

# Desarrollo Backend

Bienvenid@s

MongoDB I

*(qué son - tipos de bb.dd. - Introducción a MongoDB)*

Clase 14



Pon a grabar la clase



# Temario

- Fundamento de bases de datos
  - qué son
  - tipos de bb.dd.
  - bases de datos NoSQL
- MongoDB
  - crear un clúster
  - instalar Compass
  - visualizar bb.dd. y Documentos
  - el formato BSON
  - Local vs. Clúster
  - Importar estructuras JSON



# Fundamento de bases de datos

# Fundamento de bases 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.



# Qué son

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.



**UNTREF**

UNIVERSIDAD NACIONAL  
DE TRES DE FEBRERO

# Qué son

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**



# Qué son

## 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!*





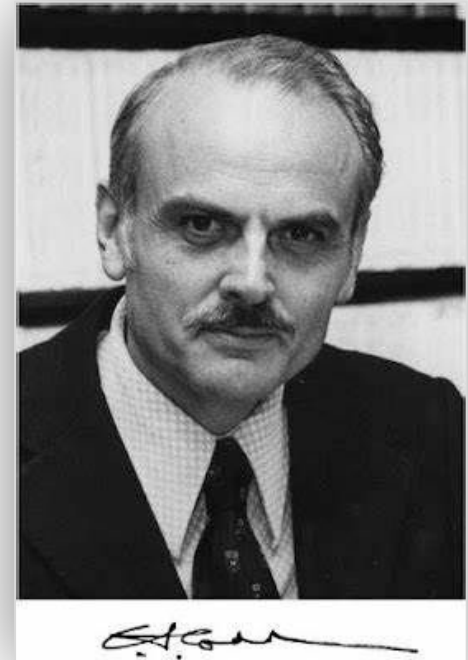
# Qué son

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

IBM vendía Mainframes comerciales y estos equipos tenían 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*.

IBM llegó recién en 1993 a este mercado, con su [software de bases de datos: DB2](#).



# Qué son

**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.

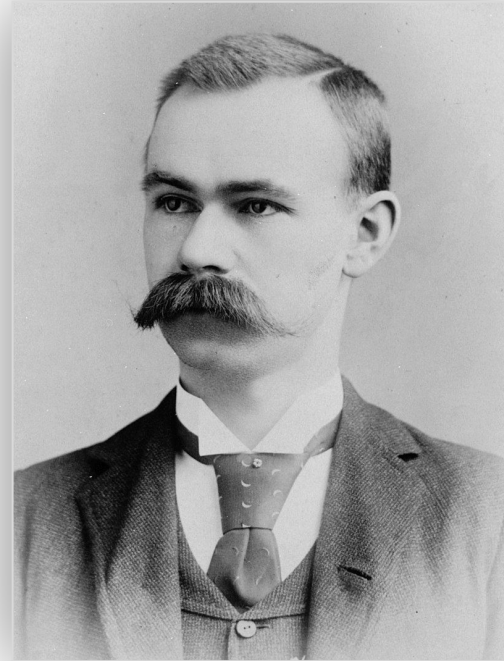
En 1884, [Herman Hollerith](#), diseñó la máquina automática de tarjetas perforadas, ante la necesidad de registrar y procesar rápidamente toda la información de los barcos que llegaban a NYC con inmigrantes europeos.



# Qué son

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.



# Tipos de bb.dd.

# Tipos de bb.dd.

Dentro de los diferentes tipos de bases de datos que encontramos 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.



# Base de datos jerárquica

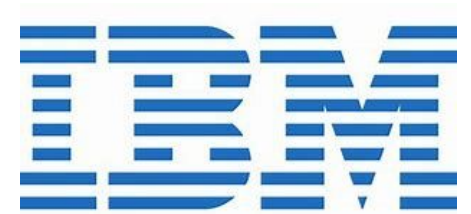
Esta base de datos corresponde a un tipo de sistema de gestión de bases de datos (DBMS) en el que los datos se organizan en una estructura de árbol.

Cada nodo en el árbol representa un registro que puede tener varios campos de datos. Cada registro puede tener uno o varios registros secundarios que están conectados a él, y cada uno de esos registros secundarios puede tener a su vez registros secundarios adicionales, creando una estructura jerárquica.



# Base de datos jerárquica

Un software de base de datos jerárquica comercial, que podemos encontrar en la actualidad, es **IBM IMS** (*Information Management System*), desarrollado por Big Blue.



IMS se utiliza en aplicaciones empresariales para almacenar y recuperar grandes cantidades de datos estructurados.



# Base de datos relacionales

Una base de datos relacional es un tipo de sistema de gestión de bases de datos (DBMS) que almacena datos en tablas que se relacionan entre sí mediante claves primarias y claves foráneas.

Cada tabla representa una entidad y cada fila de la tabla representa una instancia de esa entidad. Los datos se organizan en columnas que describen las propiedades de la entidad.



# Base de datos relacionales

Como bien hablamos al inicio, Oracle es la base de datos relacional más popular hasta hoy en día.

Si bien tiene mucha competencia en el mercado de bb.dd. relacionales, solo tres bases de datos de este tipo acaparan el 60% de este mercado (*Oracle, MySQL, SQL Server*), siendo también Oracle la empresa dueña de MySQL.



**UNTREF**

UNIVERSIDAD NACIONAL  
DE TRES DE FEBRERO

# Base de datos de objetos

Una base de datos de objetos es un tipo de sistema de gestión de bases de datos (DBMS) en el que los datos se modelan como objetos en lugar de como registros o tuplas como sucede en las bases de datos relacionales.

