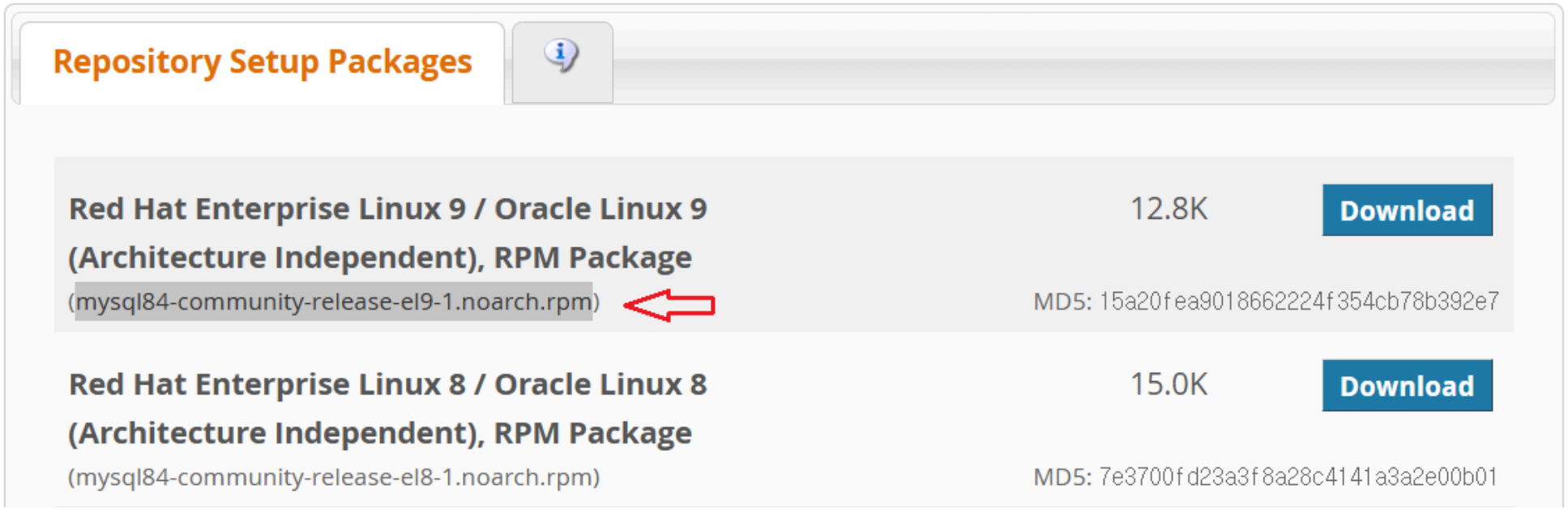
MySQL 설치

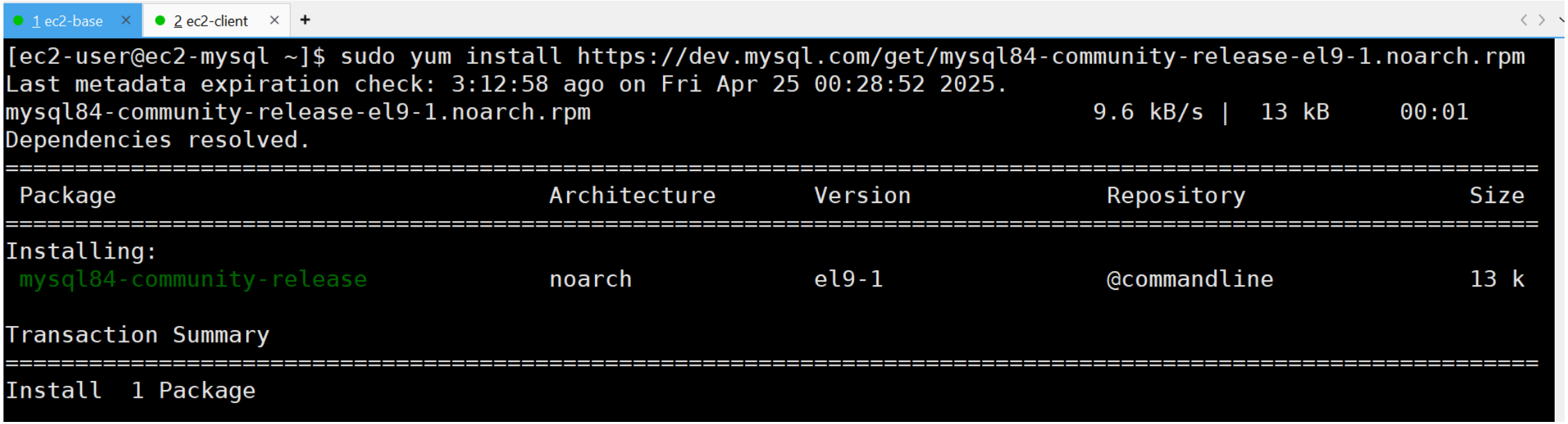
단계1: Adding the MySQL Yum Repository

- 최신버전 복사



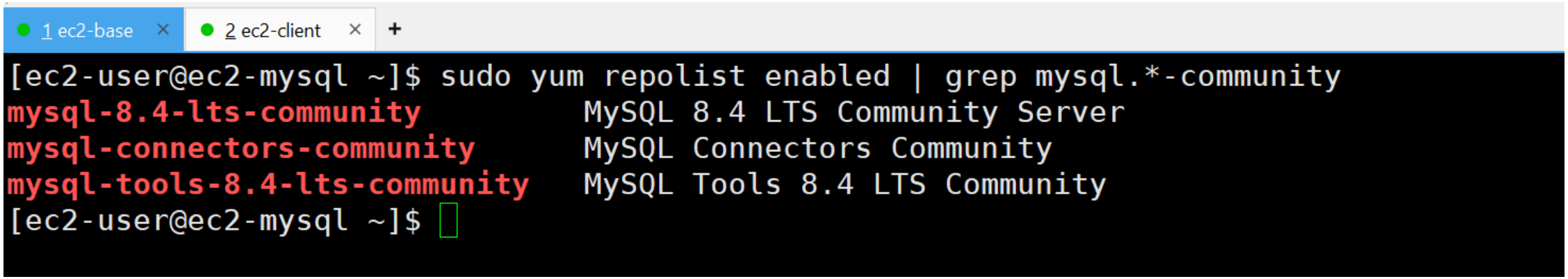
- 설치

```
# sudo yum install https://dev.mysql.com/get/{최신 