

Universidad
Industrial de
Santander



Ingeniería de Sistemas

Trabajo Final

Felix Adolfo Nieto Rangel
Henry Andres Jiménez Herrera
Bases de Datos II

Índice

1) Actividad 1.....	3
2) Actividad 2.....	4
3) Actividad 3.....	5
3.1) Migración Manual.....	5
3.2) Migración Programada.....	7
4) Referencias.....	9

Desarrollo

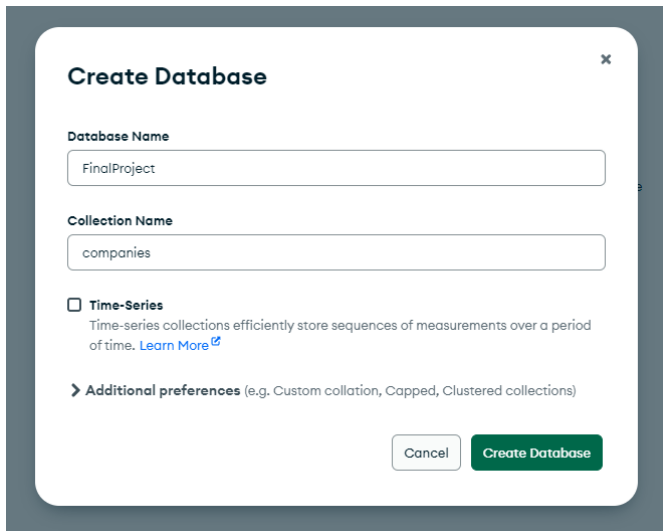
Enlace Del Proyecto Y Los Recursos: <https://github.com/FelixD02/Proyecto-Bases-de-Datos-II.git>

Archivo CSV Seleccionado: Datos de Compañías

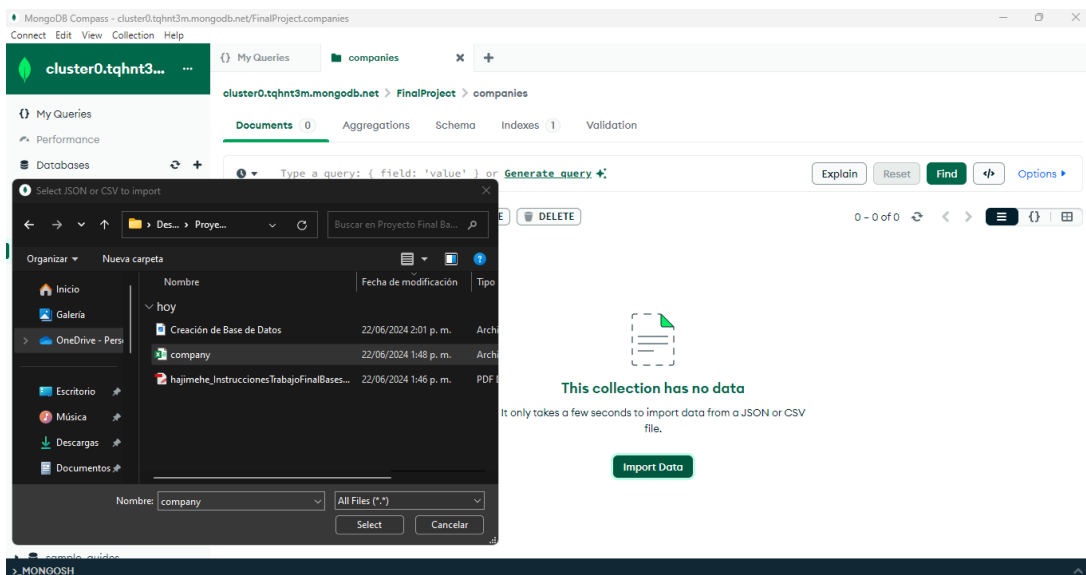
Base de Datos Seleccionada: MongoDB y PostgreSQL

ACTIVIDAD 1 – CARGA INICIAL:

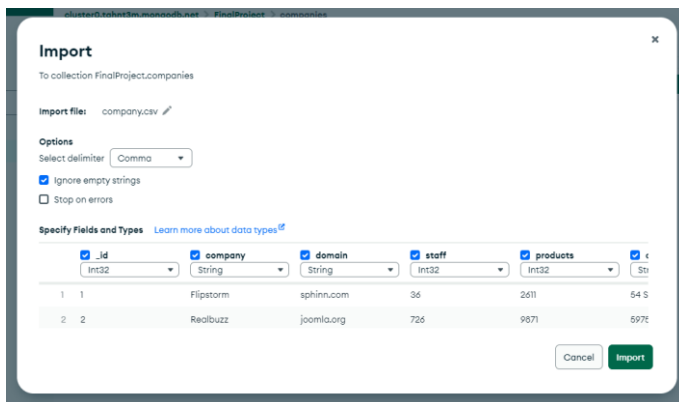
Creamos una Base de Datos y una Colección en MongoDB:



Seleccionamos el Archivo CSV que deseamos importar en nuestra colección:



Especificamos los campos que deseamos importar:

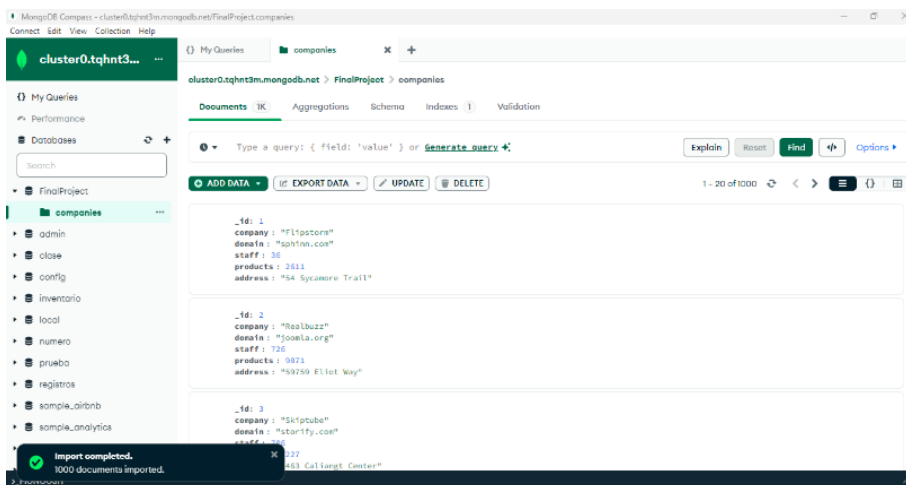


The screenshot shows the MongoDB Import Wizard for the 'FinalProject.companies' collection. The 'Import file' is 'company.csv'. Under 'Options', 'Select delimiter' is set to 'Comma', and 'Ignore empty strings' is checked. The 'Specify Fields and Types' section shows a table with columns and their corresponding data types from the CSV file.

	<input checked="" type="checkbox"/> _id	<input checked="" type="checkbox"/> company	<input checked="" type="checkbox"/> domain	<input checked="" type="checkbox"/> staff	<input checked="" type="checkbox"/> products	<input checked="" type="checkbox"/> address
	Int32	String	String	Int32	Int32	String
1	1	Flipstorm	sphinxn.com	36	2611	54 Sycamore Trail
2	2	Realbuzz	joomla.org	726	9871	59750 Elliot Way

Buttons: Cancel, Import

Comprobamos la importación de los registros en la interfaz de MongoDB:



Hacemos una consulta que cuente la cantidad de registros cargados:

```
> _MONGOSSH
> db.companies.countDocuments()
< 1000
Atlas atlas-289bmy-shard-0 [primary] FinalProject>
```

ACTIVIDAD 2:

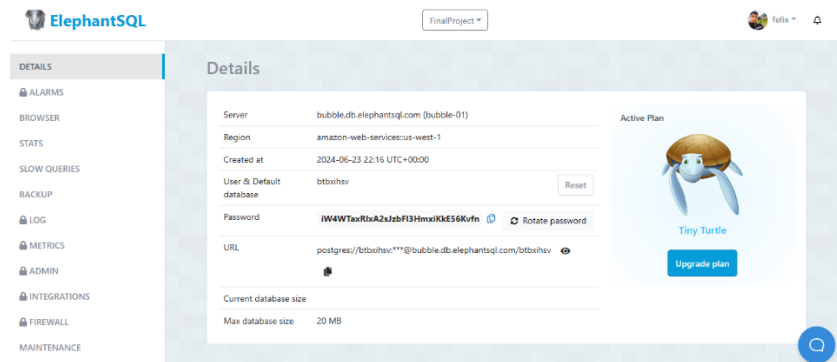
Eliminamos todas las compañías con menos de 500 productos:

```
> _MONGOSSH
> db.companies.deleteMany({products: {$lt:500}})
< {
  acknowledged: true,
  deletedCount: 52
}
> db.companies.countDocuments()
< 948
Atlas atlas-289bmy-shard-0 [primary] FinalProject>
```

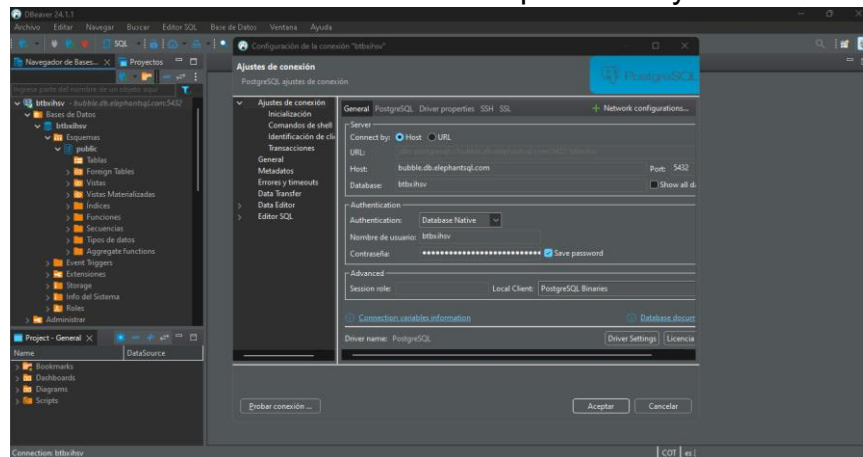
ACTIVIDAD 3:

A) Migración Manual:

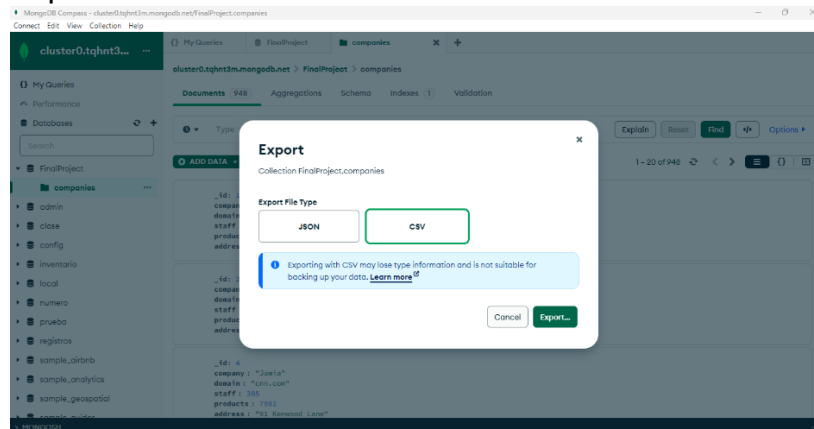
Creamos una Instancia en ElephantSQL:



Establecemos la conexión entre ElephantSQL y DBeaver:

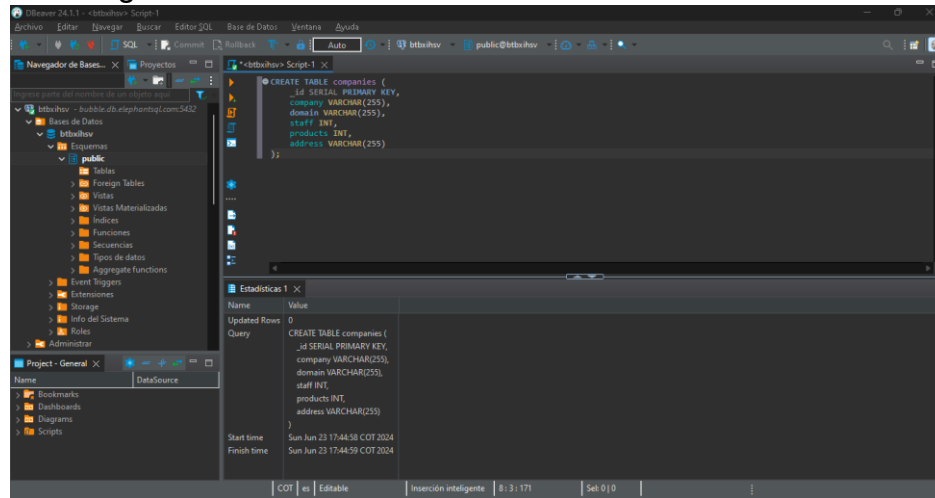


Exportamos nuestros datos en un archivo CSV desde MongoDB:

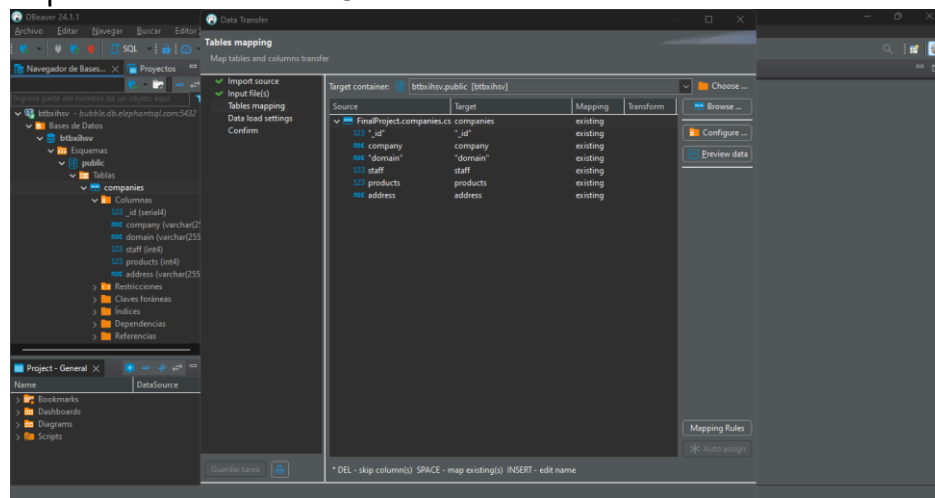


LINK: <https://github.com/FelixD02/Proyecto-Bases-de-Datos-II/blob/7ff700331341a7832effb5fbaddc5910f6ac5fb0/FinalProject.companies.csv>

