

Parte III - MongoDB
Pablo Martín Hidalgo
Gestión de Base de Datos



ÍNDICE

Enunciado

1. Debes seleccionar al menos cuatro tablas de tu proyecto que incluyan una relación N:M y una 1:N con el mayor número de tipos de datos soportados en MongoDB. Antes de continuar la profesora deberá dar el visto bueno a las mismas.
2. Debes pasar de SQL a MongoDB.
3. Inserta varios documentos utilizando los dos métodos de inserción de MongoDB
4. Elimina varios documentos utilizando los dos métodos de eliminación de MongoDB
5. Actualiza varios documentos utilizando los tres métodos de actualización de MongoDB
6. Consultas:
 - Al menos 5 consultas de datos simples.
 - Al menos 3 consultas con arrays.
 - Al menos 3 consultas con documentos embebidos.
 - Al menos 1 consulta de agrupación.

Nota: Debes utilizar proyecciones, operadores, ordenación y los modificadores sort y limit. Se valorará la complejidad de las mismas.

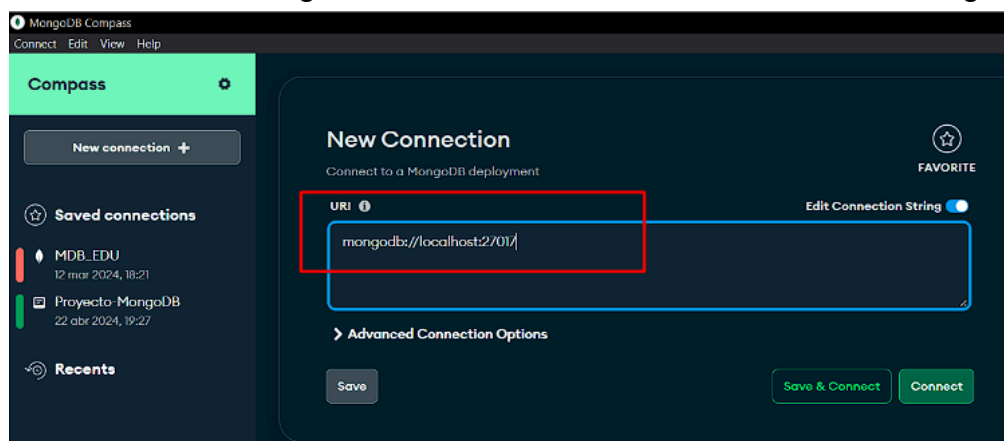
Migración

Para la migración de SQL a MongoDB he optado por usar la herramienta Studio 3t.

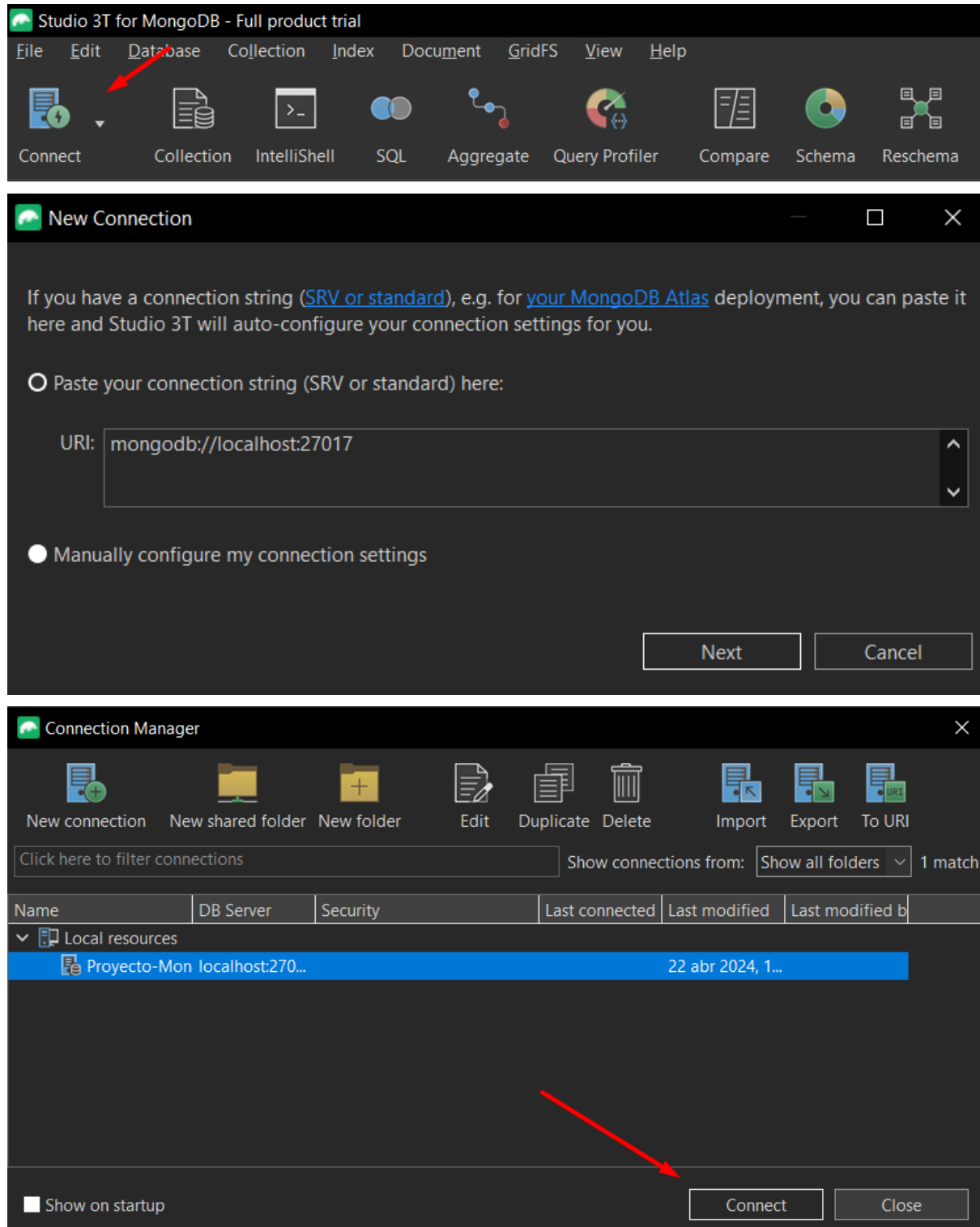
Las tablas seleccionadas para la migración serán:

