

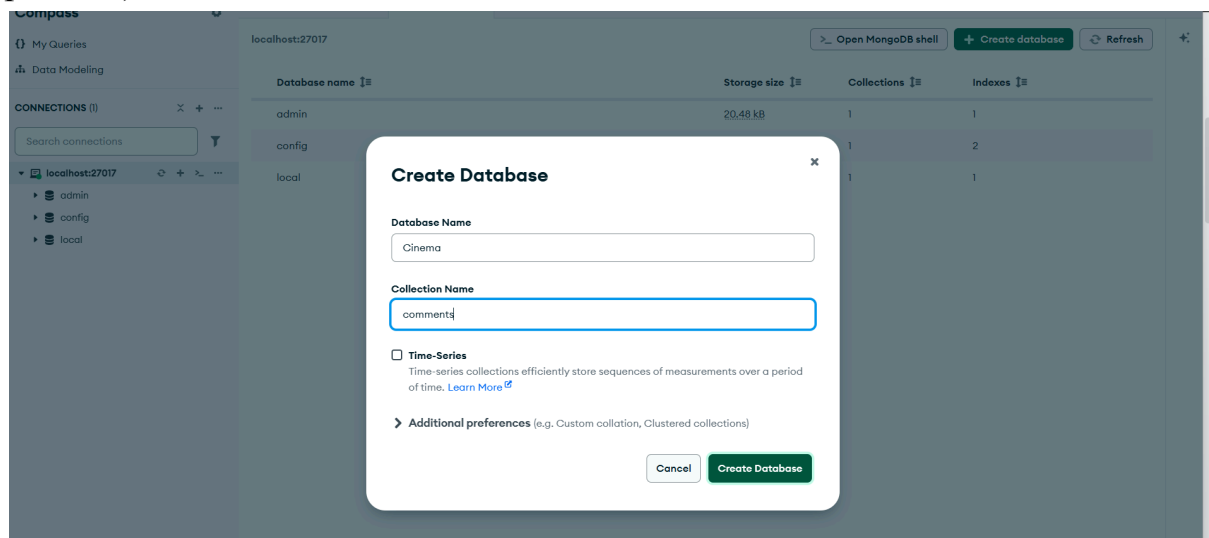
SPRINT 5



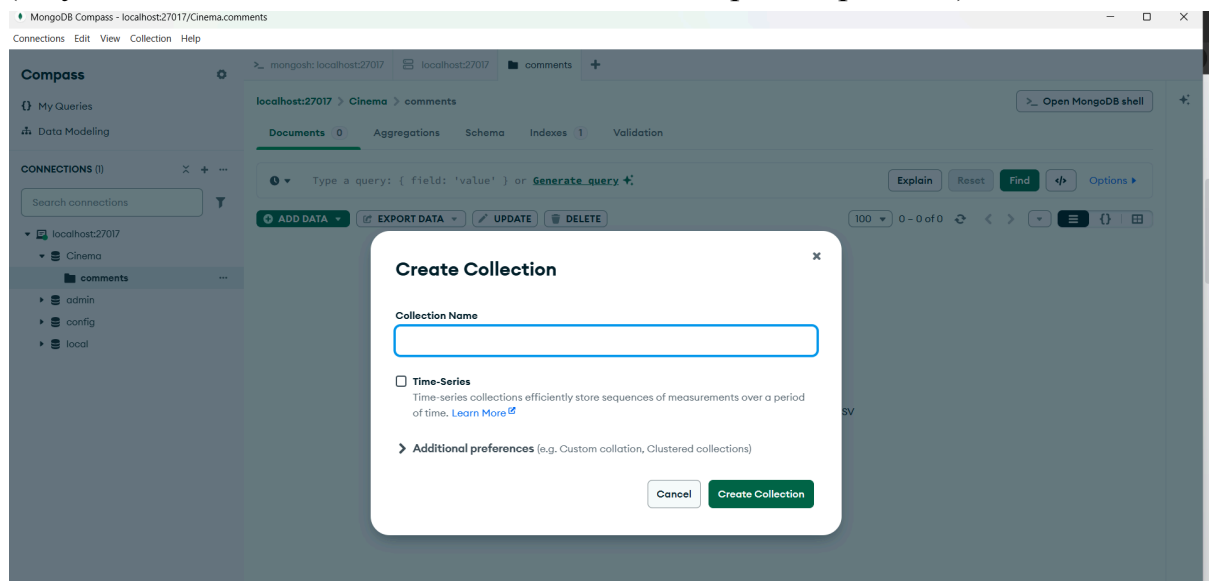
Nivell 1

Crea una base de dades amb MongoDB utilitzant com a colleccions els arxius adjunts.

