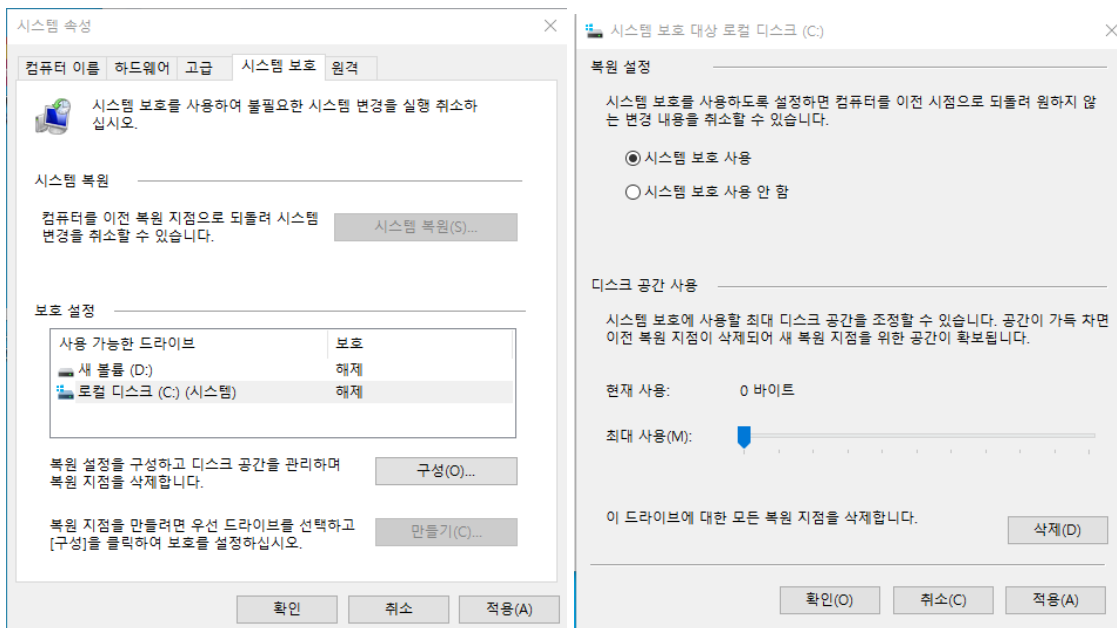


1) 암호화 방식

종류	RSA	DSA
암호화 원리	소인수 분해의 어려움에 기반한다	Discrete Logarithm 문제의 어려움에 기반한다.
Key 생성	RSA는 Key 생성이 느리다.	ElGamal 비슷하게 사용하므로, RSA보다 Key 생성이 빠르다
Encryption (Verifying)	Encryption 이 DSA보다 빠르다	Encryption이 RSA보다 느리다
Decryption (Signing)	Decryption은 DSA보다 느리다	Decryption은 RSA보다 빠르다
비고	Encryption (검증)에 최적화	Decryption (서명)에 최적화

- RSA(Rivest-Shamir-Adleman) : 비대칭키, 암호키와 복호키(해독)가 다름
- DAS(Digital Signature Algorithm) : 대칭키, 암호를 여는 열쇠가 하나만, 공인인증서

2) 복원 지점 만들기 : 설정 > 복원 검색



- 구성 : 이미지파일화해서 저장
- 최대 사용 : 4~6% 설정(권장)
- 만들기 > 이름 설정 (기억할 시점) > 시스템 복원
 - 윈도우 업데이트 완료이후 사용 권장

3) AWS

3-1) 키페어 생성 > 이름 12-11kmh

키 페어

프라이빗 키와 퍼블릭 키로 구성되는 키 페어는 인스턴스에 연결할 때 자격 증명을 증명하는 데 사용하는 보안 자격 증명 세트입니다.

이름

12-11kmh

이름에는 최대 255개의 ASCII 문자가 포함됩니다. 앞 또는 뒤에 공백을 포함할 수 없습니다.

키 페어 유형 [정보](#)

☒ RSA

☐ ED25519

프라이빗 키 파일 형식

☒ .pem
OpenSSH와 함께 사용

☐ .ppk
PuTTY와 함께 사용

태그 - [선택 사항](#)

리소스에 연결된 태그가 없습니다.