버전} -y
sudo yum install https://dev.mysql.com/get/mysql84-community-release-el9-1.noarch.rpm -y
```



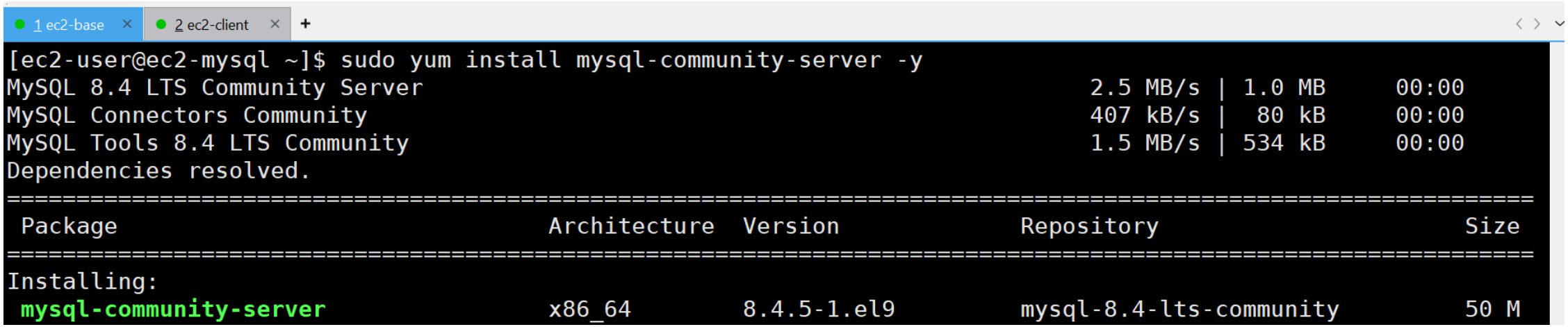
- 설치 결과 확인

```
sudo yum repolist enabled | grep mysql.*-community
```