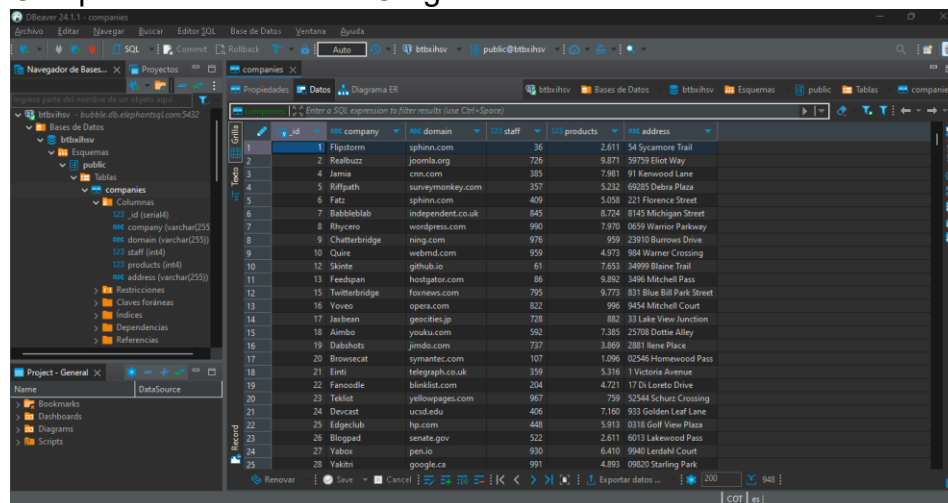
Creamos la tabla 'companies' en PostgreSQL con la misma estructura que en MongoDB:



Importamos el archivo CVS en DBeaver:

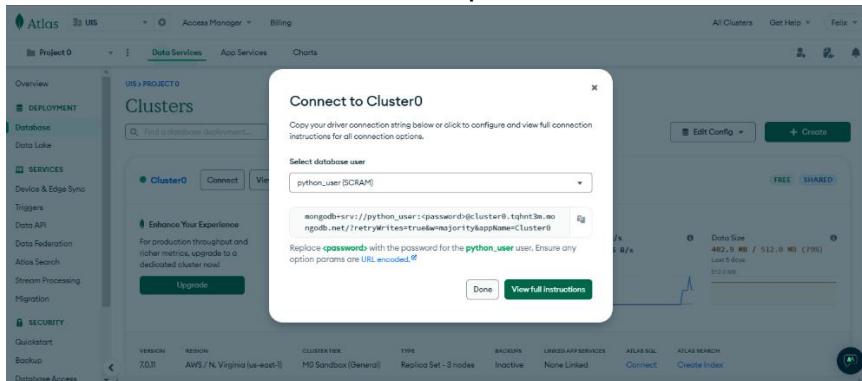


Comprobamos los Datos Cargados de Manera Manual:

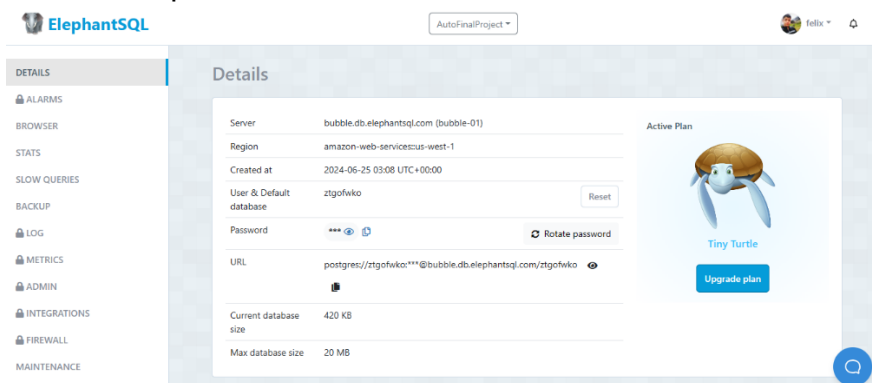


B) Migración Mediante Programación:

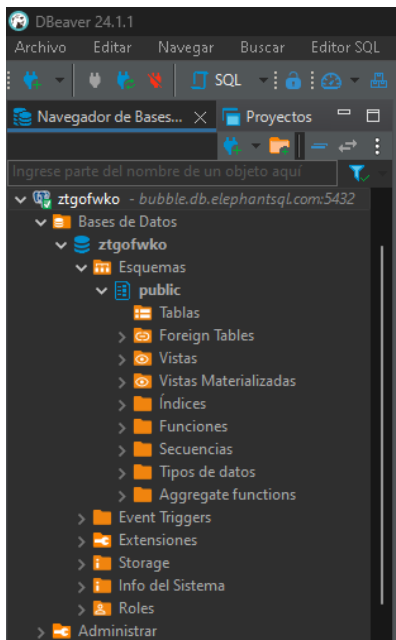
Generamos el enlace necesario para conectarnos al Cluster de Atlas:



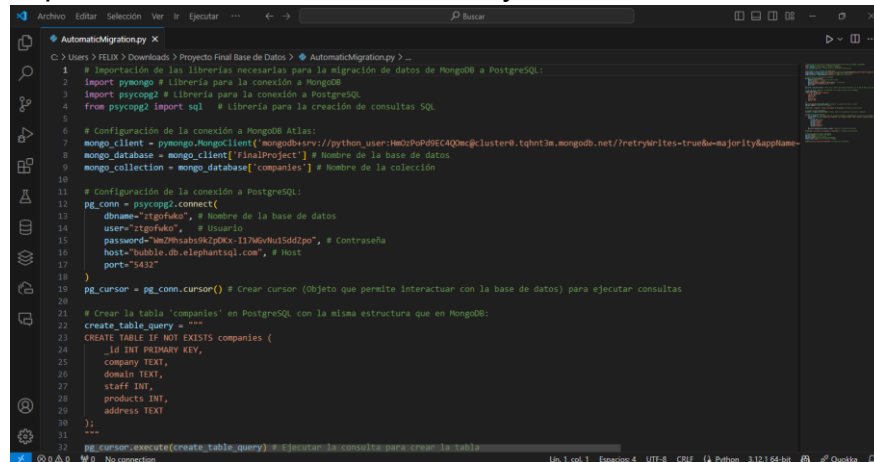
Creamos una nueva Base de Datos en PostgreSQL y tomamos los datos necesarios para conectarnos:



Conectamos nuestra Base de Datos PostgreSQL con DBeaver:



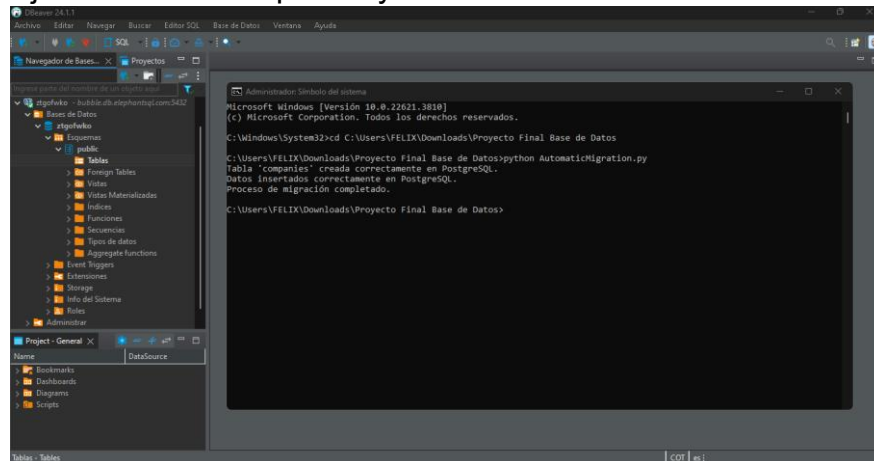
Programamos un Script en Python para crear un enlace entre nuestras bases de datos tanto en MongoDB como en PostgreSQL y que tenga la capacidad de leerlas, modificarlas y actualizarlas:



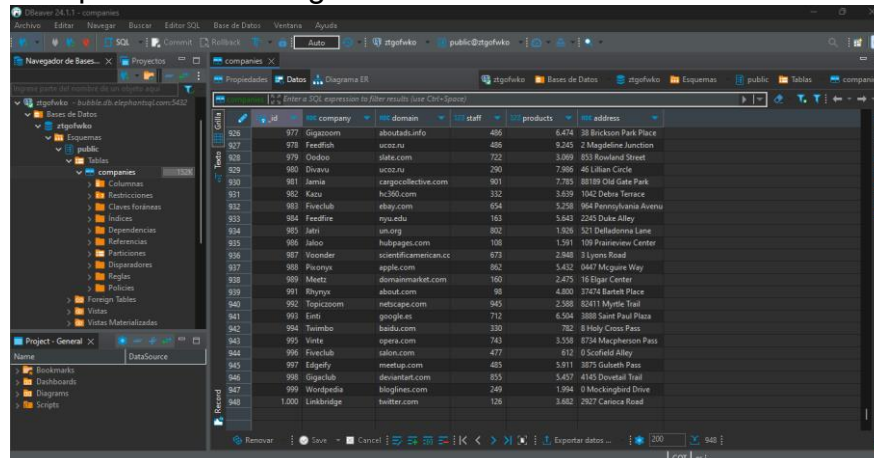
```
1 # Importación de las librerías necesarias para la migración de datos de MongoDB a PostgreSQL:
2 import pymongo # Librería para la conexión a MongoDB
3 import psycopg2 # Librería para la conexión a PostgreSQL
4 from psycopg2 import sql # Librería para la creación de consultas SQL
5
6 # Configuración de la conexión a MongoDB Atlas:
7 mongo_client = pymongo.MongoClient('mongodb+srv://python_user:1M0p4pH0EC4Q0w@cluster0.tqht3m.mongodb.net/?retryWrites=true&majorityAppname=python')
8 mongo_database = mongo_client['FinalProject'] # Nombre de la base de datos
9 mongo_collection = mongo_database['companies'] # Nombre de la colección
10
11 # Configuración de la conexión a PostgreSQL:
12 pg_conn = psycopg2.connect(
13     dbname='itgofuko', # Nombre de la base de datos
14     user='itgofuko', # Usuario
15     password='h4n2w-abrh2k0x-1176vHu15dd2p0', # Contraseña
16     host='bubble.db.elephantsql.com', # Host
17     port='5432'
18 )
19 pg_cursor = pg_conn.cursor() # Crear cursor (Objeto que permite interactuar con la base de datos) para ejecutar consultas
20
21 # Crear la tabla 'companies' en PostgreSQL con la misma estructura que en MongoDB:
22 create_table_query = """
23 CREATE TABLE IF NOT EXISTS companies (
24     id INT PRIMARY KEY,
25     company TEXT,
26     domain TEXT,
27     staff INT,
28     products INT,
29     address TEXT
30 );
31 """
32 pg_cursor.execute(create_table_query) # Ejecutar la consulta para crear la tabla
```

LINK: <https://github.com/FelixD02/Proyecto-Bases-de-Datos-II/blob/7ff700331341a7832effb5fbaddc5910f6ac5fb0/AutomaticMigration.py>

Ejecutamos el Script de Python a través del CMD en Windows:



Comprobamos la carga de datos en DBeaver:



Referencias

Psycopg – PostgreSQL database adapter for Python — Psycopg 2.9.9 documentation. (s/f). Psycopg.org. Recuperado el 27 de junio de 2024, de <https://www.psycopg.org/docs/>

The psycopg2 module content — Psycopg 2.9.9 documentation. (s/f). Psycopg.org. Recuperado el 27 de junio de 2024, de <https://www.psycopg.org/docs/module.html>

PyMongo 4.8.0 documentation. (s/f). Readthedocs.io. Recuperado el 27 de junio de 2024, de <https://pymongo.readthedocs.io/en/stable/>

Mohamed, A. (2023, agosto 28). *Migrating schema from MongoDB to PostgreSQL: A practical example*. DEV Community. <https://dev.to/ahmedmohamed/migrating-schema-from-mongodb-to-postgresql-a-practical-example-52lk>

MongoDB documentation. (s/f). Mongodb.com; MongoDB. Recuperado el 27 de junio de 2024, de <https://www.mongodb.com/docs/>

Documentation. (s/f). ElephantSQL. Recuperado el 27 de junio de 2024, de <https://www.elephantsql.com/docs/index.html>

Documentation. (s/f-b). Postgresql.org. Recuperado el 27 de junio de 2024, de <https://www.postgresql.org/docs/>