



EJERCICIO CON MOTOR NO SQL



MongoDB®

Alumna Noemí pino

Ramo de Data Science y Big data

Profesor Rene Galarce

Viernes 10 de octubre de 2025

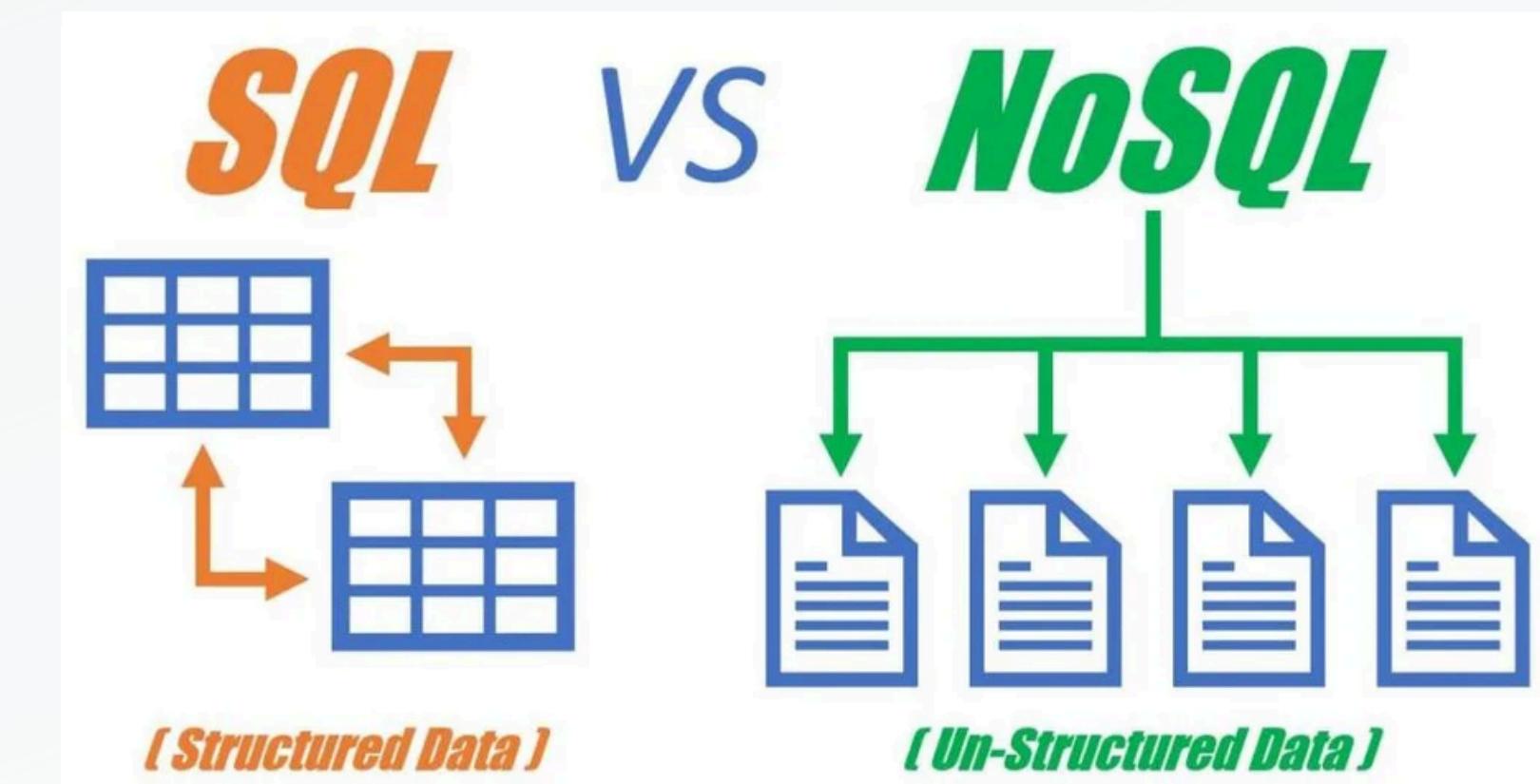
CONTENIDO

- 01** INTRODUCCIÓN
- 02** OBJETIVO
- 03** DISEÑO INICIAL
- 04** IMPLEMENTACIÓN DE INGESTA
- 05** CÓDIGO DE LA INGESTA
- 06** CONSULTAS
- 07** CONCLUSIONES DE VENTAJAS Y DESVENTAJAS

