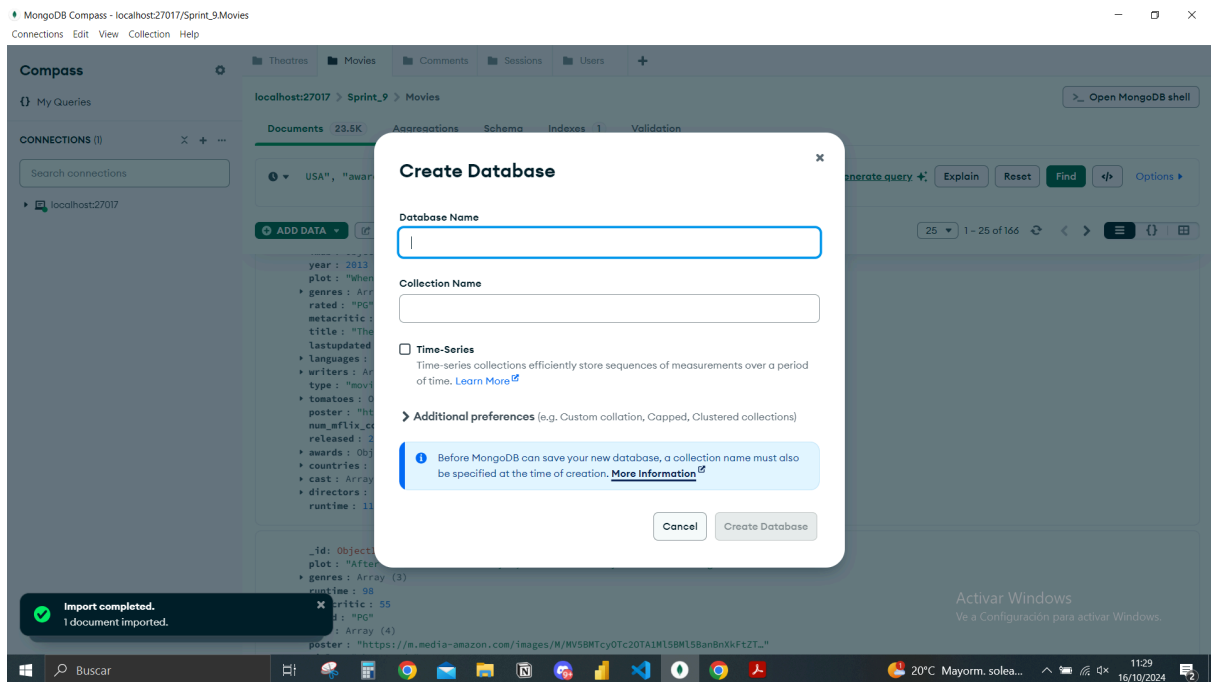


Nivell 1 : Crea una base de dades amb MongoDB utilitzant com a col·leccions els arxius adjunts.

