

Sprint 9: Base de datos No Relacionales

DESCRIPCIÓN

Trabajaremos con una base de datos que contiene colecciones relacionadas con una aplicación de entretenimiento cinematográfico:

Permite el uso del texto de forma anónima para mejorar el servicio de traducción.

users: Almacena información de usuarios/as, incluyendo nombres, emails y contraseñas cifradas.

theatres: Contiene datos de cines, como ID, ubicación (dirección y coordenadas geográficas).

sesiones: Guarda sesiones de usuario, incluyendo ID de usuario y tokens JWT para la autenticación.

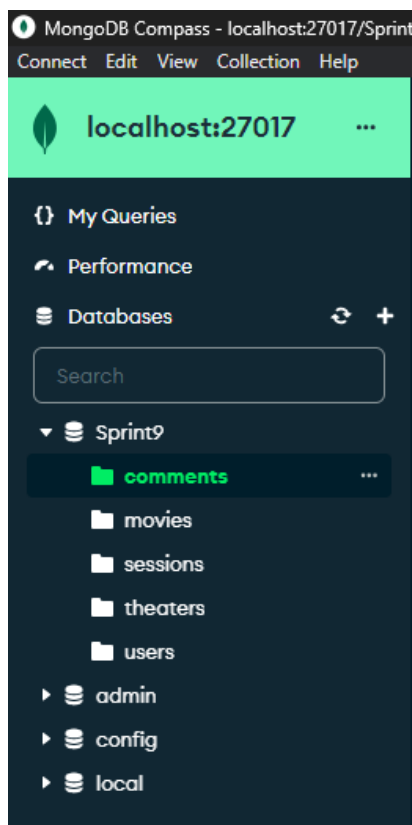
movies: Incluye detalles de películas, como trama, géneros, duración, elenco, comentarios, año de lanzamiento, directores, clasificación y premios.

comments: Almacena comentarios de usuarios/as sobre películas, con información del autor/a del comentario, ID de la película, texto del comentario y la fecha.

Realizarás algunas consultas que te pide el cliente/a, quien está midiendo si serás capaz o no de hacerte cargo de la parte analítica del proyecto vinculado con su base de datos.

NIVEL 1

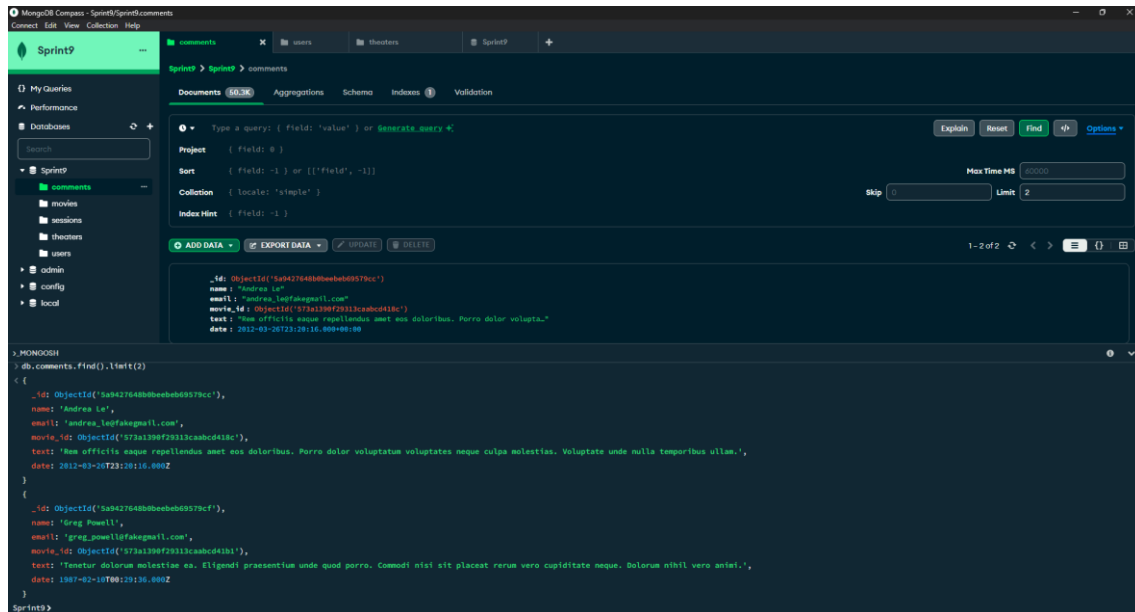
Crea una base de datos con MongoDB utilizando como colecciones los archivos adjuntos.



