



## Clase 17. Programación Backend

***MongoDB***



## ***OBJETIVOS DE LA CLASE***

- Comprender la arquitectura y el modo de funcionamiento de MongoDB.
- Instalar el servidor local de base de datos y ponerlo en marcha.
- Utilizar clientes CLI y GUI para interactuar con MongoDB.

# ***CRONOGRAMA DEL CURSO***

Clase 16



**SQL y Node.js**

Clase 17



**MongoDB**

Clase 18



**CRUD en MongoDB**

***Repaso luego del break...***

***MongoDB***



mongoDB

**CODER HOUSE**

# *¿Qué es MongoDB?*



MongoDB es una **base de datos No relacional, NoSQL**, orientada a documentos que ofrece una gran escalabilidad y flexibilidad, y un modelo de consultas e indexación avanzado.

# ***Modelo de documentos de MongoDB***



El **modelo de documentos** de MongoDB resulta muy fácil de aprender y usar, y proporciona a los desarrolladores todas las funcionalidades que necesitan para satisfacer los requisitos más complejos a cualquier escala.



# MongoDB: Opciones de implementación



MongoDB dispone de dos variantes de despliegue:

- **Local:** con *Mongo Server*, a través de sus opciones Community y Enterprise.
- **Remota:** mediante una plataforma configurada en la nube, lista para usar, llamada *Mongo Atlas*.

## Cloud

Pruebe gratis MongoDB Atlas

### Sandbox

- Free, forever
- Ideal for learning, developing, and prototyping

### Shared

- Up to 5GB storage
- Shared RAM

### Dedicated

- Consistent performance
- Advanced security
- Unlimited scaling

## Server

Descargar MongoDB

MongoDB offers both an Enterprise and Community version of its powerful distributed document database

### Community

- Feature rich
- Developer ready

### Enterprise

- Advanced features
- Performance grade





# ***MongoDB: características***



- Almacena datos en **documentos** flexibles similares a **JSON**: la estructura de datos puede cambiar con el tiempo.
- **El modelo** de documento se asigna a los objetos en el código de su aplicación para facilitar el trabajo con los datos.
- Las consultas ad hoc, la indexación y la agregación en tiempo real ofrecen maneras potentes de acceder a los datos y analizarlos.
- MongoDB es una base de datos distribuida en su núcleo, por lo que la **alta disponibilidad**, la **escalabilidad** horizontal y la distribución geográfica están integradas y son fáciles de usar.
- MongoDB es de **uso gratuito**.



# ***MongoDB: introducción***



- El concepto de bases de datos NoSQL va creciendo y se utiliza con más frecuencia. **MongoDB la base de datos NoSQL** más conocida.
- El concepto **NoSQL** define sistemas que difieren del modelo clásico SQL: *Sistema de bases de datos relacionales*. Lo más destacado de NoSQL es que **no usan SQL como lenguaje principal** de consultas.
- **MongoDB** es una base de datos **orientada a documentos**. No se **SE** basa en el concepto de Tabla Fila y Registro sino que se apoya en



# ***MongoDB: Arquitectura***



Servidor MongoDB

Database 1

Colección 1

Documentos

Colección 2

Documentos

Database 2

Colección 3

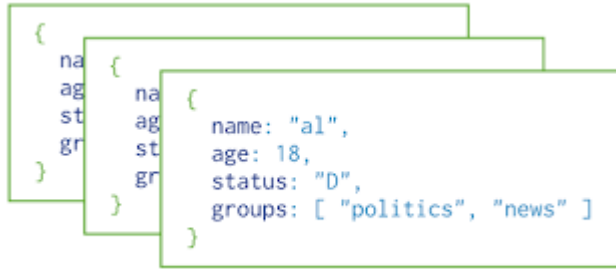
Documentos

Colección 4

Documentos

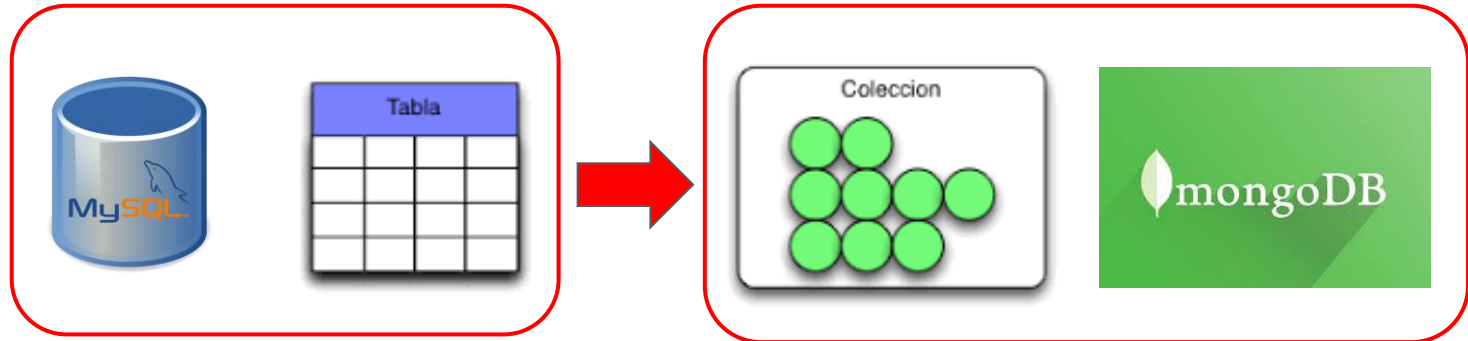


# ***MongoDB y Colecciones***



Collection

Una **colección** en MongoDB es **muy similar** a una **tabla** de una base de datos. La tabla almacena registros (filas) mientras que las **colecciones almacenan documentos**.

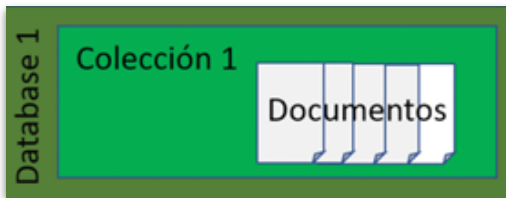


```
1 {  
2   "_id" :  
3   ↪ ObjectId("58a78d740b0fa18b936f1d9a"),  
4   "movieId" : 1,  
5   "title" : "Toy Story (1995)",  
6   "genres" : "Adventure|Animation|  
   ↪ Children|Comedy|Fantasy"  
}
```

# ***MongoDB y Documentos***



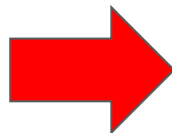
- Aquí comienzan las diferencias importantes entre una base de datos SQL y una NoSQL. El **concepto** de **fila** y de **documentos** son **bastante diferentes**.
- Una fila está compuesta de columnas y siempre son las mismas para todas ellas.
- En cambio un **documento** está **compuesto por claves y valores** (key,value) y cada documento puede tener **variaciones importantes** con respecto al anterior dentro de una colección.



