



Bases de datos No Relacionales S5.01

Elian Daghoum Dorado



Índice

Nivel 1	03
Nivel 2	14
Nivel 3	16



Nivel 1

Creación de Base de datos

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

Empezamos creando el database y añadimos las colecciones.

Comments

Storage size:
6.93 MB

Documents:
50 K

Movies

Storage size:
21.55 MB

Documents:
24 K

Avg. document size:
1.60 kB

Indexes:
1

Total index size:
364.54 kB

Sessions

Storage size:
20.48 kB

Documents:
1

Avg. document size:
540.00 B

Indexes:
1

Total index size:
20.48 kB

Theaters

Storage size:
147.46 kB

Documents:
1.6 K

Avg. document size:
223.00 B

Indexes:
1

Total index size:
32.77 kB

Users

Storage size:
36.86 kB

Documents:
185

Avg. document size:
159.00 B

Indexes:
1

Total index size:
20.48 kB

Create Database

Database Name

S4

Collection Name

Movies

☐ Time-Series

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

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

Cancel

Create Database



Nivel 1

Ejercicio 1

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

Con establecer un limit de 2 es suficiente ya que Mongo los ordena de forma natural así.

▼ Stage 1 \$limit



```
1  ▼ /**
2
3  * Muestra los 2 primeros comentarios
4  *   que hay en la base de datos.
5  */
6
7  2
```

Output after `$limit` stage (Sample of 2 documents)

```
_id: ObjectId('5a9427648b0beeb69579cc')
name: "Andrea Le"
email: "andrea_le@fakegmail.com"
movie_id: ObjectId('573a1390f29313caabcd418...')
text: "Rem officiis eaque repellendus amet
      eos doloribus. Porro dolor volupta..."
date: 2012-03-26T23:20:16.000+00:00
```

```
_id: ObjectId('5a9427648b0beeb69579cf')
name: "Greg Powell"
email: "greg_powell@fakegmail.com"
movie_id: ObjectId('573a1390f29313caabcd41b...')
text: "Tenetur dolorum molestiae ea.
      Eligendi praesentium unde quod porro.
      Co..."
date: 1987-02-10T00:29:36.000+00:00
```



Nivel 1

Ejercicio 1

¿Cuántos usuarios tenemos registrados?

Hacemos un Count para contar los registros de la colección.

▼ Stage 2 \$count ▼



```
1
2 ▼ /**
3
4   * ¿Cuántos usuarios tenemos registrados?
5
6   */
7 'totalUsuarios'
```

Output after `$count` [🔗](#) stage (Sample of 1 document)

totalUsuarios : 50304



Nivel 1


Ejercicio 1

¿Cuántos cines hay en el estado de California?

1/ Filtramos los cines que estén en california

▼ Stage 1 \$match 

```
1 ▼ /**
2   * Cines de California
3   */
4 ▼ {
5   "location.address.state": "CA"
6 }
7
8
```

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

```
_id: ObjectId('59a47286cfa9a3a73e51e72e')
theaterId: 1008
▶ location: Object
```

```
_id: ObjectId('59a47286cfa9a3a73e51e734')
theaterId: 1009
▶ location: Object
```



Nivel 1

Ejercicio 1

¿Cuántos cines hay en el estado de California?

2/ Sobre ese filtro, aplicamos un count

▼ Stage 2 \$count



```
1  ▼ /**
2
3    *Conteo de los cines de california
4    */
5    'totalCinemesCA'
```



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

`totalCinemesCA : 169`



Nivel 1

Ejercicio 1

¿Cuál fue el primer usuario/a en registrarse?

1/ Creamos un campo nuevo llamado registro para obtener la fecha y sacando esa información del campo id

▼ Stage 1 \$addFields

```
1  ▼ /**
2    * creamos un campo nuevo que
3    * saca la fecha de registro del id
4    */
5  ▼ {
6    registro: { $toDate: "$_id" }
7  }
8
9
10
```

Output after [\\$addFields](#) stage (Sample of 10 documents)

```
_id: ObjectId('59b99db4cfa9a34dcd7885b6')
name: "Ned Stark"
email: "sean_bean@gameofthron.es"
password: "$2b$12$UREFwsRUoyF0CRqGNK0Lz00HM..."
registro: 2017-09-13T21:05:56.000+00:00
```

```
_id: ObjectId('59b99db4cfa9a34dcd7885b7')
name: "Robert Baratheon"
email: "mark_addy@gameofthron.es"
password: "$2b$12$yGqxLG9LZpXA2xVDhuPnSOZd..."
registro: 2017-09-13T21:05:56.000+00:00
```




Nivel 1

Ejercicio 1

¿Cuál fue el primer usuario/a en registrarse?

