

A decorative graphic on the right side of the page featuring three concentric blue circles of varying sizes. Two thin blue lines originate from the top left and extend diagonally towards the circles. A large, partially visible concentric blue circle is at the bottom right.

Grupo #24

200915609	Gary Joan Ortiz López
201020247	Haroldo Pablo Arias Molina
201408486	Sharolin Guadalupe Lacunza González

PRACTICA 3 – BASES DE DATOS 2

Historial de Versiones

Fecha	Versión	Descripción
22/04/2021	1.0	Manual de Practica 3

Configuración del Ambiente

Instalación de MongoDB

Para esta configuración utilizamos Docker y docker compose, por lo cual configuramos el siguiente archivo yaml:

1. Archivo docker-compose.yml

```
1  version: "3.8"
2  services:
3    mongodb:
4      image : mongo
5      container_name: mongodb
6      environment:
7        - PUID=1000
8        - PGID=1000
9      volumes:
10     - /mongodb/database:/data/db
11     ports:
12     - 27017:27017
13     restart: always
```

En donde:

1. Se utilizó la imagen de mongo y se obtiene la última versión por default.
2. El uso de PUID y PGID permite que nuestros contenedores mapeen el usuario interno del contenedor a un usuario en la máquina host.
3. Se crea un almacenamiento persistente dentro de nuestra maquina con el contenedor.
4. Se exporta su puerto default.
5. Se configura que se inicie automáticamente el contenedor siempre que el servidor se reinicie.

2. Se corre el archivo con el siguiente comando

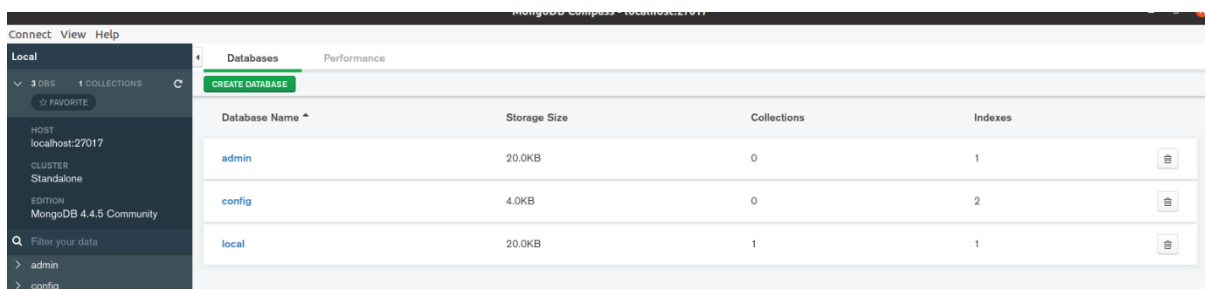
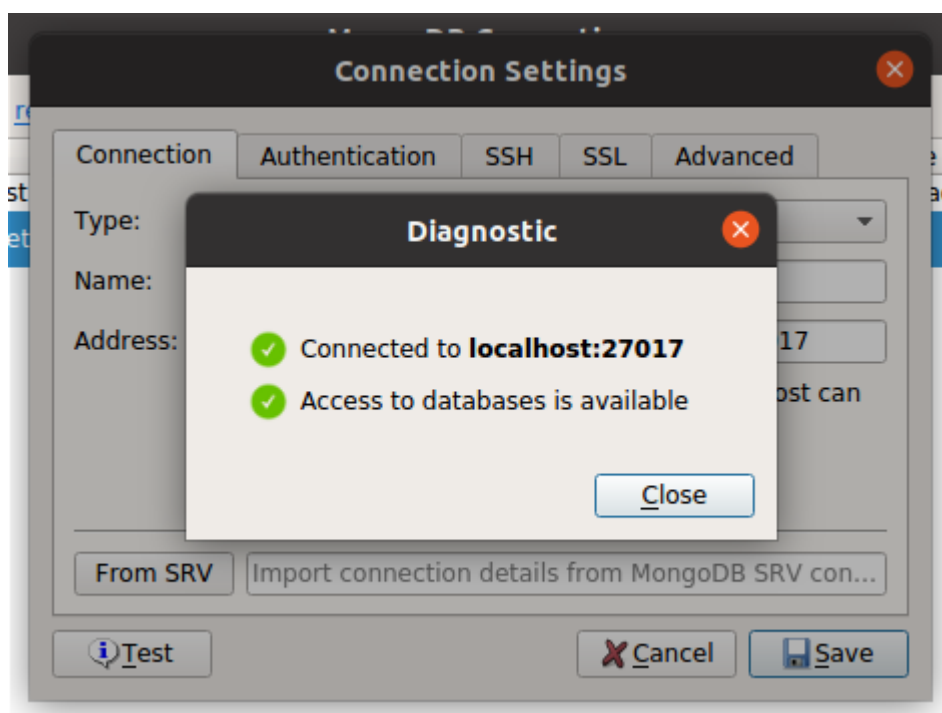
docker-compose up -d

```
root@harias-vm:/home/harias/Escritorio# docker-compose up -d
Creating mongodb-practica ... done
root@harias-vm:/home/harias/Escritorio#
```