¿QUÉ ES MONGODB

MongoDB es un sistema de base de datos NoSQL, orientado a documentos y de código abierto.

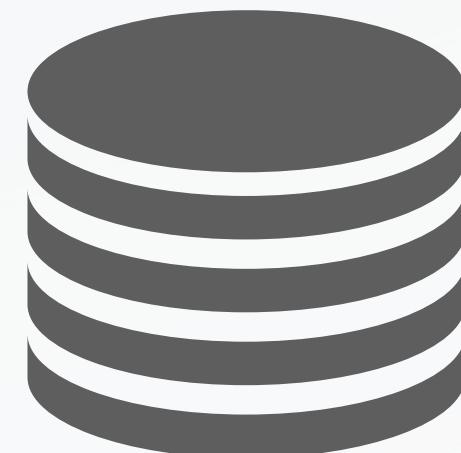
A diferencia de las bases de datos relacionales, no utiliza tablas con esquemas rígidos, sino que guarda datos en documentos JSON (o BSON, su versión binaria) dentro de colecciones, lo que le da más flexibilidad al modelar datos.



¿POR QUÉ MONGODB?

En general mongodb destaca sobre otros sistemas de datos cuando necesitas manejar datos que cambian con frecuencia, trabajar con estructuras flexibles, escalar horizontalmente (fragmentación) y mantener alta disponibilidad.

Esto lo hace ideal para aplicaciones modernas que manejan grandes volúmenes de información semiestructurada, desarrollo ágil con requisitos evolutivos, y entornos donde el rendimiento en operaciones de lectura y escritura es importante.



OBJETIVO

El objetivo de este laboratorio es implementar un sistema de base de datos NoSQL mediante la ingesta de datos desde una API pública y la ejecución de consultas útiles, demostrando las capacidades del modelo documental MongoDB para manejar datos semi-estructurados en escenarios posibles.

DISEÑO INICIAL

- Tipo de BD: modelo Documental
- Entidades almacenadas: documentos JSON
- Estructura por documento:

```
{  
  "id": 25,  
  "name": "pikachu",  
  "types": ["electric"],  
  "abilities": ["static", "lightning-rod"],  
  "stats": {  
    "hp": 35,  
    "attack": 55,  
    "defense": 40  
  },  
  "weight": 60,  
  "height": 4  
}
```

IMPLEMENTACIÓN DE INGESTA

- Fuente: PokeAPI (<https://pokeapi.co>)
- Datos obtenidos: solo algunos pokemons

PIPELINE:

OBTENCIÓN → TRANSFORMACIÓN → ALMACENAMIENTO



CÓDIGO DE INGESTA

```
#instalación de las librerías
%pip install pymongo requests
import requests #para hacer peticiones HTTP a la api
import pymongo #para interactuar con mongodb

#conectar a MongoDB
client = pymongo.MongoClient("mongodb://localhost:27017/") #conexión a la base de datos local en el puerto 27017

db = client["Pokemons"] #acceso a la base llamada "Pokemons"
collection = db["pokedex"] #acceso a la colección llamada "pokedex"

#lista y ingesta directa (obtiene los datos de cada pokemon desde la api y convierte la respuesta a json)
for nombre in ["pikachu", "charizard", "bulbasaur", "ditto", "mewtwo"]:
    datos = requests.get(f"https://pokeapi.co/api/v2/pokemon/{nombre}").json() #aca ponemos formato json
    collection.insert_one(datos)

#Contar total de documentos
total_documentos = collection.count_documents({})
print(f"total de documentos en la colección: {total_documentos}")

✓ 5.1s
```