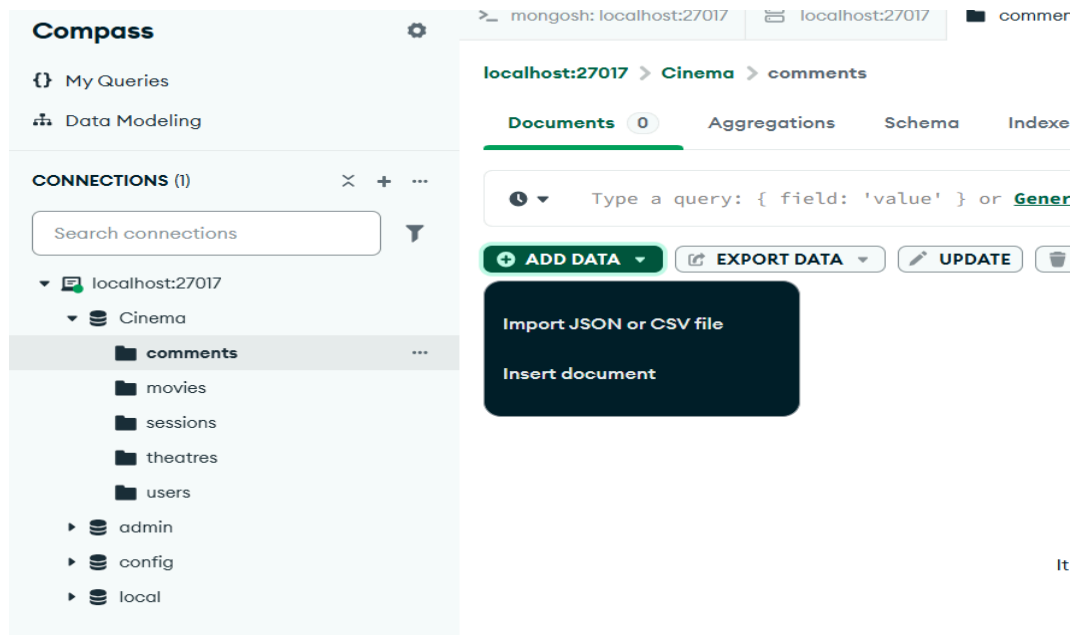
He creado la base de datos mediante la interfaz **MongoDB Compass** y he tenido que insertar una primera colección en el momento de la creación de la base de datos. (Hay el símbolo “+” al lado de “localhost:27017” a la izq. de la pantalla)



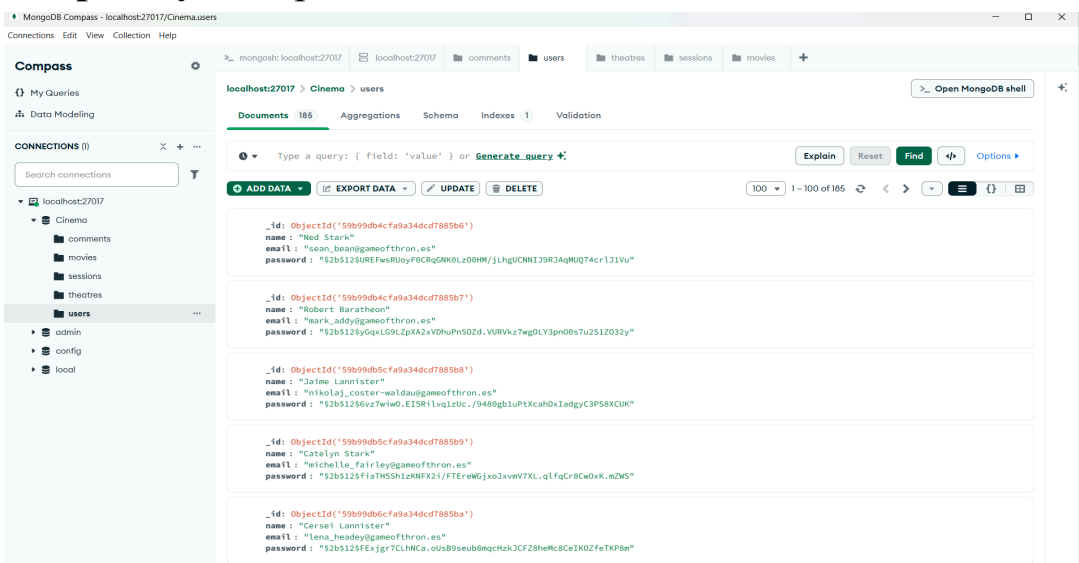
Después he ido **creando**, una por una, las distintas **colecciones** de la base de datos, mediante la interfaz a través de **Create Collection** (Hay el símbolo “+” al lado de “Cinema” a la izq. de la pantalla)



Ahora que tengo las varias colecciones puedo cargar los ficheros con la opción **Add data**:

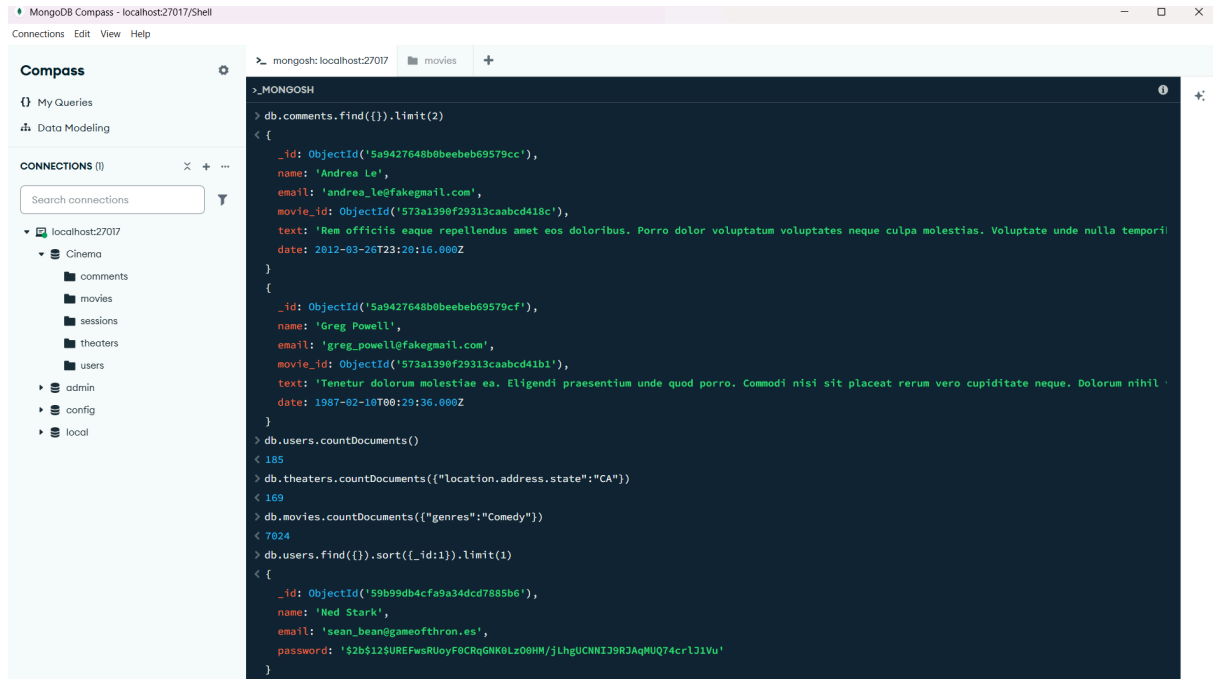


Después de cargar todos los fichero tengo la base de datos “Cinema” **completa y lista** para las consultas :



Exercici 1

- Mostra els 2 primers comentaris que hi ha en la base de dades.
- Quants usuaris tenim registrats?
- Quants cinemes hi ha en l'estat de Califòrnia?
- Quin va ser el primer usuari/ària en registrar-se?
- Quantes pel·lícules de comèdia hi ha en la nostra base de dades?



The screenshot shows the MongoDB Compass interface. The left sidebar displays the database structure with collections: comments, movies, sessions, theaters, and users. The main panel shows the following queries and results:

```
> db.comments.find({}).limit(2)
< [
  {
    _id: ObjectId('5a9427648b0beebe69579cc'),
    name: 'Andrea Le',
    email: 'andrea_le@fakegmail.com',
    movie_id: ObjectId('573a1390f29313caabdc418c'),
    text: 'Rem officii ea repellendus amet eos doloribus. Porro dolor voluptatum voluptates neque culpa molestias. Voluptate unde nulla temporil',
    date: 2012-03-26T23:20:16.000Z
  },
  {
    _id: ObjectId('5a9427648b0beebe69579cf'),
    name: 'Greg Powell',
    email: 'greg_powell@fakegmail.com',
    movie_id: ObjectId('573a1390f29313caabdc41b1'),
    text: 'Tenetur dolorum molestiae ea. Eligendi praesentium unde quod porro. Commodi nisi sit placeat rerum vero cupiditate neque. Dolorum nihil',
    date: 1987-02-10T00:29:36.000Z
  }
]
```

```
> db.users.countDocuments()
< 185
```