Los objetos pueden contener datos y métodos, y pueden estar interrelacionados a través de herencia, composición u otras relaciones de objeto.



# Base de datos de objetos

Uno de los software comerciales de base de datos de objetos más conocido, es **Versant Object Database**.

Versant se utiliza en aplicaciones empresariales que requieren una gestión de datos compleja y un alto rendimiento, además ofrece funciones de transacciones distribuidas y replicación de datos para garantizar la integridad de los datos y la disponibilidad en caso de fallos de hardware o redes.

The logo for Versant, featuring the word "VERSANT" in a red, serif, all-caps font.

**UNTREF**

UNIVERSIDAD NACIONAL  
DE TRES DE FEBRERO

# Base de datos de red

Las bases de datos de red son un tipo de sistema de gestión de bases de datos (DBMS) que se utilizó fuertemente en la década de 1970 y 1980. Aquí, los datos se organizan en una estructura de grafo; en la que cada nodo representa un registro y cada arco representa una relación entre los registros.

A diferencia de las bb.dd. relacionales, éstas permiten relaciones múltiples entre los registros, por lo cual un registro puede tener varios padres o varios hijos.



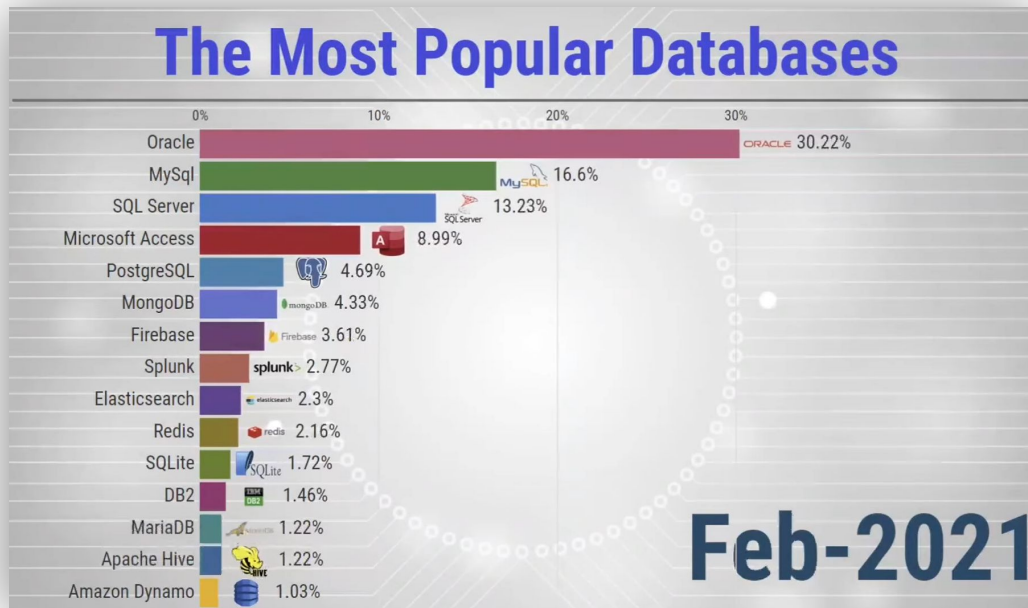
# Base de datos de red

Algunos ejemplos de software comercial de bases de datos de red son **IDMS** (*Integrated Database Management System*) de **CA Technologies** y **Univac DMS-1100** de Unisys.

Aunque las bases de datos de red ya no son tan comunes como lo eran en el pasado, todavía se utilizan en algunas aplicaciones especializadas, especialmente en el ámbito de los sistemas de información geográfica (*G/S*) y la investigación científica.



# Para conocer la evolución de las 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.

# Bases de datos NoSQL



# Base de datos NoSQL

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.



**UNTREF**

UNIVERSIDAD NACIONAL  
DE TRES DE FEBRERO

# Base de datos NoSQL

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.

