build/libs/green_airline_project-0.0.1-SNAPSHOT.jar
```

빌드된 JAR 파일을 실행하여 Spring Boot 애플리케이션을 시작한다.



2.8 애플리케이션 설정 단계(application.yml)에서 확인한 정보를 바탕으로 127.0.0.1:8080에 접속한다. 기본적인 페이지는 잘 뜬다. 하지만 아직 샘플 데이터가 입력되지 않아 완벽한 페이지가 아니다.

2.12 샘플 데이터 삽입

```
yeon@yeon-VMware-Virtual-Platform:~/green_airline_project$ mysql -h 127.0.0.1 -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 8.0.32 MySQL Community Server - GPL

Copyright (c) 2000, 2024, 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> |
mysql -h 127.0.0.1 -u root -p
```

새 터미널을 열어 ssh 접속을 해준 뒤 로컬에 있는 MySQL 서버에 root 사용자로 접속한다.
비밀번호는 명령어 실행 후 입력한다. application.yml 파일의 데이터베이스 연결 정보에 있던 root
계정의 비밀번호를 입력해주면 된다.

```
yeon@yeon-MVMware-Platform: ~  
mysql> show databases;  
+-----+  
| Database  
+-----+  
| airline_db  
| information_schema  
| mysql  
| performance_schema  
| sys  
+-----+  
5 rows in set (0.01 sec)  
  
mysql> use airline_db  
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> INSERT INTO recommend_board_tb  
-> (title, user_id, content, view_count, origin_name, file_name)  
-> VALUES  
-> ('싱가포르의 이국적인 거리들', 'abc',  
-> '<div class=\\"cabin_visual\\" style=\\"margin-bottom: 30px; letter-spacing: -0.02em; color: rgb(34, 34,  
34); font-family: &quot;Noto Sans KR&quot;; AppGothic, &quot;Malgun Gothic&quot;; &quot;맑은 고딕&quot;; Dotum, &quot;Apple SD Gothic Neo&quot;; Arial, Helvetica, sans-serif; font-size: 15px;\"><img src=\\"https://ozimg.flyasiana.com/editor/image/20200224/79121027-5782-43f3-8c9a-5ff7a4912553.jpeg\\" alt=\\"싱가포르의 이  
국적인 거리들 다국적인 거리, 아랍 스트리트와 하지레인 그리고 리틀 인디아를 소개합니다.\\" style=\\"vertical-align: top; border: 0px; letter-spacing: -0.02em;\"></div><div class=\\"prologue_box text-cabinBox\\" style=\\"padding-top: 30px; padding-bottom: 30px; letter-spacing: -0.02em; font-size: 14px; line-height: 24px; border-bottom: 1px solid rgb(229, 229, 229); background-image: url(&quot;https://ozimg.flyasiana.com/editor/image/20180802/2e355537-b733-4535-87f8-212b2e548258.jpeg&quot;); background-position: 50% 0px; background-size: initial; background-repeat: no-repeat; background-attachment: initial; background-origin: initial; background-clip: initial; color: rgb(34, 34, 34); font-family: &quot;Noto Sans KR&quot;; AppGothic, &quot;Malgun Gothic&quot;; &quot;맑은 고딕&quot;; Dotum, &quot;Apple SD Gothic Neo&quot;; Arial, Helvetica, sans-serif;\"><strong class=\\"hidden\\" style=\\"font-weight: bold; position: absolute; top: 0px; left: 0px; overflow: hidden; letter-spacing: -0.02em; width: 0px !important; height: 0px !important; font-size: 0px !important; line-height: 0 !important;\"></strong><n style=\\"margin-bottom: 0px; letter-spacing: -0.02em;\">싱가포르는 다양한 문화들이 모
```

MySQL 서버에 현재 생성된 모든 데이터베이스의 이름을 출력 후 airline_db라는 데이터베이스를 선택한다. 이후 실행되는 모든 SQL 쿼리는 airline_db 데이터베이스에 적용된다. 아래에 있는 샘플 데이터를 삽입해주면 된다.

86 lines (81 loc) · 563 KB