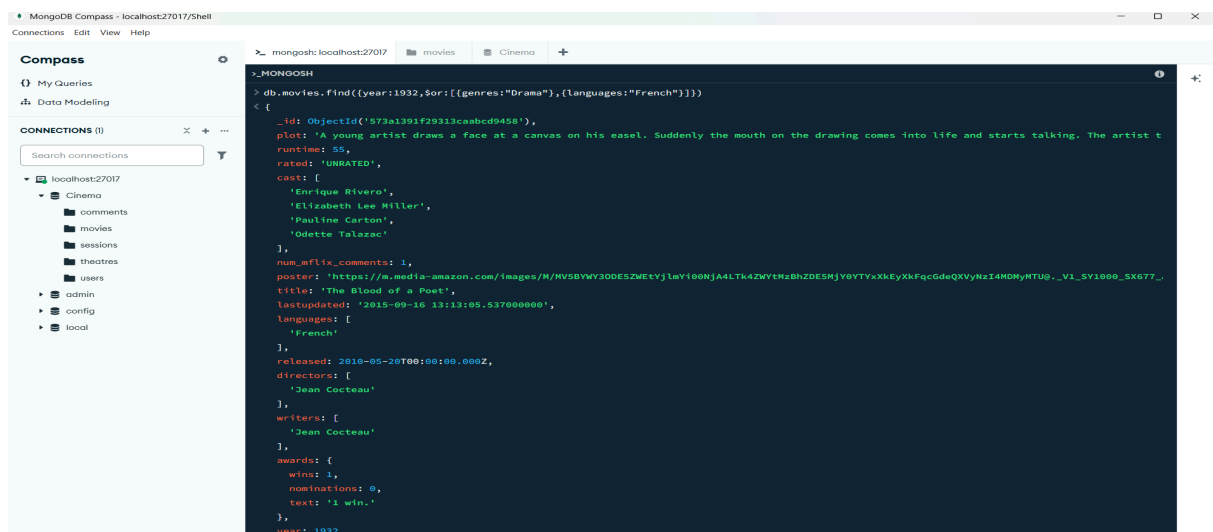
```
> db.theaters.countDocuments({"location.address.state":"CA"})
< 169
```

```
> db.movies.countDocuments({"genres":"Comedy"})
< 7024
```

```
> db.users.find({}).sort({"_id":1}).limit(1)
< [
  {
    _id: ObjectId('59b99db4cf9a34dc7885b6'),
    name: 'Ned Stark',
    email: 'sean_bean@gameofthron.es',
    password: '$2b$12$UREFwsRUoyf8CRqGNK0Lz00HM/jLhGUCHNI39RJAqMUQ74crl31Vu'
  }
]
```

Exercici 2

Mostra'm tots els documents de les pel·lícules produïdes en 1932, però que el gènere sigui drama o estiguin en francès.