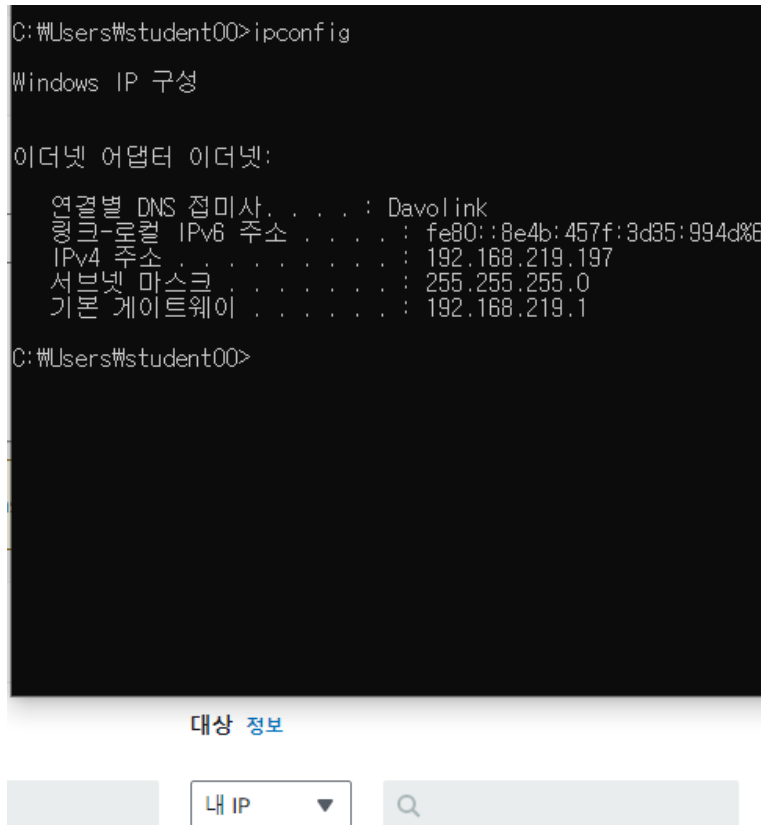
새로운 태그 추가

최대 50개의 태그를 더 추가할 수 있습니다.

- pem : 리눅스와 기타 설정 / ppk : 윈도우만, 타인 전송 어려움

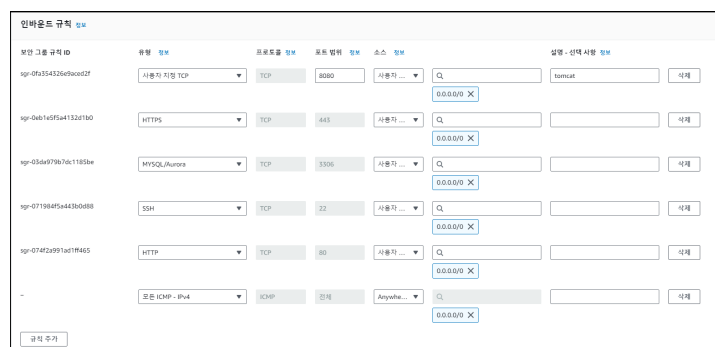
3-2) 보안그룹 > 보안그룹 생성

- 기본세부정보
 - 보안그룹 이름 : 12-11kmh
 - 설명 : exm
- 인바운드 규칙 추가 > 포트번호 : 3306(DB통신 기본 포트) > 사용자 : Anywhere IP4
 - 인바운드 규칙의 유형 : 구분을 위한 이름일 뿐 포트범위가 중요
 - 인터넷 제어 메시지 프로토콜(Internet Control Message Protocol, ICMP) : 핑 주고 받음

