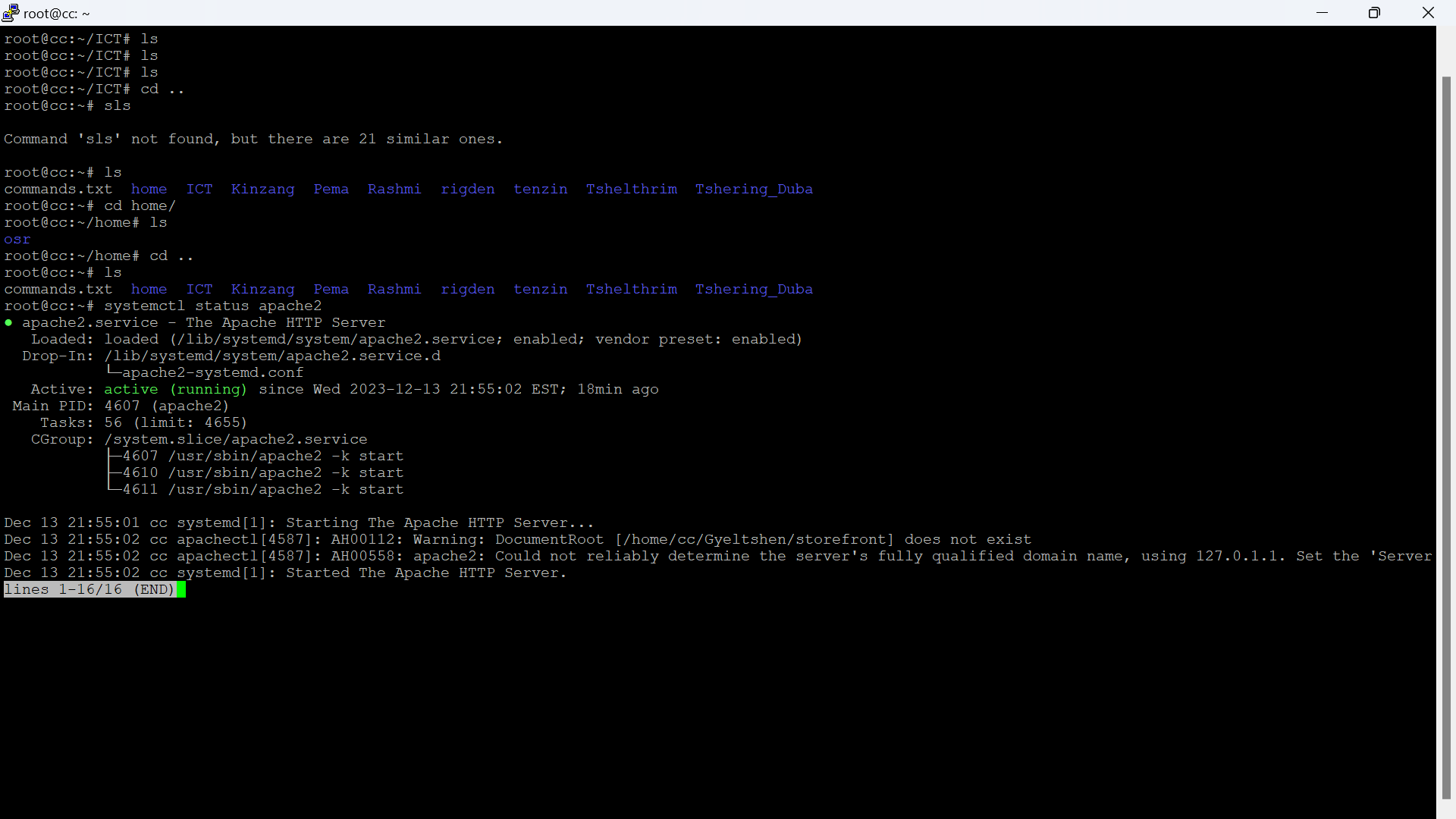
**How to Install Django with Apache on Ubuntu server?**

Step 1: sudo apt update && sudo apt upgrade –y (This commands updates the system to latest version)

Step 2: sudo apt install apache2 libapache2-mod-wsgi-py3 (This commands install the wsgi (web server gateyway interface). This wsgi helps to allow apache2 to work with python3)

Step 3: systemctl start apache2 (This commands is instructing the system to start the Apache web server service, making it actively listen for incoming web requests and serving content.)

Step 4: systemctl enable apache2 (you are telling the system to configure Apache to start automatically during the system boot process.)

Step 5: systemctl status apache2

## **Install MySQL and create a database**

Step 1: sudo apt install mysql-server libmysqlclient-dev (**libmysqlclient-dev**: This is another package you are installing. It contains development files and libraries that are necessary for compiling and linking programs with the MySQL client library.)