-cantidad_electrica
-centrales
-empresa
-estimacion_demanda

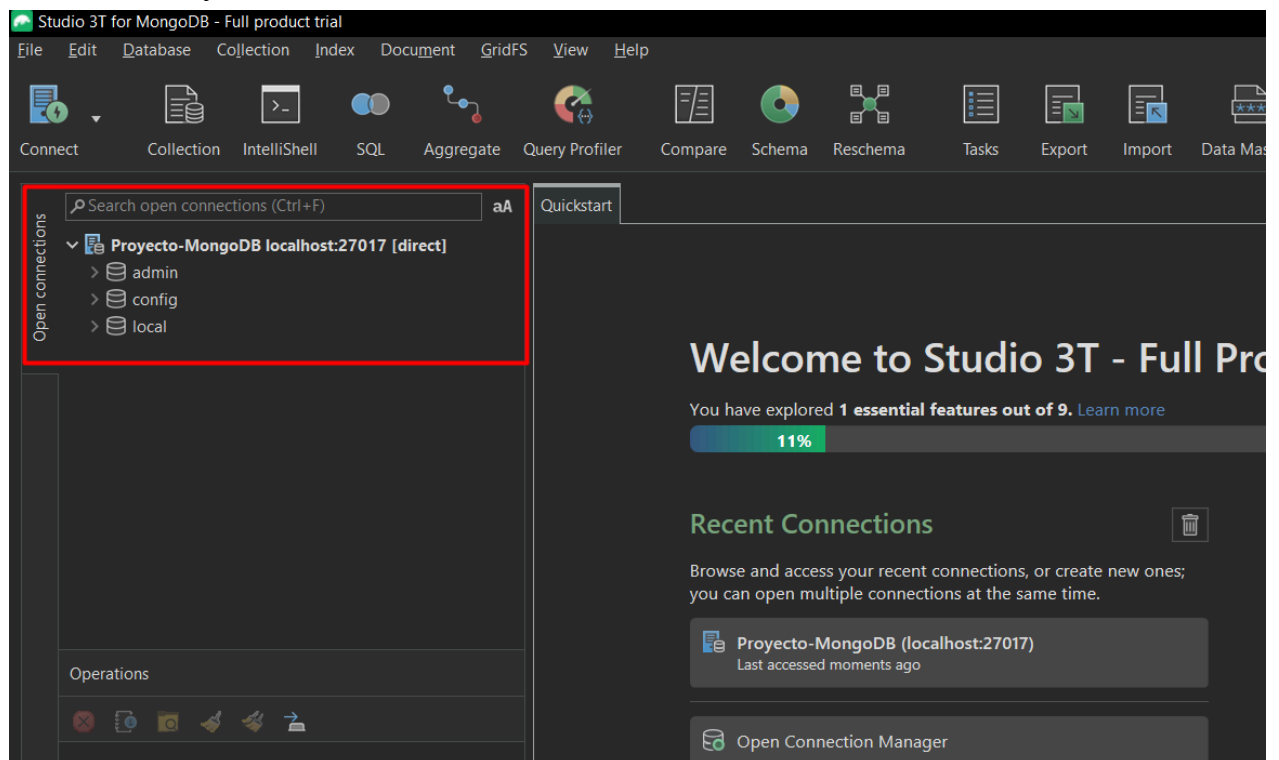
Para comenzar la migración debemos saber la URI de nuestro MongoDB Compass:



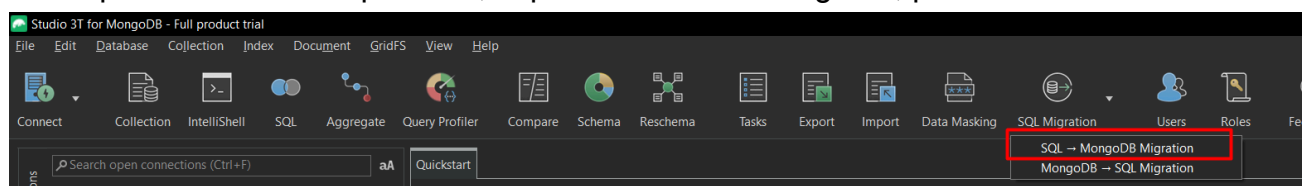
Luego, nos dirigimos al programa Studio 3t, donde tendremos que asociar nuestro MongoDB con él, para ello:



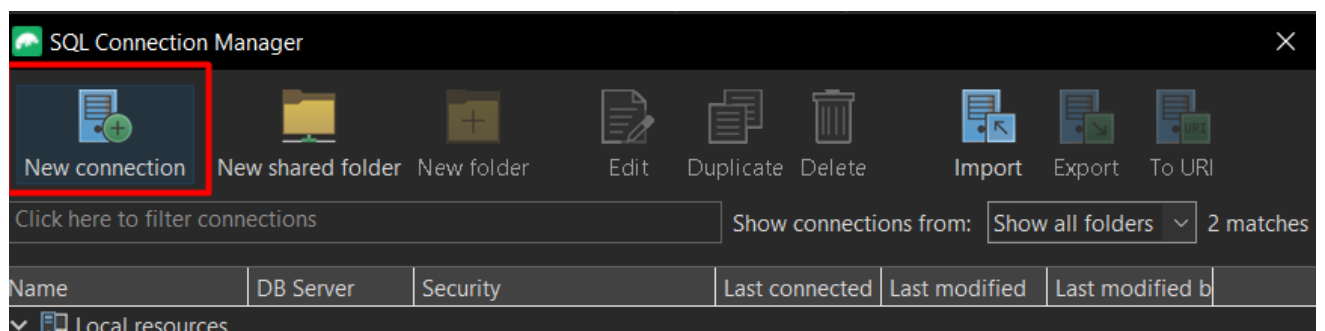
Como vemos ya lo tenemos asociado



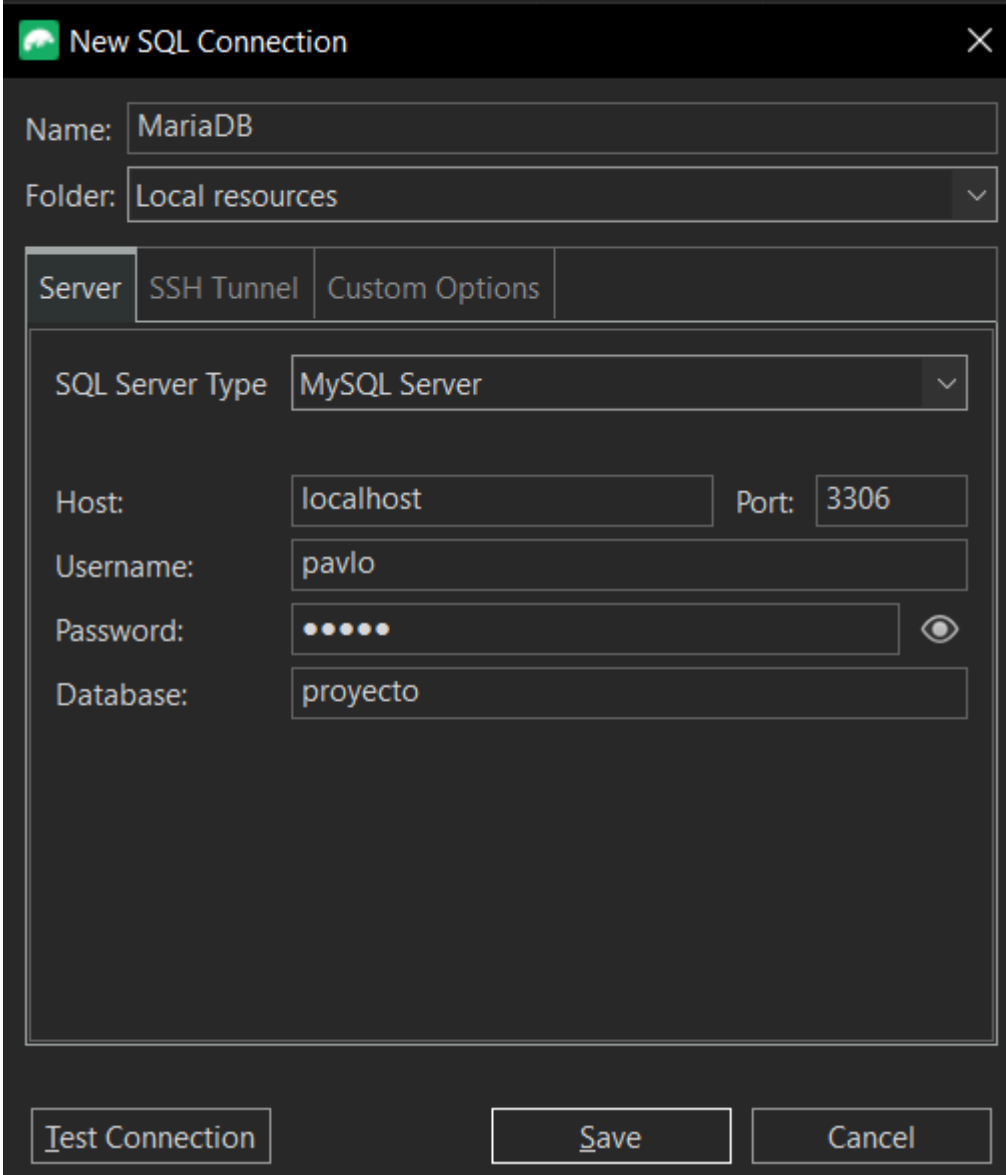
Ahora pasaremos a lo importante, el paso de SQL a MongoDB, para ello:



Creamos una nueva conexión con nuestra base de datos, en mi caso usaré MariaDB.



Añadimos nuestras credenciales para asociar el programa con nuestra base de datos:



The image shows a 'New SQL Connection' dialog box with a dark theme. It has a title bar with a green icon and a close button. The 'Name' field is 'MariaDB' and the 'Folder' is 'Local resources'. There are three tabs: 'Server' (selected), 'SSH Tunnel', and 'Custom Options'. Under the 'Server' tab, 'SQL Server Type' is 'MySQL Server'. The 'Host' is 'localhost' and 'Port' is '3306'. 'Username' is 'pavlo' and 'Password' is masked with dots. 'Database' is 'proyecto'. At the bottom are 'Test Connection', 'Save', and 'Cancel' buttons.

Name: MariaDB

Folder: Local resources

Server | SSH Tunnel | Custom Options

SQL Server Type: MySQL Server

Host: localhost Port: 3306

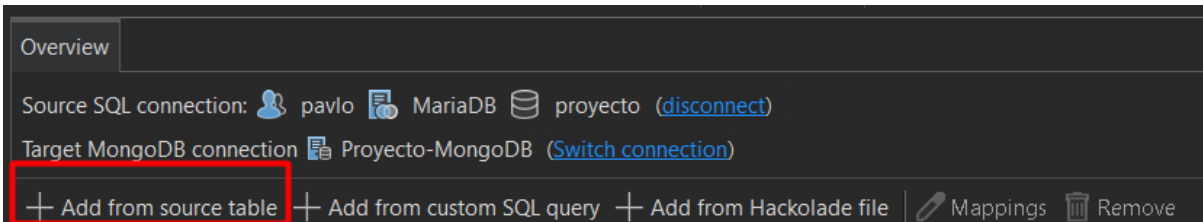
Username: pavlo

Password: [masked]

Database: proyecto

Test Connection Save Cancel

Seleccionamos las tablas que queremos migrar:



The image shows the 'Overview' panel of a migration tool. It displays the 'Source SQL connection' as 'pavlo MariaDB proyecto' with a 'disconnect' link. The 'Target MongoDB connection' is 'Proyecto-MongoDB' with a 'Switch connection' link. At the bottom, there are four buttons: 'Add from source table' (highlighted with a red box), 'Add from custom SQL query', 'Add from Hackolade file', and 'Mappings'. A 'Remove' button is also present.