# Base de datos NoSQL

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

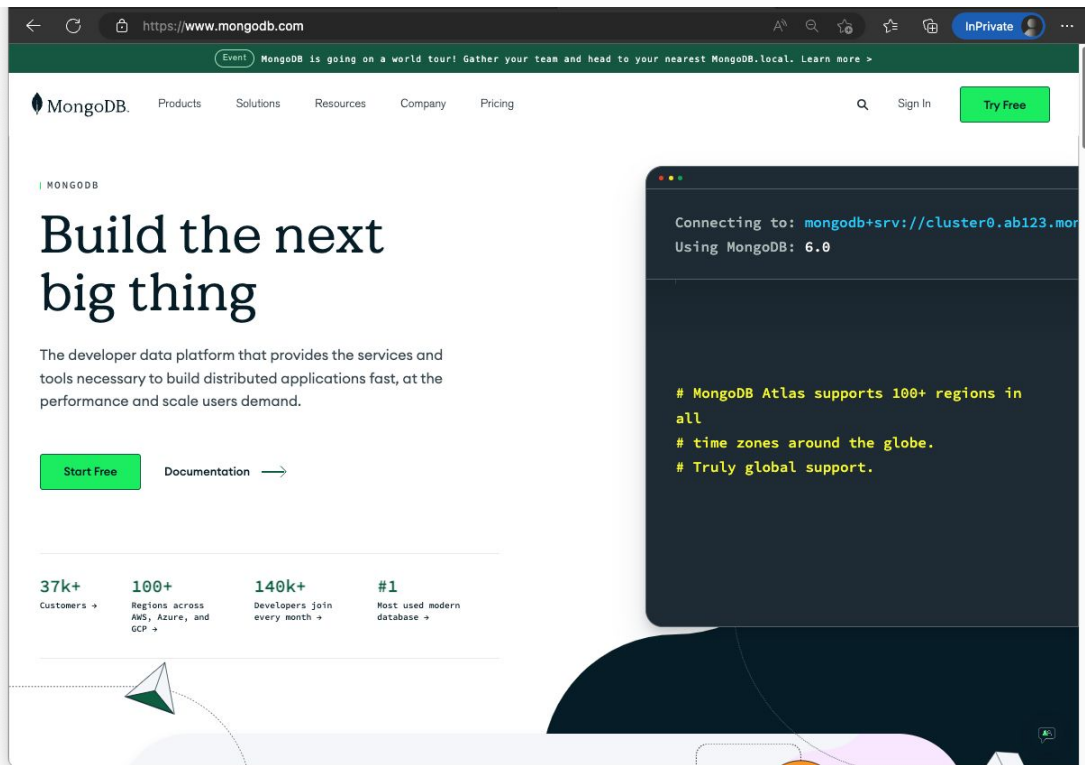


*Por popularidad, antigüedad, y por ser parte del stack [MERN - MEAN - MEVN](#),*

***MongoDB*** es la elegida para ser desarrollada, a continuación.

# MongoDB

# 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.**

**UNTREF**

UNIVERSIDAD NACIONAL  
DE TRES DE FEBRERO

# MongoDB

**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.



# MongoDB

Dentro de sus características notables, podemos destacar:

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

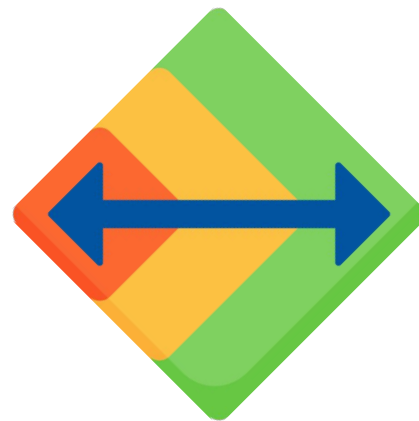


# MongoDB

## 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

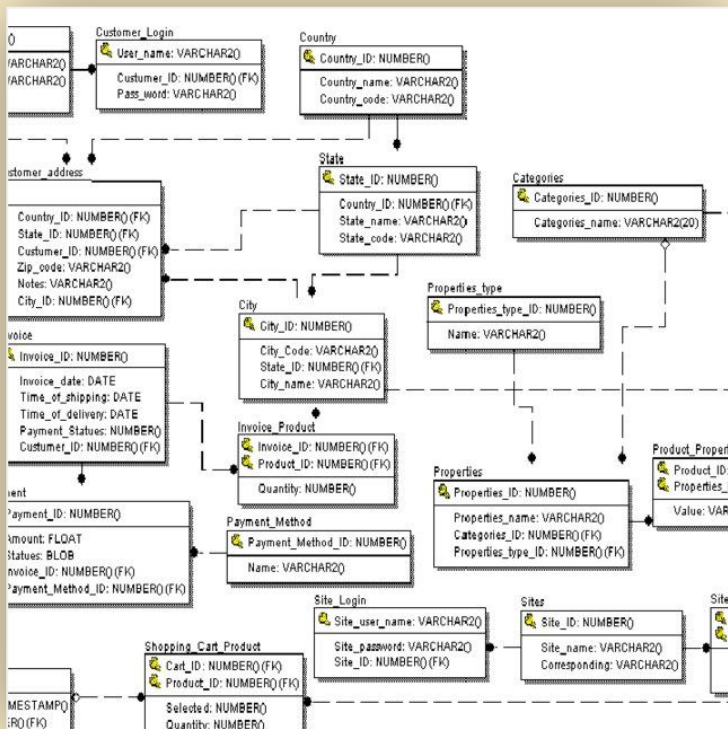
## Flexibilidad

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

# 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.



**UNTREF**

UNIVERSIDAD NACIONAL  
DE TRES DE FEBRERO

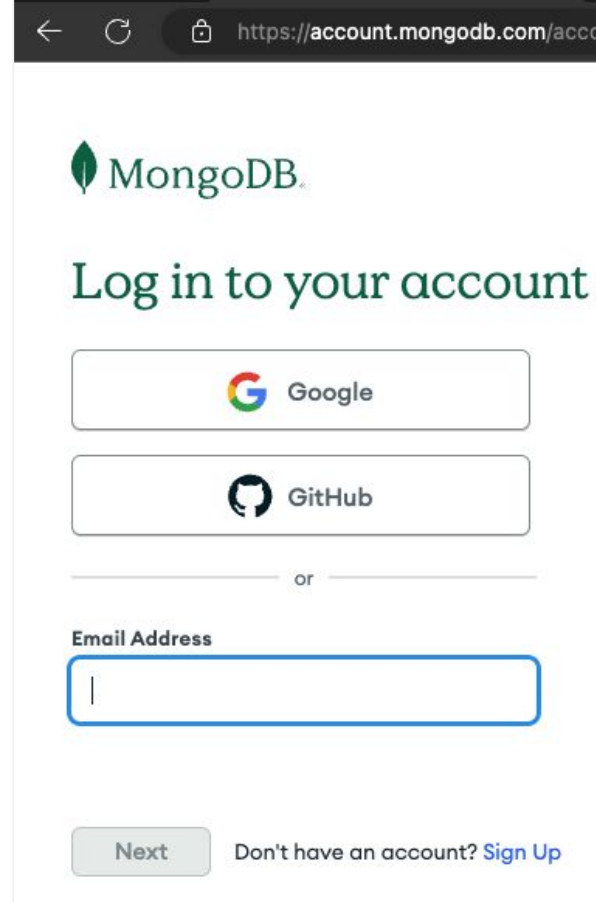
# Espacio de trabajo

# MongoDB

## SignUp

En la web oficial ([www.mongodb.com](https://www.mongodb.com)) elegimos la opción **Sign In** para registrarnos.

