



CICLO: [DAM]
MÓDULO DE [BASES DE DATOS]

[Tarea N° 07]

Alumno:
[Juan Carlos Filter Martín]
[15456141A]

Contenido

1. Documentos que se adjuntan a este informe.....	3
3. (RA7_d) Se han identificado distintas formas de gestión de la información según el tipo de base de datos no relacionales.....	3
A) Datos que tenemos para almacenar en una BD.....	3
B) Ventajas al usar una BD no relacional.....	4
Flexibilidad.....	4
Escalabilidad.....	4
Big Data.....	4
Alto rendimiento.....	4
Funcionales.....	4
Menor administración práctica.....	4
Económicas.....	4
Creado para la nube.....	4
4. (RA7_e) Se han utilizado las herramientas del sistema gestor para la gestión de la información almacenada.....	5
A) Descarga e instalación MongoDB Community Edition y MongoDB Compass.....	5
Descarga.....	5
Instalación.....	7
B) Crear una BD con el nombre del DNI “15456141A y la colección con el primer apellido “Filter”	12
C) Importar los datos anteriores a la colección como un archivo JSON.....	14
D) Hacer dos consultas.....	16
Consulta de todos los empleados.....	16
Consulta de todos los empleados que conocen el idioma Inglés.....	17

1. Documentos que se adjuntan a este informe.

A continuación se detallan los documentos que componen la presente entrega de la tarea:

1. Informe de elaboración de la tarea.
2. Archivo JSON

3. (RA7_d) Se han identificado distintas formas de gestión de la información según el tipo de base de datos no relacionales.

Se va a almacenar los datos en una Base de datos, al tener la información en archivos JSON se va a usar una base de datos documental MongoDB:

A) Datos que tenemos para almacenar en una BD

```
[
  {
    "nombre":"Raúl Cervantes",
    "profesion":"Programador",
    "edad":35,
    "lenguajes":["PHP","Javascript"],
    "disponibilidadParaViajar":true,
    "rangoProfesional": {
      "aniosDeExperiencia": 12,
      "nivel": "Senior"
    }
  },
  {
    "nombre":"Luis Boscoso",
    "profesion":"Programador",
    "edad":25,
    "lenguajes":["Java","C#"],
    "disponibilidadParaViajar":false,
    "rangoProfesional": {
      "aniosDeExperiencia": 3,
      "nivel": "Junior"
    }
  },
]
```

```
{
  "nombre":"Beatriz Molina",
  "profesion":"Team Leader",
  "edad":42,
  "idiomas":
  ["Ingles","Frances","Aleman"],
  "disponibilidadParaViajar":true,
  "rangoProfesional": {
    "aniosDeExperiencia": 14
  }
},
{
  "nombre":"Julia Fuentes",
  "profesion":"Administrativo",
  "edad":27,
  "idiomas":["Ingles"],
  "rangoProfesional": {
    "aniosDeExperiencia": 2,
    "nivel": "Junior"
  }
}
]
```

B) *Ventajas al usar una BD no relacional*

Con estos datos. ¿Qué ventajas se encuentran al usar una base de datos no relacional a diferencia de una relacional?

Flexibilidad

No hay una estructura fija a la hora de crear los esquemas de datos que permiten un desarrollo más rápido y más iterativo.

Esto lo hace perfecto para datos no estructurados o semi estructurados.

Escalabilidad

Están preparadas para soportar grandes cantidades de datos y aumentar la capacidad vía software sin tener que añadir más servidores.

Big Data

Grandes volúmenes son manejados fácilmente por las bases de datos NoSQL.

Alto rendimiento

Están diseñadas para trabajar con un modelo de datos concreto lo que nos provee de un mayor rendimiento.

Funcionales

Tiene APIs exclusivas y trabajan con cada tipo de datos de manera concreta.

Menor administración práctica.

Tiene herramientas automatizadas y modelos de datos simplificados requieren de menos trabajo de administración.

Económicas.

Al no necesitar servidores con grandes recursos, permite ir realizando inversiones más ajustadas e ir ampliando según las necesidades.

Creado para la nube.

Como pueden ser enormes y crecer exponencialmente, necesitan un entorno de alojamiento que pueda crecer y expandirse con ellos. La escalabilidad inherente de la nube la convierte en un hogar ideal para las bases de datos no relacionales.

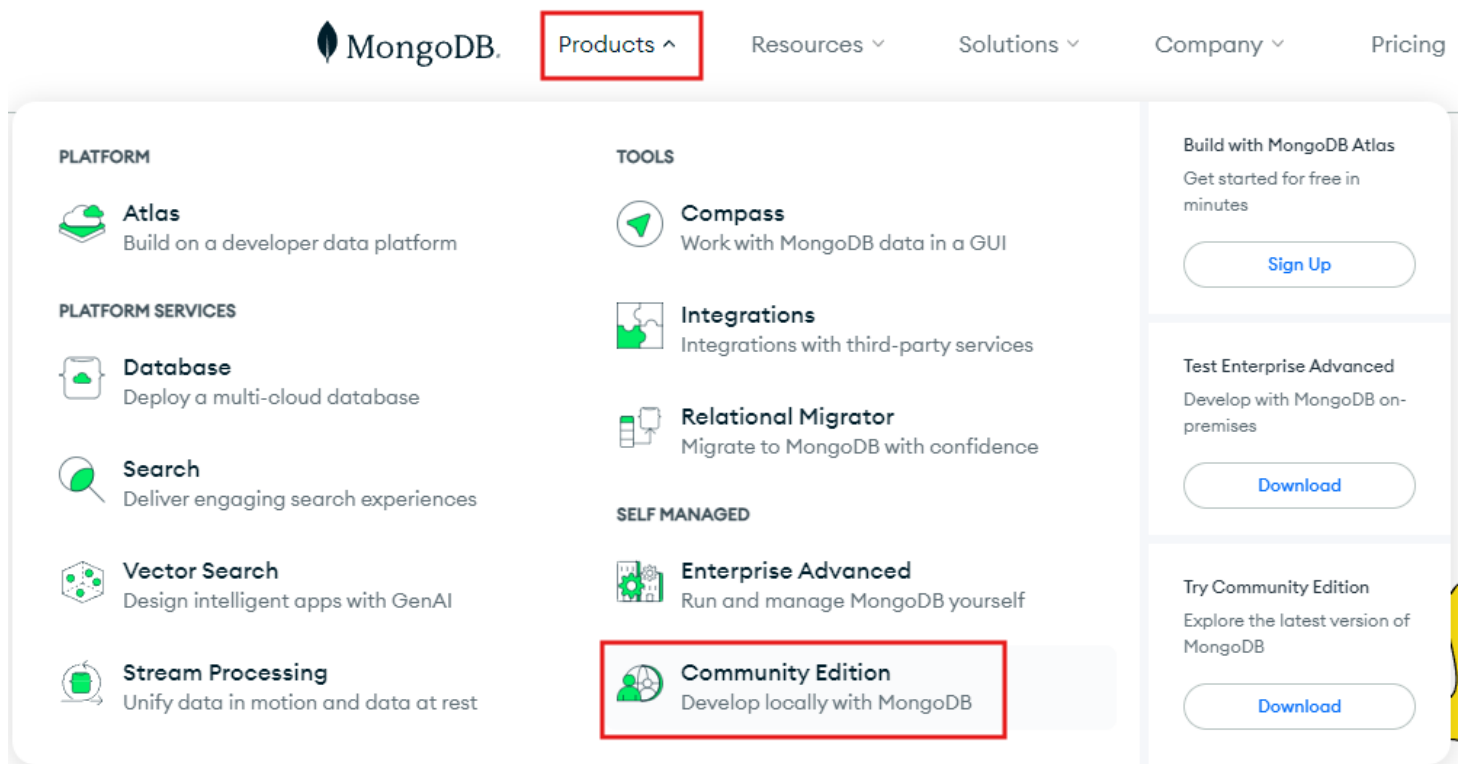
4. (RA7_e) Se han utilizado las herramientas del sistema gestor para la gestión de la información almacenada.

