

MongoDB

Integrantes:

Yuliana Casanova Lopez

Carlos David Aquino Reyes

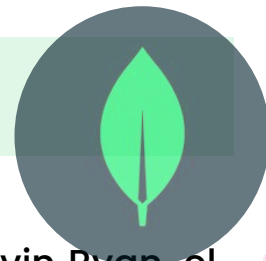
Darina Jocelyn Espinosa Tlatelpa

¿Qué es MongoDB?

Es un sistema de gestión de base de datos no relacional y de código abierto, se trata de un modelo orientado a documentos. Los documentos o colecciones de documentos de MongoDB son las unidades básicas de datos. Estos documentos, con formato JSON (Java Script Object Notation) binario, pueden almacenar varios tipos de datos y distribuirse en varios sistemas.



Historia



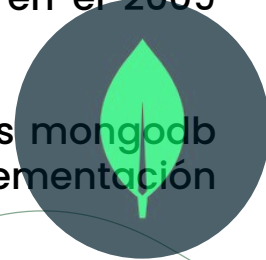
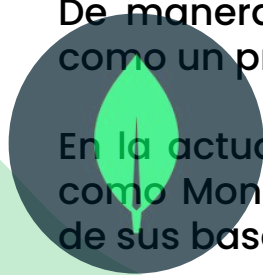
MongoDB fue fundada en 2007 por Dwight Merriman, Eliot Horowitz y Kevin Ryan, el equipo detrás de DoubleClick.

buscaban solucionar las carencias de las base de datos que existían en ese entonces, como por ejemplo, los problemas de escalabilidad y agilidad.

A diferencia de una base de datos relacional SQL tradicional, MongoDB no se basa en tablas y columnas. Los datos se almacenan como colecciones y documentos.

De manera que, este programa fue lanzado al mercado tecnológico en el 2009 como un producto independiente bajo la licencia de código abierto

En la actualidad, esta aplicación cuenta con recursos y características **mongodb** como MongoDB Atlas, que se encarga de simplificar la gestión e implementación de sus bases de datos



FUNDADORES



Dwight Merriman

Es un ejecutivo de Internet, piloto de carreras y empresario estadounidense en



Silicon Alley



Eliot Horowitz

Es fundador y ex director de tecnología de MongoDB



Kevin P. Ryan

Es un inversor y emprendedor estadounidense

Características

Escalabilidad horizontal

Añade más servidores a medida que aumenta la carga de trabajo.

Alta disponibilidad

Ofrece mecanismos para la replicación de datos, lo que garantiza que los datos estén disponibles incluso en caso de fallos en los servidores.

Facilidad de indexación

Permite crear, administrar y rehacer índices sin grandes dificultades

Características

API de consulta

La API de consulta de MongoDB admite leer y escribir operaciones (CRUD) así como: agregación de datos, búsqueda de texto y consultas geoespaciales

Lenguajes de programación

Permite usar diferentes lenguajes de programación, ya que cuenta con gran variedad de drivers oficiales para ello

Compatibilidad

Compatibilidad con varios motores de almacenamiento como:

- WiredTiger
- Memoria

Finalidad



Brindar una plataforma de base de datos flexible y escalable que se adapte a las necesidades de las aplicaciones modernas, además del almacenamiento y recuperación eficiente de datos, especialmente aquellas que manejan grandes volúmenes de datos y requieren flexibilidad en la estructura de datos.

Aplicaciones / Herramientas

1 Empresas de telecomunicaciones

Orange y Telefónica

2 Cisco

3 Bosh, Forsquare

5 eBay

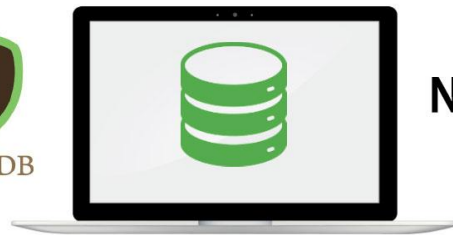
6 Expedia LinkedIn



¿NoSQL?