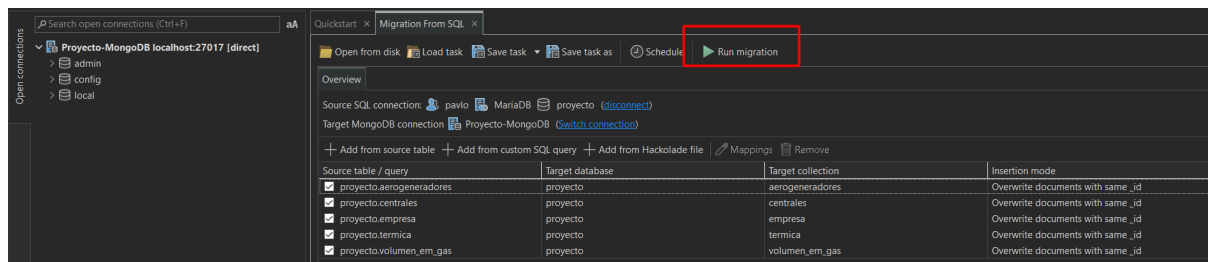
Overview

Source SQL connection: pavlo MariaDB proyecto (disconnect)

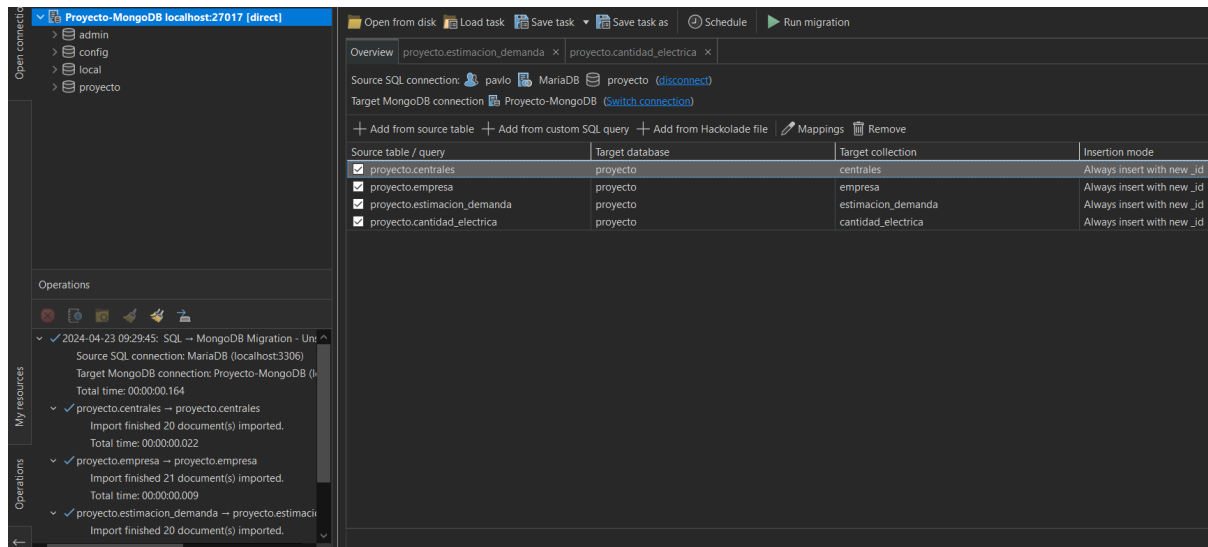
Target MongoDB connection: Proyecto-MongoDB (Switch connection)

+ Add from source table + Add from custom SQL query + Add from Hackolade file | Mappings Remove

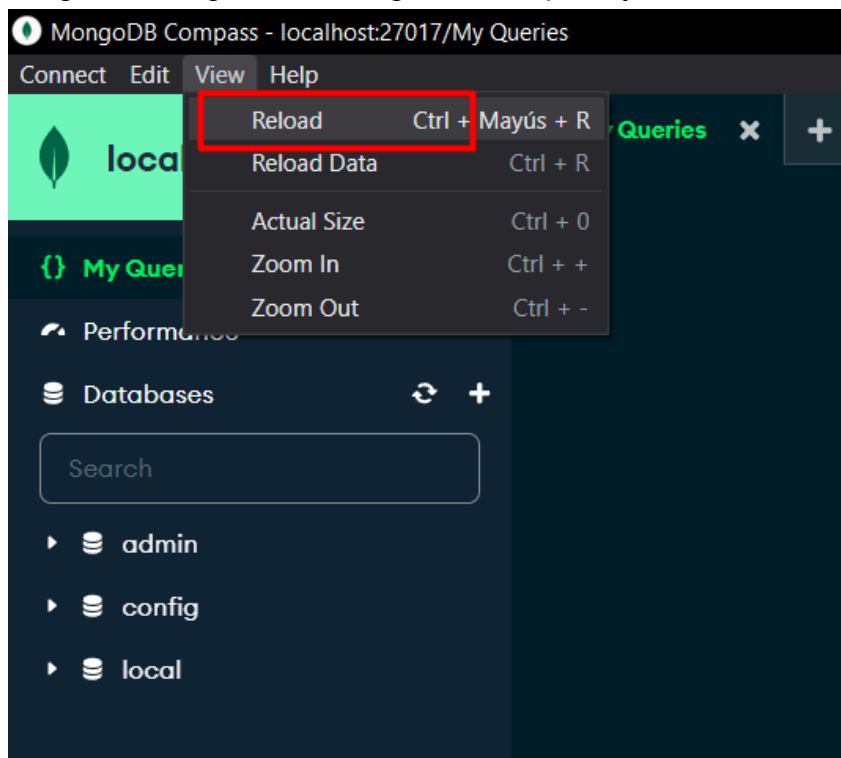
Y por último, le damos a “run migration” para finalizar.