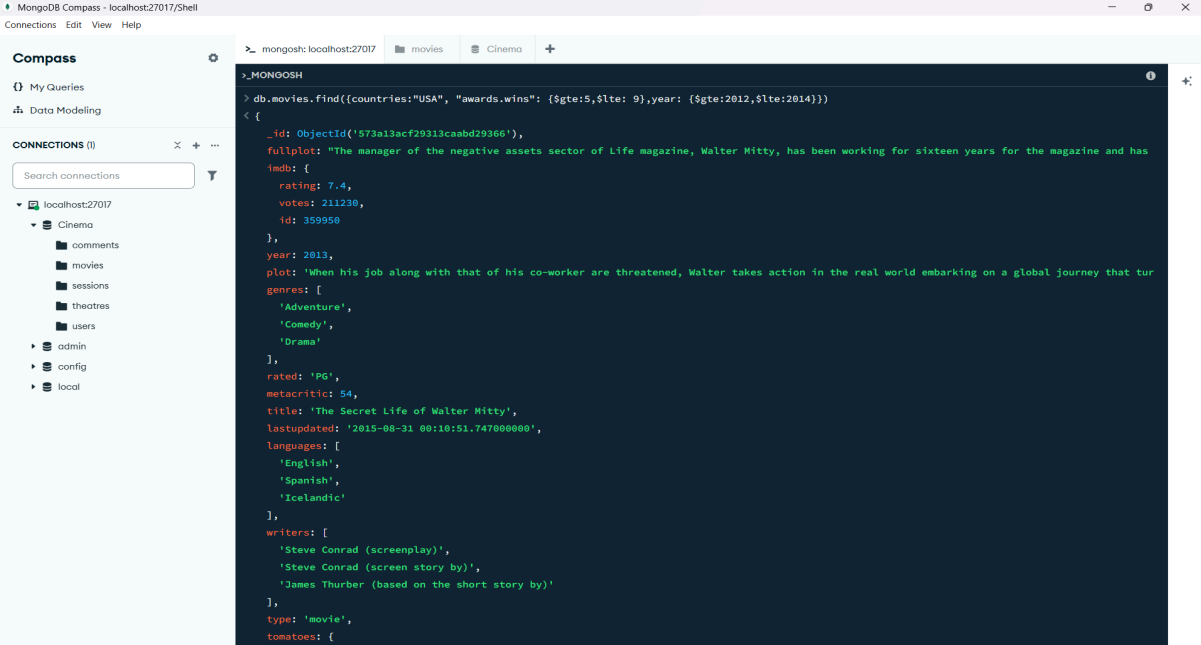


The screenshot shows the MongoDB Compass interface with the following query and result:

```
> db.movies.find({year:1932,$or:[{"genres":"Drama"}, {"languages":"French"}]})
< [
  {
    _id: ObjectId('573a1391f29313caabdc9458'),
    plot: 'A young artist draws a face at a canvas on his easel. Suddenly the mouth on the drawing comes into life and starts talking. The artist t',
    runtime: 55,
    rated: 'UNRATED',
    cast: [
      'Enrique Rivero',
      'Elizabeth Lee Miller',
      'Pauline Carton',
      'Odette Talazac'
    ],
    num_rflx_comments: 1,
    poster: 'https://m.media-amazon.com/images/H/MV5BVMY3ODE5ZWEyYjIwV100NjA4LTk4ZWYtZjY0YTYxXkEYXkFqcGdeQXVyNzI4NDMyMTU0..._V1_SV1000_5X677...',
    title: 'The Blood of a Poet',
    lastupdated: '2015-09-16 13:13:05.537000000',
    languages: [
      'French'
    ],
    released: 2010-05-20T00:00:00.000Z,
    directors: [
      'Jean Cocteau'
    ],
    writers: [
      'Jean Cocteau'
    ],
    awards: {
      wins: 1,
      nominations: 0,
      text: '1 win.'
    },
    year: 1932,
  }
]
```

Exercici 3

Mostra'm tots els documents de pel·lícules estatunidenques que tinguin entre 5 i 9 premis que van ser produïdes entre 2012 i 2014.



The screenshot shows the MongoDB Compass interface. On the left, the 'CONNECTIONS' panel shows a connection to 'localhost:27017' with a database named 'Cinema'. The main panel displays a query in the 'movies' collection:

```
> db.movies.find({countries:"USA", "awards.wins": {$gte:5,$lte: 9},year: {$gte:2012,$lte:2014}})
```

The result is a single document for the movie 'The Secret Life of Walter Mitty':

```
{
  "_id": ObjectId('573a13acf29313caabd29366'),
  "fullplot": "The manager of the negative assets sector of Life magazine, Walter Mitty, has been working for sixteen years for the magazine and has",
  "imdb": {
    "rating": 7.4,
    "votes": 211230,
    "id": 359950
  },
  "year": 2013,
  "plot": "When his job along with that of his co-worker are threatened, Walter takes action in the real world embarking on a global journey that tur",
  "genres": [
    'Adventure',
    'Comedy',
    'Drama'
  ],
  "rated": 'PG',
  "metacritic": 54,
  "title": 'The Secret Life of Walter Mitty',
  "lastupdated": '2015-08-31 00:10:51.747000000',
  "languages": [
    'English',
    'Spanish',
    'Icelandic'
  ],
  "writers": [
    'Steve Conrad (screenplay)',
    'Steve Conrad (screen story by)',
    'James Thurber (based on the short story by)'
  ],
  "type": 'movie',
  "tomatoes": {
```

Commentario:

En el ejercicio 1 he utilizado la función base **find()** para consultar documentos en las colecciones.

He combinado esta función con **limit()** para restringir el número de resultados, con **countDocuments()** para contar registros y con **sort()** para ordenarlos.

Después en el ejercicio 2 he utilizado el operador lógico **\$or**.

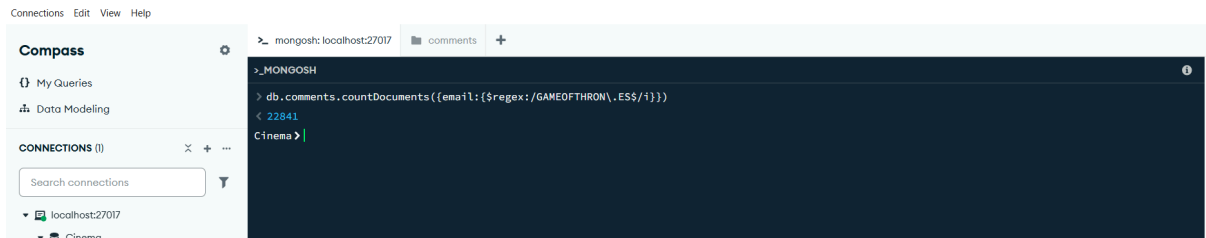
En el ejercicio 3 he visto cómo acceder a campos anidados a través de la sintaxis compuesta así : **“award.wins”**.



Nivell 2

Exercici 1

Compte quants comentaris escriu un usuari/ària que utilitza "GAMEOFTHRON.ES" com a domini de correu electrònic.