PyMongo es un paquete de Python que permite conectarse y comunicarse con MongoDB.

las solicitudes request es un mensaje que un cliente (como un navegador) envía a un servidor para obtener datos o realizar una acción, utilizando un método como GET, POST, PUT o DELETE.

```
Requirement already satisfied: pymongo in c:\users\josue\appdata\local\programs\python\python39\lib\site-packages
Requirement already satisfied: requests in c:\users\josue\appdata\local\programs\python\python39\lib\site-packages
Requirement already satisfied: dnspython<3.0.0,>=1.16.0 in c:\users\josue\appdata\local\programs\python\python39\lib\site-packages
Requirement already satisfied: charset_normalizer<4,>=2 in c:\users\josue\appdata\local\programs\python\python39\lib\site-packages
Requirement already satisfied: idna<4,>=2.5 in c:\users\josue\appdata\local\programs\python\python39\lib\site-packages
Requirement already satisfied: urllib3<3,>=1.21.1 in c:\users\josue\appdata\local\programs\python\python39\lib\site-packages
Requirement already satisfied: certifi>=2017.4.17 in c:\users\josue\appdata\local\programs\python\python39\lib\site-packages
Note: you may need to restart the kernel to use updated packages
total de documentos en la colección: 5
```

CONSULTAS

CONSULTAS BÁSICAS DE FILTRADO

1

```
{"name": "pikachu"}
```

The screenshot shows a MongoDB query interface with the following details:

- Query bar: `{"name": "pikachu"}`
- Buttons: Explain, Reset, Find, Options
- Toolbar icons: Refresh, Save, Delete, Help
- Page controls: 25, 1-1 of 1, Previous, Next, List View, Grid View
- Result pane:

```
_id: ObjectId('68e80c4afe85f1a4e40601e6')
▶ abilities : Array (2)
  base_experience : 112
▶ cries : Object
▶ forms : Array (1)
▶ game_indices : Array (20)
  height : 4
▶ held_items : Array (2)
  id : 25
  is_default : true
  location_area_encounters : "https://pokeapi.co/api/v2/pokemon/25/encounters"
▶ moves : Array (105)
  name : "pikachu"
  order : 35
▶ past_abilities : Array (1)
▶ past_types : Array (empty)
▶ species : Object
▶ sprites : Object
▶ stats : Array (6)
▶ types : Array (1)
  weight : 60
```

CONSULTAS

CONSULTAS BÁSICAS DE FILTRADO

2

```
{"weight": {"$gte": 50, "$lte": 100}}
```

The screenshot shows a MongoDB query interface with the following details:

- Documents:** 0
- Aggregations:** 1
- Schema:** Indexes: 1
- Validation:**

Query: `{"weight": {"$gte": 50, "$lte": 100}}`

Buttons: Explain, Reset, Find, Options ▾

Results:

- 25 1 - 2 of 2
- Document 1:
 - `_id: ObjectId('68e80c4cf85f1a4e40601e8')`
 - `abilities: Array (2)`
 - `base_experience: 64`
 - `cries: Object`
 - `forms: Array (1)`
 - `game_indices: Array (20)`
 - `height: 7`
 - `held_items: Array (empty)`
 - `id: 1`
 - `is_default: true`
 - `location_area_encounters: "https://pokeapi.co/api/v2/pokemon/1/encounters"`
 - `moves: Array (86)`
 - `name: "bulbasaur"`
 - `order: 1`
 - `past_abilities: Array (1)`
 - `past_types: Array (empty)`
 - `species: Object`
 - `sprites: Object`
 - `stats: Array (6)`
 - `types: Array (2)`
 - `weight: 69`
- Document 2 (partial):
 - `species: Object`
 - `sprites: Object`
 - `stats: Array (6)`
 - `types: Array (1)`
 - `weight: 60`

CONSULTAS

CONSULTAS BÁSICAS DE FILTRADO

3

```
{"types.type.name": "electric"}
```

The screenshot shows a MongoDB query interface with the following details:

- Query: `{"types.type.name": "electric"}`
- Buttons: Explain, Reset, Find, Options
- Count: 25, 1-1 of 1
- Document Preview:

```
_id: ObjectId('68e80c4afe85f1a4e40601e6')
▶ abilities : Array (2)
  base_experience : 112
▶ cries : Object
▶ forms : Array (1)
▶ game_indices : Array (20)
  height : 4
▶ held_items : Array (2)
  id : 25
  is_default : true
  location_area_encounters : "https://pokeapi.co/api/v2/pokemon/25/encounters"
▶ moves : Array (105)
  name : "pikachu"
  order : 35
▶ past_abilities : Array (1)
▶ past_types : Array (empty)
▶ species : Object
▶ sprites : Object
▶ stats : Array (6)
  ▶ types : Array (1)
    ▶ 0: Object
      slot : 1
      ▶ type : Object
        name : "electric"
        url : "https://pokeapi.co/api/v2/type/13/"
    weight : 60
```

CONSULTAS

CONSULTAS CON OPERADORES LÓGICOS

4

```
{"types.type.name": {"$in": ["fire", "water", "grass"]}}
```

The screenshot shows a MongoDB query interface with the following details:

- Documents:** 0
- Aggregations:** 1
- Schema:** Indexes: 1
- Validation:**
- Query:** {"\$in": ["fire", "water", "grass"]}
- Buttons:** Explain, Reset, Find, Options
- Results:** 25 documents found, pages 1-2 of 2. The results are displayed as JSON documents.

Two results are shown:

```
▶ meta_items : Array (empty)
  id : 6
  is_default : true
  location_area_encounters : "https://pokeapi.co/api/v2/pokemon/6/encounters"
  ▶ moves : Array (131)
    name : "charizard"
    order : 7
  ▶ past_abilities : Array (1)
  ▶ past_types : Array (empty)
  ▶ species : Object
  ▶ sprites : Object
  ▶ stats : Array (6)
  ▶ types : Array (2)
  weight : 905

  _id: ObjectId('68e80c4cf85f1a4e40601e8')
  ▶ abilities : Array (2)
  base_experience : 64
  ▶ cries : Object
  ▶ forms : Array (1)
  ▶ game_indices : Array (20)
  height : 7
  ▶ held_items : Array (empty)
  id : 1
  is_default : true
  location_area_encounters : "https://pokeapi.co/api/v2/pokemon/1/encounters"
  ▶ moves : Array (86)
    name : "bulbasaur"
    order : 1
```

CONSULTAS

CONSULTAS CON OPERADORES LÓGICOS

5

```
{"types.type.name": {"$nin":  
  ["normal", "flying"]}}
```

The screenshot shows a MongoDB query interface with the following details:

- Documents:** 0
- Aggregations:** 1
- Schema:** Indexes
- Validation:** 1

The search bar contains the query: `type: {"$nin": ["normal", "flying"]}`. The results pane displays two documents, both of which are `Bulbasaur`:

```
location_area_encounters : "https://pokeapi.co/api/v2/pokemon/1/encounters"  
  moves : Array (86)  
    name : "bulbasaur"  
    order : 1  
  past_abilities : Array (1)  
  past_types : Array (empty)  
  species : Object  
  sprites : Object  
  stats : Array (6)  
  types : Array (2)  
  weight : 69  
  
_id: ObjectId('68e80c4dfe85f1a4e40601ea')  
  abilities : Array (2)  
  base_experience : 306  
  cries : Object  
  forms : Array (1)  
  game_indices : Array (20)  
  height : 20  
  held_items : Array (empty)  
  id : 150  
  is_default : true  
  location_area_encounters : "https://pokeapi.co/api/v2/pokemon/150/encounters"  
  moves : Array (167)  
    name : "mewtwo"  
    order : 245  
  past_abilities : Array (1)  
  past_types : Array (empty)
```

CONSULTAS

CONSULTA CON PROYECCIÓN Y ORDENAMIENTO

6

```
{"name": 1, "types.type.name": 1, "weight": 1, "_id": 0}
```

The screenshot shows a MongoDB query interface with the following configuration:

- Projection:** {"name": 1, "types.type.name": 1, "weight": 1, "_id": 0}
- Sort:** { field: -1 } or [['field', -1]] (Max Time MS: 60000)
- Collation:** { locale: 'Skip' }, Limit: 0
- Index Hint:** { field: -1 }

The results section displays five documents:

- Document 1:

```
  name : "pikachu"
  types : Array (1)
    0: Object
      type : Object
        name : "electric"
      weight : 60
```
- Document 2:

```
  name : "charizard"
  types : Array (2)
  weight : 905
```
- Document 3:

```
  name : "bulbasaur"
  types : Array (2)
  weight : 69
```
- Document 4:

```
  name : "ditto"
```
- Document 5:

```
  name : "squirtle"
  types : Array (2)
  weight : 65
```

Page navigation: 25 | 1-5 of 5

CONSULTAS

CONSULTAS CON EXPRESIONES REGULARES

7

```
{"name": {"$regex": "chu",  
"$options": "i"}}
```

The screenshot shows a MongoDB query interface with the following details:

- Documents:** 0
- Aggregations:** 1
- Schema:** Indexes: 1
- Validation:**
- Search Query:** `{"$regex": "chu", "$options": "i"}`
- Buttons:** Explain, Reset, Find, Options
- Document Preview:** A single document is shown, identified by ID `_id: ObjectId('68e80c4afe85f1a4e40601e6')`. The document structure includes:
 - `abilities`: Array (2)
 - `base_experience`: 112
 - `cries`: Object
 - `forms`: Array (1)
 - `game_indices`: Array (20)
 - `height`: 4
 - `held_items`: Array (2)
 - `id`: 25
 - `is_default`: true
 - `location_area_encounters`: ["https://pokeapi.co/api/v2/pokemon/25/encounters"](https://pokeapi.co/api/v2/pokemon/25/encounters)
 - `moves`: Array (105)
 - `name`: "pikachu"
 - `order`: 35
 - `past_abilities`: Array (1)
 - `past_types`: Array (empty)
 - `species`: Object
 - `sprites`: Object
 - `stats`: Array (6)
 - `types`: Array (1)
 - `weight`: 60

CONSULTAS

CONSULTAS CON EXPRESIONES REGULARES

8

{"abilities.is_hidden": true}

The screenshot shows a MongoDB query interface with the following details:

- Documents:** 0
- Aggregations:** 1
- Schema:** Indexes
- Indexes:** 1
- Validation:**

Query: {"abilities.is_hidden": true}

Results (25 total, 1-5 of 5 shown):

```
is_default: true
location_area_encounters: "https://pokeapi.co/api/v2/pokemon/25/encounters"
> moves: Array (105)
  name: "pikachu"
  order: 35
> past_abilities: Array (1)
> past_types: Array (empty)
> species: Object
> sprites: Object
> stats: Array (6)
> types: Array (1)
  weight: 60

_id: ObjectId('68e80c4bfe85f1a4e40601e7')
> abilities: Array (2)
  base_experience: 240
> cries: Object
> forms: Array (1)
> game_indices: Array (20)
  height: 17
> held_items: Array (empty)
  id: 6
  is_default: true
  location_area_encounters: "https://pokeapi.co/api/v2/pokemon/6/encounters"
> moves: Array (131)
  name: "charizard"
  order: 7
> past_abilities: Array (1)
```

CONSULTAS

CONSULTAS CON EXPRESIONES REGULARES

9

```
{"name": {"$regex": "saur",  
"$options": "i"}}
```

The screenshot shows a MongoDB query interface with the following details:

- Documents:** 0
- Aggregations:** 1
- Schema:** Indexes: 1
- Validation:**

The search query is: `{} {"$regex": "saur", "$options": "i"} {}`. The results show one document:

```
_id: ObjectId('68e80c4cfe85f1a4e40601e8')
  abilities : Array (2)
  base_experience : 64
  cries : Object
  forms : Array (1)
  game_indices : Array (20)
  height : 7
  held_items : Array (empty)
  id : 1
  is_default : true
  location_area_encounters : "https://pokeapi.co/api/v2/pokemon/1/encounters"
  moves : Array (86)
  name : "bulbasaur"
  order : 1
  past_abilities : Array (1)
  past_types : Array (empty)
  species : Object
  sprites : Object
  stats : Array (6)
  types : Array (2)
  weight : 69
```

CONSULTAS

CONSULTA MÁS ESPECIFICA

10

```
{"stats.base_stat": {"$gt":  
100}}
```

The screenshot shows a MongoDB query interface with the following details:

- Documents:** 0 (Aggregations: 1)
- Query:** {"stats.base_stat": {"\$gt": 100}}
- Buttons:** Explain, Reset, Find (highlighted), Options ▾
- Results:** 25 documents, 1-2 of 2
- Document 1 (Charizard):**
 - _id: ObjectId('68e80c4dfe85f1a4e40601ea')
 - abilities: Array (2)
 - base_experience: 306
 - cries: Object
 - forms: Array (1)
 - game_indices: Array (20)
 - height: 20
 - held_items: Array (empty)
 - id: 150
 - is_default: true
 - location_area_encounters: "https://pokeapi.co/api/v2/pokemon/150/encounters"
 - moves: Array (167)
 - name: "mewtwo"
 - order: 245
 - past_abilities: Array (1)
 - past_types: Array (empty)
 - species: Object
- Document 2 (Charizard):**
 - _id: ObjectId('68e80c4dfe85f1a4e40601ea')
 - abilities: Array (2)
 - base_experience: 306
 - cries: Object
 - forms: Array (1)
 - game_indices: Array (20)
 - height: 20
 - held_items: Array (empty)
 - id: 150
 - is_default: true
 - location_area_encounters: "https://pokeapi.co/api/v2/pokemon/150/encounters"
 - moves: Array (167)
 - name: "charizard"
 - order: 7
 - past_abilities: Array (1)
 - past_types: Array (empty)
 - species: Object

VENTAJAS/DESVENTAJAS

VENTAJAS

- Dado a su naturales de estructura variables, cada documento puede tener campos diferentes, lo cual lo hace versátil.
- Se pueden agregar nuevos campos sin migraciones complejas
- Ideal para datos de APIs y aplicaciones modernas
- Alta velocidad en operaciones de lectura/escritura
- Interfaz visual intuitiva con colores y diseño claro

DESVENTAJAS

- Documentos muy anidados requieren más esfuerzo para interpretar
- Mayor tiempo para dominar operadores y estructura
- Mayor curva de aprendizaje para usuarios nuevos (empinada)

CONCLUSIÓN

Aprendimos que MongoDB es ideal para datos flexibles y de rápido cambio, donde la estructura evoluciona constantemente. Su modelo documental permite trabajar con información compleja y anidada de forma natural, aunque requiere mayor cuidado en el diseño de esquemas y consultas. La elección entre MongoDB y bases relacionales depende del balance entre flexibilidad y estructura que cada proyecto necesite.

**GRACIAS
A TODOS**