단계2: Installing MySQL

```
sudo yum install mysql-community-server -y
```



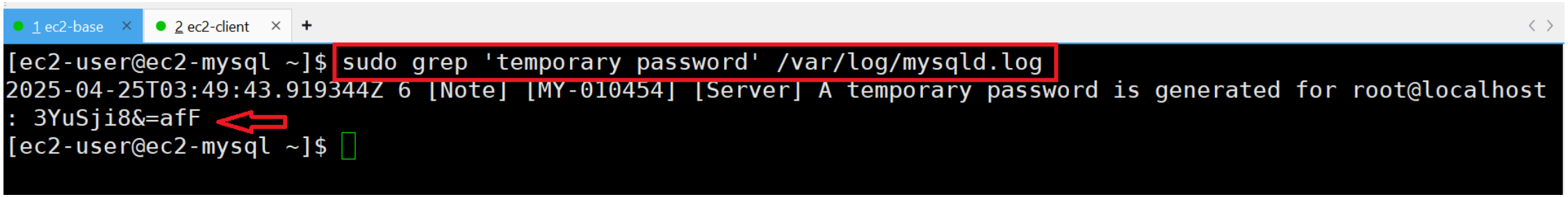
단계3: Starting the MySQL Server

```
# start mysql server
sudo systemctl start mysqld
sudo systemctl status mysqld
```

```
[ec2-user@ec2-mysql ~]$ sudo systemctl start mysqld
[ec2-user@ec2-mysql ~]$
[ec2-user@ec2-mysql ~]$ sudo systemctl status mysqld
● mysqld.service - MySQL Server
     Loaded: loaded (/usr/lib/systemd/system/mysqld.service; enabled; preset: disabled)
     Active: active (running) since Fri 2025-04-25 03:49:48 UTC; 46s ago
       Docs: man:mysqld(8)
             http://dev.mysql.com/doc/refman/en/using-systemd.html
    Process: 33080 ExecStartPre=/usr/bin/mysqld_pre_systemd (code=exited, status=0/SUCCESS)
```

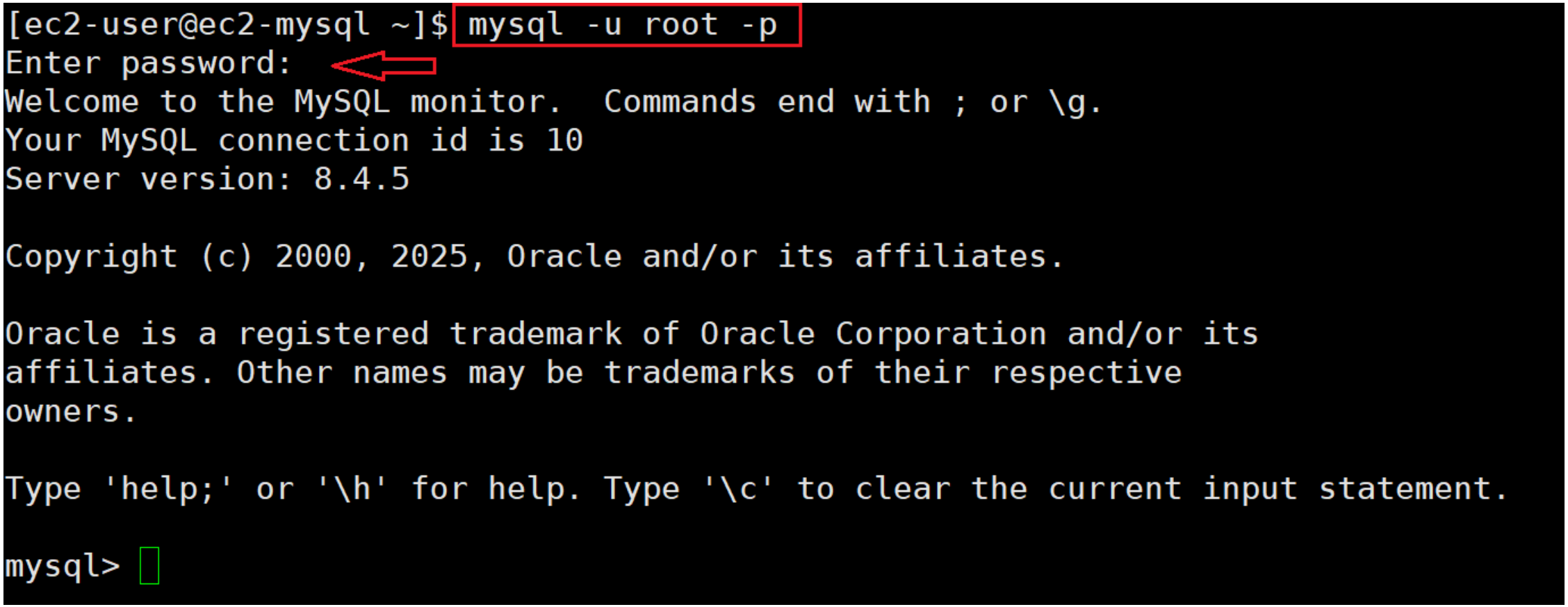
단계4: 임시 비밀번호 확인

```
sudo grep 'temporary password' /var/log/mysqld.log
```