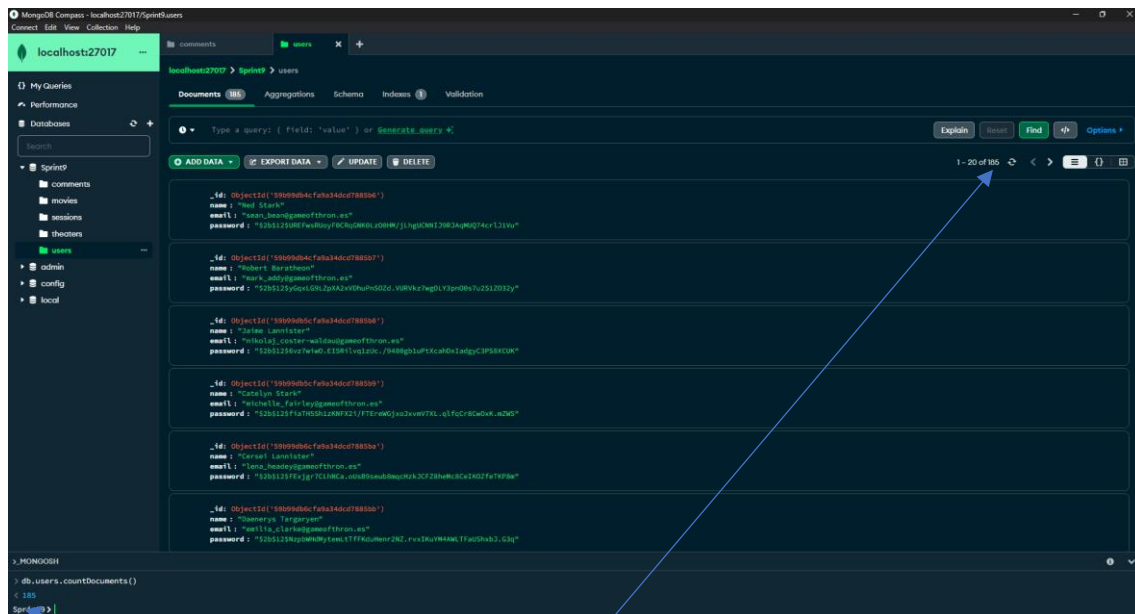
Sprint 9: Base de datos No Relacionales

Ejercicio 1

- Muestra los 2 primeros comentarios que aparecen en la base de datos.



- ¿Cuántos usuarios tenemos registrados?



En ambos casos se muestra el total de documentos almacenados en la colección, se trata de Id's únicos

Sprint 9: Base de datos No Relacionales

- ¿Cuántos cines existen en el estado de California?

The screenshot shows the MongoDB Compass interface. The left sidebar displays the database structure with 'Sprint9' selected, containing collections 'comments', 'movies', 'sessions', 'theaters', 'users', 'admin', 'config', and 'local'. The main panel shows a query for the 'theaters' collection: `{ "location.address.state": "CA" }`. The results are displayed in a table with columns: `_id`, `theaterId`, `location`, `address`, `city`, `state`, `zipCode`, and `geo`. The first row shows a theater in San Francisco. The bottom status bar indicates '1 - 20 of 169' results. A blue arrow points from the text 'Existen 169 salas de cine en el estado de California' to the '169' in the status bar. Another blue arrow points from the same text to the MongoDB shell command at the bottom: `> db.theaters.find({"location.address.state":"CA"}).count()`.