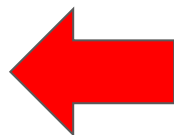
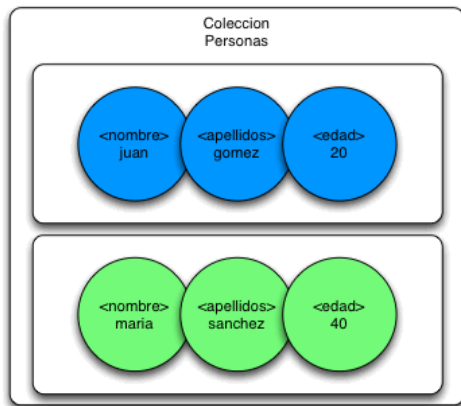
# ***MongoDB y Documentos***



Imaginémonos que tenemos el concepto de **Persona: nombre,apellidos,edad.**  
En una **base de datos relacional** dispondremos de la siguiente tabla



nombre	apellidos	edad
juan	gomez	20
maria	sanchez	40



Ahora tenemos el mismo concepto de **Persona: nombre,apellidos,edad.**  
En una **base de datos MongoDB** dispondremos de la siguiente tabla



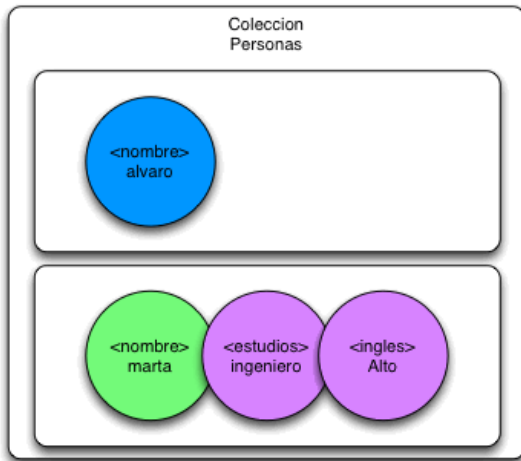
{ name: mongo, type: DB }

# ***MongoDB y Documentos***



En **MongoDB** se podrían **crear documentos muy diferentes** como estos:

- El primer documento solo dispone de un campo.
- El segundo dispone de tres, pero diferentes a los anteriores.
- Esto es lo que en NoSQL se denomina **Schema Free** o **libre** y que aporta una gran flexibilidad a la hora de trabajar.



Esta tipología de esquema viene reforzada por la capacidad de generar documentos embebidos dentro de la misma colección.



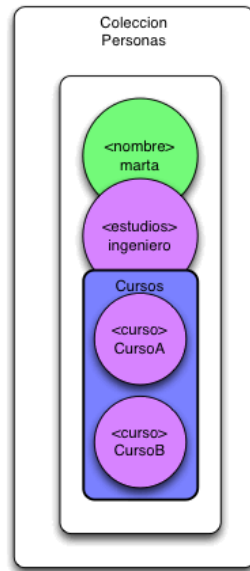
# ***MongoDB***

## ***Documentos Embebidos***



Un **documento embebido** es un documento que está **insertado dentro de otro** y que ambos están ligados a la **misma colección**.

*De esta manera las bases de datos orientadas a documentos aportan una gran flexibilidad a la hora de estructurar la información.*







