Ingesta datos Covid19

EL objetivo principal de este manual es generar los pasos necesarios para que el usuario pueda ejecutar el proyecto sin problema alguno.

Requisitos

Para poder ejecutar este proyecto necesitamos las siguientes herramientas instaladas en nuestro equipo de cómputo.

- Python
- Visual Studio code
- DBeaver
- Docker

Pasos de ejecución

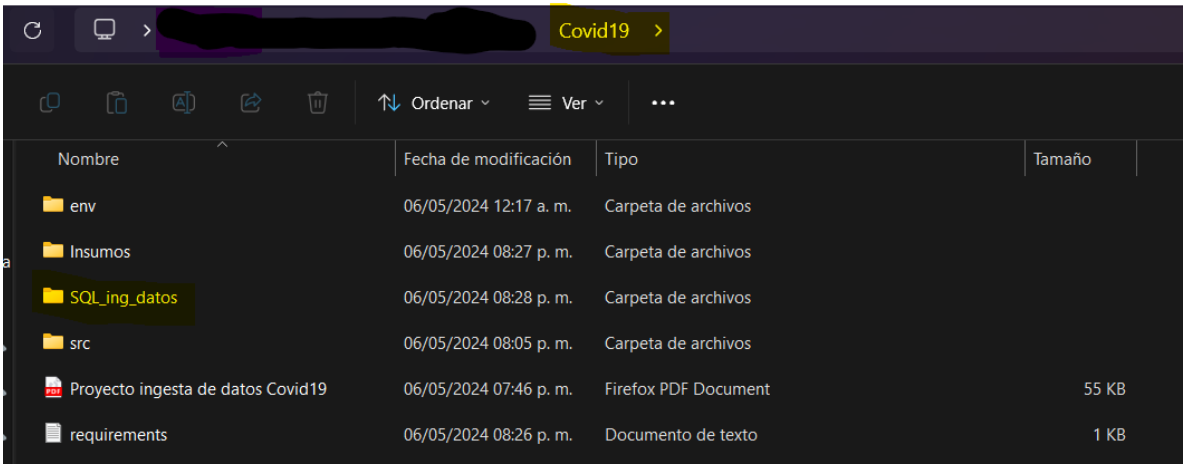
El primer paso para poder ejecutar este proyecto es descargar las imágenes para poder usar PostgreSQL desde Docker, por lo que abrimos el CMD y ejecutamos las siguientes instrucciones (Nota esto debemos tener instalado Docker).

- **docker pull postgres:15.3**
- **docker pull dpage/pgadmin4**

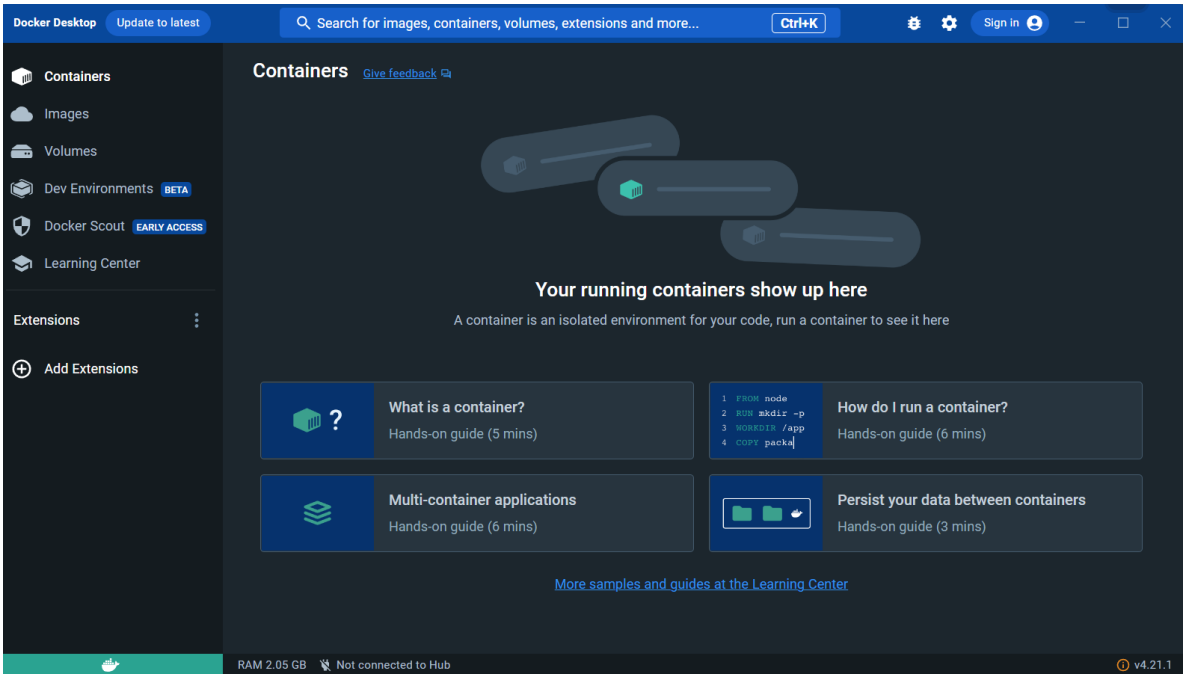
una vez que terminen de descargar las imágenes, el siguiente paso es crear una carpeta en donde guardaremos todo el proyecto.