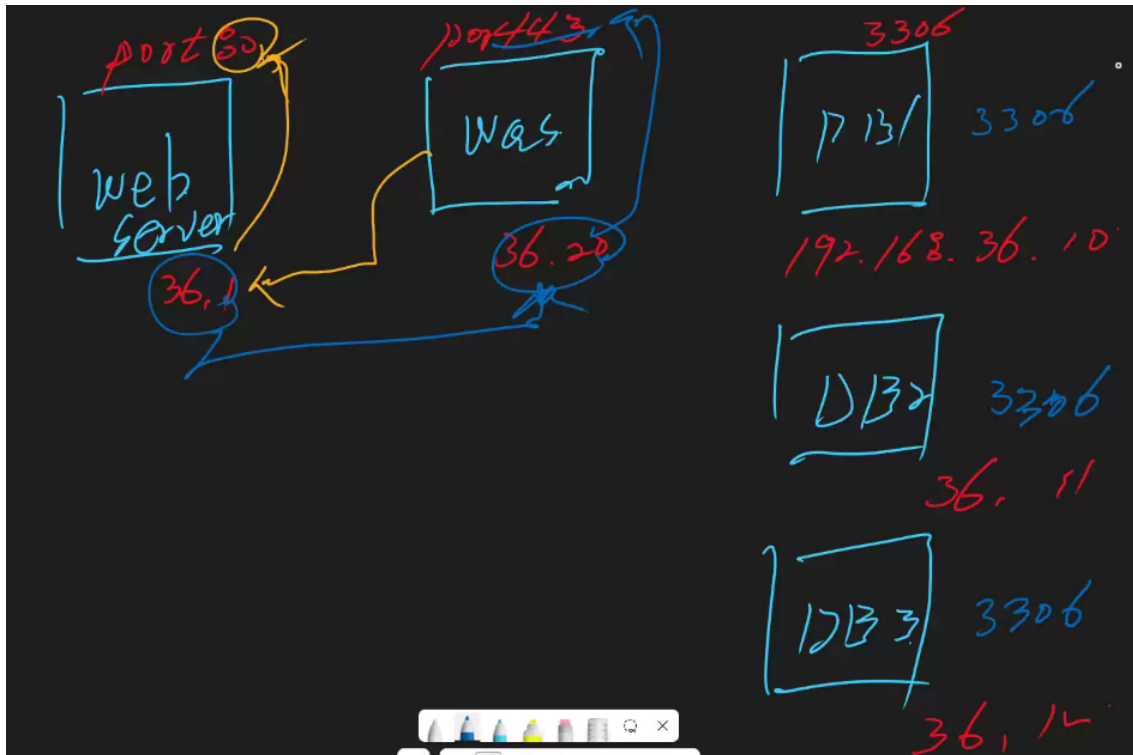


- 내 IP 로 할 경우 외부에서 접속 불가
- 모뎀(강의장)에 들어오는 IP : 112.221.230.98 / 32
- Anywhere IP4 : 0.0.0.0/0 - 전세계 어디에서든 접근 가능


- 아웃바운드 규칙 : 내보낼 때
 - 공공기관, 인트라넷 : 유형 - IP 를 설정



3-3) 탄력적 IP > 탄력적 IP 할당 > 할당 : 미설정시 과금주의



- 작업 > 탄력적 IP 주소 릴리즈 > 전체 삭제됨
- DB 와 스토리지는 보안규칙에서 적용
- 서버가 4개면 4개가 필요
- 여러 DB의 포트 번호는 3306를 중복 사용 가능, IP 주소는 다르게 할당해야 함
 - 포트는 컴퓨터상에서 통신하기 위함, 다른 컴퓨터도 같은 포트 번호로 통신가능(동번호, 210등)
 - IP는 컴퓨터간의 주소(호수번호, 1504호)

- **[AWS]**  **EIP(탄력적 IP) 개념 & 사용 세팅 정리**
 - EIP(Elastic IP Address)는 인터넷을 통해 접속할 수 있는 고정적인 공인 IP 주소를 할당할 수 있고, 인스턴스에 연결할 수 있는 서비스다.
 - 모든 인스턴스 또는 네트워크 인터페이스에 탄력적 IP 주소를 연결 가능하다.
 - EIP를 할당받고 삭제하기 전까지 IP 주소를 꼭 유지할 수 있다.
 - ENI(Elastic Network Interface)에 퍼블릭 IP를 달게 되면 외부에서 접속
 - 하지만, 인스턴스의 **Public IP**는 고정된 IP 주소가 아니라 유동적인 IP 주소 이다.
 - AWS에서는 바로 한 번 할당 받으면 절대 바뀌지 않은 **Elastic IP**를 제공한다.