Podremos optar por cargar los datos manualmente (*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/account/login'. 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' and 'GitHub'. A horizontal line with the word 'or' in the center separates these from the standard login fields. There is an 'Email Address' label 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'.

**UNTREF**

UNIVERSIDAD NACIONAL  
DE TRES DE FEBRERO

## 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

# MongoDB

## 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

512 MB

RAM

Shared

vCPU

Shared

**UNTREF**

UNIVERSIDAD NACIONAL  
DE TRES DE FEBRERO



# MongoDB

M0

FREE

For learning and exploring MongoDB in a cloud environment.

STORAGE

512 MB

RAM

Shared

vCPU

Shared

Provider

aws

Google Cloud

Azure

Region

★ Recommended region ⓘ

 Sao Paulo (sa-east-1) ★

Name

You cannot change the name once the cluster is created.

nosqlDatabase

## 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.

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](#)

**UNTREF**

UNIVERSIDAD NACIONAL  
DE TRES DE FEBRERO

# 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.


**UNTREF**

UNIVERSIDAD NACIONAL  
DE TRES DE FEBRERO

# 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

Add My Current IP Address

Add Entry

IP Access List

Description

186.18.162. **xx/xx**

My IP Address

 EDIT

 REMOVE

Finish and Close

## 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 para ello, el botón **Add My Current IP Address**.

Por último, pulsamos el botón **Finish and Close**.

# MongoDB

Project 0

Data Services App Services Charts

DEPLOYMENT

Database

Data Lake PREVIEW

SERVICES

Triggers

Data API

Data Federation

Search

SECURITY

Database Access

Network Access

Advanced

New On Atlas 1

Goto

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%

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	<a href="#">Create Index</a>

## Despliegue de bb.dd.

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

*Pulsa la opción **Load sample Datasets** para generar bases de datos de ejemplo en tu entorno. El proceso demora algunos minutos.*

# MongoDB

Project 0

Data Services App Services Charts

DEPLOYMENT

Database

Data Lake PREVIEW

SERVICES

Triggers

Data API

Data Federation

Search

SECURITY

Database Access

Network Access

Advanced

New On Atlas 1

Goto

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%

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

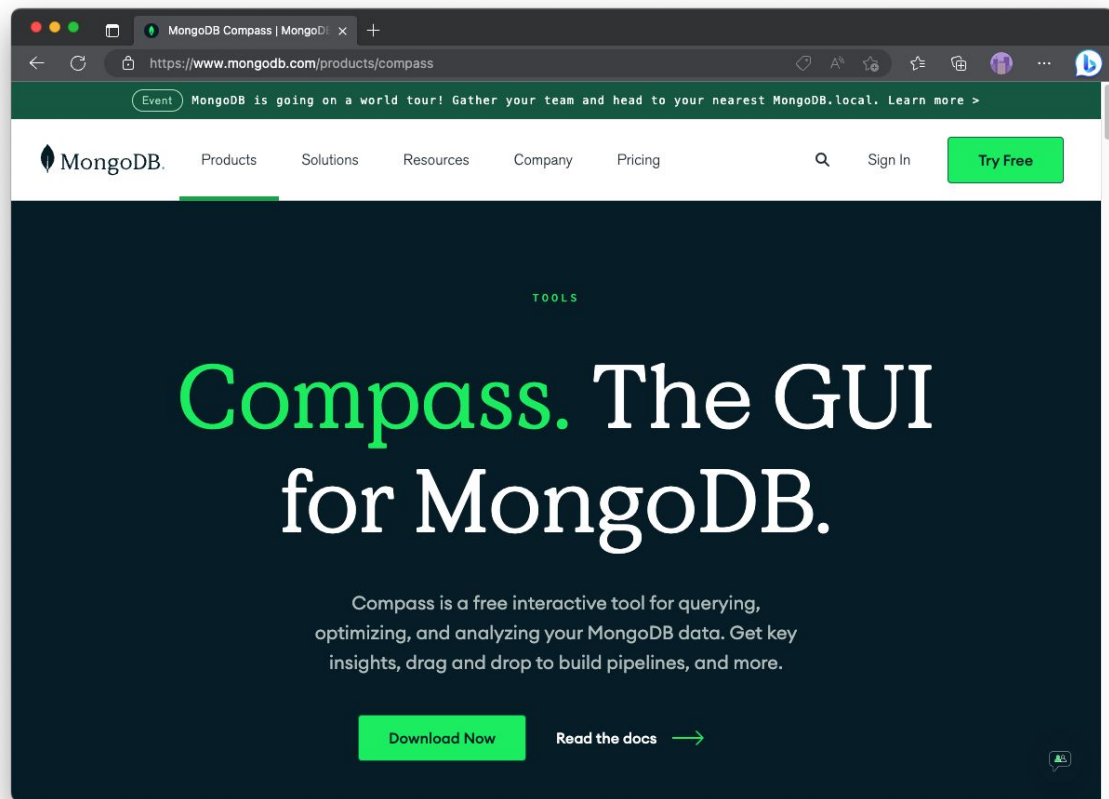
VERSION	REGION	CLUSTER TIER	TYPE	BACKUPS	LINKED APP SERVICES	ATLAS SEARCH
5.0.15	AWS / Sao Paulo (sa-east-1)	M0 Sandbox (General)	Replica Set - 3 nodes	Inactive	None Linked	<a href="#">Create Index</a>