Los datos que tenemos para almacenar en una BD se van a realizar en MongoDB.

A) *Descarga e instalación MongoDB Community Edition y MongoDB Compass*

Descarga

x Vamos a su página web oficial y **entramos a Productos (Products) > Community Edition**



x Pulsamos en **Download Community**

MongoDB Community Edition

MongoDB is a general-purpose document database. With the Community Edition you can self-manage and host it locally or in the cloud. You can also develop with MongoDB Atlas for free in your local environment, including local experiences for full-text and vector search, as well as in the cloud.

Download Community

Try Atlas →

x Una vez aquí elegimos la versión, el Sistema operativo, el paquete y pulsamos en **Download**

MongoDB Atlas

MongoDB Enterprise Advanced

MongoDB Community Edition

MongoDB Community Server

MongoDB Community Kubernetes Operator

Tools

Atlas SQL Interface

Mobile & Edge

with the following two commands:

```
$ brew install mongodb-atlas  
$ atlas setup
```

Version
7.0.9 (current)

Platform
Windows x64

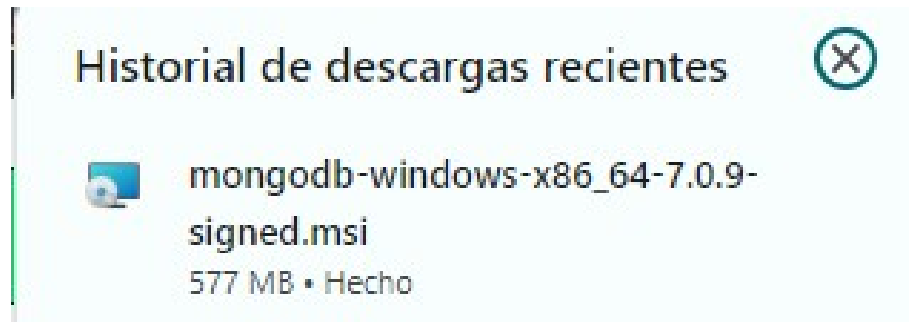
Package
msi

Download

Copy link

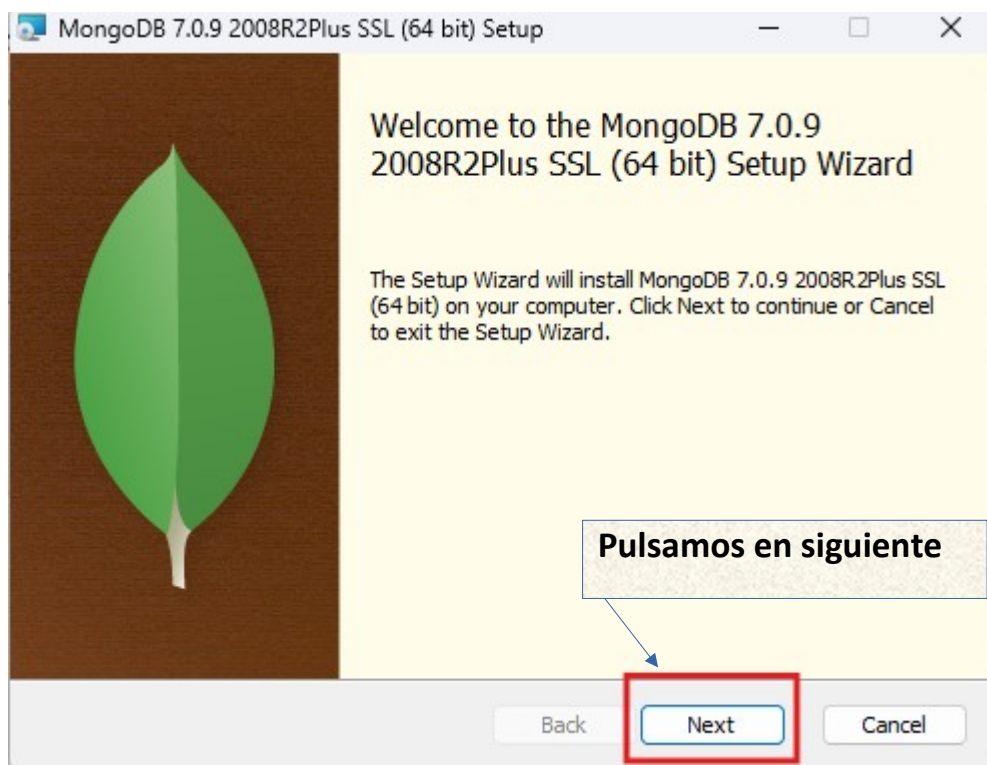
More Options

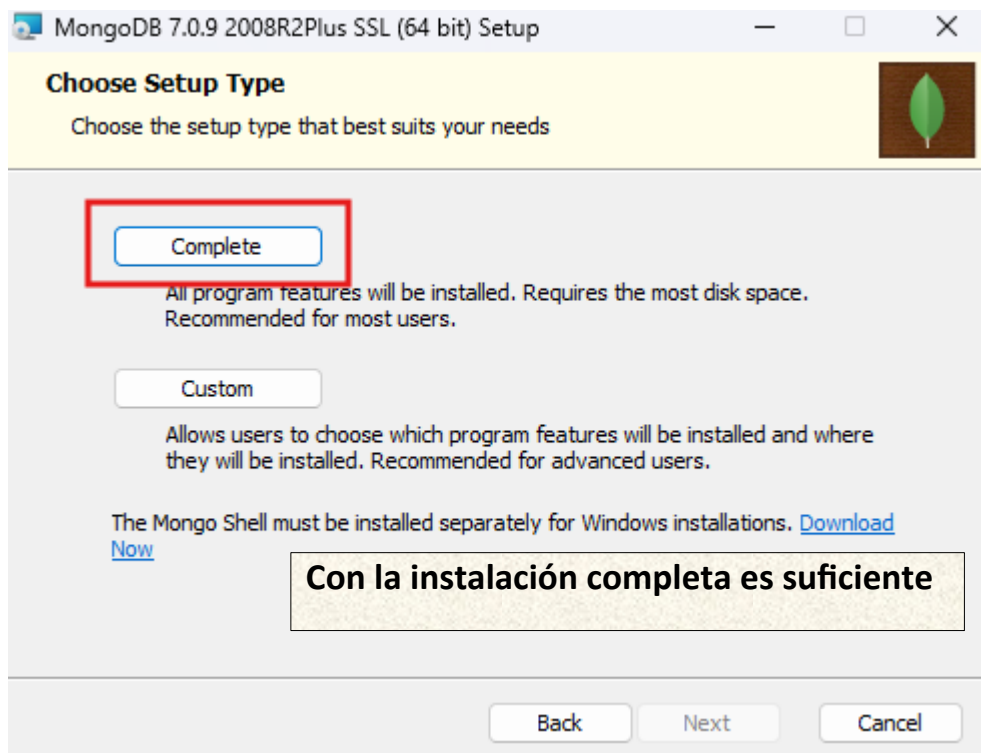
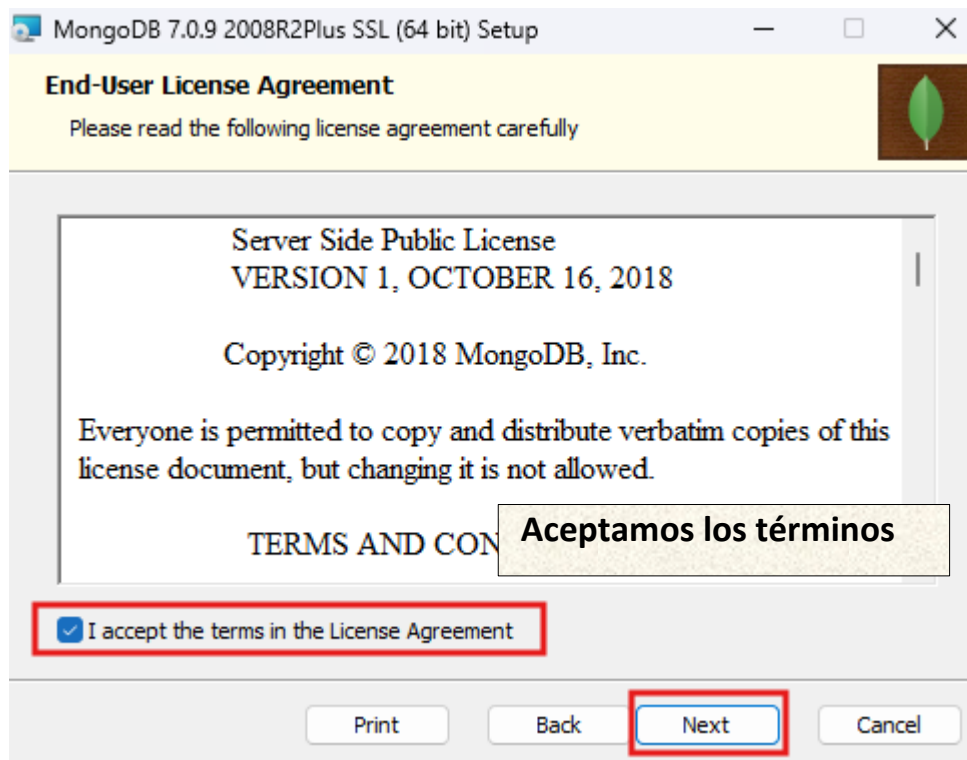
Una vez descargado procedemos a su instalación

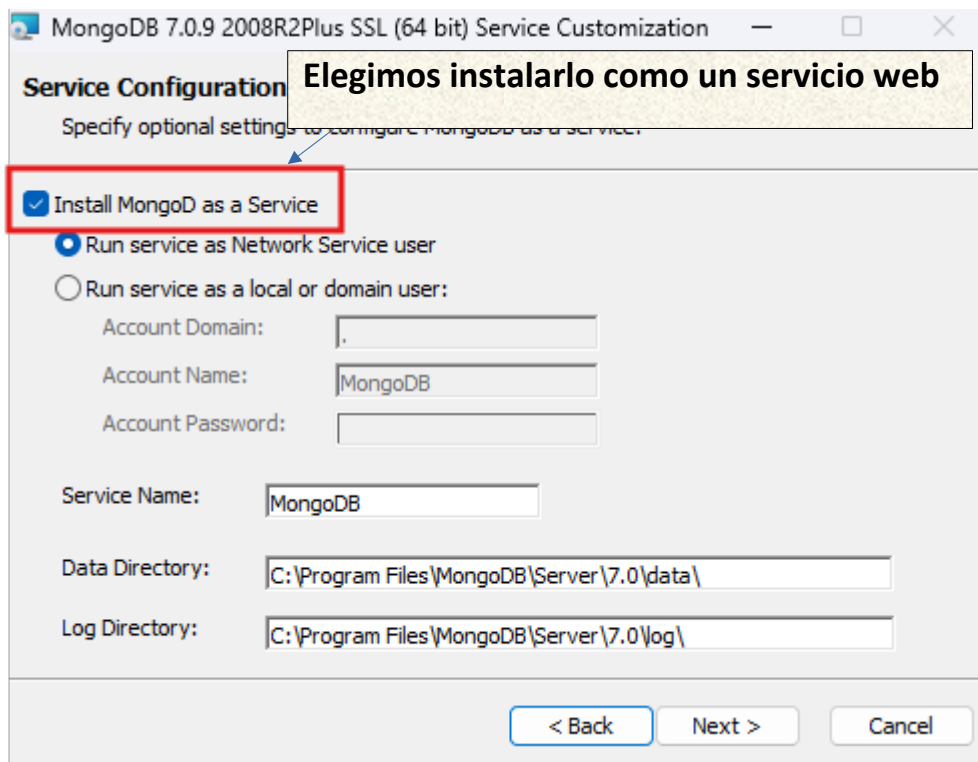


Instalación

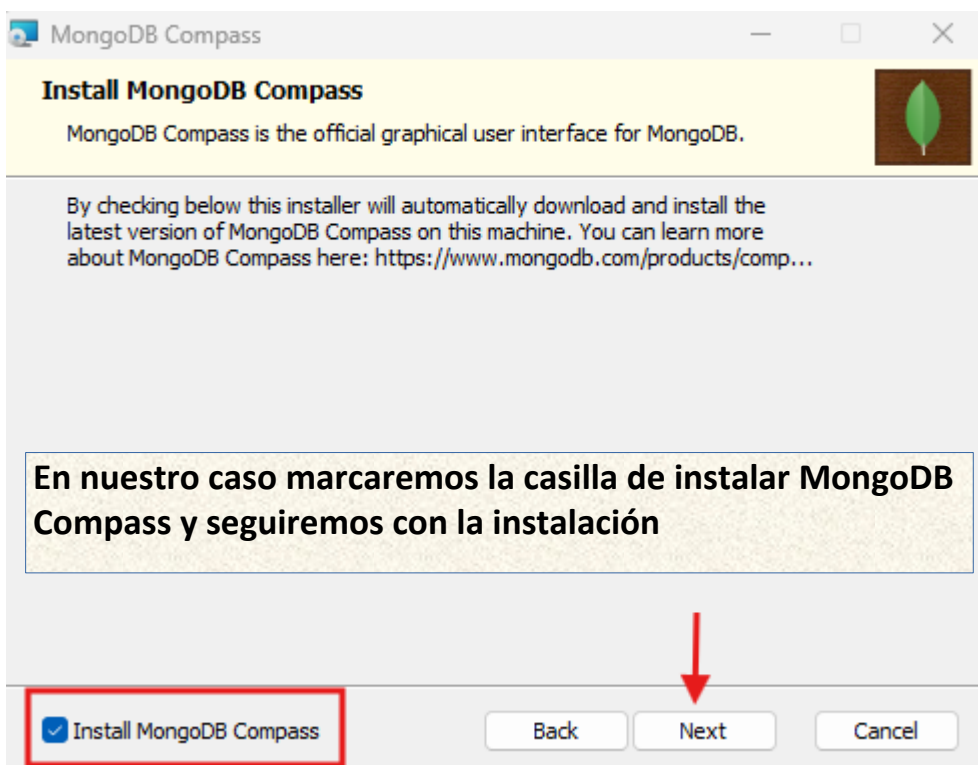
Ejecutamos el **exe** descargado y comenzará el asistente de instalación:

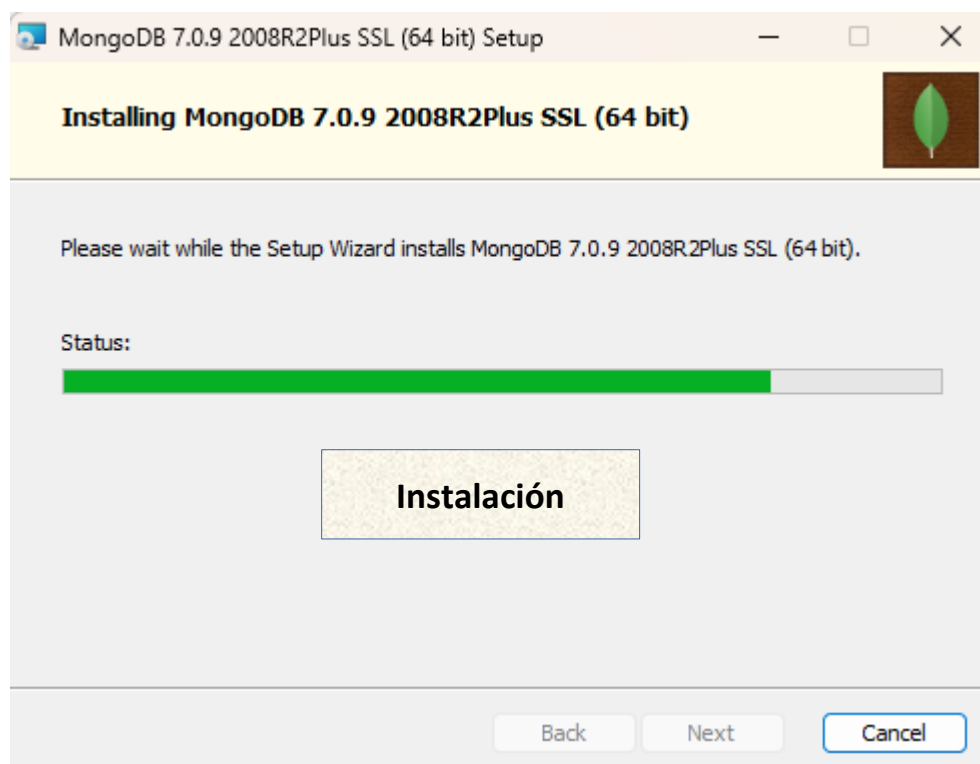
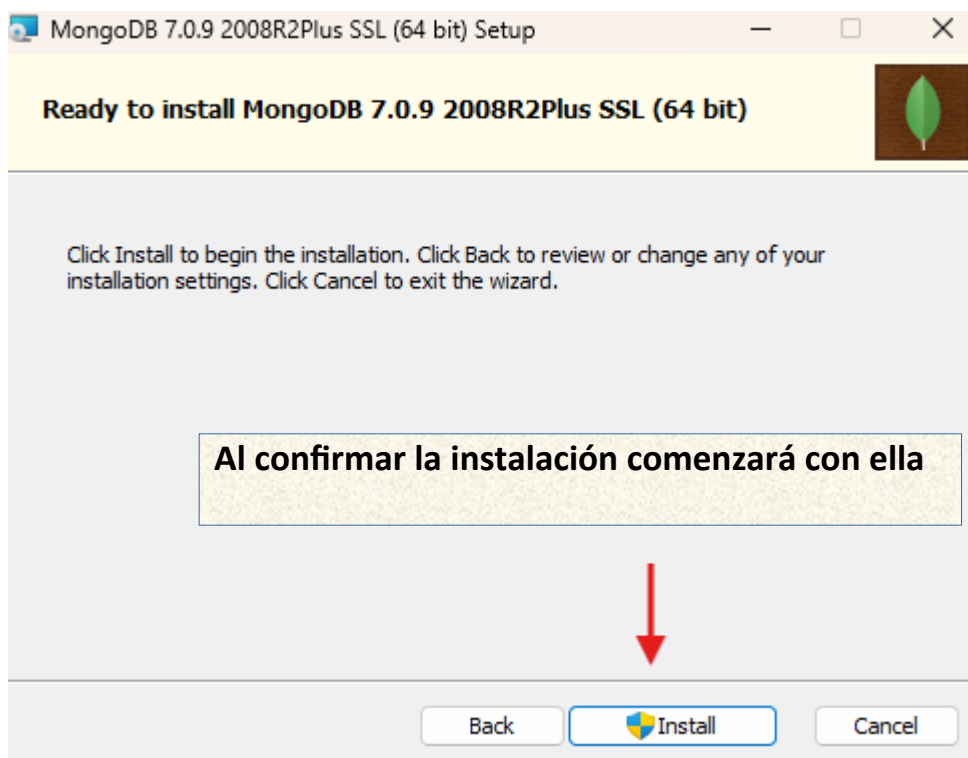