Una vez dentro de esa carpeta, crearemos una nueva carpeta y depositaremos el archivo **docker-compose.yml** el cual contiene toda la configuración para poder usar PostgreSQL

Nota: En mi caso mi carpeta principal es Covid19 y la segunda carpeta creada es SQL_ing_datos.

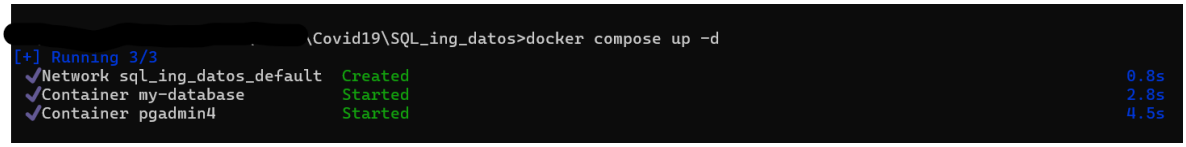


El siguiente paso es abrir Docker, una vez abierto se ve de la siguiente manera



A continuación, abriremos nuevamente el CMD e ingresaremos a nuestra carpeta creada donde se encuentra el **docker-compose.yml** y ejecutaremos el comando

- docker compose up -d



Con esto levantamos postgresql, es importante aclarar que mientras estemos realizando la ingesta de la informacion Docker siempre tiene que permanecer abierto para que funcione.

Nota: cuando terminemos de usar docker, para darlo de baja tendremos que usar el comando docker compose down

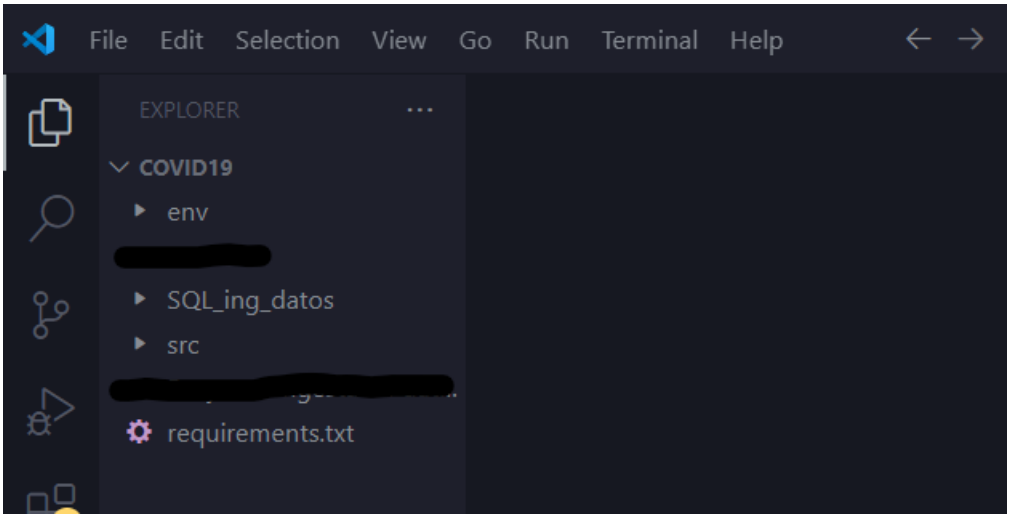
Continuando la ejecución del proyecto, abriremos visual studio code y crearemos entorno virtual dentro de nuestra carpeta donde estarán los scripts

Nota: En nuestro caso usamos virtualenv para crear este entorno virtual para instalarlo necesitamos como primer paso tener Python instalado, enseguida ejecutar el comando en la termina que nos brinda visual studio, **pip install virtualenv** una vez instalada esta librería ejecutaremos nuevamente en la terminal el comando **virtualenv env** con esto se genera nuestro entorno virtual.