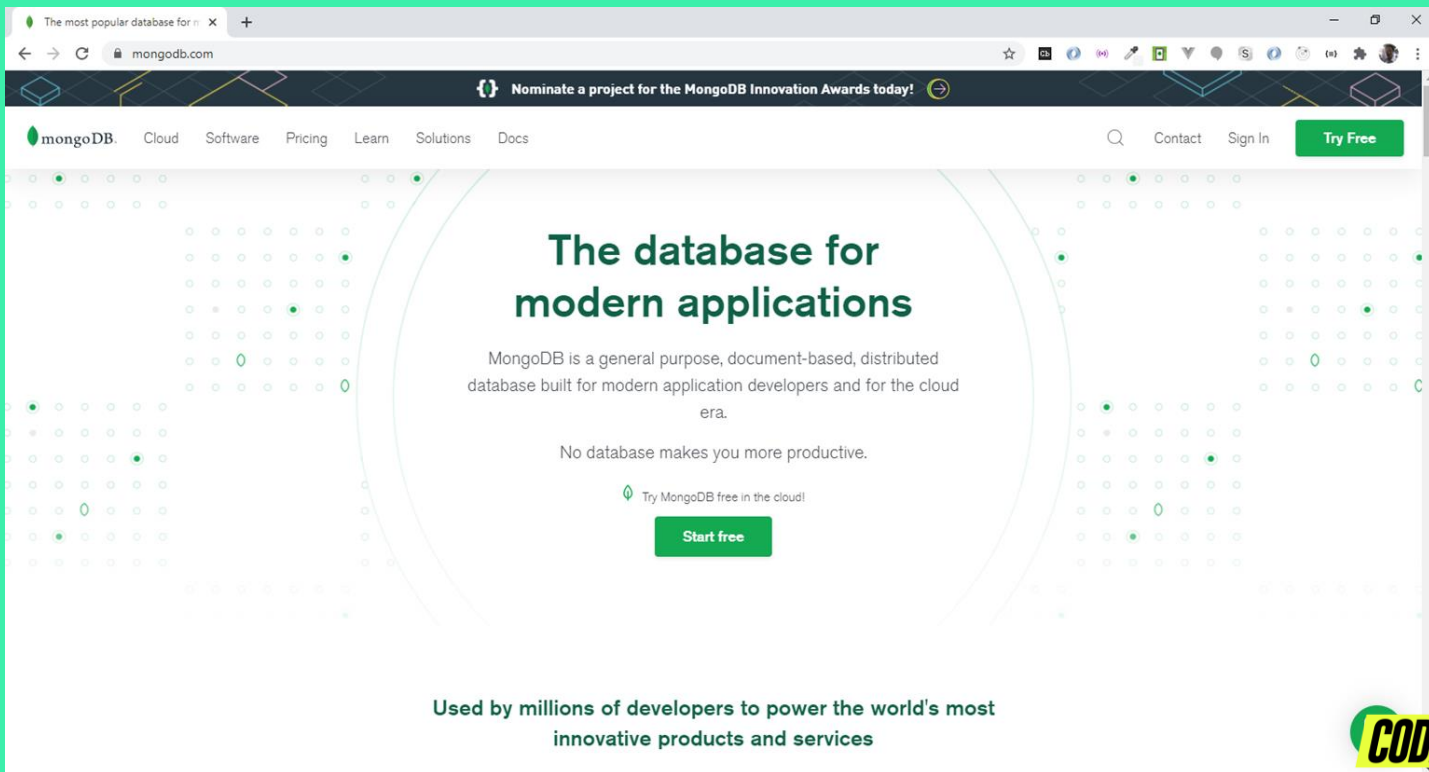
***¿Alguna pregunta hasta  
ahora?***

# ***Instalación y configuración de MongoDB***

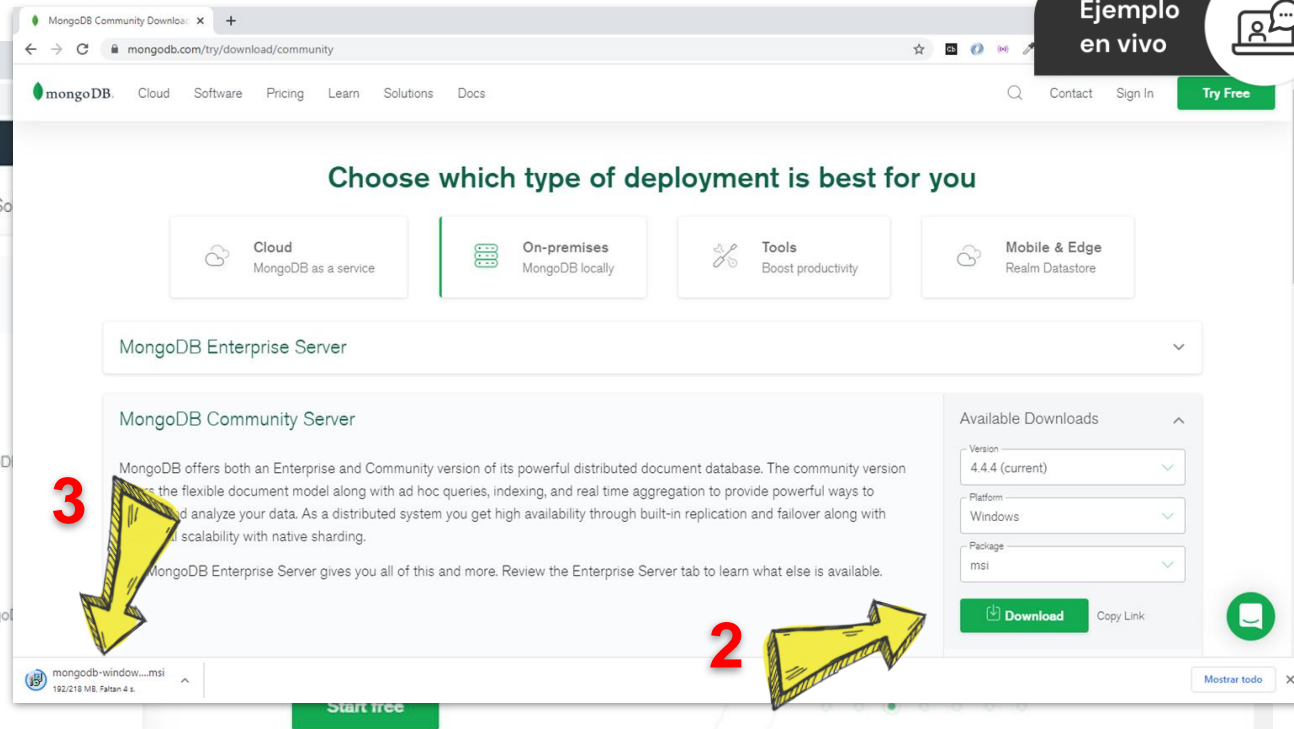
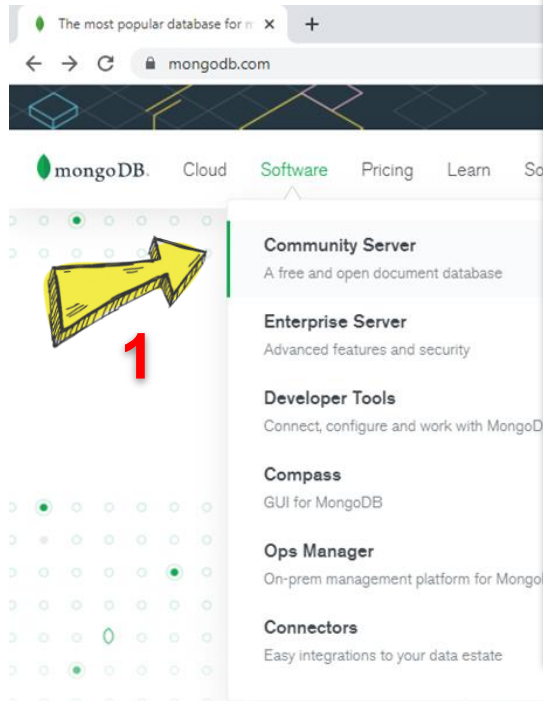


# ***MongoDB: Web oficial***

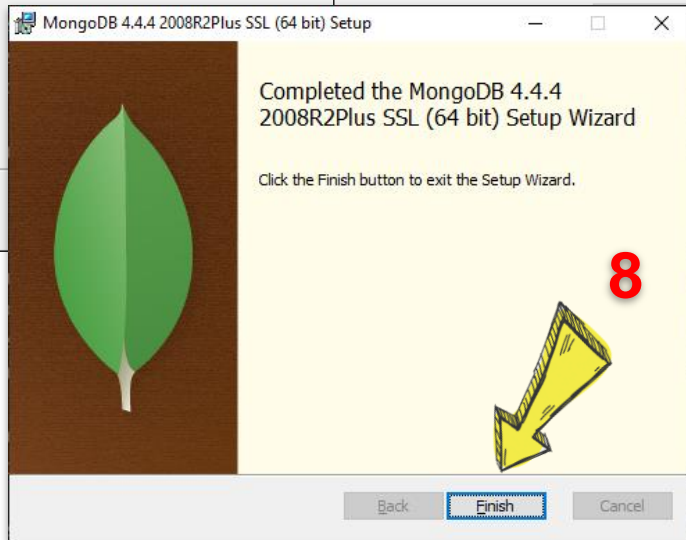
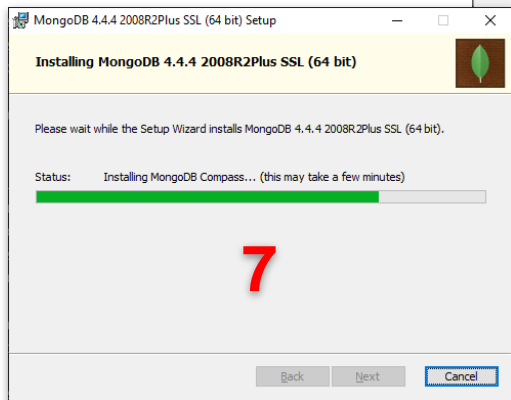
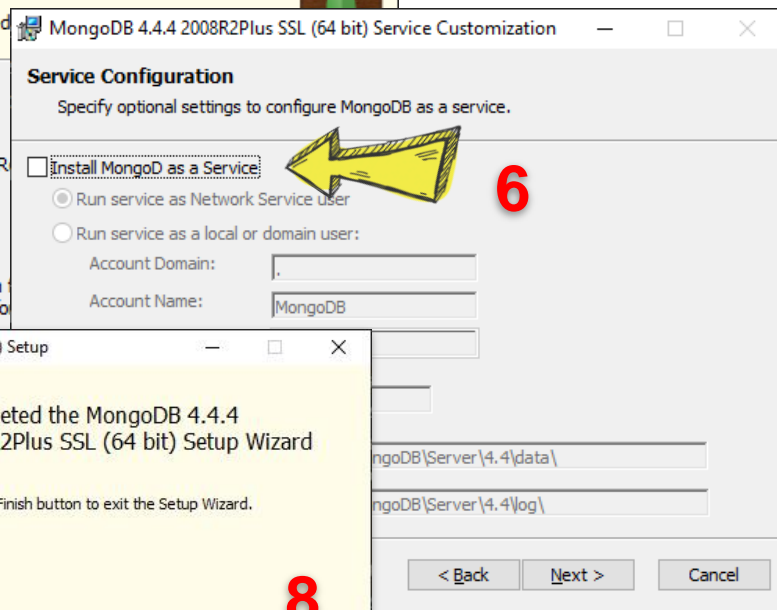
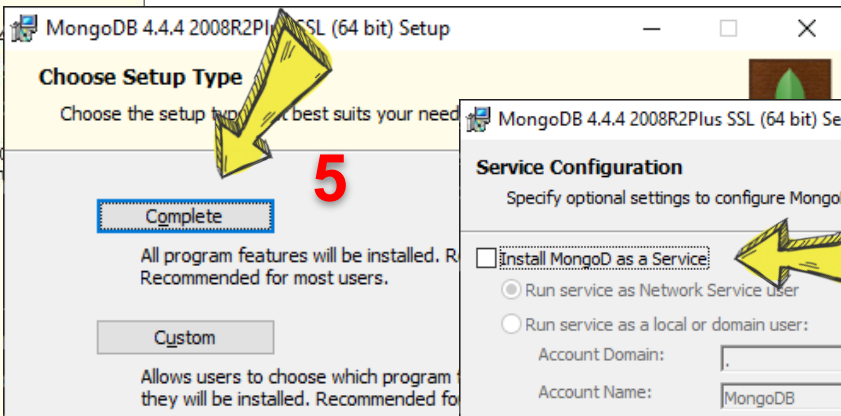
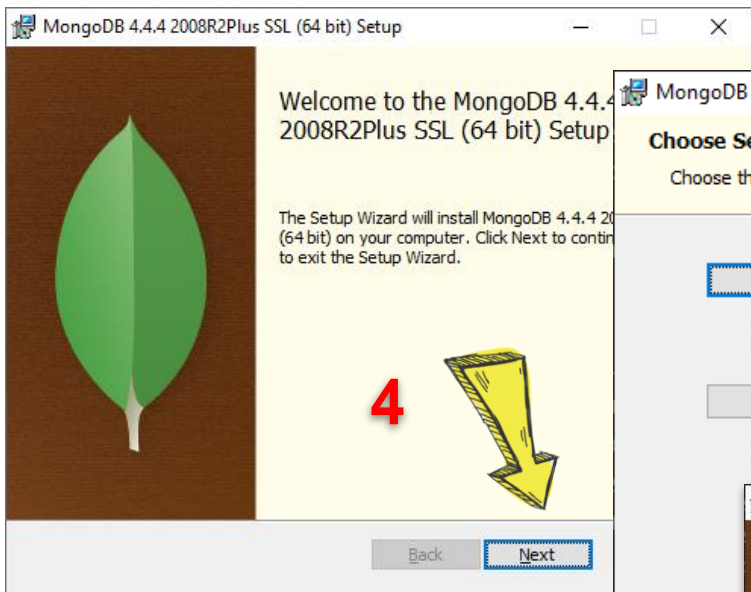
## ***<https://www.mongodb.com/>***