2/ Ordenamos los registros por orden cronológico

▼ Stage 2 \$sort

```
1 ▼ /**
2
3  * ordenamos de mas antiguo primero
4  */
5 ▼ {
6    registro: 1
7  }
```

Output after `$sort` stage (Sample of 10 documents)

```
_id: ObjectId('59b99db4cfa9a34dcd7885b7')
name: "Robert Baratheon"
email: "mark_addy@gameofthron.es"
password: "$2b$12$yGqxLG9LZpXA2xVDhuPnSOZd..."
registro: 2017-09-13T21:05:56.000+00:00
```

```
_id: ObjectId('59b99db4cfa9a34dcd7885b6')
name: "Ned Stark"
email: "sean_bean@gameofthron.es"
password: "$2b$12$UREFwsRUoyF0CRqGNK0Lz00HM..."
registro: 2017-09-13T21:05:56.000+00:00
```



Nivel 1

Ejercicio 1

¿Cuál fue el primer usuario/a en registrarse?

3/ Mostramos solo uno para obtener el primero

▼ Stage 3 \$limit



```
1  ▼  /**
2
3  * mostramos el primer usuario/a en regis
4  */
5  1
```



Output after `$limit` stage (Sample of 1 document)

```
_id: ObjectId('59b99db4cfa9a34dcd7885b6')
name: "Ned Stark"
email: "sean_bean@gameofthron.es"
password: "$2b$12$UREFwsRUoyF0CRqGNK0Lz00HM..."
registro: 2017-09-13T21:05:56.000+00:00
```



Nivel 1

Ejercicio 1

¿Cuántas películas de comedia hay en nuestra base de datos?

Filtramos por género comedia

▼ Stage 1 \$match ☐

```
1  ▼ /**
2
3    * películas con genero comedia.
4    */
5  ▼ {
6    genres : "Comedy"
7  }
```

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

```
_id: ObjectId('573a1390f29313caabcd4803')
plot: "Cartoon figures announce, via comic
      strip balloons, that they will mov..."
▶ genres : Array (3)
runtime : 7
▶ cast : Array (1)
num_mflix_comments : 1
poster : "https://m.media-
         amazon.com/images/M/MV5BYzg2NjNhNT..."
```

```
_id: ObjectId('573a1390f29313caabcd50e5')
plot: "The cartoonist, Winsor McCay, brings
      the Dinosaurius back to life in th..."
▶ genres : Array (3)
runtime : 12
▶ cast : Array (3)
num_mflix_comments : 1
poster : "https://m.media-
         amazon.com/images/M/MV5BMTQxNzI4OD..."
```



Nivel 1


Ejercicio 1

¿Cuántas películas de comedia hay en nuestra base de datos?

Hacemos un count

▼ Stage 2 \$count 

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

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

totalComedies : 7024



Nivel 1

Ejercicio 2

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

Filtramos por año y añadimos la variable optativa de genero y lenguaje

▼ Stage 3 \$match ☐

```
1  ▾ /**
2    * query: The query in MQL.
3    */
4  ▾ {
5    year : 1932,
6    $or: [
7      {genres : "Drama"},
8      {languages : "French"}
9    ]
10 }
```



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-
```

```
_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/MV5BNTQ1NTMzMT...
title: "The Blue Light"
```



Nivel 1

Ejercicio 3

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

Filtramos por país, y añadimos la orquilla de premios y fecha requerida

Stage 4 \$match



```
1  ▾ /**
2    * películas estadounidenses
3    |que tengan entre 5 y 9 premios
4    |que fueron producidas entre 2012 y 2014.
5    */
6  ▾ {
7    countries : "USA",
8    "awards.wins" : { $gte: 5, $lte: 9 },
9    "year" : { $gte: 2012, $lte: 2014 }
10 }
```



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

```
_id: ObjectId('573a13acf29313caabd29366')
fullplot : "The manager of the negative
           assets sector of Life magazine,
           Walter Mit..."
imdb: Object
year : 2013
plot : "When his job along with that of his
       co-worker are threatened, Walter t..."
genres : Array (3)
```

```
_id: ObjectId('573a13b5f29313caabd45772')
plot : "After their cave is destroyed, a
       caveman family must trek through an
       u..."
genres : Array (3)
runtime : 98
metacritic : 55
rated : "PG"
cast : Array (4)
```



Nivel 2

Ejercicio 1

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

1/Filtramos los usuarios que usan ese domino

▼ Stage 3 \$match



```
1 ▾ /**
2   * filtramos los que usan ese mail.
3   */
4 ▾ {
5   email: { $regex: "@GAMEOFTHRON.ES", $options: "i" }
6 }
```



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

a9427648b0beeb69579db')

.o'connor@gameofthron.es"
.Id('573a1390f29313caabcd413...
tis sit pariatur quas.
s officia harum ipsum
.T13:49:05.000+00:00

_id: ObjectId('5a9427648b0beeb69579ee')
name : "Theon Greyjoy"
email : "alfie_allen@gameofthron.es"
movie_id : ObjectId('573a1390f29313caabcd437...
text : "Dicta asperiores necessitatibus
corporis. Quidem fugiat eius animi
fug..."
date : 2000-12-06T07:26:29.000+00:00

_i
na
em
mo
te
da



Nivel 2

Ejercicio 1

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

2/Como hay varios usuarios que usan ese mail, agrupamos los comentarios de cada usuario

▼ Stage 4 \$group ☐

