Que hacer para ejecutar nuestro proyecto en un contenedor de Docker:  
Estructuración de carpetas

Proyecto/

├── ProyectoAnasisDatos/

├── \_\_init\_\_.py

├── settings.py

├── archivos/

├── manage.py

├── requirements.txt

├── Dockerfile

└── docker-compose.yml

## Dockerfile

# Dockerfile

FROM python:3.10

ENV PYTHONDONTWRITEBYTECODE 1

ENV PYTHONUNBUFFERED 1

WORKDIR /app

COPY requirements.txt .

RUN pip install --upgrade pip && pip install -r requirements.txt

COPY . .

CMD ["python", "manage.py", "runserver", "0.0.0.0:8000"]

## Requirements.txt

pip freeze > requirements.txt

//esto deja todas las librerías y frameworks previamente descargados, si no se tiene

pip install django pandas psycopg2-binary

## Docker-compose.yml

version: '3.9'

services:

web:

build: .

command: python manage.py runserver 0.0.0.0:8000

ports:

- "8000:8000"

volumes:

- .:/app

depends\_on:

- db

db:

image: postgres:15

environment:

POSTGRES\_DB: nombre\_db

POSTGRES\_USER: usuario

POSTGRES\_PASSWORD: contraseña

volumes:

- pgdata:/var/lib/postgresql/data

volumes:

pgdata:

//Aca la idea es ir mirando a postgres y eso, ya que se debe visualizar en las próximas entregas la descarga y creación de usuario y contraseña del propio postgre para su uso, ya que es necesario por esto mismo para la conexión postgre tiene varios requerimientos previos fuera del código para su uso correcto y para la implementación en contenedor se debe tener esto, la librería psycopg2 y

## Settings.py

DATABASES = {

'default': {

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

'NAME': 'nombre\_db',

'USER': 'usuario',

'PASSWORD': 'contraseña',

'HOST': 'db',

'PORT': 5432,

}

}

Ya cuando se tiene todo listo con Docker y su extensión Docker-compose descargada y todos los archivos correctos solo falta hacer

docker-compose up –build

docker-compose up --build -d (modo daemon)

este lee el archivo Docker-compose.yml y genera los contenedores que aparecen en los encabezados del archivo (ósea 2)