



**Argentina
programa
4.0**



Ministerio de Economía
Argentina

Secretaría de
Economía del Conocimiento

***primero
la gente***

Clase 10: Mongo DB

Temario

- A. El concepto de base de datos
- B. Bases de datos noSQL
 - a. MongoDB
 - i. El formato BSON versus JSON
- C. MongoDB
 - a. crear un clúster
 - b. instalar Compass
 - c. Visualizar bb.dd. y Documentos
 - i. Documentos
 - ii. Colecciones
 - iii. Índices
- D. Instalar MongoDB local versus clúster
- E. Importar estructuras JSON

Concepto de base de datos

El concepto de base de datos

Si bien, venimos hablando de estructuras de datos JSON, XML y derivados, hasta el momento nunca tuvimos un contacto directo con una base de datos que “*provea*” la información original.

A partir de esta unidad, nos adentraremos en los fundamentos de bb.dd. para conocer sus características, opciones, para finalmente comenzar a utilizar MongoDB.



El concepto de base de datos

Una base de datos es un conjunto organizado de información que se puede acceder, administrar, y actualizar de manera eficiente.

Comúnmente son utilizadas para almacenar y gestionar grandes cantidades de información en empresas y organizaciones.



API Restful y JSON

Y, si buscamos en la historia informática, cuándo comenzó a mencionarse el término base de datos, encontramos dos puntos distantes en esta línea de tiempo:

- **1963**
- **1884**



Historia

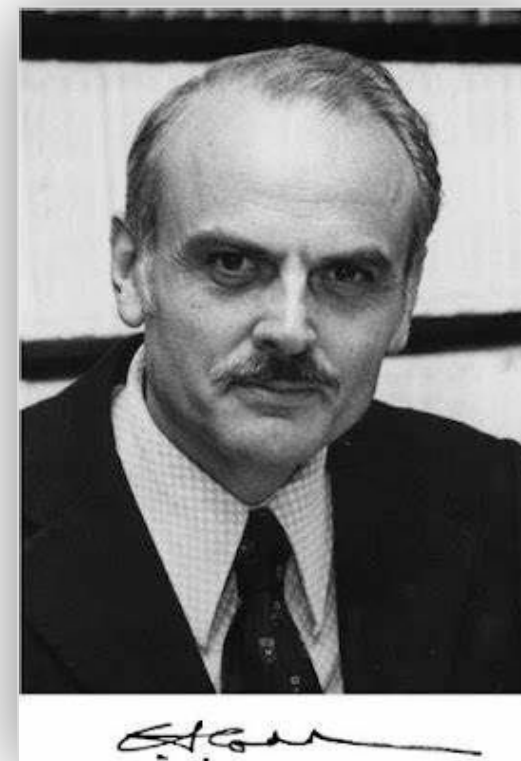
API Restful y JSON

1963

El año más próximo a nuestra era, es 1963. Allí, en un incipiente **Silicon Valley**, se escucha por primera vez el término “*Base de datos*” en un Congreso realizado en la ciudad de California. Posterior a este simposio, la frase “*base de datos*”, empieza a aparecer de manera recurrente en diferentes libros de informática.

En 1967, [Edgard Codd](#), presenta el “**Proposal Paper**”, acuñando el término de base de datos relacionales. IBM lo rechazó, pero sí le interesó a Oracle.

El resto, ¡es historia!



API Restful y JSON

Codd era empleado en IBM, por lo cual debía presentar sus propuestas técnicas en primer lugar a su empleador. Si este las rechazaba, él quedaba habilitado para hacer negocios con las mismas por fuera de su empleador.

IBM vendía Mainframes comerciales y estos equipos ya contaban con su modelo de almacenamiento de datos. Las bases de datos relacionales explotaron en popularidad a mediados de los 70 's de la mano de Oracle, gracias al Paper de *Edgar Codd*.

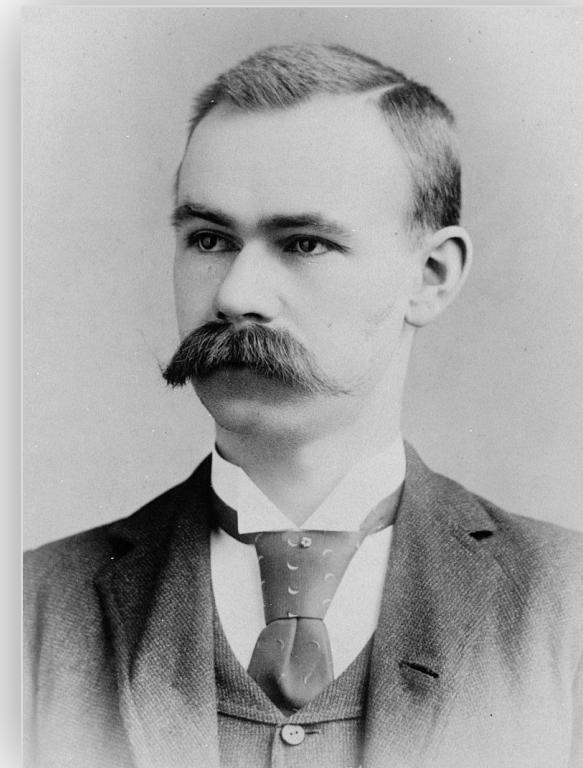


API Restful y JSON

1884

Y si bien, en la historia de la humanidad, existen desde hace siglos las bibliotecas con registros de las diferentes civilizaciones, registro de cosechas, censos poblacionales, como también de las olas inmigrantes, toda esa información registrada en papel no tenía sentido por la ausencia de mecanismos de procesamiento automático.

En 1884, [Herman Hollerith](#), diseñó la máquina automática de tarjetas perforadas, ante la necesidad de registrar y procesar rápidamente toda la información de los barcos que llegaban a NYC con inmigrantes europeos.



API Restful y JSON

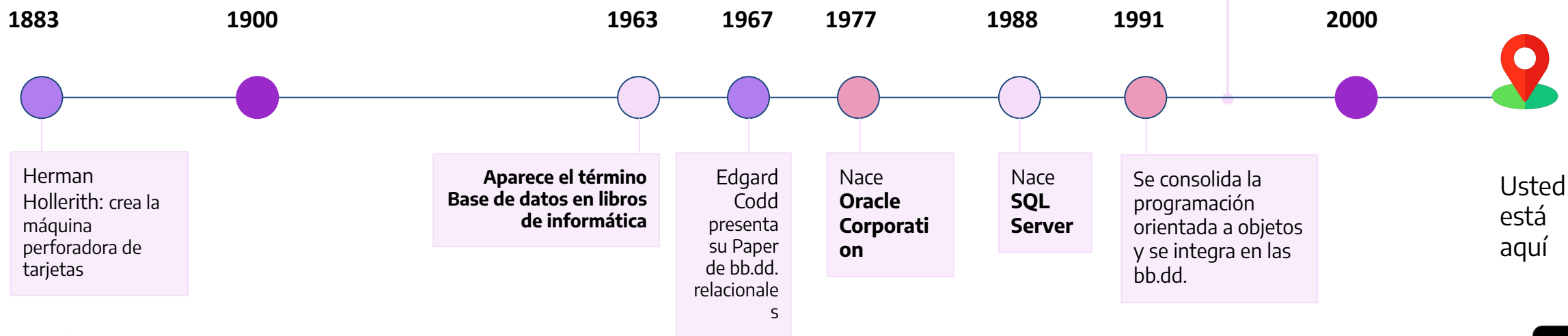
Su modelo de máquina se incorporó a los mainframes y fue utilizado hasta los años 50 's (1950), siendo reemplazado por las cintas magnéticas las cuales contaban con un poder de almacenamiento de datos más efectivo.

La invención a fines del siglo XIX, convirtió a **Herman Hollerith**, en el Primer Ingeniero Estadístico de la historia.



API Restful y JSON

IBM llegó, finalmente, en 1993 al mercado de bb.dd. con su software de bases de datos relacional, llamado DB2.



Tipos de bb.dd

Tipos de bb.dd.

Dentro de los diferentes tipos de bases de datos que existen en el mercado, podemos destacar a:

- bb.dd. relacionales
- bb.dd. de objetos
- bb.dd. jerárquicas
- bb.dd. de red
- bb.dd. NoSQL



Tipos de bb.dd.

Cada uno de estos tipos de bases de datos, tiene sus propias características y ventajas por sobre el resto de las opciones.

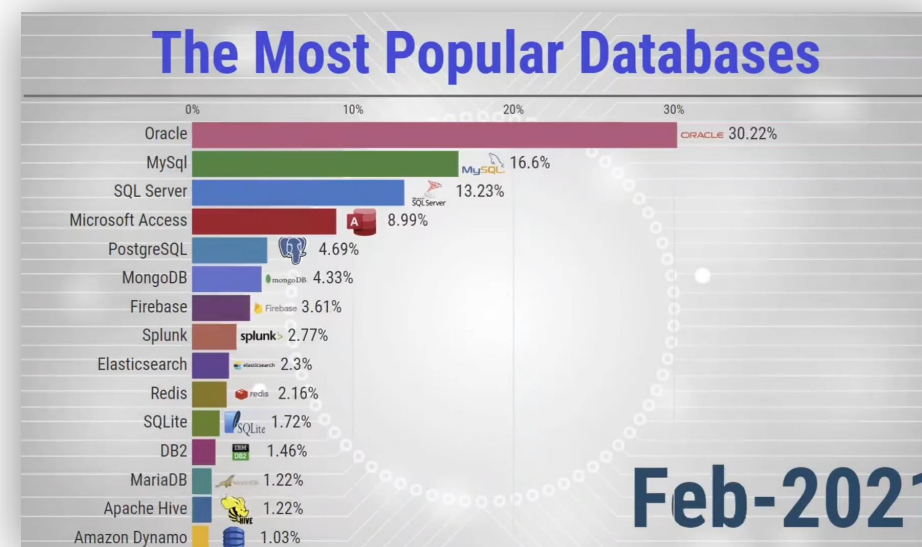
Lo que sí debemos tener en cuenta, es la importancia de elegir correctamente el tipo de bb.dd. adecuada, para el tipo de información que se debe almacenar.



Tipos de bb.dd.

Si te interesa conocer cómo evolucionó el mercado de bases de datos en estas últimas dos décadas, te invitamos a mirar el siguiente video.

Encontrarás en este que los tres primeros puestos mantienen con firmeza su lugar, variando muy ligeramente su porcentaje de mercado.



BB.DD. no SQL

Las bases de datos NoSQL son un tipo de sistema de gestión de bases de datos (DBMS) que se utilizan para almacenar y administrar grandes cantidades de datos no estructurados o semi estructurados.

A diferencia de las bb.dd. relacionales, las NoSQL no utilizan un esquema fijo ni requieren que los datos se ajusten a un modelo predefinido, además permiten una mayor flexibilidad y escalabilidad horizontal, lo que las hace ideales para aplicaciones web y móviles de alta escala.



BB.DD. no SQL

Veamos, a continuación, los diferentes tipos de bases de datos NoSQL.

Tipo de bb.dd.	Descripción
Pares clave-valor	Almacenan datos como pares clave-valor, donde cada clave es única y se utiliza para recuperar el valor asociado.
de documentos	Almacenan datos como documentos JSON, XML u otros formatos similares. Cada documento puede tener una estructura diferente y puede estar relacionado con otros documentos.
de columnas	Almacenan datos en columnas en lugar de filas. Las columnas se agrupan en familias de columnas y se utilizan para representar datos relacionados.
de grafos	Almacenan datos en nodos y relaciones, lo que permite modelar datos altamente interconectados y estructuras complejas.

BB.DD. no SQL

Las bb.dd. NoSQL se expandieron rápidamente en el mercado.

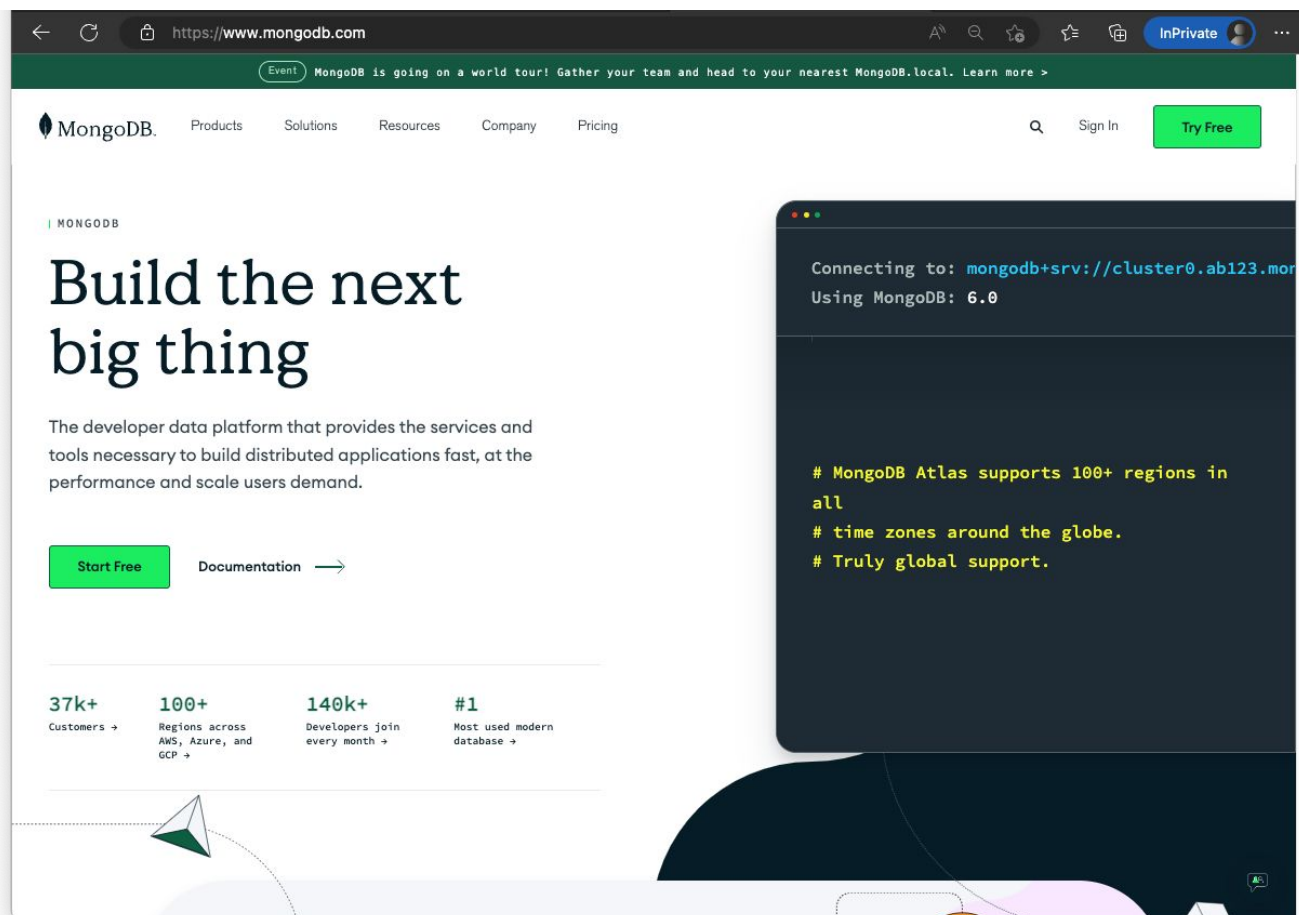
Aparecieron variantes comerciales, Open Source, Cloud y Freemium, entre otras. Dentro de los modelos comerciales, encontramos a:

- MongoDB
- Amazon DynamoDB
- Apache Cassandra
- DataStax Enterprise
- Couchbase

Por popularidad, antigüedad, y por ser parte del stack [MERN - MEAN - MEVN](#), **MongoDB es la elegida para ser desarrollada**, a continuación.

Mongo DB

MongoDB



Los Stack de programación se centran últimamente en una tecnología o un conjunto de tecnologías específicas.

MongoDB es una de las tecnologías más solicitadas en Stacks orientados al lenguaje JavaScript.

BB.DD. no SQL

Su estructura se basa en una bb.dd. orientada a documentos, escalable, y muy fácil de utilizar.

Como ya venimos trabajando con objetos JavaScript, Array de elementos y de objetos, aprender MongoDB será un camino fácil, dada su inspiración y desarrollo en estructuras de arrays multidimensionales.



BB.DD. no SQL

Dentro de sus características notables, podemos destacar:

- Manejo de documentos BSON
- Escalabilidad horizontal
- Flexibilidad
- Rendimiento

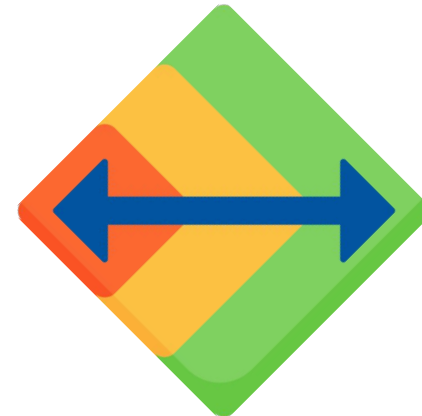


BB.DD. no SQL

Escalabilidad horizontal

MongoDB puede manejar grandes cantidades de datos y escalar horizontalmente mediante la adición de servidores adicionales.

En el mundo actual, donde la tecnología Cloud reina por sobre los servidores analógicos corporativos, MongoDB es una base de datos que puede expandirse rápidamente gracias a Clústers.



MongoDB

Flexibilidad

A diferencia de otras bb.dd. como las de tipo SQL, MongoDB es una base de datos sin esquema, lo que significa que no requiere una estructura de datos rígida. Esto hace que sea fácil agregar nuevos campos y actualizar la estructura de la base de datos.

```
Estructura de Datos

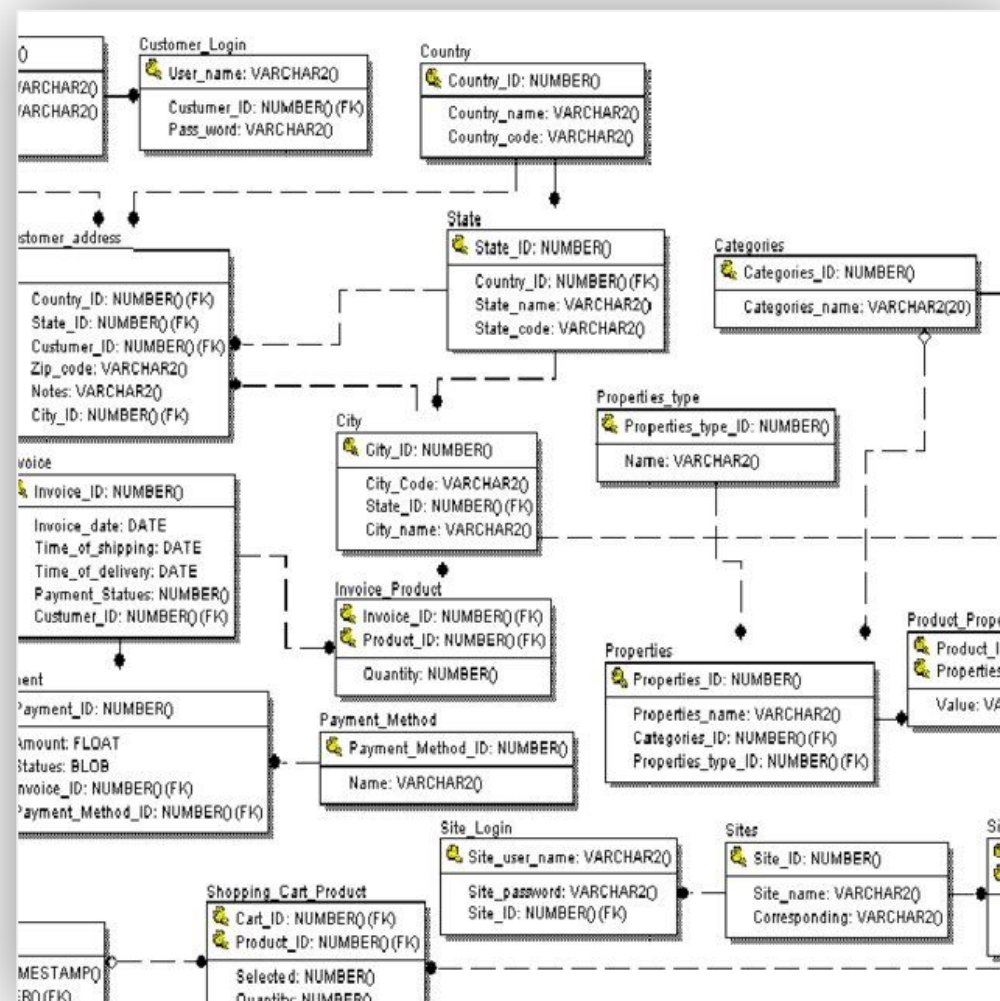
{
  id: 1234,
  nombre: 'Tablet DROID 10.1',
  importe: 42500,
  categoría: 'Tablet'
},
{
  id: 2345,
  nombre: 'Tablet PAD Air',
  importe: 192500,
  categoría: 'Tablet'
},
{
  id: 3456,
  nombre: 'Watch Xiaomi 1.8',
  importe: 22500
}
```

MongoDB

Flexibilidad

Las bb.dd. que requieren un Schema nos obligan a definir una estructura de datos fija por cada Entidad por la cual necesitamos almacenar información.

MongoDB prescinde de esto por lo cual podremos crecer o decrecer en datos de acuerdo a nuestra necesidad.



MongoDB

Rendimiento

MongoDB está diseñado para manejar grandes cantidades de datos y consultas complejas de manera eficiente.

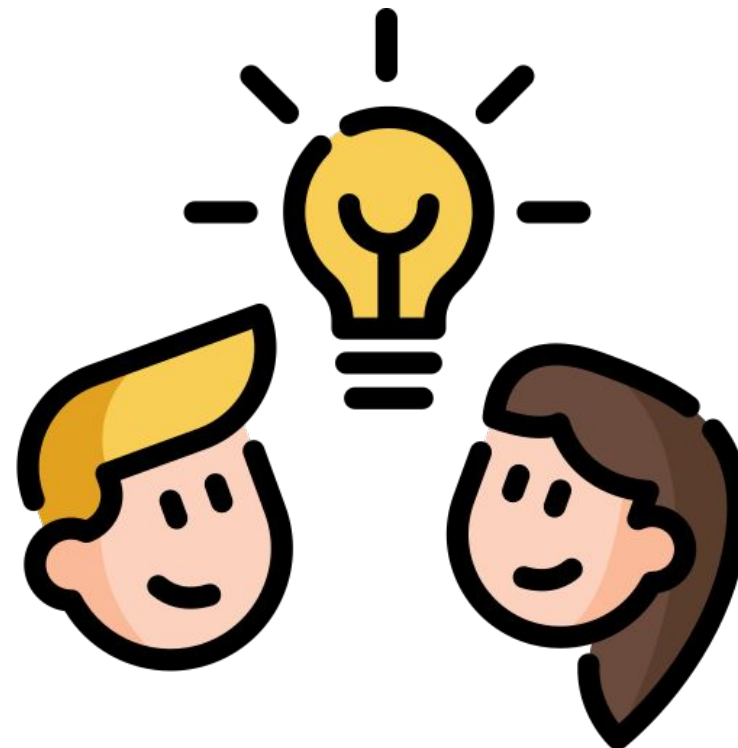
```
{
  "address": {
    "building": "1007",
    "coord": [ -73.856077, 40.848447 ],
    "street": "Morris Park Ave",
    "zipcode": "10462"
  },
  "borough": "Bronx",
  "cuisine": "Bakery",
  "grades": [
    { "date": { "$date": 1393804800000 }, "grade": "A", "score": 2 },
    { "date": { "$date": 1378857600000 }, "grade": "A", "score": 6 },
    { "date": { "$date": 1358985600000 }, "grade": "A", "score": 10 },
    { "date": { "$date": 1322006400000 }, "grade": "A", "score": 9 },
    { "date": { "$date": 1299715200000 }, "grade": "B", "score": 14 }
  ],
  "name": "Morris Park Bake Shop",
  "restaurant_id": "30075445"
}
```

Instalación de MongoDB

Instalación de MongoDB

Llegó el momento de instalar MongoDB previo a comenzar a aprender cómo utilizarla. Pero, en nuestro caso, obviaremos la descarga del motor de MongoDB en nuestra computadora, e iremos directamente a crear una base de datos MongoDB en un Clúster en la nube.

Esto no sólo nos ahorra espacio en disco, sino que también nos acerca a comenzar a experimentar con tecnologías Cloud.



Instalación de MongoDB

Crear nuestra bb.dd. MongoDB

Antes de seguir adquiriendo conocimientos técnicos de MongoDB, crearemos nuestra base de datos en un clúster Cloud.

Esto nos permitirá experimentar con MongoDB y, a su vez, entender cómo trabajar con nubes de datos.

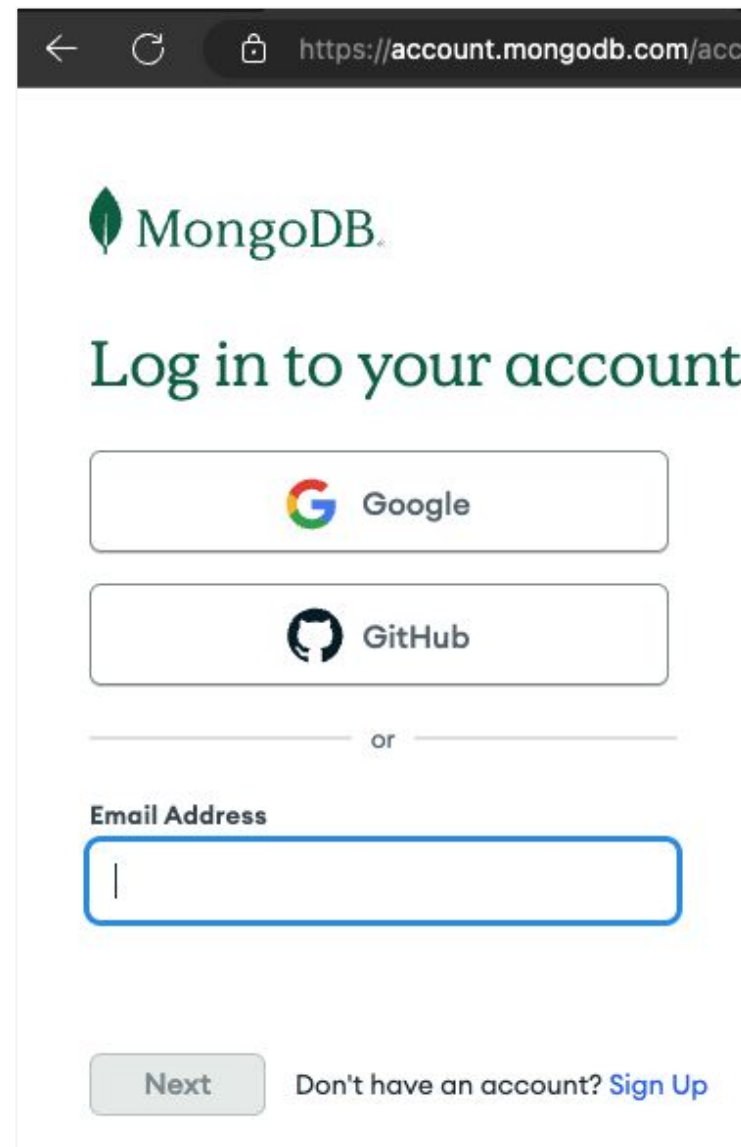


Instalación de MongoDB

Sign In

En la web oficial (www.mongodb.com) elegimos la opción **Sign In** para registrarnos.

Podremos optar por cargar los datos manualmente, a través de la función Sign Up, o utilizar una cuenta de correo **Google** o **Github**. Si puedes utilizar esta última opción, el proceso de registro será mucho más rápido.



The screenshot shows the MongoDB account login page in a web browser. The address bar displays 'https://account.mongodb.com/acc'. The page features the MongoDB logo at the top, followed by the heading 'Log in to your account'. Below this, there are two large buttons for social login: 'Google' (with the Google logo) and 'GitHub' (with the GitHub logo). A horizontal line with the word 'or' in the center separates these from the manual login section. The manual login section includes a label 'Email Address' above a text input field. At the bottom, there is a 'Next' button and a link that says 'Don't have an account? Sign Up'.

Instalación de MongoDB

What is your goal today?
Your answer will help us guide you to successfully getting started with MongoDB Atlas.

☐ Learn MongoDB
☐ Migrate an existing application
☐ Explore what I can build
☒ Build a new application

What type of application are you building?

Other

Tell us what you're building

Sample database on the cloud

What is your preferred language?
We'll use this to customize code samples and content we share with you. You can always change this later.

JS JavaScript

Finish

Información de contacto




Completamos los datos requeridos para que Atlas, el asistente cloud de MongoDB comience a diseñar nuestro perfil de usuario.

JS será el lenguaje elegido para programar sobre MongoDB, y el tipo de clúster a seleccionar, será la opción **FREE**. 🖱️


M0			FREE
For learning and exploring MongoDB in a cloud environment.			
STORAGE	RAM	vCPU	
512 MB	Shared	Shared	

Instalación de MongoDB

Provider



Region ★ Recommended region ⓘ

 Sao Paulo (sa-east-1) ★

Name

You cannot change the name once the cluster is created.

nosqlDatabase

FREE

Create

[Access Advanced Configuration](#)

Free forever! Your M0 cluster is ideal for experimenting in a limited sandbox. You can upgrade to a production cluster anytime.

[I'll deploy my database later](#)

Desplegar bb.dd. en nube

Queda elegir un **proveedor de servicios** cloud donde se desplegará tu bb.dd. AWS es el más comúnmente elegido. Luego debemos elegir la **región del datacenter**. Siempre debe ser el país/región más cercano a tu lugar de residencia.

Finalmente, elegimos **el nombre de clúster**. Dejamos por defecto lo que ofrece, o elegimos uno nosotros.

Instalación de MongoDB

1 How would you like to authenticate your connection?

Your first user will have permission to read and write any data in your project.

Username and Password

Certificate

i We autogenerated a username and password for your first database user in this project using your MongoDB Cloud registration information. **x**

Create a database user using a username and password. Users will be given the *read and write to any database* [privilege](#) by default. You can update these permissions and/or create additional users later. Ensure these credentials are different to your MongoDB Cloud username and password.

Username

foluna

Password 

8UglvnFAPPFQUNkb

 Autogenerate Secure Password

 Copy

Create User

Crear usuario y contraseña para la bb.dd.


Debemos elegir un usuario y una contraseña para la bb.dd. que estamos creando. **Es importante guardar este dato en un lugar seguro, ya que luego lo utilizaremos para conectarnos a esta base de datos.**

Recordemos que la misma está en un clúster remoto, expuesta en Internet, por lo tanto debemos asegurarnos de almacenar bien esta información para no perder el acceso a dicho clúster.

Instalación de MongoDB


✓ Where would you like to connect from?

Enable access for any network(s) that need to read and write data to your cluster.



My Local Environment

Use this to add network IP addresses to the IP Access List. This can be modified at any time.



Cloud Environment

Use this to configure network access between Atlas and your cloud or on-premise environment. Specifically, set up IP Access Lists, Network Peering, and Private Endpoints.

ADVANCED

Add entries to your IP Access List

Only an IP address you add to your Access List will be able to connect to your project's clusters. You can manage existing IP entries via the [Network Access Page](#).

IP Address	Description	
<input type="text" value="Enter IP Address"/>	<input type="text" value="Enter description"/>	<input type="button" value="Add My Current IP Address"/>
<input type="button" value="Add Entry"/>		

IP Access List	Description	
186.18.162.96/32	My IP Address	<input type="button" value="EDIT"/> <input type="button" value="REMOVE"/>

Indicar desde dónde nos conectaremos

Nos conectaremos, en principio, desde un entorno creado localmente. Esta es la opción a marcar en este paso.

Luego, podemos optar porque el servidor reconozca nuestra IP de Internet, para que tome como referencia válida cada futura petición de conexión y administración que realicemos sobre esta bb.dd.

- Pulsamos el botón **Add My Current IP Address**
- Por último, pulsamos el botón **Finish and Close**



Instalación de MongoDB

Project 0

Data Services App Services Charts

FERNANDO OMAR'S ORG - 2023-03-20 > PROJECT 0

Database Deployments

Find a database deployment...

+ Create

Load sample datasets to nosqlDatabase.

Atlas provides sample data you can load into your Atlas clusters. You can use this data to quickly get started exploring with data in MongoDB.

Load sample dataset Dismiss

nosqlDatabase Connect View Monitoring Browse Collections ... FREE

Enhance Your Experience

For production throughput and richer metrics, upgrade to a dedicated cluster now!

Upgrade

R 0 W 0 Last 37 minutes 100.0/s

Connections 0 Last 37 minutes 100.0

In 0.0 B/s Out 0.0 B/s Last 37 minutes 100.0 B/s

Data Size 0.0 B Last 37 minutes 512.0 MB

VERSION	REGION	CLUSTER TIER	TYPE	BACKUPS	LINKED APP SERVICES	ATLAS SEARCH
5.0.15	AWS / Sao Paulo (sa-east-1)	M0 Sandbox (General)	Replica Set - 3 nodes	Inactive	None Linked	Create Index

Despliegue de bb.dd.

Finalizada la configuración del clúster, ya estamos dentro de la interfaz web de MongoDB.

Pulsa, si gustas, la opción **Load sample Datasets** para generar bases de datos de ejemplo en tu entorno.

El proceso demora algunos minutos.

Instalación de MongoDB

Project 0

Data Services App Services Charts

FERNANDO OMAR'S ORG - 2023-03-20 > PROJECT 0

Database Deployments

Find a database deployment...

+ Create

Load sample datasets to nosqlDatabase.

Atlas provides sample data you can load into your Atlas clusters. You can use this data to quickly get started exploring with data in MongoDB.

Load sample dataset Dismiss

nosqlDatabase Connect View Monitoring Browse Collections ... FREE

Enhance Your Experience

For production throughput and richer metrics, upgrade to a dedicated cluster now!

Upgrade

Performance Metrics:

- R 0** (Reads): Last 37 minutes, 100.0/s
- W 0** (Writes): Last 37 minutes, 100.0/s
- Connections 0**: Last 37 minutes, 100.0
- In 0.0 B/s** (Input): Last 37 minutes, 100.0 B/s
- Out 0.0 B/s** (Output): Last 37 minutes, 100.0 B/s
- Data Size 0.0 B**: Last 37 minutes, 512.0 MB

VERSION	REGION	CLUSTER TIER	TYPE	BACKUPS	LINKED APP SERVICES	ATLAS SEARCH
5.0.15	AWS / Sao Paulo (sa-east-1)	M0 Sandbox (General)	Replica Set - 3 nodes	Inactive	None Linked	Create Index

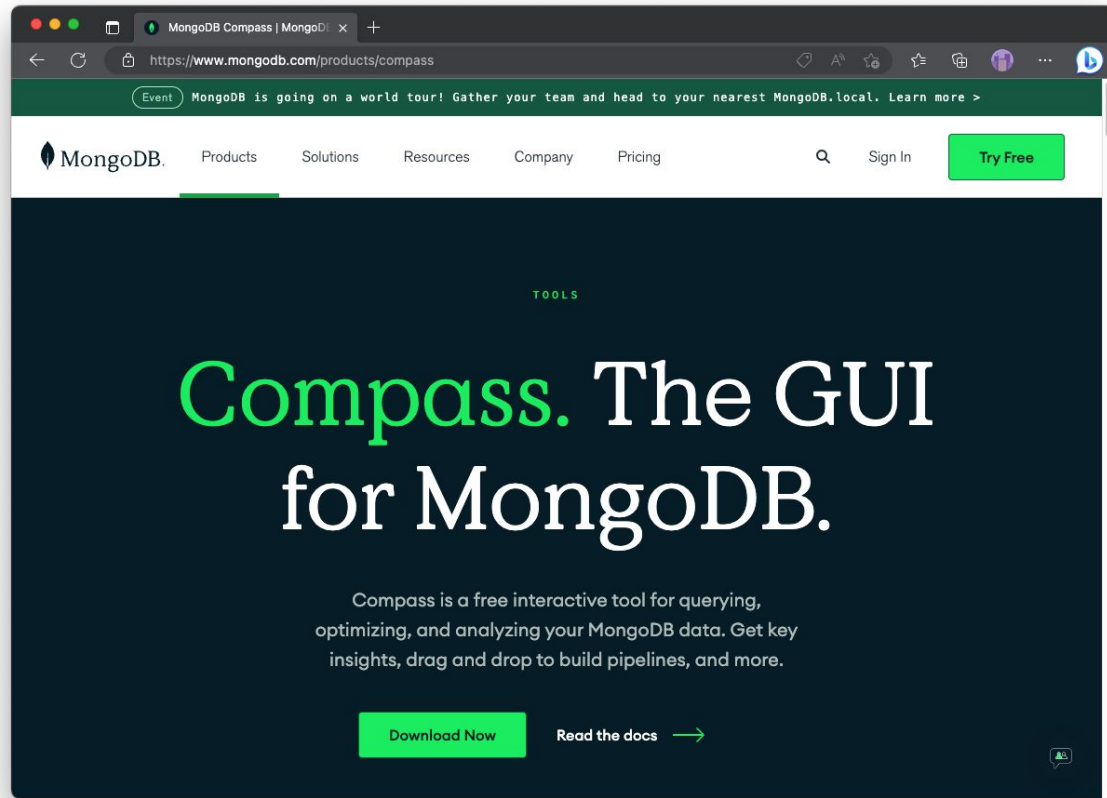
Despliegue de bb.dd.

Tenemos la opción de trabajar desde la interfaz web, o de utilizar un cliente de Administración (DBMS) remoto.

Optamos por esta última alternativa, instalando **Compass** en nuestra computadora.

MongoDB compass

MongoDB



Instalar Compass

La URL para descargar Compass es la siguiente: [MongoDB Compass | Install](https://www.mongodb.com/products/compass)

Descargado el archivo para tu S.O., instala este software tal como instalas cualquier otra aplicación en tu computadora.

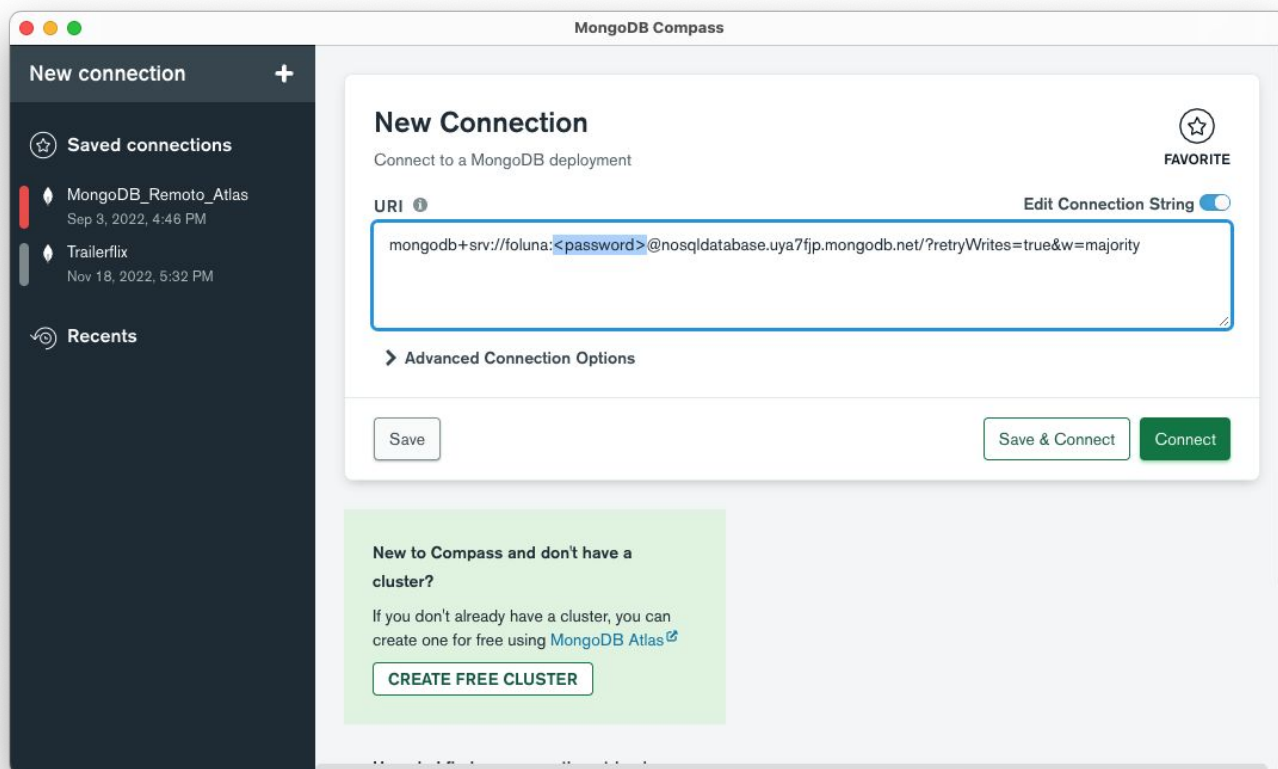
MongoDB

Conectar Compass con un Clúster

A través de la documentación de MongoDB, podemos acceder a la cadena de conexión para poder conectar el cliente Compass con nuestro clúster MongoDB.

Sobre los datos de la cadena de conexión, veremos el nombre de nuestro usuario, nombre del clúster, y demás cuestiones técnicas.

En el apartado **<password>** debemos reemplazar dicho texto, por el password generado cuando nos registramos por primera vez.

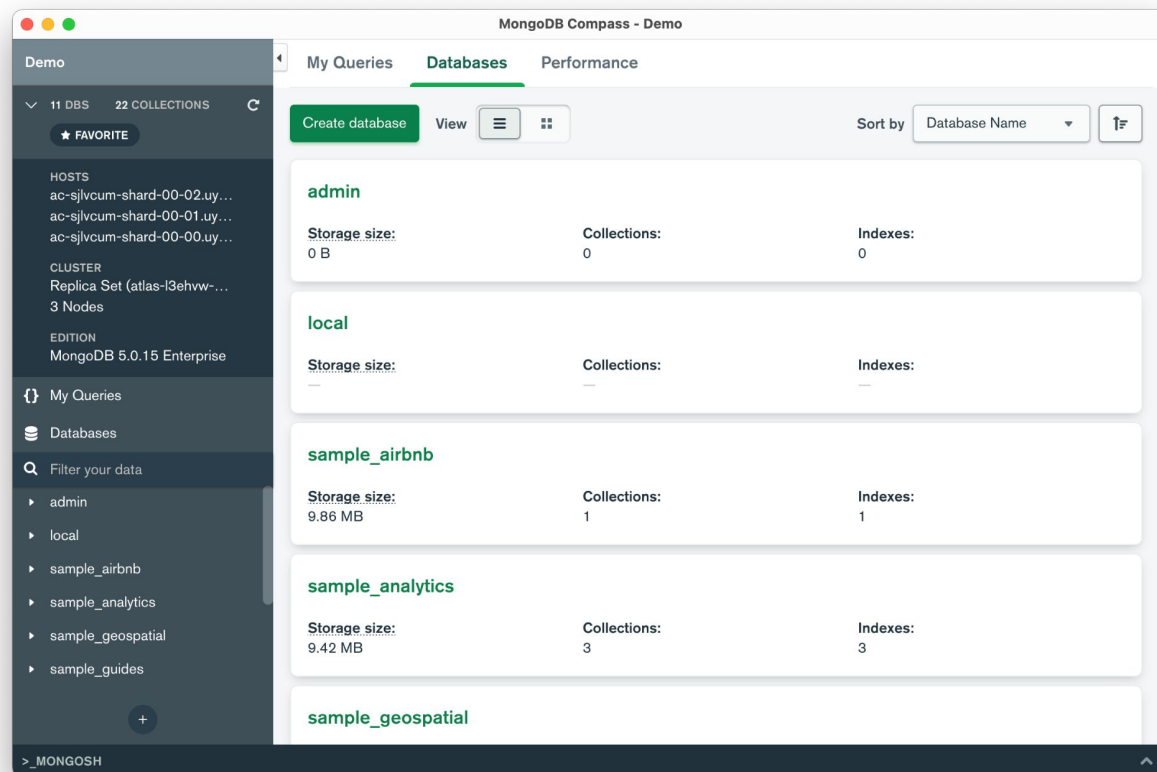


MongoDB

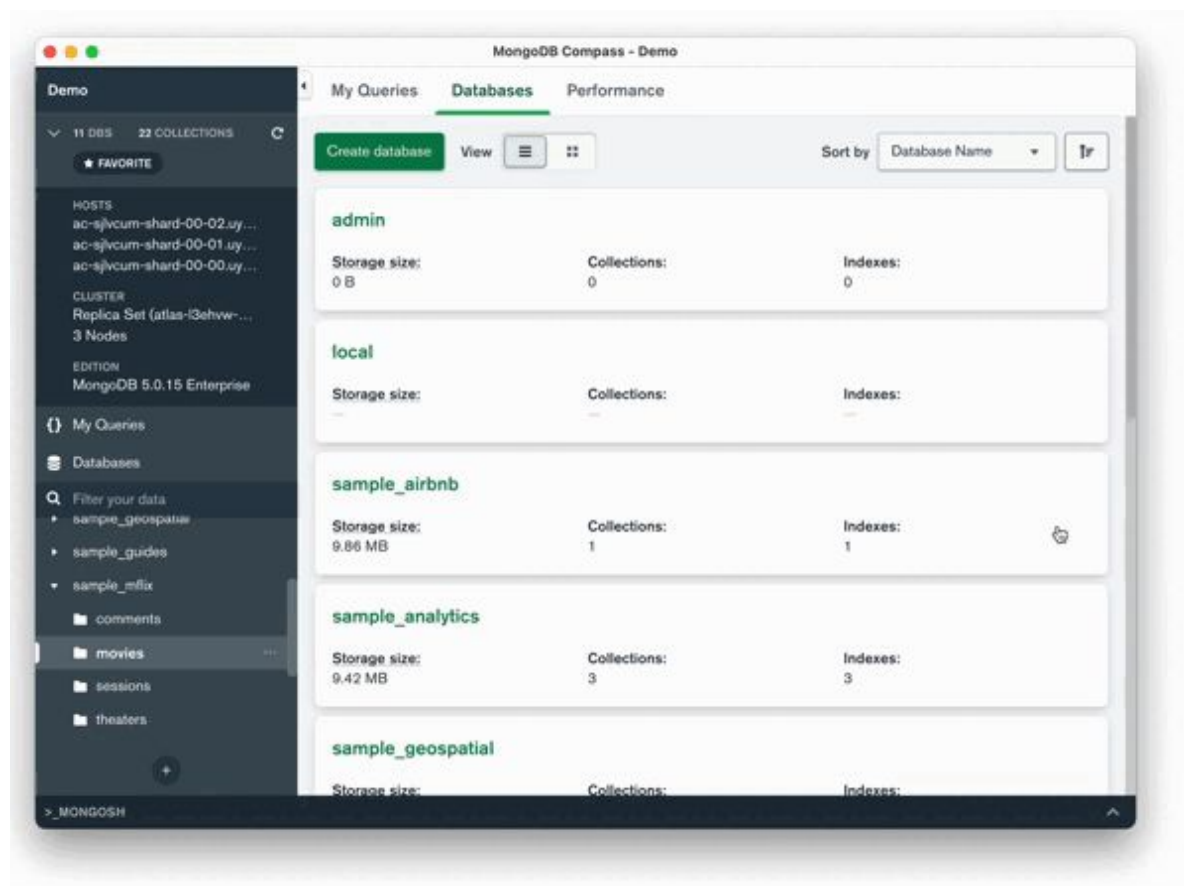
GUI de Compass

Ya conectados al Clúster, accedemos al apartado Databases para poder ver qué bases de datos tenemos.

Si optamos por la opción Load Sample Datasets ofrecida en la interfaz web de MongoDB, veremos las bases de datos ya creadas en pantalla.



MongoDB



Database

Del listado de ejemplos creado, navegamos hasta el denominado **sample_mflix**, y elegimos el documento: **Movies**. En éste veremos un detalle de películas mockup.

Cada una de estas películas tiene una estructura de datos que la representa (*género, título, director, año, etc.*). Esta es la forma en la cual MongoDB almacena la información.

Espacio de trabajo 🛠️

Veamos una Demo de la profe donde importará un set de datos a MongoDB, y navegará por el mismo para verificar el correcto proceso de importación.



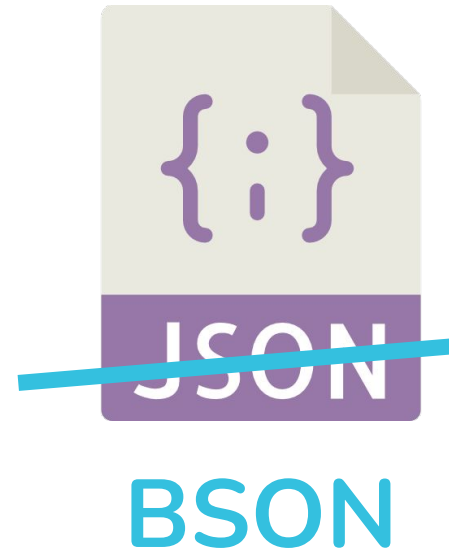
Si deseas trabajar con un set de datos propio, tenemos la opción de importar el mismo a una base de datos existente, o creando una base de datos nueva. Su importación puede realizarse desde un archivo JSON o desde un archivo CSV.

El formato BSON

El formato BSON

BSON (*Binary JSON*) es un formato de serialización binaria utilizado por MongoDB para almacenar y transferir datos.

Es una extensión del formato JSON, pero representando los datos en forma binaria, lo que lo hace más compacto y eficiente en términos de espacio y tiempo de procesamiento que el formato JSON.



El formato BSON

Veamos, a continuación, los diferentes tipos de bases de datos NoSQL.

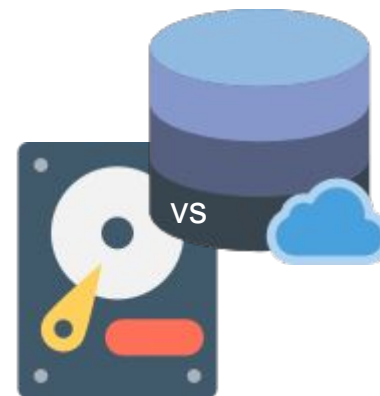
Característica	Descripción
Tipos de datos	BSON admite la mayoría de los tipos de datos utilizados en programación, incluyendo tipos de datos numéricos, de cadena, booleanos, fechas, objetos y matrices.
Soporta tipos de datos adicionales	También admite tipos de datos adicionales, como tipos de datos binarios, valores nulos, valores regulares y referencias a objetos.
Tamaño compacto	Es un formato compacto y eficiente en términos de espacio, lo que significa que utiliza menos espacio de almacenamiento que JSON.
Procesamiento rápido	Es más eficiente en el procesamiento que JSON, ya que la información se representa en forma binaria y se puede leer y escribir más rápidamente que en formato de texto plano.
Compatibilidad con lenguajes de programación	Es compatible con la mayoría de los lenguajes de programación, lo que lo hace ideal para ser utilizado como formato de intercambio de datos en aplicaciones distribuidas.

Instalación local de MongoDB versus Clúster

Instalación local de MongoDB versus Clúster

MongoDB es totalmente instalable de forma local, en nuestra computadora, además de poder utilizarse desde un clúster Cloud.

Para definir de qué forma instalar el mismo, podemos basarnos en una serie de consideraciones que pueden influir en la decisión de qué tipo de instalación debemos elegir.

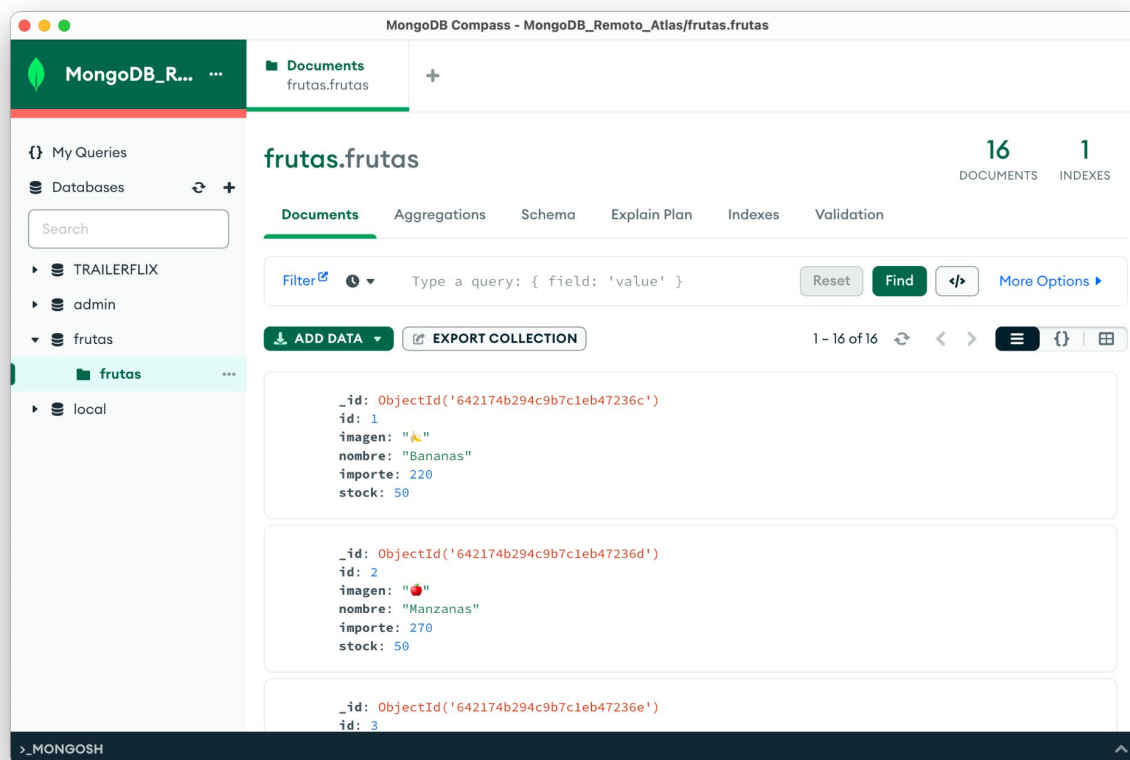


Instalación local de MongoDB versus Clúster

Características	Descripción
Control total	Instalar MongoDB localmente nos da el control sobre el hardware y software. Esto es útil si necesitamos una configuración específica y preferimos tener control sobre el hardware y la red.
Escalabilidad	Si necesitamos aumentar el almacenamiento y procesamiento de datos, un clúster de MongoDB es la mejor opción, ya que escala horizontalmente agregando nodos de manera fácil y rápida.
Disponibilidad	Si la disponibilidad de datos es crítica un clúster de MongoDB es más adecuado, ya que garantiza disponibilidad total de los datos mediante réplicas y el uso de conmutación automática por error.
Mantenimiento	Instalar MongoDB localmente lo hace más fácil de mantener, pero requiere más trabajo de configuración y backup. Un clúster Cloud es más fácil de administrar, ya que las tareas de mantenimiento se realizan automáticamente.
Lenguajes de programación	Es compatible con la mayoría de los lenguajes de programación, lo que lo hace ideal para ser utilizado como formato de intercambio de datos en aplicaciones distribuidas.
Costo	Los clúster pueden demandar con el tiempo, un presupuesto de dinero mayor si necesitamos escalar en nodos y/u otros servicios en nube.

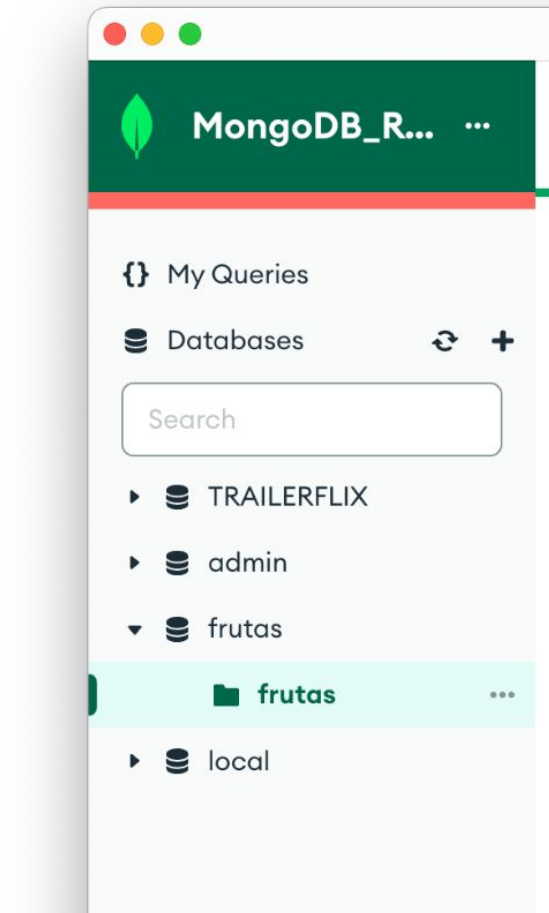
Estructura de datos

Estructura de datos



La interfaz de Compass posee una estructura bastante simple y minimalista para navegar entre los set de datos (*bases de datos*), en MongoDB.

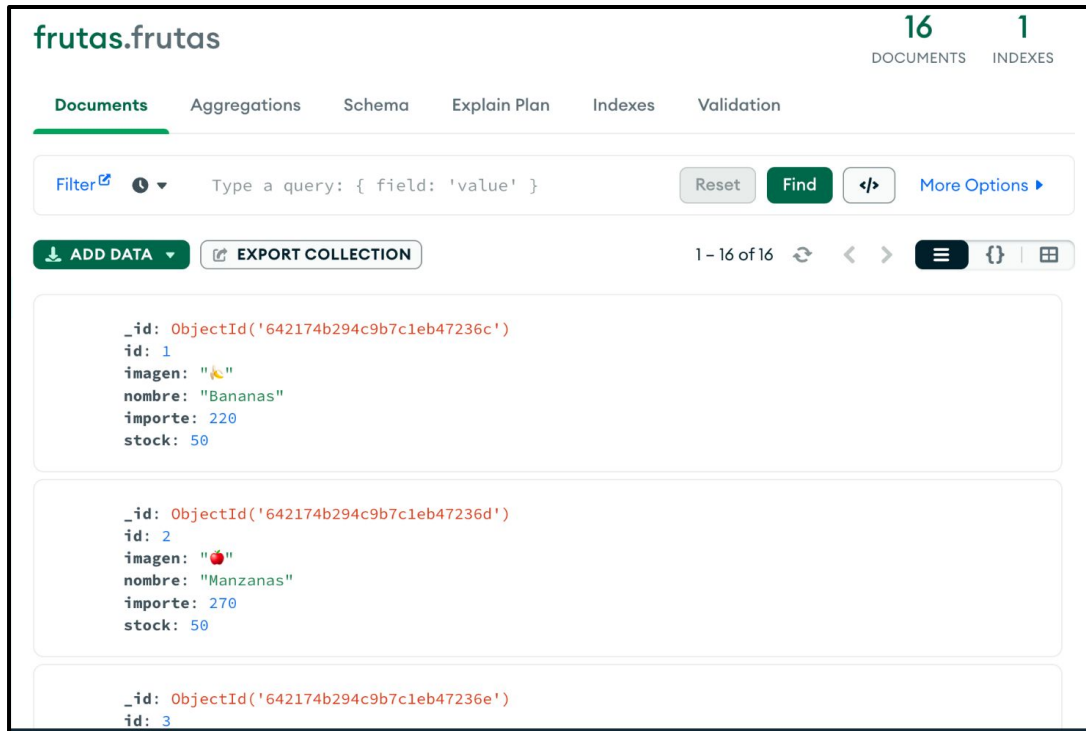
Estructura de datos



El panel lateral izquierdo, representa todas las bases de datos creadas, y en el interior de estas las colecciones (**Collections**).

Si conoces SQL, el equivalente de **colecciones** en SQL es **Tablas**. En cada colección se almacena el conjunto de datos definidos para ésta.

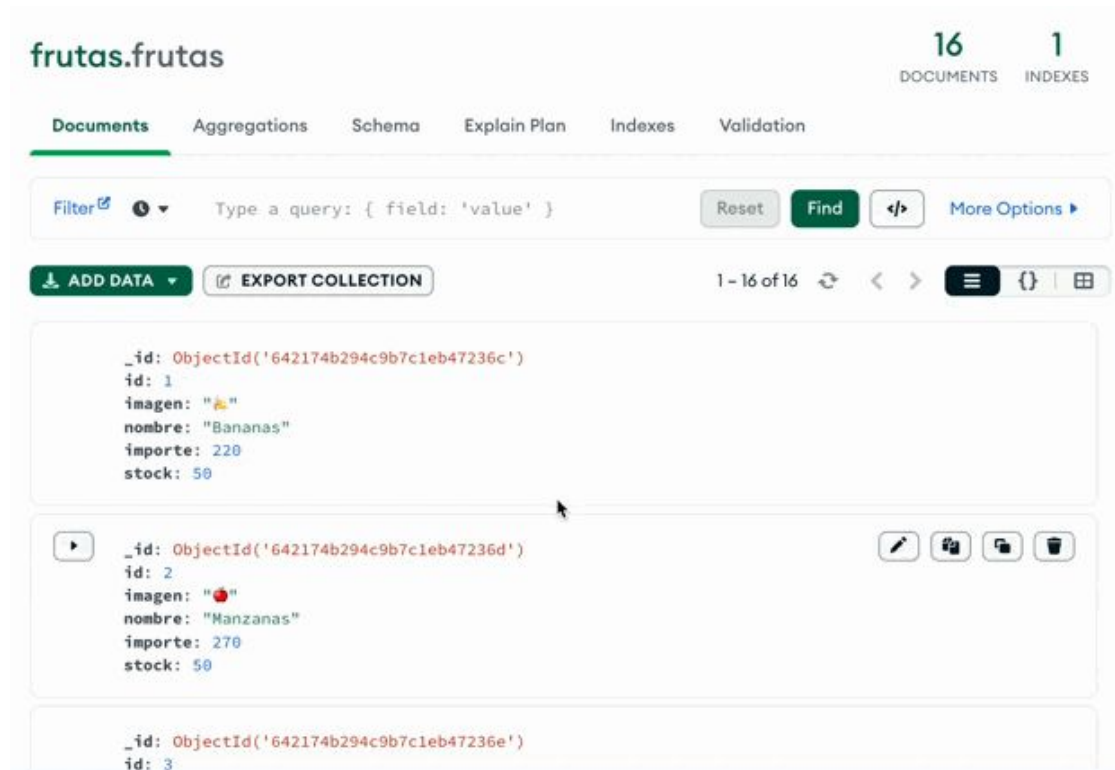
Estructura de datos



En el panel central, vemos el set de datos que conforman cada colección. Este conjunto de datos se denomina **documentos**.

En comparación nuevamente con SQL, cada **documento** equivale a un **registro** en SQL.

Estructura de datos



Podemos navegar visualmente por el conjunto de documentos de la Colección, y hasta visualizar de tres maneras diferentes dicho contenido:

- formato **lista**
- formato **objeto**
- formato **tabla**

Estructura de datos

```
_id: ObjectId('642174b294c9b7c1eb47236c')  
id: 1  
imagen: "🍌"  
nombre: "Bananas"  
importe: 220  
stock: 50
```

Más allá de que nuestros documentos posean un ID o índice definido, MongoDB suele agregar su propio índice. Este se define bajo la propiedad `_id`, y su estructura parte de lo que se conoce como **Identificador Único Universal (UUID)**.

Estructura de datos

```
_id: ObjectId('642174b294c9b7c1eb47236c')  
id: 1  
imagen: "🍌"  
nombre: "Bananas"  
importe: 220  
stock: 50
```

El identificador, **ObjectId**, es generado por MongoDB al insertar un documento en una colección, bajo una estructura representada por un identificador único de 12 bytes que consta de:

- Un **campo de tiempo de 4 bytes** que representa la hora en que se creó el ObjectId
- Un **campo de máquina de 3 bytes** que identifica la máquina donde se creó el ObjectId
- Un **campo de proceso de 2 bytes** que identifica el proceso que creó el ObjectId
- Un **contador de 3 bytes** que se incrementa para cada ObjectId creado con el mismo proceso en la misma máquina y en el mismo segundo

Estructura de datos

Dentro de los usos más efectivos para **ObjectId**, podemos destacar:

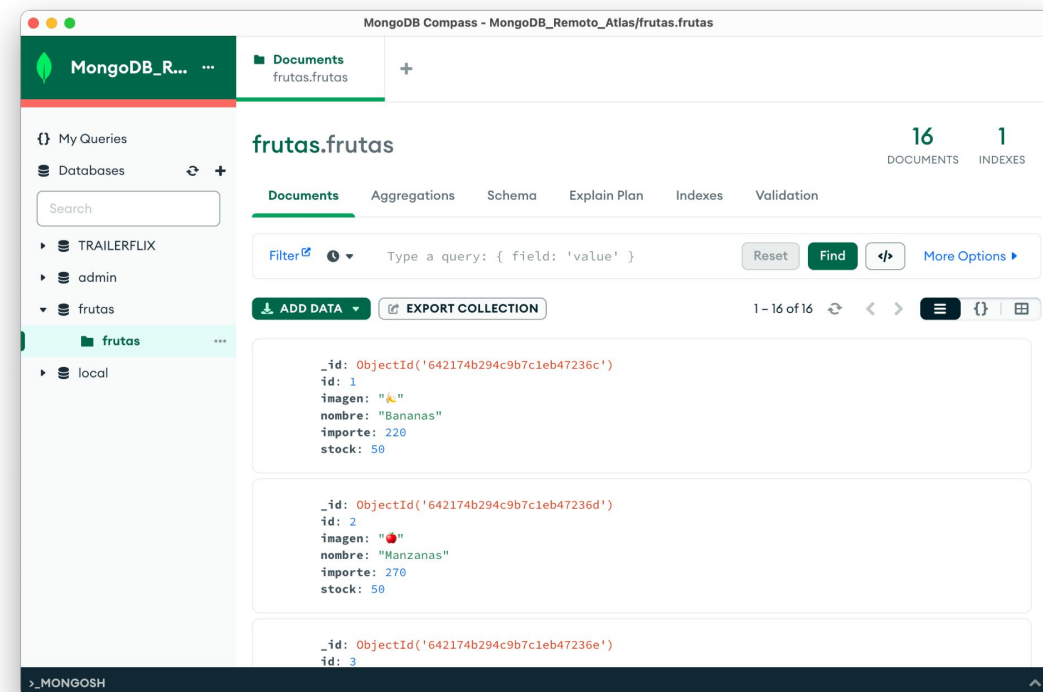
- Garantiza que cada documento tenga un identificador único seguro en una colección
- Ordena los documentos en una colección en función del orden en que se insertaron
- Proporciona un indicador visual de la fecha y hora en que se creó un documento
- Se utiliza también en algunas consultas para buscar documentos por su identificador único

```
_id: ObjectId('642174b294c9b7c1eb47236c')  
id: 1  
imagen: "🍌"  
nombre: "Bananas"  
importe: 220  
stock: 50
```

Estructura de datos

Además, Compass cuenta con un panel de acciones donde podemos representar diferentes necesidades, como ser:

- filtrar información
- aplicar funciones de agregación (totales, agrupamiento)
- explorar el Schema a nivel estadístico
- verificar los posibles índices
- definir reglas de validación para los datos a almacenar



Practicas

Utilizando la aplicación Compass, deberás crear una nueva base de datos en MongoDB.

- El nombre de la misma será '**frutas**'

El nombre de la **Colección**, será exactamente igual.

- Ingresa luego a la base de datos y a la colección creada dentro de ésta
- Utiliza el botón **Add Data** para seleccionar un archivo que desees importar
- Ubica el archivo **frutas.json** trabajado en otras Unidades, y ábrelo
- En el proceso de importación, selecciona la opción Import as JSON
- Espera a que finalice la importación de datos
- Navega luego por la colección creada, para verificar su correcta importación

Muchas gracias.



Ministerio de Economía
Argentina

Secretaría de
Economía del Conocimiento

*primero
la gente*