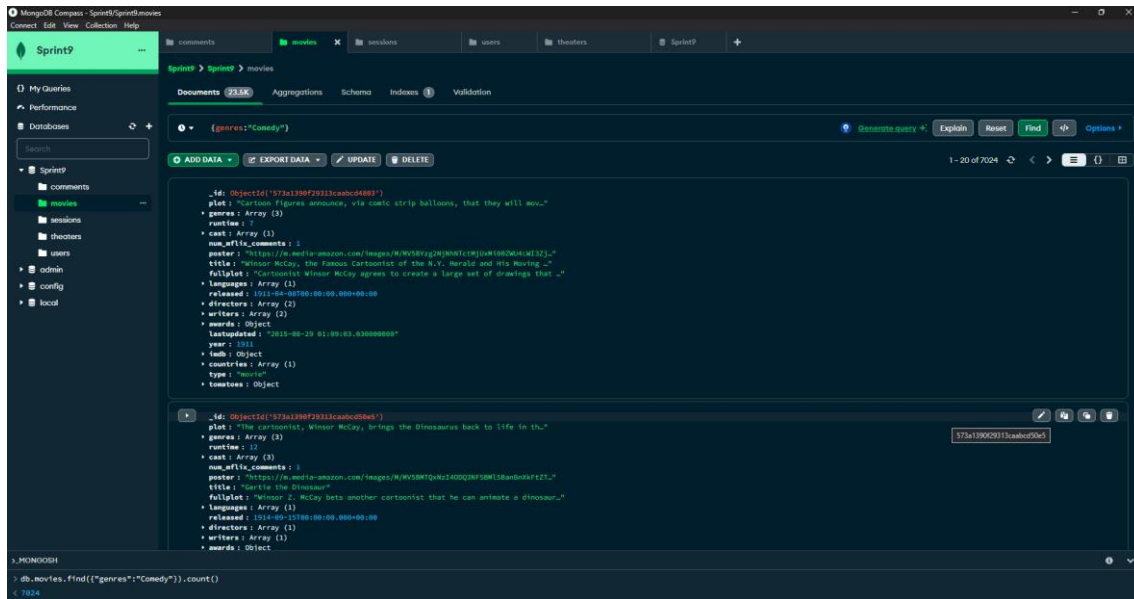
Existen 169 salas de cine en el estado de California

- ¿Cuál fue el primer usuario en registrarse?

The screenshot shows the MongoDB Compass interface. The left sidebar displays the database structure with 'Sprint9' selected, containing collections 'comments', 'movies', 'sessions', 'theaters', 'users', 'admin', 'config', and 'local'. The main panel shows a query for the 'users' collection: `{ "type": "user", "value": "GENERATE_UNIQUE" }`. The results are displayed in a table with columns: `_id`, `name`, `email`, and `password`. The first row shows a user named 'Red Stark'. The bottom status bar indicates '1 - 1 of 1' results. A blue arrow points from the text '¿Cuál fue el primer usuario en registrarse?' to the first user in the results table.

Sprint 9: Base de datos No Relacionales

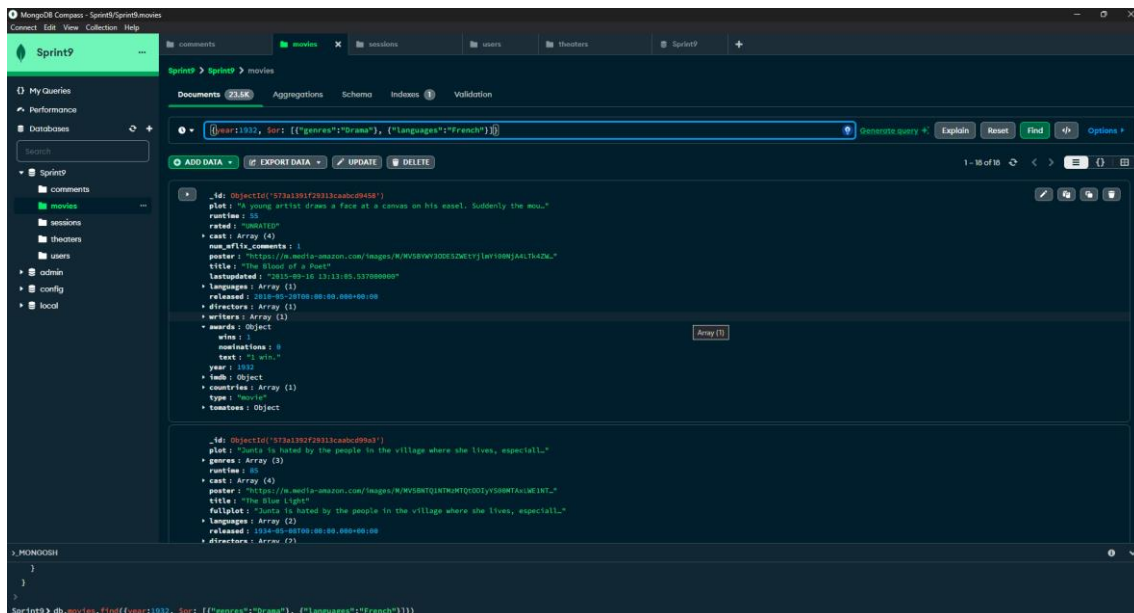
- ¿Cuántas películas de comedia existen en nuestra base de datos?