3-4) 인스턴스 시작

- 이름 : 12-11kmh
- Application and OS Images (Amazon Machine Image)
 - Ubuntu > ubuntu server 20.04 LTS 선택
 - 서버 하나기준 750시간 무료(1/n)
- 키페어 : 이전에 설정한 12-11kmh 로 설정
- 네트워크 설정 > 기존 보안그룹 선택
 - 보안그룹생성 : launch-wizard-1
- 스토리지 구성
 - 25gb
- 인스턴스 시작

3-5) 인스턴스 연결 > 연결 > 연결 : ubuntu 서버 열림

- 현재의 root 는 ubuntu / AWS의 경우 ec2-user
- 로딩이 오래걸릴 경우 F5
- 퍼블릭 IP 주소라고 확인되지만 사설이다.
- 사설 : 외부 인터넷과 접속이 불가, 모두에게 공인 IP 부여 불가능, 개인만해도 5~6개 사용
- 공인 : 외부 인터넷과 접속이 가능

```
aws | 서비스 | 🔍 검색 | [알트+S]

Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 6.2.0-1012-aws x86_64)

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

System information as of Mon Dec 11 05:07:52 UTC 2023

System load:  0.21875      Processes:            101
Usage of /:   6.4% of 24.05GB Users logged in:       0
Memory usage: 20%          IPv4 address for eth0: 172.31.8.123
Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

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.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-8-123:~$
```

i-054b8bc3b948cd2f2 (12-11kmh)

PublicIPs: 3.38.183.185 PrivateIPs: 172.31.8.123

- ICMP 열었기 때문에 ping 주고 받기 가능해짐
- AWS - EC2 를 생성시, 외부와 접속가능한 IP 하나는 필요 이것이 퍼블릭 IP
 - 퍼블릭 IP : 외부와 통신용, 공인, AWS가 배정해줌
 - 사설 : 외부와 통신 불가
 - 단, 사설을 공인처럼 바꾸는게 공유기(Router)
 - 퍼블릭 IP에 탄력적 IP 로 바꿈 : 고정 IP 가 됨, 공인
- 중지 : 퍼블릭 IP 에 주소가 없어짐, 프라이빗은 존재함
- 종료 : 인스턴스 삭제

3-6) 탄력적 IP 주소¹ 할당 > 할당 > 연결

- 프라이빗 아이피는 AWS내에서 다른 IP들과 구분하기 위함
- 인스턴스 : 기존 설정한 것 선택 (1)
- 프라이빗 아이피 주소 : 자동으로 선택됨 (2)
- 이후 인스턴스 연결에서 퍼블릭 IP 주소가 바뀐것이 확인됨
 - 인스턴스 연결시 사용자이름 **ubuntu** 가 쓰는 것이 루트명(사용자명)과 동일
 - AWS에서는 로그인 하지않고 우분투 사용가능