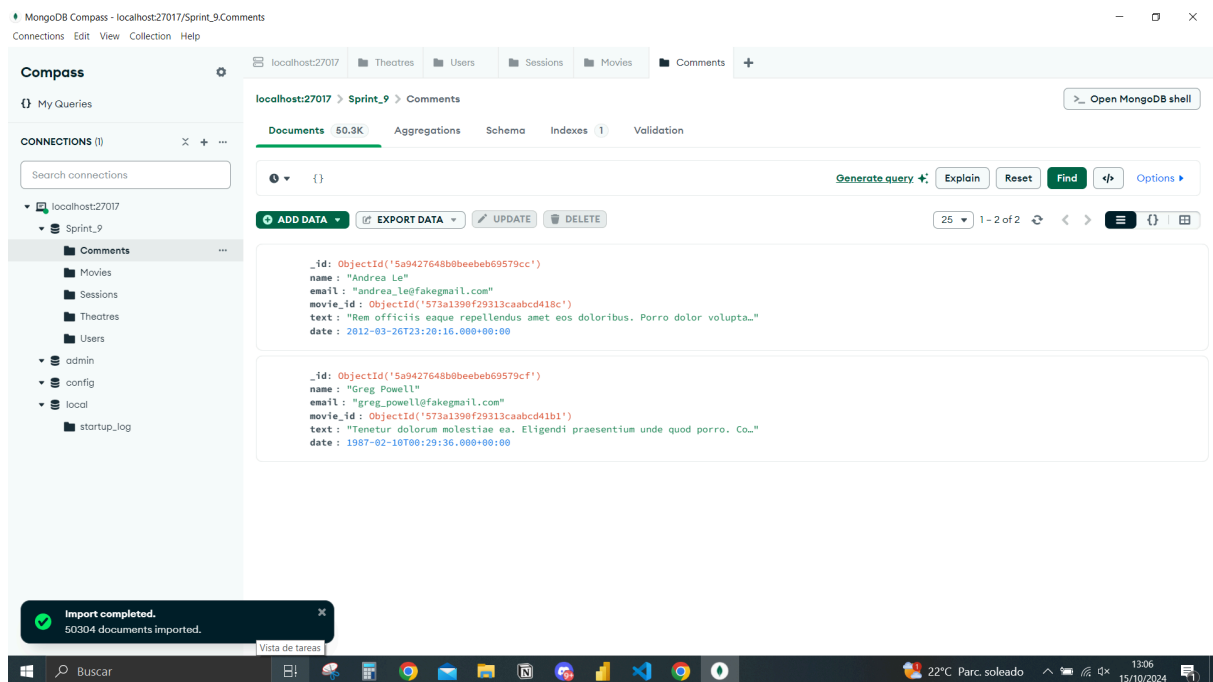


El primer pas es descargarnos todos los archivos json del sprint. Luego, creamos un nuevo database (dando al + en localhost) , llamando sprint9. En ese sprint, crearemos las carpetas o “categorías” necesarias para tener todas las tablas. En cada carpeta, le damos a la opción importar data y seleccionamos el archivo.

Yo voy a hacer todos los comandos vía el MongoDB Compass, ya que considero que es una manera fácil y ágil para poder trabajar con tus bases de datos.

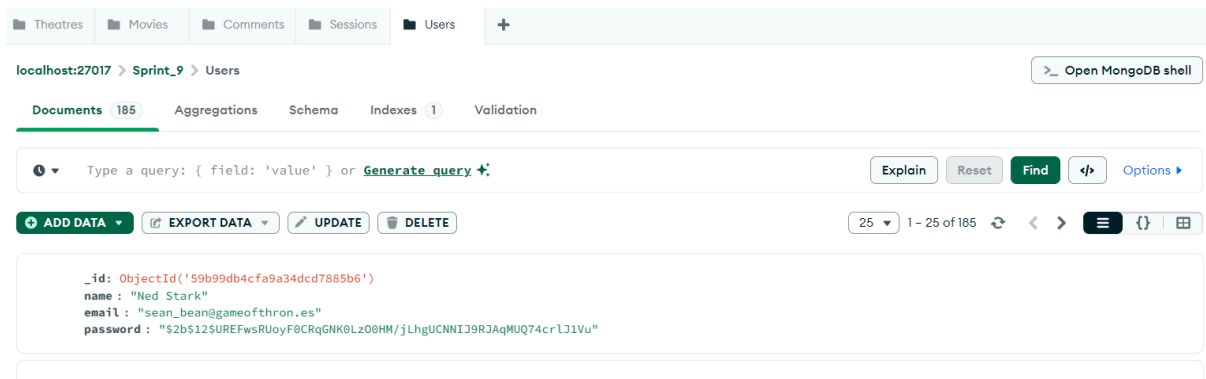
Exercici 1

- Mostra els 2 primers comentaris que hi ha en la base de dades.



Para este ejercicio, usamos el comando {} para seleccionar todos los datos de la categoría comments. Después, como solo queremos ver los dos primeros, le damos a options, y establecemos limit 2.

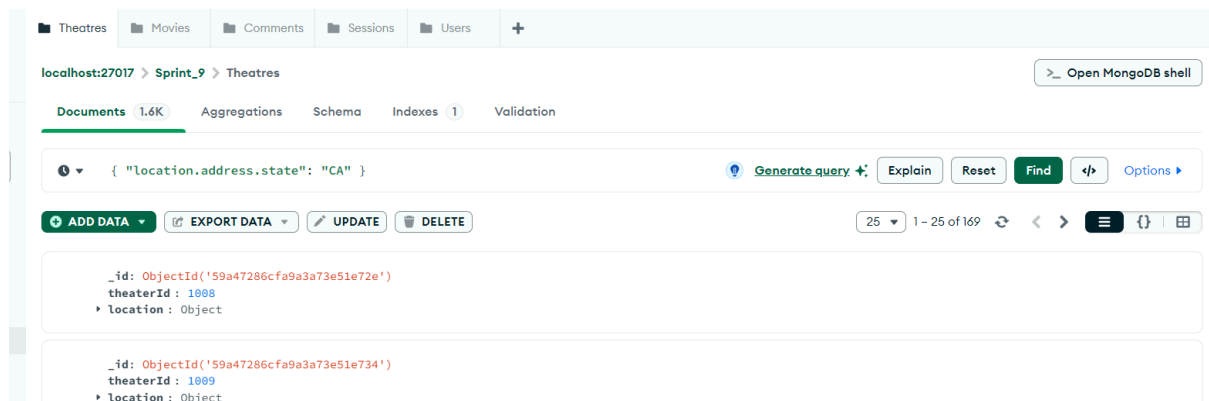
- Quants usuaris tenim registrats?