NoSQL, también conocido como "no solo SQL", "no SQL", aún puede almacenar datos que se encuentran dentro de los sistemas de gestión de bases de datos relacionales (RDBMS), se pueden consultar utilizando API de lenguaje natural, lenguajes de consulta estructurados declarativos y lenguajes de consulta mediante ejemplo,

Is MongoDB NoSQL



NoSQL

Interfaz

MongoDB ofrece una variedad de interfaces que van desde la línea de comandos hasta interfaces gráficas y librerías de programación, lo que permite a los usuarios elegir la forma más adecuada para interactuar con las base datos según sus preferencias y necesidades

Controladores de lenguajes de programación (drivers)

Permiten a las aplicaciones interactuar con la BD de manera programática

Mongo Shell

Interfaz de línea de comandos

MongoDB Compass

Interfaz gráfica de usuario (GUI)

Frameworks y librerías de desarrollo

Facilitan la integración de aplicaciones web y móviles

Funciones Generales

La función principal de MongoDB radica en su capacidad de guardar los datos como documentos y los almacena en BSON sin necesidad de un esquema. Además, sus funciones se caracterizan por la escalabilidad y flexibilidad.

Por otra parte, sus funciones constan de herramientas de automatización, monitorización y respaldo, lo que permite la escalabilidad horizontal con seguridad integral.



Funciones propias

Inserción de documentos

Permite insertar nuevos documentos en una colección(objeto JSON)

Consulta/Búsqueda

Permite buscar documentos en una colección utilizando criterios de filtrado y ordenamiento

Actualización de documentos

Utilizar operaciones como: updateOne / updateMany / findOneAndUpdate

Eliminación de documentos

Utilizar operaciones como: deleteOne / deleteMany

Índices

Ayudan acelerar las consultas y mejorar el rendimiento (geoespaciales/texto completo)

Replicación

Permite replicar los datos en varios nodos, brindando alta disponibilidad (asíncrona/síncrona)

Funciones propias

Control de acceso y seguridad

Proporciona características de seguridad, incluyendo la autenticación de usuarios, a nivel de colección y BD.

Respaldos y restauración

Son herramientas para realizar copias de seguridad de tus datos y restaurarlos en caso de pérdida

MongoDB Atlas

Simplifica la implementación y administración de clústeres de MongoDB en la nube

Fragmentación de Datos(sharding)

Divide los datos en fragmentos distribuidos en clústeres de servidores

Búsqueda de texto completo

Se pueden hacer en campos de texto utilizando índices de y texto completo

Geolocalización

Ofrece soporte nativo para datos geoespaciales, para realizar consultas espaciales y almacenar información gráfica

Plataformas

Sistemas operativos

- ✓ Linux
- ✓ Windows
- ✓ macOS

Nubes públicas

- ✓ Amazon web services (AWS)
- ✓ Microsoft azure
- ✓ Google cloud Platform (GCP)

Lenguajes de programación

- ✓ Python
- ✓ Java
- ✓ C#
- ✓ PHP
- ✓ Node.js
- ✓ Ruby

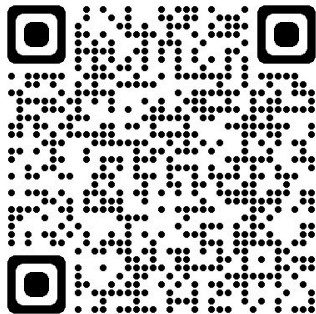
Plataformas como servicio

- ✓ MongoDB Atlas
- ✓ Otros servicios PaaS

Contenedores

- ✓ Docker
- ✓ Kubernetes

Caso practico



Simulador

instruqt

>_ terminal

Current Mongosh Log ID: 64e69f9d41fd13a75ea653e3

Connecting to: mongodb+srv:///<credentials>@instruqtest.3xfvk.mongodb.net/example?appName=mongosh+1.10.1

Using MongoDB: 6.0.9

Using Mongosh: 1.10.1

For mongosh info see: <https://docs.mongodb.com/mongodb-shell/>

To help improve our products, anonymous usage data is collected and sent to MongoDB periodically (<https://www.mongodb.com/legal/privacy-policy>). You can opt-out by running the `disableTelemetry()` command.

Atlas atlas-d3opcw-shard-0 [primary] example>

Instructions

In this lab, you will insert, find, and delete data in MongoDB using the `mongosh` in the `terminal` tab. You are now connected to an Atlas cluster and the `example` database. Click the 'Insert Documents' section title to get the instructions for the first stage of the lab.