단계5: mysql server 접속

```
mysql -u root -p # 임시 비밀번호 적용
```



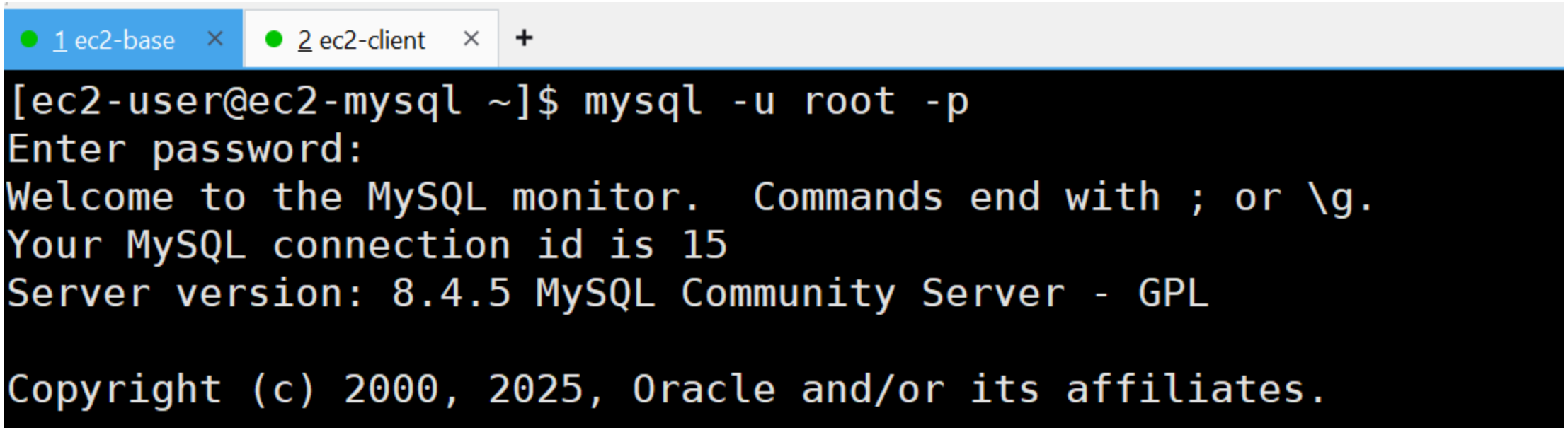
단계6: 비밀번호 변경하기

```
# 비밀번호 변경하기
ALTER user 'root'@'localhost' IDENTIFIED WITH caching_sha2_password BY 'Qwer1234!';
# 변경된 내용 적용하기
FLUSH PRIVILEGES;
# mysql server에서 나오기
exit
```