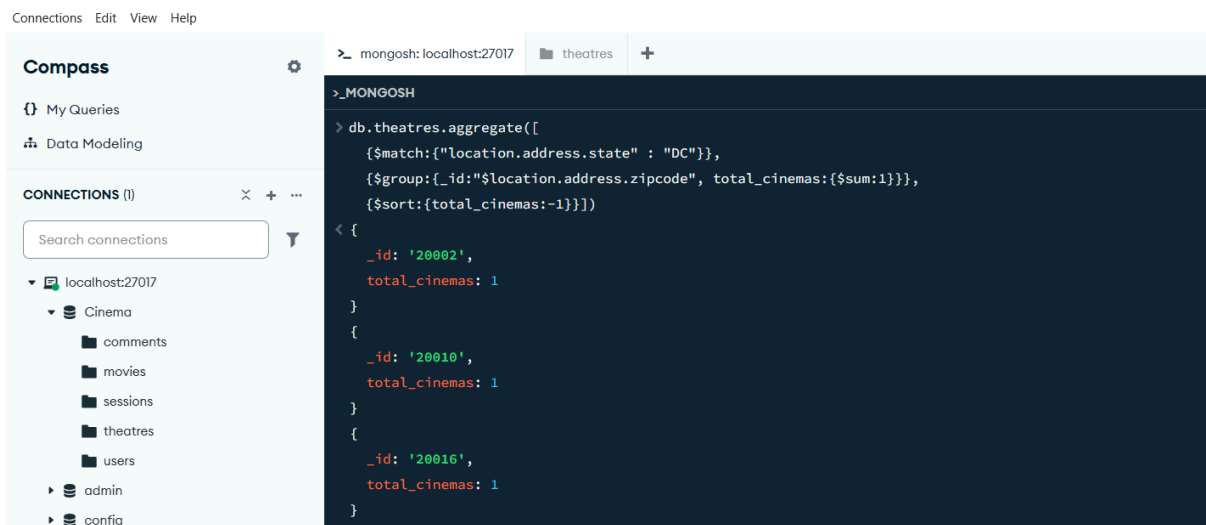


Aquí he utilitzado la función **\$regex** para buscar un texto exacto.

Es importante el uso de los delimitadores “/...../”, el símbolo \$ para indicar donde acaba el texto y la barra invertida “\” para escapar caracteres especiales como el punto. Además, la letra “i” al final indica que la búsqueda es insensible a mayúsculas y minúsculas.

Exercici 2

Quants cinemes hi ha en cada codi postal situats dins de l'estat Washington D. C. (DC)?



Aggregate: es una función avanzada que nos permite insertar una lista de comandos que se ejecutan uno por uno en orden:

Match filtra la colección “theatres” que tengan state = “DC”.

Group agrupa los documentos para el campo zipcode.

(“_id” en este caso funciona como palabra clave)

Sum dentro de cada grupo cuenta cuántos documentos hay

Sort ordena la salida del resultado, -1 da mayor a menor



Nivell 3

Exercici 1

Troba totes les pel·lícules dirigides per John Landis amb una puntuació IMDb (Internet Movie Database) d'entre 7,5 i 8.

```
> db.movies.aggregate([
  {$match:{directors:"John Landis",
    "imdb.rating":{$gte:7.5,$lte:8}}},
  {$sort:{"imdb.rating":-1}}])
< {
  _id: ObjectId('573a1397f29313caabce76f7'),
  plot: 'Jake Blues, just out from prison, puts together his old band to save the Catholic home where he and brother Elwood were raised.',
  genres: [
    'Action',
    'Comedy',
    'Crime'
  ],
  runtime: 133,
  rated: 'R',
  cast: [
    'John Belushi',
    'Dan Aykroyd',
    'James Brown',
    'Cab Calloway'
  ],
  num_mflix_comments: 1,
  poster: 'https://s.media-amazon.com/images/M/MV5BYTdUMDExOGU0N2I3MS00MjY5LWEINTAtYzYzNzIxN2M3OWY1KkEyKkFqcGdeQXVyNzkwMjQ5NzM@._V1_SV1000_SX677_AL',
  title: 'The Blues Brothers',
  fullplot: 'After the release of Jake Blues from prison, he and brother Elwood go to visit "The Penguin", the last of the nuns who raised them in',
  languages: [
    'English'
  ],
  released: 1980-06-20T00:00:00.000Z,
  directors: [
    'John Landis'
  ],
  writers: [
    'Dan Aykroyd',
```

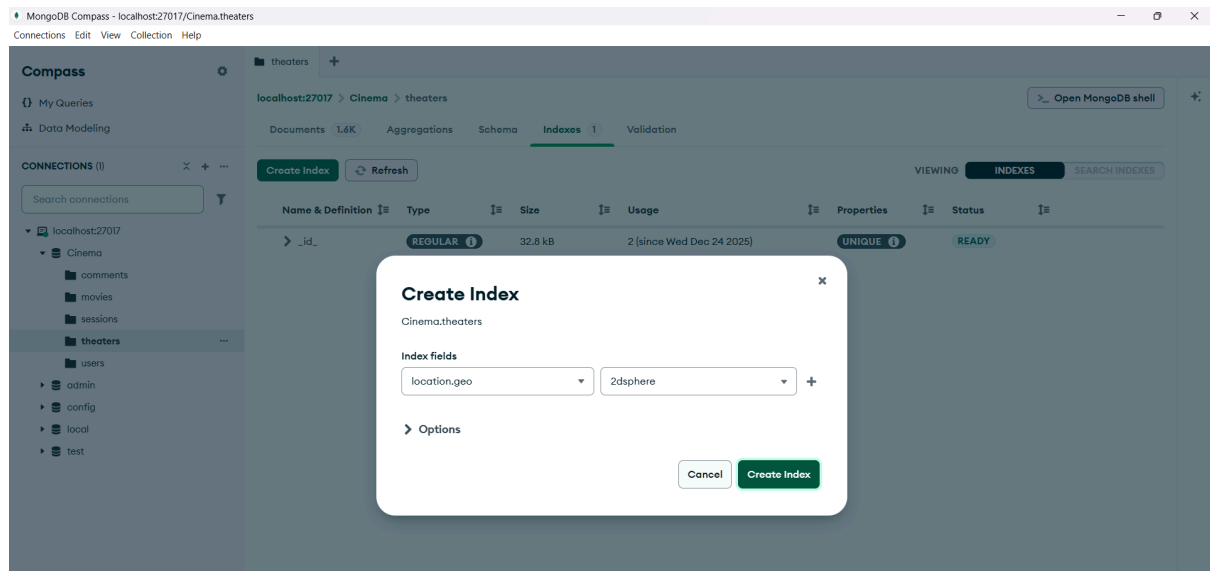
En este ejercicio la clave fue añadir los operadores **\$gte** (\geq) y **\$lte** (\leq) ,
qué son operadores de comparación numérica.