```
Code Blame Raw Copy Download Edit View Source
1  INSERT INTO recommend_board_tb
2  (title, user_id, content, view_count, origin_name, file_name)
3  VALUES
4  ('[대만] 타이베이 대만에 또 가고 싶은 이유_삼미식당', 'abc',
5   '<p data-ke-size="size14" style="box-sizing: inherit; font-size: 0.87em; margin-bottom: 28px; line-height: 1.88; color: rgb(102,
6   3371, 'Taiwan Taipei.jpeg', '/images/board/TaiwanTaipei.jpeg)'),
7   ('[태국] 방콕 위치, 시설 다 좋은 호텔 추천_그랜드 센터 포인트 터미널21 방콕', 'a123',
8   '<p data-ke-size="size14" style="box-sizing: inherit; font-size: 0.87em; margin-bottom: 28px; line-height: 1.88; color: rgb(102,
9   1108, 'Bangkok Thailand.jpeg', '/images/board/BangkokThailand.jpeg)'),
10  ('열심히 먹고 온 제주도 여행', 'a123',
11  '<p style="margin-bottom: 1.3em; orphans: 3; widows: 3; transition: all 0.2s ease-in-out 0s; font-size: 18px; font-family: -apple
12  5742, 'Jeju Island.jpg', '/images/board/JejuIsland.jpg'),
13  ('후쿠오카 여행일기', 'abc',
14  '<p style="margin-right: auto; margin-bottom: 28px; margin-left: auto; font-family: "Spoqa Han Sans", sans-serif; font-size: 15px;
15  4492, 'Fukuoka.jpeg', '/images/board/Fukuoka.jpeg'),
16  ('후쿠오카 여행일기', 'abc',
```

main ▾ green_airline_project / 게시판 샘플데이터.txt ↑ Top