i-054b8bc3b948cd2f2 (12-11kmh)
PublicIPs: 52.78.183.158 PrivateIPs: 172.31.8.123

```
ca. 명령 프롬프트
Microsoft Windows [Version 10.0.19045.3693]
(c) Microsoft Corporation. All rights reserved.

C:\Users\student00>ping 52.78.183.158

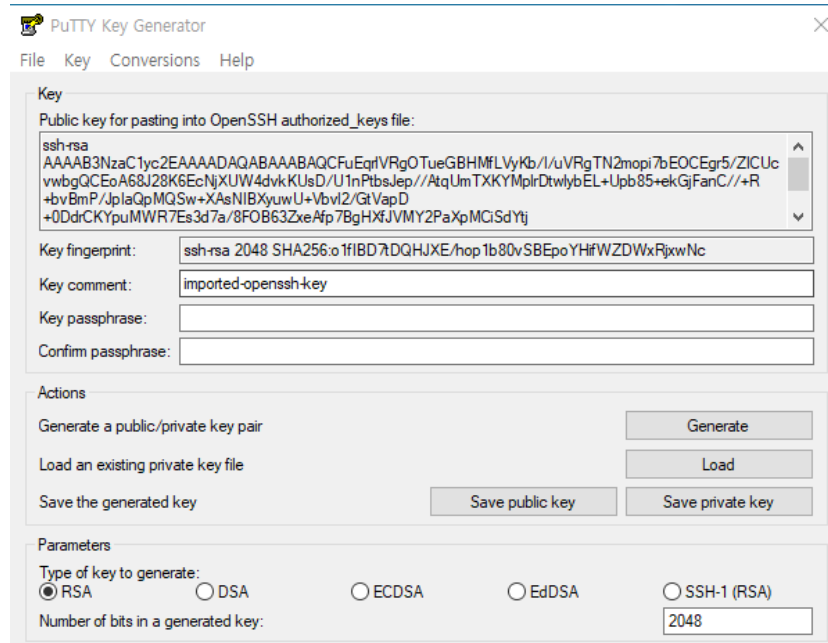
Ping 52.78.183.158 32바이트 데이터 사용:
52.78.183.158의 응답: 바이트=32 시간=4ms TTL=48
52.78.183.158의 응답: 바이트=32 시간=5ms TTL=48
52.78.183.158의 응답: 바이트=32 시간=5ms TTL=48
52.78.183.158의 응답: 바이트=32 시간=4ms TTL=48

52.78.183.158에 대한 Ping 통계:
패킷: 보낸 = 4, 받음 = 4, 손실 = 0 (0% 손실),
왕복 시간(밀리초):
    최소 = 4ms, 최대 = 5ms, 평균 = 4ms
```