## 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 Compass

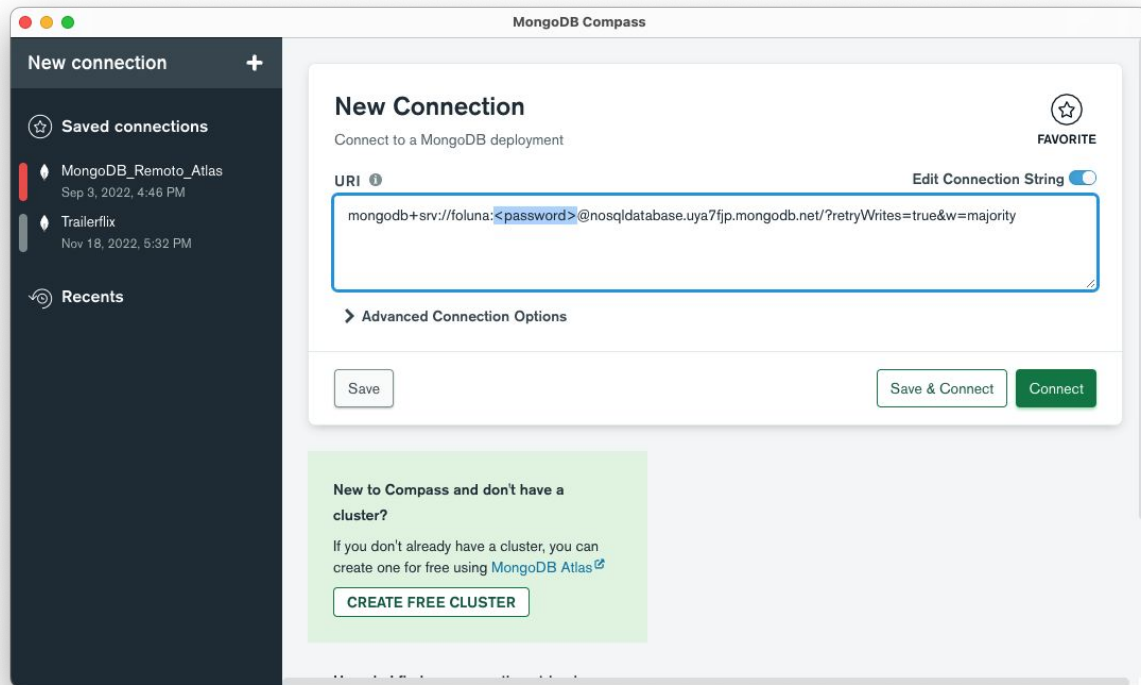


## Instalar Compass

La URL para descargar Compass es la siguiente: [MongoDB Compass | MongoDB](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 Compass



## Conectar DBMS al Clúster MongoDB

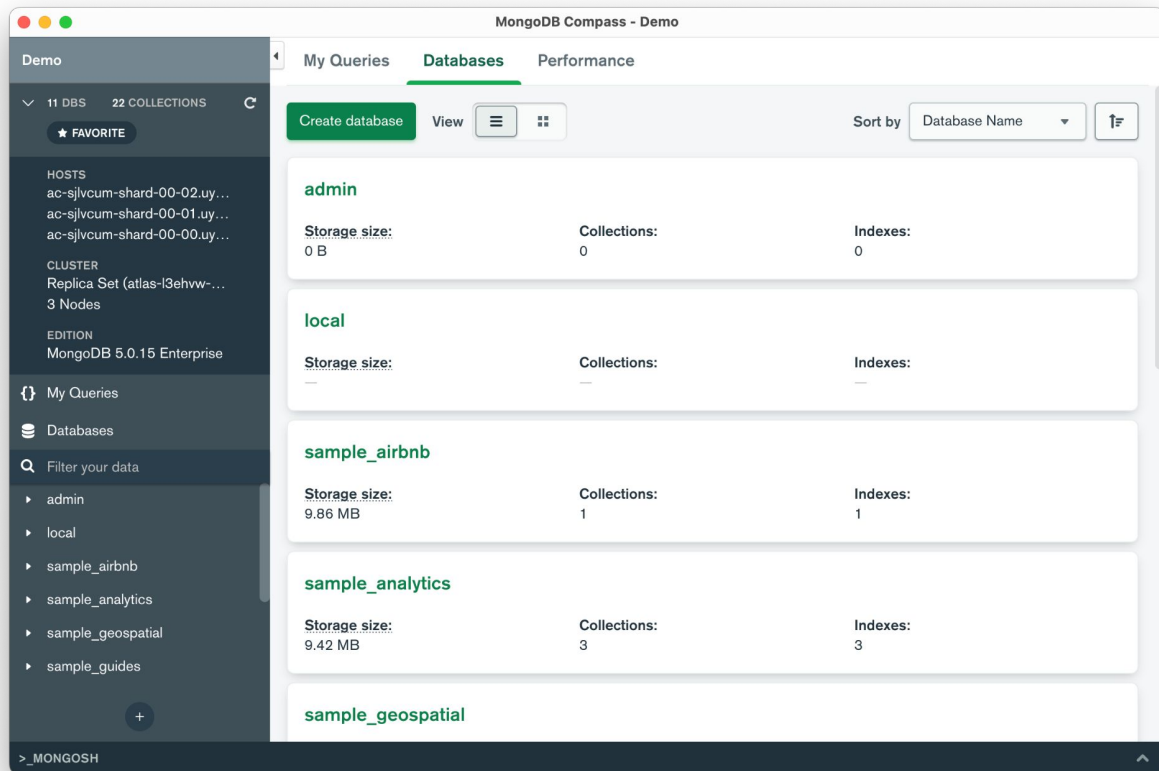
A través de la documentación de MongoDB, podemos acceder a la cadena de conexión para poder conectar el cliente Compass con nuestro clúster MongoDB.