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

mysql> ALTER user 'root'@'localhost' IDENTIFIED WITH caching_sha2_password BY 'Qwer1234!';
Query OK, 0 rows affected (0.01 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.00 sec)

mysql> exit
Bye
[ec2-user@ec2-mysql ~]$
```

- 재접속

```
mysql -u root -p # 변경한 비밀번호 적용
```



- sql 테스트

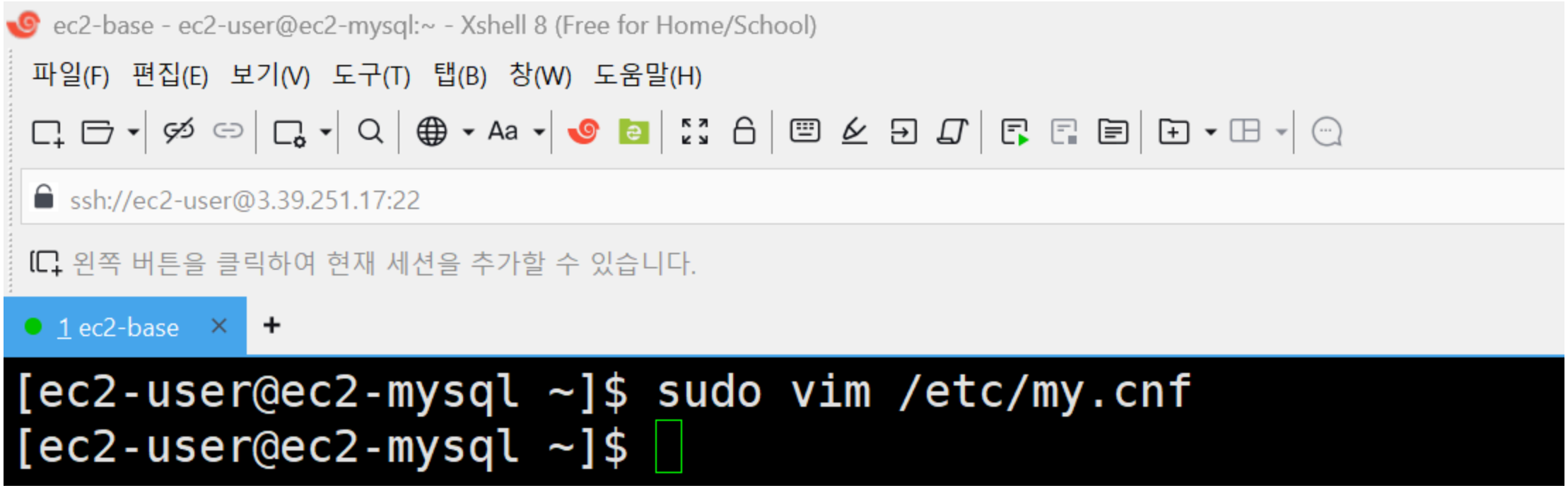
```
show databases;
exit
```

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

mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
4 rows in set (0.01 sec)

mysql> exit
Bye
[ec2-user@ec2-mysql ~]$
```

단계7: 외부아이피 허용 설정

- Ubuntu 계열: /etc/mysql/mysql.conf.d/mysqld.cnf
- 아마존 리눅스 등 CentOS 계열: /etc/my.cnf

```
sudo vim /etc/my.cnf
```

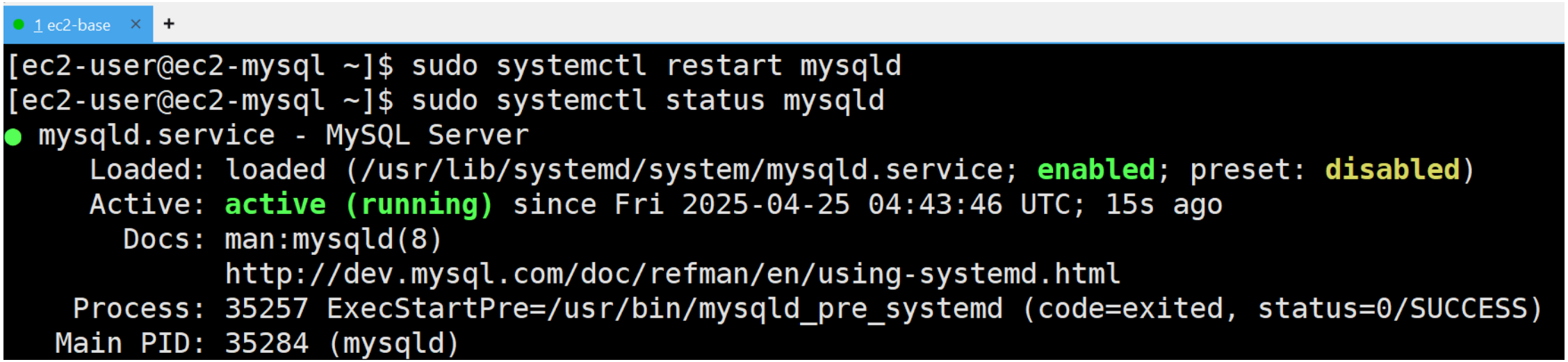


```
# 아래내용 입력
port            = 3306
bind-address            = 0.0.0.0
mysqlx-bind-address     = 0.0.0.0
```

```
pid-file=/var/run/mysqld/mysqld.pid

# 아 래  내 용  입 력
port                    = 3306
bind-address            = 0.0.0.0
mysqlx-bind-address     = 0.0.0.0
~
~
~
:wq!
```