Exercici 2

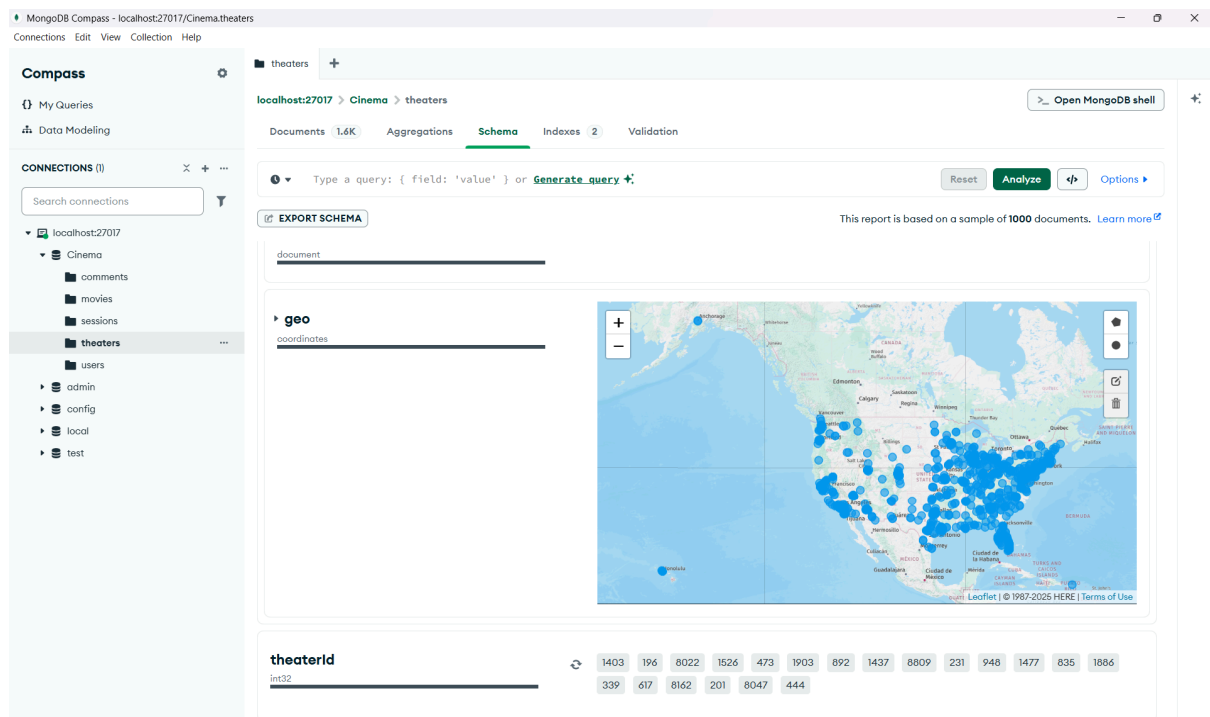
Mostra en un mapa la ubicació de tots els teatres de la base de dades.

Para ver un **mapa**, es recomendable crear un índice **geoespacial**, ya que permite buscar coordenadas de manera rápida incluso con muchos documentos y mejora la eficiencia. Además, garantiza que las consultas geoespaciales como **\$near** o **\$geoWithin** funcionen correctamente.

1.Creación de index:



2.Mapa de los theatres:



NOTA:

He guardado todos los scripts de las consultas en un fichero llamado “Script_Sprint5.js”, que voy a subir al repositorio de GitHub para tener una traza del trabajo realizado.

TODOS LOS PASAJE DE CREACIÓN Y IMPORTACIÓN PUEDEN HACERSE TAMBIÉN CON CÓDIGO, A TRAVÉS DE POWERSHELL.