Para activar el entorno virtual, se ejecuta el comando **.\env\Scripts\activate** esto nos activara nuestro entorno virtual a continuación ejecutaremos el siguiente comando para instalar las librerías que se usaran en el proyecto **pip install -r requirements.txt** para poder ejecutar esto con éxito el archivo **requirements** debe de estar dentro de la carpeta principal.

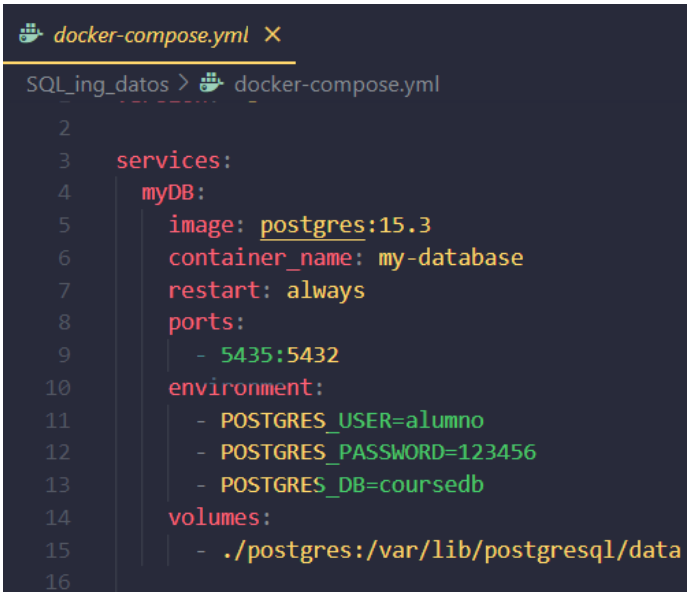
Después de esto, creamos una carpeta llamada donde colocaremos los scripts (de igual forma dentro de nuestra carpeta principal)

Hasta el momento nuestra carpeta principal debe de verse de la siguiente forma



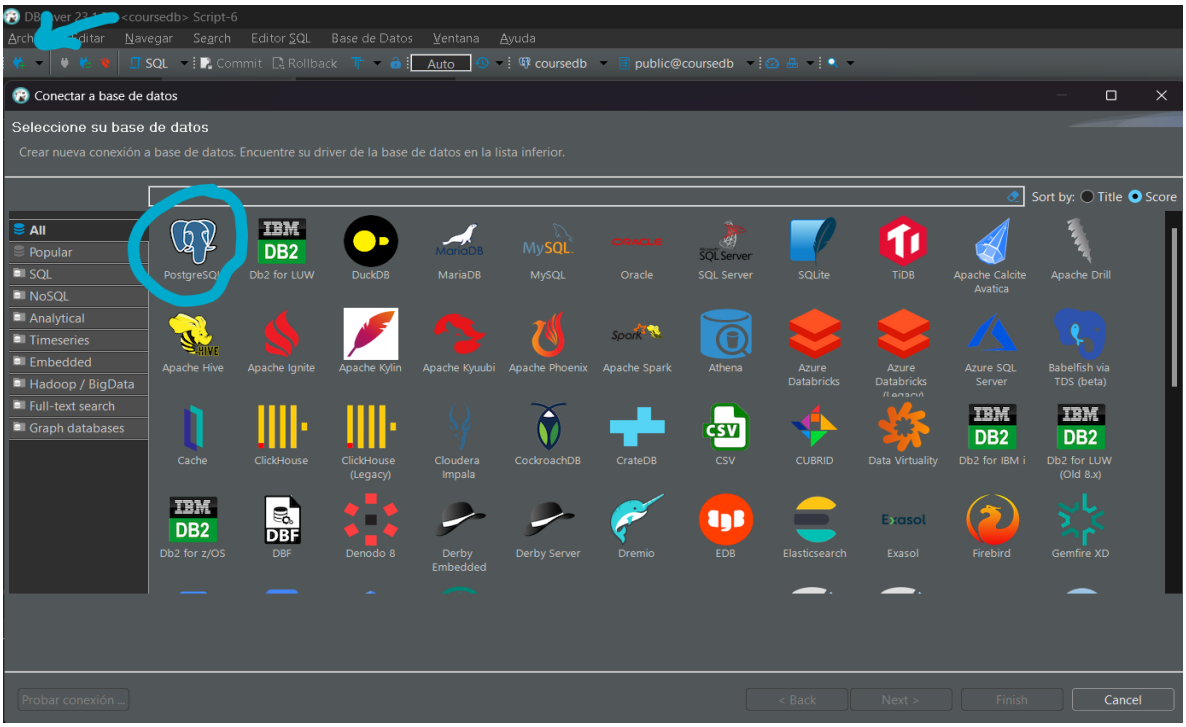
Antes de continuar tenemos que crear las tablas donde se ingestaran los datos, por lo que abriremos DBeaver y realizaremos la conexión para poder crear las tablas de interés (así mismo encontraras el archivo para crear las tablas en el repositorio).