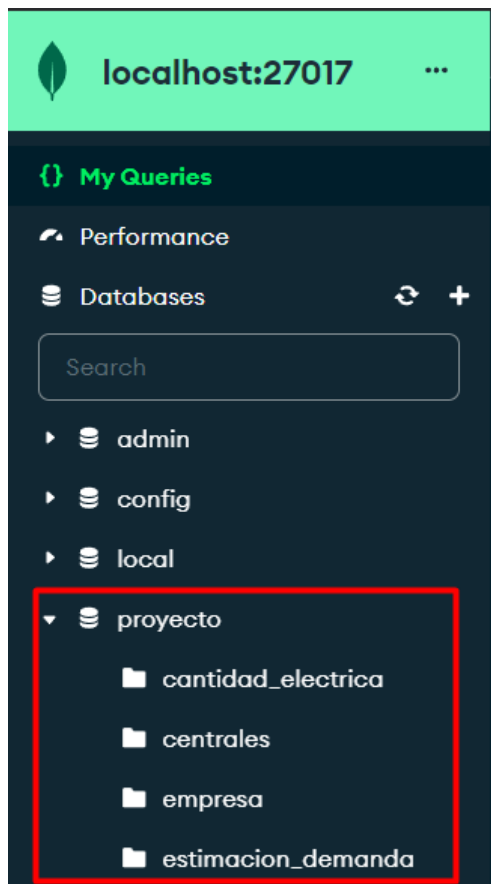
Como podemos comprobar se ha realizado todo correctamente:



Luego nos dirigimos a MongoDB Compass y reiniciamos la conexión:



Y como podemos ver, las tablas se han migrado automáticamente:



Ya solo nos quedaría usar nuestra base de datos

```
>_MONGOSH

> show dbs
< admin      40.00 KiB
  config    112.00 KiB
  local     72.00 KiB
  proyecto  160.00 KiB
> use proyecto
< switched to db proyecto
proyecto> |
```


Inserción de datos

-Insertar una cantidad eléctrica en una fecha específica con insertOne

```
> db.cantidad_electrica.insertOne({
  "FECHA": new Date("2024-04-20T00:00:00Z"),
  "CANTIDAD_E": 1620.00
});
< {
  acknowledged: true,
  insertedId: ObjectId('66276b1fe7af51df4fff2df2')
}
> db.cantidad_electrica.find({ "FECHA": new Date("2024-04-20T00:00:00Z") })
< {
  _id: ObjectId('66276b1fe7af51df4fff2df2'),
  FECHA: 2024-04-20T00:00:00.000Z,
  CANTIDAD_E: 1620
}
```

-Insertar varios documentos con insertMany

```
> db.empresa.insertMany([
  {
    CIF: "A78901234",
    Nombre: "Gastronomía Deliciosa S.C."
  },
  {
    CIF: "A32109876",
    Nombre: "Construcciones Duraderas ConstruDUR"
  },
  {
    CIF: "A65432109",
    Nombre: "Transportes VelozExpress"
  },
  {
    CIF: "A90123456",
    Nombre: "Servicios Médicos SaludTotal"
  },
  {
    CIF: "A54321098",
    Nombre: "Turismo Aventura ExploraMundo"
  }
]);
< {
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('66276cebe7af51df4fff2df3'),
    '1': ObjectId('66276cebe7af51df4fff2df4'),
    '2': ObjectId('66276cebe7af51df4fff2df5'),
    '3': ObjectId('66276cebe7af51df4fff2df6'),
    '4': ObjectId('66276cebe7af51df4fff2df7')
  }
}
```

Eliminación de datos

-deleteOne: Eliminar un solo documento que cumple con varias condiciones.

```
> db.centrales.deleteOne({ UBICACION: "Central Solar Andalucía", CAPACIDAD_MAX: { $lt: 800 } });
< {
  acknowledged: true,
  deletedCount: 1
}
```

-deleteMany: Eliminar varios documentos que cumplen con una condición y pertenecen a una fecha específica.

```
> db.estimacion_demanda.deleteMany({ TEMP: { $lt: 20 }, FECHA: { $gt: ISODate("2024-01-10") } });
< {
  acknowledged: true,
  deletedCount: 4
}
```

Modificación de datos

-updateOne: Este código actualiza la ubicación y la fecha de una central específica en la colección `centrales` de MongoDB.

```
> db.centrales.updateOne(
  { COD_CENTRAL: "C3" },
  { $set: { UBICACION: "Parque de los Príncipes", FECHA: new Date() } }
);
< {
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

-updateMany: actualiza varios documentos en la colección `estimacion_demanda` estableciendo el valor del campo `CUOTA_PRODUCCION` a 90 en aquellos documentos donde el campo `DEMANDA` sea mayor o igual a 1300.