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 Compass

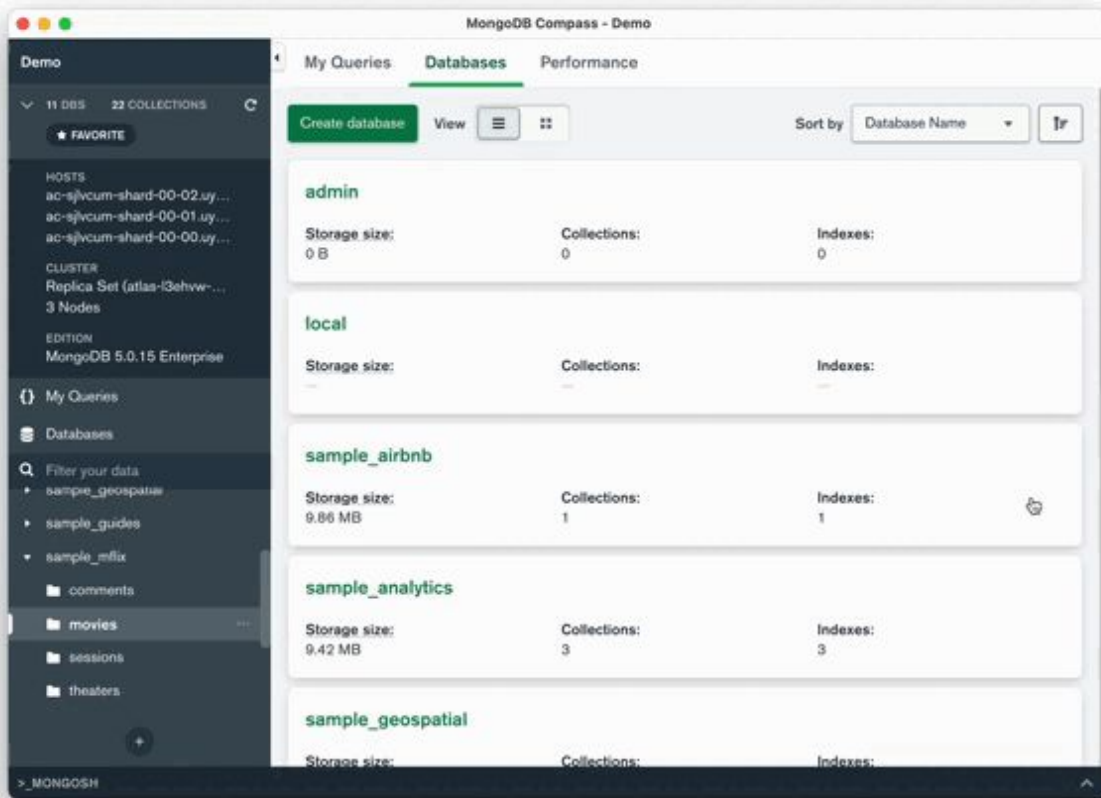


## GUI de Compass

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 Compass



## 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 🛠️

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.



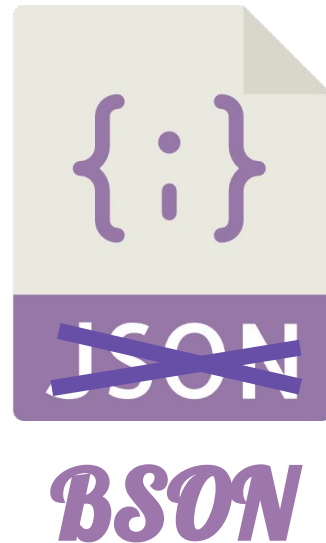
Veamos una Demo del profe donde importará un set de datos a MongoDB, y navegará por el mismo para verificar la correcta importación de su estructura.

# 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

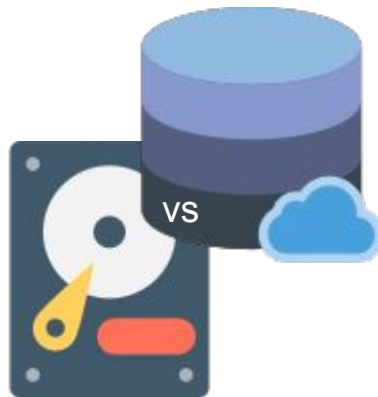
Características	Descripción
<b>Tipos de datos</b>	BSON admite la mayoría de los tipos de datos utilizados en programación, incluyendo tipos de datos numéricos, de cadena, booleanos, fechas, objetos y matrices.
<b>Soporte tipos de datos adicionales</b>	También admite tipos de datos adicionales, como tipos de datos binarios, valores nulos, valores regulares y referencias a objetos.
<b>Tamaño compacto</b>	Es un formato compacto y eficiente en términos de espacio, lo que significa que utiliza menos espacio de almacenamiento que JSON.
<b>Procesamiento rápido</b>	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.
<b>Compatibilidad con lenguajes de programación</b>	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.