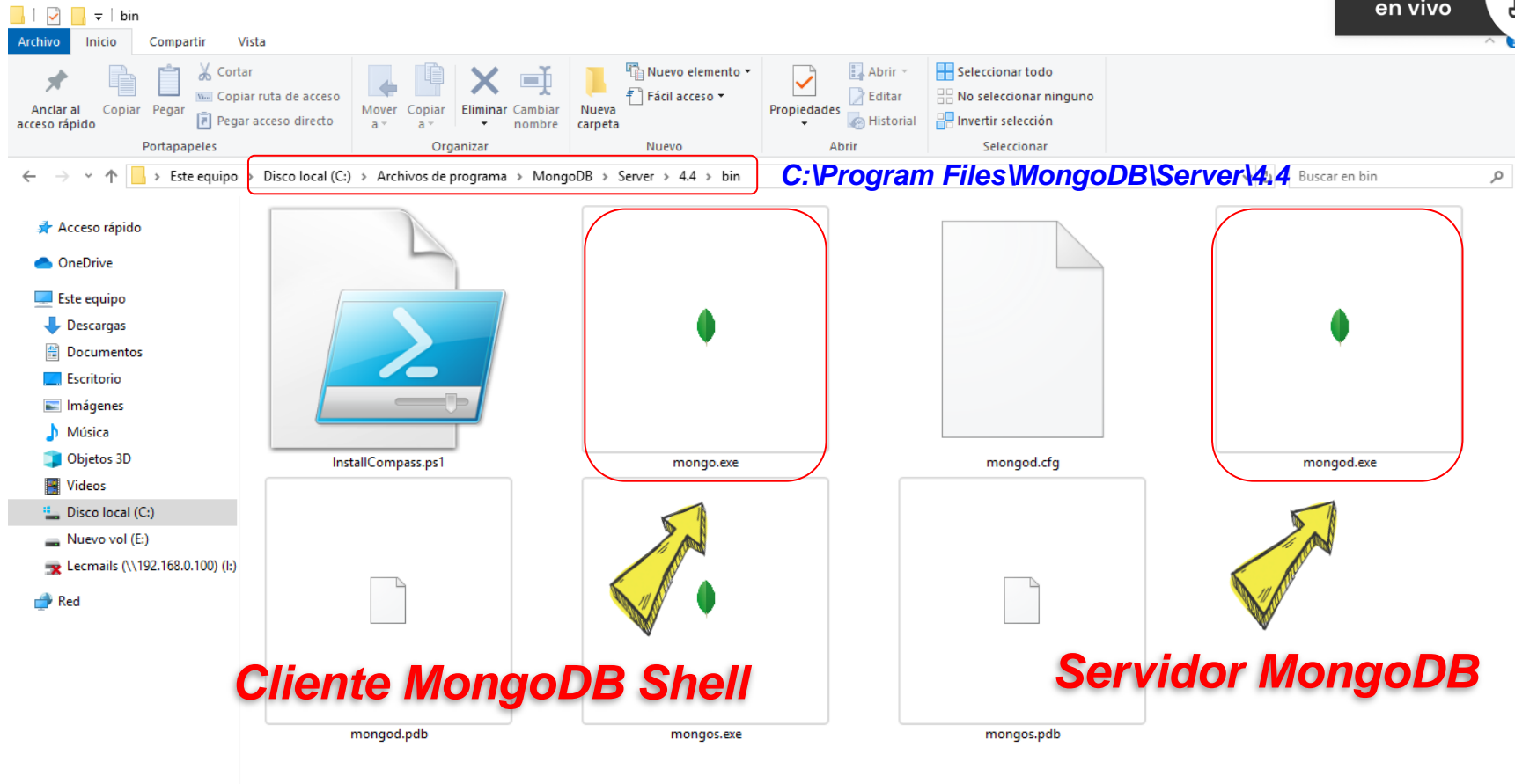


***Instalación MongoDB desde***  
***<https://www.mongodb.com/>***



Used by millions of developers to power the world's most  
innovative products and services





# ***Proceso de configuración del path del sistema operativo hacia la carpeta de los binarios de MongoDB***

***C:\Program Files\MongoDB\Server\4.4***





## Propiedades del sistema

Nombre del equipo Hardware

Opciones avanzadas Protección del sistema Remoto

Para realizar la mayoría de estos cambios, inicie sesión como administrador.

Rendimiento

Efectos visuales, programación del procesador, uso de memoria y memoria virtual

Configuración...

Perfiles de usuario

Configuración del escritorio correspondiente al inicio de sesión

Configuración...

Inicio y recuperación

Inicio del sistema, errores del sistema e información de depuración

Configuración...

Variables de entorno...

Aceptar Cancelar Aplicar

## Variables de entorno

## Variables de usuario para EducacionIT

Variable	Valor
ChocolateyLastPathUpdate	132492371894918580
OneDrive	C:\Users\EducacionIT\OneDrive
OneDriveConsumer	C:\Users\EducacionIT\OneDrive
Path	C:\Users\EducacionIT\AppData\Lo
TEMP	C:\Users\EducacionIT\AppData\Lo
TMP	C:\Users\EducacionIT\AppData\Lo

Nuevo...

## Variables del sistema

Variable	Valor
Path	C:\Python39\Scripts\;C:\Python39\
PATHEXT	.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS
PROCESSOR_ARCHITECTURE	AMD64
PROCESSOR_IDENTIFIER	Intel64 Family 6 Model 142 Steppin
PROCESSOR_LEVEL	6
PROCESSOR_REVISION	8e0c
PSModulePath	%ProgramFiles%\WindowsPowerShell\Modules;C:\Windows\svste...

Nueva...

## Editar variable de entorno

C:\Python39\Scripts\  
C:\Python39\  
%SystemRoot%\system32  
%SystemRoot%  
%SystemRoot%\System32\Wbem  
%SYSTEMROOT%\System32\WindowsPowerShell\v1.0\  
%SYSTEMROOT%\System32\OpenSSH\  
C:\Program Files\Git\cmd  
C:\xampp\php  
C:\ProgramData\ComposerSetup\bin  
C:\Program Files\Java\jdk-13.0.2\bin  
C:\Program Files\dotnet\  
C:\Program Files\Microsoft SQL Server\130\Tools\Binn\  
C:\Program Files\Microsoft SQL Server\Client SDK\ODBC\170\Tools\Bi...  
C:\Program Files (x86)\GitExtensions\  
C:\Program Files\nodejs\  
C:\ProgramData\chocolatey\bin  
C:\Program Files\MongoDB\Server\4.4\bin

Nuevo

Editar

Examinar...

Eliminar

Subir

Bajar

Editar texto...

Aceptar

Cancelar

1

2

3

4

5

Buscar 'editar variables de entorno' y ejecutar



# ***MongoDB Server y MongoDB Client CLI***

## ***Pasos para la puesta en marcha***



1. Creamos una carpeta llamada -por ejemplo- *miBaseMongo*
2. Ejecutamos en la consola el siguiente comando:  
**mongod --dbpath "ruta/hacia/la/carpeta/miBaseMongo"**
1. Verificamos que el motor de base de datos se inicie. Este escuchará peticiones de clientes en **localhost puerto 27017**.
2. Abrimos otra consola y ejecutamos el comando: **mongo**
3. En este caso se abre el cliente que se conectará al servicio de base de datos iniciado anteriormente.