```
> db.estimacion_demanda.updateMany(  
  { DEMANDA: { $gte: 1300 } },  
  { $set: { CUOTA_PRODUCCION: 90 } }  
);  
< {  
  acknowledged: true,  
  insertedId: null,  
  matchedCount: 8,  
  modifiedCount: 8,  
  upsertedCount: 0  
}
```

Consultas

Consultas sencillas

1. Consulta de las centrales ubicadas en regiones que contienen la palabra "Solar":

```
> db.centrales.find({ UBICACION: /Solar/ })  
< {  
  _id: ObjectId('6627636920e00c4d5f77a02f'),  
  COD_CENTRAL: 'C14',  
  CIF: 'A54321098',  
  FECHA: 2024-01-13T23:00:00.000Z,  
  UBICACION: 'Parque Solar Asturias',  
  CAPACIDAD_MAX: Decimal128('750.00')  
}  
{  
  _id: ObjectId('6627636920e00c4d5f77a033'),  
  COD_CENTRAL: 'C18',  
  CIF: 'A76543210',  
  FECHA: 2024-01-17T23:00:00.000Z,  
  UBICACION: 'Parque Solar Extremadura',  
  CAPACIDAD_MAX: Decimal128('1100.00')  
}
```

2. Consulta de las estimaciones de demanda donde la temperatura sea mayor a 22 grados Celsius:

```
> db.estimacion_demanda.find({ TEMP: { $gt: 22 } })
< {
  _id: ObjectId('6627636920e00c4d5f77a053'),
  FECHA: 2023-12-31T23:00:00.000Z,
  CUOTA_PRODUCCION: 75,
  TEMP: Decimal128('50.00'),
  DEMANDA: Decimal128('1200.50'),
  centrales: {}
}
{
  _id: ObjectId('6627636920e00c4d5f77a056'),
  FECHA: 2024-01-03T23:00:00.000Z,
  CUOTA_PRODUCCION: 90,
  TEMP: Decimal128('24.00'),
  DEMANDA: Decimal128('1400.00'),
  centrales: {}
}
```

3. Consulta de las centrales con una capacidad máxima entre 800 y 1000 MW:

```
> db.centrales.find({ CAPACIDAD_MAX: { $gte: 800, $lte: 1000 } })
< {
  _id: ObjectId('6627636920e00c4d5f77a02d'),
  COD_CENTRAL: 'C12',
  CIF: 'A65432109',
  FECHA: 2024-01-11T23:00:00.000Z,
  UBICACION: 'Planta Térmica Sevilla',
  CAPACIDAD_MAX: Decimal128('850.00')
}
{
  _id: ObjectId('6627636920e00c4d5f77a02e'),
  COD_CENTRAL: 'C13',
  CIF: 'A90123456',
  FECHA: 2024-01-12T23:00:00.000Z,
  UBICACION: 'Central Biomasa Zaragoza',
  CAPACIDAD_MAX: Decimal128('1000.00')
}
```

4. Consulta de la cantidad eléctrica para fechas entre el 1 de enero de 2024 y el 10 de enero de 2024:

```
> db.cantidad_electrica.find({ FECHA: { $gte: ISODate("2024-01-01"), $lte: ISODate("2024-01-10") } })
< {
  _id: ObjectId('6627636920e00c4d5f77a068'),
  FECHA: 2024-01-01T23:00:00.000Z,
  CANTIDAD_E: Decimal128('1800.75')
}
{
  _id: ObjectId('6627636920e00c4d5f77a069'),
  FECHA: 2024-01-02T23:00:00.000Z,
  CANTIDAD_E: Decimal128('1300.25')
}
```

5. Consulta de centrales ubicadas en Andalucía:

```
> db.centrales.find({ UBICACION: /Andalucía/ })
< {
  _id: ObjectId('6627636920e00c4d5f77a032'),
  COD_CENTRAL: 'C17',
  CIF: 'A43210987',
  FECHA: 2024-01-16T23:00:00.000Z,
  UBICACION: 'Central Hidroeléctrica Andalucía',
  CAPACIDAD_MAX: Decimal128('850.00')
}
```

Consultas con arrays

1. Buscar todas las empresas con un nombre que contenga la palabra "Tecnológicos":