Insert Documents

Find Documents

Delete a Document

Check

Terminal

[mongodb] [(0*\$mongosh mongodb+srv:///<credentials>@instruqtest.3xfvk.mongodb.net/example)] [24/08 0:09]

Using MongoDB: 6.0.9
Using Mongosh: 1.10.1

For mongosh info see: <https://docs.mongodb.com/mongodb-shell/>

To help improve our products, anonymous usage data is collected and sent to MongoDB periodically (<https://www.mongodb.com/legal/privacy-policy>). You can opt-out by running the `disableTelemetry()` command.

Atlas atlas-d3opcw-shard-0 [primary] example> db.movies_643fed3e_4212_11ee_9019_6b6a78236560.insertMany([

```
... {
...   title: "Titanic",
...   year: 1997,
...   genres: ["Drama", "Romance"],
...   rated: "PG-13",
...   languages: [
...     "English",
...     "French",
...     "German",
...     "Swedish",
...     "Italian",
...     "Russian",
...   ],
...   released: ISODate("1997-12-19T00:00:00.000Z"),
...   awards: {
...     wins: 127,
...     nominations: 63,
...     text: "Won 11 Oscars. Another 116 wins & 63 nominations.",
...   },
...   cast: [
...     "Leonardo DiCaprio",
...     "Kate Winslet",
...     "Billy Zane",
...     "Kathy Bates",
...   ],
... },
... directors: ["James Cameron"],
... },
... {
...   title: "The Dark Knight",
...   year: 2008,
...   genres: ["Action", "Crime", "Drama"],
```

```
acknowledged: true,
insertedIds: {
  '0': ObjectId("64e6a35dd30538fcf1c2c624"),
  '1': ObjectId("64e6a35dd30538fcf1c2c625"),
  '2': ObjectId("64e6a35dd30538fcf1c2c626"),
  '3': ObjectId("64e6a35dd30538fcf1c2c627")
}
```

Insertar

MongoDB stores documents in collections. Collections are analogous to tables in relational databases. If a collection does not exist, MongoDB creates the collection when you first store data for that collection.

The following command uses the

`db.collection.insertMany()` method to insert new documents into the `movies_643fed3e_4212_11ee_9019` collection in the `example` database.

The collection name used in this lab is intentionally unique; however, your MongoDB collection names can be shorter. See [collection naming restrictions](#) for more details.

Copy the following command

Check

```
'1': ObjectId("64e6a35dd30538fcf1c2c625"),
'2': ObjectId("64e6a35dd30538fcf1c2c626"),
'3': ObjectId("64e6a35dd30538fcf1c2c627")
}

tlas atlas-nmat34-shard-0 [primary] example> db.movies_961b1fe8_4214_11ee_afe4_0edb1d199dd8.find( { } )
```

```
{
  _id: ObjectId("64e6a35dd30538fcf1c2c624"),
  title: 'Titanic',
  year: 1997,
  genres: [ 'Drama', 'Romance' ],
  rated: 'PG-13',
  languages: [ 'English', 'French', 'German', 'Swedish', 'Italian', 'Russian' ],
  released: ISODate("1997-12-19T00:00:00.000Z"),
  awards: {
    wins: 127,
    nominations: 63,
    text: 'Won 11 Oscars. Another 116 wins & 63 nominations.'
  },
  cast: [
    'Leonardo DiCaprio',
    'Kate Winslet',
    'Billy Zane',
    'Kathy Bates'
  ],
  directors: [ 'James Cameron' ]
},
{
  _id: ObjectId("64e6a35dd30538fcf1c2c625"),
  title: 'The Dark Knight',
  year: 2008,
  genres: [ 'Action', 'Crime', 'Drama' ],
  rated: 'PG-13',
  languages: [ 'English', 'Mandarin' ],
  released: ISODate("2008-07-18T00:00:00.000Z"),
  awards: {
    wins: 144,
    nominations: 106,
    text: 'Won 2 Oscars. Another 142 wins & 106 nominations.'
  },
}
```

Buscar

In the **mongosh** terminal, you are already connected to the **example** database.