***Ambas consolas quedarán bloqueadas ejecutando sus procesos.***

*Si queremos salir, lo hacemos con **Control-C**.*

# MongoDB: Procesos en la consola



```
C:\cmdr_mini
λ cd C:\Cursos\Coderhouse\CursoBackend\Clase18

C:\Cursos\Coderhouse\CursoBackend\Clase18
λ mkdir miBaseMongo

C:\Cursos\Coderhouse\CursoBackend\Clase18
λ mongod --dbpath "C:\Cursos\Coderhouse\CursoBackend\Clase18\miBaseMongo"
{"t":{"$date":"2021-02-25T13:05:30.888-03:00"},"s":"I", "c":"CONTROL", "id":23285, "ctx":"main","msg":"Automatically disabling TLS
1.0, to force-enable TLS 1.0 specify --sslDisabledProtocols 'none'"}
{"t":{"$date":"2021-02-25T13:05:31.483-03:00"},"s":"W", "c":"ASIO", "id":22601, "ctx":"main","msg":"No TransportLayer configured
during NetworkInterface startup"}
{"t":{"$date":"2021-02-25T13:05:31.485-03:00"},"s":"I", "c":"NETWORK", "id":4648602, "ctx":"main","msg":"Implicit TCP FastOpen in use
.."}
{"t":{"$date":"2021-02-25T13:05:31.492-03:00"},"s":"I", "c":"STORAGE", "id":4615611, "ctx":"initandlisten","msg":"MongoDB starting,"

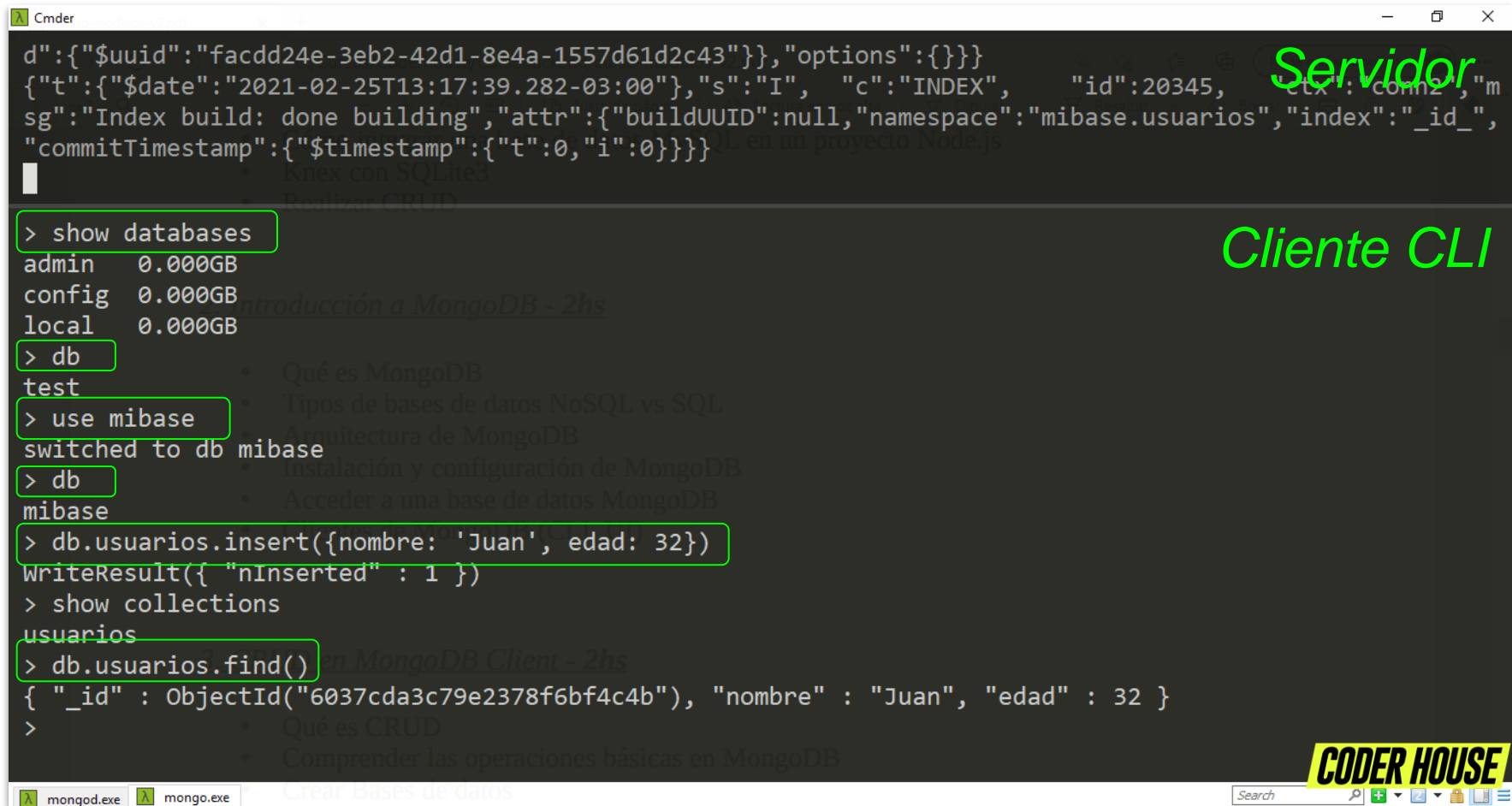
C:\cmdr_mini
λ mongo
MongoDB shell version v4.4.4
connecting to: mongodb://127.0.0.1:27017/?compressors=disabled&gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("25a0a4bb-b0da-493a-bf72-8d09171adb37") }
MongoDB server version: 4.4.4
---
The server generated these startup warnings when booting:
 2021-02-25T13:05:31.705-03:00: Access control is not enabled for the database. Read and write access to data and configuration
is unrestricted
 2021-02-25T13:05:31.705-03:00: This server is bound to localhost. Remote systems will be unable to connect to this server. Star
t the server with --bind_ip <address> to specify which IP addresses it should serve responses from, or with --bind_ip
```

**Servidor**

**Cliente CLI**

**CODER HOUSE**

# MongoDB: Prueba del cliente CLI -> primeras acciones



```
d":{"$uuid":"facdd24e-3eb2-42d1-8e4a-1557d61d2c43"},"options":{}}}  
{"t":{"$date":"2021-02-25T13:17:39.282-03:00"},"s":"I", "c":"INDEX", "id":20345, "ctx":"conn2","m  
sg":"Index build: done building","attr":{"buildUUID":null,"namespace":"mibase.usuarios","index":"_id_","  
"commitTimestamp":{"$timestamp":{"t":0,"i":0}}}}  
  
> show databases  
admin 0.000GB  
config 0.000GB  
local 0.000GB  
  
> db  
test  
  
> use mibase  
switched to db mibase  
  
> db  
mibase  
  
> db.usuarios.insert({nombre: 'Juan', edad: 32})  
WriteResult({ "nInserted" : 1 })  
  
> show collections  
usuarios  
  
> db.usuarios.find()  
{ "_id" : ObjectId("6037cda3c79e2378f6bf4c4b"), "nombre" : "Juan", "edad" : 32 }  
>
```

**Servidor**

**Cliente CLI**

**CODER HOUSE**

```
C:\>λ mongo
MongoDB shell version v4.4.4
connecting to: mongodb://127.0.0.1:27017
Implicit session: session { "id" : UUID(
MongoDB server version: 4.4.4
```

# ***MongoDB: ejemplos de Comandos en consola CLI***



Algunos comandos para iniciar la operación con la base de datos:

- **mongo:** nos conectará de forma automática con el servidor.
- **show dbs:** muestra la lista de bases de datos existentes.
- **use:** selecciona la base de datos activa (si no existe la crea):  
Ej. **use mibase**
- **show collections:** muestra la lista de colecciones de esta base
- **db.personas.insertOne({clave: valor, ...}):** crea una colección en forma implícita llamada 'personas' y le inserta un documento.
- **db.personas.find():** busca todos los documentos que están dentro de la colección 'personas'

# MongoDB Comandos desde consola CLI

Ejemplo  
en vivo



Cmdr

```
> show dbs
admin            0.000GB
config           0.000GB
dbvideoclub     0.000GB
local            0.000GB

> use mibase
switched to db mibase

> show collections

> db.personas.insert({nombre:'Juan', edad:23})
WriteResult({ "nInserted" : 1 })

> db.personas.find()
{ "_id" : ObjectId("6038cb664c7c3430bde5db94"), "nombre" : "Juan", "edad" : 23 }

> show collections
personas

> show dbs
admin            0.000GB
config           0.000GB
dbvideoclub     0.000GB
local            0.000GB
mibase           0.000GB

>
```

Integrate MongoDB with free cloud-based monitoring service, which will then receive and display metrics about your deployment (disk utilization, CPU, operation statistics, etc).