3. Se verifica que el contenedor esté funcionando

```
root@harias-vm:/home/harias/Escritorio# docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS
1792787a6b36   mongo    "docker-entrypoint.s..." 59 seconds ago Up 57 seconds
root@harias-vm:/home/harias/Escritorio#
```

4. Se valida ingresar a la base de datos con el usuario default.



Configuración y Carga de Base de Datos

Se procede a crear una base de datos y su collections

Create Database

Database Name

practica3

Collection Name

patents_grupo24

☐ Capped Collection ⓘ

☐ Use Custom Collation ⓘ

Before MongoDB can save your new database, a collection name must also be specified at the time of creation. [More Information](#)

CANCEL

CREATE DATABASE

Create Collection

Collection Name

countries_grupo24

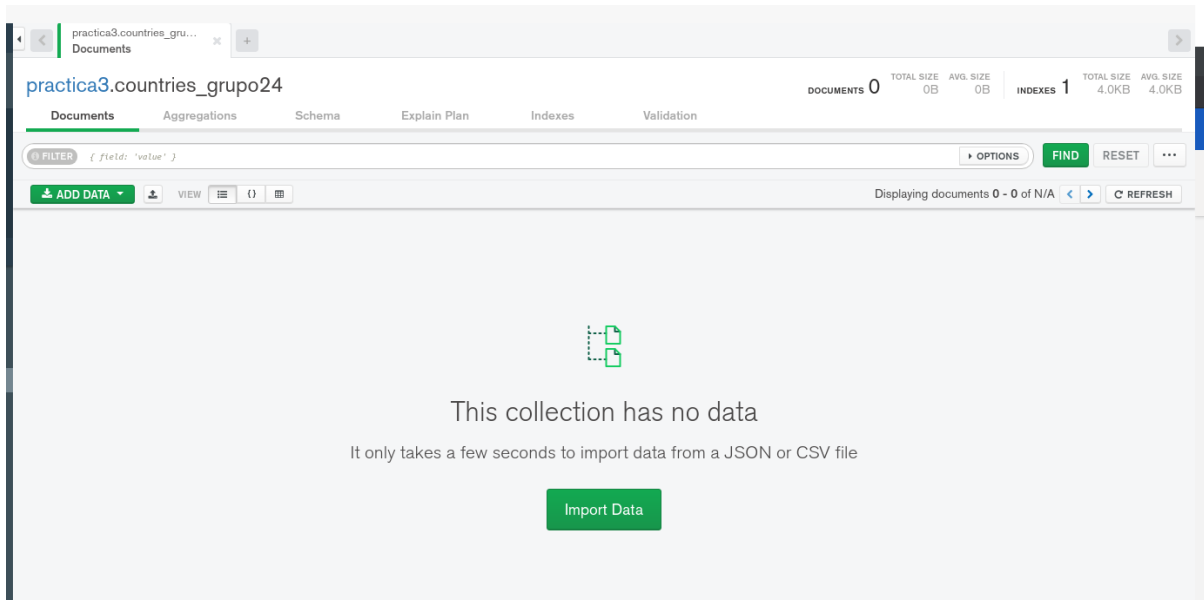
☐ Capped Collection ⓘ

☐ Use Custom Collation ⓘ

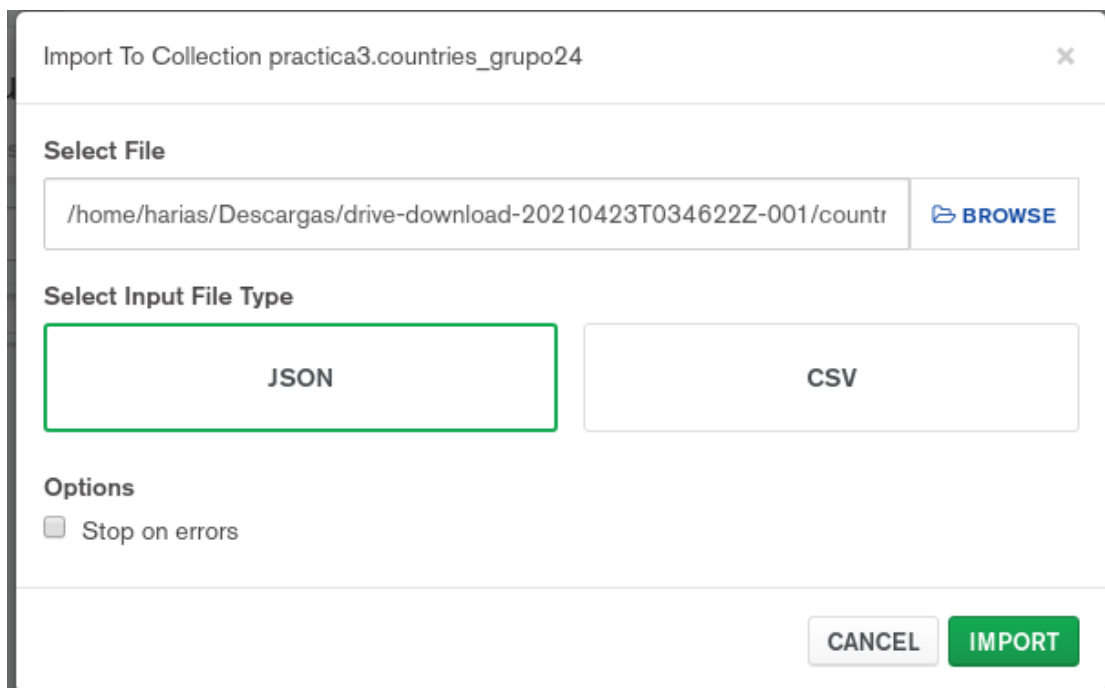
CANCEL

CREATE COLLECTION

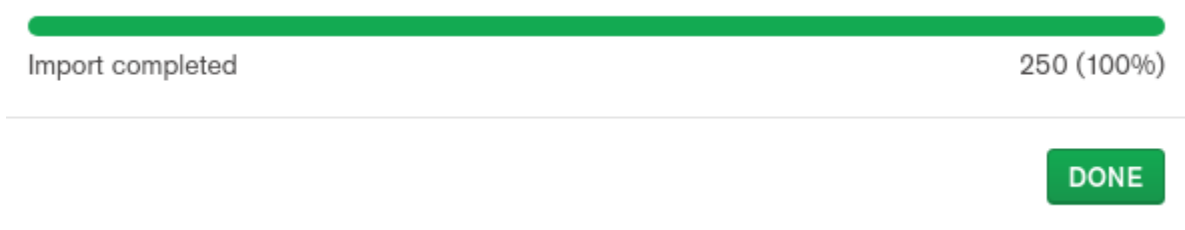
Se utilizó la herramienta MongoDB Compass para la carga de la información por lo que, se seleccionó el collection correspondiente y se cargó su respectivo archivo.



Se selecciona el archivo y se establece que es formato JSON.



Y nos mostrará un porcentaje de finalizado.



Se debe de realizar el mismo procedimiento para el otro archivo en su collection correspondiente.

Import completed

550 (100%)

DONE

COMANDOS MAS IMPORTANTES PARA REALIZAR LAS CONSULTAS

```
db.patents_grupo24.aggregate([  
  
  {$group:  
  
    { _id:"$patent_year", contador:{$sum:1}}},  
  
  {$sort:  
    {contador:-1}}  
  
]);
```

Definición de estructura de la consulta:

1. **Db.patents_grupo24:** se refiere a la colección de la base de datos a la que se está consultando.
2. **Aggregate:** Las operaciones de agregación procesan los registros de datos y devuelven los resultados calculados. Las operaciones de agregación agrupan valores de varios documentos y pueden realizar una variedad de operaciones en los datos agrupados para devolver un único resultado.