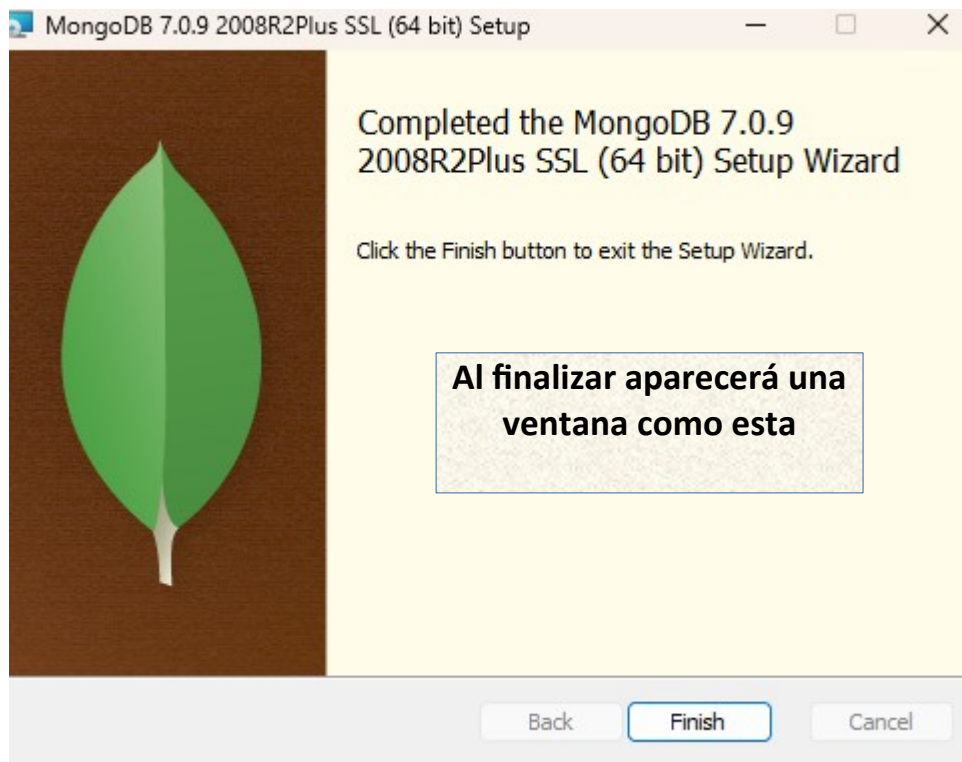




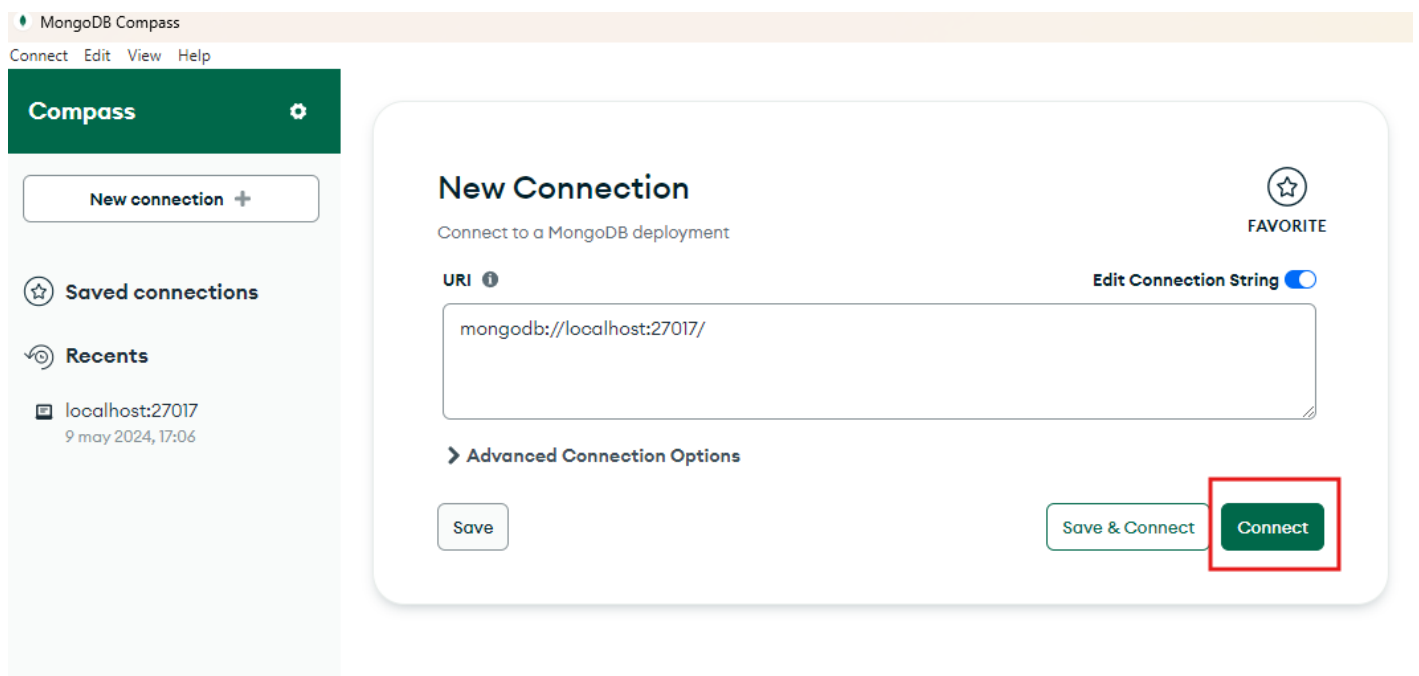
En el siguiente paso nos da la opción de instalar MongoDB Compass junto a la instalación de de mongo community manager.



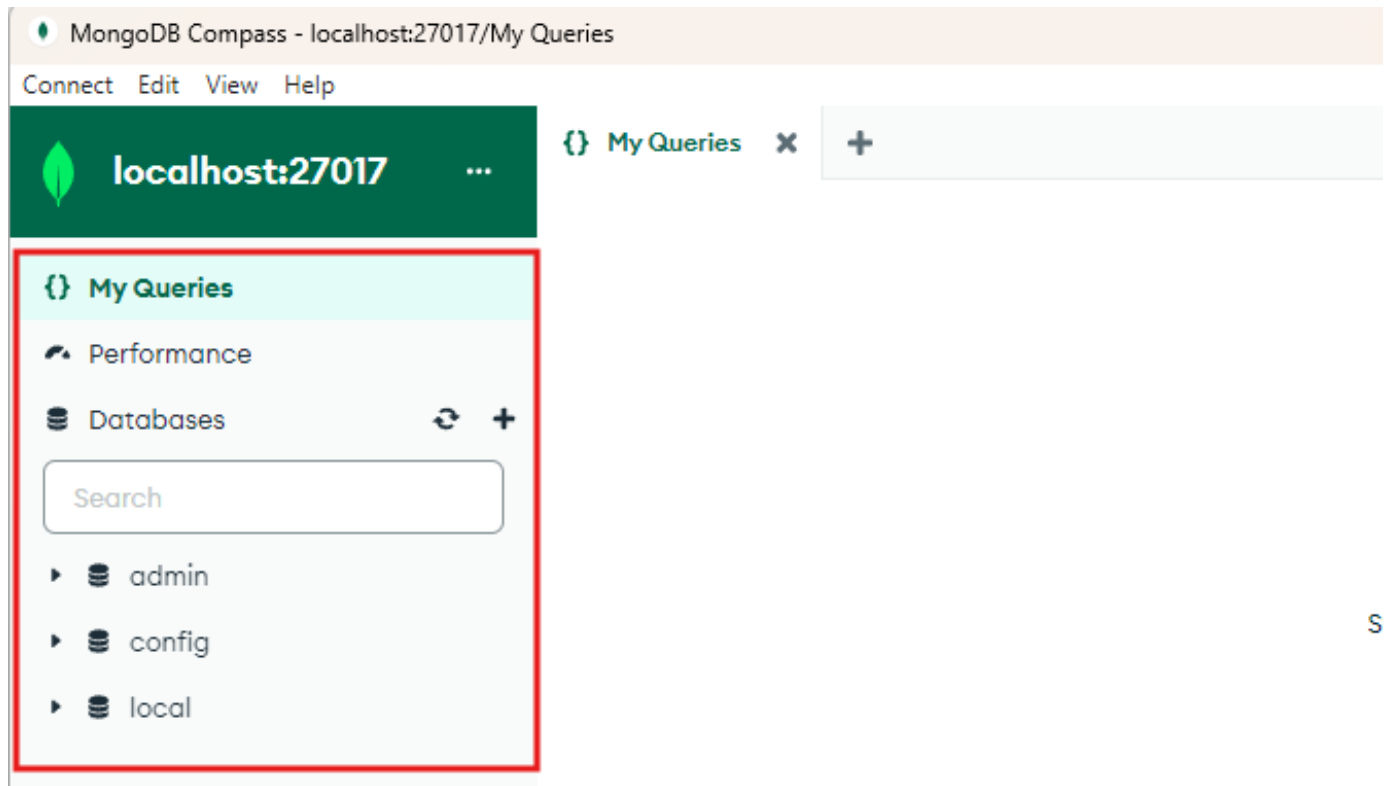




Una vez instalado y entremos accedemos a un menú de gestión del SGBD. Para conectarnos pulsamos en el botón (**connect**)