```
Code Blame Raw Copy Download Edit View Source
24
25  ('그들이 사랑했던 파리', 'abc',
26   '<div class="cabin_visual" style="margin-bottom: 30px; letter-spacing: -0.02em; color: rgb(34, 34, 34); font-family: &quot;Noto
27   987, 'thumbnail8.jpeg', '75d8d6bb-4fb3-4d06-88b9-3c016069f6d5_thumbnail8.jpeg'),
28
29  ('방콕, 숨은 매력을 찾아 떠나는 여행', 'abc',
30   '<div class="cabin_visual" style="margin-bottom: 30px; letter-spacing: -0.02em; color: rgb(34, 34, 34); font-family: &quot;Noto
31   3317, 'thumbnail9.jpeg', '5bc2ba98-1d93-4dab-a930-cfa00dcdd52f_thumbnail9.jpeg'),
32
33  ('런던, 피터삼 널서리(Petersham nurseries)에 가다.', 'abc',
34   '<div class="cabin_visual" style="margin-bottom: 30px; letter-spacing: -0.02em; color: rgb(34, 34, 34); font-family: &quot;Noto
35   1142, 'thumbnail11.jpeg', '30b662e7-4567-4aad-ac49-e3595f0bc27c_thumbnail11.jpeg'),
36
37  ('뉴욕, 거리에서 만나는 예술작품들의 세계', 'abc',
38   '<div class="cabin_visual" style="margin-bottom: 30px; letter-spacing: -0.02em; color: rgb(34, 34, 34); font-family: &quot;Noto
39   2014, 'thumbnail12.jpeg', 'fb5d89dd-aaeb-4578-8a3c-dea2a4690641_thumbnail12.jpeg');
40
41
42  -- 좋아요 샘플 데이터
43  INSERT INTO like_heart_tb (board_id, user_id)
44  VALUES |
```

세미콜론(;) 단위로 나눠 입력한다.

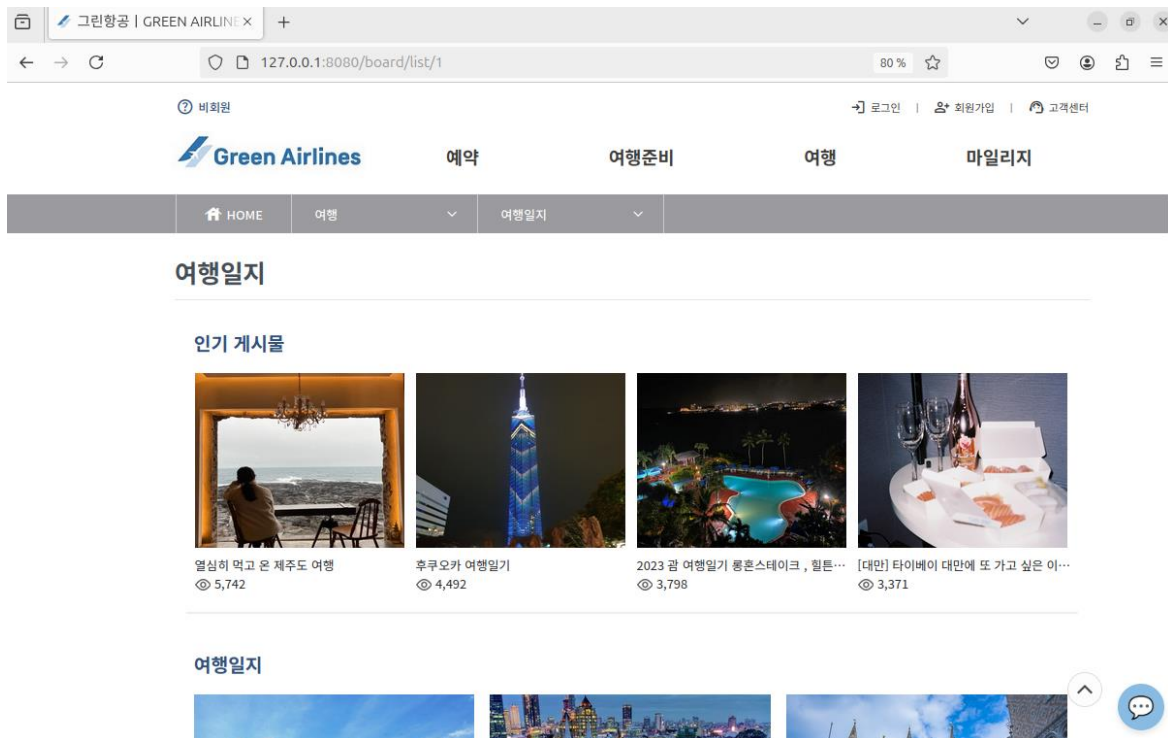
```
드 파 크 로 할 랐 트 .</p>',
-> 989, 'Sydney(2).jpeg', '/images/board/Sydney(2).jpeg');
Query OK, 23 rows affected (0.05 sec)
Records: 23 Duplicates: 0 Warnings: 0

mysql> ALTER TABLE airline_db.baggage_request_tb ADD ticket_id VARCHAR(25);
Query OK, 0 rows affected (0.02 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> alter table airline_db.baggage_request_tb add foreign key(ticket_id) references airline_db.ticket_tb(i
d) on delete cascade;
Query OK, 2 rows affected (0.04 sec)
Records: 2 Duplicates: 0 Warnings: 0

mysql> |
```

복사하여 붙여 넣을 때에는, 쿼리가 길면 쿼리 입력에 시간이 걸려 조금 기다려 주면 된다. mysql에서 나오고 싶을 땐 exit 명령어를 사용하면 된다.



샘플 데이터가 정상적으로 삽입되어 게시물을 볼 수 있다.

3. 챗봇 서버 구축 및 실행

3.1 설정 및 라이브러리 설치

```
yeon@yeon-VMware-Virtual-Platform:~$ cd green_airline_project/
yeon@yeon-VMware-Virtual-Platform:~/green_airline_project$ python3 app.py
Traceback (most recent call last):
  File "/home/yeon/green_airline_project/app.py", line 4, in <module>
    import pymysql
ModuleNotFoundError: No module named 'pymysql'
yeon@yeon-VMware-Virtual-Platform:~/green_airline_project$ |
```

```
cd green_airline_project/
```

```
python3 app.py
```

green_airline_project 디렉토리로 이동하여 파이썬 챗봇 파일을 실행하면 실행에 필요한 라이브러리를 알려준다.