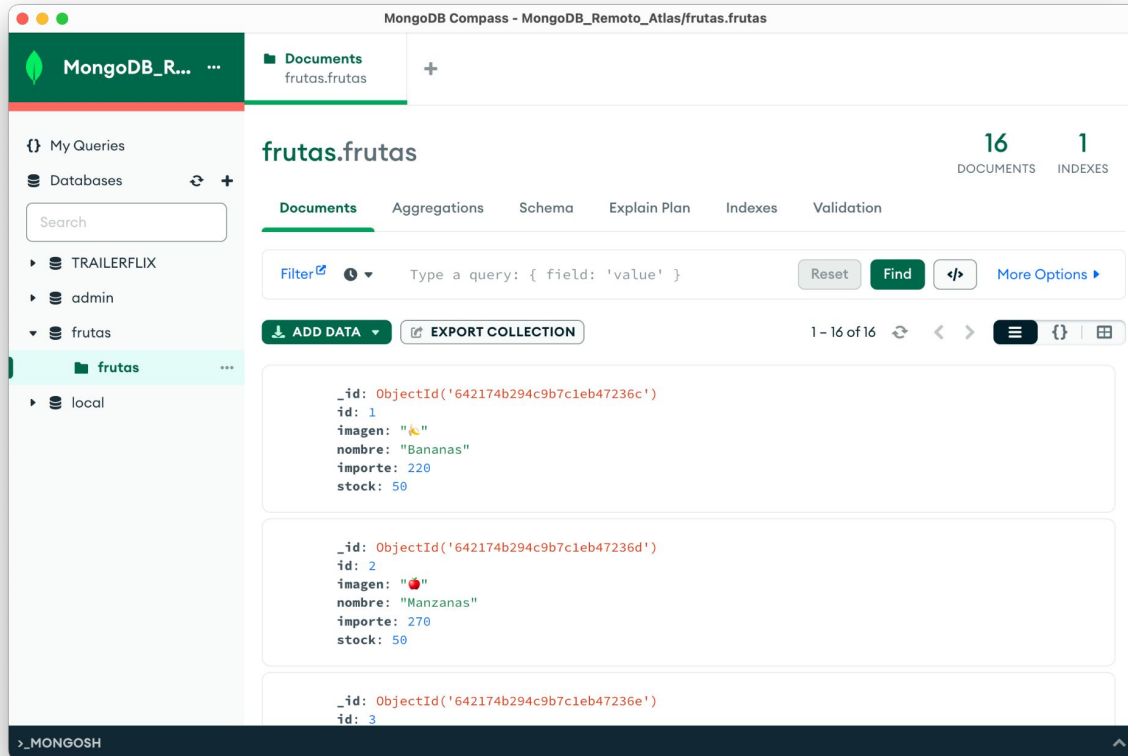


# Instalación local de MongoDB versus Clúster

Características	Descripción
<b>Control total</b>	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.
<b>Escalabilidad</b>	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.
<b>Disponibilidad</b>	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.
<b>Mantenimiento</b>	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.
<b>Lenguajes de programación</b>	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.
<b>Costo</b>	Los clúster pueden demandar con el tiempo, un presupuesto de dinero mayor si necesitamos escalar en nodos y/u otros servicios en nube.

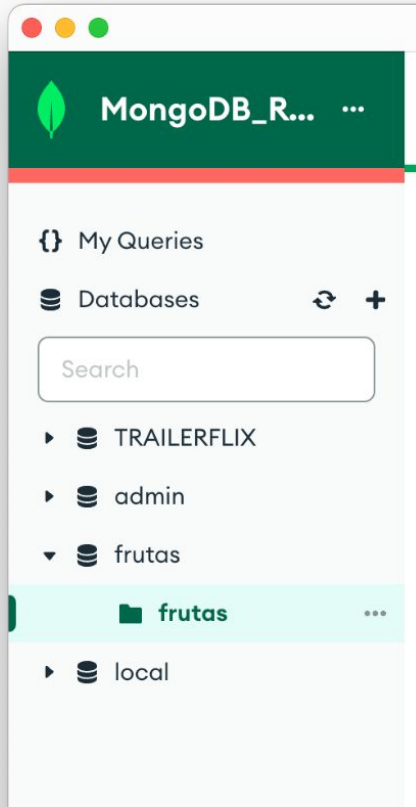
# Estructura de datos

# Estructura de datos en MongoDB



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 en MongoDB



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 MongoDB

The screenshot shows the MongoDB Compass interface for the 'frutas.frutas' collection. At the top, it indicates 16 documents and 1 index. The 'Documents' tab is selected, showing a list of documents. The first document is for 'Bananas' with an id of 1, and the second is for 'Manzanas' with an id of 2. The interface includes a search bar, 'ADD DATA' and 'EXPORT COLLECTION' buttons, and pagination controls.

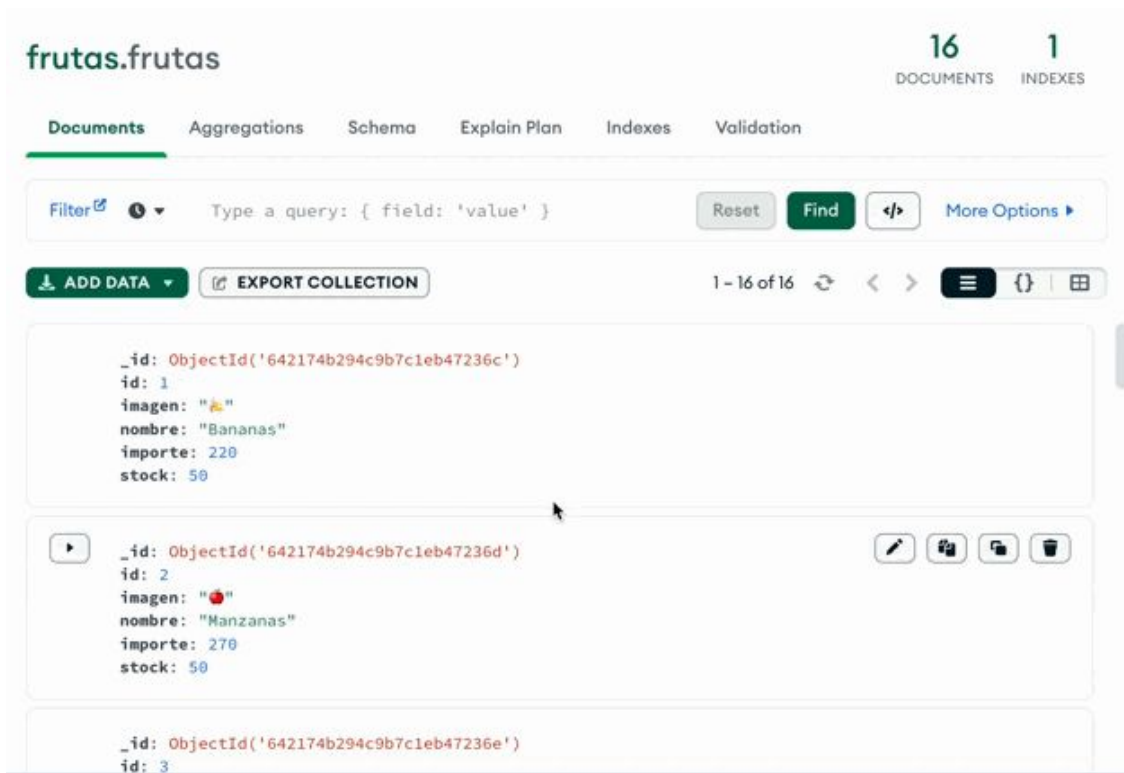
```
{ "_id": ObjectId('642174b294c9b7c1eb47236c'),  
  "id": 1,  
  "imagen": "🍌",  
  "nombre": "Bananas",  
  "importe": 220,  
  "stock": 50 }
```

```
{ "_id": ObjectId('642174b294c9b7c1eb47236d'),  
  "id": 2,  
  "imagen": "🍏",  
  "nombre": "Manzanas",  
  "importe": 270,  
  "stock": 50 }
```

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 en MongoDB



frutas.frutas

16 DOCUMENTS 1 INDEXES

Documents Aggregations Schema Explain Plan Indexes Validation

Filter ⓘ ⓘ Type a query: { field: 'value' } Reset Find </> More Options ▶

ADD DATA EXPORT COLLECTION

1 - 16 of 16

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

```
{
  "_id": ObjectId('642174b294c9b7c1eb47236d'),
  "id": 2,
  "imagen": "🍏",
  "nombre": "Manzanas",
  "importe": 270,
  "stock": 50
}
```

```
{
  "_id": ObjectId('642174b294c9b7c1eb47236e'),
  "id": 3
}
```

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**

**UNTREF**

UNIVERSIDAD NACIONAL  
DE TRES DE FEBRERO

# Estructura de datos en MongoDB

```
_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 en MongoDB

```
_id: ObjectId('642174b294c9b7c1eb47236c')
```

```
id: 1
```

```
imagen: "🍌"
```

```
nombre: "Bananas"
```

```
importe: 220
```

```
stock: 50
```

El 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 en MongoDB

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

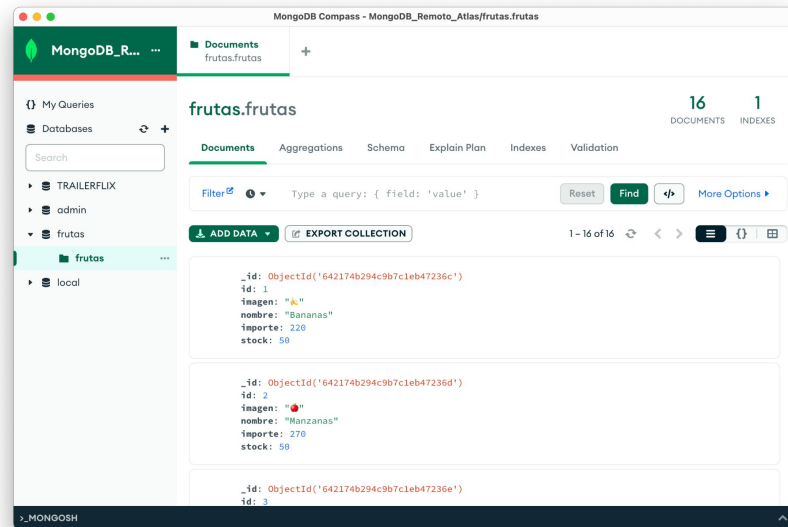
- Garantiza que cada documento tenga un identificador único seguro en una colección
- Ordena los documentos en una colección en función del orden en que se insertaron
- Proporciona un indicador visual de la fecha y hora en que se creó un documento
- Se utiliza también en algunas consultas para buscar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

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

# Estructura de datos en MongoDB

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



# Espacio de trabajo

# Espacio de trabajo

**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 anteriormente en otras Unidades, y ábrelo.
- En el proceso de importación, selecciona la opción **Import as JSON**.
- Espera a que finalice la importación de datos.
  - Luego de esto, navega por la colección que acabas de crear para verificar su correcta importación.

**Tiempo estimado: 20 minutos.** 

**UNTREF**

UNIVERSIDAD NACIONAL  
DE TRES DE FEBRERO

```
const questions = [ 'dudas', 'consultas', '🤔' ]
```



```
> node gracias.js
```