단계8: mysql server 재실행

```
sudo systemctl restart mysqld
sudo systemctl status mysqld
```



단계9: 외부아이피 허용 확인

```
# 외부 아이피 허용 확인
sudo netstat -ntlp | grep mysqld
```

단계10: 아이피로 접속 가능한 계정 생성

- mysql server 접속

```
mysql -u root -p
```

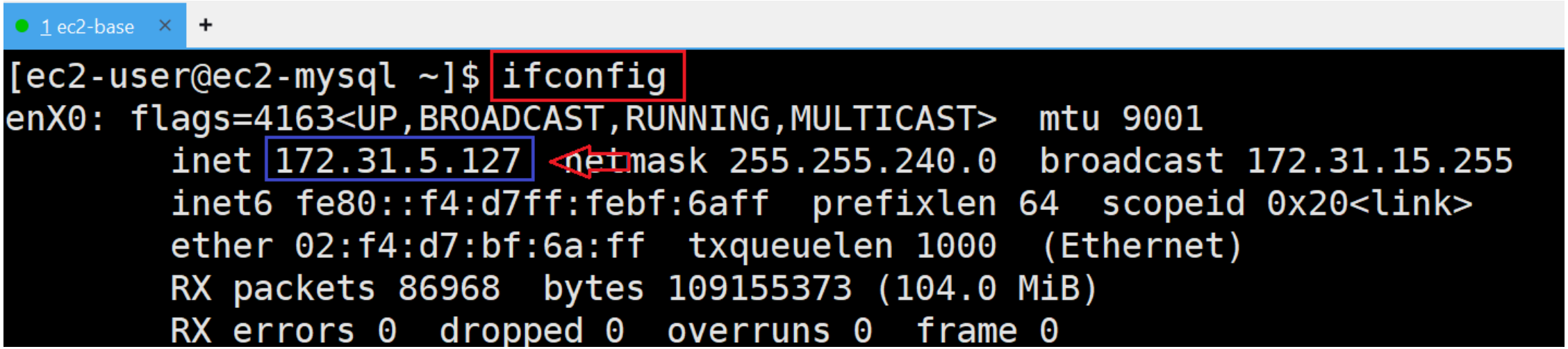
- 계정 생성

```
-- 유저 생성
create user 'ec2'@'%' identified by 'Qwer1234!';
-- 유저 조회
select user, host from mysql.user;
```

```
mysql> create user 'ec2'@'%' identified by 'Qwer1234!';
Query OK, 0 rows affected (0.02 sec)

mysql> select user, host from mysql.user;
+------------------+-----------+
| user             | host      |
+------------------+-----------+
| ec2              | %         |
| mysql.infoschema | localhost |
| mysql.session    | localhost |
| mysql.sys        | localhost |
| root             | localhost |
+------------------+-----------+
5 rows in set (0.00 sec)

mysql> exit
```