Step 2: systemctl start mysql

Step 3: systemctl enable mysql

Step 4: systemctl status mysql

Step 5: mysql -u root

mysql> CREATE DATABASE django\_db;

mysql> CREATE USER *'django\_user'*@'localhost' IDENTIFIED BY *'Pa$$word'*;

mysql> GRANT ALL ON *django\_db*.\* TO *'django\_user'*@'localhost';

mysql> FLUSH PRIVILEGES;

mysql> EXIT

Note: Italic words should replace by yourself.

## **Install Pip on Ubuntu 22.04**

Step 1: sudo apt install python3-venv python3-pip (pip3 –version)

## **Install Django Using Virtualenv**

Step 1: Create project directory using mkdir *foldername*. (You can create your project directory anywhere you want to create)

Step 2: Create your env inside your project directory using this commands: python3 -m venv *envfoldername.*

Step 3: Activate your virtual environment by using this commands: source *envfoldername*/bin/activate

Step 4: pip install django (django-admin –version)

Step 5: pip install mysqlclient(Optional)

## **Creating your Django project**

Note: Italic words should replace by yourself.

Step 1: django-admin startproject *django\_app* . (dot indicates the project is creating in same directory)

Step 2: nano django\_app/settings.py (This opens the settings.py file)

Step 3: ALLOWED\_HOSTS = ['your\_server\_ip', 'your-domain.com'] (Edit this line of code in settings.py file)

Step 4: DATABASES = {

'default': {

'ENGINE': 'django.db.backends.mysql',

'NAME': *'django\_db'*,

'USER': *'django\_user'*,

'PASSWORD': *'Pa$$word'*,

'HOST': '127.0.0.1',

'PORT' : '3306',

}

}

Step 5: import os

STATIC\_URL='*/static/*'

STATIC\_ROOT=os.path.join(BASE\_DIR, *'static/*')

MEDIA\_URL='*/media/*'

MEDIA\_ROOT=os.path.join(BASE\_DIR, *'media/*')

Note: Step 3-5 should edit in settings.py file.

Step 6: Save the file and exit.

Step 7: ./manage.py makemigrations(This commands make instance to migrate to db format)

Step 8: ./manage.py migrate(This commands migrate to db format)

Step 9: ./manage.py createsuperuser(This commands helps to create admin users)

Example:

Username (leave blank to use 'root'): **admin**

Email address: **admin@your-domain.com**

Password:

Password (again):

Superuser created successfully.

Step 10: ./manage.py collectstatic(For collecting all static files)

Note: Step 5-10 are optional.

Step 11: deactivate (This exits the env)

## **Configure Apache Web Server for Django**

Note: Italic words should replace by yourself.

Step 1: sudo nano /etc/apache2/sites-available/*django.conf*

Step 2: Enter the code as shown below.

<VirtualHost \*:80>

ServerAdmin *admin@your-domain.com*

ServerName *your-domain.com*

ServerAlias www.your-domain.com

DocumentRoot */var/www/django\_project/* (Project directory)

ErrorLog ${APACHE\_LOG\_DIR}/your-domain.com\_error.log

CustomLog ${APACHE\_LOG\_DIR}/your-domain.com\_access.log combined

Alias /static */var/www/django\_project/*static

<Directory */var/www/django\_project/*static>

Require all granted

</Directory>

Alias /media */var/www/django\_project/*media

<Directory */var/www/django\_project/*media>

Require all granted

</Directory>

<Directory */var/www/django\_project/django\_app*>

<Files wsgi.py>

Require all granted

</Files>

</Directory>

WSGIDaemonProcess *django\_app* python-path=*/var/www/django\_project* python-home=*/var/www/django\_project/django\_env*

WSGIProcessGroup *django\_app*

WSGIScriptAlias / */var/www/django\_project/django\_app/*wsgi.py

</VirtualHost>

Step 3: sudo nano /etc/apache2/sites-available/*Django-ssl.conf*

Step 4: Add the code as shown below:

Alias /static */var/www/django\_project/*static

<Directory */var/www/django\_project/*static>

Require all granted

</Directory>

Alias /media */var/www/django\_project/*media

<Directory */var/www/django\_project/*media>

Require all granted

</Directory>

<Directory */var/www/django\_project/django\_app*>

<Files wsgi.py>

Require all granted

</Files>

</Directory>

WSGIDaemonProcess *django\_app* python-path=*/var/www/django\_project* python-home=*/var/www/django\_project/django\_env*