The screenshot shows the MongoDB Compass interface. At the top, there's a navigation bar with tabs for 'Theatres', 'Movies', 'Comments', 'Sessions', and 'Users'. Below this, the breadcrumb path is 'localhost:27017 > Sprint_9 > Users'. The 'Documents' tab is selected, showing 185 documents. The query bar is empty, and the 'Find' button is highlighted. Below the query bar, there are buttons for 'ADD DATA', 'EXPORT DATA', 'UPDATE', and 'DELETE'. The document list shows a sample document with fields: '_id', 'name', 'email', and 'password'.

Para ver la totalidad de los usuarios, usamos el comando `select all : {}` .
Observamos que hay 185 usuarios en total.

- Quants cinemes hi ha en l'estat de Califòrnia?

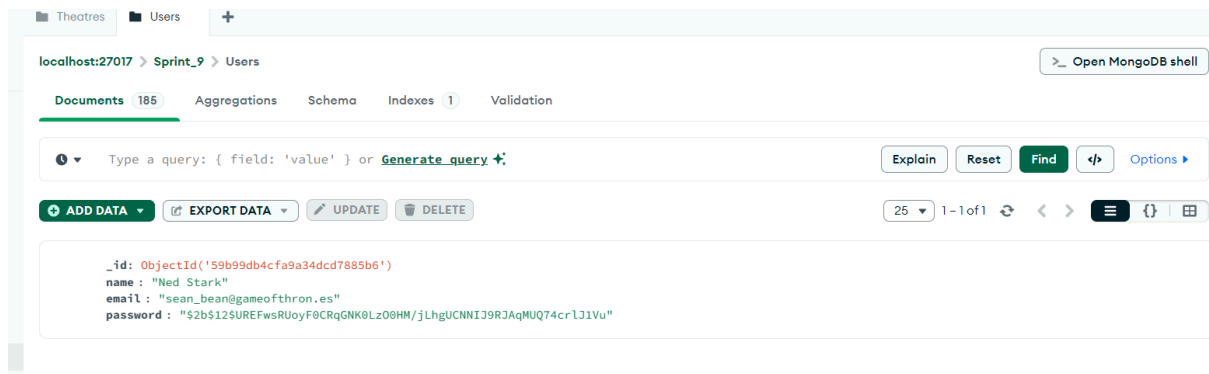


The screenshot shows the MongoDB Compass interface. At the top, there's a navigation bar with tabs for 'Theatres', 'Movies', 'Comments', 'Sessions', and 'Users'. Below this, the breadcrumb path is 'localhost:27017 > Sprint_9 > Theatres'. The 'Documents' tab is selected, showing 1.6K documents. The query bar contains the filter `{ "location.address.state": "CA" }`. The 'Find' button is highlighted. Below the query bar, there are buttons for 'ADD DATA', 'EXPORT DATA', 'UPDATE', and 'DELETE'. The document list shows two sample documents with fields: '_id', 'theaterId', and 'location'.

Para este ejercicio, es imprescindible analizar previamente la base de datos, ya que uno en un primer lugar puede pensar en que el filtro será : `location : California`. En cambio, observamos que las localidades están en siglas, por lo cual, California pasa a estar representada por “ CA”. Usamos el comando : `{“location.addres.state”: “CA”}` para observar los cines en california.