```
1  ▼ /**
2
3      * los agrupamos por mail.
4      */
5  ▼ {
6      _id: "$email",
7      totalComentarios: {$sum:1}
8
9  }
```

Output after `$group` stage (Sample of 10 documents)

```
_id: "paul_kaye@gameofthron.es"
totalComentarios : 304
```

```
_id: "eugene_simon@gameofthron.es"
totalComentarios : 285
```




Nivel 2

Ejercicio 1

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

3/Ordenamos para enseñar los que más comentarios tienen primero

▼ Stage 5 \$sort 



```
1 ▼ /**
2
3  * ordenamos primero los más comentados.
4  */
5  ▼ {
6    totalComentarios: -1
7  }
```

Output after [\\$sort](#) stage (Sample of 10 documents)

```
_id: "roger_ashton-griffiths@gameofthron.es"
totalComentarios : 331
```

```
_id: "nathalie_emmanuel@gameofthron.es"
totalComentarios : 327
```



Nivel 2

Ejercicio 2

¿Cuántos cines hay en cada código postal situados dentro del estado Washington D. C. (DC)?

1/Filtramos los cines para obtener los que están en el estado solicitado

▼ Stage 1 \$match ☒

```
1 ▾ /**
2  * filtramos por ciudad y estado.
3  */
4 ▾ {
5   "location.address.city" : "Washington",
6   "location.address.state" : "DC"
7 }
```

Output after [\\$match](#) stage (Sample of 3 documents)

```
_id: ObjectId('59a47286cfa9a3a73e51e785')
theaterId: 1092
  ▶ location: Object
```

```
_id: ObjectId('59a47287cfa9a3a73e51ec33')
theaterId: 801
  ▶ location: Object
```



Nivel 2

Ejercicio 2

¿Cuántos cines hay en cada código postal situados dentro del estado Washington D. C. (DC)?

2/Agrupamos por código postal

▼ Stage 2 \$group 

```
1  ▾ /**  
2    * agrupamos por estado.  
3    */  
4  ▾ {  
5    _id: "location.address.zipcode",  
6    totalcines : {$sum: 1}  
7  
8  }
```



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

```
_id: "location.address.zipcode"  
totalcines : 3
```



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.

Filtramos por director y rango de notas

▼ Stage 1 \$match ☐



```
1  ▼ /**
2
3    * filtramos por director y nota de imdb.
4    */
5  ▼ {
6    directors: "John Landis",
7    "imdb.rating" : { $gte: 7.5, $lte: 8 }
8  }
```

Output after `$match` stage (Sample of 4 documents)

```
_id: ObjectId('573a1397f29313caabce6d94')
fullplot: "Faber College has one frat house
so disreputable it will take
anyone. ..."
imdb: Object
year: 1978
plot: "At a 1962 college, Dean Vernon
Wormer is determined to expel the
entir..."
```

```
_id: ObjectId('573a1397f29313caabce76f7')
plot: "Jake Blues, just out from prison,
puts together his old band to save
t..."
genres: Array (3)
runtime: 133
rated: "R"
cast: Array (4)
num_mflix_comments: 1
```



Nivel 3

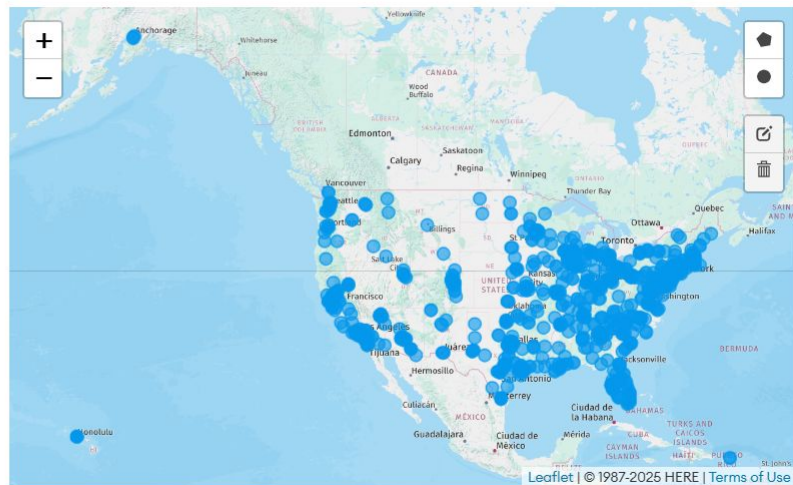
Ejercicio 2

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

Desde la pestaña de schema podemos acceder a la localización de cada teatro.

coordinates

coordinates





Gracias.