He tenido que descargar “MongoDB Tools” y añadir el PATH para que me funcionen los comandos MongoDB en el PowerShell:

```
$sysPath = [Environment]::GetEnvironmentVariable("Path","Machine")  
[Environment]::SetEnvironmentVariable("Path", $sysPath + ";C:\Program  
Files\MongoDB\Tools\100\bin", "Machine")
```

Ejemplo para importar collection con “mongoimport”:

1.paso, seleccionamos la ruta donde se encuentran los ficheros

```
cd "C:\Users\eirad\Desktop\IT ACADEMY\NoSql\BaseDeDatos\SPRINT5"
```

2.paso, comando de importación de la colección “users”:

```
mongoimport --db Cinema --collection users --file "users.json"
```

```
PS C:\Users\eirad\Desktop\IT ACADEMY\NoSql\BaseDeDatos\SPRINT5> cd "C:\Users\eirad\Desktop\IT ACADEMY\NoSql\BaseDeDatos\SPRINT5"  
>> mongoimport --db Cinema --collection users --file "users.json"  
2025-12-22T19:44:23.605+0100    connected to: mongodb://localhost/  
2025-12-22T19:44:23.709+0100    185 document(s) imported successfully. 0 document(s) failed to import.  
PS C:\Users\eirad\Desktop\IT ACADEMY\NoSql\BaseDeDatos\SPRINT5> .
```

Nota:

Si la base de datos “Cinema” y la colección “users” no existen se crean al momento de la importación, MongoDB es muy flexible en estas cosas.

Script de todas las importación de colecciones completadas:

```
Administrador: Windows PowerShell  
Windows PowerShell  
Copyright (C) Microsoft Corporation. Todos los derechos reservados.  
  
Instale la versión más reciente de PowerShell para obtener nuevas características y mejoras. https://aka.ms/PSWindows  
  
PS C:\Windows\system32> cd "C:\Users\eirad\Desktop\IT ACADEMY\NoSql\BaseDeDatos\SPRINT5"  
PS C:\Users\eirad\Desktop\IT ACADEMY\NoSql\BaseDeDatos\SPRINT5> mongoimport --db Cinema --collection users --file "users.json"  
2025-12-24T17:05:49.857+0100    connected to: mongodb://localhost/  
2025-12-24T17:05:49.967+0100    185 document(s) imported successfully. 0 document(s) failed to import.  
PS C:\Users\eirad\Desktop\IT ACADEMY\NoSql\BaseDeDatos\SPRINT5> mongoimport --db Cinema --collection comments --file "comments.json"  
2025-12-24T17:07:18.455+0100    connected to: mongodb://localhost/  
2025-12-24T17:07:19.056+0100    50304 document(s) imported successfully. 0 document(s) failed to import.  
PS C:\Users\eirad\Desktop\IT ACADEMY\NoSql\BaseDeDatos\SPRINT5> mongoimport --db Cinema --collection sessions --file "sessions.json"  
2025-12-24T17:08:21.487+0100    connected to: mongodb://localhost/  
2025-12-24T17:08:21.574+0100    1 document(s) imported successfully. 0 document(s) failed to import.  
PS C:\Users\eirad\Desktop\IT ACADEMY\NoSql\BaseDeDatos\SPRINT5> mongoimport --db Cinema --collection movies --file "movies.json"  
2025-12-24T17:08:59.990+0100    connected to: mongodb://localhost/  
2025-12-24T17:09:01.480+0100    23539 document(s) imported successfully. 0 document(s) failed to import.  
PS C:\Users\eirad\Desktop\IT ACADEMY\NoSql\BaseDeDatos\SPRINT5> mongoimport --db Cinema --collection theaters --file "theaters.json"  
2025-12-24T17:10:27.820+0100    connected to: mongodb://localhost/  
2025-12-24T17:10:27.912+0100    1564 document(s) imported successfully. 0 document(s) failed to import.  
PS C:\Users\eirad\Desktop\IT ACADEMY\NoSql\BaseDeDatos\SPRINT5> .
```


TENEMOS VARIOS FORMATO DE FICHEROS QUE PODEMOS IMPORTAR EN MONGODB.

Lista de los más comunes y sus sintaxis básicas:

1.NDJSON (Newline Delimited JSON)

```
mongoimport --db --collection --file datos.ndjson
```

2.JSON Array

```
mongoimport --db --collection --file datos.json --jsonArray
```

3. CSV (Comma Separated Values)

```
mongoimport --db --collection --type csv --file datos.csv --headerline
```

4. TSV (Tab Separated Values)

```
mongoimport --db --collection --type tsv --file datos.tsv --headerline
```

Comentario:

Aunque mis archivos tienen extensión “**.json**”, en realidad están en formato **NDJSON** (un documento por línea). Por eso puedo importarlos sin usar “**--jsonArray**”.

TENEMOS OTRA OPCIÓN PARA CREAR INDEX , ESTA VEZ A TRAVÉS DE MONGOSH.

Codigo para crear index:

