

Bases de datos no SQL

MongoDB

Tema 5. Acceso a datos



Objetivos

- Características de las bases de datos no SQL
- Instalación de MongoDB.
- Uso básico de MongoDB con Python.

Bases de datos NoSQL

- Se puede hacer referencia a las bases de datos NoSQL indistintamente como "no relacionales", "bases de datos NoSQL" o "no SQL" para destacar el hecho de que pueden administrar altos volúmenes de datos no estructurados que cambian con rapidez de formas diferentes a una base de datos relacional (SQL) con filas y tablas.

Ventajas

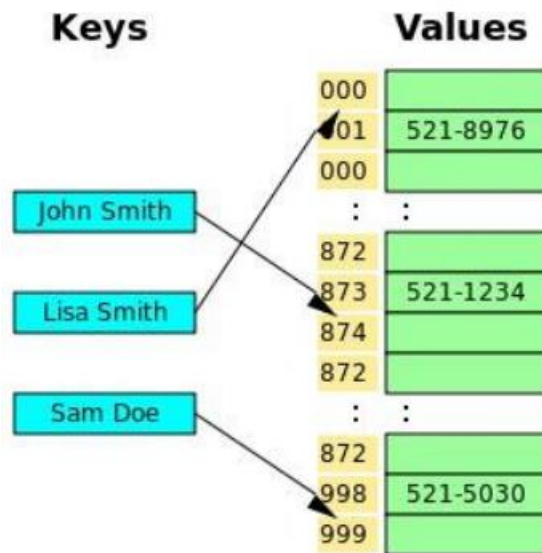
- Se ejecutan en máquinas con pocos recursos
- Escalabilidad horizontal
- Pueden manejar gran cantidad de datos
- No genera cuellos de botella

Diferencias con los SGBDR

- No utilizan SQL como lenguaje de consultas.
- No utilizan estructuras fijas como tablas para el almacenamiento de los datos.
- No suelen permitir operaciones JOIN.
- Arquitectura distribuida.

Tipos

Clave-valor



Basadas en documentos

```
{  
  "ID": 1,  
  "FIRST": "Frank",  
  "LAST": "Weigel",  
  "ZIP": "94040",  
  "CITY": "MV",  
  "STATE": "CA"  
}
```

JSON

=

User Info			
KEY	First	Last	ZIP_id
1	Frank	Weigel	2
2	Ali	Dodson	2
3	Mark	Azad	2
4	Steve	Yen	3

Ejemplos



Organización de la información

- Bases de datos
- Colecciones
- Documentos
- Campos

La información se almacena en formato JSON

JSON

JavaScript Object Notation

- Es un standard ligero para el intercambio de datos entre máquinas que sea fácilmente legible por personas.
- Actualmente se usa de forma intensiva en comunicaciones sobre HTTP (servicios web, apis,...) y prácticamente todos los lenguajes dan soporte.

JSON

JavaScript Object Notation

- Es un standard ligero para el intercambio de datos entre máquinas que sea fácilmente legible por personas.
- Actualmente se usa de forma intensiva en comunicaciones sobre HTTP (servicios web, apis,...) y prácticamente todos los lenguajes dan soporte.

JSON

Elementos de JSON

- Objetos { }
- Colecciones, listas, arrays []
- Valores de tipos primitivos
- Nombre de los campos

JSON

```
{  
  "título" : "El retorno de los dragones",  
  "autores" : [ "Tracy Hickman", "Margaret Weis" ],  
  "editorial" : {  
    "nombre" : "TimunMas",  
    "pais": "España"  
  }  
}
```

JSON

- Ejercicio práctico:
 - Queremos enviar la información generada en un TPV a un servidor central usando JSON.
 - Crea un documento JSON de ejemplo que sirva para enviar la información cada vez que se genere un ticket.