Existen 7024 películas en el género de comedia.

Ejercicio 2

Muéstrame todos los documentos de las películas producidas en 1932, pero que el género sea drama o estén en francés.

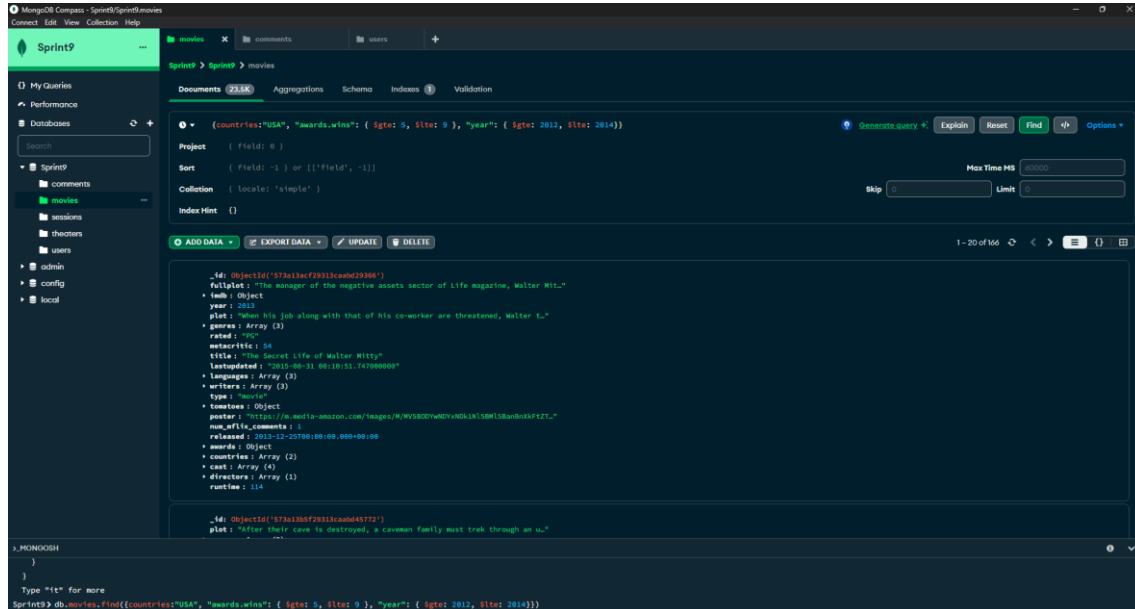


El resultado muestra 18 documentos.

Sprint 9: Base de datos No Relacionales

Ejercicio 3

Muéstrame todos los documentos de películas estadounidenses que tengan entre 5 y 9 premios que fueron producidas entre 2012 y 2014.

