

Sprint 9.01. Consultas con MongoDB

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

- users: Almacena información de usuarios/as, incluyendo nombres, emails y contraseñas cifradas.
- theaters: Contiene datos de cines, como ID, ubicación (dirección y coordenadas geográficas).
- sessions: 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.

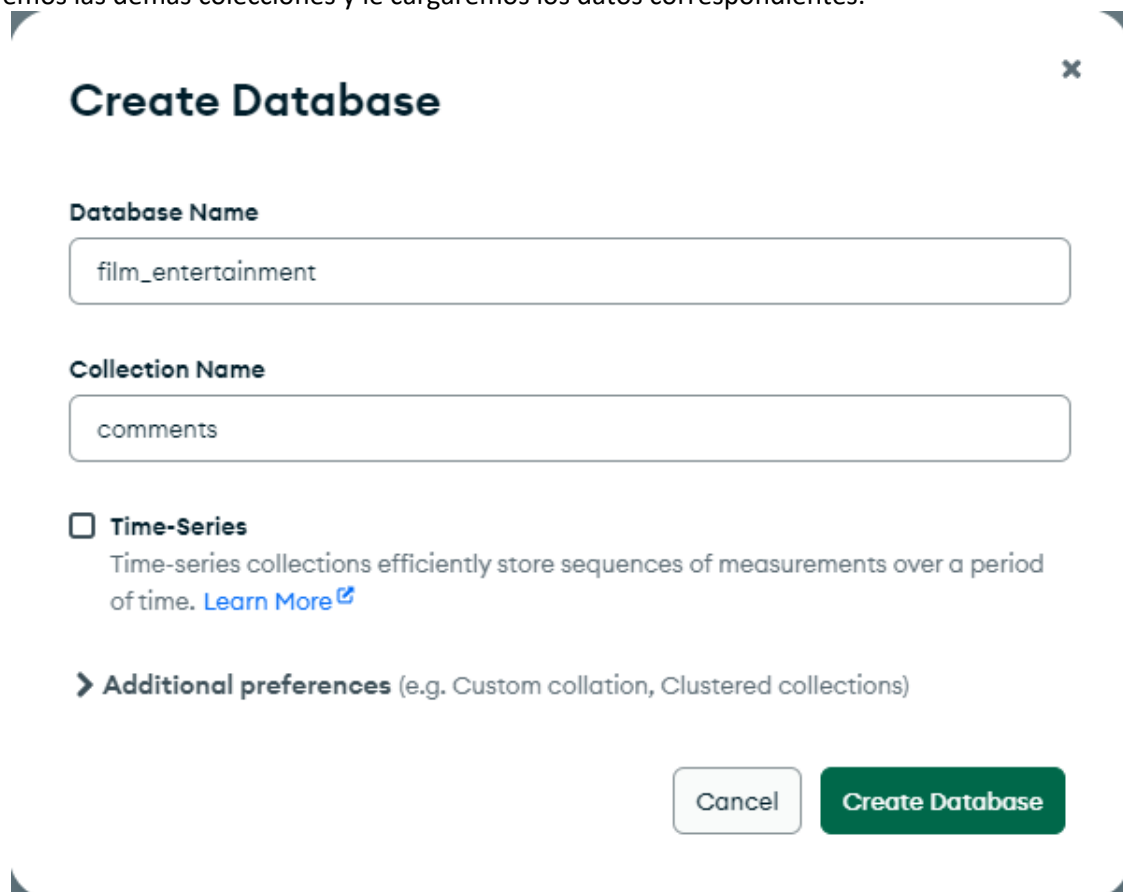
Realizarás algunas consultas que te pide el cliente/a, quien está midiendo si serás capaz o no de hacerte cargo de la parte analítica del proyecto vinculado con su base de datos.

Nivel 1

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

Antes que nada inicializamos la conexión en MongoDB, llamándola “connection”.

Seguido de eso creamos la base de datos “film_entertainment” y nombramos “comment” a la primera colección, nombraremos las demás colecciones y le cargaremos los datos correspondientes.



Create Database

Database Name

film_entertainment

Collection Name

comments

☐ **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

Ejercicio 1

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

`db.comments.find().limit(2)`

```
> _MONGOSH
> use film_entertainment
< switched to db film_entertainment
> db.comments.find().limit(2)
< {
  _id: ObjectId('5a9427648b0beeb69579cc'),
  name: 'Andrea Le',
  email: 'andrea_le@fakegmail.com',
  movie_id: ObjectId('573a1390f29313caabcd418c'),
  text: 'Rem officiis eaque repellendus amet eos doloribus. Porro dolor voluptatum voluptates neque culpa molestias. Voluptate unde
  date: 2012-03-26T23:20:16.000Z
}
{
  _id: ObjectId('5a9427648b0beeb69579cf'),
  name: 'Greg Powell',
  email: 'greg_powell@fakegmail.com',
  movie_id: ObjectId('573a1390f29313caabcd41b1'),
  text: 'Tenetur dolorum molestiae ea. Eligendi praesentium unde quod porro. Commodi nisi sit placeat rerum vero cupiditate neque. I
  date: 1987-02-10T00:29:36.000Z
}
film_entertainment>
```

- ¿Cuántos usuarios tenemos registrados?