*** CENTROS COMERCIALES FOURCARRE ***		
L U C E N A		
CIF: A28425270		
Telf.Atención al Cliente: 902202000		

1	PATAT FREIR 4K CYO	3,55
L	SEMI COVAP 1L	
	6 x (0,85)	5,10
1	BARRA 80/120 BL C	6,70
1	FUNDA PLA.C/M 43X	11,80
1	BL.ALGOD.2,5MM 20M	3,50

	TOT COMPRA:	54,77

	PAGADO	54,77
	COMERCIO:	266015783
P.V.P.IVA INCLUIDO		
Para Devoluciones conserve ticket		
Plazo de Devolucion : 15 días		
Centros Comerciales FOURCARRE S.A		
C/Campezo 16-Poligono Las Mercedes		
28022 Madrid		

JSON

```
1 {  
2   "compra": [  
3     {  
4       "cantidad": 1,  
5       "artículo": "Patat Freir 4K",  
6       "total": 3.55  
7     },  
8     {  
9       "cantidad": 6,  
10      "artículo": "L Semi COVAP 1L",  
11      "precio": 0.85,  
12      "total": 5.1  
13    },  
14    {  
15      "cantidad": 1,  
16      "artículo": "Barra 80/120",  
17      "total": 6.7  
18    }  
19  ],  
20  "total": 54.77,  
21  "pagado": 54.77,  
22  "comercio": 266015783  
23 }
```

*** CENTROS COMERCIALES FOURCARRE ***
L U C E N A
CIF: A28425270
Telf.Atención al Cliente: 902202000

1	PATAT FREIR 4K CYO	3,55
L	SEMI COVAP 1L	
6 x (0,85)	5,10
1	BARRA 80/120 BL C	6,70
1	FUNDA PLA.C/M 43X	11,80
1	BL.ALGOD.2,5MM 20M	3,50

TOT COMPRA: 54,77

PAGADO 54,77

COMERCIO: 266015783

P.V.P.IVA INCLUIDO

Para Devoluciones conserve ticket
Plazo de Devolucion : 15 días

Centros Comerciales FOURCARRE S.A
C/Campezo 16-Poligono Las Mercedes
28022 Madrid

JSON

Fallos comunes

- Olvidar las comillas en los valores de tipo texto
- Olvidar : entre el nombre y el valor
- No incluir las comas
- No ser uniforme en los nombres
- Olvidar cerrar todos los bloques que se abran

Tipos de datos

- String o cadena
- Number o número
- Boolean o booleano (true o false)
- null
- Array

Instalar MongoDB

[Products](#)[Solutions](#)[Resources](#)[Company](#)[Pricing](#)[Sign In](#)[Try Free](#)**Atlas**

MongoDB as a service

**On-premises**

MongoDB locally

**Tools**

Boost productivity

**Mobile & Edge**

Realm Datastore

MongoDB Community Server

The Community version of our distributed database offers a flexible document data model along with support for ad-hoc queries, secondary indexing, and real-time aggregations to provide powerful ways to access and analyze your data.

The database is also offered as a fully-managed service with [MongoDB Atlas](#). Get access to advanced functionality such as auto-scaling, serverless instances (in preview), full-text search, and data distribution across regions and clouds. Deploy in minutes on AWS, Google Cloud, and/or Azure, with no downloads necessary.

[Give it a try with a free, highly-available 512 MB cluster.](#)

Available Downloads

Version

5.0.5 (current) ✓

Platform

Windows ✓

Package

msi ✓

**Download**[Copy Link](#)[Current releases & packages](#)

Ejemplo de consulta básica

```
{"author": "Francisco Romero"}
```

Operadores

$\$gt$
 $\$gte$
 $\$lt$
 $\$lte$
 $\$ne$

$\$in$
 $\$nin$
 $\$or$
 $\$and$
 $\$nor$
 $\$not$

Operadores básicos

Igualdad

`{"campo" : "valor",...}`

Comparación

`{"campo" : { $gt / $lt / $ne : "valor" } }`

Contenido en

`{"campo" : { $in : ["valor1","valor2","valor3"] } }`

And, Or y Not

And

{ \$and: [{<key1>:<value1>}, { <key2>:<value2>}] }

Or

{ \$or: [{<key1>:<value1>}, { <key2>:<value2>}] }

Not

{ \$not: [{<key1>:<value1>}, { <key2>:<value2>}] }

Consultas complejas

```
{"likes": {$gt:10}, $or: [{"author":  
    "Francisco Romero"},  
    {"title": "Intro a MongoDB"}]}
```

Ejemplo de agregación

The screenshot displays the MongoDB Aggregation Framework interface with two stages defined:

Stage 1: \$match

- Query:

```
1 /**  
2  * query: The query in MQL.  
3  */  
4 {  
5  
6    publisher: "Nintendo"  
7  
8  }
```
- Output: A sample of 20 documents showing a document for "Wii Sports" with NA_sales of 41.49.