- 인스턴스 재부팅 권장

4) Putty

4-1) puttygen :: 키페어 생성

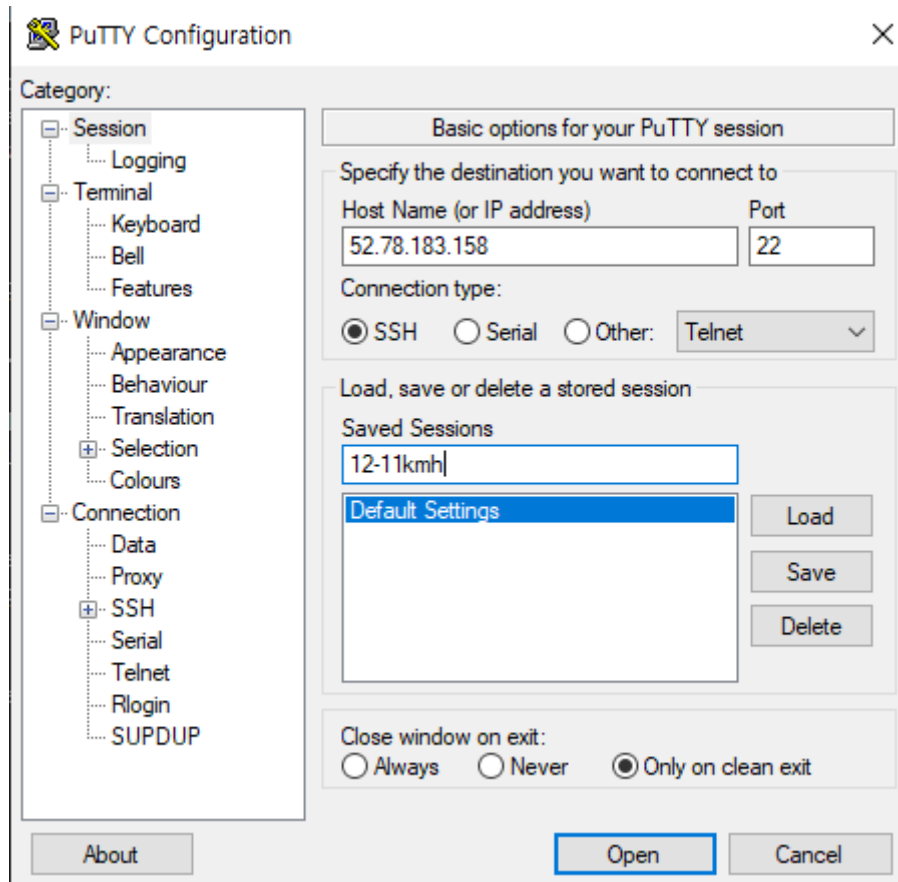


- generate : 윈도우상에서 키페어 생성, 생성시 전체 설정 동일화 필요
- load : all.files 로 찾기 > 12-11kmh.pem > successfully (1)
- save private key(.ppk = 푸티) > 12-11kmh.ppk (2)

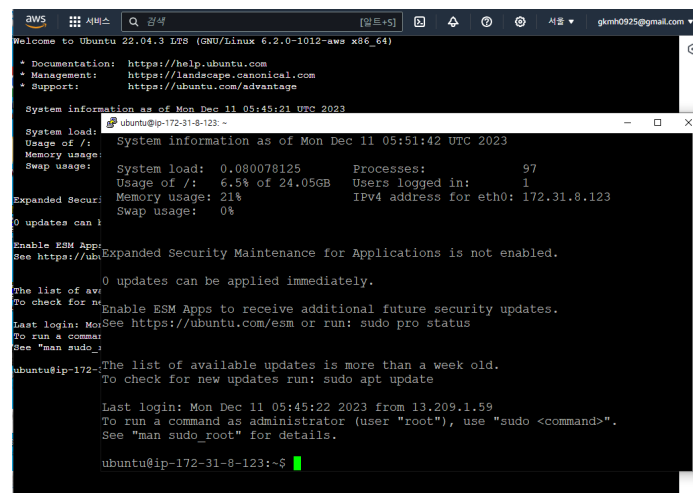
¹ <https://www.youtube.com/watch?v=pON9FLO49SY>

4-2) putty

- ubuntu 는 복붙 불가, 불편, 외부 변경 불허용
- 인스턴스 연결 > 연결 > 우분투(검은화면)의 publicIPs 확인
- PublicIPs: 52.78.183.158 을 Host Name 에 복붙 (1)



- appearance : 폰트 변경
- SSH > Auth (2)
 - Credential > Browser > 12-11kmh.ppk (2개다 설정)
 - GSSAPI > Browser > 12-11kmh.ppk
 - 주의 오픈 하지 말고, Session 으로 가서 저장 부터, 이후 open
 - 에러 메시지 무시, 초기 한번만 나옴
- accept > ubuntu : 이제 두 화면은 동일



- 우분투 서버에서는 접속되는 것과 동일

4-3) mariadb 설치 및 설정 : ubuntu 화면상에서

- `sudo apt-get update -y`

```
ubuntu@ip-172-31-8-123: ~  
kB]  
Get:33 http://security.ubuntu.com/ubuntu jammy-security/main amd64 c-n-f Metadata  
a [11.4 kB]  
Get:34 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Package  
s [1179 kB]  
Get:35 http://security.ubuntu.com/ubuntu jammy-security/restricted Translation-e  
n [191 kB]  
Get:36 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 c-n-f M  
etadata [520 B]  
Get:37 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages  
[815 kB]  
Get:38 http://security.ubuntu.com/ubuntu jammy-security/universe Translation-en  
[152 kB]  
Get:39 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 c-n-f Met  
adata [16.8 kB]  
Get:40 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 Package  
s [36.5 kB]  
Get:41 http://security.ubuntu.com/ubuntu jammy-security/multiverse Translation-e  
n [7060 B]  
Get:42 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 c-n-f M  
etadata [260 B]  
Fetched 28.7 MB in 5s (5536 kB/s)  
Reading package lists... Done  
ubuntu@ip-172-31-8-123:~$
```

- `-y` : all yes
- `sudo apt-get upgrade > y`
- `date`

```
ubuntu@ip-172-31-8-123:~$ date  
Mon Dec 11 06:07:37 UTC 2023  
ubuntu@ip-172-31-8-123:~$
```