`db.users.countDocuments()`

```
> _MONGOSH
> use film_entertainment
< switched to db film_entertainment
> db.users.countDocuments()
< 185
film_entertainment>
```

- ¿Cuántos cines existen en el estado de California?

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

```
> _MONGOSH
> use film_entertainment
< switched to db film_entertainment
> db.theaters.countDocuments({"location.address.state": "CA"})
< 169
film_entertainment>
```

- ¿Cuál fue el primer usuario en registrarse?

```
db.users.find().sort({_id : 1}).limit(1)
```

```
> MONGOSH
> use film_entertainment
< switched to db film_entertainment
> db.users.find().sort({_id : 1}).limit(1)
< {
  _id: ObjectId('59b99db4cfa9a34dcd7885b6'),
  name: 'Ned Stark',
  email: 'sean_bean@gameofthron.es',
  password: '$2b$12$UREFwsRUoyF0CRqGNK0Lz00HM/jLhgUCNNI39RJAqMUQ74cr1J1Vu'
}
film_entertainment>
```

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

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

```
> MONGOSH
> use film_entertainment
< switched to db film_entertainment
> db.movies.countDocuments({genres : "Comedy"})
< 7024
film_entertainment>
```

Ejercicio 2

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

```
db.movies.find({year : 1932, $or : [{genres : "Drama"}, {languages : "French"}]})
```

```
> MONGOSH
> use film_entertainment
< switched to db film_entertainment
> db.movies.find({year : 1932, $or : [{genres : "Drama"}, {languages : "French"}]})
< {
  _id: ObjectId('573a1391f29313caabcd9458'),
  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',
  runtime: 55,
  rated: 'UNRATED',
  cast: [
    'Enrique Rivero',
    'Elizabeth Lee Miller',
    'Pauline Carton',
    'Odette Talazac'
  ],
  num_mflix_comments: 1,
  poster: 'https://m.media-amazon.com/images/M/MV5BYWY3ODk5ZWVjY000NjA4LTk4ZWYtMzBhZDZlMjY0YTUxXkE5YXkFqcGdeQXVyNzI4MDMyMTU@._V1',
  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: [

```

Ejercicio 3

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

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

```
>_MONGOSH
> use film_entertainment
< switched to db film_entertainment
> db.movies.find({countries : "USA",
"awards.wins" : {"$gte" : 5, "$lte" : 9},
year : {"$gte" : 2012, "$lte" : 2014}})
< {
  _id: ObjectId('573a13acf29313caabd29366'),
  fullplot: "The manager of the negative assets sector of Life magazine, Walter Mitty, has been working for sixteen years for the m:
  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 :
  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'
  ]
}
```

Nivel 2

Ejercicio 1

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

```
db.comments.countDocuments({email : { $regex : "gameofthron.es$"}})
```

```
>_MONGOSH
> use film_entertainment
< switched to db film_entertainment
> db.comments.countDocuments({email : { $regex : "gameofthron.es$"}})
< 22841
film_entertainment>
```

Ejercicio 2

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

```
db.theaters.aggregate([
  {$match : {"location.address.state" : "DC"}},
  {$group : {_id : "$location.address.zipcode", totalTheaters : {$sum : 1}}},
  {$project : {_id : 0, zipcode : "$_id", totalTheaters : 1}}])
```

```
> use film_entertainment
< switched to db film_entertainment
> db.theaters.aggregate([
  {$match : {"location.address.state" : "DC"}},
  {$group : {_id : "$location.address.zipcode", totalTheaters : {$sum : 1}}},
  {$project : {_id : 0, zipcode : "$_id", totalTheaters : 1}}])
< {
  totalTheaters: 1,
  zipcode: '20010'
}
{
  totalTheaters: 1,
  zipcode: '20016'
}
{
  totalTheaters: 1,
  zipcode: '20002'
}
```

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.

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

```
> MOWGOSH
> use film_entertainment
< switched to db film_entertainment
> db.movies.find({directors : "John Landis", "imdb.rating" : {"$gte" : 7.5, "$lte" : 8}})
< {
  _id: ObjectId('573a1397f29313caabce6d94'),
  fullplot: "Faber College has one frat house so disreputable it will take anyone. It has a second one full of white, anglo-saxon, i
  imdb: {
    rating: 7.6,
    votes: 84834,
    id: 77975
  },
  year: 1978,
  plot: 'At a 1962 college, Dean Vernon Wormer is determined to expel the entire Delta Tau Chi Fraternity, but those trouble-makers
  genres: [
    'Comedy'
  ],
  rated: 'R',
  metacritic: 82,
  title: 'Animal House',
  lastupdated: '2015-09-13 00:02:47.803000000',
  languages: [
    'English',
    'Italian'
  ],
  writers: [
```