The MongoDB Cloud Platform is available on a MongoDB website with a unique URL available to you. You can use this URL to share the MongoDB Cloud Platform information to make product

mongod.exe

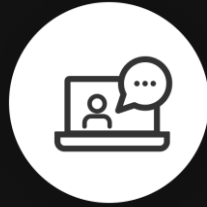
mongo.exe

notes

1366 x 738 photos

Search

**CODER HOUSE**



***¿Alguna pregunta hasta  
ahora?***



# ***NUESTRA PRIMERA BASE DE DATOS EN MONGODB***

*Tiempo: 15 minutos*





Realizar las siguientes acciones utilizando un servidor de base de datos MongoDB:

- 1) Crear una carpeta llamada 'base' para contener la estructura de base de datos.
- 2) Ejecutar en una consola el motor de base de datos MongoDB, que debe seleccionar la carpeta 'base' como almacenamiento de su estructura.
- 3) Iniciar [Mongo shell](#) en otra consola, verificando la conexión con el servidor.
- 4) Listar las bases de datos disponibles
- 5) Crear una base de datos en MongoDB llamada 'sistema' y seleccionarla con base de trabajo.
- 6) Insertar un documento en la colección usuarios que contenga los campos: nombre y edad.



- 7) Listar los documentos dentro de la colección usuarios.
- 8) Mostrar las colecciones de la base 'sistema'.
- 9) Insertar tres documentos más con el mismo formato y valores distintos en la colección usuarios.
- 10) Listarlos nuevamente verificando que haya 4 documentos dentro de la colección usuarios.
- 11) Insertar dentro de una colección llamada 'productos' cuatro documentos con los siguientes campos: nombre, precio y stock.
- 12) Mostrar las colecciones de la base 'sistema'.
- 13) Listar todos los documentos de la colección 'productos'.



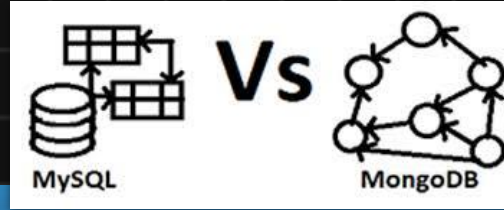
***¿Alguna pregunta hasta  
ahora?***



***BREAK***

**¡5/10 MINUTOS Y VOLVEMOS!**

# MySQL vs MongoDB



MongoDB



MySQL





# ***Equivalencias entre los elementos de MongoDB y SQL DB***

MongoDB	&	SQL DB
Servidor	=	Servidor
Base de Datos	=	Base de Datos
Colección	≈	Tabla
Documento	≈	Fila
Campo	≈	Columna



# ***SQL vs NoSQL: Conceptos***



- Al trabajar con SQL y bases de datos relacionales, el sistema de almacenamiento de datos de nuestras aplicaciones web sigue patrones que otorgan relación entre las tablas de nuestra base de datos con las claves primaria y foráneas.
- Esto es altamente beneficioso porque nos genera un **sistema de almacenamiento de datos totalmente estructurado**.
- **Al evolucionar** nuestras aplicaciones web, **debemos realizar modificaciones** en nuestro modelo conceptual de la base de datos.
- Este problema con las bases de datos **NoSQL** quedaría resuelto por la **flexibilidad** que nos ofrecen las bases de datos documentales, ya que estas bases de datos trabajan con datos en formato JSON.

# MySQL vs MongoDB Esquemas



## mySQL

```
SELECT
  Dim1, Dim2,
  SUM(Measure1) AS MSum,
  COUNT(*) AS RecordCount,
  AVG(Measure2) AS MAvg,
  MIN(Measure1) AS MMin,
  MAX(CASE
    WHEN Measure2 < 100
    THEN Measure2
  END) AS MMax
FROM DenormAggTable
WHERE (Filter1 IN ('A','B'))
AND (Filter2 = 'C')
AND (Filter3 > 123)
GROUP BY Dim1, Dim2
HAVING (MMin > 0)
ORDER BY RecordCount DESC
LIMIT 4, 8
```

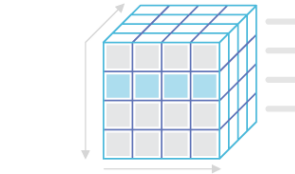
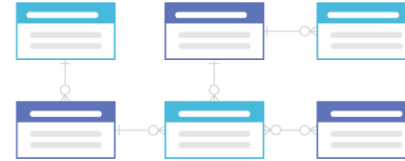
- ① Grouped dimension columns are pulled out as keys in the map function, reducing the size of the working set.
- ② Measures must be manually aggregated.
- ③ Aggregates depending on record counts must wait until finalization.
- ④ Measures can use procedural logic.
- ⑤ Filters have an ORM/ActiveRecord-looking style.
- ⑥ Aggregate filtering must be applied to the result set, not in the map/reduce.
- ⑦ Ascending 1; Descending -1

## MongoDB

```
db.runCommand({
  mapreduce: "DenormAggCollection",
  query: {
    filter1: { '$in': [ 'A', 'B' ] },
    filter2: 'C',
    filter3: { '$gt': 123 }
  },
  map: function() { emit(
    { d1: this.Dim1, d2: this.Dim2 },
    { msum: this.measure1, recs: 1, mmin: this.measure1,
      mmax: this.measure2 < 100 ? this.measure2 : 0 }
  ); },
  reduce: function(key, vals) {
    var ret = { msum: 0, recs: 0, mmin: 0, mmax: 0 };
    for(var i = 0; i < vals.length; i++) {
      ret.msum += vals[i].msum;
      ret.recs += vals[i].recs;
      if(vals[i].mmin < ret.mmin) ret.mmin = vals[i].mmin;
      if((vals[i].mmax < 100) && (vals[i].mmax > ret.mmax))
        ret.mmax = vals[i].mmax;
    }
    return ret;
  },
  finalize: function(key, val) {
    val.mavg = val.msum / val.recs;
    return val;
  },
  out: 'result1',
  verbose: true
});
db.result1.
  find({ mmin: { '$gt': 0 } }).
  sort({ recs: -1 }).
  skip(4).
  limit(8);
```

### SQL

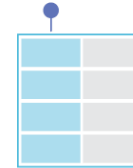
Relational Database Management Systems (RDBMS)



Online Analytical Processing (OLAP) Cube

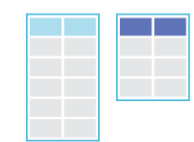
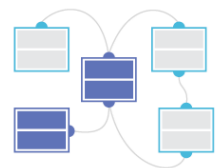
### NoSQL