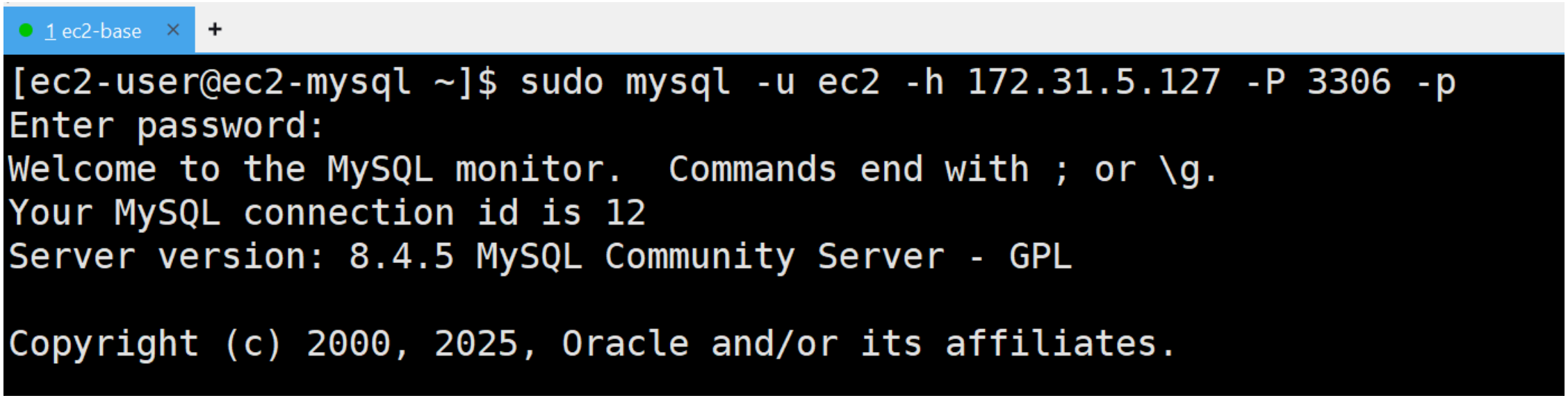
단계11: 내부 아이피 확인

```
# 내부 ip 확인
ifconfig
```



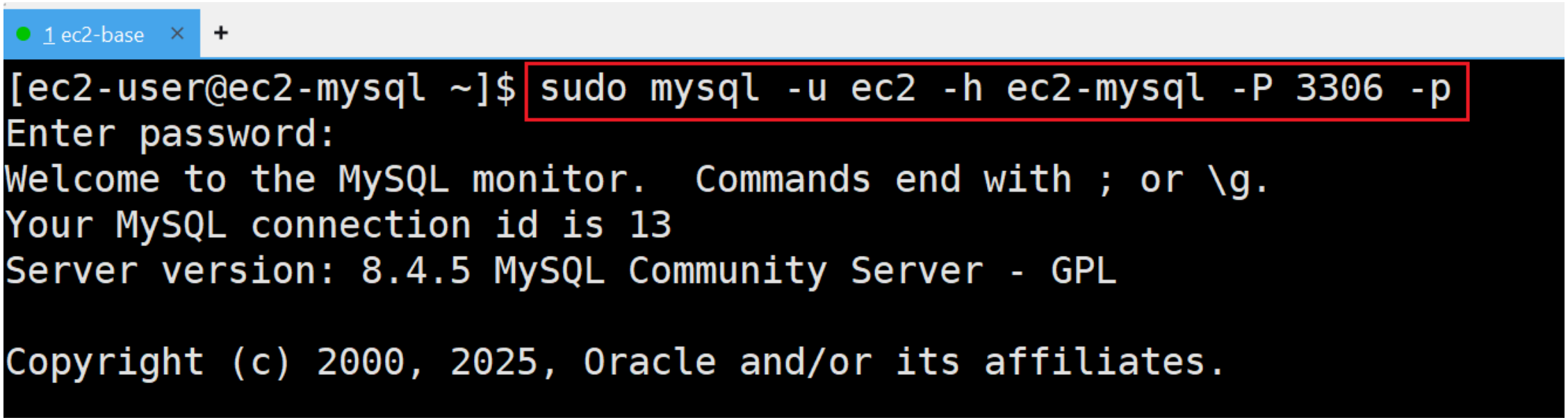
단계12: ip를 이용해서 ec2 계정으로 접속

```
# 내부 ip를 이용하여 접속
sudo mysql -u ec2 -h 172.31.5.127 -P 3306 -p
```



단계12: hostname을 이용해서 ec2 계정으로 접속

```
# 내부 ip를 이용하여 접속
sudo mysql -u ec2 -h ec2-mysql -P 3306 -p
```



단계13: 테스트

```
show databases;
```

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

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

mysql>
```