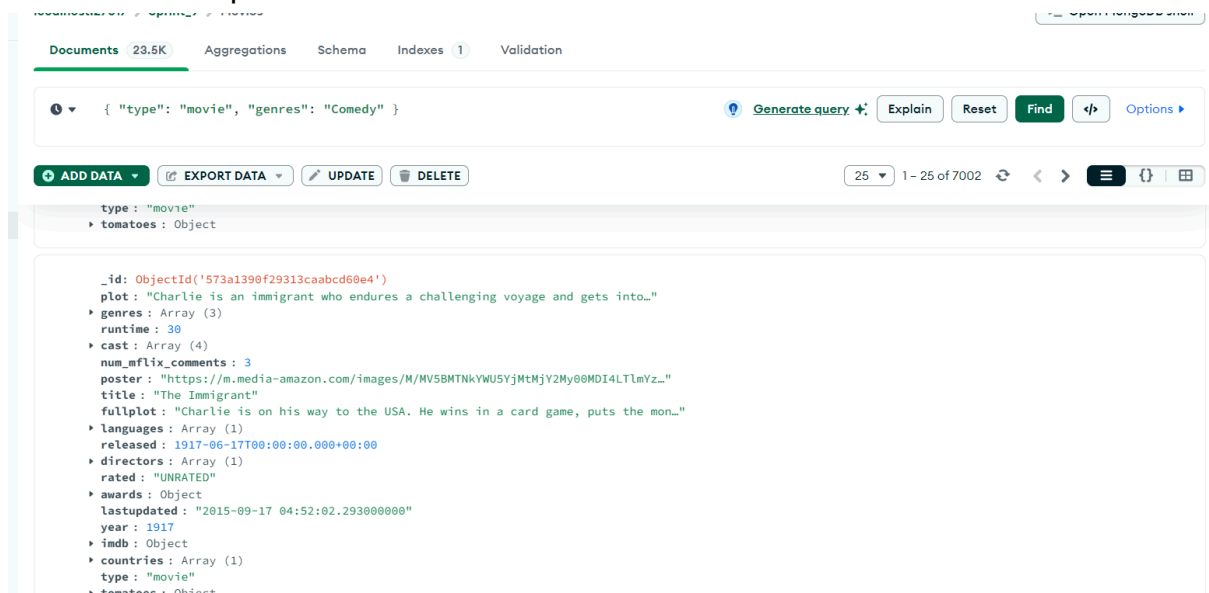
Se observa que hay 169 cinemas en la ciudad.

- Quin va ser el primer usuari/ària en registrar-se?



Para este, usamos el comando `select all {}` y filtramos `limit 1`. Nos muestra el usuario llamado Ned Stark.

- Quantes pel·lícules de comèdia hi ha en la nostra base de dades?



Para este ejercicio, usamos un doble filtro, donde busco por género y por tipo. Usamos el comando siguiente : `{"type":"movie", "genres": "Comedy"}`. Observamos que hay un total de 7002 películas de comedia.

Exercici 2

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

localhost:27017 > Sprint_9 > Movies Open MongoDB shell

Documents 23.5K Aggregations Schema Indexes 1 Validation

Generate aggregation Explain Export Run Options

Untitled - modified SAVE CREATE NEW EXPORT TO LANGUAGE PREVIEW STAGES TEXT WIZARD

Stage 1 \$match

```
1 /**
2  * query: The query in MQL.
3  */
4 {
5   type : "movie"
6 }
```

Output after \$match stage (Sample of 10 documents)

```
{
  "_id": "ObjectId('573a1398f29313caabcd4135')",
  "plot": "Three men hammer on an anvil and pass a bottle of beer around.",
  "genres": Array (1),
  "runtime": 1,
  "cast": Array (2),
  "num_mflix_comments": 1,
  "title": "Blacksmith Scene",
  "fullplot": "A stationary camera looks at a"
}
```

```
{
  "_id": "ObjectId('573a1398f29313caabcd42e8')",
  "plot": "A group of bandits stage a brazen train hold-up, only to find a determ...",
  "genres": Array (2),
  "runtime": 11,
  "cast": Array (4),
  "poster": "https://m.media-amazon.com/images/M/MV5BMTU3NjE5NzYtY...",
  "title": "The Great Train Robbery"
}
```

Stage 2 \$match

```
1 /**
2  * query: The query in MQL.
3  */
4 {
5   year: 1932
6 }
```

Output after \$match stage (Sample of 10 documents)

```
{
  "_id": "ObjectId('573a1391f29313caabcd9458')",
  "plot": "A young artist draws a face at a canvas on his easel. Suddenly the mou...",
  "runtime": 55,
  "rated": "UNRATED"
}
```

```
{
  "_id": "ObjectId('573a1392f29313caabcd99a3')",
  "plot": "Junta is hated by the people in the village where she lives, especiall...",
  "genres": Array (3),
  "runtime": 85
}
```

localhost:27017 > Sprint_9 > Movies Open MongoDB shell

Documents 23.5K Aggregations Schema Indexes 1 Validation

Generate aggregation Explain Export Run Options

Untitled - modified SAVE CREATE NEW EXPORT TO LANGUAGE PREVIEW STAGES TEXT WIZARD

Stage 3 \$match

```
1 /**
2  * query: The query in MQL.
3  */
4 {
5   $or: [
6     {genres: "Drama"},
7     {languages: "French"}
8   ]
9 }
```