Key-Value



Document

Graph



Column store

**CODER HOUSE**

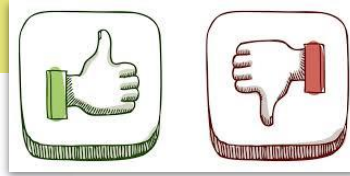
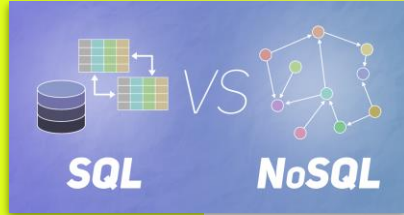


# MySQL vs MongoDB: comparaciones

Característica	MongoDB	MySQL
Cloud, SaaS, Web	si	si
Desarrolladores	MongoDB Inc.	Oracle Corporation
SO	Multiplataforma	Multiplataforma
Lenguaje query	Javascript	SQL
Mapa reducido	si	no
Convercion de DB	si	no
Analisis de performance	si	no
Virtualización	si	no
Modelo de integridad	BASE	ACID
Atomicidad	condicional	si
Aislamiento	no	si
Transacciones	no	si
Integridad referencial	no	si
CAP	CP	CA
Escalabilidad horizontal	si	condicional
Modo de replicación	Maestro-Eslavo	Maestro - Maestro/Eslavo

MongoDB	MySQL
<b>ALTA DISPONIBILIDAD</b> Ideal si necesitas una alta disponibilidad de los datos con recuperacion automatica, rapida e instantanea	<b>BAJO MANTENIMIENTO</b> Si recién comienzas y no crees que creceras muchos MySQL te ayudara con su facil y bajo mantenimiento.
<b>FRAGMENTACIÓN</b> Si piensas que creceras mucho en el futuro te conviene MongoDB ya que tienen incorporada la fragmentación	<b>PRESUPUESTO LIMITADO</b> Ideal si necesitas alta performance con un presupuesto limitado.
<b>ESQUEMA INESTABLE</b> Ideal si tienes un esquema inestable y cambiante para evitar cambiar el esquema repetidamente	<b>ESQUEMA FIJO</b> Ideal si tu esquema no cambia y siempre guardas los mismos tipos de datos. Por ejemplo un blog.
<b>SIN ADMINISTRADOR DE BD</b> Si no quieres usar un administrador de base de datos. Si creces demasiado deberas utilizar uno de todas maneras.	<b>TRANSACCIONES ALTAS</b> Si requieres de una alta tasa de transacciones
<b>EN LA NUBE</b> Si todos tus servicios estan basados en la nube. MongoDB te resultara mas conveniente.	<b>SEGURIDAD DE LOS DATOS</b> Si la seguridad de los datos es tu prioridad, MySQL es el manejador de bases de datos más seguro.

# ***Ventajas y Desventajas de MySQL y MongoDB***



PROS



CONS





# ***MySQL: ventajas***



- Podemos **ejecutar sentencias SQL** directamente en nuestra base de datos.
- Posibilidad de **abstracción** de nuestra base de datos con algún ORM estilo Doctrine o Hibernate.
- Almacenamiento de datos totalmente **organizado** y **jerarquizado** con claves primarias y foráneas.
- Nos permite **evitar** la **duplicidad** de registros.
- Mejora notable en **mantenimiento** de datos en relación a la **seguridad** requerida de los mismos.



# ***MySQL: desventajas***



- Si nuestro sistema **escala** y evoluciona, tendremos que haber diseñado nuestra base de datos según los posibles nuevos requerimientos.
- Requiere **más espacio** de almacenamiento que una base NoSQL.
- Las **transacciones** de datos son más **pesadas** frente a las bases de datos NoSQL.
- Los **límites en los campos** de las tablas nos pueden hacer perder datos si no los configuramos adecuadamente según el tamaño del dato que nos puedan introducir los usuarios.



# ***MongoDB: ventajas***



- La **escalabilidad** y su **carácter descentralizado** hacen que soporte estructuras distribuidas.
- Permiten realizar **sistemas más abiertos y flexibles** debido a su fácil adaptación de nuevas evoluciones de nuestras aplicaciones web.
- **No se requieren potentes recursos** para poder trabajar con bases de datos NoSQL.
- **Optimización** de las **consultas** en base de datos para grandes cantidades de datos almacenados.



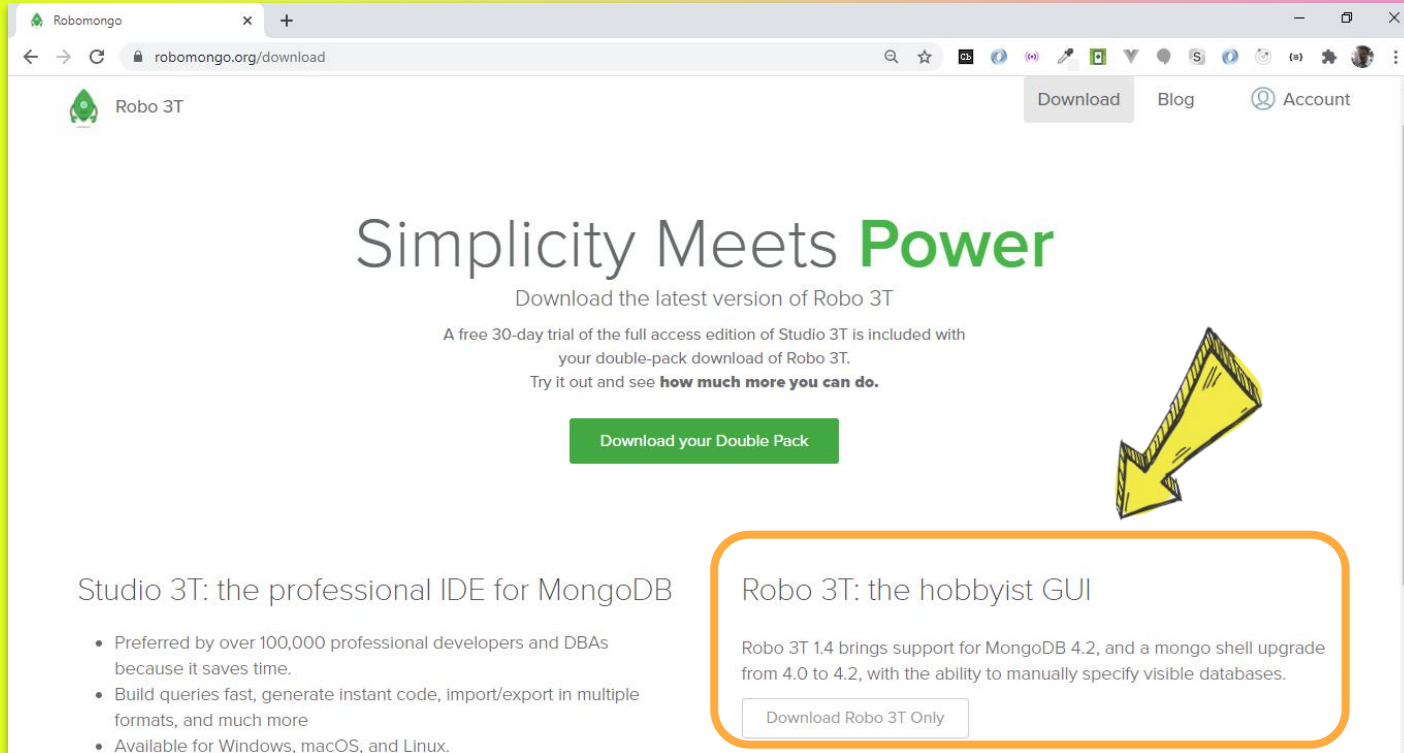
# ***MongoDB: desventajas***



- **Problemas con sentencias SQL** ya que no admiten el 100% de las consultas existentes.
- La principal desventaja de MongoDB es que **carece** de algo tan fundamental como los **Joins**.
- **Falta de estandarización** entre las diferentes bases de datos NoSQL.