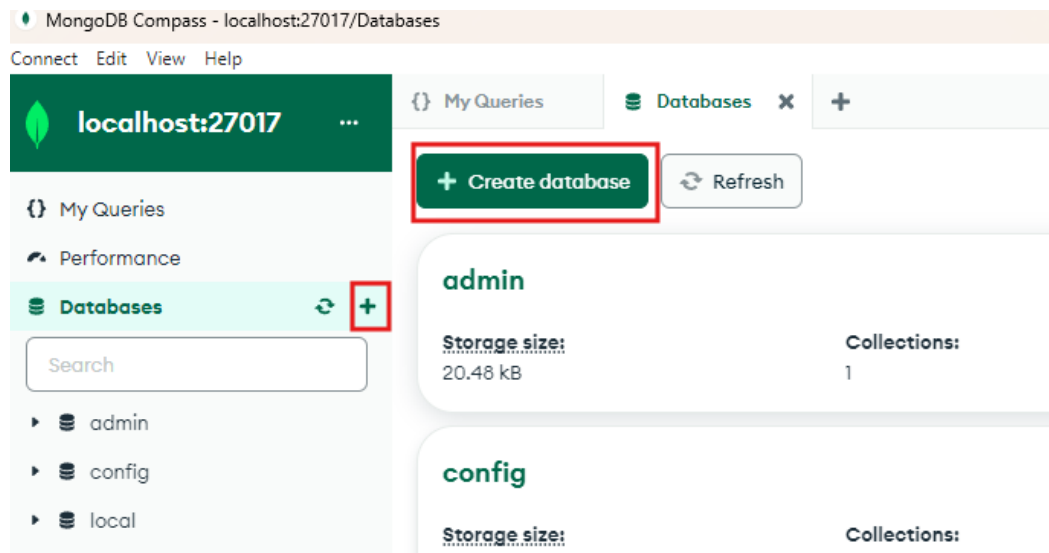
Al estar conectados tendremos el menú de gestión, el acceso a las bases de datos y diferentes opciones como pueden ser consultas



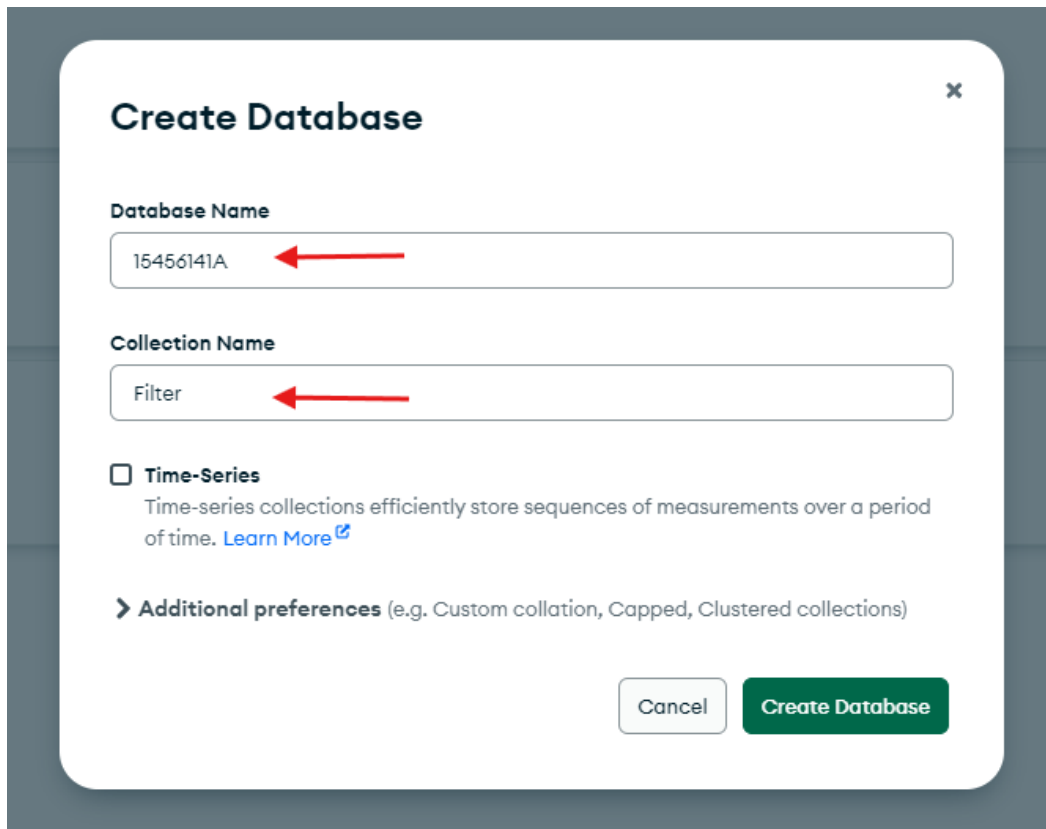
S

B) Crear una BD con el nombre del DNI “15456141A y la colección con el primer apellido “Filter”.

- x Para crear una base de datos o bien entramos en el apartado Databases y pulsamos en “**Create database**” o directamente en el icono que aparece del “+”



- x Se abrirá una ventana para crear la base de datos en la que se nos pedirá el nombre de la base de datos “15456141A” y el nombre de la colección “Filter”



Create Database

Database Name

15456141A

Collection Name

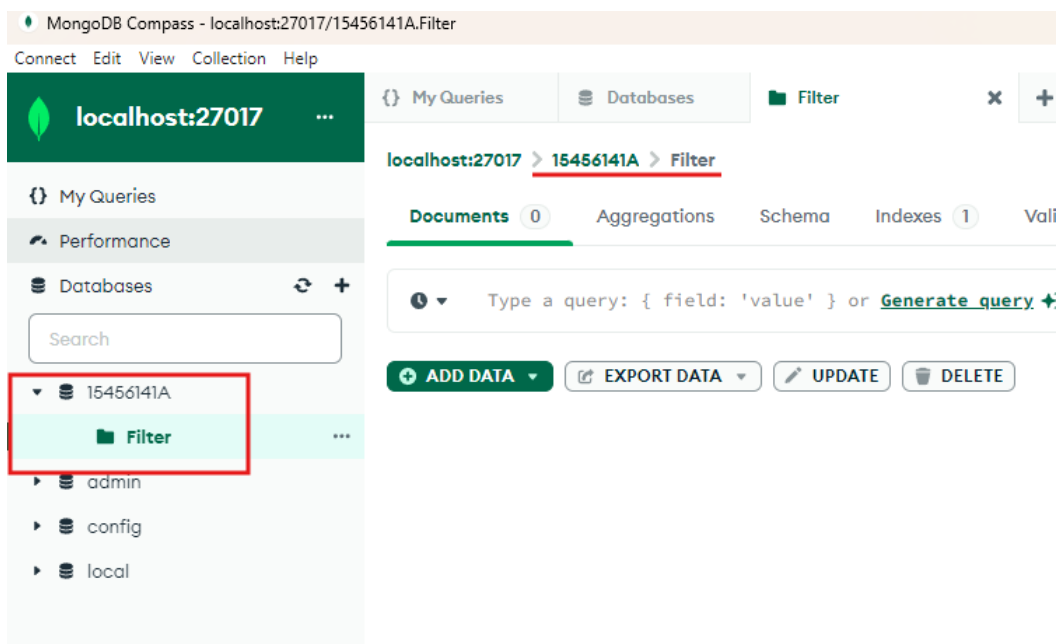
Filter

☐ **Time-Series**
Time-series collections efficiently store sequences of measurements over a period of time. [Learn More](#)