Para establecer la conexión, debemos de abrir el archivo **docker-compose.yml** y ubicaremos la siguiente información



ADVERTENCIA: En cualquier otra situación estos parámetros son estrictamente confidenciales ya que pueden comprometer los datos. Este paso solo se realiza por dos razones por esta ocasión son parámetros que no comprometen de mucho la información y para la facilitar la ejecución del proyecto.

Para establecer la conexión, hacemos click donde esta la flecha y escogemos en este caso a PostgreSQL



A continuación, aparece el cuadro para ingresar los datos, previamente presentados.

Nota: Host se mantiene como localhost, los otros parámetros database, port,nombre usuario y contraseña se cambian por los valores presentados previamente

Después creamos un nuevo scrip y colocamos las querys para crear las tablas (recordando estos querys se encuentran dentro de los archivos del repositorio.

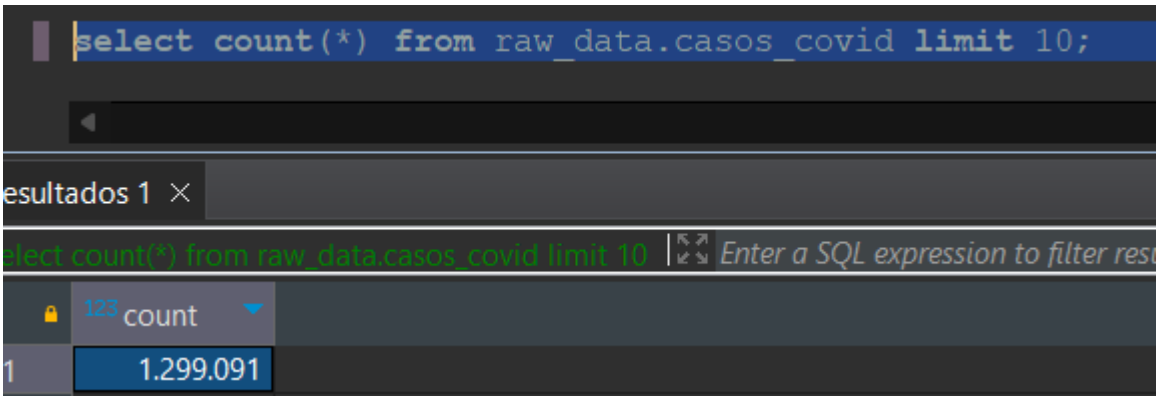
Una vez creadas estas tablas regresamos a visual studio code y ejecutamos los scripts en el siguiente orden:

- extract_data.py: Crea la carpeta donde se depositaran los archivos y descomprimirá estos mismos para poder usarlos.
- modified_cat_file.py: modifica ciertos catálogos ya que no todos tienen el mismo formato
- load_cat.py: Realiza la carga de datos de los catálogos a PostgreSQL
- load_estados.py: Realiza la carga de los datos de los catálogos entidades y municipios.
- load_covid_data.py: Realiza la carga de los datos relacionados al covid19

el tiempo de ejecución de estos Scripts es relativamente rápido, en especial los scripts

extract_data.py, modified_cat_file.py, load_cat.py, load_estados.py ya que no tardan mas de 3 minutos en copiar la información, caso contrario al script **load_covid_data.py** este tarda alrededor de 1 hora 40 min, por la cantidad de datos ya que son un poco mas de 1,000,000 de registros.

Una vez cargada la informacion podemos ir a nuestro DBeaver y ejecutar una consulta a cualquier tabla y veremos información dentro de las tablas.



Por último, para salir del entorno virtual usamos deactivate.

Ahora ya podemos manipular la información directamente desde Dbeaver. el cual resulta bastante cómodo pues ya no tenemos que estar cargando el archivo csv.