3. **Group:** La \$group etapa agrupa los documentos por patent_year para calcular la suma del monto de cada uno y devolverla en el contador
4. **Sort:** se utiliza para ordenar los registros devueltos de la consulta en este caso de forma descendente

```
db.getCollection("patents_grupo24").find(
```

5. **GetCollection:** Devuelve una colección o un objeto de vista que es funcionalmente equivalente a usar la db.<collectionName> sintaxis. El método es útil para una colección o una vista cuyo nombre puede interactuar con el mongo propio shell, como los nombres que coinciden con un método de shell de base de datos .
6. **Find:** Ejecuta una consulta y devuelve el primer lote de resultados y la identificación del cursor, a partir del cual el cliente puede construir un cursor.

```
"$unwind" : {
  "path" : "$inventors",
  "preserveNullAndEmptyArrays" : true
}
```

7. **Unwind:** Deconstruye un campo de matriz a partir de los documentos de entrada para generar un documento para cada elemento. Cada documento de salida es el documento de entrada con el valor del campo de matriz reemplazado por el elemento.

```
{
  "$limit" : 5.0
}
```

8. **Limit:** Limita el número de documentos se pasan a la siguiente etapa de resultado de la consulta.

```
{
  "allowDiskUse" : false
}
```

9. **AllowDiskUse:** permite a MongoDB utilizar archivos temporales en el disco para almacenar datos que superen el límite de memoria del sistema de 100 megabytes mientras se procesa una operación de clasificación de bloqueo.

```
"$match" : {
  "$or" : [
    {
      "languages.name" : "Spanish"
    },
    {
      "languages.name" : "English"
    }
  ]
}
```

10. **Match:** Filtra los documentos para pasar solo los documentos que coinciden con las condiciones especificadas a la siguiente etapa de canalización.