> **Additional preferences** (e.g. Custom collation, Capped, Clustered collections)

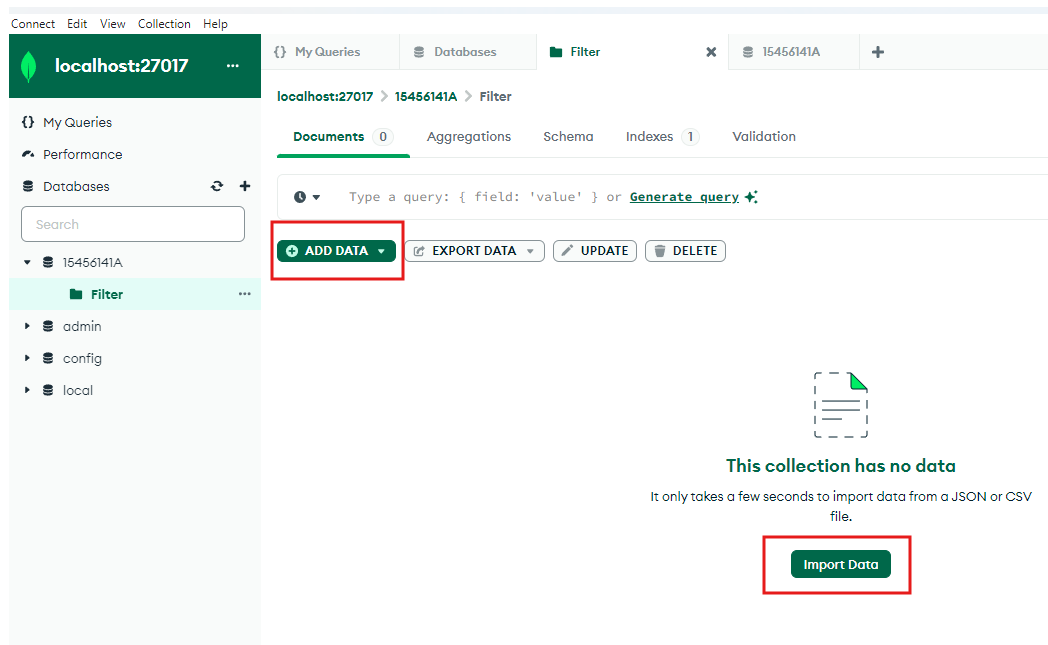
Cancel Create Database

- x Al pulsar en “create Database” tendremos la base de datos creada.

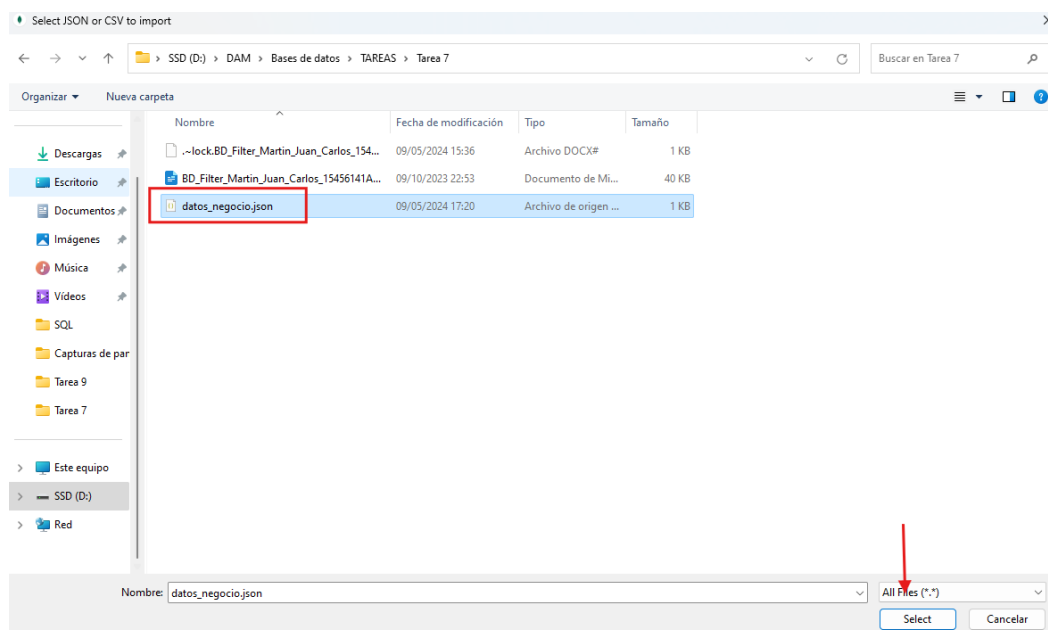


C) Importar los datos anteriores a la colección como un archivo JSON.

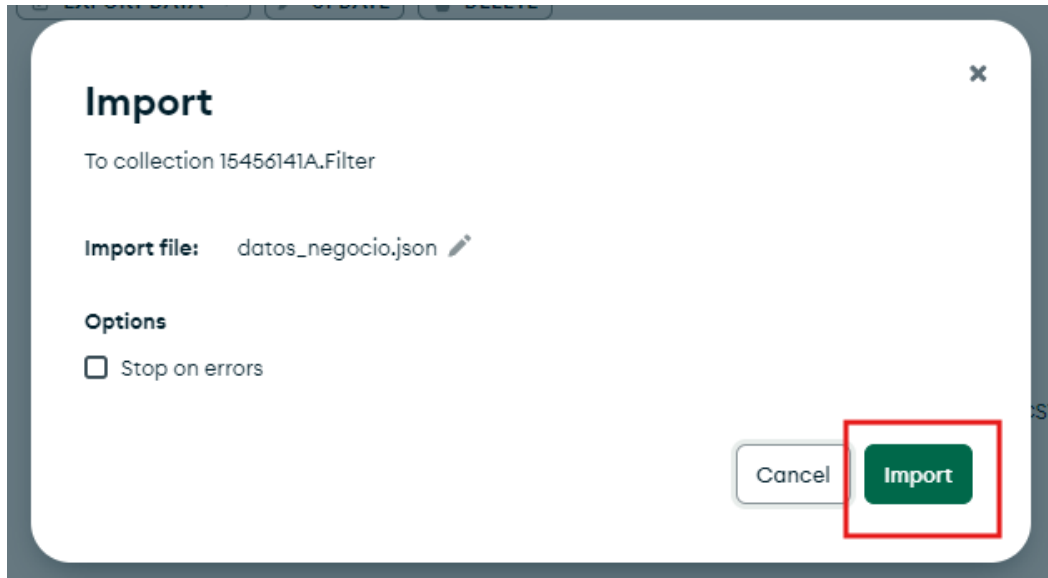
Para importar los datos que están en un archivo JSON **pulsamos en “ADD DATA” o “Import Data”**