```
> db.empresa.find({ "Nombre": /Tecnológicos/ })
< {
  _id: ObjectId('6627636920e00c4d5f77a03e'),
  CIF: 'A01234567',
  Nombre: 'Desarrollos Tecnológicos Avanzados'
}
```

2. Buscar todas las empresas con CIF en una lista específica y cuyos nombres no contienen la palabra "Internacional":

```
> db.empresa.find({ "CIF": { $in: ["A12345678", "A98765432", "A23456789"] }, "Nombre": { $not: /Internacional/ } })
< {
  _id: ObjectId('6627636920e00c4d5f77a041'),
  CIF: 'A12345678',
  Nombre: 'Empresa Solar S.A.'
}
{
  _id: ObjectId('6627636920e00c4d5f77a043'),
  CIF: 'A23456789',
  Nombre: 'Industrias Tecnológicas Avanzadas'
}
```

3. Buscar todas las centrales con una capacidad máxima superior a 1000 MW y que están ubicadas en una lista específica de regiones:

```
> db.centrales.find({ "CAPACIDAD_MAX": { $gt: 1000 }, "UBICACION": { $in: [/Cataluña/, /Valencia/, /Madrid/] } })
< {
  _id: ObjectId('6627636920e00c4d5f77a031'),
  COD_CENTRAL: 'C16',
  CIF: 'A21098765',
  FECHA: 2024-01-15T23:00:00.000Z,
  UBICACION: 'Parque Eólico Cataluña',
  CAPACIDAD_MAX: Decimal128('1200.00')
}
{
  _id: ObjectId('6627636920e00c4d5f77a039'),
  COD_CENTRAL: 'C5',
  CIF: 'A56789012',
  FECHA: 2024-01-04T23:00:00.000Z,
  UBICACION: 'Planta Biomasa Valencia',
  CAPACIDAD_MAX: Decimal128('1200.00')
}
```

Consultas con documentos embebidos

1. Consulta para obtener empresas que tienen al menos una central asociada, pero solo mostrando el nombre de la empresa y el número de centrales:

```
> db.empresa.aggregate([
  {
    $lookup: {
      from: "centrales",
      localField: "CIF",
      foreignField: "CIF",
      as: "centrales"
    }
  },
  {
    $match: {
      "centrales": { $exists: true, $not: { $size: 0 } }
    }
  },
  {
    $project: {
      "_id": 0,
      "Nombre": 1,
      "Numero_de_centrales": { $size: "$centrales" }
    }
  }
])
< {
  Nombre: 'Desarrollos Tecnológicos Avanzados',
  Numero_de_centrales: 1
}
```

2. Consulta para encontrar todas las centrales con información sobre la empresa propietaria, pero solo mostrando el nombre de la empresa y el código de la central:

```
> db.centrales.aggregate([
  {
    $lookup: {
      from: "empresa",
      localField: "CIF",
      foreignField: "CIF",
      as: "empresa_propietaria"
    }
  },
  {
    $project: {
      "_id": 0,
      "COD_CENTRAL": 1,
      "Nombre_de_empresa_propietaria": "$empresa_propietaria.Nombre"
    }
  }
])
< {
  COD_CENTRAL: 'C10',
  Nombre_de_empresa_propietaria: [
    'Gastronomía Deliciosa S.C.',
    'Gastronomía Deliciosa S.C.'
  ]
}
```


3. Consulta para obtener estimaciones de demanda junto con la información de la central asociada, pero solo para aquellas estimaciones de demanda con una temperatura superior a 40 grados Celsius:

```
> db.estimacion_demanda.aggregate([
  {
    $lookup: {
      from: "centrales",
      localField: "FECHA",
      foreignField: "FECHA",
      as: "central"
    }
  },
  {
    $match: {
      "TEMP": { $gt: 40 }
    }
  }
])
< {
  _id: ObjectId('6627636920e00c4d5f77a053'),
  FECHA: 2023-12-31T23:00:00.000Z,
  CUOTA_PRODUCCION: 75,
  TEMP: Decimal128('50.00'),
  DEMANDA: Decimal128('1200.50'),
  centrales: {},
  central: []
}
```

Consultas de agrupación

1. Agregación para encontrar la temperatura mínima y máxima registrada:

```
> db.estimacion_demanda.aggregate([
  {
    $group: {
      _id: null,
      min_temp: { $min: "$TEMP" },
      max_temp: { $max: "$TEMP" }
    }
  }
])
< {
  _id: null,
  min_temp: Decimal128('16.50'),
  max_temp: Decimal128('50.00')
}
```

2. Agregación para calcular la capacidad total de todas las centrales:

```
> db.centrales.aggregate([
  {
    $group: {
      _id: null,
      total_capacity: { $sum: "$CAPACIDAD_MAX" }
    }
  }
])
< {
  _id: null,
  total_capacity: Decimal128('18400.00')
}
```