Output after \$match stage (Sample of 10 documents)

```
{
  "_id": "ObjectId('573a1391f29313caabcd9458')",
  "plot": "A young artist draws a face at a canvas on his easel. Suddenly the mou...",
  "runtime": 55,
  "rated": "UNRATED",
  "cast": Array (4),
  "num_mflix_comments": 1,
  "poster": "https://m.media-amazon.com/images/M/MV5BYWY3ODESZWEtY...",
  "title": "The Blue Light"
}
```

```
{
  "_id": "ObjectId('573a1392f29313caabcd99a3')",
  "plot": "Junta is hated by the people in the village where she lives, especiall...",
  "genres": Array (3),
  "runtime": 85,
  "cast": Array (4),
  "poster": "https://m.media-amazon.com/images/M/MV5BNTQ1NTMzMTQ0T...",
  "title": "The Blue Light"
}
```

Stage 4 \$count

```
1 /**
2  * Provide the field name for the count.
3  */
4 'string'
```

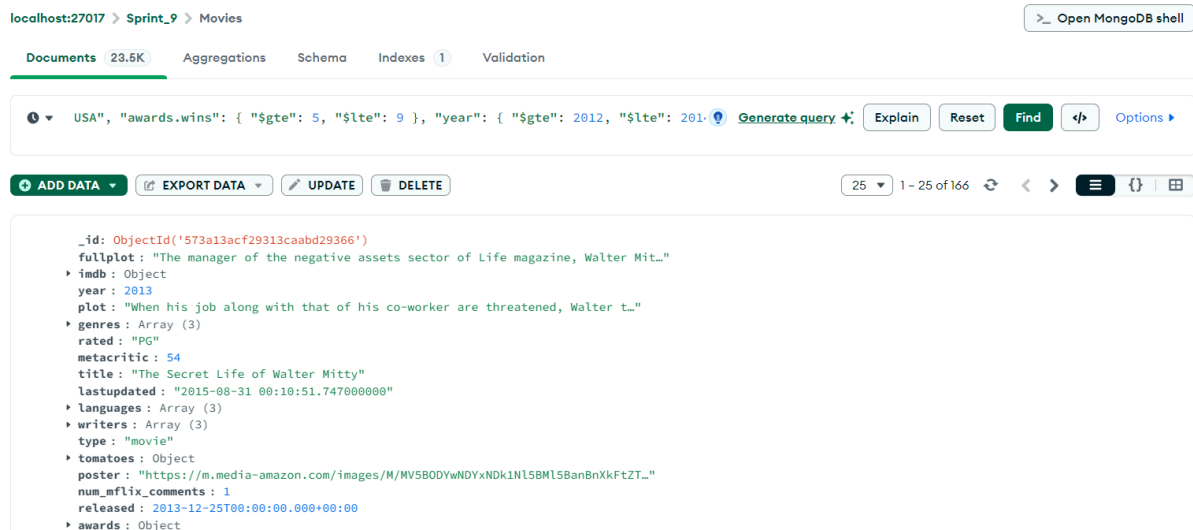
Output after \$count stage (Sample of 1 document)

```
{
  "string": 18
}
```

Para este ejercicio, se ha usado otro método para trabajar con la base de datos, usando las agregaciones llamadas “stages”. Vendría a ser cómo crear un filtro por stage. Entonces, vemos que hemos creado el primer stage para buscar en la base de datos, que sea la tipología película. Acto seguido, un nuevo stage donde se filtra por año, después se filtra por género e idioma. Por último, en vez de hacer un match (que se usa para buscar coincidencias en la base de datos), usamos el count para hacer el recuento, ya que el ejercicio requiere mostrar resultados filtrados. Se observa que hay un total de 18 películas con el filtro requerido.

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. The breadcrumb navigation at the top indicates the path: localhost:27017 > Sprint_9 > Movies. The 'Documents' tab is selected, showing 23.5K documents. The query filter is set to: `USA", "awards.wins": { "$gte": 5, "$lte": 9 }, "year": { "$gte": 2012, "$lte": 2014 }`. Below the query bar, there are buttons for 'ADD DATA', 'EXPORT DATA', 'UPDATE', and 'DELETE'. The results list shows 1 document, 'The Secret Life of Walter Mitty', with details such as year (2013), genres (Comedy, Drama, Romance), and awards (1).

Aquí, usamos el filtro conjunto utilizando código, en vez de por stages. El código usado es el siguiente : `{ countries: "USA", "awards.wins": { $gte: 5, $lte: 9 }, year: { $gte: 2012, $lte: 2014 } }`.

Se observa que hay un total de 166 resultados.