```
yeon@yeon-VMware-Virtual-Platform:~/green_airline_project$ sudo apt install python3-pip
[sudo] yeon 암호 :
패키지 목록을 읽는 중입니다 ... 완료
의존성 트리를 만드는 중입니다 ... 완료
상태 정보를 읽는 중입니다 ... 완료
다음의 추가 패키지가 설치될 것입니다 :
binutils binutils-common binutils-x86-64-linux-gnu build-essential dpkg-dev fakeroot g++ g++-
g++-13-x86-64-linux-gnu g++-x86-64-linux-gnu gcc gcc-13 gcc-13-x86-64-linux-gnu
gcc-x86-64-linux-gnu javascript-common libalgorithm-diff-perl libalgorithm-diff-xs-perl
libalgorithm-merge-perl libasan8 libbinutils libcc1-0 libctf-nobfd0 libctf0 libdpkg-perl
libexpat1-dev libfakeroot libfile-fcntllock-perl libgcc-13-dev libgprofng0 libhwasan0 libit
libjs-jquery libjs-sphinxdoc libjs-underscore liblsan0 libpython3-dev libpython3.12-dev
libquadmath0 libsframe1 libstdc++-13-dev libtsan2 libubsan1 lto-disabled-list make python3-
python3-setuptools python3-wheel python3.12-dev zlib1g-dev
제안하는 패키지 :
binutils-doc gprofng-gui debian-keyring g++-multilib g++-13-multilib gcc-13-doc gcc-multilib
```



```
sudo apt install python3-pip
```

우선 Python의 패키지 관리자인 pip를 설치한다. Python 라이브러리를 설치할 때 필요하다. 설치 후 pip를 사용하려 하면 아래와 같은 에러가 뜬다.

```
yeon@yeon-VMware-Virtual-Platform:~/green_airline_project$ pip install pymysql
error: externally-managed-environment

× This environment is externally managed
↳ To install Python packages system-wide, try apt install
python3-xyz, where xyz is the package you are trying to
install.

If you wish to install a non-Debian-packaged Python package,
create a virtual environment using python3 -m venv path/to/venv.
Then use path/to/venv/bin/python and path/to/venv/bin/pip. Make
sure you have python3-full installed.

If you wish to install a non-Debian packaged Python application,
it may be easiest to use pipx install xyz, which will manage a
virtual environment for you. Make sure you have pipx installed.

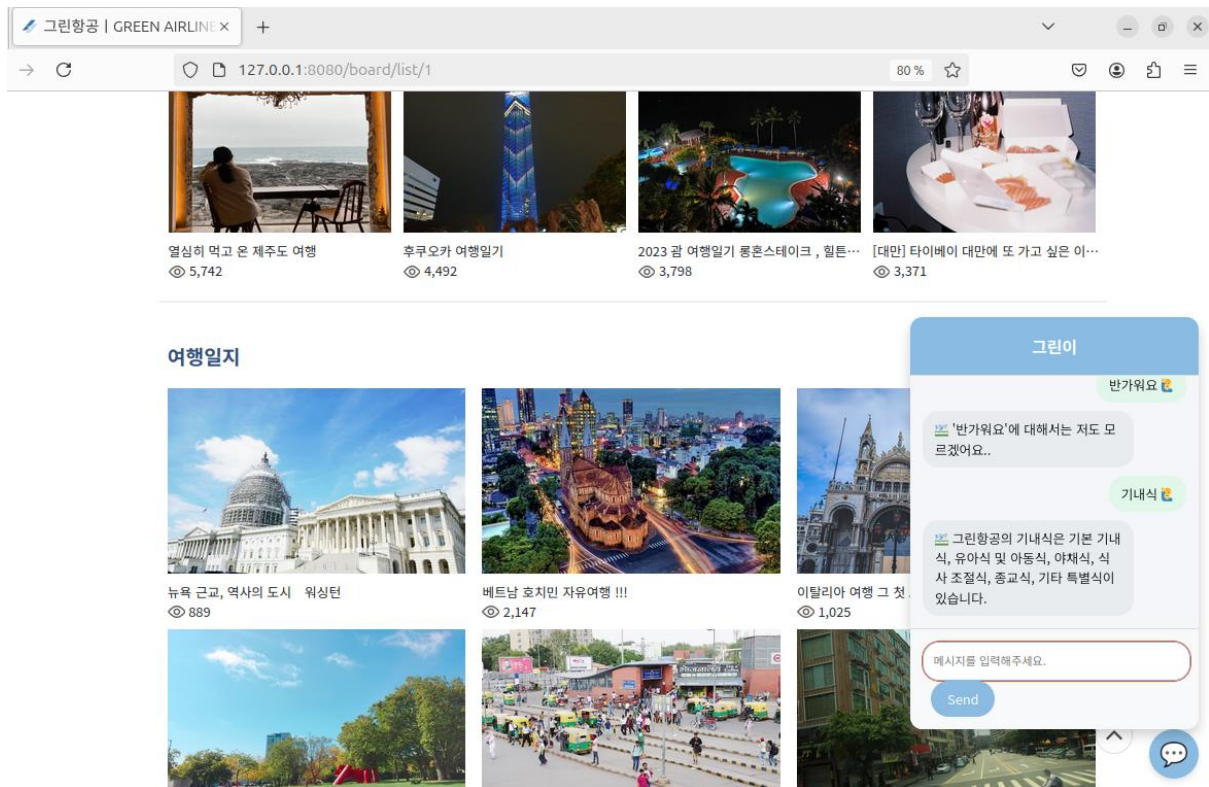
See /usr/share/doc/python3.12/README.venv for more information.

note: If you believe this is a mistake, please contact your Python installation
provider. You can override this, at the risk of breaking your Python installation
and system-packages.
hint: See PEP 668 for the detailed specification.
```

```
yeon@yeon-VMware-Virtual-Platform:~/green_airline_project$ python3 -m pip config set global.break-system-packages true
Writing to /home/yeon/.config/pip/pip.conf
yeon@yeon-VMware-Virtual-Platform:~/green_airline_project$
python3 -m pip config set global.break-system-packages true
```

이 에러는 pip의 설정을 변경하여, 시스템 패키지와의 충돌을 허용하는 옵션을 활성화 해주는 것으로 해결한다.

```
yeon@yeon-VMware-Virtual-Platform:~/green_airline_project$ pip install pymysql
Defaulting to user installation because normal site-packages is not writeable
Collecting pymysql
  Downloading PyMySQL-1.1.1-py3-none-any.whl.metadata (4.4 kB)
  Downloading PyMySQL-1.1.1-py3-none-any.whl (44 kB)
----- 45.0/45.0 kB 26.0 kB/s eta 0:00:00
Installing collected packages: pymysql
Successfully installed pymysql-1.1.1
yeon@yeon-VMware-Virtual-Platform:~/green_airline_project$ python3 app.py
Traceback (most recent call last):
  File "/home/yeon/green_airline_project/app.py", line 5, in <module>
    from flask import Flask, jsonify, render_template, render_template_string, re
ModuleNotFoundError: No module named 'flask'
yeon@yeon-VMware-Virtual-Platform:~/green_airline_project$ pip install flask
Defaulting to user installation because normal site-packages is not writeable
Collecting flask
  Downloading flask-3.0.3-py3-none-any.whl.metadata (3.2 kB)
Collecting werkzeug>=3.0.0 (from flask)
  Downloading werkzeug-3.0.4-py3-none-any.whl.metadata (3.7 kB)
```

챗봇이 정상적으로 실행되는 것을 볼 수 있다.

4. 도커 서비스 정지 및 실행 방법

4.1 명령어 사용 예시

도커 컨테이너 프로세스 정보 조회

```
sudo docker ps -> 실행 중인 프로세스 조회
```

```
sudo docker ps -a -> 모든 프로세스 조회
```

조회한 정보를 바탕으로 프로세스 정지, 실행, 삭제 작업 ('mysql'은 프로세스 예시임)

```
sudo docker stop mysql -> 정지
```

```
sudo docker start mysql -> 시작
```

```
sudo docker rm mysql -> 삭제
```

SK Shieldus Rookies