- mariaDB 설치 : `sudo apt-get install mariadb-server > y`
 - EC2 에서 설치된 상태
- `cd /` : 최상위 루트 상태

```
ubuntu@ip-172-31-8-123:~$ cd /  
ubuntu@ip-172-31-8-123:/$
```

- `sudo mysql_secure_installation > (enter) > y> 1234, 1234 > y > y > y > y`
- `mysql -u root -p > 1234` : 접속 불가
- `sudo mariadb` : 접속됨

- show databases;

```
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
+-----+
```

- select host, user, plugin from mysql.user;

```
+-----+-----+-----+
| host      | user  | plugin      |
+-----+-----+-----+
| localhost | root  | unix_socket |
+-----+-----+-----+
```

- unix_socket : 으로 인해 mysql -u root -p 가 통용되지 않음
 - 외부 접속 불가일경우 사용 가능

- select version();

```
MariaDB [(none)]> select version();
+-----+
| version() |
+-----+
| 10.6.12-MariaDB-0ubuntu0.22.04.1 |
+-----+
1 row in set (0.000 sec)
```

- update mysql.user set plugin = 'mysql_native_password' where user='root';
- flush privileges;
 - mysql 전체에 소켓-> mysql_native_password 강제로 입력
- select host, user, plugin from mysql.user; plugin -> mysql_native_password 변경 확인됨
- select @@global.time_zone, @@session.time_zone;

```
MariaDB [(none)]> select @@global.time_zone, @@session.time_zone;
+-----+-----+
| @@global.time_zone | @@session.time_zone |
+-----+-----+
| SYSTEM              | SYSTEM              |
+-----+-----+
1 row in set (0.000 sec)
```

- 우분투의 시간 설정과 동일

- set global time_zone='Asia/Seoul';
set time_zone='Asia/Seoul'
 - 실패, mariadb 자체에서 시각을 사용하지 않으므로
- 마리아디비에서 exit 한후 실행
 - sudo timedatectl set-timezone Asia/Seoul &&
sudo timedatectl set-ntp true;

```
ubuntu@ip-172-31-8-123:~$ sudo timedatectl set-timezone Asia/Seoul && sudo timedatectl set-ntp true;
ubuntu@ip-172-31-8-123:~$ date
Mon Dec 11 16:09:05 KST 2023
```

- sudo dpkg -i | grep mariadb
 - 설치된 **db** 확인
 - dpkg : 데비안 패키지
- sudo apt-get purge mariadb-server
 - **db 삭제** 중 server에 해당하는 파일만 지움
 - 리눅스는 하나씩
 - -get : 메모리 상주나 해당하는 내용 전체 로드

```
ubuntu@ip-172-31-8-123:~$ sudo dpkg -i | grep mariadb
ii  libmariadb3:amd64          1:10.6.12-0ubuntu0.22.04.1
64      MariaDB database client library
ii  mariadb-client-10.6        1:10.6.12-0ubuntu0.22.04.1
64      MariaDB database client binaries
ii  mariadb-client-core-10.6   1:10.6.12-0ubuntu0.22.04.1
64      MariaDB database core client binaries
ii  mariadb-common             1:10.6.12-0ubuntu0.22.04.1
64      MariaDB common configuration files
ii  mariadb-server-10.6        1:10.6.12-0ubuntu0.22.04.1
64      MariaDB database server binaries
ii  mariadb-server-core-10.6   1:10.6.12-0ubuntu0.22.04.1
64      MariaDB database core server files
ubuntu@ip-172-31-8-123:~$ sudo apt-get purge mariadb-server^C
ubuntu@ip-172-31-8-123:~$ sudo apt-get purge
```

sudo apt-get purge 상태에서 지우고자 하는 디렉토리를 마우스 우클릭하면 복붙

- 완전 삭제시 sudo dpkg -i | grep mariadb 하면 아무 리스트가 안뜸
- sudo systemctl start mariadb
- sudo systemctl stop mariadb
- sudo systemctl status mariadb
 - 확인시

```
Active: active (running)
Active: inactive (dead)
```

- ctrl+c + (enter) 로 exit
- AWS의 경우 ec2-user