# *Instalación de Cliente GUI Robo 3T para MongoDB*

## *Website: <https://robomongo.org/download>*



The screenshot shows the Robomongo website's download page. The browser address bar displays 'robomongo.org/download'. The page features a navigation bar with 'Download', 'Blog', and 'Account' links. The main heading is 'Simplicity Meets Power', with 'Power' in green. Below it, text states: 'Download the latest version of Robo 3T. A free 30-day trial of the full access edition of Studio 3T is included with your double-pack download of Robo 3T. Try it out and see how much more you can do.' A large green button labeled 'Download your Double Pack' is centered. To the right, a large yellow hand-drawn arrow points downwards. At the bottom, two sections are visible: 'Studio 3T: the professional IDE for MongoDB' with a list of features, and 'Robo 3T: the hobbyist GUI' with a description and a 'Download Robo 3T Only' button. The 'Robo 3T' section is highlighted with an orange rounded rectangle.

Robomongo

robomongo.org/download

Robo 3T

Download Blog Account

## Simplicity Meets Power

Download the latest version of Robo 3T

A free 30-day trial of the full access edition of Studio 3T is included with your double-pack download of Robo 3T. Try it out and see **how much more you can do.**

[Download your Double Pack](#)

### Studio 3T: the professional IDE for MongoDB

- Preferred by over 100,000 professional developers and DBAs because it saves time.
- Build queries fast, generate instant code, import/export in multiple formats, and much more
- Available for Windows, macOS, and Linux.

### Robo 3T: the hobbyist GUI

Robo 3T 1.4 brings support for MongoDB 4.2, and a mongo shell upgrade from 4.0 to 4.2, with the ability to manually specify visible databases.

[Download Robo 3T Only](#)

**CODER HOUSE**

# Conexión de Cliente GUI Robo 3T a MongoDB

Ejemplo  
en vivo



Robo 3T - 1.3

File View Options Window Help

New release available. Find out [what's new in Robo 3T](#) - [Download here](#).

Welcome x

**Robo 3T 1.3 now with MongoDB 4.2 support**

MongoDB Connections

Create, edit, remove, clone or reorder connections via drag'n'drop.

Name	Address	Attributes	Auth. Database / User
Mi conexión local MongoDB	localhost:27017		

Connect Cancel

Connection Settings

Connection Authentication SSH SSL Advanced

Type: Direct Connection

Name: Mi conexión local MongoDB

Address: localhost : 27017

Specify host and port of MongoDB server. Host can be either IPv4, IPv6 or domain name.

From SRV Import connection details from MongoDB SRV connection string

Test Save Cancel

Robo 3T

Jun 2017  
Mongo 1.0 — Official Release  
Apr 2017  
Robo 3T has been acquired by 3T  
Mar 2017  
Change the name of Robomongo  
Feb 2017  
Robomongo 1.0 RC1 brings support to Replica Set Clusters  
Thu, 02 Feb 2017

All Blog Posts

Logs

**1**

**2**

**3**

**CODER HOUSE**



# Uso de Cliente GUI Robo 3T para MongoDB

Ejemplo  
en vivo



Robo 3T - 1.3

File View Options Window Help

Mi conexión local MongoDB (4)

- System
- config
- mibase
  - Collections (1)
    - usuarios
  - Functions
  - Users

Welcome x db.getCollection('usuarios').find... x

Mi conexión local MongoDB localhost:27017 mibase

```
db.getCollection('usuarios').find({})
```

usuarios 0.003 sec.

Key	Value	Type
(1) ObjectId("6037cda3c79e2378f6bf4c4b")	{ 3 fields }	Object
_id	ObjectId("6037cda3c79e2378f6bf4c4b")	ObjectId
nombre	Juan	String
edad	32.0	Double

Logs

**CODER HOUSE**

# Contenido de la carpeta de la base de datos

Ejemplo  
en vivo



miBaseMongo

Inicio Compartir Vista

Anclar al acceso rápido Copiar Pegar Cortar Copiar ruta de acceso Pegar acceso directo

Mover a Copiar a Eliminar Cambiar nombre Nueva carpeta Nuevo elemento Fácil acceso

Propiedades Abrir Historial

Seleccionar todo No seleccionar ninguno Invertir selección

Portapapeles Organizar Nuevo Abrir Seleccionar

Este equipo > Disco local (C:) > Cursos > Coderhouse > CursoBackend > Clase18 > miBaseMongo

Buscar en miBaseMongo

Nombre	Fecha de modifica...	Tipo	Tamaño
diagnostic.data	25/2/2021 14:00	Carpeta de archivos	
journal	25/2/2021 13:05	Carpeta de archivos	
_mdb_catalog.wt	25/2/2021 13:18	Archivo WT	36 KB
collection-0-8586343578430127360.wt	25/2/2021 13:06	Archivo WT	20 KB
collection-2-8586343578430127360.wt	25/2/2021 13:06	Archivo WT	20 KB
collection-4-8586343578430127360.wt	25/2/2021 14:00	Archivo WT	24 KB
collection-7-8586343578430127360.wt	25/2/2021 13:18	Archivo WT	20 KB
index-1-8586343578430127360.wt	25/2/2021 13:06	Archivo WT	20 KB
index-3-8586343578430127360.wt	25/2/2021 13:06	Archivo WT	20 KB
index-5-8586343578430127360.wt	25/2/2021 14:00	Archivo WT	24 KB
index-6-8586343578430127360.wt	25/2/2021 14:00	Archivo WT	24 KB
index-8-8586343578430127360.wt	25/2/2021 13:18	Archivo WT	20 KB
mongod.lock	25/2/2021 13:05	Archivo LOCK	1 KB
sizeStorer.wt	25/2/2021 14:00	Archivo WT	36 KB
storage.bson	25/2/2021 13:05	Archivo BSON	1 KB
WiredTiger	25/2/2021 13:05	Archivo	1 KB
WiredTiger.lock	25/2/2021 13:05	Archivo LOCK	0 KB
WiredTiger.turtle	25/2/2021 14:00	Archivo TURTLE	2 KB
WiredTiger.wt	25/2/2021 14:00	Archivo WT	76 KB
WiredTigerHS.wt	25/2/2021 13:05	Archivo WT	4 KB

20 elementos

**CODER HOUSE**



***¿Alguna pregunta hasta  
ahora?***



# ***SQL A MONGODB***

*Tiempo: 15 minutos*



A continuación se muestra un enunciado para trabajar con bases de datos SQL:

- 1) *Crear una base de datos llamada 'mibase'*
- 2) *Crear una tabla dentro de esa base con el nombre 'usuarios' que contenga los siguientes campos:*
  - *'nombre' del tipo varchar no nulo*
  - *'apellido' del tipo varchar no nulo*
  - *'edad' del tipo entero sin signo*
  - *'email' del tipo varchar no nulo*
  - *'id' clave primaria autoincremental no nula*
- 1) *Insertar estos 3 usuarios en esa tabla*
  - *Juan Perez edad 23 jp@gmail.com*
  - *Pedro Mei edad 21 pm@gmail.com*
  - *Juana Suarez edad 25 js@gmail.com*
- 1) *Listar las tablas existentes*
- 2) *Listar los usuarios agregados*



En base al enunciado:

- 1) Conformar las operaciones de consulta apropiadas para realizar los mismos procedimientos utilizando un cliente CLI de MongoDB.
- 2) Crear una conexión local hacia el motor de base de datos MongoDB mediante el cliente ROBO 3T
- 3) Con ROBO 3T acceder y visualizar la base, colecciones y documentos creados.

### **Notas:**

El campo id lo crea automáticamente el motor de base de datos MongoDB y se llama `_id` almacenando un `ObjectID`.

***¿PREGUNTAS?***



# ***¡MUCHAS GRACIAS!***

Resumen de lo visto en clase hoy:

- Bases de datos nosql
- Mongodb: Instalación y configuración
- mySQL vs MongoDB





***OPINA Y VALORA ESTA CLASE***

***#DEMOCRATIZANDOLAEDUCACIÓN***