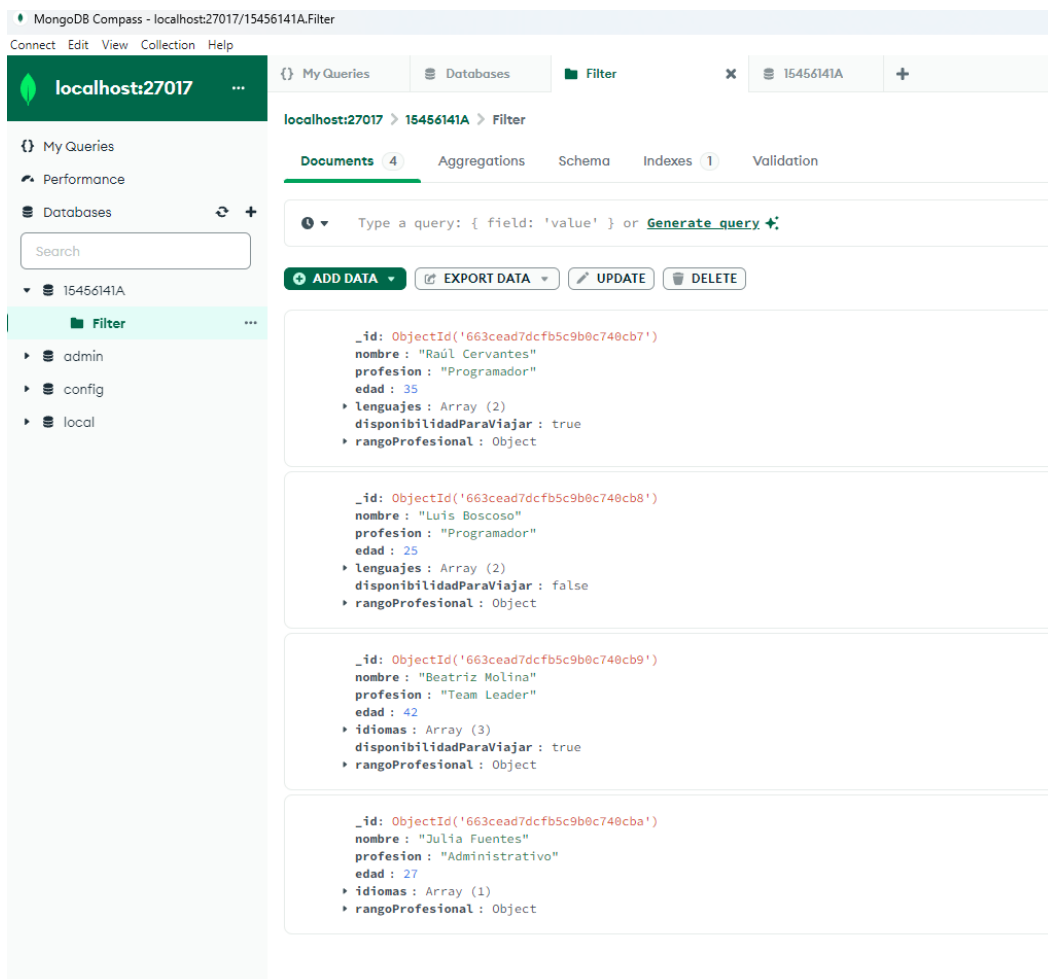
Buscamos donde se encuentra el archivo, lo seleccionamos y **pulsamos en “Select”**



Aparecerá una ultima ventana antes de importar, simplemente **pulsamos en “Import”**



Y con esto tendríamos los datos importados a la colección Filter de la base de datos 15456141A.



D) Hacer dos *consultas*

Consulta de todos los empleados

Para esta consulta simplemente dejamos el campo sin filtro y mostrará todos los empleados

The screenshot shows the MongoDB Compass interface. The left sidebar displays the database structure: localhost:27017 > 15456141A > Filter. The main panel shows the 'Documents' tab with a query field containing an empty filter ({}). Below the query field are buttons for 'ADD DATA', 'EXPORT DATA', 'UPDATE', and 'DELETE'. The results pane shows four documents, each with fields like _id, nombre, profesion, edad, lenguajes, disponibilidadParaViajar, and rangoProfesional.

localhost:27017 > 15456141A > Filter

Documents 4 Aggregations Schema Indexes 1 Validation

Generate query

ADD DATA EXPORT DATA UPDATE DELETE

```
{
  "_id": ObjectId("663cead7dcfb5c9b0c740cb7"),
  "nombre": "Raúl Cervantes",
  "profesion": "Programador",
  "edad": 35,
  "lenguajes": Array (2),
  "disponibilidadParaViajar": true,
  "rangoProfesional": Object
}
```

```
{
  "_id": ObjectId("663cead7dcfb5c9b0c740cb8"),
  "nombre": "Luis Boscoso",
  "profesion": "Programador",
  "edad": 25,
  "lenguajes": Array (2),
  "disponibilidadParaViajar": false,
  "rangoProfesional": Object
}
```

```
{
  "_id": ObjectId("663cead7dcfb5c9b0c740cb9"),
  "nombre": "Beatriz Molina",
  "profesion": "Team Leader",
  "edad": 42,
  "idiomas": Array (3),
  "disponibilidadParaViajar": true,
  "rangoProfesional": Object
}
```

```
{
  "_id": ObjectId("663cead7dcfb5c9b0c740cba"),
  "nombre": "Julia Fuentes",
  "profesion": "Administrativo",
  "edad": 27,
  "idiomas": Array (1),
  "rangoProfesional": Object
}
```


Consulta de todos los empleados que conocen el idioma Inglés

Indicamos el campo idiomas y que contenga "Inglés"

Como se puede ver en el resultado los 2 empleados que aparecen conocen el idioma "Inglés".

The screenshot shows the MongoDB Compass interface. The top bar indicates the connection to 'localhost:27017/15456141A.Filter'. The left sidebar shows the database structure with '15456141A' expanded and 'Filter' selected. The main panel shows the 'Documents' tab with a filter query `{idiomas: 'Ingles'}` highlighted by a red box. Below the filter, there are buttons for 'ADD DATA', 'EXPORT DATA', 'UPDATE', and 'DELETE'. The results section, also highlighted by a red box, displays two documents. The first document is for Beatriz Molina, a Team Leader, 42 years old, who speaks three languages: 'Ingles', 'Frances', and 'Aleman'. The second document is for Julia Fuentes, an Administrativo, 27 years old, who speaks one language: 'Ingles'. Red arrows point to the 'Ingles' entry in the 'idiomas' array for both documents.

MongoDB Compass - localhost:27017/15456141A.Filter

Connect Edit View Collection Help

localhost:27017 ...

My Queries Performance Databases Search

15456141A

Filter

localhost:27017 > 15456141A > Filter

Documents 4 Aggregations Schema Indexes 1 Validation

`{idiomas: 'Ingles'}`

ADD DATA EXPORT DATA UPDATE DELETE

```
{
  "_id": ObjectId('663cead7dcfb5c9b0c740cb9'),
  "nombre": "Beatriz Molina",
  "profesion": "Team Leader",
  "edad": 42,
  "idiomas": Array (3)
    0: "Ingles"
    1: "Frances"
    2: "Aleman"
  "disponibilidadParaViajar": true,
  "rangoProfesional": Object
}

{
  "_id": ObjectId('663cead7dcfb5c9b0c740cba'),
  "nombre": "Julia Fuentes",
  "profesion": "Administrativo",
  "edad": 27,
  "idiomas": Array (1)
    0: "Ingles"
  "rangoProfesional": Object
}
```