WSGIProcessGroup *django\_app*

WSGIScriptAlias / */var/www/django\_project/django\_app/*wsgi.py

Step 5: sudo chown -R www-data:www-data */home/django\_project/* OR sudo chown -R www-data:www-data /home/ubuntu/second\_products/ OR sudo chmod -R 755 /home/ubuntu/second\_products/(This gives permission to execute the project by apache2 server)

Step 5: a2ensite *django.conf* (Make sure to enter into the location of the italic file)

Step 6: systemctl restart apache2

Step 7: Enter your server ip or domain whether your website is up or not

**Checking logs**

Cat –t /var/log/apache2/error.log

tail -n 50 /var/log/apache2/your-domain.com\_error.log

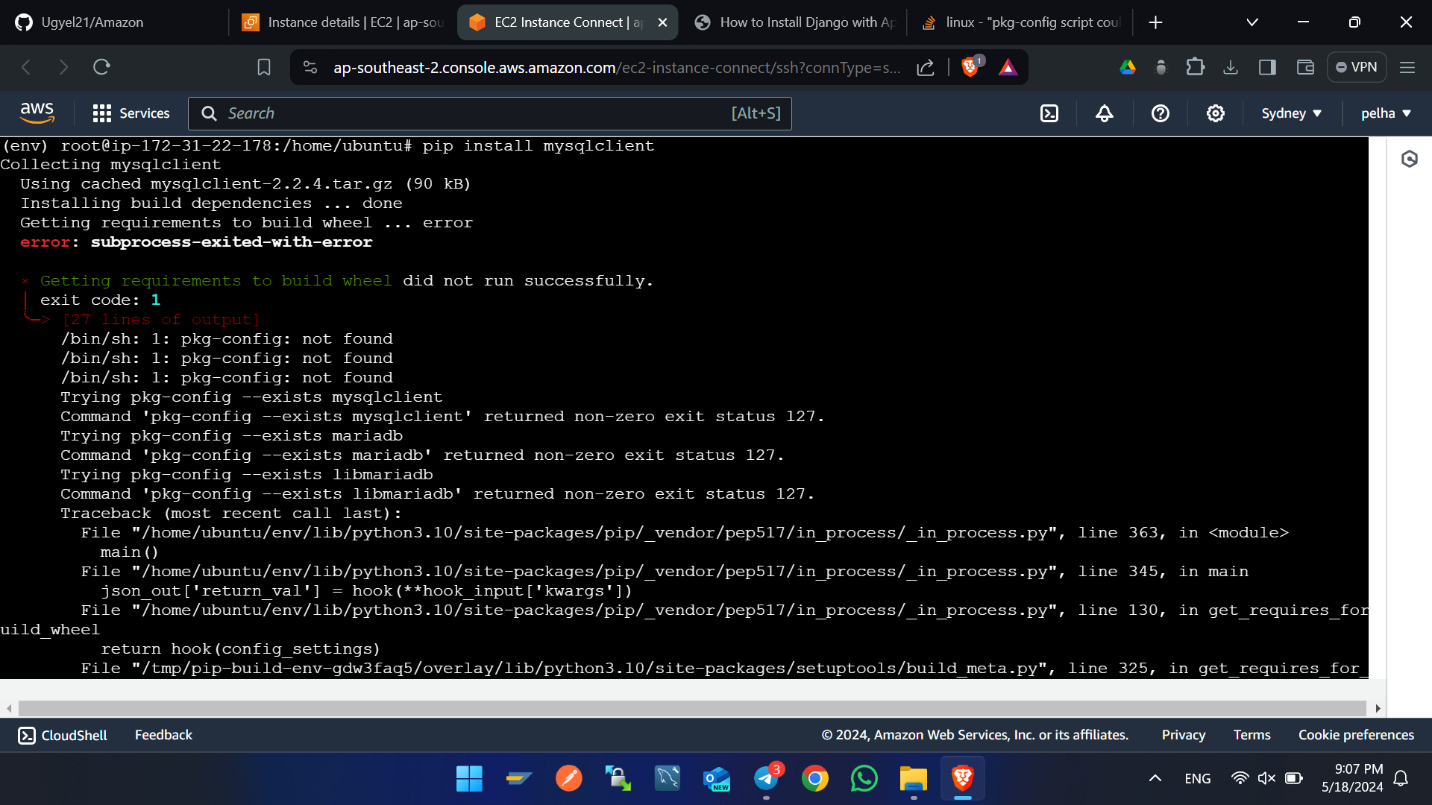
tail –f /var/log/apache2/error.log

tail –f /var/log/mysql/error.log

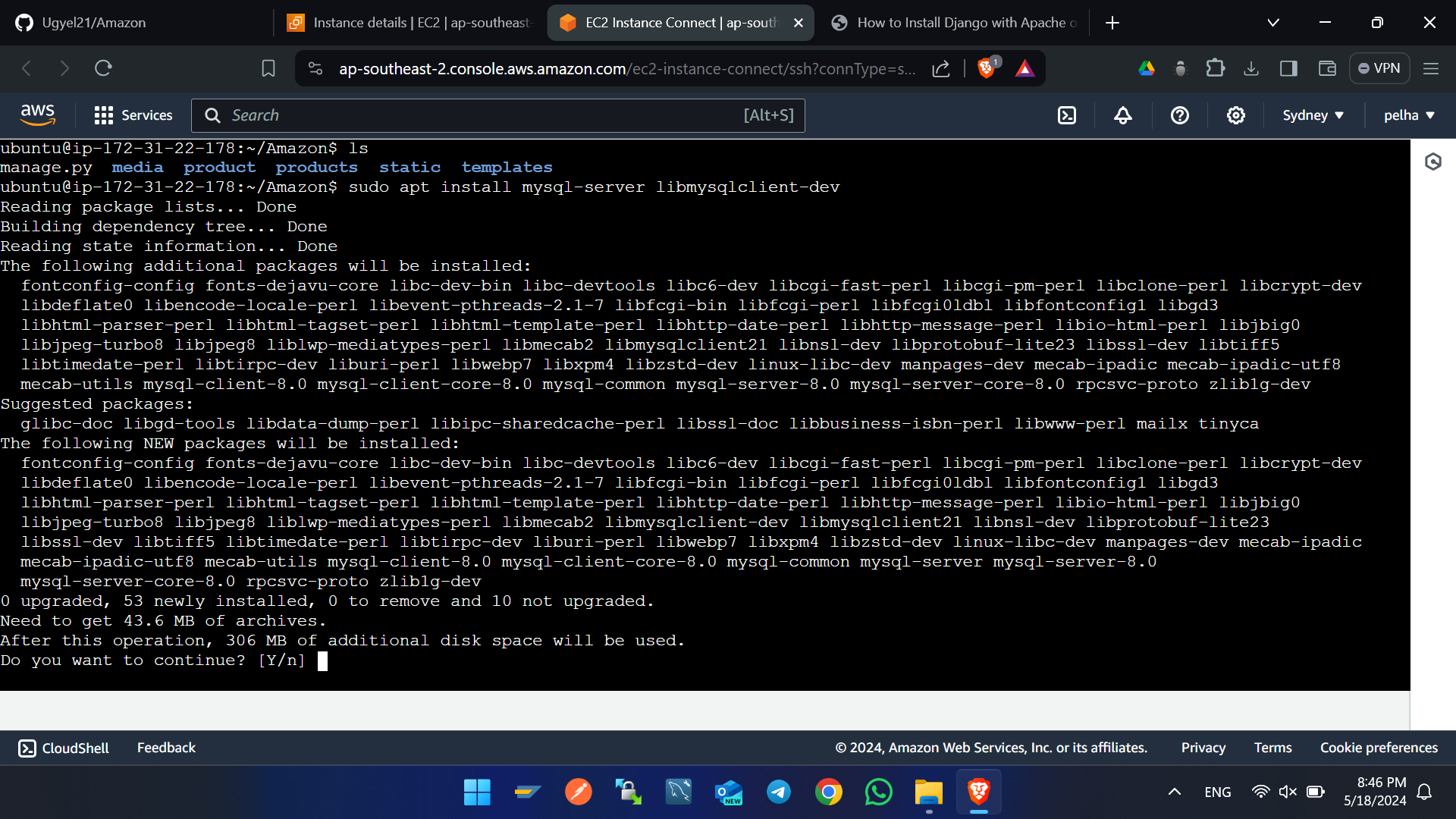
tail -f /var/log/syslog

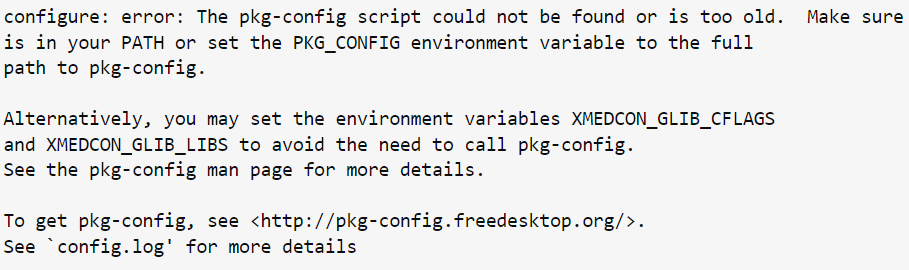
**Some of the issue face and solution**

Step a. If error occurs like below, the solution is given in step b.

****

Step b. sudo apt install mysql-server libmysqlclinet-dev

****

Error below

Solution: apt-get install -y pkg-config