Stage 2: \$count

- Query:

```
1 /**  
2  * Provide the field name for the count.  
3  */  
4 'Cantidad'
```
- Output: A sample of 1 document showing the total count: **Cantidad: 703**.

MongoDB con Python

```
Terminal: Local x +
(c) Microsoft Corporation. Todos los derechos reservados.

(venv) C:\Users\frome\PycharmProjects\blog>pip install pymongo
Collecting pymongo
  Downloading pymongo-4.0.1-cp39-cp39-win_amd64.whl (354 kB)
    |████████████████████████████████████████| 354 kB 2.2 MB/s
Installing collected packages: pymongo
Successfully installed pymongo-4.0.1

(venv) C:\Users\frome\PycharmProjects\blog>
```


MongoDB con Python

```
import pymongo

myclient = pymongo.MongoClient("mongodb://localhost:27017/")
mydb = myclient["mydatabase"]
mycol = mydb["customers"]

mydict = { "name": "John", "address": "Highway 37" }

x = mycol.insert_one(mydict)
```

Comparativa con SQL

SQL and Mongo DB Select Command

SQL SELECT Statement	MongoDB find() Statement
<pre>SELECT * FROM JavaTpoint</pre>	<pre>db.JavaTpoint.find()</pre>
<pre>SELECT id, user_id, status FROM JavaTpoint</pre>	<pre>db.JavaTpoint.find({ }, { user_id: 1, status: 1 })</pre>
<pre>SELECT user_id, status FROM JavaTpoint</pre>	<pre>db.JavaTpoint.find({ }, { user_id: 1, status: 1, _id: 0 })</pre>
<pre>SELECT * FROM JavaTpoint WHERE status = "B"</pre>	<pre>db.JavaTpoint.find({ status: "A" })</pre>

<https://www.javatpoint.com/sql-to-mongodb-mapping>

MongoDB con Python

```
myclient = pymongo.MongoClient("mongodb://localhost:27017/")
mydb = myclient["mydatabase"]
mycol = mydb["customers"]

myquery = { "address": { "$gt": "S" } }

mydoc = mycol.find(myquery)

for x in mydoc:
    print(x)
```

MongoDB Compass

MongoDB Compass - localhost:27017

Connect View Help

Local

4 DBS 2 COLLECTIONS

☆ FAVORITE

HOST
localhost:27017

CLUSTER
Standalone

EDITION
MongoDB 5.0.5 Community

Filter your data

- admin
- blog
- config
- local

+>_MONGOSH

Databases Performance

CREATE DATABASE

Database Name	Storage Size	Collections	Indexes
admin	20.5KB	0	1
blog	36.9KB	1	1
config	36.9KB	0	2
local	20.5KB	1	1

MongoDB Compass

MongoDB Compass - localhost:27017/blog.posts

Connect View Collection Help

Local

4 DBS 2 COLLECTIONS

☆ FAVORITE

HOST
localhost:27017

CLUSTER
Standalone

EDITION
MongoDB 5.0.5 Community

Filter your data

- admin
- blog
 - posts
- config
- local

blog.posts Documents

blog.posts

DOCUMENTS 13 TOTAL SIZE 1.0KB AVG. SIZE 80B INDEXES 1 TOTAL SIZE 36.9KB AVG. SIZE 36.9KB

Documents Aggregations Schema Explain Plan Indexes Validation

FILTER { field: 'value' } OPTIONS FIND RESET

ADD DATA VIEW

Displaying documents 1 - 13 of 13 REFRESH

	_id ObjectId	titulo String	contenido String	user_id Int32	
1	61c7758adae4e73e5fd4ef63	"John"	"Highway 37"	No field	
2	61c77594b86737512aa58fbf	"John"	"Highway 37"	No field	
3	61c77598144332dce8e38305	"John"	"Highway 37"	No field	
4	61c775d2b5157f839be72d0a	"John"	"Highway 37"	23	
5	61c775d6e687e61947cc2726	"John"	"Highway 37"	23	
6	61c775e5ea11e18ae6a6a745	"John"	"Highway 37"	22	
7	61c775eddf5745ba75805e2d	"John"	"Highway 37"	22	
8	61c776de5b51637828f97fd2	"John"	"Highway 37"	22	