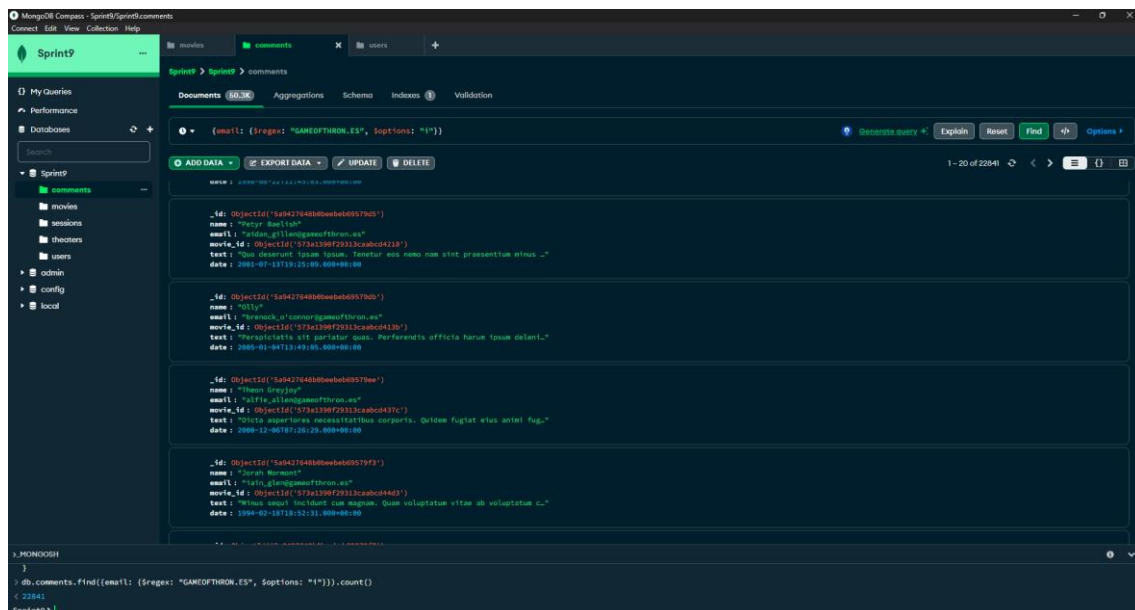


Como resultado se muestran 166 documentos

Nivel 2

Ejercicio 1

Cuenta cuántos comentarios escribe un usuario que utiliza "GAMEOFTHRON.ES" como dominio de correo electrónico.

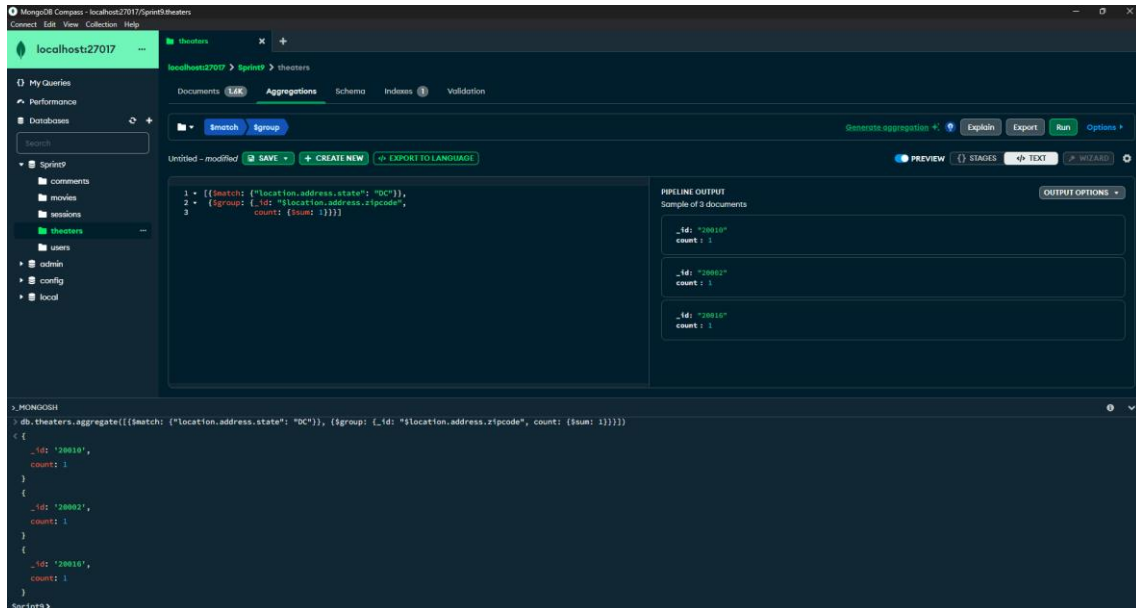


La query arroja un resultado de 22841 documentos en la colección “comments” que cumplen la condición

Sprint 9: Base de datos No Relacionales

Ejercicio 2

¿Cuántos cines existen en cada código postal situados dentro del estado Washington DC (DC)?



The screenshot shows the MongoDB Compass interface. The left sidebar displays the database structure with collections like 'comments', 'movies', 'sessions', 'theaters', 'users', 'admin', 'config', and 'local'. The main panel is set to the 'theaters' collection. The 'Aggregations' tab is active, showing a pipeline with three stages:

1. `{ $match: { "location.address.state": "DC" } }`
2. `{ $group: { "_id": "$location.address.zipcode", count: { $sum: 1 } } }`
3. `{ $sort: { count: -1 } }`

The 'PIPELINE OUTPUT' section shows a sample of 3 documents:

```
{ "_id": "20010", "count": 1 }
{ "_id": "20002", "count": 1 }
{ "_id": "20016", "count": 1 }
```

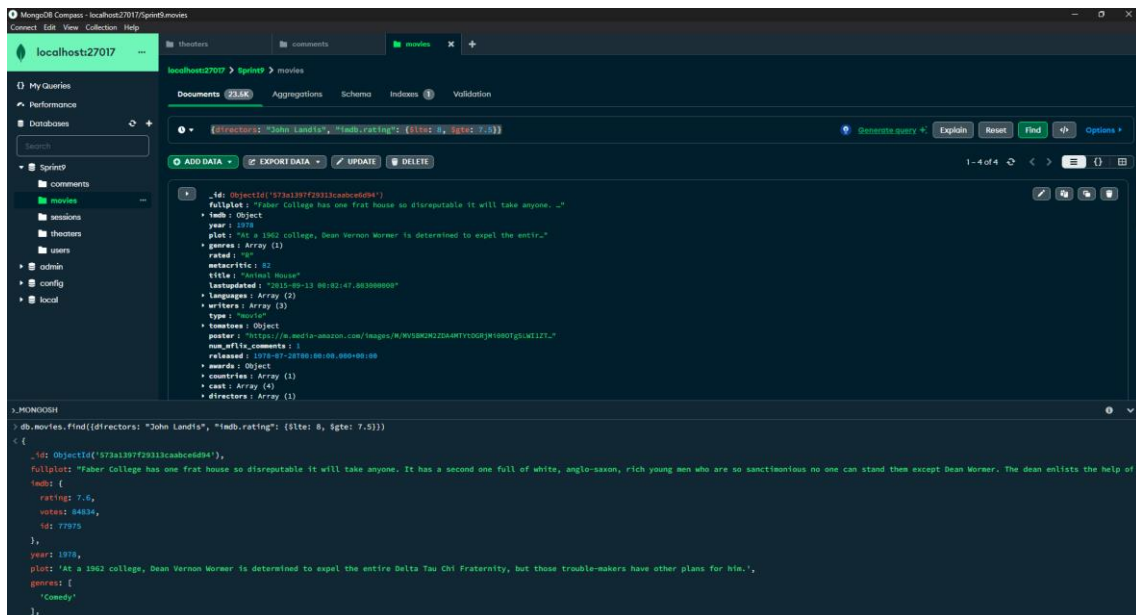
The command window at the bottom shows the following query:

```
> db.theaters.aggregate([{$match: {"location.address.state": "DC"}}, {$group: {_id: "$location.address.zipcode", count: {$sum: 1}}}], {pretty: true})
```

Nivel 3

Ejercicio 1

Encuentra todas las películas dirigidas por John Landis con una puntuación IMDb (Internet Movie Database) de entre 7,5 y 8.



The screenshot shows the MongoDB Compass interface with the 'movies' collection selected. The 'Documents' tab is active, displaying a list of documents. The query bar shows the following query:

```
{ "directors": "John Landis", "imdb.rating": { "$gte": 7.5, "$lte": 8 } }
```

The command window at the bottom shows the following query:

```
> db.movies.find({'directors': 'John Landis', 'imdb.rating': {'$gte': 7.5, '$lte': 8}}, {pretty: true})
```

Como resultado se muestran 4 documentos

Sprint 9: Base de datos No Relacionales

Ejercicio 2

Muestra en un mapa la ubicación de todos los teatros de la base de datos.