Copy and paste the following query to return all documents in the

movies_961b1fe8_4214_11ee_afe4_0edb1d199dd8 collection:

```
db.movies_961b1fe8_4214_1
```

Run the following query to find movies that were directed by Christopher Nolan:

```
db.movies_961b1fe8_4214_1
```

Run the following query to return movies that were released before the year 2000:

```
db.movies_961b1fe8_4214_1
```

Check

```
'Paul Henreid',
'Claude Rains'
],
languages: [ 'English', 'French', 'German', 'Italian' ],
released: ISODate("1943-01-23T00:00:00.000Z"),
directors: [ 'Michael Curtiz' ],
awards: {
  wins: 9,
  nominations: 6,
  text: 'Won 3 Oscars. Another 6 wins & 6 nominations.'
},
lastupdated: '2015-09-04 00:22:54.600000000',
year: 1942
}
```

```
tlas atlas-nmat34-shard-0 [primary] example> db.movies_961b1fe8_4214_11ee_afe4_0edb1d199dd8.find( { "directors": "Christopher Nolan" } )
```

```
{
  _id: ObjectId("64e6a35dd30538fcf1c2c625"),
  title: 'The Dark Knight',
  year: 2008,
  genres: [ 'Action', 'Crime', 'Drama' ],
  rated: 'PG-13',
  languages: [ 'English', 'Mandarin' ],
  released: ISODate("2008-07-18T00:00:00.000Z"),
  awards: {
    wins: 144,
    nominations: 106,
    text: 'Won 2 Oscars. Another 142 wins & 106 nominations.'
  },
  cast: [
    'Christian Bale',
    'Heath Ledger',
    'Aaron Eckhart',
    'Michael Caine'
  ],
  directors: [ 'Christopher Nolan' ]
}
```

```
tlas atlas-nmat34-shard-0 [primary] example>
```

```
mongodb ][ (0*$mongosh mongodb+srv://<credentials>@instruqt-test-shared-1.3xfvk.mongodb.net/example)
```

```
][ 24/08 0:35 ]
```

```
:_0edb1d199dd8.find( { } )
```

Run the following query to find movies that were directed by Christopher Nolan:

```
db.movies_961b1fe8_4214_1
```

Run the following query to return movies that were released before the year 2000:

```
db.movies_961b1fe8_4214_1
  released: { $lt: ISODat
})
```

Click the 'Delete a Document' section title to get the instructions for the next stage of the lab.

Check

Búsqueda específica



```

directors: [ 'Michael Curtiz' ],
awards: {
  wins: 9,
  nominations: 6,
  text: 'Won 3 Oscars. Another 6 wins & 6 nominations.'
},
lastupdated: '2015-09-04 00:22:54.600000000',
year: 1942
}
]
Atlas atlas-nmat34-shard-0 [primary] example> db.movies_961b1fe8_4214_11ee_afe4_0edb1d199dd8.deleteOne({'title' : 'Titanic'})
{ acknowledged: true, deletedCount: 1 }
Atlas atlas-nmat34-shard-0 [primary] example> db.movies_961b1fe8_4214_11ee_afe4_0edb1d199dd8.find(
...   released: { $lt: ISODate("2000-01-01") },
... })
[
  {
    _id: ObjectId("64e6a35dd30538fcf1c2c627"),
    title: 'Casablanca',
    genres: [ 'Drama', 'Romance', 'War' ],
    rated: 'PG',
    cast: [
      'Humphrey Bogart',
      'Ingrid Bergman',
      'Paul Henreid',
      'Claude Rains'
    ],
    languages: [ 'English', 'French', 'German', 'Italian' ],
    released: ISODate("1943-01-23T00:00:00.000Z"),
    directors: [ 'Michael Curtiz' ],
    awards: {
      wins: 9,
      nominations: 6,
      text: 'Won 3 Oscars. Another 6 wins & 6 nominations.'
    },
    lastupdated: '2015-09-04 00:22:54.600000000',
    year: 1942
  }
]
Atlas atlas-nmat34-shard-0 [primary] example> [
[ mongodb ][ 0*$mongosh mongodbsrv://<credentials>@instruqt-test-shared-1.3xfvk.mongodb.net/example ] [ 24/08 0:39 ]

```

Eliminar

your request:

```
{ acknowledged: true, del
```

To verify that the **Titanic** document was deleted, rerun the query for movies released before 2000:

```

db.movies_961b1fe8_4214_1
  released: { $lt: ISODate
})

```

You should now see one document in the results.

Click the **Check** button on your screen after you have deleted the document to continue the lab.

Now that you're familiar with some basic CRUD commands,

Check

Video: **Instalación**



Facultad
Autodidacta



mongoDB

1

*Recursos
bases de datos*

*Instalar y configurar
MongoDB*



Gracias por
su atención